

Computation of stable and unstable manifolds of hyperbolic trajectories in two-dimensional, aperiodically time-dependent vector fields

Ana M. Mancho^a, Des Small^a, Stephen Wiggins^{a,*}, Kayo Ide^b

^a School of Mathematics, University of Bristol, University Walk, Bristol, BS8 1TW, UK

^b Department of Atmospheric Sciences, and Institute of Geophysics and Planetary Physics, UCLA, Los Angeles, CA 90095-1565, USA

Received 14 October 2002; received in revised form 29 January 2003; accepted 31 March 2003

Communicated by E. Kostelich

Abstract

In this paper, we develop two accurate and fast algorithms for the computation of the stable and unstable manifolds of hyperbolic trajectories of two-dimensional, aperiodically time-dependent vector fields. First we develop a benchmark method in which all the trajectories composing the manifold are integrated from the neighborhood of the hyperbolic trajectory. This choice, although very accurate, is not fast and has limited usage. A faster and more powerful algorithm requires the insertion of new points in the manifold as it evolves in time. Its numerical implementation requires a criterion for determining when to insert those points in the manifold, and an interpolation method for determining where to insert them. We compare four different point insertion criteria and four different interpolation methods. We discuss the computational requirements of all of these methods. We find two of the four point insertion criteria to be accurate and robust. One is a variant of a criterion originally proposed by Hobson. The other is a slight variant of a method due to Dritschel and Ambaum arising from their studies of contour dynamics. The preferred interpolation method is also due to Dritschel. These methods are then applied to the computation of the stable and unstable manifolds of the hyperbolic trajectories of several aperiodically time-dependent variants of the Duffing equation.

© 2003 Elsevier B.V. All rights reserved.

PACS: 02.60.-x; 05.45.-a; 05.60.Cd; 47.52.+j

Keywords: Nonautonomous systems; Aperiodically time-dependent vector fields; Stable and unstable manifolds of hyperbolic trajectories; Numerical methods

1. Introduction

It is an accepted fact in dynamical systems theory that for autonomous and time-periodic vector fields, as well as maps, knowledge of the stable and unstable manifolds of hyperbolic equilibria or hyperbolic periodic orbits may play a central role in understanding many global, dynamical issues. For a given dynamical system, the only general way of studying such stable and unstable manifolds is by computing them numerically. Consequently, a

* Corresponding author.

number of people have developed algorithms for computing the stable and unstable manifolds for autonomous and time-periodic vector fields, as well as for maps. For example, see [1,10,17,18,24–26,31]. An advantage offered by such vector fields and maps is stationarity. In particular, if we consider the familiar case of a two-dimensional, time-periodic vector field with period \mathcal{T} and define a Poincaré section by fixing some $t = t_0$. Because the stable and unstable manifolds in any Poincaré section at $t_0 - n\mathcal{T}$ ($n \in \mathbb{Z}$) are stationary, it is relatively simple to numerically compute them.

In this paper, we consider a new aspect of this subject: the computation of the stable and unstable manifolds of hyperbolic trajectories of two-dimensional *aperiodically time-dependent* vector fields (by “aperiodic” we simply mean “not periodic”). As in the situation for autonomous and time-periodic vector fields as well as maps, we expect that the stable and unstable manifolds should play an important role in understanding a variety of global, dynamical issues. Moreover, study of their evolution in general systems also requires numerical computation.

The dynamics generated by aperiodically time-dependent vector fields have not received as much attention in the dynamical systems community as time-periodic vector fields or maps. Part of the reason is that many of the standard dynamical systems techniques and approaches (such as the notion of a Poincaré map) do not apply in the same way when the time-dependence of the vector field is aperiodic. Nevertheless, general frameworks for dealing with different aspects of aperiodically time-dependent vector fields have been around for some time; notably, the skew-product flow approach (see e.g. [29]), the cocycle approach (see e.g. [16]), and the approach of dynamics generated by an infinite sequence of maps (see e.g. [15]). Within these frameworks fundamental issues such as attractivity, stability, and asymptotic behavior have been formulated (see e.g. [16,19,23,29]). In recent years, there has been the beginnings of an effort to develop the building blocks of a geometric theory for the dynamics generated by aperiodically time-dependent vector fields. Aspects that have been studied include bifurcation theory [19], normal form theory [30], shadowing lemmas [2,23] and chaos [20,22,28,32–34].

While there are some similarities in the computation of the stable and unstable manifolds of hyperbolic trajectories in two-dimensional aperiodically time-dependent vector fields with their counterparts in time-periodic vector fields, autonomous vector fields, and maps, there are also some fundamental differences. Perhaps the biggest difference is the fact that the dynamics of an aperiodically time-dependent vector field are generated by a *non-stationary process*. Practically, this means that the unstable (resp., stable) manifold at any fixed time depends on the evolution of trajectories from the infinite past (resp., future). In computing the stable and unstable manifolds of hyperbolic periodic orbits of maps (or time-periodic vector fields) once a segment of the manifold is computed further iterates of the map do not alter this segment. However, when there is no periodicity in the vector field, as time evolves *the entire length of the computed manifold* also evolves. Consequently, the numerical computations are much more intensive, from the point of view of time, memory, and storage, and a new point of view must be adopted. A more detailed comparison of issues involved with the computation of the stable and unstable manifolds of hyperbolic trajectories in time-periodic vector fields, versus time aperiodic vector fields, is given in [Appendix A](#).

The goal of this paper is to develop a practical and efficient numerical method for computing the stable and unstable manifolds of a hyperbolic trajectory for a two-dimensional, aperiodically time-dependent vector field. The outline is as follows. In [Section 2](#), we discuss the notion of hyperbolic trajectories and the existence of stable and unstable manifolds of hyperbolic trajectories for aperiodically time-dependent vector fields. This material is not new, but it is outside the mainstream of applied dynamical systems theory and therefore we feel that a brief review is necessary. With these notions in hand, we then discuss our numerical approach to computing the stable and unstable manifolds of hyperbolic trajectories. The numerical method involves two key features: a criterion for determining when new points must be added to the manifold, and a way of inserting these new points (interpolation). In [Section 3](#), we discuss several insertion criteria and interpolation methods. In addition to the “distance between points” and “Hobson” criteria (related to curvature [10]) which have been used in the past, we present two criteria:

a variant of Hobson’s criteria (mentioned by Hobson [10], but not used in his work), and a method due to Dritschel and Ambaum (which uses local curvature to determine a desired point density [7]) from the study of contour advection. We present a detailed comparison between these four criteria. With a point insertion criteria in hand, we then explore interpolation methods. We consider linear interpolation, Lagrange polynomials, cubic splines, and a more geometrical method due to Dritschel developed in his studies of contour dynamics. We present a detailed comparison between these four methods. In Section 4, we summarize these results and conclude that there are two viable methods for practical use. Either the variant of Hobson’s method or the Dritschel–Ambaum method can be used as a point insertion criterion in conjunction with Dritschel’s interpolation method. For vector fields where the time-dependence is strongly spatially variable the Dritschel–Ambaum approach may offer some advantages as it incorporates a point redistribution technique. This section serves as the main conclusion of this paper. In Section 5, we apply the optimal methods to several examples of aperiodically time-dependent vector fields to compute the stable and unstable manifolds of a hyperbolic trajectory. Appendix A describes computation of stable and unstable manifolds of hyperbolic trajectories in time-periodic vector fields and how the procedure must be fundamentally different for aperiodically time-dependent vector fields. Appendix B describes our “benchmark method”.

2. Background and set-up

We consider a two-dimensional velocity field of the form:

$$\dot{x} = v(x, t), \quad x \in \mathbb{R}^2, \quad t \in \mathbb{R}. \quad (1)$$

We assume that the vector field is C^r ($r \geq 1$) in x , and continuous in t . This will allow for unique solutions to exist, and also permit linearization. The dynamics of aperiodically time-dependent vector fields are fundamentally different, as well as less familiar, than those of time-periodic vector fields and maps. For this reason, we will need to develop some of the basic background in more detail than for the latter cases.

2.1. Hyperbolic trajectories

Hyperbolicity of a trajectory is a property of the linear system obtained by linearizing (1) about the trajectory. The dynamics of the linear system are then described by the notion of an *exponential dichotomy*, which we now define, and for which the standard references are [4,9,21].

Definition 2.1 (Exponential dichotomy). Consider the following linear ordinary differential equation with time-dependent coefficients:

$$\dot{\xi} = A(t)\xi, \quad \xi \in \mathbb{R}^2, \quad (2)$$

where $A(t) \in \mathbb{R}^{2 \times 2}$ is a continuous function of t . Suppose $X(t) \in \mathbb{R}^{2 \times 2}$ is the *fundamental solution matrix* of (2), i.e., for any initial condition ξ_0 , $\xi(t) = X(t)\xi_0$ is the solution of (2) passing through ξ_0 at $t = 0$ and $X(0) = \text{id}$. Let $\|\cdot\|$ denote a norm on \mathbb{R}^2 . Then (2) is said to possess an *exponential dichotomy* if there exists a projection operator $P \in \mathbb{R}^{2 \times 2}$, $P^2 = P$, and constants $K_1, K_2, \lambda_1, \lambda_2 > 0$, such that

$$\|X(t)PX^{-1}(\tau)\| \leq K_1 \exp(-\lambda_1(t - \tau)), \quad t \geq \tau, \quad (3)$$

$$\|X(t)(\text{id} - P)X^{-1}(\tau)\| \leq K_2 \exp(\lambda_2(t - \tau)), \quad t \leq \tau. \quad (4)$$

With the notion of exponential dichotomy in hand we can now define the notion of a hyperbolic trajectory of a general time-dependent vector field.

Definition 2.2 (Hyperbolic trajectory). Let $\gamma(t)$ denote a trajectory of the vector field (1). Then $\gamma(t)$ is said to be a *hyperbolic trajectory* if the associated linearized system:

$$\dot{\xi} = D_x v(\gamma(t), t)\xi \tag{5}$$

has an exponential dichotomy.

We now describe the geometry behind the notion of exponential dichotomy in terms of the *extended phase space*:

$$\mathcal{E} \equiv \{(x, t) \in \mathbb{R}^2 \times \mathbb{R}\}, \tag{6}$$

i.e., we append the dependent variable t to the phase space. We consider the vector field defined on the extended phase space by appending the (trivial) evolution of t to (1) as follows:

$$\dot{x} = v(x, t), \quad \dot{t} = 1 \tag{7}$$

and the hyperbolic trajectory in the extended phase space \mathcal{E} is denoted by

$$\Gamma(t) = (\gamma(t), t).$$

We define a *time slice* of the extended phase space \mathcal{E} as follows:

$$\Sigma_\tau \equiv \{(x, t) \in \mathcal{E} | t = \tau\}. \tag{8}$$

Then in the extended phase space, the hyperbolic trajectory $\Gamma(t)$ intersects Σ_τ in the unique point $\gamma(\tau)$.

In Definition 2.1, suppose the projection operator P has rank 1, the case of interest for us since we are concerned with saddle-like behavior. Then (3) implies that on the time slice Σ_τ , there is a one-dimensional subspace of \mathbb{R}^2 , $E^s(\tau)$, corresponding to initial conditions of trajectories of the linearized equations (5) that decay to zero at an exponential rate as $t \rightarrow \infty$. Similarly, (4) implies that on the time slice Σ_τ , there is a one-dimensional subspace of \mathbb{R}^2 , $E^u(\tau)$, corresponding to initial conditions of trajectories of the linearized equations (5) that decay to zero at an exponential rate as $t \rightarrow -\infty$. Moreover, the angle between $E^s(\tau)$ and $E^u(\tau)$ is bounded away from zero for all τ . We illustrate this geometrically in Figs. 1 and 2.

2.2. Stable and unstable manifolds of hyperbolic trajectories

We can now state the result that tangent to these linearized, time-varying eigenspaces we have invariant manifolds for the full nonlinear vector field.

Let $D_\rho(\tau) \in \Sigma_\tau$ denote the ball of radius ρ centered at $\gamma(\tau)$. Then

$$\mathcal{N}_\rho(\Gamma(t)) \equiv \bigcup_{\tau \in \mathbb{R}} (D_\rho(\tau), \tau)$$

is a *tubular neighborhood* of $\Gamma(t)$ in \mathcal{E} . We have the following theorem.

Theorem 2.1 (Local stable and unstable manifolds). *For the set-up and hypotheses described above, in the extended phase space \mathcal{E} there exists a two-dimensional C^r manifold $W_{loc}^s(\Gamma(t)) \subset \mathcal{E}$, and a two-dimensional C^r manifold $W_{loc}^u(\Gamma(t)) \subset \mathcal{E}$ and ρ_0 sufficiently small such that for $\rho \in (0, \rho_0)$:*

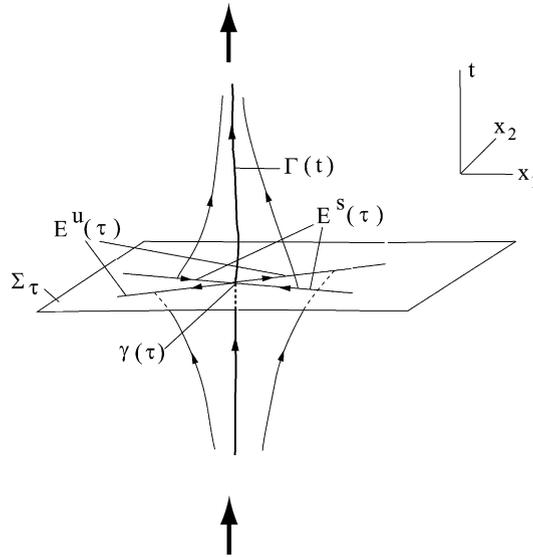


Fig. 1. Geometry of the stable and unstable subspaces of the linearized system associated with the hyperbolic trajectory, both in the extended phase space and on a time slice. (The arrows are meant to indicate that the structure extends for all time.) In this figure, we show trajectories with initial (resp., final) conditions in $E^s(\tau)$ (resp., $E^u(\tau)$).

- (1) $W_{loc}^s(\Gamma(t))$, the local stable manifold of $\Gamma(t)$, is invariant under the forward time evolution generated by (1),
 $W_{loc}^u(\Gamma(t))$, the local unstable manifold of $\Gamma(t)$, is invariant under the backward time evolution generated by (1).
- (2) $W_{loc}^s(\Gamma(t))$ and $W_{loc}^u(\Gamma(t))$ intersect along $\Gamma(t)$, and the angle between the manifolds is bounded away from zero uniformly for all $t \in \mathbb{R}$.

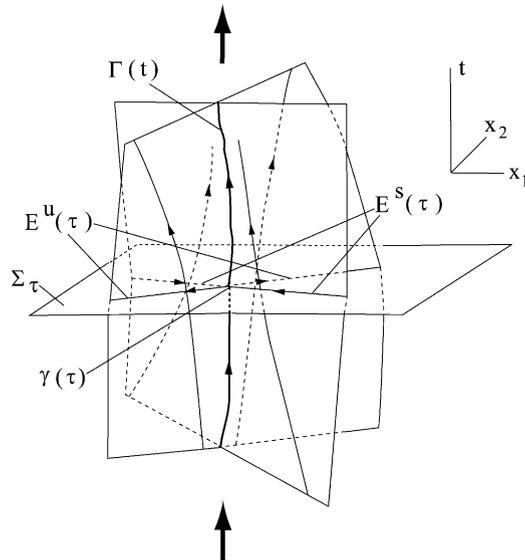


Fig. 2. Geometry of the stable and unstable subspaces of the linearized system associated with the hyperbolic trajectory, both in the extended phase space and on a time slice. (The arrows are meant to indicate that the structure extends for all time.) In this figure, we not only show trajectories with initial (resp., final) conditions in $E^s(\tau)$ (resp., $E^u(\tau)$), but also the stable and unstable subspaces in the full extended subspace. Note how they may “twist”.

- (3) Every trajectory on $W_{loc}^s(\Gamma(t))$ can be continued to the boundary of $\mathcal{N}_\rho(\Gamma(t))$ in backward time, and every trajectory on $W_{loc}^u(\Gamma(t))$ can be continued to the boundary of $\mathcal{N}_\rho(\Gamma(t))$ in forward time.
- (4) Trajectories starting on $W_{loc}^s(\Gamma(t))$ at time $t = \tau$ approach $\Gamma(t)$ at an exponential rate $e^{-\lambda^s(t-\tau)}$ as $t \rightarrow \infty$ for some constant $\lambda^s > 0$. Similarly, trajectories starting on $W_{loc}^u(\Gamma(t))$ at time $t = \tau$ approach $\Gamma(t)$ at an exponential rate $e^{-\lambda^u|t-\tau|}$ as $t \rightarrow -\infty$, for some constant $\lambda^u > 0$.
- (5) Any trajectory in $\mathcal{N}_\rho(\Gamma(t))$ not on either $W_{loc}^s(\Gamma(t))$ or $W_{loc}^u(\Gamma(t))$ will leave $\mathcal{N}_\rho(\Gamma(t))$ in both forward and backward time.

