ELSEVIER

24 August 1998

PHYSICS LETTERS A

Physics Letters A 245 (1998) 399–406

Optimal targeting of chaos

Erik M. Bollt [a], Eric J. Kostelich [b]

[a] *Mathematics Department, 572 Holloway Rd., US Naval Academy, Annapolis, MD 21402-5002, USA*
[b] *Department of Mathematics, Arizona State University, Tempe, AZ 85287-1804, USA*

Received 16 February 1998; accepted for publication 31 March 1998
Communicated by C.R. Doering

Abstract

Standard graph theoretic algorithms are applied to chaotic dynamical systems to identify orbits that are optimal relative to a prespecified cost function. We reduce the targeting problem to the problem of finding optimal paths through a graph. Numerical experiments on one-dimensional maps suggest that periodic saddle orbits of low period are typically less expensive to target (relative to a family of smooth cost functions) than periodic saddle orbits of high period. © 1998 Elsevier Science B.V.

PACS: 05.45.+b

1. Introduction

The "control of chaos" refers to a procedure wherein one applies a sequence of small perturbations to confine a chaotic trajectory to a small neighborhood of a given saddle periodic orbit embedded in the attractor [1]. "Targeting" refers to a process wherein one uses a suitable sequence of small perturbations to steer an initial condition, on an attractor, to a neighborhood of a prespecified point (target) as quickly as possible [2]. Targeting algorithms can be used to rapidly switch a chaotic process between different periodic orbits [3].

A variety of different performance objectives may be important in applications of control and targeting. Consider a discrete dynamical system that is a Poincaré map of a flow. Successive iterates on the surface of section may correspond to widely varying times of flight. A targeting algorithm that reaches a given point on the surface of section in a minimum number of iterates typically does not correspond to the minimum time of flight in the original flow. Thus, it may be desirable to define a "cost" or "performance" function of points on the surface of section and choose a targeting path that minimizes the total cost of reaching the target (or alternatively maximizes performance).

Targeting a spacecraft from the earth to the moon is an example where a minimum-time orbit might be desirable. Alternative performance goals might be to minimize overall fuel consumption or to avoid certain regions of space.

In this Letter, we address two distinct, but related, questions. (1) Targeting algorithms such as those discussed in Ref. [2] may yield many orbits that lead to a neighborhood of a given target point. Which orbits have the lowest cost? (2) It is possible to use targeting algorithms to steer from one periodic point to another. On the average, do orbits leading to a periodic point of low period have a lower cost per step compared to

PII S0375-9601(98)00270-9

orbits leading to periodic points of high period?

2. Finding optimal orbits

We use well-established algorithms from graph theory to minimize the total cost of targeting from one point to a neighborhood of another point. These algorithms apply in a natural way to transition diagram descriptions of dynamical systems [4,5]. The breadth-first search (BFS) algorithm finds the shortest path through an unweighted directed graph, and Dijkstra's algorithm finds the optimal path through a directed graph with positive weights [6]. Our results suggest that transition diagrams can be well approximated by direct observation of a sufficiently long chaotic orbit. As we now show, the targeting problem can be traded for the problem of finding optimal paths through a graph, which in turn are found by generating spanning-trees through the graph. The advantage of this trade is that path searching is a mature topic in graph theory [6] and in computer science; there are algorithms to find paths that satisfy a variety of different restrictions.

The generating symbol partition of a dynamical system allows the use of symbol dynamics to represent the system in the equivalent but abstract setting of a Bernoulli shift map on a symbol space; the orbit of each initial condition in the phase space has a corresponding infinite symbol sequence. (The sequence is bi-infinite if the map is invertible.) Given the generating partition, symbol dynamics lends itself particularly well to representation by a directed graph [4,5,7,8]. Often, however, the generating partition is difficult to find: no general method is known for more than two dimensions, and even in two dimensions, the methods must be tailored to specific maps [5]. While the Markov partition is important for applications such as computations of entropy, Lyapunov exponents, and the like, it is not needed for path searching applications. This is fortunate, because an arbitrary grid is not expected to be Markov.

