CHAPTER III

PERIODIC ORBITS AND HOW TO FIND THEM

In the previous chapter, we have emphasized the crucial role of UPOs in studying a dynamical system and suggested application of the periodic orbit theory to spatiotemporal chaos. The successful implementation of these ideas depends heavily on efficient numerical techniques for finding UPOs. Various methods have been developed to find UPOs in low-dimensional chaotic systems, either discrete or continous [51], while efficient methods are still called upon in high-dimensional systems. In this chapter, we will review common techniques and then introduce a novel variational scheme for finding periodic orbits. Finally, further developments of the method and some possible applications are proposed.

3.1 Review of known techniques for locating UPOs

In essence, any method for numerically finding periodic orbits is based on devising a new dynamical system which possesses the desired orbit as an attracting fixed point with a sizable basin of attraction. Beyond that, there is much freedom in constructing such systems. In the following, several techniques are briefly reviewed with their advantages and disadvantages being mentioned.

3.1.1 Inverse iteration and anti-integral limit method

The inverse iteration is most easily described in the $1-d$ map and can be generalized to higher-dimensional cases under certain cercumstances. The anti-integral limit method is closely related to the inverse iteration. Both of them enjoy direct and heuristic applications of symbolic dynamics.

To find a prime cycle with a given symbol sequence in the unimodal map, we read the symbol sequence backward, pick up a point in the subinterval corresponding to the last symbol, and then start the inverse iteration. A point has two preimages and only one of them is picked up according to the reversed symbol sequence. As $f(x)$ is expanding, the inverse iteration is contracting. The resulting orbit point will approach the desired periodic orbit exponentially fast since the discrepancy shrinks exponentially under the contractive inverse map.

In higher dimensions, maps or flows usually have both contracting and expanding directions. Inverse iteration does not apply directly. If in the contracting directions, the SIS is thin enough to be treated approximately as an zero-dimensional object - a finite set of points (actually a fractal set with a dimension between 0 and 1; by "thin", we mean that the dimension is close to 0), we are able to build a non-invertible expanding map by

projecting the original map onto the unstable manifolds and investigate the simplied model system. As there are only expanding directions now, inverse iteration may be invoked again to give periodic orbits of the model system. These orbits are often quite close to the true UPOs of the original system, and are profitably used as the starting guesses for other UPO determining methods. The anti-integral limit [11] is a good example in this direction [241].

In spirit, the anti-integrable limit is similar to the slaving function [249] for flows. Consider a smooth flow

$$\dot{x} = f(x, y)\,,\ \dot{y} = \frac{-y}{\epsilon} + g(x, y)\,, \tag{37}$$

where $x, f \in \mathbb{R}^d\,; y, g \in \mathbb{R}^m$, and ϵ is a small parameter. If we consider the motion in a bounded neighborhood $\mathcal{N}$ of the origin, in the limit $\epsilon \to 0$, we may set $y = 0$ due to the strong contraction term $-y/\epsilon$ in the second equation of (37). To the zeroth order in ϵ, $y = 0$ defines a slaving function of the motion, putting geometrical constraints on the asymptotic dynamics which is determined by the first equation of (37). When ϵ assumes a small but finite value, the motion is not strictly confined though very close to the slaving manifold. Hyperbolic motion on the manifold at $\epsilon = 0$ can easily be continued to motion at $\epsilon > 0$. Alternatively, we may multiply the second equation by ϵ and ignore the time evolution term on the left hand side due to the smallness of ϵ. The idea can be conveniently carried over to a continuous family of maps $x_{k+1} = f_\epsilon(x_k)$ with $x, f \in \mathbb{R}^d\,, \epsilon \in \mathbb{R}$, which can be written as

$$\epsilon f_1(x_{k+1}, x_k) + f_2(x_{k+1}, x_k) = 0.$$

In the anti-integrable limit $\epsilon \to 0$, we have the geometrical constraints $f_2(x_{k+1}, x_k) = 0$, rendering a new map hopefully less complex than the original one. In particular, when f_2 only involves x_k, the constraints $f_2(x_k) = 0$ give a countable (usually finite) collection of fixed points. If we assign a symbol to each of these points, the map reduces to arbitrary transitions among these points and can be conveniently described by a symbolic dynamics. Under quite general assumptions, the orbits, especially the UPOs, prescibed by the symbolic dynamics can be uniquely continued to the orbit for $\epsilon \neq 0$ [241].

We now see an example. Consider the circle map

$$x_{i+1} = x_i + \alpha + k \sin x_i\,, \tag{38}$$

which is equivalent to

$$\sin x_i - \epsilon(x_{i+1} - x_i - \alpha) = 0\,,$$

with $\epsilon = 1/k$. In the anti-integral limit, $\epsilon \to 0$, the dynamical system becomes implicit and defined by the equation

$$sinx_i = 0$$

which yields the points $x_i = m_i\pi\,, m_i \in \mathbb{Z}$. So the dynamics of (38) for large k is coded by an infinite sequence of symbols. On a circle, x_i is defined up to multiples of 2π, the symbol sequence reduces to two symbols. See [11] for more examples.

In both the anti-integrable limit or the inverse iteration, the deterministic nature of maps is lost as the current state cannot uniquely determine the future state. We need a symbol sequence to make the map uniquely defined. The current position in the expanding direction is determined by the future itinerary and in the contracting direction determined

by the past itinerary. Even the crudest description - the symbolic dynamics will determine the position accurately. So, the topology prescribed by the symbolic dynamics is robust and very constraining. In this sense, the symbolic dynamics and the stability eigenvalues give a detailed intrinsic description of the phase dynamics on the SIS and define an abstract dynamical object. How to embed this object to a specific phase space is a geometrical problem determined by the specific dynamical system.

3.1.2 Phase-space partition method and cyles from long time series

Very often, we have a long discrete time series extracted from either experimental measurements or numerical simulations on a chaotic system. To find embedded UPOs, we have to depend on near recurrences, having in mind the ergodic property of chaotic systems. We prefix a positive integer n as the period of the periodic orbits under search and start to compare all data pairs that are separated by $n-1$ intermediate points. If the data in a pair are close to each other, it is likely that they are close to a periodic orbit of period n. These pairs can be used as a starting point for further calculations. In this way, sets of periodic orbits and their stability eigenvalues can be calculated quite accurately [12].

When the phase space is known or reconstructed from time series, we may do a little bit better. One natural way to approximate the chaotic dynamics on a compact invariant set is to partition the phase space (or Poincaré section) into a set of controllable cells and study mappings among these cells. If a cell contains a fixed point, then it must map to itself. To reduce workload, a coarse partition is used at the beginning and the mappings of the cells are checked. Refinements are made only with those which contain themselves in their images. In ref. [123], linearized maps are used to approximate the nonlinear map near the lattice points and successive refinements locate the fixed points quite accurately.

The good part of this global searching method is that it only needs evaluation of the map on the lattice points, which avoids the usual time-consuming calculation of the Jacobian of the map. There are two disadvantages. If the phase space dimension is high, it requires a good many cells to cover the SIS and this may make the program very slow. Secondly, if a long cycle is under search, the linear region of the repeated map is so small that the partition has to be very fine to capture the essential dynamics, which of course could tremendously increase the computation load and lower numerical accuracy.

3.1.3 Shooting method

If we have a good guess of the position of one or more points on a periodic orbit, simple or multipoint shooting [242, 51] is a nice method to apply. They are variants of Newton's root-searching scheme and retain its quadratic convergence when the guess is close to a true solution.

