Computation of saddle-type slow manifolds using iterative methods

Kristiansen, Kristian Uldall

Published in:
S I A M Journal on Applied Dynamical Systems

Link to article, DOI:
10.1137/140961948

Publication date:
2015

Document Version
Publisher's PDF, also known as Version of record

SIAM J. Applied Dynamical Systems
Vol. 14, No. 2, pp. 1189–1227

Computation of Saddle-Type Slow Manifolds Using Iterative Methods*

K. Uldall Kristiansen†

Abstract. This paper presents an alternative approach for the computation of trajectory segments on slow manifolds of saddle type. This approach is based on iterative methods rather than collocation-type methods. Compared to collocation methods, which require mesh refinements to ensure uniform convergence with respect to ϵ, appropriate estimates are directly attainable using the method of this paper. The method is applied to several examples, including a model for a pair of neurons coupled by reciprocal inhibition with two slow and two fast variables, and the computation of homoclinic connections in the FitzHugh–Nagumo system.

Key words. slow-fast systems, slow manifolds of saddle type, reduction methods

AMS subject classifications. 34E15, 34E13, 37M99

DOI. 10.1137/140961948

1. Introduction.

Slow-fast systems of the form

$$\dot{x} = \epsilon X(x,y), \quad \dot{y} = Y(x,y) \tag{1.1}$$

or, equivalently,

$$x' = X(x,y), \quad y' = \epsilon^{-1} Y(x,y), \quad X, Y \in C^r,\ C^\infty, \text{ or } C^\omega, \tag{1.2}$$

with $x \in \mathbb{R}^{n_s}$ and $y \in \mathbb{R}^{n_f}$ being the slow and fast variables, respectively, arise in a wide variety of scientific problems. Here $\dot{()}$ denotes the derivative with respect to the fast time t, whereas $()'$ denotes differentiation with respect to the slow time $\tau = \epsilon t$. The vector-fields X and Y may in general also depend upon the constant ϵ that measures the time-scale separation. For simplicity, however, in this paper the ϵ-dependency will always be suppressed. Slow-fast systems appear in neuroscience [20, 59, 60, 58, 61, 69], chemical reaction dynamics [57], laser systems [9, 21, 23, 24, 25], meteorology and short-term weather forecasting [49, 48, 50, 62, 68], molecular physics and the Born–Oppenheimer approximation [53], the evolution and stability of the solar system [46, 47], modeling of water waves in the presence of surface tension [2], and the modeling of tethered satellites [65, 66]. The identification of slow and fast variables is extremely useful because of dimension reduction. Indeed, the two limit systems $(1.1)_{\epsilon=0}$ and $(1.2)_{\epsilon=0}$ enable, in many cases, a description of the system with $\epsilon > 0$ but sufficiently small. The actual identification of a time-scale separation parameter ϵ in a particular problem can, however, be a challenging task, even in planar problems; see, e.g., [10].

*Received by the editors March 24, 2014; accepted for publication (in revised form) by T. Kaper May 2, 2015; published electronically June 30, 2015. This research was supported by an H. C. Ørsted post doc grant.
http://www.siam.org/journals/siads/14-2/96194.html
†Department of Applied Mathematics and Computer Science, Technical University of Denmark, 2800 Kgs. Lyngby, Denmark (krkri@dtu.dk).

Although all of the problems mentioned above can be written in the form of (1.1) or (1.2), they are typically dynamically very different. Some are dissipative, and all the interesting dynamics takes place on a lower dimensional manifold [44, 54]. Others are conservative and oscillatory [1, 2, 49, 66]. In this case there is no complete theory (except for the case with only one slow and one fast degree of freedom [3, 30]) that relates the two limit systems $(1.1)_{\epsilon=0}$ and $(1.2)_{\epsilon=0}$ to $\epsilon > 0$ but small. Finally, there are cases where different lower dimensional objects interact through stable and unstable manifolds to form very nontrivial dynamics; see, e.g., [16, 33, 57]. In dynamical systems, numerical computations can often offer great insight. However, in slow-fast systems with both attracting and repelling lower dimensional manifolds, the time-scale separation makes the computation of such dynamics a challenging task [35].

