Interval propagation method for finding trajectories of chaotic maps

2004 J. Phys. A: Math. Gen. 37 6491

(http://iopscience.iop.org/0305-4470/37/25/005)

INSTITUTE OF PHYSICS PUBLISHING JOURNAL OF PHYSICS A: MATHEMATICAL AND GENERAL

J. Phys. A: Math. Gen. **37** (2004) 6491–6506 PII: S0305-4470(04)72267-2

Interval propagation method for finding trajectories of chaotic maps

Konstantin L Kouptsov

Department of Physics, Washington State University, Webster Physical Sciences, Pullman, WA 99164, USA

E-mail: kouptsov@wsu.edu

Received 18 November 2003, in final form 13 April 2004
Published 9 June 2004
Online at stacks.iop.org/JPhysA/37/6491
doi:10.1088/0305-4470/37/25/005

Abstract

The algorithm for calculation of trajectories of chaotic maps, based on the interval analysis, is proposed. Each of the image points is constrained by enclosing it in a corresponding interval. Improvement of one of the constraints results in the chain of adjustments of other interval bounds propagating along the trajectory, eventually causing the constraints to converge. No knowledge of well-defined symbolic dynamics is necessary since the pruning rules and non-uniqueness of the symbolic path are automatically resolved. For cycles and fixed-end orbits the algorithm provides linear uniform convergence. The algorithm is demonstrated for the Hamiltonian system where existence of both positive and negative Lyapunov exponents allows introduction of a simple interval-contracting procedure.

PACS number: 05.45.Mt

1. Introduction

The importance of periodic orbits in forming the underlying structure of chaotic dynamics is well known [1–6]. A number of algorithms for finding the periodic orbits have been developed. They are briefly reviewed below.

The earliest and simplest approach proposed was to identify the candidates for the periodic orbits from a set of numerically calculated trajectories and then use the Newton–Raphson iterations to improve precision [7].

Another technique [8] to find an n-periodic orbit $\{x_1, \ldots, x_n, x_{n+1} = x_1\}$ of the map $\mathbf{T}$ (which satisfies the equation $\mathbf{T}^n x = x$) is to consider an n-dimensional vector $X = \{x_i\}$, which satisfies the equation $(T - I)X = 0$, where T is the (nonlinear) n-dimensional representation of $\mathbf{T}$. The solution of the last equation is found by iterating the generalized Newton–Raphson

method $X_{n+1} = X_n + \left(T'_{X_n} - I\right)^{-1}(X_n - T X_n)$. Here T'_{X_n} is the linear tangent map (Jacobian matrix) at point X_n and I is the unitary matrix. The iterations converge very rapidly for a good initial guess. The drawback of this method is the one well known for the classical Newton's method: if the initial guess is ill chosen, the iterations may diverge, end up on a cycle, or wander off to a different point. With large n the map $\mathbf{T}$ is fast oscillating and the choice of initial point becomes even more difficult.

The generalization of the Newton–Raphson method to intervals was proposed by Z Galias [9] using the *Krawczyk operator*, a version of the interval Newton operator,

$$\mathbf{K}(\mathbf{x}) = x_0 - Cf(x_0) - (Cf'(\mathbf{x}) - I)(\mathbf{x} - x_0).$$

Here, $\mathbf{x}$ is an interval (or m-dimensional interval vector) containing a real number x_0 (m-dimensional vector, correspondingly) and C is the preconditioning matrix. The Krawczyk operator is commonly used in verification of the existence of solutions of nonlinear equations. In Galias' method one finds a fixed point of the equation $f(x) = \mathbf{T}^n x = x$ by enclosing the proposed region in an mn-dimensional cube and checking if it contains a fixed point of $\mathbf{T}^n$. The enclosing intervals are then subdivided until the desired precision is achieved. The method [9] thus allows finding all periodic orbits of a given length. Particularly, all periodic orbits up to period 15 were found for the Ikeda map.

Dynamics of certain chaotic systems was well studied to the degree that a good symbolic grammar for the orbits has been found. In the case of the stadium billiard, an example to be considered later, see, for instance, [10]. If the symbols are well ordered, the method proposed in [11] can be used. It is based on associating points (x, y) on the Poincaré plane with numeric values (δ, γ) of the symbolic string. Then by comparing (δ, γ) with (δ_C, γ_C), the symbolic values corresponding to the given string C, one can make subsequent approximations to the desired (x_C, y_C). The method also requires that the directions of the δ and γ axes be specially chosen parallel to the stable and unstable manifolds respectively.

The most popular and widely used methods are based on optimization. Thus in the case of a stadium billiard [12] an n-variable function is minimized when the reflection condition between the incoming and outgoing directions is satisfied at each point. Alternatively, according to the principle of least action, the true n-point orbit has extremal length with respect to small variations of its reflection points.

Yet another optimization technique, demonstrated for the Hénon [13, 14] and the standard [15] maps, is based on reproducing the map by a Hamiltonian H of a system of n interacting particles placed in an external field. The stable and unstable configurations, found by solving dynamics of the system, provide extremum to the energy and uniquely correspond to a periodic orbit of the map. The external field is adjusted with a set of parameters to obtain a particular orbit. The reader may find the discussion of these methods and further references in [16, chapter 14].

Another interesting method, the characteristic bisection method [17–20], is based on the topological degree theory [21]. It is a generalization of a well-known bisection method to arbitrary many dimensions, with the topological degree serving as a criterion for the existence of the root of the function of n variables inside a chosen n-dimensional polyhedron. The enclosing polyhedron is refined until the desired precision is achieved. The topological degree of an N-dimensional map $H(\mathbf{x}) : \mathbb{R}^N \to \mathbb{R}^N$, proved to be the number of the solutions of equation $H(\mathbf{x}) = 0$, can in general be found numerically by calculating the Kronecker integral. The method proposed in [17] avoids this calculation by checking that only the sign of the topological degree is preserved. The drawback of the characteristic bisection method is that in some cases it is unable to detect all roots of the map due to the bad choice of the initial polyhedron.

Optimization methods mentioned above are designed to solve the problem of finding the root of the function H of n variables, which, however, is stated more generally than the problem of finding the orbit of a map. A map brings every phase point into another phase point

$$(q_n, p_n) \longrightarrow (q_{n+1}, p_{n+1}). \tag{1}$$

The n steps of the map define a set of n separate equations to be satisfied. The essence of the method proposed in this paper is to approach the solution by successively satisfying these equations.

Development of the rigorous theory of the interval propagation method goes beyond the scope and intentions of the current paper, in which the method supported by several examples is explained.

2. Interval constraints propagation method

The mapping of the phase space

$$\begin{cases} q_{n+1} = Q(q_n, p_n) \\ p_{n+1} = P(q_n, p_n) \end{cases} \tag{2}$$

can, under quite general conditions, be converted to an equivalent second-order recurrence formula

$$f(x_{n-1}, x_n, x_{n+1}) = 0 \tag{3}$$

with the appropriate change of coordinates $x = x(p, q), x \in X$. One of the equations in (2) may serve as an additional equation to render the new system of equations complete. Use of delayed coordinates (x_{n-1}, x_n) is, in a way, a choice of Poincaré section. Equation (3) is an implicit function $x_n = F(x_{n-1}, x_{n+1})$, in general multivalued.

Domain X is partitioned into several nonintersecting domains (the Markov partition) $\{X_i\}$ labelled by the symbols of an alphabet. The orbit $\{x_i\}$ to be calculated is defined by its symbolic signature, or sequence of symbols $\{\alpha_i\}$. At every step of calculation each point of the orbit is bounded by an interval

$$x_i \in I_{\alpha_i} = [\underline{x}_i, \overline{x}_i] \subset X_{\alpha_i} \tag{4}$$

where $\underline{x}_i, \overline{x}_i$ are real numbers. Thus, the whole orbit corresponds to the vector of intervals (*interval vector*) $\mathbf{I} = \{I_i\}$. Initially $I_{\alpha_i} = X_{\alpha_i}$. Note that we do not require the knowledge of either the symbolic grammar to ensure the uniqueness of the orbit, or the *a priori* pruning rules.