Consider a map $f : \mathbb{R}^d \to \mathbb{R}^d$. We are trying to find one of its fixed points, say $\bar{x} : f(\bar{x}) = \bar{x}$, starting from the present guess x_k. Assume that x_k is close to $\bar{x}$, that is to say, $\bar{x} = x_k + \delta x_k$ with $|\delta x_k|$ small. From the fixed point condition, we have

$$x_k + \delta x_k = f(x_k + \delta x_k) \approx f(x_k) + \delta x_k \cdot \nabla f(x_k)\,.$$

Solving for δx_k and replacing x_k by $x_{k+1} = x_k + \delta x_k$ result in Newton-Raphson iteration,

$$x_{k+1} = x_k + (1 - \nabla f(x_k))^{-1} \cdot (f(x_k) - x_k)\,, \tag{39}$$

which is a widely employed method for finding roots of nonlinear equations.

Small modifications easily adapt the method to treating flows. Consider a flow $f^t : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}^d$. A Poincaré section $\mathcal{P} = \{x \in \mathbb{R}^d : a \cdot x = 0\}$, where $a \in \mathbb{R}^d$ is given, will produce a map out of the flow if orbits of the flow intersect $\mathcal{P}$ transversely. Fixed points on $\mathcal{P}$ are periodic orbits for the flow. Suppose that we have a guess of x_k on $\mathcal{P}$ and the return time t_k at a cycle $\bar{x} = x_k + \delta x_k$ with the return time $\bar{t} = t_k + \delta t_k$. From the periodicity condition and the assumption that $\delta x_k, \delta t_k$ are small, we have

$$\begin{aligned} x_k + \delta x_k &= f^{t_k + \delta t_k}(x_k + \delta x_k) \\ &\approx f^{t_k}(x_k) + \mathbb{J}_k \cdot \delta x_k + v_k \delta t_k\,, \end{aligned} \tag{40}$$

where

$$v_k = \frac{\partial f^{t_k}}{\partial t}(x_k)\,,\ J_k = \frac{\partial f^{t_k}}{\partial x}(x_k)\,.$$

In matrix form, we have

$$\begin{pmatrix} 1 - J_k & -v_k \\ a & 0 \end{pmatrix} \begin{pmatrix} \delta x_k \\ \delta t_k \end{pmatrix} = \begin{pmatrix} f^{t_k}(x_k) - x_k \end{pmatrix}\,, \tag{41}$$

where the last row keeps the coordinates on the Poincaré section. Solving (41) gives the corrections δx_k and δt_k. For very long or very unstable orbits, multipoint shooting replaces the simple shooting to make each of the shooting segments controllable, thus leading to a more stable numerical procedure [242].

Since the multipoint shooting only involves a local orbit segment, it is very adaptive. An example is associated with the development of automatic differentiation [105]. By applying the method, the local orbit segment is conveniently expressed in terms of polynomials of the time variable from the truncated Taylor series. For finding UPOs, a set of points are sprinkled along the initial guessed orbit positions. The polynomials give local solution curves through each point. Matching values of the polynomials and their derivatives at intermediate points leads to the location of true periodic orbits [108].

3.1.4 Variational and relaxation approach

If the equations of motion in a dynamical system are associated with a variational principle, extremization techniques can be invoked to produce periodic orbits. For example, in a classical mechanical system, with Lagrangian $\mathcal{L}(q, \dot{q}, t)$, where $q \in \mathbb{R}^d$, the equations of motion are determined by the action principle

$$\delta R[q] = \delta \int_{t_1}^{t_2} dt\, \mathcal{L}(q, \dot{q}, t) = 0\,. \tag{42}$$

Euler-Langrian equation is a consequence of (42) if the initial configuration at t_1 and the final configuration at t_2 are fixed. The action principle can also be used directly for searching periodic orbits. Creating a loop, a smooth closed curve $q(t)$ with $q(t+T) = q(t)$ as the initial guess and extremizing (42) leads to a periodic orbit. This can be done in either Fourier space (in view of the periodicity) or configuration space. The advantage of this approach is that cycles are produced with a given topology since the initial guess is not a point but a whole loop and the topological feature of the targeting cycle is very robust.

In Lagrangian maps, orbits can also be expressed as extrema of a variational principle. Periodic orbits are stationary points of a finite set of ODEs [149, 81, 253]. Even when the map is not area-preserving, sometimes it is still possible to find a variational principle for the orbits. For example, Biham and Wenzel [18] proposed a fictitious time τ "Hamiltonian"

$$H = \frac{1}{2}\Sigma_k\, b^{-k}\left(\frac{dx_k}{dt}\right)^2 + (-b)^{-k}\left[x_k(x_{k+1}-x_{k-1}) - (b^{-1}+1)\left(ax_k - \frac{1}{3}x_k^3\right)\right] \tag{43}$$

whose stationary points are orbits of the Hénon map

$$x_{k+1} = a - x_k^2 + by_k\,,\ y_{k+1} = x_k\,. \tag{44}$$

The fictitious time flow toward a guessed n-point cycle is governed by

$$\frac{dx_k}{dt} = s_k F_k\,,\ k = 1, \cdots, n\,, \tag{45}$$

where $F_k = (-b)^{-k}(b^{-1}+1)[-x_{k+1}+a-x_k^2+bx_{k-1}]$, $s_k = \pm 1$ and $x_{n+1} = x_1$. The stationary points of (45) correspond to the period n cycles of (44). If an equilibrium is stable, it attracts the corresponding cycle guess in its basin of attraction. In general, a stationary point can be an attractor, a repeller or a saddle point, and in Biham-Wenzel approach a stationary point of (45) is made stable with appropriate choice of prefactors s_k. Depending on the a, b values, almost all admissible periodic orbits up to some length are found if 2^n different choices of $\{s_k\}_{k=1}^n$ are tried. However, K. T. Hansen pointed out that the method fails to converge for some cycles [115]. In ref. [102], it is shown that the method might fail in two ways: the search ends up with a limit cycle of (45), or two different sequences of $\{s_k\}$ result in the same orbit. Later, in another paper [19], the authors extended the method to cycle search on the complex plane. There, all the 2^n cycles (including the complex ones) of length n are found.

The idea of changing stability of orbits was further pursued in refs. [61, 64, 204, 203, 224]. Suppose that we have a map $f : \mathbb{R}^d \to \mathbb{R}^d$. The fixed points are obtained by designing a new map

$$L_n :\ x_{k+1} = x_k + S_n[f(x_k) - x_k]\,, \tag{46}$$

where S_n is a constant invertible matrix. Fixed points of L_n are by construction also the fixed points of f. It is expected that by multiplying with an appropriate S_n the desired periodic orbit becomes stable for L_n, and with other choices of S_n's, L_n converges to other cycles. The advantage of the approach is that an explicit analytical expression for the map is not necessary. Problems of such approaches are similar to those mentioned above for the Biham-Wenzel's method for the Hénon map, and .

In sect. 3.2, we will devise a variational method for searching periodic orbits in a general flow. Our method combines the robustness of the relaxation approach with the fast

convergence of the Newton-Raphson method when the search is sufficiently close to a true UPO. The idea of our method is to make an informed rough guess of what the desired cycle looks like globally, and then use a variational method to drive the initial guess toward the exact solution. For robustness, we replace the guess of a single orbit point by a guess of an entire orbit. For numerical safety we replace the Newton-Raphson iteration by the "Newton descent", a differential flow that minimizes monotonically a cost function computed as deviation of the approximate flow from the true flow along a smooth loop approximation to a cycle.

3.2 The Newton descent method in loop space

The multipoint shooting method eliminates the long-time exponential instability of unstable orbits by splitting an orbit into a number of short segments, each with a controllable expansion rate. Combined with the Newton-Raphson root-seaching technique it is an efficient tool for locating periodic orbits of maps [38]. A search for periodic orbits of a continuous time flow can be reduced to a multiple shooting search for periodic orbits of a set of maps by constructing a set of phase space Poincaré sections such that an orbit leaving one section reaches the next one in a qualitatively predictable manner, without traversing other sections along the way. In turbulent, high-dimensional flows such sequences of sections are hard to come by. One solution might be a large set of Poincaré sections, with the intervening flight segments short and controllable.