Proof. In some sense this theorem has been known for some time, although the autonomous or time-periodic version is much more widely known. The theorem can be obtained from simple modifications of results found in [3,8]. The theorem in this form can be found in [14], see also [35]. A discrete time version can be found in [15]. Less well-known work that gives similar results is [5,12]. □

This theorem provides the mathematical basis for computing the global stable and unstable manifolds of $\Gamma(t)$ as follows. Given a piece of the local stable manifold $W_{loc}^s(\Gamma(t))$ on the time slice Σ_t , a piece of the global stable manifold $W^s(\Gamma(\tau))$ on the time slice Σ_τ can be obtained for $\tau < t$ by evolving the trajectories starting on the piece of $W_{loc}^s(\Gamma(t))$ backward in time. Similarly, the forward evolution of the trajectories starting on a piece of the local unstable manifold $W_{loc}^u(\Gamma(t))$ on Σ_t gives a piece of $W^u(\Gamma(\tau))$ on Σ_τ for $\tau > t$. We illustrate this situation geometrically in Fig. 3.

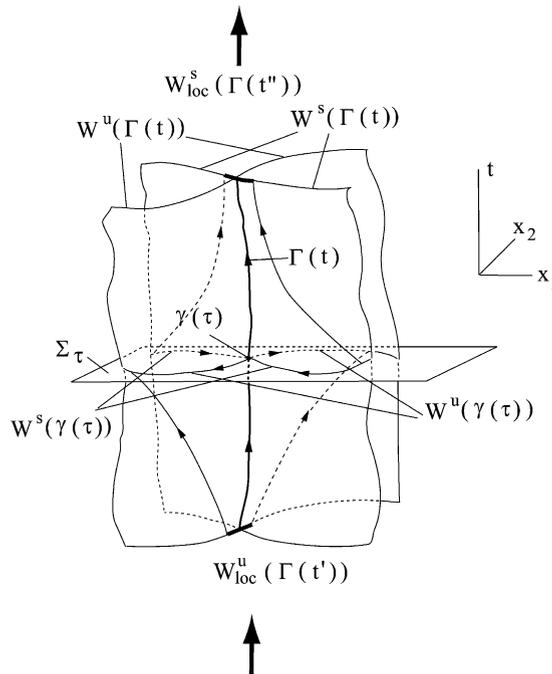


Fig. 3. Geometry of the stable and unstable manifolds of the nonlinear system associated with the hyperbolic trajectory, both in the extended phase space and on a time slice. (The arrows are meant to indicate that the structure extends for all time.) The figure also indicates the computation of pieces of the global stable (resp., unstable) manifolds on a particular time slice from a segment of the local stable (resp., unstable) manifold on a later (resp., earlier) time slice. The evolution of a segment of $W_{loc}^u(\Gamma(t))$ (shown in bold) from the time slice $\Sigma_{t'}$ to the time slice Σ_τ is shown. Similarly, the evolution of a segment of $W_{loc}^s(\Gamma(t))$ (shown in bold) from the time slice $\Sigma_{t''}$ to the time slice Σ_τ is shown.

2.3. Computation of the stable and unstable manifold of hyperbolic trajectories

Now we can turn to the main point of this paper: a numerical approach to computing the stable and unstable manifolds of a hyperbolic trajectory for a two-dimensional, aperiodically time-dependent vector field.

First, we make two *assumptions*. One is that we know a hyperbolic trajectory, $\Gamma(t)$, of (1). This is actually not a trivial assumption. For autonomous and time-periodic vector fields as well as maps, there is a variety of numerical methods for locating hyperbolic periodic orbits, see e.g. [25,26]. However, it is only in recent years that numerical algorithms for computing hyperbolic trajectories of vector fields with aperiodic time-dependence have been developed, see [11,13]. In addition, for the purposes of the exposition of the method, we temporarily assume that we also know $W_{\text{loc}}^u(\Gamma(t))$ and $W_{\text{loc}}^s(\Gamma(t))$ in the small neighborhood of $\Gamma(t)$. We show how to obtain their approximations numerically later in this section.

Next, we establish some notation associated with the time-evolution map for the aperiodically time-dependent vector field. We let $\{t_k | k \in \mathbb{Z}\}$ be a sequence of times with $t_k < t_{k+1}$ for any k . The corresponding time slices are $\Sigma_{t_k} = \{(x, t) \in \mathcal{E} | t = t_k\}$.

The time increment $dT_k \equiv t_{k+1} - t_k$ can be, but is not necessarily, constant. Now, let $x(t, t_0, x_0)$ be the trajectory of (1) passing through the point x_0 at $t = t_0$. For each $k \in \mathbb{Z}$, we denote the forward time-evolution map between Σ_{t_k} and $\Sigma_{t_{k+1}}$ by

$$f_k : \Sigma_{t_k} \rightarrow \Sigma_{t_{k+1}}, \quad (x_k, t_k) \mapsto (x_{k+1}, t_{k+1}) = f_k(x_k, t_k) \equiv (x(t_{k+1}, t_k, x_k), t_{k+1}) \quad (9)$$

and the backward time-evolution map by

$$f_k^{-1} : \Sigma_{t_{k+1}} \rightarrow \Sigma_{t_k}, \quad (x_{k+1}, t_{k+1}) \mapsto (x_k, t_k) = f_k^{-1}(x_{k+1}, t_{k+1}) \equiv (x(t_k, t_{k+1}, x_{k+1}), t_k). \quad (10)$$

The subscript k of x_k here corresponds to the time slice, not the index of a coordinate, in the two-dimensional space. By the definition of a hyperbolic trajectory, we have $f_k(\Gamma(t_k)) = \Gamma(t_{k+1})$ and $f_k^{-1}(\Gamma(t_{k+1})) = \Gamma(t_k)$.

With these assumptions and notations in hand, we derive expressions for the global stable and unstable manifolds of $\Gamma(t_i)$ at $t = t_i$ that are amenable to computation. The technical expressions are given by

$$W^u(\Gamma(t_i)) = \lim_{n \rightarrow \infty} (f_{i-1} \circ \cdots \circ f_{i-n}(W_{\text{loc}}^u(\Gamma(t_{i-n})))), \quad (11)$$

$$W^s(\Gamma(t_i)) = \lim_{n \rightarrow \infty} (f_{i+1}^{-1} \circ \cdots \circ f_{i+n-1}^{-1}(W_{\text{loc}}^s(\Gamma(t_{i+n}))). \quad (12)$$

By letting t_i and t_n be τ and t , we recover the results in Section 2.2. Moreover, this expression gains some familiarity by relating it to the similar expressions for time-periodic vector fields as we show in Appendix A. Still, these formulae probably appear a bit incomprehensible. Now we explain in detail what they mean.

First, we consider the unstable manifold. The global unstable manifold $W^u(\Gamma(t_i))$ on Σ_{t_i} (left-hand side) can be obtained by the known local unstable manifold $W_{\text{loc}}^u(\Gamma(t_{i-n}))$ on a *past* time slice $\Sigma_{t_{i-n}}$ and forward time-evolution map f_k between the adjacent time slices for $i - n \leq k \leq i - 1$ (right-hand side). Theorem 2.1 says that, if $W_{\text{loc}}^u(\Gamma(t_{i-n})) \subset f_{i-(n+1)}(W_{\text{loc}}^u(\Gamma(t_{i-(n+1)})))$, then

$$f_{i-1} \circ \cdots \circ f_{i-n}(W_{\text{loc}}^u(\Gamma(t_{i-n}))) \subset f_{i-1} \circ \cdots \circ f_{i-n} \circ f_{i-(n+1)}(W_{\text{loc}}^u(\Gamma(t_{i-(n+1)}))). \quad (13)$$

Taking the limit $n \rightarrow \infty$ gives the entire global unstable manifold on the time slice Σ_{t_i} as illustrated in Fig. 4. These expressions give rise to a computational approach for obtaining $W^u(\Gamma(t_i))$ on the time slice Σ_{t_i} . In Section 3, we develop a practical and efficient method to compute this expression.

Next we consider the global stable manifold $W^s(\Gamma(t_i))$ on Σ_{t_i} . The formula is similar to that of $W^u(\Gamma(t_i))$ with the time direction reversed. The global stable manifold $W^s(\Gamma(t_i))$ on Σ_{t_i} (left-hand side) can be obtained by the

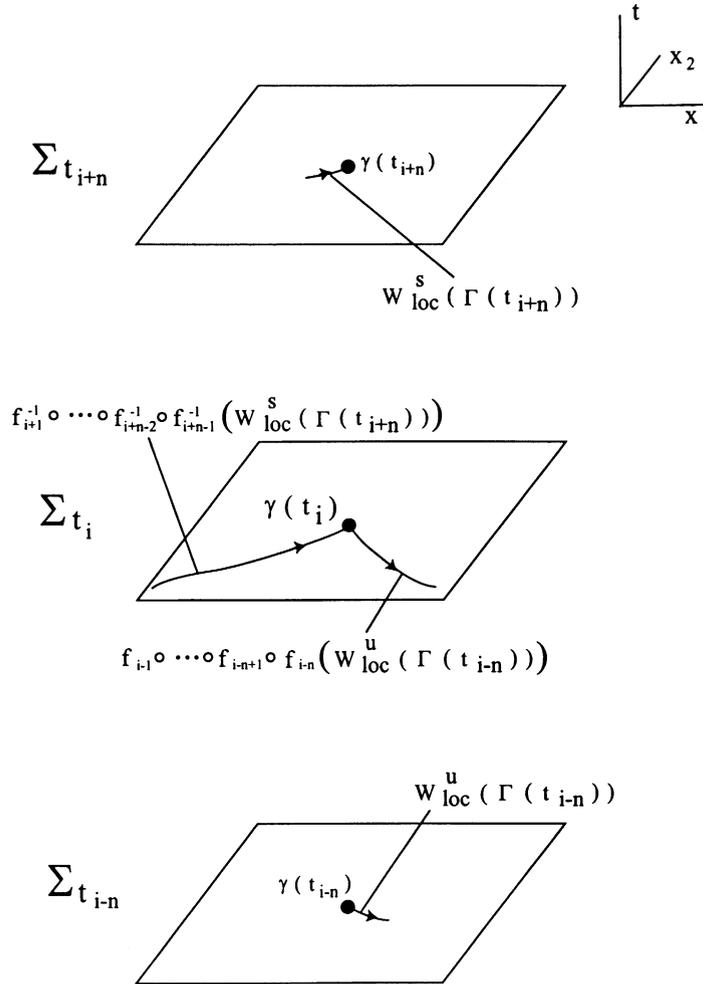


Fig. 4. A geometrical illustration of the approach to computing segments of the global stable (resp., unstable) manifold of a hyperbolic trajectory on a given time slice from a segment of the local unstable manifold on a later (resp., earlier) time slice.

known local stable manifold $W_{loc}^s(\Gamma(t_{i+n}))$ on the *future* time slice $\Sigma_{t_{i+n}}$ and backward time-evolution map f_k^{-1} between the adjacent time slices for $i+n > k > i$ (right-hand side). **Theorem 2.1** says that, if $W_{loc}^s(\Gamma(t_{i+n})) \supset f_{i+(n+1)}(W_{loc}^s(\Gamma(t_{i+(n+1)})))$, then

$$f_{i+1}^{-1} \circ \dots \circ f_{i+n}^{-1}(W_{loc}^s(\Gamma(t_{i+n}))) \subset f_{i+1}^{-1} \circ \dots \circ f_{i+n}^{-1} \circ f_{i+(n+1)}^{-1}(W_{loc}^s(\Gamma(t_{i+(n+1)}))). \quad (14)$$

Taking the limit $n \rightarrow \infty$ gives the entire global stable manifold on the time slice Σ_{t_i} .

We summarize the method for computing a segment of the unstable manifold at time t_i in the following three steps. The method for computing a segment of the stable manifold is analogous with the reversed time direction.

- Step 1. Choose an initial time t_{i-n} , $n > 0$. The larger the choice of n , the longer the segment of unstable manifold that can be computed on the later time slice Σ_{t_i} (subject to control of numerical errors).
- Step 2. On the time slice $\Sigma_{t_{i-n}}$, choose a small segment of $W_{loc}^u(\Gamma(t_{i-n}))$.
- Step 3. Evolve this segment forward in time to the time slice Σ_{t_i} .

The numerical implementation of this algorithm gives rise to the following two difficulties that must be overcome:

- (1) We have assumed that $\Gamma(t_{i-n})$ and $W_{\text{loc}}^u(\Gamma(t_{i-n}))$ were known in Step 2. How do we actually obtain them?
- (2) Given a segment of $W_{\text{loc}}^u(\Gamma(t_{i-n}))$, we cannot computationally realize it as a continuous curve. Rather, we approximate it by closely spaced points. The more closely spaced the points, the better the approximation to a continuous curve. Nevertheless, as time evolves, significant “gaps” may develop between points that were closely spaced “neighbors” on the original time slice. How do we treat the issues that arise as a result of this phenomenon?

The first difficulty can be addressed as follows. Numerical algorithms for computing a hyperbolic trajectory $\gamma(t)$ can be found in [11,13]. As discussed in Section 2.2, approximating $W_{\text{loc}}^u((\gamma(t), t))$ by $E^u(t)$ is often sufficient in practice. Obtaining $E^u(t)$ generally requires the numerical solution of (5); a numerical algorithm for computing it can be found also in [11]. We let $x_{\text{DHT}}(t)$ denote the (numerically obtained) hyperbolic trajectory where the subscript stands for *distinguished hyperbolic trajectory* (DHT). In addition, we let $e^s(t)$ and $e^u(t)$ denote unit vectors in the direction of $E^s(t)$ and $E^u(t)$, respectively. Then, an approximation $W_{\text{approx}}^s((x_{\text{DHT}}(t), t))$ to $W_{\text{loc}}^s((\gamma(t), t))$ is given by the straight line-segment between $x_{\text{DHT}}(t)$ and

$$x_W^s(t) = x_{\text{DHT}}(t) + \lambda e^s(t). \quad (15)$$

Therefore, the first difficulty amounts to the choice of λ which varies from case to case. Section 3 provides some guidance.

Similarly, $W_{\text{approx}}^u((x_{\text{DHT}}(t), t))$ defined by the straight line-segment between $x_{\text{DHT}}(t)$ and

$$x_W^u(t) = x_{\text{DHT}}(t) + \lambda e^u(t) \quad (16)$$

gives an approximation to $W_{\text{loc}}^u((\gamma(t), t))$.

The second difficulty is the main subject of the next section.

3. Numerical implementation

In this section, we discuss how to numerically implement the computation of the stable and unstable manifolds of a hyperbolic trajectory. We follow the three-step algorithm described in Section 2.3. Henceforth, we will restrict our discussion to the computation of the unstable manifold. The discussion for the stable manifold is analogous to that for the unstable manifold with the direction of time reversed.

In Step 1, we choose $t_0 (=t_{i-n}) \equiv 0$ and $t_n (=t_i) \equiv T$. In any implementation, we are inevitably restricted to a finite time range T and $n + 1$ time slices. The time increment, $dT = T/n$, can be constant for simplicity.

In Step 2 on the time slice Σ_{t_0} , Eq. (16) provides an approximation $W_{\text{approx}}^u((x_{\text{DHT}}(t_0), t_0))$ to the local unstable manifold $W_{\text{loc}}^u(\Gamma(t_0))$ for a given λ . A number of points are placed over the length λ on $W_{\text{approx}}^u((x_{\text{DHT}}(t_0), t_0))$ so that the gaps between adjacent points are small.

In Step 3, the points starting on $W_{\text{approx}}^u((x_{\text{DHT}}(t_0), t_0))$ evolve according to the forward time-evolution given by (13).

The second difficulty discussed in Section 2.3 arises in Step 3. As the segment of unstable manifold stretches, visible gaps between the adjacent trajectories may develop. There are two approaches with which we can deal with this:

- (1) On Σ_{t_0} , place the points with such a distribution that visible gaps do not develop on Σ_{t_n} .
- (2) Insert additional points on the time slice Σ_{t_k} for $0 \leq k < n$ according to an interpolation scheme.

The first approach, which forms the basis of our “benchmark” calculation as described in [Appendix B](#), is doomed to failure. Limited machine precision means that there is a maximum number M_{\max} of points that can be numerically distinguishable over a given segment length λ along $W_{\text{approx}}^u((x_{\text{DHT}}(t_0), t_0))$ on Σ_{t_0} , and it may happen that those points do develop visible gaps at Σ_{t_n} for long time ranges T . Moreover, the computer time required to achieve a manifold of sufficient length may be too large to make it a practical method in applied problems. Nevertheless, if the points can be sufficiently closely spaced for a short time, then a very accurate manifold can be computed by this approach.

The second approach is computationally more practical. The insertions take place only on the discrete time slice Σ_{t_k} for $0 \leq k < n$ and requires various considerations, as we shall see.

An insertion technique can be developed as follows. On the time slice Σ_{t_k} , we examine *whether or not* a new point should be inserted in the neighborhood of two nodes. This requires an insertion criterion. Once the insertion is found to be necessary, then we determine the coordinates of the point to be inserted *on the previous time slice* $\Sigma_{t_{k-1}}$. This requires a method for interpolation between points on the manifold. The new point is then integrated up to the time slice Σ_{t_k} . This process is repeated until the set of points on Σ_{t_k} do not require any further insertion.

The rest of this section is devoted to a discussion of various insertion criteria ([Section 3.2](#)) and interpolation methods ([Section 3.3](#)). To select the optimal insertion technique, we use a dynamical system that has been studied intensively. The performance of the insertion techniques is evaluated based on accuracy in comparison with the benchmark calculation.