For an orbit $\{x_i\}$ of length n we have a *chain* of n separate equations

$$f_{\alpha_{i-1}\alpha_i\alpha_{i+1}}(x_{i-1}, x_i, x_{i+1}) = 0 \qquad i = 1, \ldots, N \tag{5}$$

instead of a single equivalent equation $H(x_1, x_2, \ldots, x_n) = 0$ considered in other methods. Since the method allows calculation of fixed-end, periodic or open-end orbits, equations in the beginning and the end of the chain may differ from (5), depending on the boundary conditions.

In equations (5), every x_i belongs to the corresponding interval I_i. In this sense the function $F_{\alpha_{i-1}\alpha_i\alpha_{i+1}}(x_{i-1}, x_{i+1})$ is the *interval-valued* function of *interval arguments*, i.e. the range of values of F is

$$F(I_{i-1}, I_{i+1}) = \{F(x, y) : x \in I_{i-1}, y \in I_{i+1}\}. \tag{6}$$

The *interval arithmetic* technique was well developed in the literature starting with the classic book by Moore [22]. There exist interval analogues to elementary arithmetic operations $+, -, \times, /$ defined as the range of values of the corresponding function, the way similar to (6).

Combination of these operations generally allows one to calculate the range of values for the real rational expression with variables replaced by their interval ranges. However, this technique is known to cause undesirable widening of the resulting interval when the function has several entries of the same variable (each x in $x \cdot x$ is considered independent).

For the algorithm to work, we expect that F is *inclusion monotonic*

$$F(X, I) \subset F(Y, I) \qquad \text{and} \qquad F(I, X) \subset F(I, Y) \qquad \text{if} \quad X \subset Y. \tag{7}$$

and there is a positive real number L such that

$$w(F(X_1, X_2)) \leqslant L \max_{i=1,2} w(X_i) \tag{8}$$

where $w([a, b]) = b - a$ is the *width* of the interval ($b > a$).

If for some i the interval value of F is a strict subset of I_i

$$I_i' = F(I_{i-1}, I_{i+1}) \subsetneq I_i \tag{9}$$

the corresponding entry in $\mathbf{I}$ is updated. Now, due to (7), the neighbour intervals I_{i-1} and I_{i+1} must be recalculated, thereby causing a sequence of further updates. One way to proceed is to update only the right $(i + 1)$ neighbour, and continue moving in that direction. For periodic orbits, once the right end of the interval vector is reached, one continues updating from the left end, taking into account the periodicity of the vector. For open- and close-ended orbits, one continues from the right and moves in another direction. The process continues until the sequence $\mathbf{I}_i$ converges. The algorithm stops when either the widths of all intervals I_i become zero (which happens for fixed-end and periodic orbits) or condition (9) is no longer satisfied. The last case, in which a *bundle* of open-ended orbits is found, will be demonstrated below.

This algorithm resembles the method of solving an ordinary differential equation with fixed end points proposed in [23]. But the bounds for all intermediate points in the latter method are found simultaneously by using interval generalization of Newton's method [22, 24] or by solving a matrix equation. Sequential solution used in the current method has another advantage that will be addressed in the next section. This method also resembles the 'constraint satisfaction problem' (CSP) algorithms first introduced in [25]. For an overview of CSP see [26].

3. Pruning rules and uniqueness

Generally, there are two types of pruning rules [10]. *Parameter independent* rules arise from mere geometric setup of the problem (for example, in a billiard, the double bounce off the straight wall is not possible). They can be used to eliminate the impossible sequences prior to calculations. *Dynamic* or *parametric* pruning rules are much harder, if at all possible, to determine. They are applicable not to all possible sequences since they depend on additional parameters, such as the end points of the orbit.

As said above, this method does not require the knowledge of the pruning rules. Indeed, obtaining an empty interval value I' in (9) means that the corresponding symbolic sequence $\{\alpha_i\}$ (with the given interval vector) is pruned. Pruned sequences are usually eliminated during the first few rounds of the algorithm.

If (9) gives more than one interval, the particular symbolic sequence may correspond to multiple orbits. The symbolic grammar therefore needs to be extended by adding new letters. However, for the purpose of finding the orbits, this is not necessary. Nonuniqueness can be dealt with by using the *stack* or *queue* algorithmic storage mechanisms. Suppose the value of (9) is two nonintersecting intervals

$$I_i' \cup I_i'' = F(I_{i-1}, I_{i+1}) \subset I_i. \tag{10}$$

Then two interval vectors are formed

$$\mathbf{I}' = \{\ldots, I_i', \ldots\} \qquad \text{and} \qquad \mathbf{I}'' = \{\ldots, I_i'', \ldots\}.$$

The second, $\mathbf{I}''$, interval vector is stored in a stack while calculations are performed using the first one. During calculations additional intermediate interval vectors may be added to the storage. Once the first possibility is examined, i.e. either it is pruned out, or calculations converged, the *next* vector in storage is taken. Calculations are continued until the stack is empty.

With addition of the stack, we now have the complete algorithm.

Dynamically storing the intermediate data is more space and time efficient than identifying the m possible branching points along the path and generating 2^m possible interval vectors. Efficiency is achieved because the elimination of one pruned path may exclude more than one branch from further consideration.

4. Examples: Bunimovich stadium

4.1. Geometry

As an illustration of the above algorithm we consider a $\gamma = 2$ Bunimovich stadium—two unit semicircles joined by straight edges of length γ. A point inside the billiard is moving with a constant speed and makes elastic reflections off the boundaries. It is common to take the middle point of the right semicircle as the origin and measure the distance q_n to the nth reflection point clockwise along the boundary. The conjugate (momentum) coordinate p_n is the angle between the outbound trajectory and the forward tangent direction at point q_n. The delayed coordinates (q_n, q_{n+1}) are the coordinates of the end points of the outbound chord (figure 1(a)).

The boundary is naturally divided into four parts which we label T, B for the straight edges and L, R for the circular caps. The four-letter alphabet, however, is not enough to uniquely define an orbit; we would need to split L and R into L_+, L_-, R_+, and R_-: T is a bounce from the top horizontal border, B is a bounce from the bottom, $L_\pm$ is a bounce from the left semicircle clockwise/anticlockwise with respect to its centre and $R_\pm$ is a clockwise/anticlockwise bounce from the right cap.

Based on numerical evidence, Biham and Kvale conjectured [12] that no two distinct orbits can be described by the same symbol sequence of this six-letter alphabet. Later studies showed that the number of needed letters can be reduced to 5 by symmetry considerations. In the current paper it is shown that the use of the four-letter alphabet still allows finding all the orbits of the given symbolic signature.

For convenience, the coordinates q are taken to be local, i.e. relative to the corresponding boundary segment. The full coordinate of a bounce point is a pair $\{\alpha, \xi\}$, where $\alpha \in \{T, R, B, L\}$, and $\xi \in [0, 2]$ for $a = T$ or B or $\xi \in [0, \pi]$ for $a = L$ or R, figure 1(b). The boundary coordinates (α, ξ) for the point on the boundary can also be converted to the Cartesian coordinates (x, y). Further both of these representations are used interchangeably.

Our further goal is to define the chain functions (5) and to find a way to calculate the interval value (9) of function F.

For a trajectory passing through the space points (x_1, y_1) and (x_2, y_2), and reflecting off the flat boundary $x = 0$ the reflection point may be found by considering an image $(x_1', y_1') = (x_1, -y_1)$ of one of the points, figure 1(c). For reflection point at the origin, we have $x_1 y_2 + x_2 y_1 = 0$, or

$$X_1^T M X_2 = 0 \qquad \text{where} \quad X = \begin{pmatrix} x \\ y \end{pmatrix} \quad M = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \tag{11}$$