When symbol dynamics are impractical, we propose the use of a simplistic, but serviceable, grid-based method. Suppose we require an optimal orbit from A to B using perturbations that move points by distances less than ϵ in phase space. Given a map F_λ parametrized by λ, this constraint requires

$$\epsilon \approx \frac{\partial F_\lambda}{\partial \lambda}\lambda_{\max}. \tag{1}$$

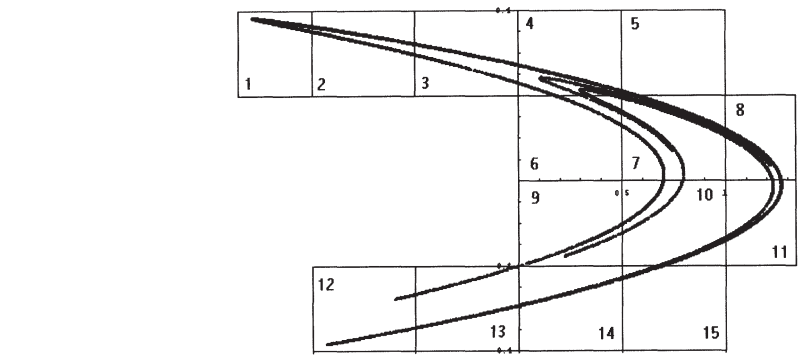
Fig. 1a illustrates an ϵ-grid covering the Hénon attractor, generated by the map $(x_{n+1}, y_{n+1}) = F(x_n, y_n) = (1 - \alpha x_n^2 + y_n, \beta x_n)$. For clarity of the illustration, we have chosen the crude grid $(\epsilon_x, \epsilon_y) = (0.5, 0.2)$. Given an orbit of length M, we determine the number N of occupied ϵ-boxes and identify the occupied boxes with a directed graph of N nodes. If the point (x_i, y_i) lies in box j and (x_{i+1}, y_{i+1}) lies in box k, then we draw an arrow from node j to node k in the corresponding graph; see Fig. 1b. This graph representation of the discretized global action of the map on the grid, while not Markov, is a surjection from the phase space onto the graph of *observed* transitions; each graph path has a corresponding observed ϵ-chain pseudo-orbit.

For a fixed ϵ-grid, the accuracy of the observed transition diagram improves as the test orbit gets longer. For a fixed-length test orbit, the accuracy deteriorates as ϵ decreases, because some boxes are missed. For good accuracy, the orbit must be long enough so that there are many ϵ-recurrences, which requires $N \ll M$.