Slow-fast theory. Consider a compact set of constrained equilibria $M_0 = \{(x, y)|Y(x, y) = 0\}$ with the spectrum $\text{spec}\,(\partial_y Y|_{M_0})$ satisfying

$$\text{dist}\,(\text{spec}\,(\partial_y Y|_{M_0}), i\mathbb{R}) \geq c > 0, \quad c \text{ independent of } \epsilon. \tag{1.3}$$

Here $\partial_y Y$ is the Jacobian of $Y(x, \cdot)$. Condition (1.3) implies, by the implicit function theorem, that M_0 is a graph of some function

$$y = \eta_0(x), \tag{1.4}$$

that is, $M_0 = \{(x, y)|y = \eta_0(x)\}$. For $\epsilon = 0$ this manifold M_0 is a fixed point set for (1.1), which is normally hyperbolic. It is referred to as the critical manifold. Fenichel's theory [26, 27] then applies to M_0 so that there exists an invariant manifold $M_h = \{y = \eta(x)\}$, with η smooth, which is $\mathcal{O}(\epsilon)$-close to M_0. The slow manifold M_h is attracting if $\text{spec}\,(\partial_y Y|_{M_0}) \subset \{z \in \mathbb{C}|\text{Re}z < 0\}$ or repelling if $\text{spec}\,(\partial_y Y|_{M_0}) \subset \{z \in \mathbb{C}|\text{Re}z > 0\}$. Otherwise it is of saddle type. In this case there are both a stable manifold $W^s(M_h)$, on which trajectories are attracted exponentially fast towards M_h forward in time, and an unstable manifold $W^u(M_h)$, on which trajectories are attracted exponentially fast towards M_h backwards in time [38]. Fenichel's theory also says that $W^s(M_h)$ and $W^u(M_h)$ are $\mathcal{O}(\epsilon)$-close to the stable and unstable manifolds of the fix point set $M_0|_{\epsilon=0}$ of $(1.1)_{\epsilon=0}$. The normally hyperbolic slow manifolds are like center manifolds [15] but, in contrast to center manifolds, slow manifolds are only local in the fast variables. Slow manifolds are "global" in the slow variables in the sense that Fenichel's description of these objects only fails locally where (1.3) is violated.

If, on the other hand, $\text{spec}\,(\partial_y Y|_{M_0})$ is not disjoint from the imaginary axis, but instead only satisfies

$$\text{dist}\,(\text{spec}\,(\partial_y Y|_{M_0}), 0) \geq c > 0, \tag{1.5}$$

c independent of ϵ, then the motion normal to M_0 is still fast but there is in general no invariant slow manifold nearby [52]. However, if the vector-field

$$U = \begin{pmatrix} \epsilon X \\ Y \end{pmatrix} \tag{1.6}$$

is analytic, then in this case there is some M_e on which the restriction of the vector-field has exponentially small angle $\mathcal{O}(e^{-c/\epsilon})$ with the tangent space [29, 67]. The slow manifold M_e is

therefore exponentially close to being invariant. This holds even in the normally elliptic case where spec $(\partial_y Y|_{M_0}) \subset i\mathbb{R}$, which is relevant for Hamiltonian systems. Only in the case of one fast degree of freedom does there exist a theory for the description of the fast dynamics of the slow manifold [29].

Numerical methods. There are traditionally two numerical approaches for the computation of slow manifolds. The first approach is to use collocation in the solution of an associated boundary value problem. The advantages of using a collocation based approach are many. One advantage is that nonlinear differential equations are effectively replaced with nonlinear algebraic ones, and the method therefore circumvents issues related to dynamic stability. This enables the computation of highly unstable orbit segments. The nonlinear algebraic equations can be solved by Newton's method provided a good initial guess is known. Collocation based approaches are also highly adaptable and can be directly integrated within the AUTO bifurcation analysis software [19] to perform bifurcation analysis. The second approach for the computation of slow manifolds is simply to use direct integration (also called the "sweeping method" [16]). Direct integration is easy to use. Also, whereas a collocation method requires an accurate initial guess to converge, direct integration can be used to explore the phase space. In fact, an initial guess for a collocation approach is often obtained using direct integration. Direct integration, however, has some documented disadvantages; see, e.g., [22]. In particular, this approach is limited to the computation of attracting slow manifolds (by forward integration) and repelling slow manifolds (by backwards integration). The computation of trajectories following saddle-type slow manifolds M_h for a long time, $t = \mathcal{O}(\epsilon^{-1})$ or $\tau = \mathcal{O}(1)$, cannot be achieved by any "stiff" integration method. Even an exact initial value solver in the presence of round-off errors of magnitude δ will amplify this error to unit size in a time of order $\mathcal{O}(\epsilon \log \delta^{-1})$ [35]. Such highly unstable orbit segments will be referred to as canards or, more accurately, canard segments.