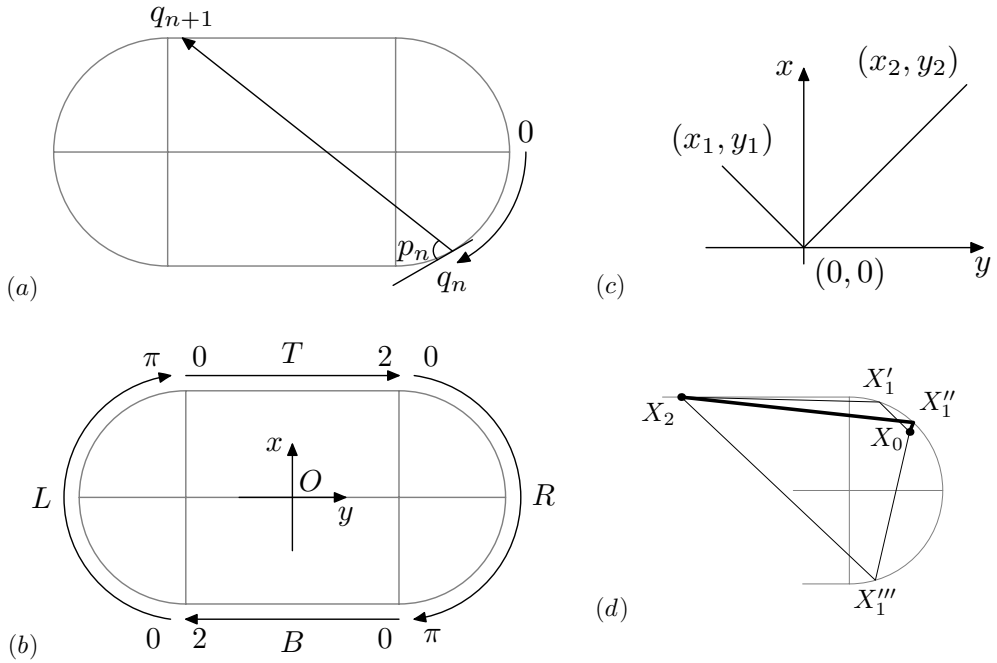

Figure 1. (*a*) Canonic (Birkhoff) (q_n, p_n) and delayed (q_n, q_{n+1}) coordinates, (*b*) local coordinates (α, ξ) on each of the boundary segments, (*c*) simple reflection geometry and (*d*) 'plane mirror' reflection orbit $X_0 X_1'' X_2$.

An arbitrary fragment (q_{n-1}, q_n, q_{n+1}) of trajectory, where q_n lies on the boundary, is reduced to this form by shifting (T_{q_n}) the middle point to the origin, and rotating the coordinates (R_ξ) to align the normal to the boundary at point q_n with the vertical. Rotation is necessary only for L and R bounces. Due to our choice of coordinates the rotation angle is the coordinate ξ_n. Assuming that points $X_1 = (x_1, y_1)$, X_2, X_3 correspond to coordinates (α, ξ_1), (β, ξ_2), (γ, ξ_3) on the boundary, the general form of equation (5) becomes

$$\begin{aligned} f_{\alpha\beta\gamma}(\xi_1, \xi_2, \xi_3) &= X_1^T T_{X_2}^T \left(R_{\xi_2}^T\right)^\nu M (R_{\xi_2})^\nu T_{X_2} X_3 \\ &= X_\alpha(\xi_1)^T Z_\beta(\xi_2) X_\gamma(\xi_3) = 0 \end{aligned} \tag{12}$$

where ξ_1, ξ_2, ξ_3 are the coordinates on the boundary segments α, β, γ correspondingly. Here, $T_{X_2} X_1 = (x_1 - x_2, y_1 - y_2)$ is the translation operator, R_ξ is the rotation by angle $\xi \in [0, \pi]$, $X_1 \equiv X_\alpha(\xi_1)$, $Z_\beta \equiv T_{X_2}^T \left(R_{\xi_2}^T\right)^\nu M \left(R_{\xi_2}\right)^\nu T_{X_2}$, and $\nu = 1$ for $\beta = L, R$ or $\nu = 0$ for $\beta = T, B$.

The chain functions (12) reflect the billiard's invariance under vertical ($T \leftrightarrow B$), horizontal ($L \leftrightarrow R$) and inversion ($T \leftrightarrow B$ and $L \leftrightarrow R$) symmetries. Also $f_{\alpha\beta\gamma}(x, y, z) = f_{\gamma\beta\alpha}(z, y, x)$, due to time reflection symmetry. The functions $f_{\alpha\beta\gamma}$ are listed in the appendix.

In order to use the described algorithm we must show that the functions satisfy conditions (7), (8). Each equation $f(x, y, z) = 0$, where the functions $f(x, y, z)$ are listed in the appendix, implicitly defines functions $y(x)$ with $z = \text{const}$, and $y(z)$ with $x = \text{const}$. The fact that both $y(x)$ and $y(z)$ are continuous and monotonic can be verified directly, but it can also be seen geometrically if one remembers that y, z are two consecutive bounce points of the ray emitted from the same point x on the boundary (and similarly for y and x). Therefore $y(x)$ and $y(z)$ are inclusion monotonic.

As far as (8) is concerned, these conditions follow from Bunimovich criteria for stochasticity of two-dimensional billiards [27], which in physical terms means that a parallel beam of trajectories reflecting off the focusing boundary diverges to a size larger than its original size before it reflects again. The billiard under consideration is a standard example of stochastic billiards satisfying Bunimovich criteria. Thus both conditions are satisfied.

For orbit with fixed end points X_1 and X_2 we have two more equations of the form

$$\begin{aligned} X_1^T Z_{q_1} X_{q_2} &= 0 \\ X_{q_{n-1}}^T Z_{q_n} X_2 &= 0 \end{aligned} \tag{13}$$

which together with (12) give us a complete chain.

For multiple bounces off the curved boundary, for example $\alpha = \beta = R$, equation (12) has an irrelevant root corresponding in this case to $\xi_1 = \xi_2$. This root is undesirable, since specifying the orbit by a symbolic sequence, we assume that all bounce points are different. If two bounce points coincide, the orbit would correspond to another, shorter, symbolic sequence.

To understand the appearance of this root, we note that for a flat wall, as one of the space points, say (x_1, y_1), approaches the boundary, it is also approached by the reflection point, until they finally merge. For a flat wall this reflection point is the only possible one. If the boundary is curved, there in general may be several reflection points, figure 1(*d*). In this case, if the space point X_0 is close enough to the boundary, the boundary may be locally considered as flat, and one of the bounce points, X_1'', will be in this flat neighbourhood, near X_0. If X_0 is inside the billiard, the path $X_0X_1''X_2$ is legitimate. On the other hand, for two bounces off the R wall, the case mentioned above, the path $X_0X_1''X_2$ becomes degenerate, since $X_0 = X_1''$, and must be eliminated. The functions (12) listed in the appendix are modified to exclude this root.

4.2. Interval value of a function

As said above, there are standard methods allowing one to calculate the interval value of the function (9). Here we use a more direct way based on the vertex method [28] stating that the bounds of a continuous function with no extreme points lie on the vertices of the n-dimensional rectangular region.

Let $f(x, y)$ be a real function, $\partial f/\partial y \neq 0$ for $x \in [\underline{x}, \overline{x}]$ and $y \in [\underline{y}, \overline{y}]$. Then $f(x, y) = 0$ implicitly defines (possibly many) branches of the function $y = y(x)$. Let $\partial f/\partial x \neq 0$ for each branch. If $\underline{y}$ and $\overline{y}$ are the roots of f for the corresponding values of x, $f(\underline{x}, \underline{y}) = 0$ and $f(\overline{x}, \overline{y}) = 0$ (we assume $\partial y/\partial x > 0$), then

$$f(\underline{x}, y) \cdot f(\overline{x}, y) < 0 \qquad \text{for} \quad \underline{y} < y < \overline{y} \tag{14}$$

and similarly for the other sign of the derivative.