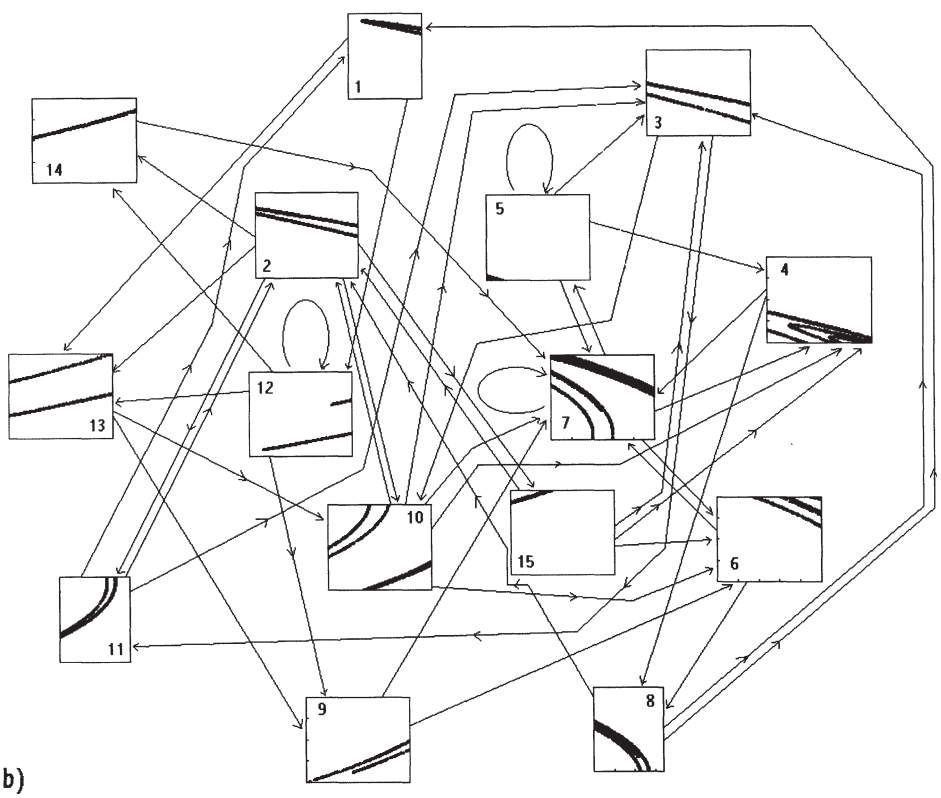
Suppose that the attractor is contained in the region $[x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}]$. We assign a number to each box in the ϵ-grid. By allocating a new node in the graph whenever an iterate first visits a new box, no storage is required for unoccupied boxes. The memory requirements are proportional to the box dimension [9] of the attractor, which may be considerably less than the dimension of the phase space. The time required to construct the transition diagram from a test orbit of length M is $O(M^2)$.

Any sufficiently long chaotic test orbit can be represented as a directed graph, including embeddings of time series from laboratory experiments. We have applied the method to numerically generated orbits from the logistic map, the standard map (which is difficult to target [10,11]), and a Poincaré map derived from the flow of the Lorenz equations. The method also works for higher dimensional dynamical systems, allowing only for the memory requirements of covering an attractor of (possibly large) box dimension with a (possibly small) ϵ-grid [12].

Once an orbit has been interpreted as a directed graph, a path through the graph can be realized by targeting iterates of the orbit that correspond to nodes



a)



b)

Fig. 1. (a) The Hénon attractor with $\alpha = 1.4$, $\beta = 0.3$, and an $(\epsilon_x, \epsilon_y) = (0.5, 0.2)$ grid covering. We enumerate and show only the grid squares that are occupied by an iterate of the map. (b) The observed action of the Hénon map on the grid; an arrow is drawn from node j to node k whenever an iterate goes from box j to box k on the attractor. The edges may have equal weights or may carry weights according to a nonnegative cost function.

in the desired path sequence. Given an approximate orbit (ϵ-chain), one can numerically "shoot" at the stable manifolds of the targeted sequence of points using small variations of an adjustable map parameter [10,13]. Hence, the targeting problem reduces to the problem of finding optimal paths through a graph.

The BFS algorithm finds the shortest path from node A (corresponding to the box containing the initial condition) to node B in an unweighted directed graph [6]. Starting at node A, we inspect each adjacent node, that is, the nodes that can be reached in one step. From these nodes, we check all adjacent nodes that have not yet been visited; these nodes can be reached from A in exactly two steps. We continue until we find the target node among the nodes reachable in exactly k steps.

The computer implementation maintains two lists. The second list contains the nodes that have never been visited (initially, all nodes except A), and the first list contains the nodes that have been reached in k or fewer steps (initially, only node A). When a node is added to the first list, it is removed from the second list. Hence, the number of steps through the graph is bounded by the number of nodes, N. Starting at A, the program can find the distance (number of steps) to every node in the graph in time complexity $O(E)$, where E is the number of edges in the graph. To find the path, we maintain the first list as a $3 \times N$ array, where the first entry is an identifier for the node, the second entry is the number of steps to reach the node, and the third entry identifies the predecessor of the node. Given the graph's connectivity, we can backtrack through the array to find the shortest path from node A to node B.