There are many examples (e.g., the Van der Pol system [34], the model for reciprocal inhibition [35, 33], the FitzHugh–Nagumo system [36, 35, 37, 38, 42]) where important orbits have canard segments. Such orbits are referred to as canard orbits, and these were first analyzed in planar slow-fast planar systems by Benoît et al. [6, 7, 8, 18]. They found canard orbits as stable limit cycles that only existed in an exponentially small parameter regime. These appeared as the intersections of attracting and repelling slow manifolds. In $\mathbb{R}^3$ with two slow variables and only one fast, canard orbits appear persistently. Collocation based methods have in general proven very useful for the computational analysis of such canards; see, e.g., [17]. However, it is also possible to compute these orbits in $\mathbb{R}^3$ by a simpler approach using direct integration combined with shooting to a section by applying forward integration on the attracting slow manifold and backwards integration on the repelling one [32, 70]. For canard segments on saddle-type slow manifolds there exists to date, to the best of the author's knowledge, no alternative to collocation methods.

SMST algorithm. Guckenheimer and Kuehn in [35] developed an algorithm SMST (slow manifolds of saddle type) based on collocation for the computation of trajectories near a saddle-type slow manifold. The SMST method starts from an initial guess provided by the reduced system:

$$x' = X(x, \eta_0(x)), \tag{1.7}$$

with $x(0) = x_0$ and $\tau \in [0, T]$. Here $T = \mathcal{O}(1)$ with respect to ϵ. Set $z = (x, y)$ and let $x_T = x(T)$. The SMST algorithm then solves for a solution $z = z(\tau)$ that approaches the slow manifold near $z_0 \equiv (x_0, \eta_0(x_0))$ and exits it near $z_T \equiv (x_T, \eta_0(x_T))$. For this, time is discretized to $0 = \tau_0 < \tau_1 < \cdots < \tau_N = T$, and on each mesh $\tau_i \leq \tau \leq \tau_{i+1}$ the $z = z(\tau)$ is replaced by a cubic interpolation based on the values $z_i \equiv z(\tau_i)$, $z_{i+1} \equiv z(\tau_{i+1})$ and the tangent vectors $z'_i \equiv V(z_i)$, $z'_{i+1} \equiv V(z_{i+1})$. Here $V = \epsilon^{-1}U$ with U given in (1.6). The dynamical constraint $z' = V(z)$ is then enforced at the midpoints $\tau_{i+1/2} \equiv \frac{1}{2}(\tau_i + \tau_{i+1})$ using this cubic interpolation of $z = z(\tau)$. See also equation (2.1) in [35]. This gives $n \times N$ equations for the $n \times (N + 1)$ unknowns z_0, $z_1, \ldots, z_N$. The remaining n equations are obtained from the boundary conditions, which may be included in the following way. By assumption the matrix $\partial_y Y(z)$ introduces a splitting of the form $E^z_s \oplus E^z_u = \mathbb{R}^{n_f}$, where $E^z_s = E^z_s(x)$ and $E^z_u = E^z_u(x)$ can be interpreted as the stable and unstable eigenspaces of the *constrained hyperbolic equilibria* $y = \eta(x)$ of $\dot{y} = Y(x, y)$, x here being constrained as a parameter. Fenichel's theory guarantees that E^z_s and E^z_u are transverse to $W^u(M_h)$ and $W^s(M_h)$, respectively. Let

$$\begin{aligned} \pi^z_s &: \quad \text{the projection onto } E^z_s, \\ \pi^z_u &: \quad \text{the projection onto } E^z_u. \end{aligned} \tag{1.8}$$

Then at $\tau = 0$ one specifies $x(0) = x_0$ and the "stable components" of $y(0) = \eta_0(x_0) + y_{s0} + y_{u0}$ by fixing the value of

$$y_{s0} = \pi^{z_0}_s(y(0) - \eta_0(x_0)). \tag{1.9}$$

In [35] the value is fixed to 0. At $\tau = T$, on the other hand, one specifies the "unstable components" of $y(T) = \eta_0(x_T) + y_{sT} + y_{uT}$ by fixing the value of

$$y_{uT} = \pi^{z_T}_u(y(T) - \eta_0(x_T)). \tag{1.10}$$

The value is set to 0 in [35]. Here $\eta_0(x_T)$ is the value of the fast variables, when using (1.7) for the propagation of the slow variables, at $\tau = T$. From the cubic interpolation the sparse Jacobian can be computed explicitly, and a Newton method can be used to obtain an accurate solution. Note that Fenichel's theory implies that both y_{u0} and y_{sT} are $\mathcal{O}(\epsilon)$ since the stable and unstable fibers are $\mathcal{O}(\epsilon)$ close to the unperturbed ones. The time T can be included as a separate variable upon inclusion of a further boundary condition.

As the SMST method is formulated in [35], it cannot be used to approach trajectories on the slow manifold directly. Trajectories will always include transitions at the ends. In [45, section 4.2] a related collocation based method is used to compute trajectories on a one-dimensional (1D) attracting slow manifold using a continuation mechanism to *push out* the fast part at the ends. It may be possible to extend this approach to saddle-type slow manifolds.