To find a range of y, i.e. a set of subintervals of $[\underline{y}, \overline{y}]$ for which y satisfies the equation $f(x, y, z) = 0$ when $x \in [\underline{x}, \overline{x}]$ and $z \in [\underline{z}, \overline{z}]$, one performs the following steps:

1. Find all the roots

$$R(\underline{x}, \overline{x}, \underline{z}) = \{y_i\}_{i=1}^n = \{\underline{y}\} \cup \{\overline{y}\} \cup \{y | f(\underline{x}, y, \underline{z}) = 0 \text{ or } f(\overline{x}, y, \underline{z}) = 0\} \tag{15}$$

on the interval $[\underline{y}, \overline{y}]$ and include the end points $\underline{y}$ and $\overline{y}$. This will partition the interval into subintervals. From all the subintervals find those satisfying the condition (14)

$$\mathbf{I}_{\underline{z}} = \{[y_i, y_{i+1}] | f(\underline{x}, y, \underline{z}) \cdot f(\overline{x}, y, \underline{z}) < 0 \text{ for } y \in [y_i, y_{i+1}]\} \qquad i = 1, \ldots, n-1. \tag{16}$$

2. Repeat the same for $R(\underline{x}, \overline{x}, \overline{z})$ to find $\mathbf{I}_{\overline{z}}$. These intervals do not necessarily overlap, which is illustrated in figure 2(*a*). Points 1, 2, 3, 4, corresponding to minimum and

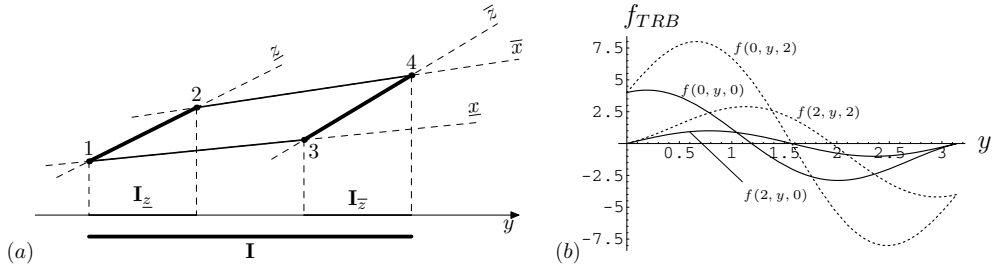

Figure 2. Finding the range of values of y in function $f(x, y, z) = 0$.

maximum values of x and z, lie on the surface $f(x, y, z) = 0$. Variation of x while keeping $z = \underline{z} = \text{const}$ produces the interval $\mathbf{I}_{\underline{z}}$. Similarly for $z = \overline{z}$ we get $\mathbf{I}_{\overline{z}}$. The union of these intervals, however, does not give the true interval $\mathbf{I}$. In order to fix that we repeat calculations with x and z exchanged, to obtain $\mathbf{I}_{\underline{x}}$, and $\mathbf{I}_{\overline{x}}$. The resulting interval is the union $\mathbf{I} = \mathbf{I}_{\underline{z}} \cup \mathbf{I}_{\overline{z}} \cup \mathbf{I}_{\underline{x}} \cup \mathbf{I}_{\overline{x}}$.

To illustrate the above method, let us consider the path *TRB*. Let the three points of the path be defined by the coordinates $X_0 = (T, x)$, $X_1 = (R, y)$ and $X_2 = (B, z)$. We are interested in the possible values of the middle point provided the other two points vary along the straight edges of the billiard. Subsequent points X_0, X_1 and X_2 on the boundary are associated with the equation $f_{\text{TRB}}(x, y, z) = 0$.

If the value of z is fixed, and the value of x varies in the interval $[0, 2]$, the plot of function $f(x, y, 0)$ gradually changes, shifting the value of its root $y = y(x)$. The two extremal positions of the plot are shown by solid lines in figure 2(*b*). Conditions $\partial f/\partial x \neq 0$ and $\partial f/\partial y \neq 0$ ensure that each root always shifts in the same direction as x increases, and that the merger of two roots never happens. Thus, the interval of possible values of y is determined by the values of the root at two extremal values of x: $x = 0$ and $x = 2$. In the case under consideration, $\mathbf{I}_{\underline{z}} = [1.187\,68, 1.5708]$ for $z = 0$. Similarly, one obtains the values of the other intervals, $\mathbf{I}_{\overline{z}} = [1.5708, 1.953\,91]$, $\mathbf{I}_{\underline{x}} = \mathbf{I}_{\underline{z}}$, $\mathbf{I}_{\overline{x}} = \mathbf{I}_{\overline{z}}$, and finally $\mathbf{I} = [1.187\,68, 1.953\,91]$.

Note that the range of values of the middle variable, y, was reduced from $[0, \pi]$ to the one above using only the requirement that it must be consistent with the possible values of the other variables. Now that the new range for y was established, this consistency requirement can be applied to x, or z, or both, causing a chain of updates. Another comment is that the validity of the conditions $\partial f/\partial x \neq 0$ and $\partial f/\partial y \neq 0$ implies the appropriate choice of the Markov partition.

4.3. Example 1: fixed-end orbit

As a simple illustration of the method we choose the fixed-end orbit passing from X_0 to X_1 by bouncing from L and T walls (x_0LTx_1, figure 3). The chain system consists of two equations $\{x_0, y_0\} \cdot Z_L(\xi_1)X_T(\xi_2) = 0$ and $X_L(\xi_1)Z_T(\xi_2) \cdot \{x_1, y_1\} = 0$. One starts from the initial intervals $[0, \pi]$ and $[0, 2]$ for ξ_1, ξ_2. Using the method of the previous section, one concludes that the range of values of ξ_0 consistent with $\xi_1 \in [0, 2]$ is $[1.139\,177, 1.646\,373]$. Now, assuming that $\xi_0 \in [1.139\,177, 1.646\,373]$, the range of values for ξ_1 is updated to be $[0.442\,633, 0.797\,404]$. This concludes the first step of the algorithm, and the new ranges for all variables, ξ_0 and ξ_1, are written in the first row of table 1.

At each step of the algorithm one updates ξ_0, then ξ_1. For longer orbits, one starts by updating ξ_0, then proceeds to sequentially update $\xi_1, \xi_2, \ldots$ until the end of the orbit. Then

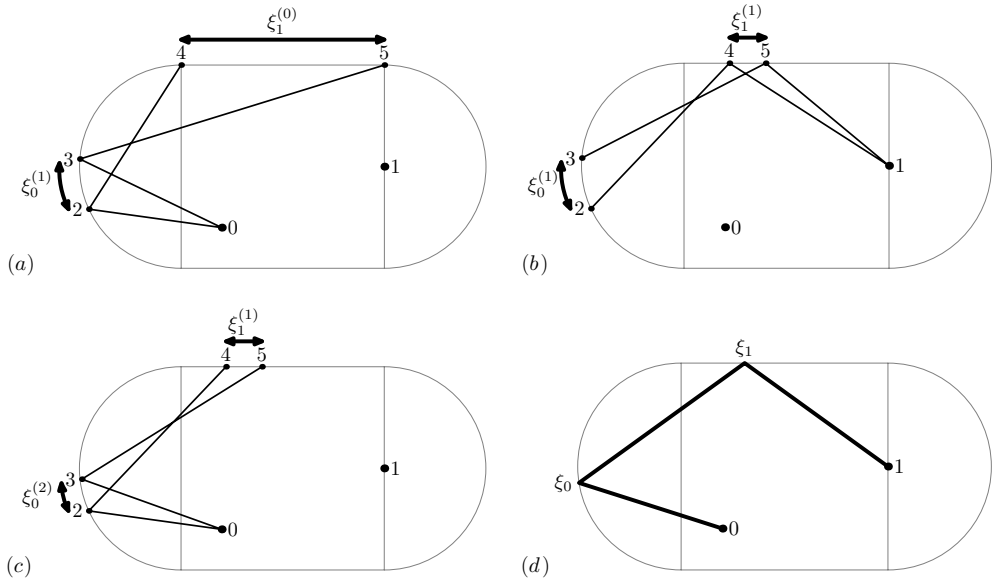

Figure 3. Interval method for the fixed-end path x_0LTx_1. (*a*) Condition that ξ_1 must lie in the top part of the boundary defines the possible values of $\xi_0^{(1)}$. (*b*) Paths originating from the interval for ξ_0 reach point 1 only if they pass through the narrower segment of the top boundary. (*c*) Second step of the algorithm refines the values of ξ_0. (*d*) True path to which iterations converge.

Table 1. Convergence of the intervals for the orbit x_0LTx_1.

	ξ_0	ξ_1
0	[0.000 000, 3.141 593]	[0.000 000, 2.000 000]
1	[1.139 177, 1.646 373]	[0.442 633, 0.797 404]
2	[1.351 985, 1.464 457]	[0.578 275, 0.657 617]
3	[1.400 229, 1.425 290]	[0.606 321, 0.624 035]
4	[1.409 339, 1.414 950]	[0.613 652, 0.617 617]
5	[1.411 674, 1.412 930]	[0.615 080, 0.615 967]
6	[1.412 127, 1.412 408]	[0.615 449, 0.615 647]
7	[1.412 244, 1.412 307]	[0.615 521, 0.615 565]
8	[1.412 267, 1.412 281]	[0.615 539, 0.615 549]
9	[1.412 272, 1.412 276]	[0.615 543, 0.615 545]
10	[1.412 274, 1.412 274]	[0.615 544, 0.615 544]
11	[1.412 274, 1.412 274]	[0.615 544, 0.615 544]
*	[1.412 274, 1.412 274]	[0.615 544, 0.615 544]

the chain of updates starts at the end of the orbit and proceeds backwards until the range of ξ_0 is again updated. This is the single step of the algorithm. Since all the intervals are updated in a single step, it can be shown that the convergence is uniform.