When the unweighted graph exactly corresponds to the symbolic dynamics of the process, the shortest path represents the optimal orbit (in a minimum iterate sense) from the initial condition to an ϵ-neighborhood of the target. The value of ϵ, which determines the precision of the graph, depends on the allowable size of the parameter perturbations; see Eq. (1).

The efficacy of this approach depends on how well the graph approximates the symbolic dynamics of the map. Typically, the existence of a short path between two nodes implies that it is possible to reach a neighborhood of a target point relatively quickly from a given initial condition. For example, in Fig. 1b, there is an arrow from node 13 to node 9. Hence, it may be possible to reach an ϵ-neighborhood of a target point in box 9 from initial conditions in box 13 in one iterate; the size of the required parameter perturbations is determined by Eq. (1). On the other hand, if the test orbit were sufficiently short, the transition from node 13 to node 9 might not be observed. In this case, the shortest path goes through nodes 10, 6, 8, 1, 12, and then 9.

Table 1
Lengths of shortest paths from fixed starting and target nodes on directed graphs constructed from test orbits of length M from the Hénon map with $\alpha = 1.4$, $\beta = 0.3$. The ϵ-chain pseudo-orbit has an error bounded by, the ϵ of the grid of fixed precision ($\epsilon \approx 0.005$) in the region $[-1.8, 1.8] \times [-1.8, 1.8]$

M	2^{11}	2^{12}	2^{13}	2^{14}	2^{15}	2^{16}	2^{17}
Path length	63	45	28	25	16	16	16

Table 1 shows the length of the shortest paths from a fixed starting node and ending node, using BFS applied to graphs derived from an observed trajectory of the Hénon map. The directed graphs follow observed test orbits of length M through a grid of fixed precision ($\epsilon \approx 0.005$) in the region $[-1.8, 1.8] \times [-1.8, 1.8]$. When the test orbit is sufficiently short (say $M < 2^{15}$), the fastest observed paths are suboptimal because some transitions between grid boxes are not observed. As M is increased, more transitions are observed, and the BFS paths become shorter. Eventually, the length of the optimal paths approaches a constant value (here 16), because no new transitions are observed with further increases in M. This result is typical for a variety of parameter values, grid sizes, and maps, including the standard map and the logistic map [12].

Dijkstra's algorithm finds a cost-optimal path through a directed graph whose arcs have arbitrary positive weights. If two different paths have exactly equal weights, then Dijkstra's algorithm selects the tree-wise leftmost equivalent path. In any case, the surjection yields an optimal ϵ-pseudo orbit that is no more costly than any other ϵ-pseudo orbit of the map. A detailed description of the algorithm can be found for example in Ref. [6]. The time complexity of Dijkstra's algorithm is $O(N^2)$, where N is the number of nodes in the graph. This time requirement is small compared to the $O(M^2)$ time needed to construct the graph from a test orbit of length M.

A natural choice for a positive cost function $F(x)$

is the time of flight from x to its next return to the surface of section for a Poincaré return map. The weight from node A to node B in the corresponding directed graph might be the average of F over all points in the ϵ-neighborhood identified with A that move in one iterate of the Poincaré map to the ϵ-neighborhood identified with B. Dijkstra's algorithm can be used to find an orbit on the Poincaré section that corresponds to a very fast orbit in the original flow.

Weighted graphs and targeting algorithms can yield a control strategy that allows one to avoid prespecified regions of a chaotic attractor. For example, given a directed graph constructed from a test orbit on the full attractor, nodes corresponding to undesirable regions on the attractor can be eliminated, either manually [14] or by assigning an infinite weight to the incoming edges. In either case, Dijkstra's algorithm avoids paths containing the undesirable nodes.