For the computation of a full orbit, the SMST algorithm will in general have to be combined with a separate part that computes the remaining trajectory segments (e.g., via direct integration of (1.2)).

Assume that the SMST method converges to a solution $\sigma = \sigma(t)$ and that $z = z(t)$ is a

true solution of $z' = V(z)$ that satisfies the $n(N+1)$ conditions. Then by Taylor's formula,

$$\begin{aligned}\|\sigma - z\| &\leq \frac{1}{24} \max_{\tau\in[0,T]} \|z^{(4)}(\tau)\| \max_i |\tau_{i+1} - \tau_i|^4 \\ &\leq \mathcal{O}\Big(\epsilon^{-4} \max_i |\tau_{i+1} - \tau_i|^4\Big).\end{aligned}$$

The factor ϵ^{-4} appears from estimating $\|z^{(4)}\|$. This is too pessimistic on the slow manifold since there, $z^{(4)} = \mathcal{O}(1)$ (by definition of being slow), but it is appropriate if the connections at the ends are fast. If a mesh and boundary conditions are fixed, then, based on this estimate, one will expect the error to grow as ϵ goes to zero. For example, the reference [41] describes the use of collocation to solve the boundary value problem

$$\epsilon u''(\tau) + u'(\tau) = 1, \quad u(0) = 1 = u(1), \tag{1.11}$$

and it is shown that in order to ensure convergence estimates that are uniform with respect to ϵ for this problem, a fixed mesh must be replaced by an adaptive Shishkin mesh [41]. A Shiskin mesh is basically a piecewise uniform mesh that places more points at ends where the fast transitions occur. It is the main aim of this article to establish an alternative to collocation for the computation of saddle-type slow manifolds that accurately resolves both the slow motion along the slow manifold and the fast transitions by splitting the computation into two subproblems. The splitting will be obtained by the application of two iterative reduction methods: SO and SOF.

Reduction methods. The SO method (the method of *straightening out*, also referred to as the iterative method of Fraser and Roussel; see [39]) is an example of a reduction method that enables the computation of slow manifolds without direct reference to a small parameter as, e.g., it is required when using asymptotic expansions. There are several alternative methods: the intrinsic low-dimensional manifold (ILDM) method of Maas and Pope [51], the zero-derivative principle (ZDP) [28, 71], and the computational singular perturbation (CSP) method initially due to Lam and Goussis [43, 44] and later thoroughly analyzed by Zagaris, Kaper, and Kaper [72]. The SO method has the following interesting and numerically advantageous features:

(i) It leads to exponential accurate slow manifolds.
(ii) It can written in a form (see (2.2) below) that only involves the vector-field and its Jacobian matrix.
(iii) It does not require smoothness of X and Y in ϵ.
(iv) It leads to slow manifold approximations that include equilibria.

For the purpose of this work (ii) is an important property. It means that the approach is easy to implement. In comparison with the other methods, where the number of partial derivatives required depends on the desired accuracy, the SO method only requires the vector-field and its first partial derivatives. Property (iii) might seem rather academic, but it highlights the methods potential in ϵ-free systems (see [64, Section 8] and [11, 10, 12, 40]): The proof of statement (i) is not based on comparisons with asymptotic expansions in ϵ. In the paper [12] the authors apply the SO method in ϵ-free systems.

The SOF (*straightening out fibers*) method [64] is also an iterative method, built as an extension to the SO method, that enables approximation of fibers in slow-fast systems.

However, only the SOF method enjoys all the properties listed above. In [64, section 8] it was furthermore demonstrated that the SOF method performed far better on a problem where the slow and fast variables had not been properly identified.

This paper aims to demonstrate that the iterative methods, SO and SOF, can be used to compute saddle-type slow manifolds where direct integration does not provide a viable simple alternative to collocation methods.

Aims of this paper. The idea behind the presented method is simply to split the computations into two nonstiff subproblems: a computation on the slow manifold and a computation for the connection to and from the slow manifold. This approach is well known. In fact, it is at the very foundation of the theory of singular perturbation theory, geometric [26, 27, 38] or nongeometric [4, Chapter 10], [63], and the theory's aim to connect $\epsilon \neq 0$ with $\epsilon = 0$ of (1.1) and (1.2). The novelty here, however, is to obtain the splitting using the SO and SOF methods. In particular, the SO method will be used in a quadrature scheme for the propagation on the slow manifold. This procedure also applies to attracting or repelling slow manifolds (where direct integration of the full system probably offers a better approach) and even normally elliptic ones. Although, normally elliptic slow manifolds have not received as much attention as their hyperbolic counterparts, they do appear in a wide range of problems in science [1, 2, 49, 66].