4.4. Example 2: periodic orbit

The next example demonstrates how the algorithm resolves ambiguity in the calculation of the periodic orbit $\overline{RLLRL}$ (table 2). To see that, we consider the first step of the algorithm in

Table 2. Finding two solutions of the periodic orbit with intermediate storage of interval vectors. Notation: $> S$—new interval vector is put on the stack; $S >$—next interval vector is taken from the stack; *—interval vector converged to a real vector corresponding to true trajectory; ×—some of the variables have empty interval value, interval vector discarded.

	$\xi_0 = \xi_5$	$\xi_1(R)$	$\xi_2(L)$	$\xi_3(L)$	$\xi_4(R)$	$\xi_5(L)$	$\xi_6 = \xi_1$
$> S$	[0.000, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[0.000, 3.142]
$S >$	[0.000, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[0.000, 3.142]
$> S$	[0.000, 3.142]	[1.107, 2.034]	[2.105, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[1.107, 2.034]
$> S$	[0.000, 3.142]	[1.107, 2.034]	[0.000, 1.036]	[1.953, 2.925]	[0.000, 3.142]	[0.000, 3.142]	[1.107, 2.034]
×	[1.417, 1.653]	[1.107, 2.034]	[0.000, 1.036]	[0.000, 0.363]	[1.107, 1.606]	[1.417, 1.653]	[1.107, 2.034]
$S >$	[0.000, 3.142]	[1.107, 2.034]	[0.000, 1.036]	[1.953, 2.925]	[0.000, 3.142]	[0.000, 3.142]	[1.107, 2.034]
	[1.472, 1.664]	[1.107, 1.702]	[0.386, 0.887]	[2.110, 2.864]	[1.439, 2.002]	[1.472, 1.664]	[1.107, 1.702]
	[1.556, 1.584]	[1.372, 1.475]	[0.503, 0.817]	[2.366, 2.605]	[1.682, 1.748]	[1.556, 1.584]	[1.372, 1.475]
	[1.565, 1.576]	[1.401, 1.451]	[0.608, 0.711]	[2.456, 2.510]	[1.709, 1.722]	[1.565, 1.576]	[1.401, 1.451]
	[1.569, 1.572]	[1.417, 1.434]	[0.646, 0.670]	[2.477, 2.489]	[1.714, 1.717]	[1.569, 1.572]	[1.417, 1.434]
	[1.570, 1.571]	[1.423, 1.427]	[0.655, 0.661]	[2.481, 2.484]	[1.715, 1.716]	[1.570, 1.571]	[1.423, 1.427]
	[1.570, 1.570]	[1.425, 1.426]	[0.657, 0.658]	[2.482, 2.483]	[1.715, 1.716]	[1.570, 1.570]	[1.425, 1.426]
	[1.570, 1.570]	[1.425, 1.425]	[0.658, 0.658]	[2.483, 2.483]	[1.716, 1.716]	[1.570, 1.570]	[1.425, 1.425]
	[1.570, 1.570]	[1.425, 1.425]	[0.658, 0.658]	[2.483, 2.483]	[1.716, 1.716]	[1.570, 1.570]	[1.425, 1.425]
*	[1.570, 1.570]	[1.425, 1.425]	[0.658, 0.658]	[2.483, 2.483]	[1.716, 1.716]	[1.570, 1.570]	[1.425, 1.425]
$S >$	[0.000, 3.142]	[1.107, 2.034]	[2.105, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[1.107, 2.034]
$> S$	[0.000, 3.142]	[1.438, 2.034]	[2.169, 3.142]	[2.801, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[1.438, 2.034]
	[1.476, 1.673]	[1.438, 2.034]	[2.169, 3.142]	[0.242, 1.187]	[1.134, 1.727]	[1.476, 1.673]	[1.438, 2.034]
	[1.554, 1.593]	[1.652, 1.810]	[2.307, 2.707]	[0.529, 0.806]	[1.391, 1.464]	[1.554, 1.593]	[1.652, 1.810]
	[1.564, 1.577]	[1.687, 1.750]	[2.427, 2.548]	[0.629, 0.691]	[1.418, 1.434]	[1.564, 1.577]	[1.687, 1.750]
	[1.568, 1.572]	[1.706, 1.726]	[2.470, 2.498]	[0.651, 0.665]	[1.423, 1.427]	[1.568, 1.572]	[1.706, 1.726]
	[1.570, 1.571]	[1.713, 1.718]	[2.480, 2.486]	[0.656, 0.660]	[1.425, 1.426]	[1.570, 1.571]	[1.713, 1.718]
	[1.570, 1.570]	[1.715, 1.716]	[2.482, 2.484]	[0.657, 0.658]	[1.425, 1.425]	[1.570, 1.570]	[1.715, 1.716]
	[1.570, 1.570]	[1.715, 1.716]	[2.483, 2.483]	[0.658, 0.658]	[1.425, 1.425]	[1.570, 1.570]	[1.715, 1.716]
	[1.570, 1.570]	[1.716, 1.716]	[2.483, 2.483]	[0.658, 0.658]	[1.425, 1.425]	[1.570, 1.570]	[1.716, 1.716]
*	[1.570, 1.570]	[1.716, 1.716]	[2.483, 2.483]	[0.658, 0.658]	[1.425, 1.425]	[1.570, 1.570]	[1.716, 1.716]
$S >$	[0.000, 3.142]	[1.438, 2.034]	[2.169, 3.142]	[2.801, 3.142]	[0.000, 3.142]	[0.000, 3.142]	[1.438, 2.034]
×							

detail. Note that the first and the last columns in the table are the duplicates of the other two, because of the circularity of the path. They are included for the convenience of calculations.

Initially, all the variables are bounded by intervals $[0, \pi]$, and the corresponding vector is put on the stack, as shown in the first row of table 2. The interval for ξ_1 is updated to be $[1.107, 2.034]$ by solving the interval equation $f_{LRL}([0, \pi], \xi_1, [0, \pi]) = 0$. The algorithm proceeds to the right by considering the interval equation $f_{RLL}([1.107, 2.034], \xi_2, [0, \pi]) = 0$, where $\xi_2 \in [0, \pi]$, and arriving at $\xi_2 \in [0, 1.036] \cup [2.105, \pi]$. One interval is used to form the interval vector

$$\mathbf{I} = \{L[0, \pi], R[1.107, 2.034], L[2.105, \pi], L[0, \pi], R[0, \pi], L[0, \pi], R[1.107, 2.034]\}$$

which is stored on a stack (row 3). Another interval, $[0, 1.036]$, is used to update the corresponding range for ξ_2.

We now proceed to the next variable, $\xi_3 \in [0, \pi]$. By solving the equation $f_{LLR}([0, 1.036], \xi_3, [0, \pi]) = 0$, we obtain $\xi_3 \in [0, 0.363] \cup [1.953, 2.925]$. Again, the new interval vector

$$\mathbf{I} = \{L[0, \pi], R[1.107, 2.034], L[0, 1.036], L[1.953, 2.925],$$
$$R[0, \pi], L[0, \pi], R[1.107, 2.034]\}$$

is stored on a stack (row 4), whereas the range for ξ_3 is updated, $\xi_3 \in [0, 0.363]$. Finally, it is found that $\xi_4 \in [1.107, 1.606]$ and $\xi_5 \in [1.417, 1.653]$.

After the first step of the algorithm we arrive at the interval vector

$$\mathbf{I} = \{L[1.417, 1.653], R[1.107, 2.034], L[0, 1.036], L[0, 0.363]$$
$$R[1.107, 1.606], L[1.268, 1.799], R[1.107, 2.034]\}$$

which is listed in row 5 of table 2. Now the algorithm starts the second round. The range for ξ_1 is updated to be [1.296, 1.521]. When solving the equation $f_{RLL}([1.296, 1.521], \xi_2, [0, 0.363]) = 0$, one finds that the interval for ξ_2 is empty. Hence, this interval vector is discarded, and the next one, shown in row 6, is popped out of the stack. When the current interval vector converged, the next one from the stack is taken (row 16, $S>$). The calculations are continued until the stack is empty.