We close this section with an observation that relates two equivalent viewpoints found in dynamical systems theory and in graph theory. A principal feature that makes path searching problems through a graph of N nodes computationally simple is the fact that, when building spanning trees, at each step, the next (arc) closest node is selected from a decreasing list of nodes that have not yet been visited. Therefore, the longest possible path and the longest possible search are bounded by a path that visits all N nodes. Translated to the language of dynamical systems, one's ability to cover the attractor with the ϵ-grid requires a compact attractor. The pigeonhole principle can be used to show that an ϵ-recurrence must occur in a time bounded above by N iterations. (This is a form of the Poincaré recurrence theorem [10].) If we consider the graph as generating a finite-type subshift symbol space on N symbols with the discrete topology, then in that setting, path searching is equivalent to finding grammatically legal words (a sequence of symbols) joining two symbols (nodes). When this symbolic representation is derived from the map by a generating partition, then the representation is equivalent by semi-conjugacy. (See, for example, Ref. [8].)

Finally, we note that Hsu [18] has studied "cell-to-cell" mapping methods for characterizing a dynamical system on a grid, with much attention to refinement, where needed, to capture fine details. Cell-to-cell mapping considers the orbit of a grid square to be determined by the orbit of the point at the square's center. In contrast, we determine the orbit possibilities of a grid square by all iterates of a test orbit that ergodically wander into the square; hence, we consider branching in a more natural way. To our knowledge, there has been no attempt to use cell-to-cell mapping as a vehicle for path searches.

3. The costs of targeting periodic orbits

So far we have considered the question of how to identify one or more orbits to a target that are optimal with respect to a given criterion. In this section, we introduce an initial numerical study of the optimality of families of targeted orbits. We are interested in the question of whether periodic orbits of low period are less expensive to target, on the average, than those of high period, given a prespecified "cost" or performance function.

An important question in the study of dynamical processes is the long-time average value of a smooth function F over a given orbit $\{x_i\}$, which for a discrete dynamical system is

$$\langle F\rangle = \lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} F(x_i).$$

Hunt and Ott [15] have investigated chaotic processes where "typical" orbits (with respect to the Lebesgue measure of initial conditions in the phase space) have well-defined long-time averages. Saddle periodic orbits embedded in the attractor are not typical in this sense, and they may yield values of $\langle F\rangle$ that are different from those of the typical (aperiodic) orbits within the attractor.

The function F also may be regarded as a "cost" or performance function. Hunt and Ott examined a large number of saddle periodic orbits embedded in the attractors of some low-dimensional dynamical systems and considered a large family of performance functions, F_γ, parametrized by γ. They found that for most choices of γ, the value of $\langle F_\gamma\rangle$ was more likely to be larger on periodic orbits of low period than the value of $\langle F_\gamma\rangle$ on orbits of high period. Thus, if one regards F_γ as a measure of system performance, periodic orbits of low period are more likely to be the "optimal" periodic orbits in a chaotic attractor.

As discussed in the Introduction, there are many possible performance measures that one may wish to optimize in a given chaotic process. Hunt and Ott found that $\langle F \rangle$ is often largest when evaluated on periodic orbits of low period. We are interested in the values of $\langle F \rangle$ that are obtained when the average is computed over a large ensemble of targeted orbits that reach a small neighborhood of a periodic orbit. We now consider some numerical experiments that address the question of whether periodic orbits of low period are less expensive to target, on the average, than those of high period, for a given family of performance functions.

As a first example, we consider the tent map

$$T(x_n) = x_{n+1} = 2x_n, \qquad 0 \leqslant x_n \leqslant 1/2,$$
$$= 2 - 2x_n, \quad 1/2 < x_n \leqslant 1. \tag{2}$$

Let y be a periodic point of T, and suppose that z_0 is eventually periodic to y in k iterates; that is, $T^k(z_0) = y$ but $T^j(z_0) \notin \text{orbit}(y)$ for $0 \leqslant j < k$. We define the *average targeting cost per step to reach y from* z_0 as

$$\frac{1}{k} \sum_{j=1}^{k} F_\gamma \left(T^j(z_0) \right),$$

where F_γ is one of a family of smooth performance functions parametrized by γ. (Here T^k denotes the tent map composed with itself k times.)