Here we follow a different strategy, and discard Poincaré sections altogether; we replace maps between spatially fixed Poincaré sections, by maps induced by discretizing the time evolution into small time steps. For sufficiently small time steps such maps are small deformations of identity. We distribute many points along a smooth loop L, our initial guess of a cycle location and its topological layout. If both the time steps and the loop deformations are taken infinitesimal, a partial differential equation governs the "Newton descent", a fictitious time flow of a trial loop L into a genuine cycle p, with exponential convergence in the fictitious time variable. We then use methods developed for solving PDEs to get the solution.

In this section we derive the partial differential equation which governs the evolution of an initial guess loop toward a cycle and the corresponding cost function. An extension of the method to Hamiltonian systems and systems with higher time derivatives is presented in sect. 3.3. Simplifications due to symmetries and details of our numerical implementation of the method are discussed in sect. 3.4. In sect. 3.5 we test the method on the Hénon-Heiles system, the restricted three body problem, and a weakly turbulent Kuramoto-Sivashinsky system. We summarize our results and discuss possible improvements of the method in sect. 3.6.

3.2.1 A variational equation for the loop evolution

A periodic orbit is a solution (x, T), $x \in \mathbb{R}^d$, $T \in \mathbb{R}$ of the *periodic orbit condition*

$$f^T(x) = x\,, \qquad T > 0 \tag{47}$$

for a given flow or discrete time mapping $x \mapsto f^t(x)$. Our goal is to determine periodic orbits of flows defined by first order ODEs

$$\frac{dx}{dt} = v(x)\,, \qquad x \in \mathcal{M} \subset \mathbb{R}^d\,, \qquad (x,v) \in \mathbf{T}\mathcal{M} \tag{48}$$

in many (even infinitely many) dimensions d. Here $\mathcal{M}$ is the phase space (or state space) in which evolution takes place, $\mathbf{T}\mathcal{M}$ is the tangent bundle [8], and the vector field $v(x)$ is assumed to be smooth (sufficiently differentiable) almost everywhere.

We make our initial guess at the shape and the location of a cycle p by drawing a loop L, a smooth, differentiable closed curve $\tilde{x}(s) \in L \subset \mathcal{M}$, where s is a loop parameter. As the loop is periodic, we find it convenient to restrict s to $[0, 2\pi]$, with the periodic condition $\tilde{x}(s) = \tilde{x}(s + 2\pi)$. Assume that L is close to the true cycle p, pick N pairs of nearby points along the loop and along the cycle

$$\begin{aligned} \tilde{x}_n &= \tilde{x}(s_n)\,, \qquad 0 \le s_1 < \ldots < s_N < 2\pi\,, \\ x_n &= x(t_n)\,, \qquad 0 \le t_1 < \ldots < t_N < T_p\,, \end{aligned} \tag{49}$$

and denote by $\delta\tilde{x}_n$ the deviation of a point x_n on the periodic orbit p from the nearby point $\tilde{x}_n$,

$$x_n = \tilde{x}_n + \delta\tilde{x}_n\,.$$

The deviations $\delta\tilde{x}$ are assumed small, vanishing as L approaches p.

The orientation of the s-velocity vector tangent to the loop L

$$\tilde{v}(\tilde{x}) = \frac{d\tilde{x}}{ds}$$

is intrinsic to the loop, but its magnitude depends on the (still to be specified) parametrization s of the loop.

At each loop point $\tilde{x}_n \in L$ we thus have two vectors, the loop tangent $\tilde{v}_n = \tilde{v}(\tilde{x}_n)$ and the flow velocity $v_n = v(\tilde{x}_n)$. Our goal is to deform L until the directions of $\tilde{v}_n$ and v_n coincide for all $n = 1, \ldots, N$, $N \to \infty$, that is $L = p$. To match their magnitude, we introduce a local time scaling factor

$$\lambda(s_n) \equiv \Delta t_n / \Delta s_n\,, \tag{50}$$

where $\Delta s_n = s_{n+1} - s_n, n = 1, \ldots, N-1\,, \Delta s_N = 2\pi - (s_N - s_1)$, and likewise for Δt_n. The scaling factor $\lambda(s_n)$ ensures that the loop increment Δs_n is proportional to its counterpart $\Delta t_n + \delta t_n$ on the cycle when the loop L is close to the cycle p, with $\delta t_n \to 0$ as $L \to p$.

Let $x(t) = f^t(x)$ be the state of the system at time t obtained by integrating (48) with initial state x, and $\mathbf{J}(x,t) = dx(t)/dx$ be the corresponding Jacobian matrix obtained by integrating

$$\frac{d\mathbf{J}}{dt} = A\mathbf{J}\,, \quad A_{ij} = \frac{\partial v_i}{\partial x_j}\,, \qquad \text{with } \mathbf{J}(x,0) = \mathbb{1}\,. \tag{51}$$

Since the point $x_n = \tilde{x}_n + \delta\tilde{x}_n$ is on the cycle,

$$f^{\Delta t_n + \delta t_n}(\tilde{x}_n + \delta\tilde{x}_n) = \tilde{x}_{n+1} + \delta\tilde{x}_{n+1}\,. \tag{52}$$

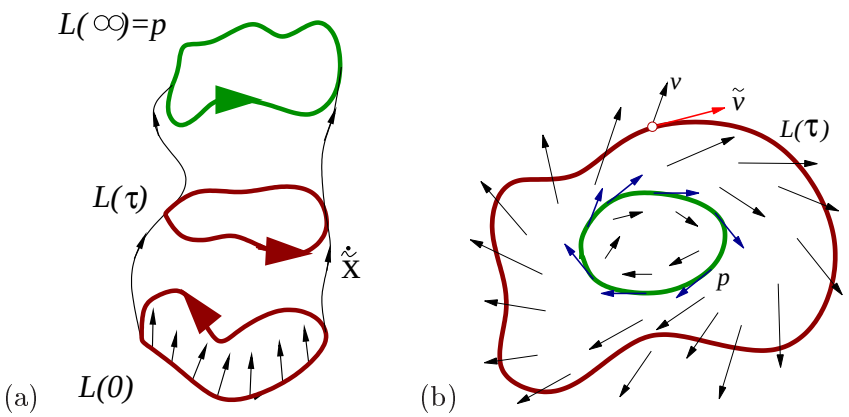

Figure 2: (a) An annulus $L(\tau)$ swept by the Newton descent flow $d\tilde{x}/d\tau$, connecting smoothly the initial loop $L(0)$ to the periodic orbit $p = L(\infty)$. (b) In general the loop velocity field $\tilde{v}(\tilde{x})$ does not coincide with $\lambda v(\tilde{x})$; for a periodic orbit p, it does so at every $x \in p$.

Linearization

$$f^{\delta t}(x) \approx x + v(x)\delta t\,, \qquad f^{t}(x+\delta x) \approx x(t) + \mathbf{J}(x,t)\delta x\,,$$

of (52) about the loop point $\tilde{x}_n$ and the time interval Δt_n to the next cycle point leads to the multiple shooting Newton-Raphson equation, for any step size Δt_n:

$$\delta\tilde{x}_{n+1} - \mathbf{J}(\tilde{x}_n, \Delta t_n)\delta\tilde{x}_n - v_{n+1}\delta t_n = f^{\Delta t_n}(\tilde{x}_n) - \tilde{x}_{n+1}\,. \tag{53}$$

Provided that the initial guess is sufficiently good, the Newton-Raphson iteration of (53) generates a sequence of loops L with a decreasing cost function [56]

$$F^2(\tilde{x}) \equiv \frac{N}{(2\pi)^2}\sum_{i=1}^{N}(f^{\Delta t_n}(\tilde{x}_n) - \tilde{x}_{n+1})^2, \qquad \tilde{x}_{N+1} = \tilde{x}_1\,. \tag{54}$$

The prefactor $N/(2\pi)^2$ makes the definition of F^2 consistent with (59) in the $N \to \infty$ limit. If the flow is locally strongly unstable, the neighborhood in which the linearization is valid could be so small that the full Newton step would overshoot, rendering F^2 bigger rather than smaller. In this case the step-reduced, damped Newton method is needed. As proved in ref. [144], under conditions satisfied here, F^2 decreases monotonically if appropriate step size is taken. If infinitesimal steps are taken, decrease of F^2 is ensured. We parametrize such continuous deformations of the loop by a *fictitious time* τ.