Calculations in table 2 automatically found two solutions out of four possibilities. The existence of two solutions is the effect of the symmetry of the path with respect to the reflection $T \leftrightarrow B$.

4.5. Example 3: bundle of homoclinic orbits

In the third example we show how to calculate the open-end orbits, for which the ends are allowed to vary. As a result, the intervals bounding each bounce point converge to some minimal interval instead of a single point. Particularly we consider the calculation of a homoclinic orbit in the neighbourhood of the simplest periodic orbit $\overline{LR}$.

The orbit $\overline{LR}$ is a solution $(\pi/2, \pi/2)$ of the equation $f_{LRL}(x, y, x) = 0$ where f_{LRL} is given in the appendix. We are interested in the orbits starting at the nearby points. Expanding $f_{LRL}(x + \pi/2, y + \pi/2, z + \pi/2) = 0$, we obtain an approximation for z

$$z = -x + 6y + x^2y - 6xy^2 + 11y^3 + O(x, y, z)^5. \tag{17}$$

Considering only linear terms, we have

$$\begin{pmatrix} y \\ z \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -1 & 6 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}. \tag{18}$$

The eigenvalues λ and $1/\lambda$ of the matrix, where $\lambda = 3 - 2\sqrt{2} \approx 0.171\,57\ldots$ ($\lambda^2 = 6\lambda - 1$), indicate that it has a contracting and an expanding manifold. As shown in [29], a set of initial conditions centred around the periodic orbit $\overline{LR}$ will be stretched upon return into a thin strip parallel to the unstable direction. Going backward in time, this set will be stretched along the stable direction. This situation is illustrated in figure 4(*a*), where the initial conditions for orbits in each of the families i_k are chosen to lie along the unstable manifold, so the orbits diverge. On the other hand, the corresponding orbits from each family lie along the stable manifold, so these orbits converge.

The matrix is diagonal in the new coordinates (s, u), where

$$\begin{cases} x = s + \lambda u \\ y = u + \lambda s. \end{cases} \tag{19}$$

Here, s and u are the coordinates along *stable* (contracting) and *unstable* (expanding) directions.

Parameter λ is the scaling parameter, or the rate of decrease of the successive coordinates for the orbit lying on the stable manifold. This however is true only for small x. Thus,

$$f_{LRL}\left(\frac{\pi}{2} + x, \frac{\pi}{2} + \lambda x, \frac{\pi}{2} + \lambda^2 x\right) = O(x^3). \tag{20}$$

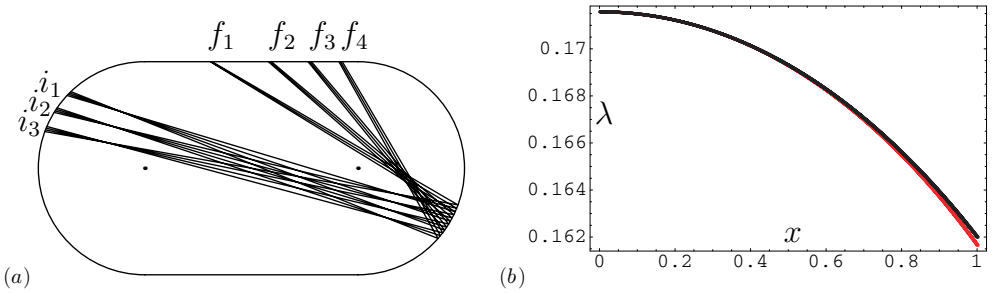

Figure 4. (*a*) Divergence and convergence of orbits lying on stable and unstable manifolds. (*b*) Coordinate dependence of the scaling parameter $\lambda(x)$.

Taking into account the higher powers in (17) leads to dependence of λ on coordinates (figure 4(*b*)). The difference between the two curves indicates the correction due to the fifth and all the higher powers.

The pair (s, u) simultaneously defines the positions of the first two points of the orbit; another pair similarly defines the last two points. The chain of equations looks similar to the fixed-end case, except that the first and last equations of the chain must be written in terms of new coordinates. Since we consider small deviations of the first and last bounces from $\pi/2$, the first two and the last two letters of the path can be either LR or RL. The chain system of equations for the path $LR\alpha_1 \cdots \alpha_n LR$, for example, becomes

$$\begin{cases} g_0(u_0, s_0, \xi_1) \equiv f_{LR\alpha_1}\left(\dfrac{\pi}{2} + s_0 + \lambda u_0, \dfrac{\pi}{2} + u_0 + \lambda s_0, \xi_1\right) = 0 \\ f_{R\alpha_1\alpha_2}(\xi_0, \xi_1, \xi_2) = 0 \\ \cdots \\ f_{\alpha_{n-1}\alpha_n L}(\xi_{n-1}, \xi_n, \xi_{n+1}) = 0 \\ g_1(\xi_n, u_1, s_1) \equiv f_{\alpha_n LR}\left(\xi_n, \dfrac{\pi}{2} + s_1 + \lambda u_1, \dfrac{\pi}{2} + u_1 + \lambda s_1\right) = 0 \end{cases} \tag{21}$$

where

$$\xi_0 = \pi/2 + u_0 + \lambda s_0 \qquad \text{and} \qquad \xi_{n+1} = \pi/2 + s_1 + \lambda u_1. \tag{22}$$

As an example of calculations we choose the homoclinic $(L)RTBL(R)$ orbit. Letters in parentheses indicate directions of the ends of the orbit. There are no bounces at those points for the fragment of trajectory under consideration. The parameter vector is taken as $\{s_0, u_0, \xi_1, \xi_2, s_1, u_1\}$.

The calculations given in table 3 are performed when allowing the ends of the orbit vary freely in the intervals $s_0, u_1 \in [-0.4, 0.4]$. The calculated bounds do converge; however, the resulting intervals for the inner coordinates appear too wide. The widening arises from allowing every bounce point vary independently within the corresponding interval, whereas for the billiard they are laced together with a single orbit. To determine the minimal intervals for each coordinate, we refer back to the fixed-end case, and find the orbits when s_0 and u_1 assume their extreme values (table 4).

The bundle of homoclinic orbits obtained in this calculation is shown in figure 5(*a*). The orbits in this bundle undergo mixing which is demonstrated by numbering the ends of each orbit by numbers 1, 2, 3 and 4. These numbers correspond to the rows in table 4. Figure 5(*b*) shows the initial and final coordinates of these orbits. All the initial conditions are located in the narrow strip i, parallel to the stable direction, which is then transformed into the strip f along the unstable direction. Intersection of these strips indicates the periodic orbit $\overline{LRTB}$.

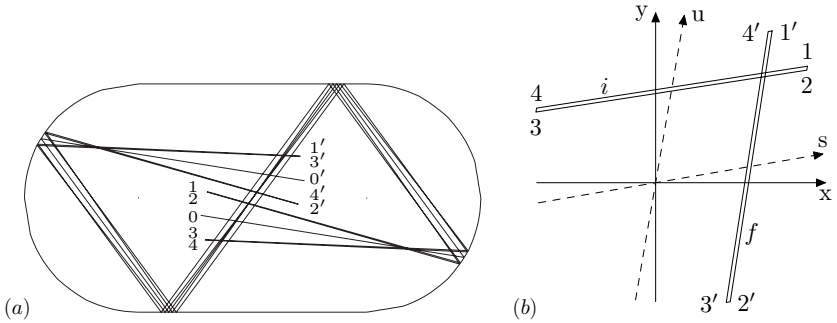

Figure 5. A bundle of homoclinic orbits (a) and the phase portrait of their initial and final states (b).