The average targeting cost per step depends on the particular periodic orbit y and the initial condition z_0. For a given periodic orbit, there are arbitrarily many eventually periodic points, as k may be any positive integer. In most applications, however, one wants to reach a given target point relatively quickly, so we are interested primarily in cases where k is relatively small. In the numerical results presented below, we have taken $k = 7$, but there is nothing special about this choice; we have found similar results for other small values of k.

Having fixed k, we now consider all the period-p points of the tent map for a given p. (We regard y as a period-p point if $T^p(y) = y$ but $T^j(y) \neq y$ for $0 < j < p$.) Let $S_p(k)$ be the set of all points z_0 such that z_0 is eventually periodic to a period-p point of the map T in k steps. The set $S_p(k)$ represents all the possible "perfect" targeting orbits, as each one lands directly on a period-p target in k steps.

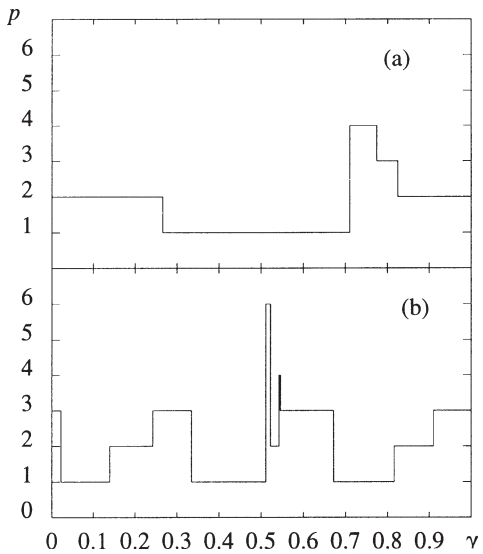


Fig. 2. Period of the orbit with minimal average targeting cost per step, as a function of γ, which parametrizes the family of performance functions. The average targeting cost per step is computed over all orbits that are eventually periodic to a period-p point in 7 steps for the tent map. (a) Results using the cost function $F_\gamma(x) = \cos 2\pi(x-\gamma)$; (b) results using the cost function $F_\gamma(x) = \cos 2\pi(x-\gamma) + \sin 6\pi(x-\gamma)$.

We define the *average targeting cost per step to reach a period-p point in k steps* as

$$C_\gamma(p) = \frac{1}{kN} \sum_{z_0 \in S_p(k)} \sum_{j=1}^{k} F_\gamma \left(T^j(z_0) \right), \tag{3}$$

where N is the number of points in $S_p(k)$. The quantity $C_\gamma(p)$ is simply the value of $\langle F_\gamma \rangle$, where the average is computed over the set of all eventually periodic orbits that lead to a period-p point in k steps. The set $S_p(k)$ is straightforward to generate using the symbol sequence associated with each orbit of the tent map.

We are interested to know how the average targeting cost per step for periodic points depends on the performance function F_γ and the period p when k is fixed (as mentioned above, we take $k = 7$). For each period $p \leqslant 14$, we evaluate $C_\gamma(p)$ for 10^5 uniformly spaced values of γ between 0 and 1, using the performance function $F_\gamma(x) = \cos 2\pi(x-\gamma)$. This family of functions is the same used by Hunt and Ott [15]; the cosine simply provides a convenient, smooth family.

Fig. 2a shows, as a function of γ, the period p that minimizes the value of $C_\gamma(p)$. For this family of performance functions, the numerical experiment suggests that the low-period orbits have the lowest average targeting cost per step. Fig. 2b shows analogous results for the family of performance functions $F_\gamma(x) = \cos 2\pi(x-\gamma) + \sin 6\pi(x-\gamma)$.