We fix Δs_n and proceed by $\delta\tau$ each step of the iteration, that is, multiply the right hand side of (53) by $\delta\tau$. According to (50), the change of Δt_n with respect to τ is equal to $\delta t_n = \frac{\partial\lambda}{\partial\tau}(s_n,\tau)\delta\tau\Delta s_n$. As $\delta\tilde{x}_n = \frac{\partial}{\partial\tau}\tilde{x}(s_n,\tau)\delta\tau$, dividing both sides of (53) by $\delta\tau$ yields

$$\begin{aligned}\frac{d\tilde{x}_{n+1}}{d\tau} - \mathbf{J}(\tilde{x}_n,\Delta t_n)\frac{d\tilde{x}_n}{d\tau} - v_{n+1}\frac{\partial\lambda}{\partial\tau}(s_n,\tau)\Delta s_n \\ = f^{\Delta t_n}(\tilde{x}_n) - \tilde{x}_{n+1}\,.\end{aligned} \tag{55}$$

In the $N \to \infty$ limit, the step sizes $\Delta s_n, \Delta t_n = O(1/N) \to 0$, and we have

$$\begin{aligned} v_{n+1} &\approx v_n\,, & \tilde{x}_{n+1} &\approx \tilde{x}_n + \tilde{v}_n\Delta s_n\,, \\ \mathbf{J}(\tilde{x}_n,\Delta t_n) &\approx 1 + A(\tilde{x}_n)\Delta t_n\,, & f^{\Delta t_n}(\tilde{x}_n) &\approx \tilde{x}_n + v_n\Delta t_n\,. \end{aligned}$$

Substituting into (55) and using the scaling relation (50), we obtain

$$\frac{\partial^2 \tilde{x}}{\partial s \partial \tau} - \lambda A \frac{\partial \tilde{x}}{\partial \tau} - v \frac{\partial \lambda}{\partial \tau} = \lambda v - \tilde{v}. \tag{56}$$

This PDE, which describes the evolution of a loop $L(\tau)$ toward a periodic orbit p, is our central result. The family of loops so generated is parametrized by $\tilde{x} = \tilde{x}(s,\tau) \in L(\tau)$, where s denotes the position along the loop, and the fictitious time τ parametrizes the deformation of the loop, see figure 2(a). We refer to this infinitesimal step version of the damped Newton-Raphson method as the "Newton descent".

The important feature of this equation is that a decreasing cost functional exists. Rewriting (56) as

$$\frac{\partial}{\partial \tau}(\tilde{v} - \lambda v) = -(\tilde{v} - \lambda v)\,, \tag{57}$$

we have

$$\tilde{v} - \lambda v = e^{-\tau}(\tilde{v} - \lambda v)|_{\tau=0}\,, \tag{58}$$

so the fictitious time τ flow decreases the cost functional

$$F^2[\tilde{x}] = \frac{1}{2\pi} \oint_{L(\tau)} ds\ (\tilde{v}(\tilde{x}) - \lambda v(\tilde{x}))^2 \tag{59}$$

monotonically as the loop evolves toward the cycle.

At each iteration step the differences of the loop tangent velocities and the dynamical flow velocities are reduced by the Newton descent. As $\tau \to \infty$, the fictitious time flow aligns the loop tangent $\tilde{v}$ with the dynamical flow vector $\tilde{v} = \lambda v$, and the loop $\tilde{x}(s,\tau) \in L(\tau)$, see figure 2(b), converges to a genuine periodic orbit $p = L(\infty)$ of the dynamical flow $\dot{x} = v(x)$. Once the cycle p is reached, by (50), $\lambda(s,\infty) = \frac{dt}{ds}(\tilde{x}(s,\infty))$, and the cycle period is given by

$$T_p = \int_0^{2\pi} \lambda(\tilde{x}(s,\infty)) ds\,.$$

Of course, as at this stage we have already identified the cycle, we may pick instead an initial point on p and calculate the period by a direct integration of the dynamical equations (48).

3.2.2 Marginal directions and accumulation of loop points

Numerically, two perils lurk in a direct implementation of the Newton descent (56).

First, when a cycle is reached, it remains a cycle under a cyclic permutation of the representative points, so on the cycle the operator

$$\bar{A} = \frac{\partial}{\partial s} - \lambda A$$

has a marginal eigenvector $v(\tilde{x}(s))$ with eigenvalue 0. If λ is fixed, as the loop approaches the cycle, (56) approaches its limit

$$\bar{A}\,\frac{\partial x}{\partial \tau} = 0\,.$$

Therefore, on the cycle, the operator $\bar{A}^{-1}$ becomes singular and the numerical woes arise.

The second potential peril hides in the freedom of choosing the loop (re-)parametrization. Since s is related to the time t by the yet unspecified factor $\lambda(s,\tau)$, uneven distributions of the sampling points over the loop L could arise, with the numerical discretization points $\tilde{x}_n$ clumping densely along some segments of L and leaving big gaps elsewhere, thus degrading the numerical smoothness of the loop.

We remedy these difficulties by imposing constraints on (56). In our calculation for Kuramoto-Sivashinsky system of sect. 3.5, the first difficulty is dealt with by introducing one Poincaré section, for example, by fixing one coordinate of one of the sampling points, $\tilde{x}_1(s_2,\tau) = const$. This breaks the translational invariance along the cycle. Other types of constraints might be better suited to a specific problem at hand. For example, we can demand that the average displacement of the sampling points along the loop vanishes, thus avoiding a spiraling descent toward the desired cycle.

We deal with the second potential difficulty by choosing a particularly simple loop parametrization. So far, the parametrization s is arbitrary and there is much freedom in choosing the best one for our purposes. We pick $s-$independent constant scaling $\lambda(s,\tau) = \lambda(\tau)$. With uniform grid size $\Delta s_n = \Delta s$ and fixed λ (in s), the loop parameter $s = t/\lambda$ is proportional to time t, and the discretization (56) distributes the sampling points along the loop evenly in time. As the loop approaches a cycle, $\frac{\partial\tilde{x}}{\partial\tau}$ is numerically obtainable from (56), and on the cycle the period is given by $T_p = 2\pi\lambda$.

Even though here we concentrate on searching for periodic orbits, the Newton descent is a general method. With appropriate modifications of boundary conditions and scaling of time, (56) can be adapted to determination of homoclinic or heteroclinic orbits between equilibrium points or periodic orbits of a flow, or more general boundary value problems. Applied to 2-point boundary value problems, Newton descent is similar to the quasilinearization [242] but has the advantage that the free parameters $\lambda(s,\tau)$ are available for adjusting scales in the problem and that searches can be restricted to phase space submanifolds of interest. A simple example of a restriction to a submanifold are searches for cycles of a given energy, constrained to the $H(q,p) = E$ energy shell in the phase space of a Hamiltonian system. Furthermore, as we shall show now, the symplectic structure of Hamilton's equations greatly reduces the dimensionality of the submanifold that we need to consider.

3.3 Extensions of Newton descent

As mentioned above, in classical mechanics particle trajectories are also solutions of a variational principle, the Hamilton's variational principle. For example, variational methods are the key ingredient of the Aubry-Mather theory of area-preserving twist maps, discrete-time Hamiltonian dynamical systems particularly suited to explorations of the K.A.M. theorem. Proofs of the Aubry-Mather theorem [173] on existence of quasi-periodic solutions are variational. It was quickly realized that the variational methods can also yield reliable, high precision computations of long periodic orbits of twist map models in 2 or more dimensions, needed for K.A.M. renormalization studies [150].