Table 3. Convergence of intervals for $(L)RTBL(R)$.

	s_0	u_0	ξ_1	ξ_2	s_1	u_1
	[−0.4000, 0.4000]	[−0.8000, 0.8000]	[0.0000, 2.0000]	[0.0000, 2.0000]	[−0.8000, 0.8000]	[−0.4000, 0.4000]
0	[−0.4000, 0.4000]	[0.2292, 0.6616]	[1.5205, 2.0000]	[1.4971, 1.9486]	[0.4382, 0.6616]	[−0.4000, 0.4000]
1	[−0.4000, 0.4000]	[0.4693, 0.6616]	[1.5343, 1.9095]	[1.5456, 1.9182]	[0.4718, 0.6369]	[−0.4000, 0.4000]
2	[−0.4000, 0.4000]	[0.4733, 0.6252]	[1.5676, 1.8990]	[1.5652, 1.8978]	[0.4795, 0.6225]	[−0.4000, 0.4000]
3	[−0.4000, 0.4000]	[0.4831, 0.6210]	[1.5739, 1.8891]	[1.5741, 1.8899]	[0.4843, 0.6187]	[−0.4000, 0.4000]
4	[−0.4000, 0.4000]	[0.4849, 0.6172]	[1.5779, 1.8865]	[1.5777, 1.8865]	[0.4857, 0.6165]	[−0.4000 0.4000]
5	[−0.4000, 0.4000]	[0.4862, 0.6162]	[1.5792, 1.8851]	[1.5792, 1.8852]	[0.4864, 0.6158]	[−0.4000, 0.4000]
6	[−0.4000, 0.4000]	[0.4865, 0.6156]	[1.5798, 1.8846]	[1.5798, 1.8846]	[0.4867, 0.6155]	[−0.4000, 0.4000]
7	[−0.4000, 0.4000]	[0.4867, 0.6154]	[1.5801, 1.8844]	[1.5801, 1.8844]	[0.4868, 0.6154]	[−0.4000, 0.4000]
8	[−0.4000, 0.4000]	[0.4868, 0.6153]	[1.5802, 1.8843]	[1.5802, 1.8843]	[0.4868, 0.6153]	[−0.4000, 0.4000]
9	[−0.4000, 0.4000]	[0.4868, 0.6153]	[1.5802, 1.8842]	[1.5802, 1.8842]	[0.4868, 0.6153]	[−0.4000, 0.4000]
+	[−0.4000, 0.4000]	[0.4868, 0.6153]	[1.5802, 1.8842]	[1.5802, 1.8842]	[0.4868, 0.6153]	[−0.4000, 0.4000]

Table 4. Individual orbits in a homoclinic bundle $(L)RTBL(R)$.

0	[0.0000, 0.0000]	[0.5474, 0.5474]	[1.7355, 1.7355]	[1.7355, 1.7355]	[0.5474, 0.5474]	[0.0000, 0.0000]
1	[0.4000, 0.4000]	[0.5489, 0.5489]	[1.7039, 1.7039]	[1.7039, 1.7039]	[0.5489, 0.5489]	[0.4000, 0.4000]
2	[0.4000, 0.4000]	[0.5370, 0.5370]	[1.6690, 1.6690]	[1.8006, 1.8006]	[0.5596, 0.5596]	[−0.4000, −0.4000]
3	[−0.4000, −0.4000]	[0.5477, 0.5477]	[1.7670, 1.7670]	[1.7670, 1.7670]	[0.5477, 0.5477]	[−0.4000, −0.4000]
4	[−0.4000, −0.4000]	[0.5596, 0.5596]	[1.8006, 1.8006]	[1.6690, 1.6690]	[0.5370, 0.5370]	[0.4000, 0.4000]

5. Conclusion

The presented algorithm makes use of the interval analysis technique. Generally, the interval technique is thought to be precise, compared to the regular floating point methods, because during the calculations the precise value to be calculated is made sure to be always enclosed by the corresponding interval. With the currently available interval analogues for trigonometric and other special functions, the interval technique is easy to implement on the computer.

As has already been mentioned in the introduction, a majority of the current techniques are designed to solve a problem of finding a fixed point of an n-dimensional function f. In interval methods, to approach the fixed point starting from a certain approximate value, one applies the Newton [22] or Krawczyk [9] interval *contracting* operators. In Galias' method,

the contracting operator is used only to verify the existence of a fixed point inside a region, and convergence is achieved by subsequent bisections of the intervals.

The proposed method is similar to the latter method, with the major difference that the selection of the intervals containing the solution is done dynamically, hence eliminating the need to check all the subintervals. Also, in the proposed method there is no need for the contracting operator, since contraction of the enclosing intervals is imposed by the divergence property of the map itself. The proposed method also makes the connection of the interval dynamics with the topology of the map, e.g. symbolic dynamics.

The method provides linear convergence which is slower than the more widely accepted minimization method; however, in addition to periodic orbits it allows calculation of trajectories with fixed or loose ends. The method provides uniform convergence along the trajectory regardless of its exponential instability.

The method does not require a well-defined symbolic dynamics (i.e. a symbolic sequence does not have to characterize only one trajectory). Pruning and resolution of multiple orbits is done automatically.

Conditions (7) and (8) by themselves do not guarantee that iterations will converge. Since every single step of the algorithm updates all intervals along the orbit, it is the combination of parameters L in (8) for each of the bounces that affect convergence. The exact form of the convergence condition depends on the order in which the intervals are updated.

The algorithm above uses constraints generated only for the middle point of each bounce. However, the end points of each bounce are also subject to constraints in some cases (for example TBR bounce is possible for all values of ξ_T: $\xi_T \in [0, 2]$. The other two coordinates become more restricted: $\xi_B \in [0, 1]$ and $\xi_R \in [0, \pi/2]$). Taking end points into consideration is a technical issue, and is not expected to affect the speed of convergence.

Another factor to affect the convergence of the algorithm is the choice of the Markov partition. As mentioned above, it does not have to be complete to be able to resolve individual orbits. On the other hand, the choice of partition affects the values of parameters L in (8).

Acknowledgments

The author thanks the reviewers for useful suggestions.

Appendix. Chain functions

As local coordinates, one can choose $X_L(\xi) = (-1 - \sin(\xi), -\cos(\xi), 1)$, $X_R(\xi) = (1 + \sin(\xi), \cos(\xi), 1)$, $X_T(\xi) = (-1 + \xi, 1, 1)$, $X_B(\xi) = (1 - \xi, 1, 1)$, where $\xi \in [0, \pi]$ for L or R, and $\xi \in [0, 2]$ for T or B. Note that the translation operator introduced in the text cannot be represented by a 2×2 matrix. Therefore here the 3×3 representation is chosen. Rotation, translation matrices and M are taken to be

$$R_\xi = \begin{pmatrix} \cos(\xi) & -\sin(\xi) & 0 \\ \sin(\xi) & \cos(\xi) & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad T_{(a,b)} = \begin{pmatrix} 1 & 0 & -a \\ 0 & 1 & -b \\ 0 & 0 & 1 \end{pmatrix} \qquad M = \begin{pmatrix} 0, 1, 0 \\ 1, 0, 0 \\ 0, 0, 0 \end{pmatrix}$$

In order to eliminate the spurious 'plain mirror' roots of the functions corresponding to the double LL or RR bounce, we adjust them as follows,

$$f'_{iRR}(x, y, z) \equiv f_{iRR}(x, y, z) \Big/ \sin\left(\frac{y - z}{2}\right)$$

$$f'_{RRR}(x, y, z) \equiv f_{RRR}(x, y, z) \Big/ \left(\sin\left(\frac{x - y}{2}\right) \sin\left(\frac{y - z}{2}\right)\right)$$

Table A1. Functions $f_{Tij}(x, \phi, y)$.

TRR	$\cos\frac{y-3\phi}{2} - \cos\frac{y-\phi}{2} + (2-x)\sin\frac{y-3\phi}{2}$
TRB	$(2-x+y)\cos\phi + (2-x-y)\cos 2\phi + (1+2y-xy)\sin 2\phi$
TRL	$(2-x)\cos(y-2\phi) + (4-x)\cos\phi - 2\cos 2\phi - \sin(y-2\phi) + \sin(y-\phi) + \sin\phi + (4-2x)\sin 2\phi$
TRT	$(5+xy-2x-2y)\cos\frac{\phi}{2} + (3+xy-2x-2y)\cos\frac{3\phi}{2} + (4-x-y)\sin\frac{3\phi}{2}$
TBR	$x-2+3\phi + (x+\phi-2)\cos y + 2\sin y$
TBT	$x+y+2\phi-4$
TBL	$x-6+3\phi + (2-x-\phi)\cos y - 2\sin y$
TLL	$\cos\frac{y-3\phi}{2} + \cos\frac{y-\phi}{2} - x\sin\frac{y-3\phi}{2}$
TLT	$-(x+y)(\cos\phi + \cos 2\phi) + 2\sin\phi - (xy-1)\sin 2\phi$
TLR	$-x\cos(y-2\phi) - 2\cos 2\phi - \sin(y-2\phi) - \sin(y-\phi) + \sin\phi - (2+x+4x\sin\phi)\cos\phi$
TLB	$-(2+x-y)\cos\phi + (x+y-2)\cos 2\phi - (1+2x-xy)\sin 2\phi$

Table A2. Functions $f_{Lij}(x, \phi, y)$.