3.1. Set-up for evaluation

As discussed in [Section 2](#), $W_{\text{approx}}^u((x_{\text{DHT}}(t), t))$ relies on $x_{\text{DHT}}(t)$ as well as $e^u(t)$ and $e^s(t)$, which are in general obtained numerically. However, there are cases where $x_{\text{DHT}}(t)$, $e^u(t)$ and $e^s(t)$ can be obtained analytically up to some order in a small parameter $\epsilon \in \mathbb{R}$. This eliminates the extra layer of difficulty and puts the main focus on the purpose of this paper: the numerical computation of the stable and unstable manifolds of hyperbolic trajectories in aperiodically time-dependent vector fields. We use the periodically forced Duffing equation initially for this purpose, where $x_{\text{DHT}}(t)$ can be computed analytically to $\mathcal{O}(\epsilon^3)$, while $e^u(t)$ and $e^s(t)$ can be computed to $\mathcal{O}(\epsilon^2)$; however, we will *not* use the time-periodicity property in our manifold calculations. We will turn to aperiodically time-dependent examples in [Section 5](#).

The periodically forced Duffing equation is

$$\dot{x}_1 = x_2, \tag{17}$$

$$\dot{x}_2 = x_1 - x_1^3 + \epsilon \sin t, \tag{18}$$

where $\epsilon = 0.01$. In Step 1 of our three-step algorithm in [Section 2.3](#), we set $t_n = T = 6\pi$, i.e., three times the forcing period 2π , and treat $dT = T/n$ as a parameter for the insertion technique. Using perturbation theory for small ϵ , we obtain:

$$x_{\text{DHT}}(t) = -\frac{\epsilon}{2} \begin{pmatrix} \sin t \\ \cos t \end{pmatrix} + \mathcal{O}(\epsilon^3) \tag{19}$$

and also

$$e^u(t) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \mathcal{O}(\epsilon^2), \quad e^s(t) = \frac{1}{\sqrt{2}} \begin{pmatrix} -1 \\ 1 \end{pmatrix} + \mathcal{O}(\epsilon^2). \tag{20}$$

Therefore, λ is the only control parameter in $x_W^u(t)$ that determines $W_{\text{approx}}^u((x_{\text{DHT}}(t), t))$ (16) in Step 2. To focus on the insertion technique in Step 3, we set $\lambda = 0.1$ throughout this paper. This value specifies the limits of a

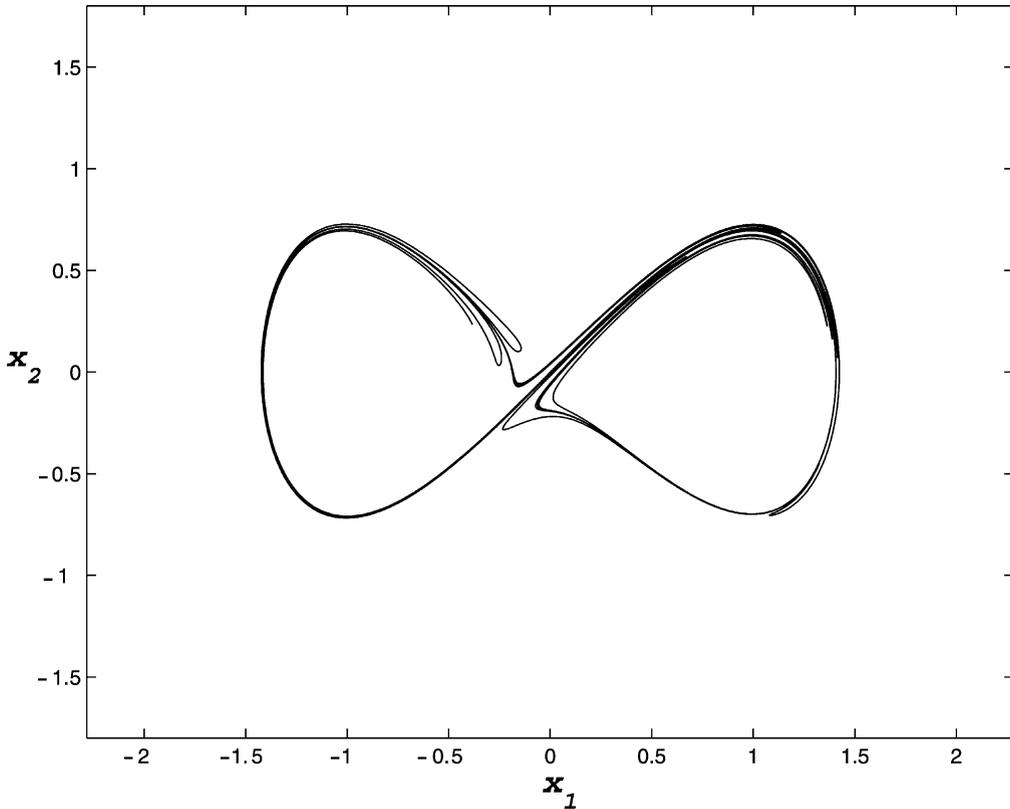


Fig. 5. Computation of the unstable manifold for the periodically forced Duffing equation for $\epsilon = 0.01$ using the benchmark method. Results are shown at $T = 6\pi$. The segment of the unstable manifold shown in the figure contains $NP_T = 3844$ points, where the maximum allowed distance between adjacent points is $\tau = 0.05$. The computing time used was CPU = 5904 s.

segment where the local manifold is well approximated by a line. Visual inspection of the manifold shown in Fig. 5 would seem to indicate this choice of $\lambda = 0.1$ as self consistent since the manifold has linear appearance near the hyperbolic trajectory. In order to quantify the initial deviation of the manifold from a line, we take advantage of the periodicity of the manifold (i.e. it is the same at time 6π and 0) and we measure the distance from the endpoints of $x_W^u(6\pi)$ for $\lambda = 0.1$ to the grown manifold. This distance is below 10^{-4} , which is within ϵ^2 , and therefore validates this choice of λ . The length of the manifold shown in this figure at $T = 6\pi$ is determined by the value of λ , since the endpoints of the initial approximation to the unstable manifold remain the endpoints of the manifold as it evolves in time. In general, an appropriate choice of λ will depend both on the eigenvalues of the linearized problem and on the nonlinearity of the system.

While $W_{\text{approx}}^u(x_{\text{DHT}}(t), t)$ can be obtained analytically, individual trajectories must be numerically integrated between the time slices. In this study, we use a fifth-order adaptive Runge–Kutta algorithm with a variable time step of initially $\Delta t = 0.05$, and an error tolerance controlled by the parameter $\epsilon_{rk} = 1 \times 10^{-8}$ (see [27] for the use of this).

We also enforce the constraint that the trajectories are computed by the Runge–Kutta method on the time slices t_k —whenever they cross a time slice, the integrator is used again with a new adjusted step that ends exactly on the time slice.

The unstable manifold calculated using the benchmark method is shown in Fig. 5 and is used as a reference to evaluate other methods. The figures shown in this section are the magnification of small regions near this

manifold. The codes used in computing all the manifolds in this article are written in Python and can be found at <http://lacms.maths.bris.ac.uk/projects/>. The codes have been run on a 1 GHz processor with 1 Gb RAM on a PC running Red Hat Linux.

3.2. Insertion criteria

An insertion technique has two components: an insertion criterion and an interpolation method. In evaluating the performance of different insertion criteria we used the most robust interpolation method, which is the Dritschel interpolation method discussed in Section 3.3.3. In this case, we feel confident that any shortcomings in the manifold computation can be attributed to the insertion criterion.

Below we discuss four different criteria for determining if new points should be inserted. The “distance criterion” (Section 3.2.1), the “Hobson criterion” (Section 3.2.2) and the “Hobson variant criterion” (Section 3.2.3) have been discussed by Hobson [10]. The “Dritschel–Ambaum criterion” (Section 3.2.4) derives from a technique originally developed to study the complex evolution of vorticity contours, i.e., contour surgery for contour dynamics in fluid mechanics. Our work analyzes for the first time the applicability of this method to the computation of invariant manifolds.

Note that throughout this section x_j denotes the j th point along the manifold, considered as an ordered sequence of points. In order to avoid confusion we note here that in two dimensions when it is necessary to consider the components of x_j we use the notation $(x_{1,j}, x_{2,j})$.

3.2.1. Distance criterion

The first criterion used is the simplest; on each Σ_{t_k} , the distance between a pair of adjacent points x_j and x_{j+1} corresponding to adjacent trajectories:

$$d_j \equiv |x_{j+1} - x_j|, \quad d_j \in \mathbb{R}$$

are checked against a threshold τ , which we call the “tolerance”. Its value is chosen by considering the maximum distance between points that would allow us to recover a smooth manifold such as the one displayed in Fig. 5. If $d_j > \tau$, then a new point is inserted between the two points on the *previous* time slice $\Sigma_{t_{k-1}}$ that evolve to x_j and x_{j+1} on the time slice Σ_{t_k} using one of the interpolation methods discussed in Section 3.3. The new point is then integrated to the time slice Σ_{t_k} and the checking process is repeated until all neighboring pairs of points satisfy the criterion. This iterative checking is also used with all the criteria described below.

Therefore, the distance criterion uses one threshold τ , and requires a computation of d_j for all x_j on each Σ_{t_k} .

Fig. 6 compares details of folds in the unstable manifold using this criterion for $\tau = 0.02$ and $dT = T/89$ with the benchmark.

3.2.2. Hobson criterion

In an attempt to capture details of the folds more accurately, we use an insertion criterion due to [10], which uses the angle α_j shown in Fig. 7; as explained in [10]: “This angle is a measure of the spacing of the image points relative to the local radius of curvature”, and is defined by

$$\alpha_j = 2 \sin^{-1} \frac{|\bar{x}_{j-1} - x_{j-1}|}{2|x_j - x_{j-1}|},$$

where

$$\bar{x}_{j-1} = x_j + \frac{x_j - x_{j+1}}{|x_j - x_{j+1}|} |x_j - x_{j-1}|.$$

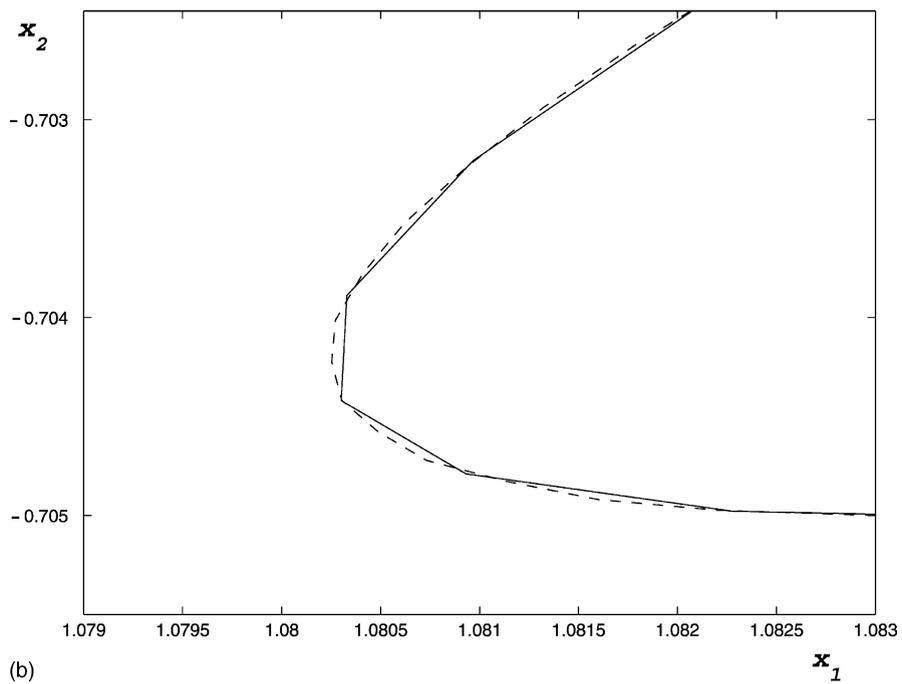
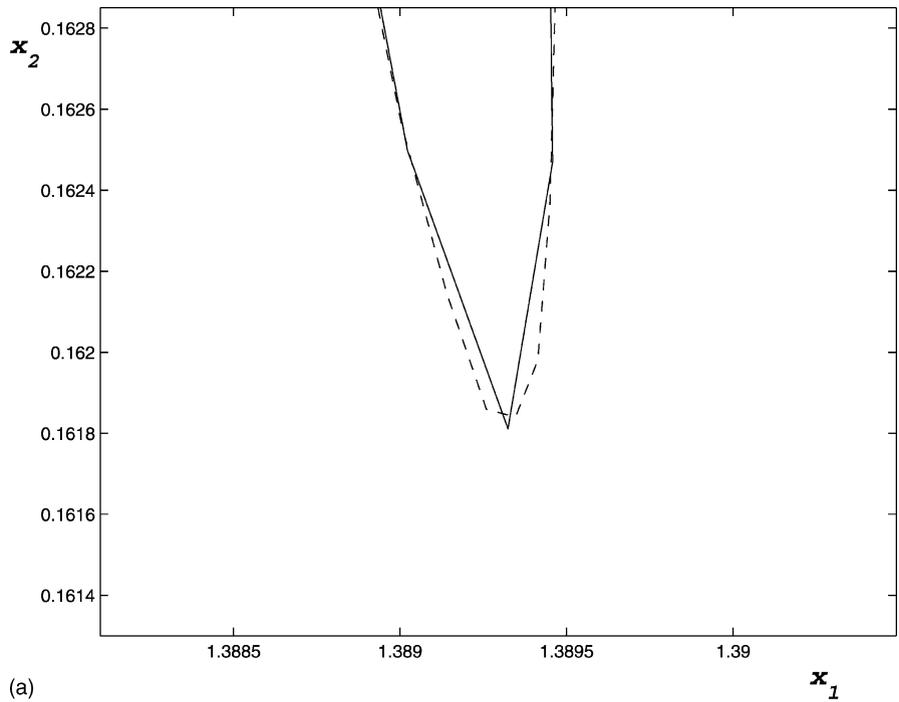


Fig. 6. The two panels show details of two different folds of the unstable manifold on Σ_T for the periodically forced Duffing equation with $\epsilon = 0.01$ using the distance criterion with $\tau = 0.02$, shown as the dashed curve. Point insertion was based on $dT = T/89$, and at T the number of points comprising the manifold is 9635. The computation time for this segment of the manifold was 1382 s. The benchmark with $\tau = 0.05$ is also shown as a solid black curve for comparison.

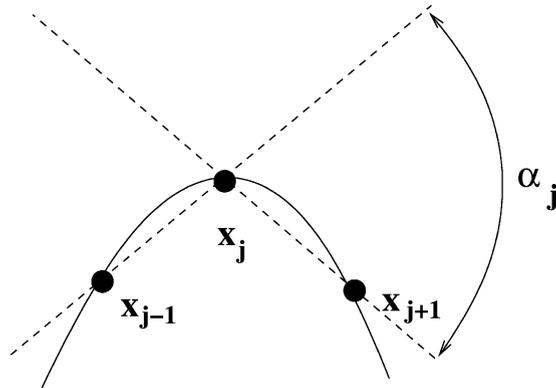


Fig. 7. The definition of the angle α_j used for the Hobson and Hobson variant criteria.

It follows from the definition that $0 \leq \alpha_j \leq \pi$. In [10], this technique is used to calculate the unstable manifold of a hyperbolic fixed point of the Poincaré map of the periodically forced Duffing system; in our framework this would correspond to sampling the manifold at time intervals $dT = T/3 = 2\pi$.

On each time slice Σ_{t_k} , we compute the angle α_j and the product $\alpha_j d_j$ for all points x_j . Then for each j we check them against the thresholds α and $d\alpha$, respectively. If $\alpha_j > \alpha$ or $\alpha_j d_j > d\alpha$, then a new point is inserted between the two points on the *previous* time slice $\Sigma_{t_{k-1}}$ that evolve to x_j and x_{j+1} on the time slice Σ_{t_k} using the best of the interpolation methods discussed in Section 3.3. Typical values for those parameters are discussed in [10]. We also require $d_j > \delta$, where δ is a minimum distance between adjacent points; otherwise when the distance between points becomes comparable to the errors in the numerical integration the α test can be triggered to insert points by these small errors and the method can fail. We found $\delta = 10^{-6}$ to be suitable for this problem.

In summary, the Hobson criterion uses three thresholds ($\alpha, \delta, d\alpha$) and requires computation of α_j, d_j , and $\alpha_j d_j$ for all x_j in Σ_{t_k} . For the Hobson criterion, as shown in Fig. 8, we find that the performance of this insertion technique depends on dT as well.

Fig. 8a shows a fold of the unstable manifold computed using $dT = 2\pi$, with $(\alpha, \delta, d\alpha) = (0.3, 10^{-6}, 0.001)$. As in [10], the folds of the manifold are computed with great resolution and accuracy. When the sampling time step dT is changed so that it does not match the period of the system the accuracy of the scheme diminishes, as Fig. 8b shows. Note that this phenomenon would not be observed using Poincaré map techniques; so far as we know it has not previously been noticed, and we are not currently able to offer an explanation for it.