Given an arbitrary map, it is not convenient to characterize all of the periodic orbits and eventually periodic orbits using symbolic sequences alone, because one does not usually have a formula for the n-bit word intervals. Instead, we use a targeting algorithm to get within a small neighborhood of a target point in a certain number of steps. We now ask whether results are similar to the tent map example, Eq. (2).

Consider the quadratic map, $Q_a(x_n) = x_{n+1} = ax_n(1-x_n)$. If $a = 3.72$, then the map $Q_{3.72}$ appears to have chaotic orbits, but it is not conjugate to the tent map. We can locate all the periodic points for $p \leqslant 9$ using a careful numerical search. (There are no orbits of period 3 or 5 for $a = 3.72$.) We then target each periodic point, starting from random initial conditions, using the algorithm of Shinbrot et al. [16].

More precisely, given a periodic point as the target, we start from a random initial condition in the unit interval and seek a sequence of parameters in the interval $[3.69, 3.75]$ to steer the orbit to within 10^{-4} of the target point in 7 steps. That is, given the initial condition x_0, we attempt to find a sequence of parameters $a_0, a_1, \ldots, a_6$ in $[3.69, 3.75]$ such that the sequence $x_1 = Q_{a_0}(x_0)$, $x_2 = Q_{a_1}(x_1)$, $\ldots$, $x_7 = Q_{a_6}(x_6)$ is an orbit such that x_7 lies within 10^{-4} of a prespecified, target periodic point. (As in the tent map example, there is nothing special about the choice of 7 steps, and we have found similar results for other, relatively short targeted orbits.) For each target, the algorithm is repeated until a set of 10^5 initial conditions is found that satisfies these criteria.

We then repeat the algorithm using each periodic point of period $p \leqslant 9$ as a target point. For each p, we let S_p denote the set of all the initial conditions found above that lead to one of the period-p points in $k = 7$ steps. In analogy with Eq. (3), we define

$$C_\gamma(p) = \frac{1}{kN} \sum_{z_0 \in S_p} \sum_{j=0}^{k-1} F_\gamma\left(Q_{a_j}(x_j)\right), \tag{4}$$

where N is the number of points in S_p.

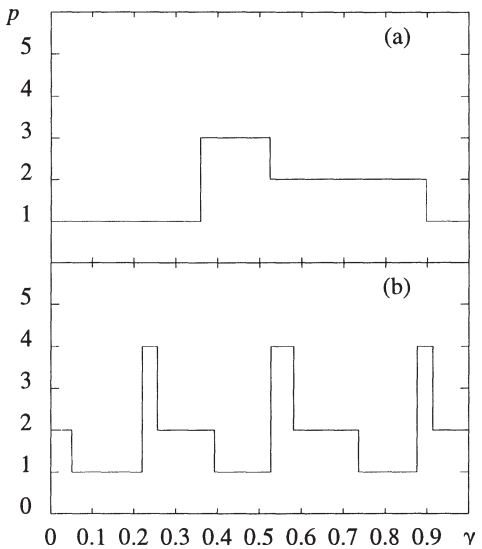


Fig. 3. Period of the orbit with minimal the average targeting cost per step for the quadratic map with $a = 3.72$. (a) The cost function $F_\gamma(x) = \cos 2\pi(x-\gamma)$; (b) the cost function $F_\gamma(x) = \cos 2\pi(x-\gamma) + \sin 6\pi(x-\gamma)$.

Fig. 3a shows the period p that minimizes $C_\gamma(p)$ as a function of γ. The family of performance functions is $F_\gamma(x) = \cos 2\pi(x-\gamma)$. This numerical experiment supports the expectation that for this family of performance functions, the low-period orbits have the lowest average targeting cost per step.