In [56] an alternative numerical scheme is suggested for the propagation on the slow manifold. This is based on asymptotic expansions which require several partial derivatives of the vector-field U with respect to the slow and fast variables but also with respect to the small parameter. The SO method only requires U and the Jacobian $\partial_z U$ (see (ii) above).

The SOF method enables, through an accurate projection onto the slow manifold, the computation of connections to and from a trajectory on the slow manifold. This computation will involve collocation, but it is performed on the fast space only, using $\mathcal{O}(1)$-many time intervals of the fast time t, and will therefore not involve any ϵ^{-1}-factors (as opposed to collocation on the full space). The full method, which will be called SO-SMST (*straightening out* for *slow manifolds* of *saddle types*), will be described in full detail in section 3. It is among our main aims to demonstrate the use of the SO-SMST method and describe its performance. This will include an analysis of discretized SO and SOF methods used in the implementation. The SO-SMST method will be applied to several examples, and comparisons will be made with the SMST method. A thorough comparison with the SMST method is, however, not among the aims of this paper. This must be a topic for future research. Nevertheless, some potential advantages of the iterative method will be highlighted. For one thing, it will be stressed that the method presented here does not have any issues with $\epsilon \to 0$. This is, for example, documented by the inclusion of a linear test problem (1.11) in section 5.2, where the SO-SMST method captures the limit $\epsilon \to 0$ accurately. Hence, for certain specially structured systems, there could be potentially interesting applications for the SO-SMST method. On the other hand, it should be pointed out that a certain disadvantage of the iterative approach taken here is that for *larger* values of ϵ, SO and SOF may take longer time before reaching a specified tolerance. Worse yet, this tolerance may not be reached at all since the iterative methods may require an ϵ that is smaller than what is required by Fenichel's theory. In these cases, it is very likely that the SMST method will perform far better.

Outline of paper. In section 2 the two different iterative methods are presented. This includes a modification of the SO method which is due to Neishtadt [55]. Section 3 presents the SO-SMST method for the computation of canard segments and their transients. Appendix A includes some error estimates. In section 4 some results on the numerical implementation of the iterative methods via finite differences are presented. This section furthermore covers the use of the SO method in a Runge–Kutta scheme. Finally, in section 5 the SO-SMST method is applied to five different examples, including a nonlinear model of reciprocal inhibition with two slow and two fast variables. The results will be compared with trajectories computed using the SMST algorithm. As a further proof of concept, homoclinic connections for the FitzHugh–Nagumo model are computed.

Main results. The main theoretical results of the paper are collected in the following:

- Section 3 contains the most important result by demonstrating the SO-SMST method and how the iterative methods can be applied for the approximation of canard segments for saddle-type slow manifolds. Proposition A.1 and Proposition A.2 describe the errors associated with this approximation.
- In Corollary 4.2 a discrete version of the SO method is presented. It is this discretized version which will be used in SO-SMST. It is shown that the discretized method approximates the slow manifold of (1.1) up to an error of order $\mathcal{O}(e^{-c/\epsilon} + \epsilon^2 h^p)$. Here h describes the grid size in an order p finite difference operator.
- In Proposition 4.3 a discrete version of the SOF method is presented. This discretized version is used in SO-SMST. It is shown that this discretized method approximates the tangent spaces of the fibers of (1.1) up to an error of order $\mathcal{O}(e^{-c/\epsilon} + \epsilon^2 h^p)$.

Notation. All norms will be denoted by $\|\cdot\|$ including operator norms. This should not cause unnecessary confusion. If $\mathcal{U} \subset \mathbb{R}^n$, $n \in \mathbb{N}$, then $\mathcal{U} + i\chi$ will denote its complex χ-neighborhood:

$$\mathcal{U} + i\chi = \left\{ x \in \mathbb{C}^n | \sup_{y \in \mathcal{U}} \|x - y\| < \chi \right\}.$$

Consider $f : \mathcal{U} + i\chi \to \mathbb{C}^m$, $m \in \mathbb{N}$, being analytic and bounded. Then Cauchy-estimates apply to f in the following sense:

$$\sup_{x_0 \in \mathcal{U} + (\chi - \xi)} \|\partial_x f(x_0)\| \leq \frac{\sup_{x \in \mathcal{U} + \chi} \|f(x)\|}{\xi},$$

which will be written as

$$\|\partial_x f\|_{\chi - \xi} \leq \frac{\|f(x)\|_\chi}{\xi}. \tag{1.12}$$

Superscripts with $n \in \mathbb{N}_0$ will be used to denote partial sums such as

$$\eta^n = \sum_{i=0}^{n} \eta_i, \quad n \geq 0, \tag{1.13}$$

with each of the terms in the sum being enumerated through subscripts. Following this convention means that $\eta^0 = \eta_0$.