Reducing $d\alpha$ will cause points on the manifold to be placed closer together [10], increasing accuracy but also the computational load. For instance in Fig. 9, the total number of points of the unstable manifolds jumps from 6280 to 19 173 by decreasing $d\alpha$ from 0.001 to 0.0001.

3.2.3. Hobson variant criterion

The wide separation of points that the Hobson criterion allows in “flat” regions of the manifold may hinder its response to the development of bends. A possible remedy is to combine the distance criterion, which prevents wide separation of points, with the Hobson criterion, which gives a better distribution of points in regions of large curvature. We call this combination the *Hobson variant criterion*. It uses the thresholds (α, δ, τ) and requires evaluation of α_j , and d_j for all x_j . To make the Hobson criterion symmetric with respect to the angle we have modified it slightly by redefining the angle as $\alpha'_j = (\alpha_j + \alpha_{j+1})/2$.

Fig. 10 shows the results for this method. The improvement from the Hobson criterion is quite remarkable, and the accuracy is as good as the more elaborate Dritschel–Ambaum criterion which we now discuss.

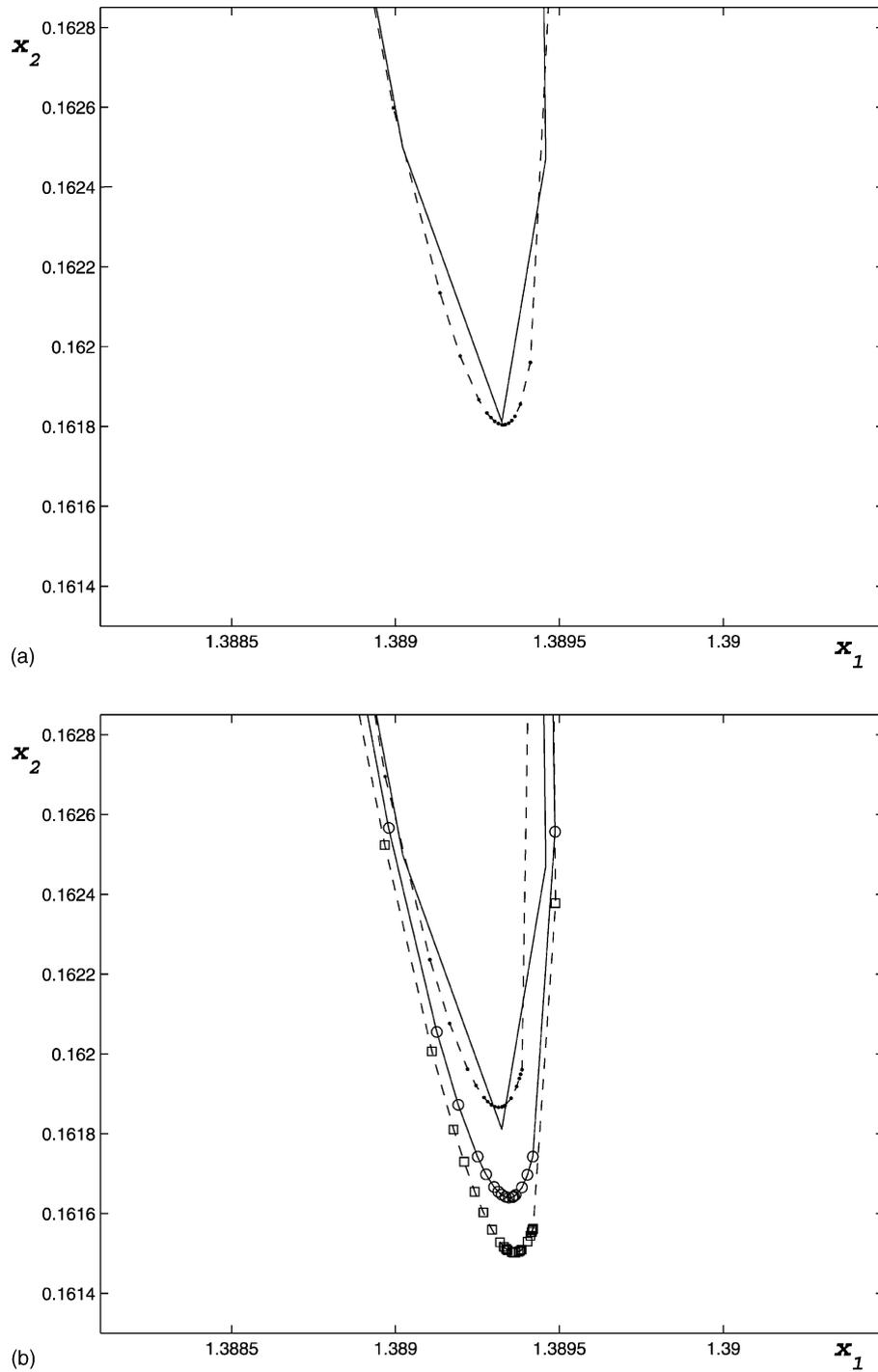


Fig. 8. The effect of dT using the Hobson criterion on Σ_T , in comparison to the benchmark (solid line) with $\tau = 0.05$ (we are showing the same fold as in Fig. 6a). All cases use $(\alpha, \delta, d\alpha) = (0.3, 10^{-6}, 0.001)$. (a) Dashed-dots line, $dT = T/3$ as in [10]. At this time the number of points comprising the manifold is $NP_T = 4002$. (b) Dashed-dots line with $dT = T/4$ ($NP_T = 4611$); solid-circle line with $dT = T/5$ ($NP_T = 4790$); dashed-square line with $dT = T/89$ ($NP_T = 6280$).

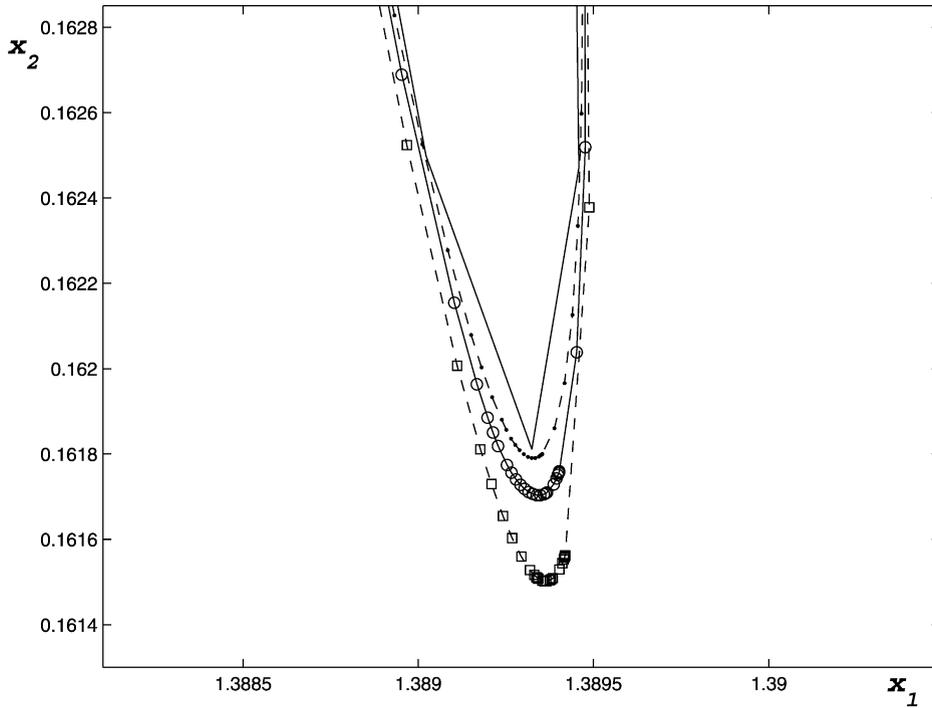


Fig. 9. Effect of $d\alpha$ while keeping $dT = T/89$: dashed-dot line with $d\alpha = 0.0001$ gives $NP_T = 19\,173$ and the computational time was $CPU = 2948$ s, solid-circle line with $d\alpha = 0.0005$ gives $NP_T = 8780$, dashed-square line with $d\alpha = 0.001$ gives $NP_T = 6280$ ($CPU = 985$ s).

3.2.4. Dritschel–Ambaum criterion

The Hobson criterion controls the point density by an estimate of relative point-spacing to curvature based on only three points—this works remarkably well much of the time, but can cause problems if the points move slightly away from their “true” positions: errors resulting from numerical integration and interpolation can cause point insertion after a false-positive diagnosis that the curve is not well resolved. The new points will be more closely spaced, and thus yet more vulnerable to misdiagnosis due to errors of a given size. It may be anticipated, then, that in some especially demanding problems a more elaborate scheme to control point density may become necessary (an example of this is shown in Fig. 20). Such a criterion has been developed by Dritschel and Ambaum [7] in the very different context of vortex dynamics in fluid mechanics. This method additionally incorporates a point redistribution procedure for use after the insertion phase which attempts to ensure that the points are concentrated in regions of the manifold, where greater definition is needed and can remove points from less demanding parts of the manifold and thereby reduce the total number of points included in the manifold. In Duffing systems this is not a major issue, but in problems where time-dependence strongly varies in phase space it can be an important matter to deal with.

For each pair of consecutive points x_j and x_{j-1} in the current manifold this method first calculates a desired density, ρ_j , and then calculates $\sigma_j = \rho_j d_j$, where $d_j \equiv x_j - x_{j-1}$. If $\sigma_j > 1$ a new point is inserted. The density ρ_j is set to be

$$\rho_j \equiv \frac{(\bar{\kappa}_j L)^{1/2}}{\mu L} + \bar{\kappa}_j, \quad (21)$$

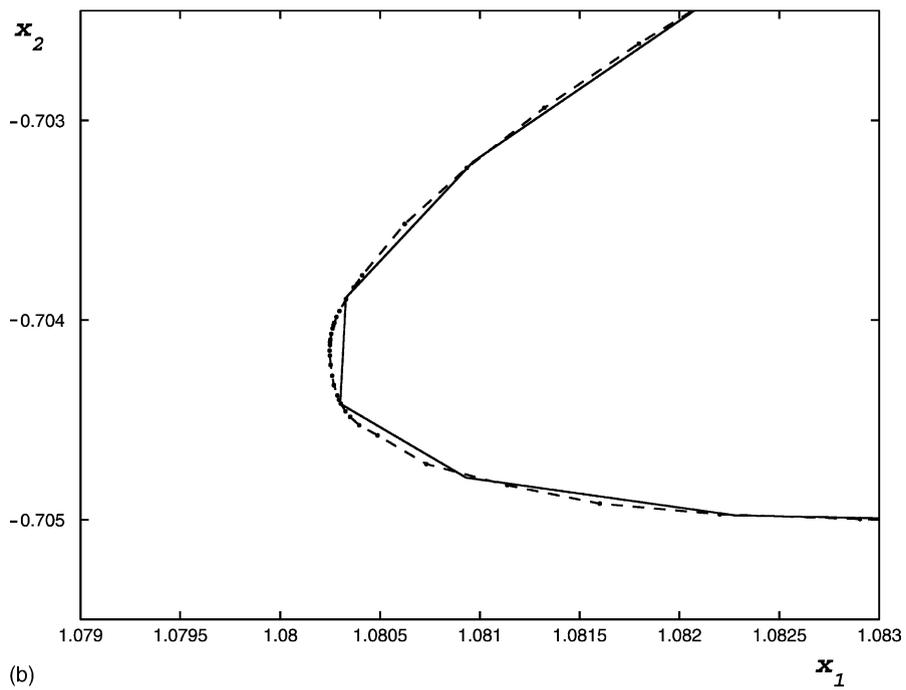
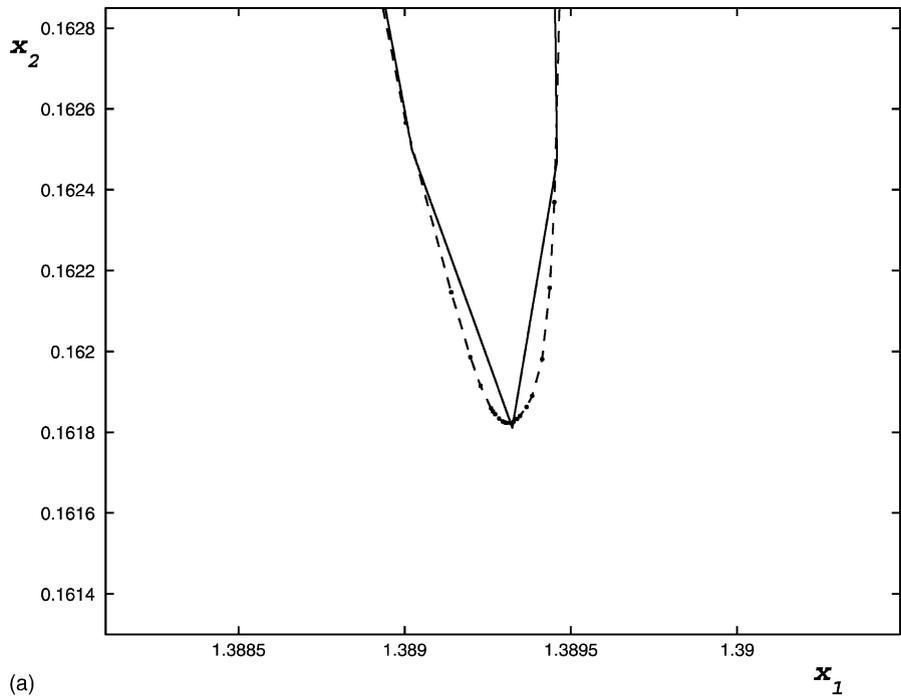


Fig. 10. A comparison of the unstable manifold computed with the Hobson variant criterion (dashed line) with $\tau = 0.02$, $\alpha = 0.3$, and $\delta = 10^{-6}$ with the benchmark (solid line). The number of points in the manifold computed with the Hobson variant criterion is $NP_T = 9845$ and the computational time was $CPU = 1544$ s: (a) the same fold as shown in Fig. 6a; (b) the same fold as shown in Fig. 6b.

or $2/\delta$, whichever is smaller (so that nodes cannot get closer than $\delta/2$; we set the parameter $\delta = 10^{-6}$). The parameter μ in this equation is the primary means by which the density is ultimately controlled. Small values of μ correspond to a high point density, but this parameter needs tuning for individual problems.

In Eq. (21):

$$\bar{\kappa}_j \equiv \frac{1}{2}(\check{\kappa}_j + \check{\kappa}_{j+1})$$

is defined in terms of $\check{\kappa}$, which in turn is defined as

$$\check{\kappa}_j = \frac{w_{j-1}\tilde{\kappa}_{j-1} + w_j\tilde{\kappa}_j}{w_{j-1} + w_j}$$

which uses the weighting $w_j = d_j/(d_j^2 + 4\delta^2)$ and the curvature $\tilde{\kappa}_j$, which itself is defined by

$$\tilde{\kappa}_j = \sqrt{\kappa_j^2 + \frac{1}{L^2}},$$

where κ_j , finally, is the local curvature defined by a circle through the three points x_{j-1} , x_j and x_{j+1} (an explicit formula is given in Eq. (26)), and L is defined in [7] to be a length typical of the large-scale vorticity distribution; we have substituted a typical length of the domain in which the manifold grows, i.e., $L = 3$.

Once the insertion criterion is satisfied along the whole length of the manifold at t_k the node redistribution process begins, following [6]. During redistribution the end points of the manifold are held fixed.

Let n be the number of nodes at t_k . Compute

$$q = \sum_{j=1}^n \sigma_j$$

and define $\tilde{n} = [q] + 2$ (i.e., two more than the nearest integer to q). The $n - 2$ “old” nodes between the end points will be replaced by $\tilde{n} - 1$ entirely new nodes in such a way that the spacing of new nodes is approximately consistent with the desired average density, controlled by the parameter μ .

To redistribute the points along a manifold defined by a finite collection of points requires us to make assumptions about the behavior of the curve between points. This is essentially the same problem as that of interpolation, which we address in Section 3.3 below. The Dritschel redistribution procedure shares a great deal with what we have called below (Section 3.3.3) Dritschel interpolation, and we shall refer the reader to that section for some of the details.

Let $\sigma'_j = \sigma_j \tilde{n} / q$ so that $\sum_{j=1}^n \sigma'_j = n$. Then, the positions of the new nodes $i = 2, \dots, \tilde{n}$ are found successively by seeking, for each i , a j and a p such that

$$\sum_{l=1}^{j-1} \sigma'_l + \sigma'_j p = i - 1.$$

This value of p is the parametric distance along the implicit curve between the old nodes j and $j + 1$ to be used to interpolate the position for the point, as described in Section 3.3.3 below.

Now, attempts to use only this redistribution scheme to accommodate the growth of the manifold were not successful, as Fig. 11 shows. The difficulties arise from the time difference $dT \equiv t_k - t_{k-1}$ at which manifolds are adjusted: we wish to take this much larger than the time step appropriate for integrating individual trajectories on the manifold. This means that large gaps between points on the manifold can develop, and the redistribution algorithm, which relies on only *small* changes between applications, is unable to fill them in a reasonable way.