A fictitious time gradient flow for orbits of mappings, similar to the one discussed in ref. [56], was introduced by Anegent [2] for twist maps, and used by Gole [99] in his proof of the Aubry-Mather theorem. Mathematical bounds on the regions of stability of K.A.M. tori are notoriously restrictive compared to the numerical indications, and de la Llave, Falcolini and Tompaidis [81, 253] have found the gradient flow formulation advantageous both in studies of the analyticity domains of the K.A.M. stability, as well as proving the Aubry-Mather theorem for extended systems.

As far as we know, all numerical applications so far have been to low-dimensional Hamiltonian maps, not to continuous time flows. Instead of attempting to implement the least-action variational principle as loop dynamics in a fictitious time, here we shall implement our Newton descent as a flow that again minimizes a cost function, this time one that penalizes misalignment of accelerations, the true one and the one computed on the loop approximation to a cycle. We discuss the merits of the two kinds of variational principles in Appendix B.

To motivate what follows on level of everyday intuition, consider how the least action periodic orbit search works for a billiard: Connect points by a rubber band with a desired topology, and then move the points along the billiard walls until the length (that is, the action) of the rubber band is extremal (maximal or minimal under infinitesimal changes of the boundary points). Note that the extremization of action requires only D configuration coordinate variations, not the full $2D$-dimensional phase space variations.

Can we exploit this property of the Newtonian mechanics to reduce the dimensionality of our variational calculations? The answer is yes, and easiest to understand in terms of the Hamilton's variational principle which states that classical trajectories are extrema of the Hamilton's principal function (or, for fixed energy E, the action $S = R + Et$)

$$R(q_1, t_1; q_0, t_0) = \int_{t_0}^{t_1} dt\, \mathcal{L}(q(t), \dot{q}(t), t)\,,$$

where $\mathcal{L}(q, \dot{q}, t)$ is the Lagrangian. Given a loop $L(\tau)$ we can compute not only the tangent "velocity" vector $\tilde{v}$, but also the local loop curvature or "acceleration" vector

$$\tilde{a} = \frac{\partial^2 \tilde{x}}{\partial s^2}\,,$$

and indeed, as many s derivatives as needed. Matching the dynamical acceleration $a(\tilde{x})$ (assumed to be functions of $\tilde{x}$ and $v(\tilde{x})$) with the loop "acceleration" $\tilde{a}(\tilde{x})$ results in a new cost function and the corresponding PDE (57) for the evolution of the loop

$$\frac{\partial}{\partial \tau}(\tilde{a} - \lambda^2 a) = -(\tilde{a} - \lambda^2 a)\,.$$

We use λ^2 instead of λ in order to keep the notation consistent with (50), that is $t = \lambda\, s$. Expressed in terms of the loop variables $\tilde{x}(s)$, the above equation becomes

$$\frac{\partial^3 \tilde{x}}{\partial^2 s \partial \tau} - \lambda \frac{\partial a}{\partial v}\frac{\partial^2 \tilde{x}}{\partial s \partial \tau} - \lambda^2 \frac{\partial a}{\partial \tilde{x}}\frac{\partial \tilde{x}}{\partial \tau} + \left(\frac{\partial a}{\partial v}\frac{\partial \tilde{x}}{\partial s} - 2\lambda a\right)\frac{\partial \lambda}{\partial \tau} = \lambda^2 a - \tilde{a}\,, \tag{60}$$

where $v = \frac{\partial \tilde{x}}{\lambda \partial s}$. Although (60) looks more complicated than (56), in numerical fictitious time integrations, we are rewarded by having to keep only half of the phase space variables.

More generally, if a differential equation has the form:

$$x^{(m)} = f(x, x^{(1)}, \cdots, x^{(m-1)}) , \tag{61}$$

where $x^{(k)} = \frac{d^k x}{dt^k}$, $k = 1, \cdots, m$ and $x \in \mathbb{R}^d$, the same technique can be used to match the highest derivatives $\lambda^m x^{(m)}$ and $\tilde{x}^{(m)}$,

$$\frac{\partial}{\partial \tau}(\tilde{x}^{(m)} - \lambda^m x^{(m)}) = -(\tilde{x}^{(m)} - \lambda^m x^{(m)}) ,$$

with $\tilde{x}^{(m)} = \frac{\partial^m}{\partial s^m}\tilde{x}(s)$ calculated directly from $\tilde{x}(s)$ on the loop by differentiation. In loop variables $\tilde{x}(s)$ we have,

$$\frac{\partial^{m+1}\tilde{x}}{\partial s^m \partial \tau} - \lambda^m \sum_{k=0}^{m} \frac{\partial f}{\partial x^{(k)}} \cdot \frac{\partial}{\partial \tau} \frac{\partial^k \tilde{x}}{\lambda^k \partial s^k} - m\lambda^{m-1}\tilde{x}^{(m)} \frac{\partial \lambda}{\partial \tau} = \lambda^m x^{(m)} - \tilde{x}^{(m)} , \tag{62}$$

where $x^{(0)} = x$ and $x^{(k)} = \frac{\partial^k \tilde{x}}{\lambda^k \partial^k s}$, $k = 1, \cdots, m-1$ are assumed. Conventionally, (61) is converted to a system of md first order differential equations, whose discretized derivative (see (63) below) are banded matrices with band width of $5md$. Using (62), we only need d equations and for the same accuracy the corresponding band width is $(m+4)d$. The computing load has been greatly reduced, the more so the larger m is. Nevertheless, choice of a good initial loop guess and visualization of the dynamics are always aided by a plot of the orbit in the full md-dimensional phase space, where loops cannot self-intersect and topological features of the flow are exhibited more clearly.

3.4 Implementation of Newton descent

As the loop points satisfy a periodic boundary condition, it is natural to employ truncated discrete Fast Fourier Transforms (FFT) in numerical integrations of (56). Since we are interested only in the final, stationary cycle p, the accuracy of the fictitious time integration is not crucial; all we have to ensure is the smoothness of the loop throughout the integration. The Euler integration with fairly large fictitious time steps $\delta\tau$ suffices. The computationally most onerous step in implementation of the Newton descent is the inversion of the large matrix $\bar{A}$ in (56). When the dimension of the dynamical phase space of (48) is high, the inversion of $\bar{A}$ needed to get $\frac{\partial \tilde{x}}{\partial \tau}$ takes most of the integration time, making the evolution extremely slow. This problem is partially solved if the finite difference methods are used. The large matrix $\bar{A}$ then becomes sparse and the inversion can be done far more quickly.

3.4.1 Numerical implementation

In a discretization of a loop, numerical stability requires accurate discretization of loop derivatives such as

$$\tilde{v}_n \equiv \left.\frac{\partial \tilde{x}}{\partial s}\right|_{\tilde{x}=\tilde{x}(s_n)} \approx (\hat{D}\tilde{x})_n .$$

In our numerical work we use the four-point approximation [24],

$$\hat{D} = \frac{1}{12h} \begin{pmatrix} 0 & 8 & -1 & & & & & & 1 & -8 \\ -8 & 0 & 8 & -1 & & & & & & 1 \\ 1 & -8 & 0 & 8 & -1 & & & & & \\ & & & & & \cdots & & & & \\ & & & & & & 1 & -8 & 0 & 8 & -1 \\ -1 & & & & & & & 1 & -8 & 0 & 8 \\ 8 & -1 & & & & & & & 1 & -8 & 0 \end{pmatrix} \tag{63}$$

where $h = 2\pi/N$. Here, each entry represents a $[d \times d]$ matrix, $8 \to 8\mathbb{1}$, *etc.*, with blank spaces filled with zeros. The two $[2d \times 2d]$ matrices

$$M_1 = \begin{pmatrix} \mathbb{1} & -8\mathbb{1} \\ 0 & \mathbb{1} \end{pmatrix}, \qquad M_2 = \begin{pmatrix} -\mathbb{1} & 0 \\ 8\mathbb{1} & -\mathbb{1} \end{pmatrix},$$

located at the top-right and bottom-left corners take care of the periodic boundary condition.