2. The iterative methods. In this section, the different iterative methods used in this paper are presented. Section 2.1 presents the SO method. This section also includes the modification due to Neishstadt [55]. Section 2.2 presents the SOF method.

2.1. The SO method: Approximation of the slow manifold. The SO method is an iterative approach to approximating an invariant slow manifold $M = \{y = \eta(x)\}$. The point of departure is the invariance equation

$$0 = -\epsilon\partial_x\eta(x)X(x,\eta(x)) + Y(x,\eta(x)), \tag{2.1}$$

which is obtained from (1.1) by enforcing the invariance of the graph $y = \eta(x)$. Basically, the SO method aims to solve this equation iteratively by considering the equations

$$-\epsilon\partial_x\eta^{n-1}(x)X(x,\eta^n(x)) + Y(x,\eta^n(x)) = 0 \tag{2.2}$$

for $n \geq 1$ starting from $\eta^0(x) = \eta_0(x)$ (see (1.4)) for $n = 1$. Here superscripts are used because the functions η^n will be obtained as partial sums. Recall (1.13). In the form (2.1) the SO method is also known as the iterative method of Fraser and Roussel; see [39]. Each step of the method involves the solution of a nonlinear equation. There are some simple alterations to the method that make the method computationally simpler. To present these, it is, however, advantageous to take a different viewpoint, which we highlight in the following:

1° The method can be initiated from any initial guess.
2° The method leads to exponentially accurate approximations.
3° The method applies to any M_0 satisfying (1.5).
4° The method can be altered so that if η_0 is known, then the method only involves the solution of linear equations.

Point 4° is perhaps not surprising because asymptotic expansions possess this property. However, most reduction methods are posed as fully nonlinear algebraic equations. The SO method does not require the slow-fast system to be written in the canonical form (1.1). In section 5.5 we consider the Lindemann mechanism

$$\begin{aligned}\dot{x} &= -x(x-y),\\ \dot{y} &= x(x-y) - \epsilon y,\end{aligned} \tag{2.3}$$

which is a slow-fast system where the slow and fast variables have not been properly identified. The iterative methods SO and SOF still apply to such systems [64, section 8] but it is unclear how to apply these methods if one is presenting these using ϵX for $\dot{x}$. Clearly, $\dot{x}$ is not small throughout for (2.3). Therefore, in this section we replace ϵX by

$$X^\epsilon \equiv \epsilon X$$

and consider the equations

$$\begin{aligned}\dot{x} &= X^\epsilon(x,y),\\ \dot{y} &= Y(x,y)\end{aligned}$$

instead of (1.1). Having said that, the main focus below will still be on the case where X^ϵ is ϵX and small throughout. All the proofs of the statements are based on the canonical slow-fast form (1.1). The reason why the iterative methods still apply when X^ϵ is not small throughout is that one in fact only needs

$$X^\epsilon(x, \eta_0(x)) = \mathcal{O}(\epsilon) \tag{2.4}$$

to be small with respect to ϵ.

Now suppose that (1.3) holds true so that M_0 is normally hyperbolic. Suppose furthermore that $y = \zeta_0(x)$ is an approximation to the slow manifold in the sense that it satisfies (2.1) up to a small error $\delta_0 = \sup_x \|\rho_0(x)\|$:

$$\rho_0(x) = -\partial_x \zeta_0(x) X^\epsilon(x, \zeta_0(x)) + Y(x, \zeta_0(x)). \tag{2.5}$$

The function $\rho_0 = \rho_0(x)$ is the *obstacle* to invariance of the slow manifold: if $\rho_0 \equiv 0$, then $\zeta_0 = \zeta_0(x)$ satisfies the invariance equation (1.3) and defines an invariant slow manifold. The approximation ζ_0 could be η_0 from (1.4). Then introduce y_0 by

$$y = \zeta_0(x) + y_0. \tag{2.6}$$

The transformation (2.6) *straightens out* the approximation of the slow manifold $y = \zeta_0(x)$ to $y_0 = 0$. The new equations for y_0 are

$$\begin{aligned} \dot{y}_0 = Y_0(x, y_0) &\equiv -\partial_x \zeta_0 X^\epsilon(x, \zeta_0(x) + y_0) + Y(x, \zeta_0(x) + y_0) \\ &= \rho_0(x) + A_0(x) y_0 + R_0(x, y_0), \end{aligned} \tag{2.7}$$