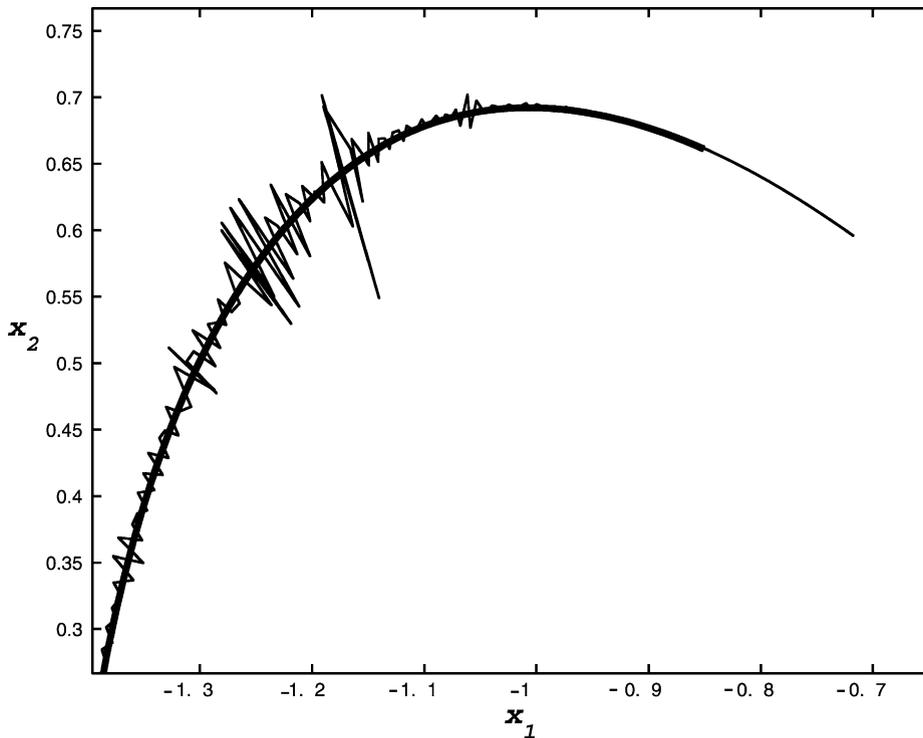


Fig. 11. A fold of the unstable manifold using the Dritschel–Ambaum criterion with redistribution only for $\mu = 0.005$ after 21 steps of size dT (thick line) and 22 steps of size $dT = T/89$ (thin line). The method has clearly failed between these two time steps.

Accordingly, we have modified the algorithm given in [7] by developing from it a criterion for point insertion. Since, as discussed in Section 3.2.1, we always insert points on the previous time step to satisfy a criterion on the current one ($\sigma_j < 1$), we prevent gaps developing which could give problems to the redistribution algorithm. Once the criterion is satisfied for each pair of points we may then take advantage of the redistribution algorithm to reduce the number of points used to represent the manifold and be confident that the resulting points will still be a good representation.

Fig. 12 shows the results for this method. The performance is as good as for the Hobson variant criterion although, at least in this example, the computational time increases considerably. Nonetheless, the sophisticated redistribution method could prove to be valuable in problems where the time-dependence of the problem varies extensively in space and there are important accumulations of trajectories in the manifold.

3.3. Methods for interpolation

The criteria discussed above examine *whether or not* points should be inserted on the time slice Σ_{t_k} . We now determine *what* the spatial coordinates of the inserted points on the previous time slice $\Sigma_{t_{k-1}}$ should be. This utilizes an interpolation method. If the insertion is found to be necessary between $x_j(t_k)$ and $x_{j+1}(t_k)$, then we apply an interpolation method of our choice to $x_m(t_{k-1})$'s on the previous time slice $\Sigma_{t_{k-1}}$. A new point $x'(t_{k-1})$ is first inserted between the two points $x_j(t_{k-1})$ and $x_{j+1}(t_{k-1})$ according to the interpolation method. Then, the forward evolution map gives the position of the new point $(x'(t_k), t_k) = f_{k-1}(x'(t_{k-1}), t_{k-1})$ on the current time slice Σ_{t_k} .

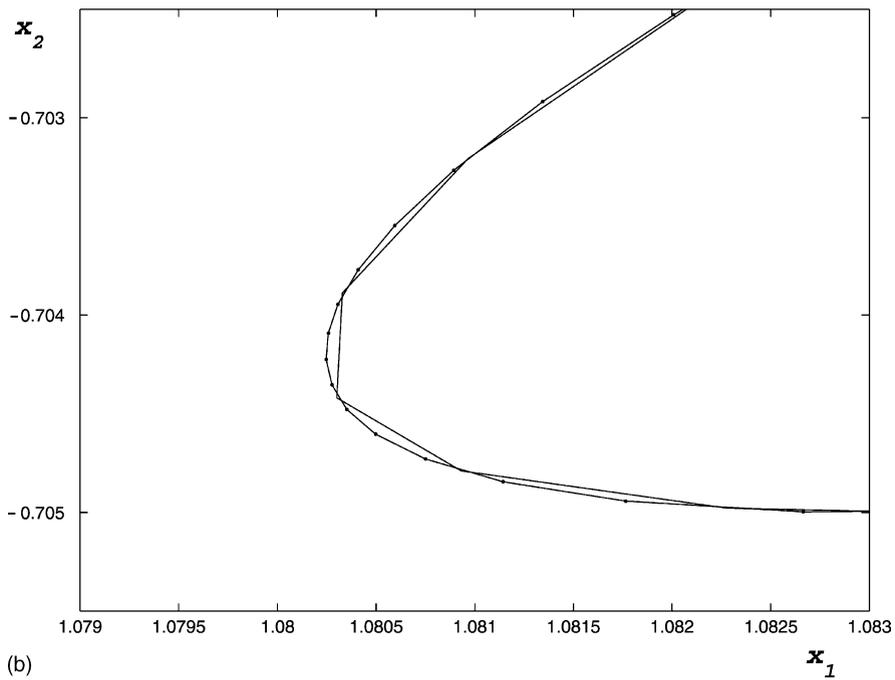
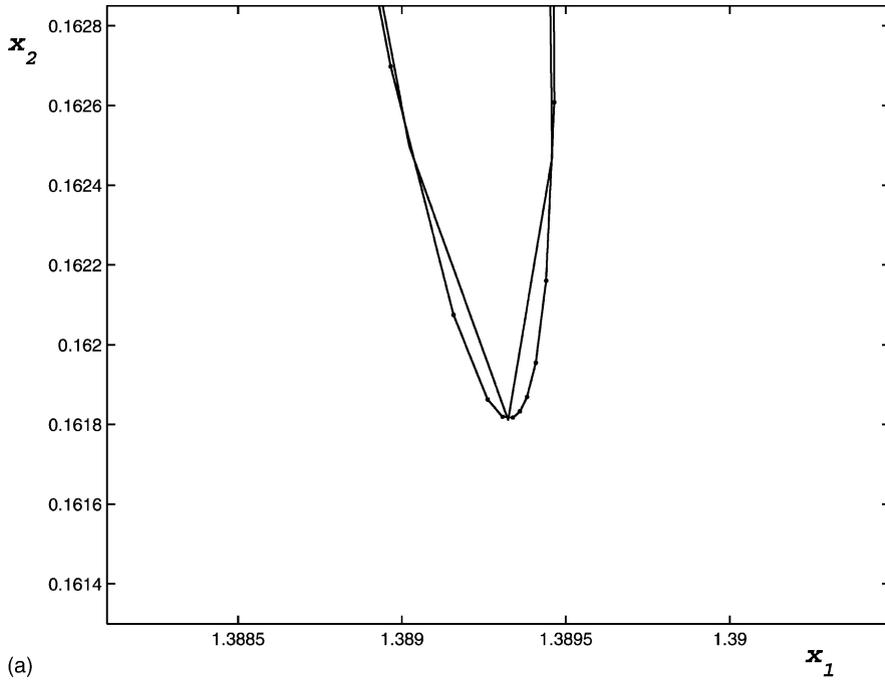


Fig. 12. The two panels show two folds of the unstable manifold using the Dritschel–Ambaum criterion on $\Sigma_{T=6\pi}$ for $\mu = 0.005$ (solid-dots line) and benchmark (solid line). The number of points in the manifold is $NP_T = 9879$ and the computational time is $CPU = 4358$ s.

3.3.1. Linear interpolation

The simplest method is linear interpolation. It uses two adjacent points x_j and x_{j+1} and adds a new point x' midway between, given by

$$x' = \frac{1}{2}(x_j + x_{j+1}).$$

This interpolation method has been used with the distance criterion described in Section 3.2.1.

The piecewise linearity assumed in linear interpolation is a poor approximation to the manifold, unless a sufficiently high point density is enforced, which is prohibitively expensive for aperiodically time-dependent vector fields. The subsequent evolution of points inserted via this method often leads to “jaggedness” in the computed manifold, as is shown in Fig. 13. Sophisticated methods for controlling point insertion mistake this spurious jaggedness for curvature of the manifold and attempt to resolve it. A feedback loop of insertion then develops as these methods attempt to resolve finer and finer scales, and the method breaks down (Fig. 14 shows this in the case of the Hobson variant point insertion criterion). This phenomenon also occurs with most other interpolation techniques and this means that the distance criterion for point insertion is the only one possible for these techniques since it does not attempt to resolve these high frequency oscillations of the computed “manifold”.

3.3.2. Lagrange polynomial and cubic spline interpolation

It might be hoped that higher order interpolation schemes would be less vulnerable to the high frequency oscillations. Accordingly we have also tried two third-order interpolation schemes. The third-order Lagrange polynomial

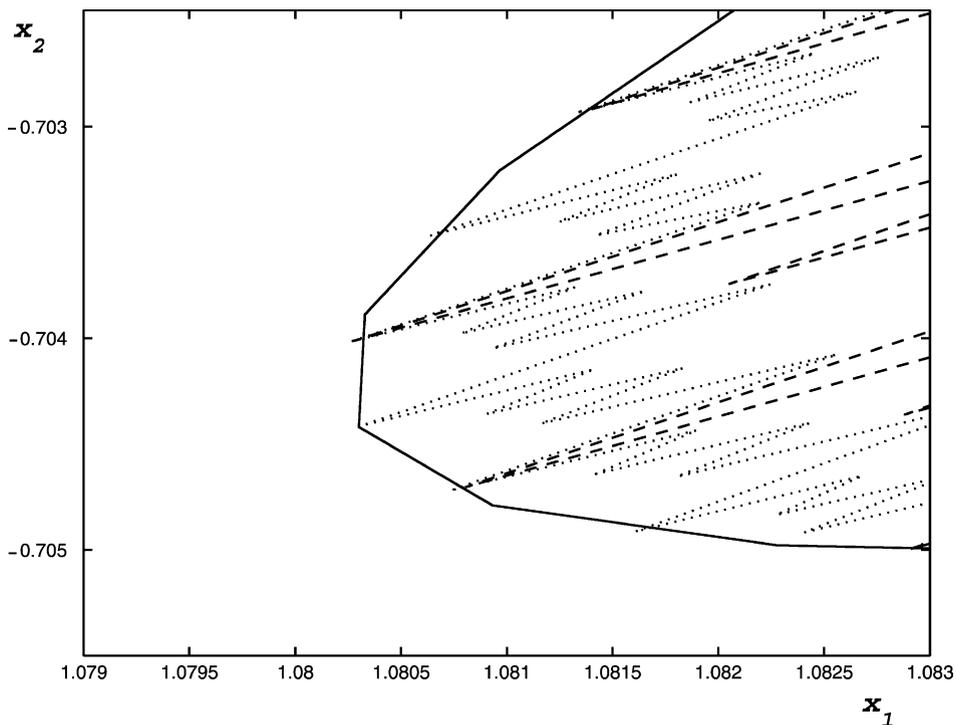


Fig. 13. A fold of the unstable manifold computed using linear interpolation and the distance criterion on Σ_T at $T = 6\pi$: the benchmark is the solid line, the dashed line is obtained by using the distance criterion ($\tau = 0.02$) for point insertion and linear interpolation with ($NP_T = 9739$, CPU = 1402 s). The dotted line uses $\tau = 0.01$ ($NP_T = 19484$, CPU = 2800 s).

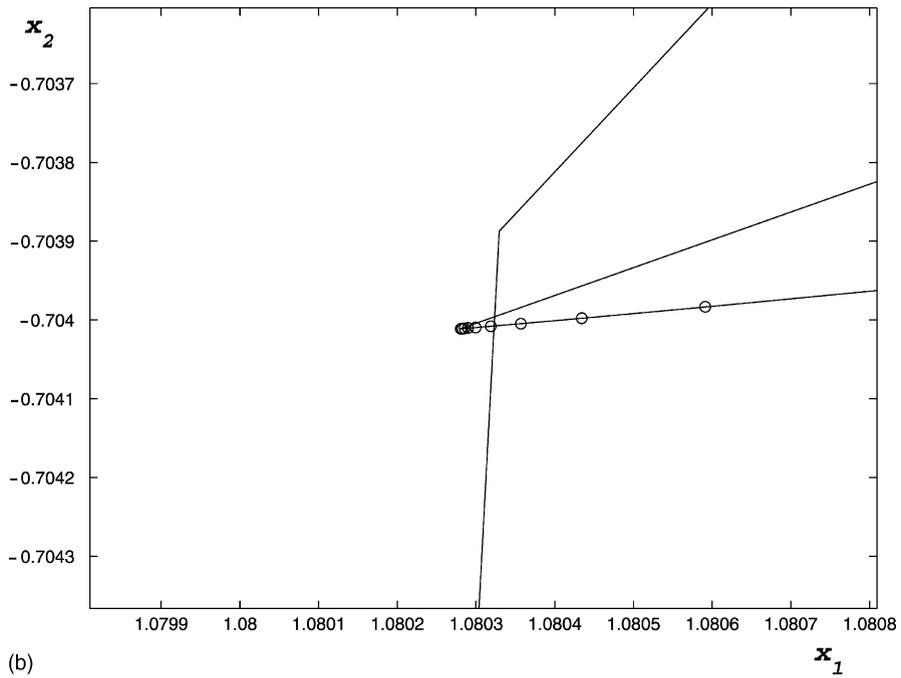
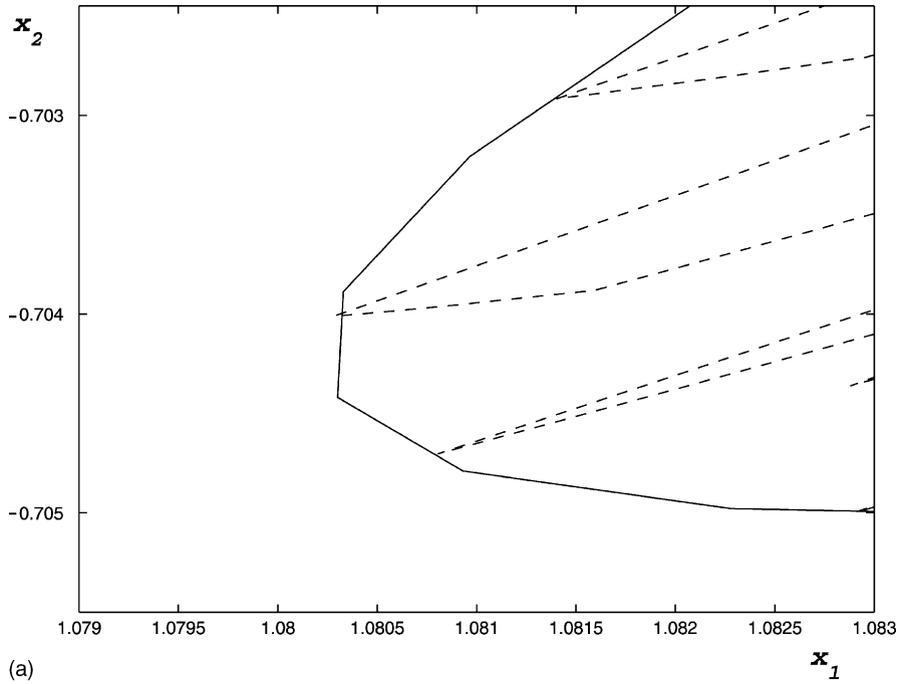


Fig. 14. The two panels show details of a fold of the unstable manifold computed using linear interpolation and the Hobson variant criterion on Σ_T at $T = 6\pi$: (a) benchmark is the solid line, the dashed line uses $\tau = 0.02$, $\alpha = 0.3$ and $\delta = 10^{-6}$ ($NP_T = 50\,815$, CPU = 9534 s); (b) in this magnification circles show points inserted in an attempt to resolve spurious artifacts.