The discretized version of (56) with a fictitious time Euler step $\delta\tau$ is

$$\begin{pmatrix} \hat{A} & -\hat{v} \\ \hat{a} & 0 \end{pmatrix} \begin{pmatrix} \delta\hat{x} \\ \delta\lambda \end{pmatrix} = \delta\tau \begin{pmatrix} \lambda\hat{v} - \hat{\tilde{v}} \\ 0 \end{pmatrix}, \tag{64}$$

where

$$\hat{A} \;=\; \hat{D} - \lambda\, \mathrm{diag}[A_1, A_2, \cdots, A_N]\,,$$

with $A_n = A(\tilde{x}(s_n))$ defined in (51), and

$$\begin{aligned} \hat{v} &= (v_1, v_2, \cdots, v_N)^t\,, \qquad \text{with } v_n = v(\tilde{x}(s_n))\,, \\ \hat{\tilde{v}} &= (\tilde{v}_1, \tilde{v}_2, \cdots, \tilde{v}_N)^t\,, \qquad \text{with } \tilde{v}_n = \tilde{v}(\tilde{x}(s_n))\,, \end{aligned}$$

are the two column vectors that we want to match during the evolution of the loop. $\hat{a}$ is an Nd dimensional row vector which imposes the constraint on the coordinate variations $\delta\hat{x} = (\delta\tilde{x}_1, \delta\tilde{x}_2, \cdots, \delta\tilde{x}_N)$. The discretized Newton descent (64) is an infinitesimal time step variant of the multipoint (Poincaré section) shooting equation for flows(53). In formulating a variational method for periodic orbit searches in a 3-d generalized standard map, Tompaidis [253] also derived an expression similar to (64). In order to solve for the deformation of the loop coordinates and period, $\delta\hat{x}$ and $\delta\lambda$, we need to invert the $[(N\,d+1)\times(N\,d+1)]$ matrix on the left hand side of (64).

In our numerical work, this matrix is inverted using the banded LU decomposition on the embedded band-diagonal matrix, and the Woodbury formula [208] on the cyclic and border terms. The LU decomposition takes most of the computation time and considerably slows down the fictitious time integration. We speed up the integration by a new inversion scheme which relies on the smoothness of the flow in the loop space. It goes as follows. Once we have the LU decomposition at one step, we use it to approximately invert the matrix in the next step, with accurate inversion achieved by the iterative approximate inversions [208]. In our applications we find that a single LU decomposition can be used for many $\delta\tau$ evolution steps. The further we go, the more iterations at each step are needed to implement the

inversion. After the number of such iterations exceeds some given fixed maximum number, we perform another LU decomposition and proceed as before. The number of integration steps following one decomposition is an indication of the smoothness of the evolution, and we adjust accordingly the integration step size $\delta\tau$: the greater the number, the bigger the step size. As the loop approaches a cycle, the evolution becomes so smooth that the step size can be brought all the way up to $\delta\tau = 1$, the full undamped Newton-Raphson iteration step. In practice, one can start with a small but reasonable number of points, in order to get a coarse solution of relatively low accuracy. After achieving that, the refined guess loop can be constructed by interpolating more points, and can be processed with a more accurate calculation in which $\delta\tau$ can be set as large as the full Newton step $\delta\tau = 1$, recovering the rapid quadratic convergence of the Newton-Raphson method.

It is essential that the smoothness of the loop is maintained throughout the calculation. We monitor the smoothness by checking the Fourier spectrum of $\tilde{x}(\cdot, \tau)$. An unstable difference scheme for loop derivatives might lead to unbounded sawtooth oscillations [252]. An empirical local linear stability analysis (described in [215]) indicates that our scheme is stable, and that the high frequency components do not generate instabilities.

3.4.2 Initialization and convergence

As in any other method, a qualitative understanding of the dynamics is a prerequisite to successful cycle searches. We start by numerical integration with the dynamical system (48). Numerical experiments reveal regions where a trajectory spends most of its life, giving us the first hunch as to how to initialize a loop. We take the FFT of some nearly recurred orbit segment and keep only the lowest frequency components. The inverse Fourier transform back to the phase space yields a smooth loop that we use as our initial guess. Since any generic orbit segment is not closed and might exhibit large gaps, the Gibbs phenomenon can take the initial loop so constructed quite far away from the region of interest. We deal with this problem by manually deforming the orbit segment into a closed loop before performing the FFT. Searching for longer cycles with multiple circuits requires more delicate initial conditions. The hope is that a few short cycles can help us establish an approximate symbolic dynamics, and guess for longer cycles can be constructed by cutting and gluing the short, known ones. For low dimensional systems, such methods yield quite good systematic initial guesses for longer cycles [37].

An alternative way to initialize the search is by utilizing adiabatic deformations of dynamics, or the homotopy evolution [43]. If the dynamical system (48) depends on a parameter μ, short cycles might survive as μ varies passing through a family of dynamical systems, giving in the process birth to new cycles through sequences of bifurcations. Most short unstable cycles vary little for small changes of μ. So, a cycle existing for parameter value μ_1 can be chosen as the initial trial loop for a nearby cycle surviving a small change $\mu_1 \to \mu_2$. In practice, one or two iterations often suffice to find the new cycle.

A good choice of the initial loop significantly expedites the computation, but there are more reasons why good initial loops are crucial. First of all, if we break the translational invariance by imposing a constraint such as $\tilde{x}_1(s_2, \tau) = c$, we have to make sure that both the initial loop and the desired cycle intersects this Poincaré plane. Hence, the initial

loop cannot be wildly different from the desired cycle. Second, in view of (58), the loop always evolves toward a local minimum of the cost functional (59), with discretization points moving along the $\tilde{v} - \lambda v$ fixed direction, determined by the initial condition. If the local minimum corresponds to a zero of the cost functional, we obtain a true cycle of the dynamical flow (48). However, if the value of cost functional is not equal to zero at the minimum while the gradient is zero, (64) yields a singular matrix $\hat{A}$. In such cases the search has to be abandoned and restarted with a new initial loop guess. In the periodic orbit searches of sect. 3.5 starting with blind initial guesses (guesses unaided by a symbolic dynamics partition), such local minima were encountered in about 30% of cases.

3.4.3 Symmetry considerations

The system under consideration often possesses certain symmetries. If this is the case, the symmetry should be both be feared for possible marginal eigendirections, and be embraced as a guide to potential simplifications of the numerical calculation.

If the dynamical system equations (48) are invariant under a discrete symmetry, the concept of fundamental domain [54, 51] can be utilized to reduce the length of the initial loop when searching for a cycle of a given symmetry. In this case, we need to discretize only an irreducible segment of the loop, decreasing significantly the dimensionality of the loop representation. Other parts of the loop are replicated by symmetry operations, with the full loop tiled by copies of the fundamental domain segment. The boundary conditions are no longer periodic, but all that we need to do is modify the cyclic terms. Instead of using M_1 and M_2 in (63), we use $M_1 Q$ and $M_2 Q^{-1}$, where Q is the relevant symmetry operation that maps the fundamental segment to the neighbor that precedes it. In this way, a fraction of the points represent the cycle with the same accuracy, speeding up the search considerably.

If a continuous symmetry is present, it may complicate the situation at first glance but becomes something that we can take advantage of after careful checking. For example, for a Hamiltonian system unstable cycles may form continuous families [121, 227], with one or more members of a family belonging to a given constant energy surface. In order to cope with the marginal eigendirection associated with such continuous family, we search for a cycle on a particular energy surface by replacing the last row of equation (64) by an energy shell constraint [38]. We put one point of the loop, say $\tilde{x}_2$, on the constant energy surface $H(\tilde{x}) = E$, and impose the constraint $\bigtriangledown H(\tilde{x}_2) \cdot \delta\tilde{x}_2 = 0$, so as to keep $\tilde{x}_2$ on the surface for all τ. The integration of (56) then automatically brings all other loop points to the same energy surface. Alternatively, we can look for a cycle of given fixed period T by fixing λ and dropping the constraint in the bottom line of (64). These two approaches are conjugate to each other, both needed in applications. In most cases, they are equivalent. One exception is the harmonic oscillator for which the oscillations have identical period but different energy. Note that in both cases the translational invariance is restored, as we have discarded the Poincaré section condition of sect. 3.2.2. As explained in [266], this causes no trouble in numerical calculations.