with ρ_0 as in (2.5),

$$A_0(x) = -\partial_x \zeta_0(x) \partial_y X^\epsilon(x, \zeta_0(x)) + \partial_y Y(x, \zeta_0(x)), \tag{2.8}$$

and $R_0 = \mathcal{O}(y_0^2)$. The equality in (2.7) is due to the Taylor expansion of Y_0 about $y_0 = 0$. The condition (1.3) implies that $\|(\partial_y Y|_{M_0})^{-1}\| \ll \epsilon^{-1}$. The matrix-valued function $A_0 = A_0(x)$ in (2.8) is therefore invertible for ϵ sufficiently small. Also since δ_0 is assumed to be small, the contraction mapping theorem implies that there exists a solution $\eta_1 = \eta_1(x) \approx -A_0(x)^{-1}\rho_0(x)$ of $Y_0(x, \eta_1) = 0$,

$$0 = \rho_0(x) + A_0(x)\eta_1 + R_0(x, \eta_1), \tag{2.9}$$

satisfying

$$\sup_x \|\eta_1(x)\| = \mathcal{O}(\delta_0). \tag{2.10}$$

The solution η_1 is analytic if X^ϵ, $Y \in C^\omega$. Note also that (2.9) (cf. (2.7)) can be written as

$$0 = -\partial_x \zeta_0(x) X^\epsilon(x, \zeta_0(x) + \eta_1(x)) + Y_0(x, \zeta_0(x) + \eta_1(x)). \tag{2.11}$$

Now, straighten out this new approximation $y_0 = \eta_1(x)$ of the slow manifold to $y_1 = 0$ by setting $y_0 = \eta_1(x) + y_1$ so that

$$\dot{y}_1 = Y_1(x, y_1) = \rho_1(x) + A_1(x)y_1 + R_1(x, y_1),$$

with

$$\rho_1(x) = -\partial_x \eta_1(x) X^\epsilon(x, \zeta_0(x) + \eta_1(x)). \tag{2.12}$$

If the vector-fields X^ϵ and Y are analytic, then one can apply Cauchy estimates (1.12) to estimate $\sup_x \|\partial_x \eta_1(x)\|$ in terms of $\sup_x \|\eta_1(x)\|$ on a smaller domain so that

$$\delta_1 \equiv \sup_x \|\rho_1(x)\| = \mathcal{O}(\epsilon\delta_0).$$

Hence, the new error is of the order of ϵ times the previous error. If one starts with $\zeta_0 = \eta_0$, then the error ρ_0 is $\mathcal{O}(\epsilon)$, and applying the procedure successively therefore directly leads to formal error estimates of the form $\mathcal{O}(\epsilon^{n+1})$, even when the vector-field U is only C^r, $r \geq n+1$, [39]. In terms of the original variables the approximation takes the form

$$y = \eta^n(x) = \zeta_0(x) + \eta_1(x) + \cdots + \eta_n(x).$$

The form in (2.11) immediately implies that the procedure can be written compactly as (2.2) for the approximation $y = \eta^n(x)$ of $y = \eta(x)$ satisfying (2.1). The alternative presentation of the SO method above, which is due to MacKay [52], has the advantage that it shows that one does not need to start the procedure from η_0. One could also just start from a guess $y = \zeta_0(x)$. The new error will still be ϵ times a C^1 estimate of the previous error; cf. (2.10). This explains point 1°.

The $\mathcal{O}(\epsilon^{k+1})$-estimate is not uniform in k: in the analytic case the domain of definition will eventually vanish when iteratively applying the Cauchy estimates. Using Neishtadt-type estimates it was, however, shown in [67] that the error can be made exponentially small. This explains point 2°. From this presentation, it is also clear that condition (1.3) is not needed. The importance is just that A_0 can be inverted, and for this (1.5) suffices. Therefore, the result of [67] not only apply to normally hyperbolic M_0's but also holds for normally elliptic slow manifolds, which confirmed a conjecture by MacKay [52]. This shows point 3°. A remarkable property of the SO method is in fact that it does not require A_0 to be bounded. Only A_0^{-1} is measured, making the method potentially useful in the analysis of slow ODE–fast PDE systems like the one in [65].

Computationally the SO method involves solving a nonlinear equation at each step n. In practice, the method therefore involves two loops: an outer loop updating n and an inner loop using, e.g., a Newton method for the solution η^n of the nonlinear equation (2.2). A result of Neishtadt in [55, Lemma 1] shows, however, that this inner loop is actually not necessary. Furthermore, the matrix-valued functions $A_i = A_i(x)$ that appear by the procedure outlined above do not need to be updated.