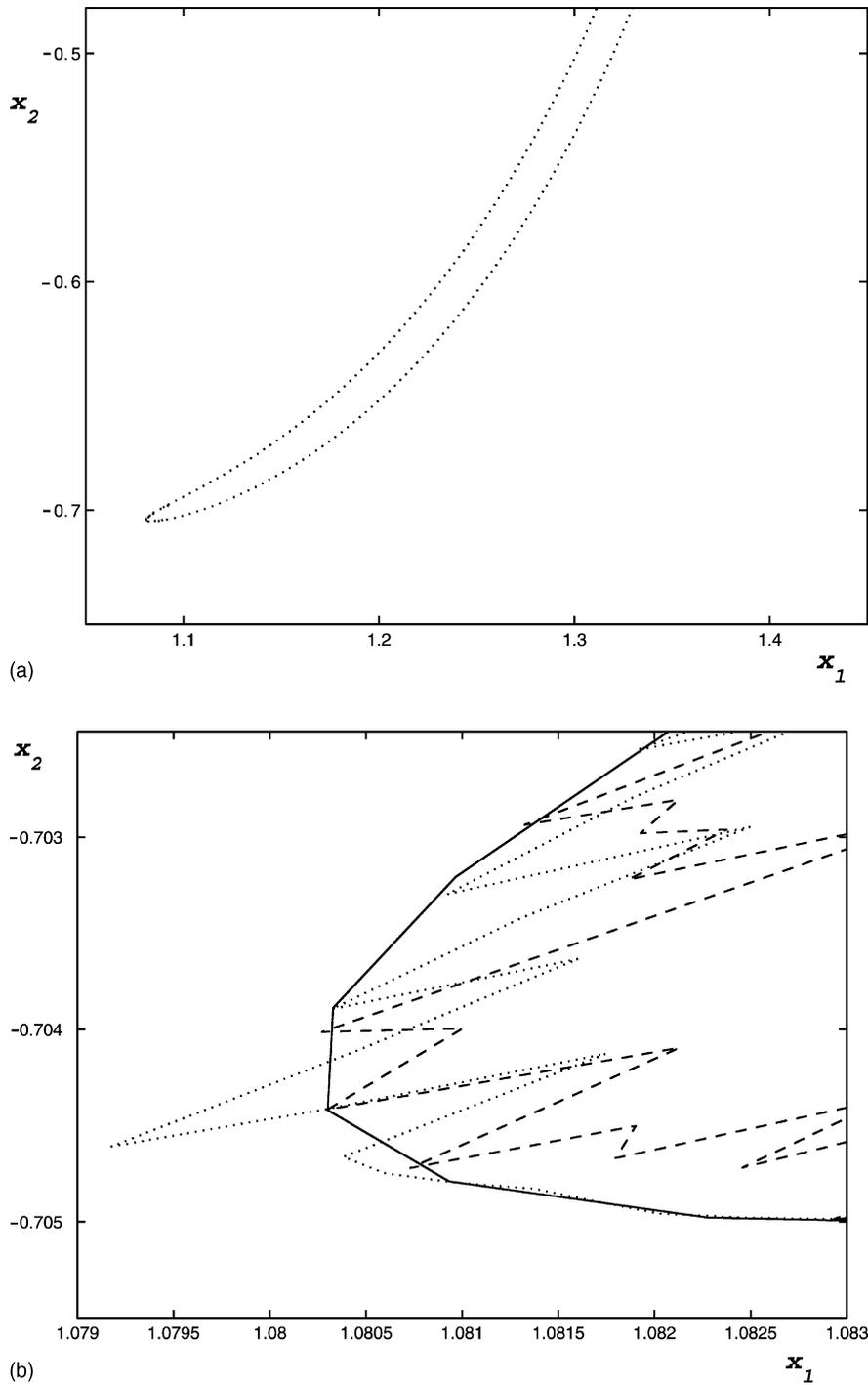


Fig. 15. A fold, and a zoom in showing details, of the unstable manifold using the distance criterion with $\tau = 0.02$ on Σ_T at $T = 6\pi$: (a) a fold using cubic splines; (b) magnification for the tip of the fold—the solid line is benchmark with $\tau = 0.05$, the dashed line is the Lagrange polynomials interpolation with $\tau = 0.02$ ($NP_T = 9683$, CPU = 1394 s), the dotted line uses cubic splines with $\tau = 0.02$ ($NP_T = 9636$, CPU = 1408 s).

interpolation [27] uses the four points x_{j-1} , x_j , x_{j+1} and x_{j+2} to describe a curve between x_j and x_{j+1} , denoted by $x_{\text{int}}(s)$ for $0 \leq s \leq 1$. In contrast, the cubic spline method uses all existing x_m 's to describe $x_{\text{int}}(s)$ between any two adjacent points (see [27] for details of both methods). The results are shown in Fig. 15, where it is seen that these methods do not eliminate the shortcomings of the linear interpolation method described above.

3.3.3. Dritschel's curvature method

The last interpolation method that we consider is Dritschel's curvature method [6]. It uses four consecutive points x_{j-1} , x_j , x_{j+1} and x_{j+2} like the third-order Lagrange polynomial interpolation method. This curvature method attempts to represent a segment of a smooth curve between x_j and x_{j+1} :

$$x_{\text{curv}}(p) = x_j + pt_j + \eta_j(p)n_j \quad (22)$$

for $0 \leq p \leq 1$ with $x_{\text{curv}}(0) = x_j$ and $x_{\text{curv}}(1) = x_{j+1}$, where

$$t_j = (a_j, b_j) = x_{j+1} - x_j, \quad t_j \in \mathbb{R}^2, \quad (23)$$

$$n_j = (-b_j, a_j), \quad n_j \in \mathbb{R}^2, \quad (24)$$

$$\eta_j(p) = \mu_j p + \beta_j p^2 + \gamma_j p^3, \quad \eta_j \in \mathbb{R}. \quad (25)$$

The cubic interpolation coefficients μ_j , β_j and γ_j are

$$\mu_j = -\frac{1}{3}d_j\kappa_j - \frac{1}{6}d_j\kappa_{j+1}, \quad \beta_j = \frac{1}{2}d_j\kappa_j, \quad \gamma_j = \frac{1}{6}d_j(\kappa_{j+1} - \kappa_j),$$

where $d_j = |x_{j+1} - x_j|$ and

$$\kappa_j = 2 \frac{a_{j-1}b_j - b_{j-1}a_j}{|d_{j-1}^2 t_j + d_j^2 t_{j-1}|} \quad (26)$$

is the local curvature defined by a circle through the three points, x_{j-1} , x_j , and x_{j+1} . Therefore, $x_{\text{curv}}(p)$ for $0 \leq p \leq 1$ uses the local curvature to describe the smooth curve between x_j , and x_{j+1} .

Fig. 6b shows a detail of the unstable manifold on Σ_T generated with Dritschel's curvature method and the distance criterion for point insertion. It completely eliminates the high frequency oscillations generated by the linear interpolation method (Fig. 13), the third-order Lagrange polynomial method, and the cubic spline method (Fig. 15); this justifies our choice of this method for evaluating the different point insertion criteria.

4. Discussion

In this section, we discuss in detail the performance of the different insertion techniques, which are summarized in Table 1. Our rule for tuning the parameters is that each technique should give a comparable average density of points along the manifold at the final time. There are three exceptions to this rule in Table 1. One is the benchmark calculation. The second is an additional case of the Hobson criterion with the Dritschel interpolation method, to improve accuracy. This technique works well with small enough values for $d\alpha$, but the average density of points in the manifold is huge, and, consequently, it requires significantly more computing time. The third exception is the Hobson variant insertion criterion with linear interpolation, where the large number of points illustrates the problems this technique has with high frequency oscillations.

The Hobson variant criterion with the Dritschel interpolation method, and with a reasonable average density of points and computing time, provides an excellent result. The choice of the Dritschel–Ambaum insertion criterion

Table 1
Comparing results of different methods

Method	Parameters	NP_T	CPU (s)	Comment
Benchmark	$\tau = 0.05$	3844	5904	Most accurate
Distance & Linear	$\tau = 0.02$	9739	1402	Oscillations
Distance & Linear	$\tau = 0.01$	19484	2800	Oscillations
Distance & Lagrange polynomial	$\tau = 0.02$	9683	1394	Oscillations
Distance & Cubic spline	$\tau = 0.02$	9636	1408	Oscillations
Distance & Dritschel	$\tau = 0.02$	9635	1382	Rough folds
Hobson & Dritschel	$\alpha = 0.3, d\alpha = 0.001, \delta = 10^{-6}$	6280	985	Inaccurate folds
Hobson & Dritschel	$\alpha = 0.3, d\alpha = 0.0005, \delta = 10^{-6}$	8780	1359	Inaccurate folds
Hobson & Dritschel	$\alpha = 0.3, d\alpha = 0.0001, \delta = 10^{-6}$	19173	2948	Accurate
Hobson variant & Dritschel	$\alpha = 0.3, \tau = 0.02, \delta = 10^{-6}$	9845	1544	accurate
Hobson variant & Linear	$\alpha = 0.3, \tau = 0.02, \delta = 10^{-6}$	50815	9534	Oscillations
Dritschel–Ambaum & Dritschel	$\mu = 0.005, \delta = 10^{-6}$	9879	4358	accurate

with the Dritschel interpolation method also produces accurate results, but with a heavier computational requirement in this case as compared with the Hobson variant criterion.

It might be imagined that we could exploit our knowledge of the manifold at t_k to develop a more sophisticated procedure than iteratively inserting one point at t_{k-1} between each adjacent pair of points which fail to satisfy the criterion when evolved to t_k : in particular, a multiple point insertion technique. But methods to accomplish this, which rely on local geometric approximations (even of third order) to the two-dimensional manifold in extended phase space, are likely to fail when dT is taken to be large, as we wish to do. It is quite clear that no such local approximation will characterize the dynamics of the system over long times.

In the criteria described above, one has to be careful in the choice of an adequate number of initial points along $W_{\text{approx}}^u((x_{\text{DHT}}(t_0), t_0))$. In criteria using distance as a control parameter, it is simplest to tentatively place only two points at the extremes of the initial segment spanning the DHT; if the number of initial points is insufficient for the criterion at time $t = dT$, then the number of points at $t = 0$ will be increased accordingly. Other criteria, such as Hobson or Dritschel–Ambaum, make use of curvature-related parameters. Since we think of our manifolds as being comprised of two “half-manifolds”, each of which is bounded at one end by the DHT, and we wish to be sure that the curvature of each half-manifold can be monitored from the beginning, we require at least three points on each half-manifold initially, of which the one nearest the DHT may be shared. This leads us to use five points at $t = 0$ for such methods; this has been shown to be enough for the criteria to act. Fig. 16 shows the evolution of the number of points in the unstable manifold against time. That figure shows two curves for the Dritschel–Ambaum criterion to illustrate the effect of point redistribution. Diamonds represent the number of points in the manifold at each time step before redistributing points and squares after redistributing points.

5. Aperiodically time-dependent examples

We have used the methods described in the previous section to calculate stable and unstable manifolds for Duffing-type systems with both periodic and aperiodic forcing.

Specifically, we use our variation of the Hobson criterion to determine when to insert points and Dritschel’s method to perform the interpolation, since these techniques offer the best compromise between speed and precision. Some results are also shown for the Dritschel–Ambaum point insertion criterion for comparison purposes.

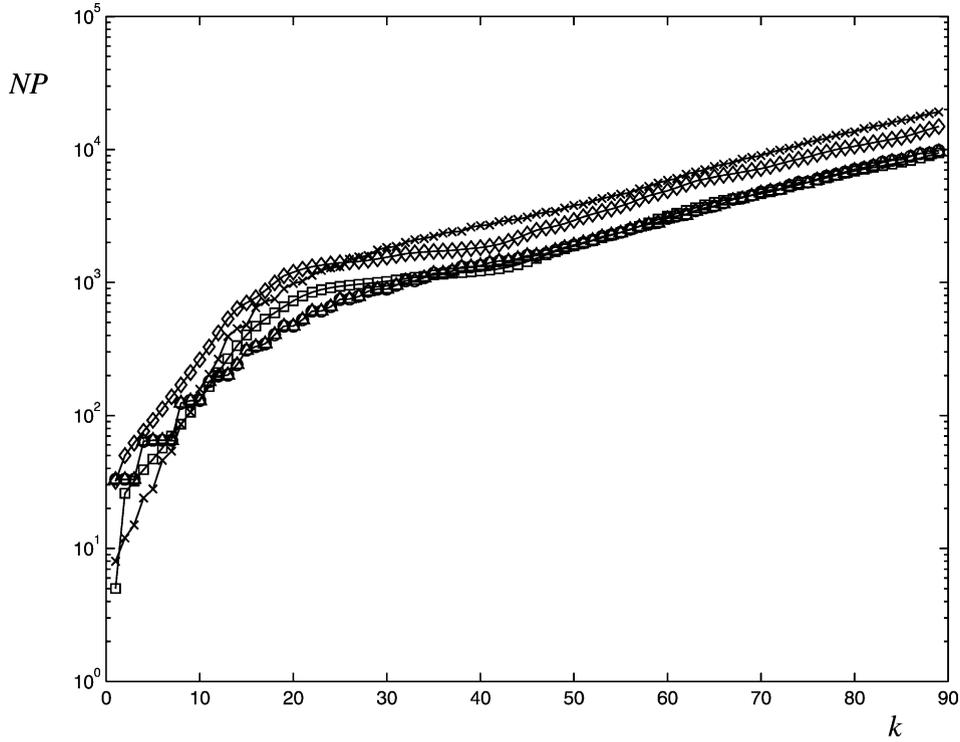


Fig. 16. The number of points in the unstable manifold versus the time index for different insertion methods. Crosses denote the distance criterion with $\tau = 0.02$; triangles denote the Hobson criterion ($\alpha = 0.3$, $d\alpha = 0.0001$); squares and diamonds are for the Dritschel–Aмбаum criterion with $\mu = 0.005$, at each time step squares represent the number of points in the manifold after the redistribution and diamonds before; circles are for the Hobson variant ($\alpha = 0.3$, $d = 0.02$).

5.1. Periodically forced rotating Duffing equation

To ensure the generality of our insertion technique for aperiodically time-dependent vector fields, we first use the rotating Duffing equation:

$$\begin{pmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \end{pmatrix} = \begin{pmatrix} \sin 2\beta t & \cos 2\beta t + \beta \\ \cos 2\beta t - \beta & -\sin 2\beta t \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} + (\epsilon \sin t - [\cos \beta t \xi_1 - \sin \beta t \xi_2]^3) \begin{pmatrix} \sin \beta t \\ \cos \beta t \end{pmatrix}. \quad (27)$$

This Duffing equation is quasi-periodic in time when the rotation rate β is irrational. It is obtained from (17) and (18) by applying the linear, rotation coordinate transform $x = R(t)\xi$ [11], where

$$R(t) = \begin{pmatrix} \cos \beta t & -\sin \beta t \\ \sin \beta t & \cos \beta t \end{pmatrix}. \quad (28)$$

As in the periodically forced Duffing equation, the DHT as well as the linearized stable and unstable subspaces can also be obtained through perturbation methods:

$$\xi_{\text{DHT}}(t) = R(t)^{-1} x_{\text{DHT}}(t), \quad e_{\xi}^u(t) = R(t)^{-1} e^u(t), \quad e_{\xi}^s(t) = R(t)^{-1} e^s(t),$$

where $x_{\text{DHT}}(t)$, $e^u(t)$ and $e^s(t)$ are given in (19) and (20). Therefore, the accuracy of $\xi_{\text{DHT}}(t)$ is $\mathcal{O}(\epsilon^3)$ and that of $e_{\xi}^u(t)$ and $e_{\xi}^s(t)$ is $\mathcal{O}(\epsilon^2)$.

We apply the three-step algorithm of Section 2.3 to the aperiodically forced Duffing equation with $\epsilon = 0.01$ and $\beta = \sqrt{2}$.

As before, to compute the unstable manifold, we choose $t_0 = 0$, $t_n = T = 6\pi$ and the time increment $dT = T/89$ for Step 1, and also $\lambda = 0.1$ for Step 2. We use the Hobson variant criterion—Dritschel curvature interpolation method with $(\alpha, \delta, \tau) = (0.3, 10^{-6}, 0.02)$. Fig. 17 shows the resulting unstable manifold on Σ_T , as well as the stable manifold starting on Σ_0 .

5.2. Exponentially forced Duffing equation

We now consider another variation of Duffing equation with aperiodic (exponential) forcing:

$$\dot{x}_1 = x_2, \quad (29)$$

$$\dot{x}_2 = x_1 - x_1^3 + \epsilon e^{-(t+2)} \cos(t+2) \quad (30)$$

with $\epsilon = 0.1$. As in the case of the periodically forced Duffing equation, the DHT can be approximated analytically:

$$x_{\text{DHT}}(t) = \frac{\epsilon e^{-(t+2)}}{5} \begin{pmatrix} -2 \sin(t+2) - \cos(t+2) \\ 3 \sin(t+2) - \cos(t+2) \end{pmatrix} + \mathcal{O}(\epsilon^3) \quad (31)$$

up to $\mathcal{O}(\epsilon^3)$. Given the order of the DHTs in (19) and (31), the linearized unstable and stable subspaces for both systems are the same as in (20) up to $\mathcal{O}(\epsilon^2)$.

To compute the manifolds, we once again take the three-step algorithm using the Hobson variant criterion—Dritschel curvature interpolation method with $(\alpha, \delta, \tau) = (0.3, 10^{-6}, 0.02)$ with $(dT, \lambda) = (T/89, 0.1)$. The results are shown for $t_n = 6\pi$ and $t_0 = 0$ in Fig. 18.