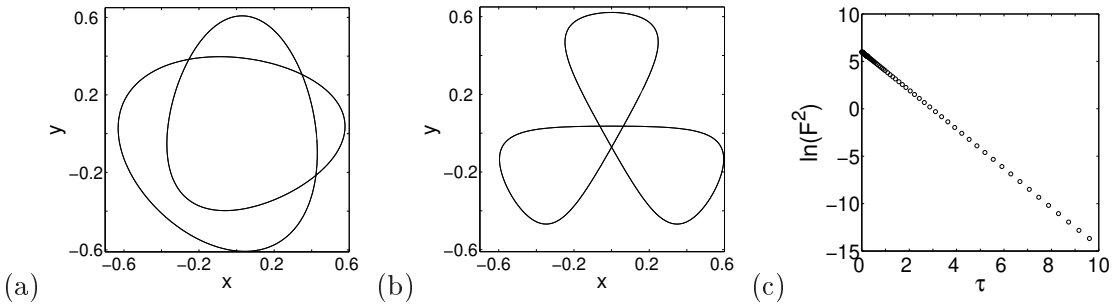

Figure 3: The Hénon-Heiles system in a chaotic region: (a) An initial loop $L(0)$, and (b) the unstable periodic orbit p of period $T = 13.1947$ reached by the Newton descent (60). (c) The exponential decrease of the cost function, $\ln(F^2) \approx -2.0502\,\tau + 6.0214$.

3.5 Applications

We have checked that the iteration of (64) yields quickly and robustly the short unstable cycles for standard models of low-dimensional dissipative flows such as the Rössler system [213]. A more daunting challenge are searches for cycles in Hamiltonian flows, and searches for spatio-temporally periodic solutions of PDEs. In all numerical examples that follow, the convergence criterion is $F^2 < 10^{-5}$.

3.5.1 Hénon-Heiles system and restricted three-body problem

First, we test the Hamiltonian version of the Newton descent derived in Sect. 3.3 by applying the method to two Hamiltonian systems, both with two degrees of freedom. In both cases, our initial loop guesses are rather arbitrary combinations of trigonometric functions. Nevertheless, the observed convergence is fast.

The Hénon-Heiles system [122] is a standard model in celestial mechanics, described by the Hamiltonian

$$H = \frac{1}{2}(p_x^2 + p_y^2 + x^2 + y^2) + x^2 y - \frac{y^3}{3}\,. \tag{65}$$

It has a time reversal symmetry and a three-fold discrete spatial symmetry. Figure 3 shows a typical application of (60), with the Newton descent search restricted to the configuration space. The initial loop, figure 3(a), is a rather coarse initial guess. We fix arbitrarily the scaling $\lambda = 2.1$, that is, we search for a cycle p of the fixed period $T_p = 13.1947$, with no constraint on the energy. Figure 3(b) shows the cycle found by the Newton descent, with energy $E = 0.1794$, and the full discrete symmetry of the Hamiltonian. This cycle persists adiabatically for a small range of values of λ; with λ changed much, the Newton descent takes the same initial loop into other cycles. Figure 3(c) verifies that the cost functional F^2 decreases exponentially with slope -2 throughout the $\tau = [0, 10]$ integration interval, as predicted by (58). The points get more and more sparse as τ increases, because our numerical implementation adaptively chooses bigger and bigger step sizes $\delta\tau$.

In the Hénon-Heiles case, the accelerations a_x, a_y depend only on the configuration variables x, y. More generally, the accelerations could also depend on $\dot{x}, \dot{y}$. Consider as an

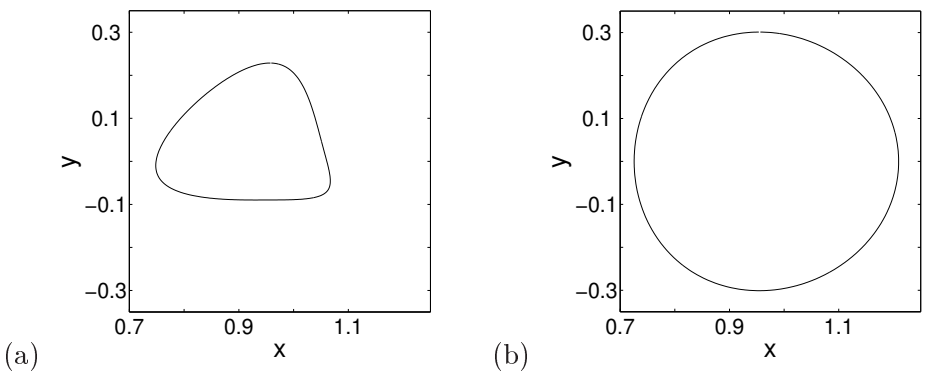

Figure 4: (a) An initial loop $L(0)$, and (b) the unstable periodic orbit p of period $T_p = 2.7365$ reached by the Newton descent (60), the restricted three body problem (66) in the chaotic regime, $\mu = 0.04$.

example the equations of motion for the restricted three-body problem [246],

$$\begin{aligned} \ddot{x} &= 2\dot{y} + x - (1-\mu)\frac{x+\mu}{r_1^3} - \mu\frac{x-1+\mu}{r_2^3}\,, \\ \ddot{y} &= -2\dot{x} + y - (1-\mu)\frac{y}{r_1^3} - \mu\frac{y}{r_2^3}\,, \end{aligned} \tag{66}$$

where $r_1 = \sqrt{(x+\mu)^2 + y^2}$, $r_2 = \sqrt{(x-1+\mu)^2 + y^2}$. These equations describe the motion of a test particle in a rotating frame under the influence of the gravitational force of two heavy bodies with masses 1 and $\mu \ll 1$ fixed at $(-\mu, 0)$ and $(1-\mu, 0)$ in the (x, y) coordinate frame. The stationary solutions of (66) are called the Lagrange points, corresponding to a circular motion of the test particle in phase with the rotation of the heavy bodies. The periodic solutions in the rotating frame correspond to periodic or quasi-periodic motion of the test particle in the inertial frame. Figure 4 shows an initial loop and the cycle to which it converges, in the rotating frame. Although the cycle looks simple, the Newton descent requires advancing in small $\delta\tau$ steps in order for the initial loop to converge to it.

In order to successfully apply the Hamiltonian version of the Newton descent (60), we have to ensure that the test particle keeps a finite distance from the origin. If a cycle passes very close to one of the heavy bodies, the acceleration can become so large that our scheme of uniformly distributing the loop points in time might fail to represent the loop faithfully. Another distribution scheme is required in this case, for example, making the density of points proportional to the magnitude of acceleration.

3.5.2 Periodic orbits of Kuramoto-Sivashinsky system

The Kuramoto-Sivashinsky equation arises as an amplitude equation for interfacial instability in a variety of contexts [155, 232]. In 1-dimensional space, it reads

$$u_t = (u^2)_x - u_{xx} - \nu u_{xxxx}, \tag{67}$$

where ν is a "super-viscosity" parameter which controls the rate of dissipation and $(u^2)_x$ is the nonlinear convection term. As ν decreases, the system undergoes a series of bifurcations, leading to increasingly turbulent, spatio-temporally chaotic dynamics.