LLL	$-\sin\left(\phi - \frac{x+y}{2}\right)$
LLT	$\cos\frac{x-3\phi}{2} + \cos\frac{x-\phi}{2} - y\sin\frac{x-3\phi}{2}$
LLR	$\cos\frac{x+2y-3\phi}{2} + \cos\frac{x-\phi}{2} - 2\sin\frac{x-3\phi}{2}$
LLB	$-\cos\frac{x-3\phi}{2} + \cos\frac{x-\phi}{2} + (y-2)\sin\frac{x-3\phi}{2}$
LTB	$-2+y+3\phi + (y+\phi-2)\cos x + 2\sin x$
LTR	$-2+2\phi + (\phi-2)\cos x - \phi\cos y + \sin x - \sin y - \sin(x+y)$
LRR	$\cos\frac{2x+y-3\phi}{2} + \cos\frac{y-\phi}{2} - 2\sin\frac{y-3\phi}{2}$
LRB	$y\cos(x-2\phi) + 2\cos 2\phi + \sin(x-2\phi) + \sin(x-\phi) - \sin\phi + (2+y+4y\sin\phi)\cos\phi$
LRL	$2\cos(x-2\phi) + 2\cos(y-2\phi) + 4\cos\phi + \sin(x+y-2\phi) + \sin(x-\phi) + \sin(y-\phi) + 4\sin 2\phi$
LRT	$(2-y)\cos(x-2\phi) + (4-y)\cos\phi - 2\cos 2\phi - \sin(x-2\phi) + \sin(x-\phi) + \sin\phi + 4\sin 2\phi - 2y\sin 2\phi$
LBR	$2\phi - 2 - \phi\cos x + (\phi-2)\cos y - \sin x + \sin y - \sin(x+y)$
LBT	$y-6+3\phi + (2-y-\phi)\cos x - 2\sin x$

and similarly for all others. The functions used in calculations are given in tables A1 and A2. This is necessary only for path segments where x and y bounce points lie on the boundary. All other functions are obtained from the symmetry properties of the billiard.

The functions are invariant under inversion symmetry ($T \leftrightarrow B$ and $L \leftrightarrow R$). Therefore the second half of the functions is not listed. Also, one can verify that these functions are symmetric, up to the sign, with respect to both vertical ($T \leftrightarrow B$)

$$X_{L(R)}(\phi) \leftrightarrow X_{L(R)}(\pi - \phi) \text{ and } X_{T(B)}(\xi) \leftrightarrow X_{B(T)}(2 - \xi)$$

or horizontal ($L \leftrightarrow R$)

$$X_{L(R)}(\phi) \leftrightarrow X_{R(L)}(\pi - \phi) \text{ and } X_{T(B)}(\xi) \leftrightarrow X_{T(B)}(2 - \xi)$$

reflections. Similar is true for $Z_i(x)$. Thus, for example, $f_{\mathrm{TRR}}(x, y, z) = f_{\mathrm{TLL}}(2 - x, \pi - y, \pi - z)$.

References

[1] Poincaré H 1892 *Les méthodes nouvelles de la méchanique céleste* (Paris: Gauthier-Villars)
[2] Van Vleck J H 1926 Quantum principles and line spectra *Bull. Natl. Res. Council* **10** 1–316
[3] Gutzwiller M C 1970 Energy spectrum according to classical mechanics *J. Math. Phys.* **11** 1791
[4] Gutzwiller M C 1971 Periodic orbits and classical quantization conditions *J. Math. Phys.* **12** 343
[5] Berry M V and Tabor M 1976 Closed orbits and the regular bound spectrum *Proc. R. Soc.* **349** 101–23

[6] Berry M V and Tabor M 1977 Calculating the bound spectrum by path summation in action-angle variables *J. Phys. A: Math. Gen.* **10** 371–9
[7] Auerbach Ditza, Cvitanović Predrag, Eckman Jean-Pierre and Gunaratne Gemunu 1987 Exploring chaotic motion through periodic orbits *Phys. Rev. Lett.* **58** 2387–9
[8] Percival Ian C 1991 Recent developments in classical mechanics *Chaos and Quantum Physics* ed M-J Giannoni, A Voros and J Zinn-Justin (Les Houches Summer School of Theoretical Physics) (Amsterdam: North-Holland) pp 11–85
[9] Galias Zbigniew 2002 Rigorous investigations of the Ikeda map by means of interval arithmetic *Nonlinearity* **15** 1759–79
[10] Hansen Kai T 1993 Symbolic dynamics II: The stadium billiard *Preprint* chao-dyn/9301004 v1
[11] Hansen Kai T 1995 Alternative method to find orbits in chaotic systems *Phys. Rev.* E **52** 2388–91 (*Preprint* chao-dyn/9507003)
[12] Biham Ofer and Kvale Mark 1992 Unstable periodic orbits in the stadium billiard *Phys. Rev.* A **46** 6334–9
[13] Biham Ofer and Wenzel Wolfgang 1989 Characterization of unstable periodic orbits in chaotic attractors and repellers *Phys. Rev. Lett.* **63** 819–22
[14] Biham Ofer and Wenzel Wolfgang 1990 Unstable periodic orbits and the symbolic dynamics of the complex hénon map *Phys. Rev.* A **42** 4639–46
[15] Wenzel Wolfgang, Biham Ofer and Jayaprakash C 1991 Periodic orbits in the dissipative standard map *Phys. Rev.* A **43** 6550–7
[16] Christiansen F 2003 Fixed points and how to get them *Chaos: Classical and Quantum* ed P Cvitanović, R Artuso, R Mainieri, G Tanner and G Vattay (Copenhagen: Niels Bohr Institute) chapter 14 (http://www.nbi.dk/ChaosBook)
[17] Vrahatis Michael N 1995 An efficient method for locating and computing periodic orbits of nonlinear mappings *J. Comput. Phys.* **119** 105–19
[18] Vrahatis Michael N, Bountis T and Kollmann M 1996 Periodic orbits and invariant surfaces of 4D nonlinear mappings *Int. J. Bifurcation Chaos* **6** 1425–37
[19] Polymilis C, Servizi G, Skokos Ch, Turchetti G and Vrahatis M N 2003 Locating periodic orbits by topological degree theory *Libration Point Orbits and Applications* ed G Gomez, M W Lo and J J Masdemont (Singapore: World Scientific) (*Preprint* nlin.CD/0211044)
[20] Polymilis C, Servizi G, Skokos Ch, Turchetti G and Vrahatis M N 2003 Topological degree theory and local analysis of area preserving maps *Chaos* **13** 94–104
[21] Lloyd Noel Glynne 1978 *Degree Theory* (*Cambridge Tracts in Mathematics* vol 73) (Cambridge: Cambridge University Press)
[22] Moore Ramon E 1966 *Interval Analysis* (Englewood Cliff, NJ: Prentice-Hall)
[23] Hansen Eldon 1969 On solving two-point boundary-value problems using interval arithmetic, ed Eldon Hansen *Topics in Interval Analysis* (Oxford: Oxford University Press) chapter 7
[24] Hansen Eldon 1968 On solving systems of equations using interval arithmetic *Math. Comput.* **22** 374–84
[25] Mackworth A K 1977 Consistency in networks of relations *Artif. Intell.* **8** 99–118
[26] Tsang Edward 1993 *Foundations of Constraints Programming* (London: Academic)
[27] Bunimovich L A 1991 Conditions of stochasticity of two-dimensional billiards *Chaos* **1** 187–93
[28] Dong Weimin and Shah Haresh C 1987 Vertex methods for computing functions of fuzzy variables *Fuzzy Sets Syst.* **24** 65–78
[29] Tomsovic Steven and Heller Eric J 1993 Long-time semiclassical dynamics of chaos: the stadium billiard *Phys. Rev.* E **47** 282–99