5.3. Duffing system with Gaussian forcing

We have also considered a variation of the Duffing equation with Gaussian-type forcing:

$$\dot{x}_1 = x_2, \quad (32)$$

$$\dot{x}_2 = x_1 - x_1^3 + \epsilon \left(\frac{11}{2}(t+3\pi)^2 - 1 - 2(t+3\pi)^4 \right) e^{-(t+3\pi)^2} \quad (33)$$

with $\epsilon = 0.1$. As in the other variations of Duffing's equation, the DHT can be approximated analytically:

$$x_{\text{DHT}}(t) = \epsilon \begin{pmatrix} -\frac{1}{2}(t+3\pi)^2 e^{-(t+3\pi)^2} \\ ((t+3\pi)^3 - (t+3\pi)) e^{-(t+3\pi)^2} \end{pmatrix} + \mathcal{O}(\epsilon^3)$$

up to $\mathcal{O}(\epsilon^3)$. Once again, $e^u(t)$ and $e^s(t)$ are the same as in (20) up to $\mathcal{O}(\epsilon^2)$.

For computation of the manifolds, we use the Hobson variant criterion—Dritschel curvature interpolation method with $(\alpha, \delta, \tau) = (0.3, 10^{-6}, 0.02)$. The results are shown for $t_n = 6\pi$ and $t_0 = 0$ in Fig. 19. Using the Dritschel–Ambaum criterion—Dritschel curvature interpolation method with $\mu = 0.005$ gives a similar performance. An identical figure is obtained where the number of points in the stable and unstable manifolds (after redistribution) is $\text{NP}_T = 10\,247$ and their computing time is $\text{CPU} = 2486$ s.

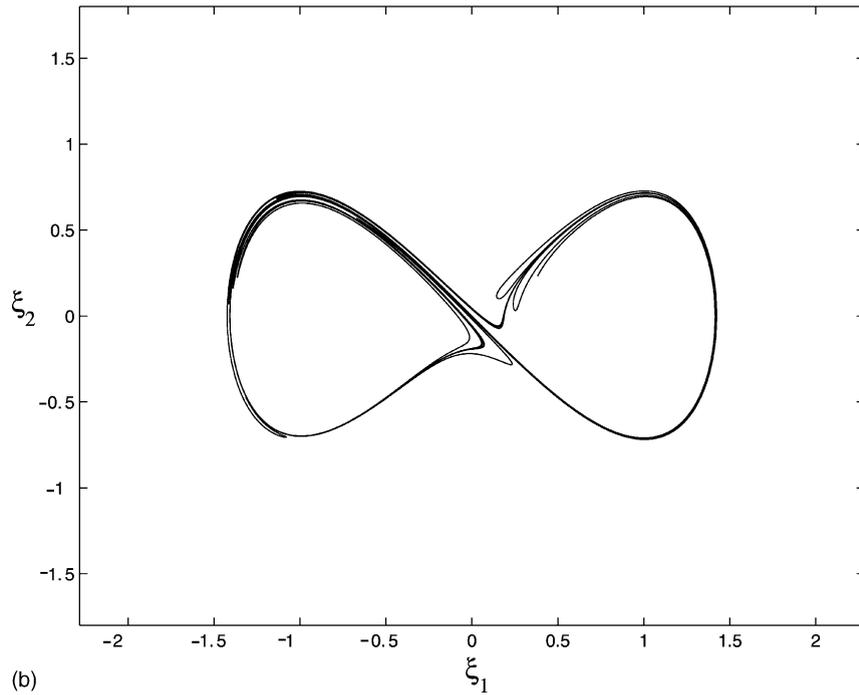
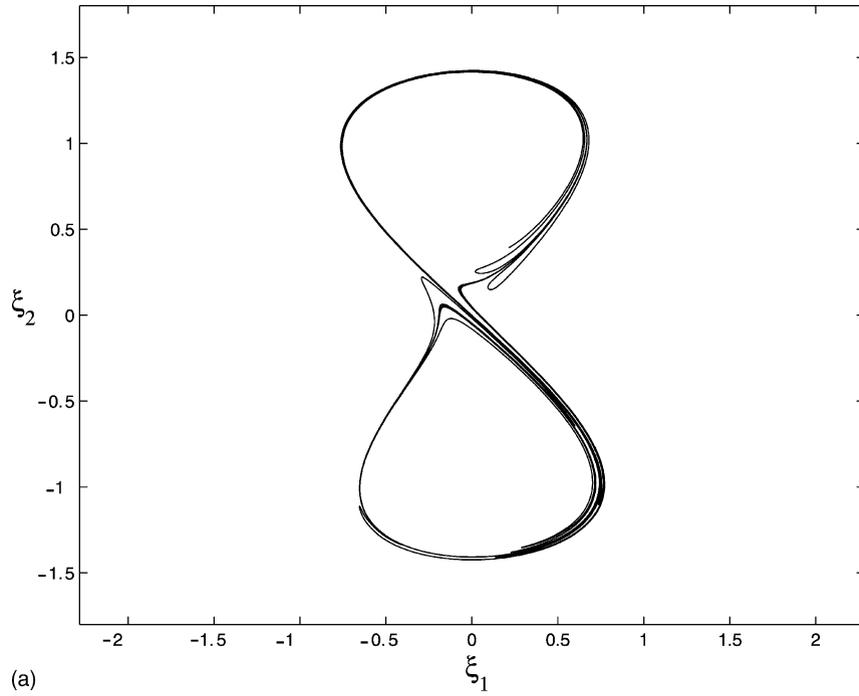


Fig. 17. The unstable manifold on Σ_T (a) and the stable manifold on Σ_0 (b) for the rotating Duffing equation with $\epsilon = 0.01$ and $\beta = \sqrt{2}$. The parameters used for the Hobson variant criterion—Dritschel curvature interpolation method are $(\alpha, \delta, \tau) = (0.3, 10^{-6}, 0.02)$ with $(dT, \lambda) = (T/89, 0.1)$. These figures have been computed in the same time as the nonrotating case (CPU = 1544 s) and contain the same number of points ($NP_T = 9845$).

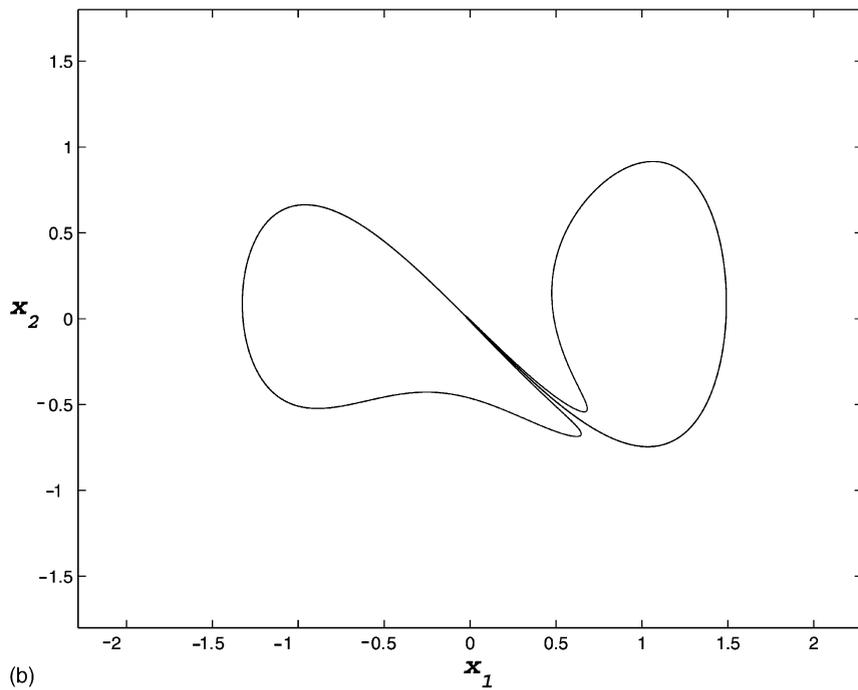
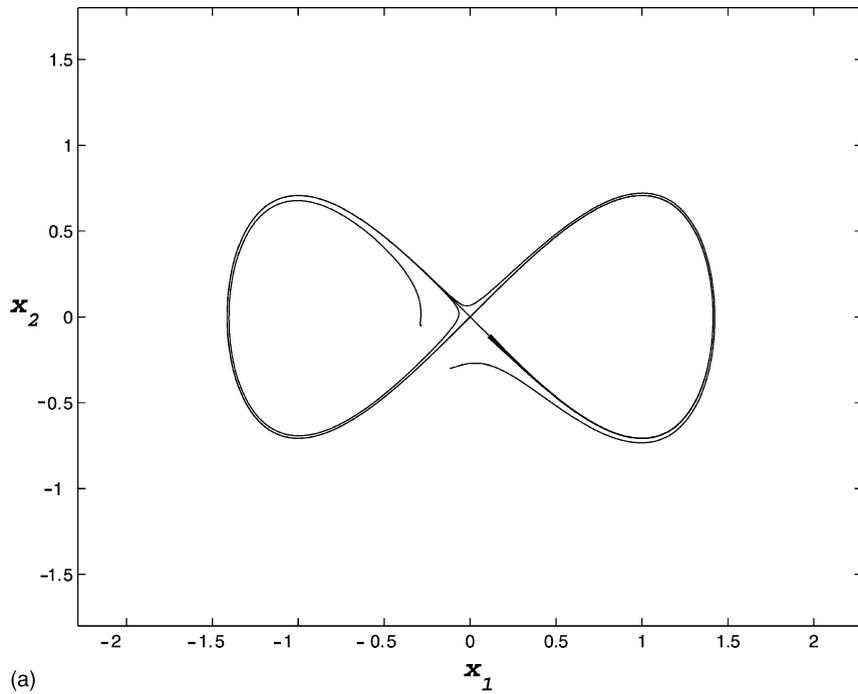


Fig. 18. The unstable manifold on Σ_T ($NP_T = 10\,696$, CPU = 1628 s) (a) and the stable manifold on Σ_0 ($NP_T = 21\,995$, CPU = 3705 s) (b) for the exponentially forced Duffing equation with $\epsilon = 0.1$. The parameters used for the Hobson variant criterion—Dritschel curvature interpolation method are $(\alpha, \delta, \tau) = (0.3, 10^{-6}, 0.02)$ with $(dT, \lambda) = (T/89, 0.1)$.

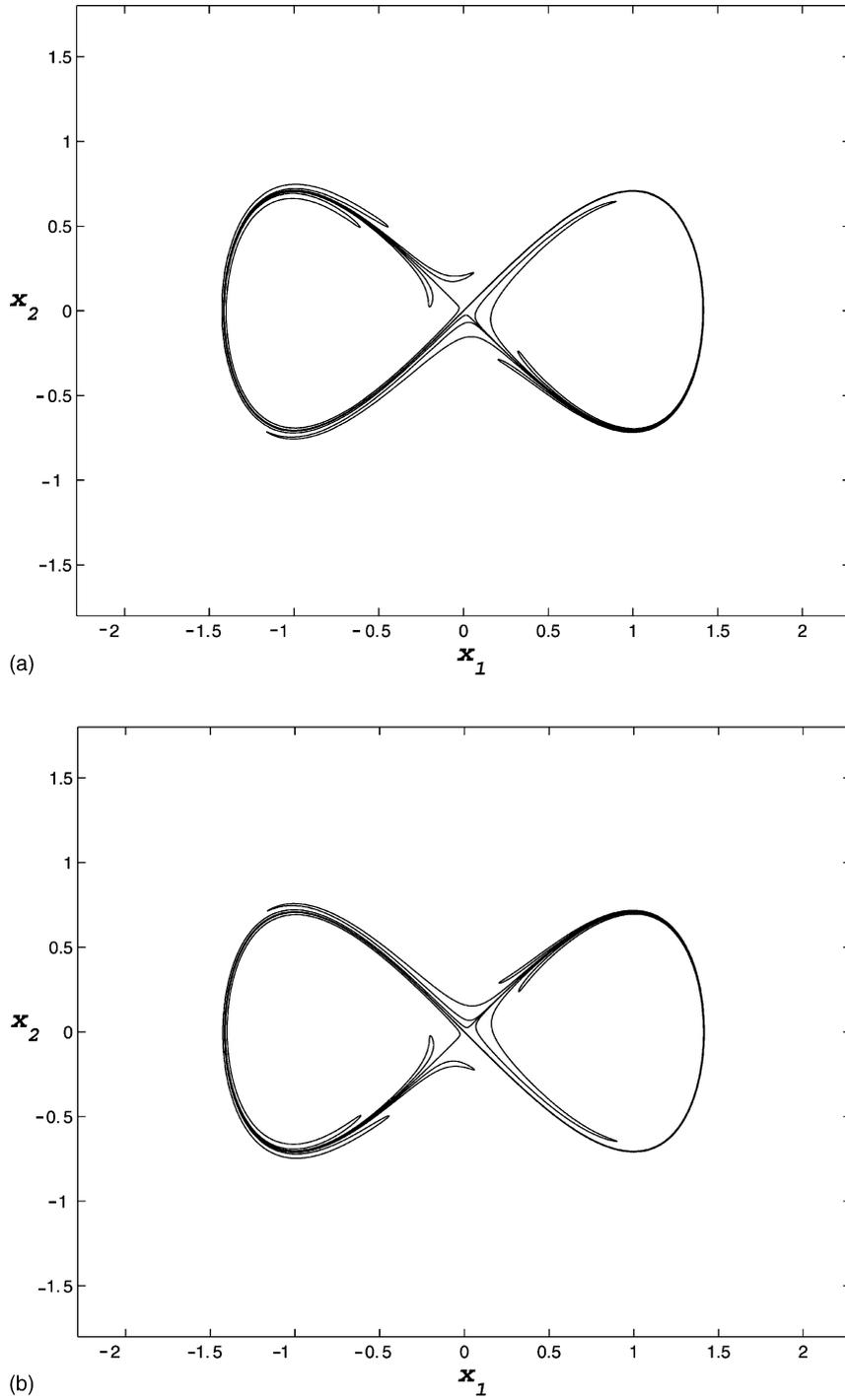


Fig. 19. The unstable manifold on Σ_T ($NP_T = 14\,629$, CPU = 2172 s) (a) and the stable manifold on Σ_0 ($NP_T = 14\,629$, CPU = 2170 s) (b) for the Gaussian-type forced Duffing equation at $\epsilon = 0.1$. The parameters used for the Hobson variant criterion—Dritschel curvature interpolation method are $(\alpha, \delta, \tau) = (0.3, 10^{-6}, 0.02)$ with $(dT, \lambda) = (T/89, 0.1)$.

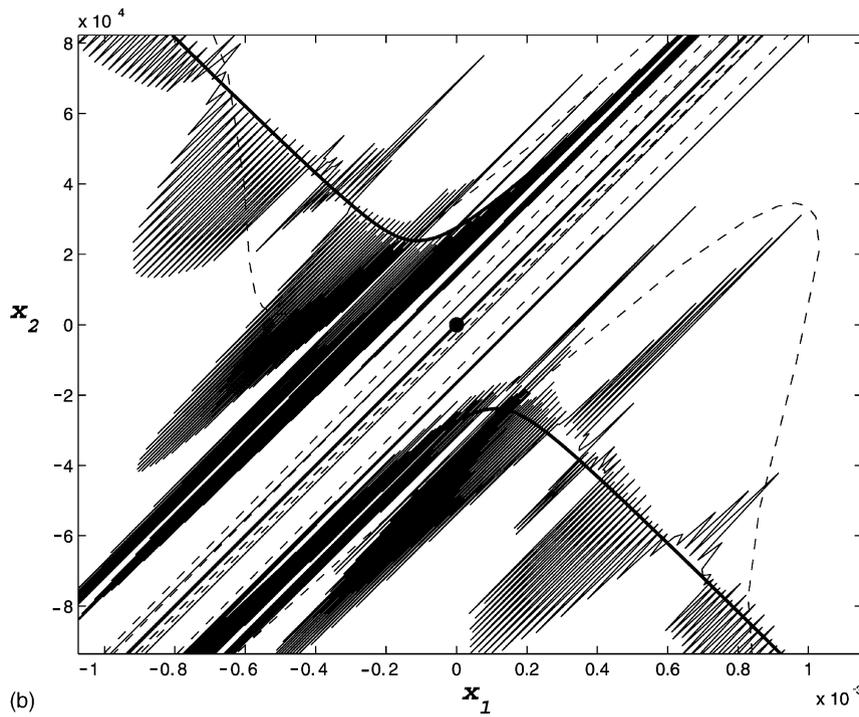
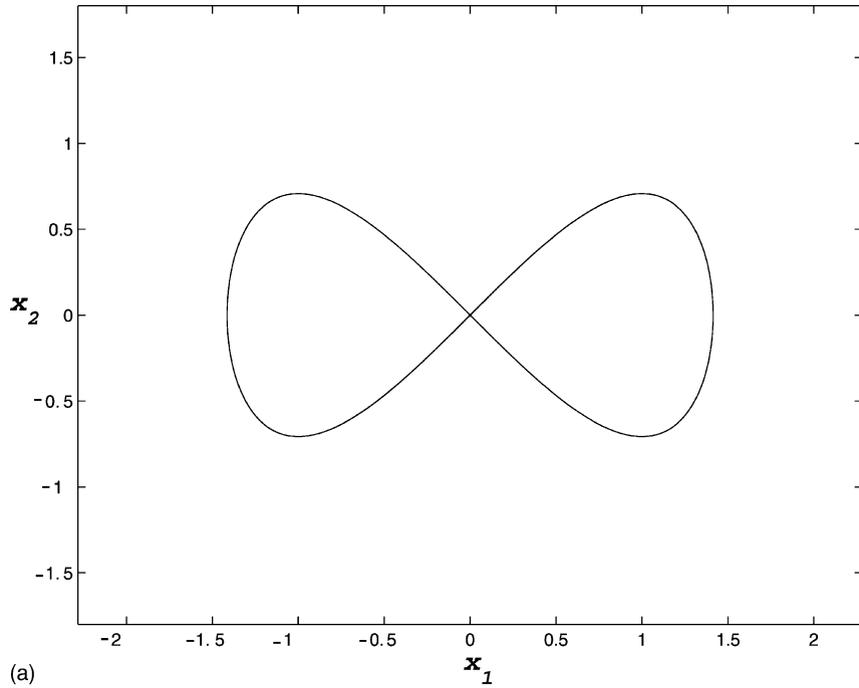


Fig. 20. (a) Unstable manifold of the DHT of the integrable Duffing equation. (b) Magnification near the hyperbolic point. The thin solid line is for the Hobson variant criterion $((\alpha, \delta, \tau) = (0.3, 10^{-6}, 0.02))$, while the dashed line is for the Dritschel–Ambaum criterion $((\mu, \delta) = (0.005, 10^{-6}))$. The thick line in this figure is the benchmark result.