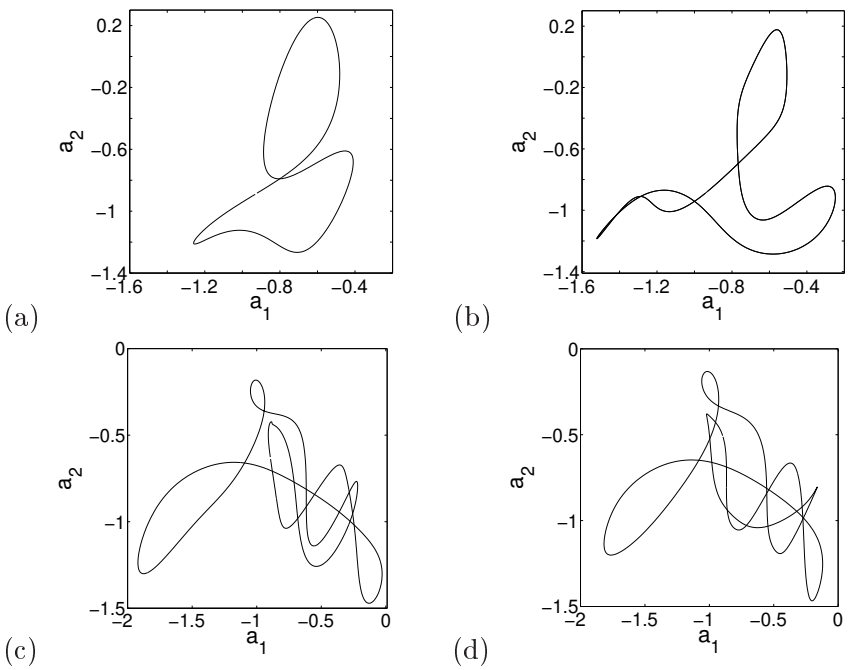

Figure 5: The Kuramoto-Sivashinsky system in a spatio-temporally turbulent regime (viscosity parameter $\nu = 0.015$, $d = 32$ Fourier modes truncation). (a) An initial guess L_1, and (b) the periodic orbit p_1 of period $T_1 = 0.744892$ reached by the Newton descent. (c) Another initial guess L_2, and (d) the resulting periodic orbit p_2 of period $T_2 = 1.184668$.

If we impose the periodic boundary condition $u(t, x + 2\pi) = u(t, x)$ and choose to study only the odd solutions $u(-x, t) = -u(x, t)$, $u(x, t)$ can be expanded in a discrete spatial Fourier series [37],

$$u(x,t) = i \sum_{k=-\infty}^{\infty} a_k(t) e^{ikx}, \tag{68}$$

where $a_{-k} = -a_k \in \mathbb{R}$. In terms of the Fourier components, PDE (133) becomes an infinite ladder of ODEs,

$$\dot{a_k} = (k^2 - \nu k^4) a_k - k \sum_{m=-\infty}^{\infty} a_m a_{k-m} \,. \tag{69}$$

In numerical simulations we work with the Galerkin truncations of the Fourier series since in the neighborhood of the strange attractor the magnitude of a_k decreases very fast with k, high frequency modes playing a negligible role in the asymptotic dynamics [195]. In this way Galerkin truncations reduce the dynamics to a finite but large number of ODEs. We work with $d = 32$ dimensions in our numerical calculations. In ref. [37], multipoint shooting has been successfully applied to obtain periodic orbits close to the onset of spatiotemporal chaos ($\nu = 0.03$). In this regime, our method is so stable that big time steps $\delta\tau$ can be employed even at the initial guesses, leading to extremely fast convergence. We attribute this robustness to the simplicity of the structure of the attractor at high viscosity values.

The challenge comes with decreasing ν, with the dynamics turning more and more turbulent. Already at $\nu = 0.015$, the system is moderately turbulent and the phase space portraits of the dynamics reveal a complex labyrinth of "eddies" of different scales and

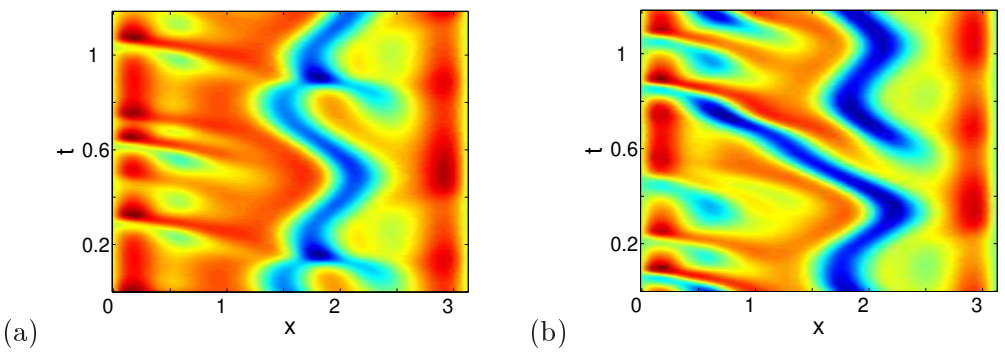

Figure 6: Level plot of the space-time evolution $u(x,t)$ for the two spatio-temporally periodic solutions of figure 5: (a) the evolution of p_1, with start of a repeat after the cycle period $T_1 = 0.744892$, and (b) one full period $T_2 = 1.184668$ in the evolution of p_2.

orientations. While the highly unstable nature of orbits and intricate structure of the invariant set hinder applications of conventional cycle-search routines, in this setting our variational method shines through. We design rather arbitrary initial loops from numerical trajectory segments, and the calculation proceeds as before, except that now a small $\delta\tau$ has to be used initially to ensure numerical stability. Topologically different loops are very likely to result in different cycles, while some initial loop guesses may lead to local nonzero minima of the cost functional F^2. As explained in Sect. 3.4, in such cases the method diverges, and the search is restarted with a new initial loop guess.

Two initial loop guesses are displayed in figure 5, along with the two periodic orbits detected by the Newton descent. In discretization of the initial loops, each point has to be specified in all d dimensions; here the coordinates $\{a_1, a_2\}$ are picked so that topological similarity between initial and final loops is visually easy to identify. Other projections from $d = 32$ dimensions to subsets of 2 coordinates appear to make the identification harder, if not impossible. In both calculations, we molded segments of typical trajectories into smooth closed loops by the Fourier filtering method of Sect. 3.4. As the desired orbit becomes longer and more complex, more sampling points are needed to represent the loop. We use $N = 512$ points to represent the loop in the (a)-(b) case and $N = 1024$ points in the (c)-(d) case. The space-time evolution of $u(x,t)$ for these two unstable spatio-temporally periodic solutions is displayed in figure 6. As $u(x,t)$ is antisymmetric on $[-\pi, \pi]$, it suffices to display the solutions on the $x \in [0, \pi]$ interval.

3.6 Summary

In order to cope with the difficulty of finding periodic orbits in high-dimensional chaotic flows, we have devised the *Newton descent method*, an infinitesimal step variant of the damped Newton-Raphson method. Our main result is the PDE (56) which solves the variational problem of minimizing the cost functional (59). This equation describes the fictitious time τ flow in the space of loops which decreases the cost functional at uniform exponential rate, see (58). Variants of the method are presented for special classes of systems, such as the Hamiltonian systems. An efficient integration scheme for the PDE is devised and tested on the Kuramoto-Sivashinsky system, the Hénon-Heiles system and the

restricted three-body problem.

Our method uses information from a large number of points in phase space, with the global topology of the desired cycle protected by insistence on smoothness and a uniform discretization of the loop. The method is quite robust in practice.

The numerical results presented here are only a proof of principle. We do not know to what periodic orbit the flow (56) will evolve for a given dynamical system and a given initial loop. Empirically, the flow goes toward the "nearest" periodic orbit, with the largest topological resemblance. Each particular application requires much work in order to elucidate and enumerate the relevant topological structures. The hope is that the short spatiotemporally periodic solutions revealed by the Newton descent searches will serve as the basic building blocks for systematic investigations of chaotic and perhaps even "turbulent" dynamics.