These numerical experiments are preliminary, but they do suggest that, in many cases, it is likely to be less expensive to target a periodic point of low period than a periodic point of high period, where the expense is defined in terms of the average value of a smooth cost function computed over a large number of targeting orbits. Additional investigations are underway to examine targeting orbits for periodic points in higher dimensional maps, such as the kicked double rotor map [17].

4. Conclusions

A variety of performance objectives may be important in targeting applications in chaotic dynamical systems. Two types of questions may be considered: Given a target, how can one identify an "optimal" orbit? Given a variety of targets (such as periodic saddle

orbits of various periods), which ones are least expensive to target, relative to some family of cost functions whose value is averaged over a large number of targeting orbits? Graph theoretic methods can be used to address the first question, and we have presented a preliminary set of numerical experiments that suggest that periodic orbits of low period may, in many cases, be less expensive to target than periodic orbits of high period.

Acknowledgment

We thank Brian Hunt for helpful comments, and James D. Meiss for a helful email. E.B. is supported in part by the National Science Foundation under grant DMS-9704639. E.K. is supported in part by the National Science Foundation's Computational and Applied Mathematics Program under grant DMS-9501077.

References

[1] E. Ott, C. Grebogi, J.A. Yorke, Phys. Rev. Lett. 64 (1990) 1196.
[2] E.M. Bollt, J.D. Meiss, Physica D 81 (1995) 280;
E.J. Kostelich, C. Grebogi, E. Ott, J.A. Yorke, Phys. Rev. E 47 (1993) 305;
T. Shinbrot, C. Grebogi, E. Ott, J.A. Yorke, Nature 363 (1993) 411;
J. Schweizer, M.P. Kennedy, Phys. Rev. E 52 (1995) 4865.
[3] E. Barreto, E.J. Kostelich, C. Grebogi, E. Ott, J.A. Yorke, Phys. Rev. E 51 (1995) 4169.
[4] C. Robinson, Dynamical Systems: Stability, Symbol Dynamics, and Chaos (CRC Press, Boca Raton, 1995).
[5] P. Cvitanović, G. Gunaratne, I. Procaccia, Phys. Rev. A 38 (1988) 1503;
P. Grassberger, H. Kantz, U. Moenig, J. Phys. A 22 (1989) 5217;
F. Christiansen and A. Politi, Nonlinearity 9 (1996) 1623.
[6] R. Gould, Graph Theory (Benjamin/Cummings, Menlo Park, 1988);
J.A. Bondy, U.S.R. Murty, Graph Theory with Applications (Elsevier, New York, 1976).
[7] E. Bollt, M. Dolnik, Phys. Rev. E 64 (1990) 1196.
[8] D. Lind, B. Marcus, An Introduction to Symbolic Dynamics and Coding (Cambridge Univ. Press, Cambridge, 1995).
[9] J.D. Farmer, E. Ott, J.A. Yorke, Physica D 7 (1983) 153.
[10] E. Bollt, J. Meiss, Physica D 81 (1995) 280.
[11] Y. Lai, M. Dong, C. Grebogi, Phys. Rev. E 47 (1992) 86.
[12] E. Bollt, E.J. Kostelich, in preparation.
[13] E.J. Kostelich, C. Grebogi, E. Ott, J.A. Yorke, Phys. Rev. E 47 (1993) 305.
[14] E.M. Bollt, Y.-C. Lai, C. Grebogi, Phys. Rev. Lett. 79 (1997) 3787.
[15] B.R. Hunt, E. Ott, Phys. Rev. E 54 328 (1996).
[16] T. Shinbrot, E. Ott, C. Grebogi, J.A. Yorke, Phys. Rev. Lett. 65 (1990) 3250.
[17] F.J. Romeiras, C. Grebogi, E. Ott, W.P. Dayawansa, Physica D 58 (1992) 165.
[18] C. Hsu, Cell-to-Cell Mapping (Springer, New York, 1987).