5.4. Integrable Duffing equation

Finally, we apply the two methods to the integrable, time-independent Duffing equation. This is equivalent to setting $\epsilon = 0$ in the periodically forced cases (17) and (18) in Section 3.1, as well as the examples in Sections 5.2 and 5.3. The DHT is now a fixed point at the origin, while $e^u(t)$ and $e^s(t)$ are given exactly by (20).

We apply the methods as if this were an aperiodically forced velocity field like the other examples, using the Hobson variant criterion—Dritschel curvature interpolation method with $(\alpha, \delta, \tau) = (0.3, 10^{-6}, 0.02)$ and the Dritschel–Ambaum criterion—Dritschel curvature interpolation method with $\mu = 0.005$. The results are shown for $t_n = 15$ in Fig. 20. The global structure (Fig. 20a) suggest that the manifold has been recovered accurately. Fig. 20b, however, shows an enlargement of the region near the origin, and demonstrates that in approaching a hyperbolic point the influence of the unstable subspace will exaggerate even the slightest errors in the interpolation scheme. The thin solid line is for Hobson variant criterion while the dashed line is for the Dritschel–Ambaum criterion. It may be seen that in this case oscillations are smoothed. The thick line in this figure is the benchmark result.

Acknowledgements

We thank the referees for their careful reading of the manuscript and their helpful suggestions. In particular, we are grateful to one of the referees for bringing to our attention the joint work of Dritschel and Ambaum. The third author’s research was supported by ONR Grant No. N00014-01-1-0769. The fourth author’s research was supported by ONR Grant No. N00014-99-1-0020.

Appendix A. Comparison with time-periodic vector fields

Suppose the vector field (1) is periodic in time. Then it is well known that the trajectories can be studied by passing to the associated Poincaré map. Let \mathcal{T} denote the period of the vector field, and choose the sequence of times such that $dT_k = t_{k+1} - t_k = \mathcal{T}$ for all n . Then we have $f_i(\cdot) = f_j(\cdot) \equiv f(\cdot)$, for all i and j , and $f(\cdot)$ is the Poincaré map. Furthermore, we suppose the hyperbolic trajectory $\gamma(t)$ is a periodic trajectory with period \mathcal{T} (more generally, our discussion would apply if the period of $\gamma(t)$ were some rational multiple of \mathcal{T}). Then, for some arbitrary initial time (i.e., choice of Poincaré section), say t_0 at $i = 0$, $\gamma(t_0) \equiv \bar{\gamma}$ is a hyperbolic fixed point of the Poincaré map $f(\cdot)$, i.e., $f(\bar{\gamma}) = \bar{\gamma}$. Eqs. (11) and (12) then reduce to the familiar expressions:

$$W^u(\bar{\gamma}, t_k) = \lim_{n \rightarrow \infty} f^n(W_{\text{loc}}^u(\bar{\gamma}, t_0)), \quad (\text{A.1})$$

$$W^s(\bar{\gamma}, t_k) = \lim_{n \rightarrow \infty} f^{-n}(W_{\text{loc}}^s(\bar{\gamma}, t_0)) \quad (\text{A.2})$$

for all k . A comparison of these formulae with (11) and (12) reveal a fundamental difference between computing the stable and unstable manifolds of hyperbolic trajectories in aperiodically time-dependent vector fields versus time-periodic vector fields. In iterating $W_{\text{loc}}^u(\bar{\gamma}, t_0)$ under the Poincaré map *each iterate* is on the Poincaré section Σ_{t_0} . Hence, each iterate creates a segment of the unstable manifold at the desired time. Therefore, once a segment of the manifold is created, we no longer need to be concerned with that segment and the computation can then be focussed on creating additional length of the manifold. In contrast, a segment of the unstable manifold of the hyperbolic trajectory at t_0 is created by evolving in forward time a segment of the unstable manifold starting at some time in the past, t_{-n} , $n > 0$. During the course of the time evolution *none* of the manifold is created at time t_0 until the final time step. In other words, because of the lack of periodicity in the vector field as the manifold is grown in time *the entire length* of the manifold must be evolved in time.

We emphasize the differences further by making a direct comparison with a standard approach to computing the unstable manifold of a hyperbolic fixed point of a map. Consider a C^r ($r \geq 1$) diffeomorphism $g : \mathbb{R}^2 \rightarrow \mathbb{R}^2$ having a hyperbolic fixed point at $x = x_0$, and denote its local unstable manifold by $W_{\text{loc}}^u(x_0)$. Many algorithms for computing the unstable manifold use the idea of iterating a *fundamental domain*, which we now describe. Choose a point $\tilde{x} \in W_{\text{loc}}^u(x_0)$. Then the segment of $W^u(x_0)$ between \tilde{x} and $g(\tilde{x})$ is called a *fundamental domain*, and denoted by $D_{\tilde{x}}^0$. Then one of the branches of the local unstable manifold is “globalized” (or “grown”) by iterating this fundamental domain, i.e.:

$$W^u(x_0) = W_{\text{loc}}^u(x_0) \cup \left(\bigcup_{n=0}^{\infty} D_{\tilde{x}}^n \right),$$

where

$$D_{\tilde{x}}^n \equiv g^n(D_{\tilde{x}}^0)$$

(the other branch of the unstable manifold is grown by starting with a fundamental domain in that branch and following the same procedure). Of course, numerically, we can only grow the manifold for a finite number of iterates. So consider a finite piece of a branch of the unstable manifold grown in this way:

$$W_{\text{loc}}^u(x_0) \cup \left(\bigcup_{n=0}^{k_1} D_{\tilde{x}}^n \right), \quad \text{manifold after } k_1 \text{ iterates.} \quad (\text{A.3})$$

We can grow this branch longer by considering additional iterates of the fundamental domain:

$$W_{\text{loc}}^u(x_0) \cup \left(\bigcup_{n=0}^{k_1} D_{\tilde{x}}^n \cup \bigcup_{n=k_1+1}^{k_2} D_{\tilde{x}}^n \right), \quad \text{manifold after } k_2 \text{ iterates,} \quad (\text{A.4})$$

where $k_2 > k_1$.

Now in going from (A.3) and (A.4), we need no longer be concerned with evolving the manifold after k_1 iterates. We only need to append $k_2 - k_1$ iterates of $D_{\tilde{x}}^{k_1} \equiv g^{k_1}(D_{\tilde{x}}^0)$ to the manifold obtained after k_1 iterates in order to obtain the manifold after $k_2 > k_1$ iterates. In other words, to grow the manifold further in time we can “forget” about the manifold grown earlier in time since it is unchanged for all later times. This is a characteristic of the computation of unstable (and stable) manifolds of hyperbolic trajectories that is specific to maps and time-periodic vector fields which is *not* shared by aperiodically time-dependent vector fields. In the latter case, in evolving the manifold from an earlier time to a later time, the *entire* manifold from the earlier time must be evolved to the later time. This requires significantly more computational resources. In particular, many more points must be evolved to obtain comparable length manifolds in the aperiodically time-dependent case. As a result, we cannot “seed” the manifold initially with a sufficiently high density of points. This is why sophisticated point insertion criteria and interpolation methods are needed in the aperiodically time-dependent case (and why they are not needed when periodicity or *stationarity* can be exploited).

Appendix B. Benchmark

First we discuss a method that inserts points only at $t = t_0$. Such a method eliminates the errors associated with inserting points at other times and may therefore serve as a benchmark for point insertion strategies, although, as mentioned above, we expect it to be too numerically expensive for practical use.

Table 2

The benchmark algorithm

```

define Benchmark( $x_{\text{DHT}}, e^u, \tau$ )
  ( $\lambda, \Delta$ ) := ( $\lambda_0, \Delta_0$ )
  trajectories :=  $\emptyset$ 
   $x := x_{\text{DHT}} + \lambda e^u$ 
  traj := make_trajectory( $x$ )
  Outer loop begins here.
  while  $\lambda < \lambda_{\text{max}}$  do
    trajectories := trajectories  $\cup$  {traj}
    finished := false
    Inner loop begins here:
    while not finished do
       $\lambda' := \lambda + \Delta$ 
       $x := x_{\text{DHT}} + \lambda' e^u$ 
      trajnew := make_trajectory( $x$ )
      if max_dist(traj, trajnew) <  $\tau$  then
        finished := true
         $\lambda := \lambda'$ 
         $\Delta := 3\Delta$ 
        traj := trajnew
      else
         $\Delta := \Delta/2$ 
      end if
    end while
  end while
return trajectories

```

All points initially lie on the linear manifold W_{approx}^u given in Eq. (16), whose notation we adopt, at the initial time of the data set.

We assume the existence of subroutines “make_trajectory(x)”, which calculates and returns a numerical trajectory corresponding to a spatial position x at the start of the time interval and stores its values at all of the t_j of the chosen time discretization, and “max_dist(traj₁, traj₂)”, which returns the maximum distance between two given trajectories at any one of the t_j .

Using these assumptions, the subroutine “Benchmark($x_{\text{DHT}}, e^u, \tau$)” computes a set of trajectories, given the position, x_{DHT} , of the hyperbolic trajectory at t_0 , the unit vector e^u , and a tolerance τ that defines the maximum permissible distance between two neighboring trajectories at any t_j .

The algorithm, shown in Table 2, moves along W_{approx}^u by increasing λ in steps of Δ and storing the trajectories calculated in the set variable *trajectories*; this is the purpose of the outer loop.

Most of the subtlety of the code is concerned with picking a step Δ such that a new trajectory traj_{new} will remain close enough to its (already calculated and stored) neighbor traj. The inner loop of the subroutine does this by reducing Δ by half after unsuccessful trajectories and increasing it by a factor of three after successful trajectories. The Boolean variable *finished* indicates whether the inner loop has found a satisfactory new trajectory; when it has done so control returns to the outer loop, the trajectory is stored and the outer loop resumes.

The computation of the stable manifold is completely analogous.

References

- [1] H.W. Broer, H.M. Osinga, G. Vegter, Algorithms for computing normally hyperbolic invariant manifolds, Z. Angew. Math. Phys. 48 (1997) 480–524.

- [2] S.N. Chow, X.B. Lin, K.J. Palmer, A shadowing lemma with applications to semilinear parabolic equations, *SIAM J. Math. Anal.* 20 (1989) 547–557.
- [3] E.A. Coddington, N. Levinson, *Theory of Ordinary Differential Equations*. McGraw-Hill, New York, 1955.
- [4] W.A. Coppel, *Dichotomy in Stability Theory*, Lecture Notes in Mathematics, vol. 629, Springer, New York, 1978.
- [5] F.S. de Blasi, J. Schinas, On the stable manifold theorem for discrete time dependent processes in Banach spaces, *Bull. London Math. Soc.* 5 (1973) 275–282.
- [6] D.G. Dritschel, Contour dynamics and contour surgery: numerical algorithms for extended, high-resolution modelling of vortex dynamics in two-dimensional, inviscid, incompressible flows, *Comput. Phys. Rep.* 10 (1989) 77–146.
- [7] D.G. Dritschel, M.H.P. Ambaum, A contour-advective semi-Lagrangian numerical algorithm for simulating fine-scale conservative dynamical fields, *Quart. J. Roy. Meteor. Soc.* 123 (1997) 1097–1130.
- [8] J.K. Hale, *Ordinary Differential Equations*, Robert E. Krieger Publishing Co., Inc., Malabar, FL, 1980.
- [9] D. Henry, *Geometric Theory of Semilinear Parabolic Equations*, Lecture Notes in Mathematics, vol. 840, Springer, New York, 1981.
- [10] D. Hobson, An efficient method for computing invariant manifolds for planar maps, *J. Comput. Phys.* 104 (1993) 14–22.
- [11] K. Ide, D. Small, S. Wiggins, Distinguished hyperbolic trajectories in time dependent fluid flows: analytical and computational approach for velocity fields defined as data sets, *Nonlin. Process. Geophys.* 9 (3/4) (2002) 237–263.
- [12] M.C. Irwin, Hyperbolic time-dependent processes, *Bull. London Math. Soc.* 5 (1973) 209–217.
- [13] N. Ju, D. Small, S. Wiggins, Existence and computation of hyperbolic trajectories of aperiodically time dependent vector fields and their approximations, *Int. J. Bifurcat. Chaos*, to be published.
- [14] T.J. Kaper, On the structure of separatrix swept regions of slowly modulated Hamiltonian systems, On the quantification of mixing in chaotic Stokes flows: the eccentric journal bearing, Caltech Ph.D. Thesis, 1992.
- [15] A. Katok, B. Hasselblatt, *Introduction to the Modern Theory of Dynamical Systems*, Cambridge University Press, Cambridge, 1995.
- [16] P.E. Kloeden, B. Schmalfuss, Nonautonomous systems, cocycle attractors, and variable time-step discretization, *Numer. Algorithms* 14 (1997) 141–152.
- [17] E.J. Kostelich, J.A. Yorke, Z. You, Plotting stable manifolds: error estimates and noninvertible maps, *Physica D* 93 (1996) 210–222.
- [18] B. Krauskopf, H. Osinga, Growing 1D and quasi-2D unstable manifolds of maps, *J. Comput. Phys.* 146 (1998) 404–419.
- [19] J.A. Langa, J.C. Robinson, A. Suárez, Stability, instability, and bifurcation phenomena in non-autonomous differential equations, *Nonlinearity* 15 (2002) 887–903.
- [20] L.M. Lerman, L.P. Silnikov, Homoclinical structures in nonautonomous systems: nonautonomous chaos, *Chaos* 2 (1992) 447–454.
- [21] J.L. Massera, J.J. Schäffer, *Linear Differential Equations and Function Spaces*, Academic Press, New York, 1966.
- [22] K.R. Meyer, G.R. Sell, Melnikov transforms, Bernoulli bundles, and almost periodic perturbations, *Trans. Am. Math. Soc.* 314 (1989) 63–105.
- [23] K.R. Meyer, X. Zhang, Stability of skew dynamical systems, *J. Diff. Eqs.* 132 (1996) 66–86.
- [24] G. Moore, E. Hubert, Algorithms for constructing stable manifolds of stationary solutions, *IMA J. Num. Anal.* 19 (1999) 375–424.
- [25] H.E. Nusse, J.A. Yorke, *Dynamics: Numerical Explorations*, Springer, New York, 1997.
- [26] T.S. Parker, L.O. Chua, *Practical Numerical Algorithms for Chaotic Systems*, Springer, New York, 1989.
- [27] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, *Numerical Recipes in Fortran 77, The Art of Scientific Computing*, 2nd ed., Cambridge University Press, Cambridge, 1999.
- [28] J. Scheurle, Chaotic solutions of systems with almost periodic forcing, *Z. Angew. Math. Phys.* 37 (1986) 12–26.
- [29] G.R. Sell, *Topological Dynamics and Differential Equations*, Van Nostrand Reinhold, London, 1971.
- [30] S. Siegmund, Normal forms for nonautonomous differential equations, *J. Diff. Eq.* 178 (2002) 541–573.
- [31] C. Simó, On the analytical and numerical approximation of invariant manifolds, in: D. Benest, C. Froeschlé (Eds.), *Les Méthodes Modernes de la Mécanique Céleste*, Goutelas, 1989, p. 285.
- [32] D. Stoffer, Transversal homoclinic points and hyperbolic sets for non-autonomous maps. I, *J. Appl. Math. Phys. (ZAMP)* 39 (1988) 518–549.
- [33] D. Stoffer, Transversal homoclinic points and hyperbolic sets for non-autonomous maps. II, *J. Appl. Math. Phys. (ZAMP)* 39 (1988) 783–812.
- [34] S. Wiggins, Chaos in the dynamics generated by sequences of maps, with applications to chaotic advection in flows with aperiodic time dependence, *Z. Angew. Math. Phys.* 50 (1999) 585–616.
- [35] Y. Yi, A generalized integral manifold theorem, *J. Diff. Eq.* 102 (1993) 153–187;
Y. Yi, Stability of integral manifold and orbital attraction of quasi-periodic motion, *J. Diff. Eq.* 103 (1993) 278–322.