Proposition 2.1 (modified SO method, Lemma 1 in [55]). *Consider the slow-fast system* (1.1) *with* X^ϵ *and* Y *analytic on some complex* (χ, ν)*-neighborhood* $(x, y) \in (\mathcal{U} + i\chi) \times (\mathcal{V} + i\nu)$ *of*

$\mathcal{U} \times \mathcal{V}$ for some compact sets $\mathcal{U}$ and $\mathcal{V}$ in $\mathbb{R}^{n_s}$ and $\mathbb{R}^{n_f}$, respectively. Assume furthermore that $y = \zeta_0(x)$ is an approximation to the slow manifold so that $A_0 = A_0(x)$ (2.8) is invertible on $\mathcal{U}$ and that the error $\delta_0 = \|\rho_0\|_\chi$ in (2.5) is sufficiently small. Then for ϵ and δ_0 sufficiently small the transformation

$$y = \eta(x) + \tilde{y},$$

$\eta \equiv \eta^{N(\epsilon)} = \zeta_0 + \sum_{n=1}^{N(\epsilon)} \eta_n$, with $N(\epsilon) = \mathcal{O}(\epsilon^{-1})$ and $\eta_n = \eta_n(x)$ satisfying

$$\begin{aligned} \eta_n(x) &= -A_0(x)^{-1}\rho_{n-1}(x), \\ \rho_{n-1}(x) &= -\partial_x \eta_{n-1}(x) X^\epsilon(x, \eta^{n-1}(x)) + Y(x, \eta^{n-1}(x)), \end{aligned} \tag{2.13}$$

with A_0 as in (2.8), will on $\mathcal{U} \times \mathcal{V}$ transform (1.1) into

$$\dot{\tilde{y}} = \tilde{\rho}(x) + \tilde{A}(x)\tilde{y} + \tilde{R}(x, \tilde{y}),$$

with $\tilde{R} = \mathcal{O}(\tilde{y}^2)$ and

$$\|\tilde{\rho}\|_0 = \mathcal{O}(e^{-c/\epsilon}),$$

with $c > 0$ independent of ϵ and δ.

Proof. The proof is only sketched. For all the details see [55]. At the nth step the equations take the form

$$\begin{aligned} \dot{x} &= X^\epsilon_{n-1}(x, y_{n_1}), \\ \dot{y}_{n-1} &= Y_{n-1}(x, y_{n-1})\rho_{n-1}(x) + (A_0(x) + a_{n-1}(x))y_{n-1} + R_{n-1}(x, y_{n-1}), \end{aligned}$$

with $\delta_{n-1} = \|\rho_{n-1}\|_{\nu_{n-1}}$ and $\|a_{n-1}\|_{\nu_{n-1}} = \mathcal{O}(\epsilon\delta_0)$. The variable y_n is then introduced in accordance with (2.13),

$$y_{n-1} = \eta_n(x) + y_n,$$

giving

$$\begin{aligned} \dot{x} &= X^\epsilon_n(x, y_n), \\ \dot{y}_n &= \rho_n(x) + (A_0(x) + a_n(x))y_n + R_n(x, y_n), \end{aligned}$$

with

$$\rho_n = -\partial_x \eta_n(x) X^\epsilon_{n-1}(x, \eta_n) + a_{n-1}\eta_n + R_{n-1}(x, \eta_n), \tag{2.14}$$

$$a_n = -\partial_x \eta_n \partial_y X^\epsilon_{n-1}(x, \eta_n) + a_{n-1}(x) + \partial_y R(x, \eta_n), \tag{2.15}$$

and $X^\epsilon_n(x, y_n) = X^\epsilon_{n-1}(x, \eta_n + y_n)$. This gives

$$\begin{aligned} \|\rho_n\|_{\nu_n} &\le c_n \left(\epsilon\xi_n^{-1}\delta_{n-1} + \epsilon\delta_0\delta_{n-1} + \delta_{n-1}^2\right), \\ \nu_n &= \nu_{n-1} - \xi_n, \end{aligned}$$

for some $c_n > 0$, upon applying a Cauchy estimate. The last two terms are subordinate to the first term and one can therefore take $\xi_n = 4c_n\epsilon$ so that for δ_{n-1} sufficiently small,

$$\|\rho_n\|_{\nu_n} \leq \frac{1}{2}\delta_{n-1}.$$

Also $\|a_n\|_{\nu_n} = \mathcal{O}(\epsilon + \delta_0)$. One can then easily bound $c_n \leq 2c_0$ for ϵ sufficiently small and therefore uniformly bound the ξ_n's and take $\mathcal{O}(\epsilon^{-1})$ steps before the domain vanishes. This gives the exponential estimate. ■

This shows point 4°.

Remark 1. If $X^\epsilon = \epsilon X$ is small, then one could just replace A_0 in Proposition 2.1 by $\partial_y Y(x, \eta_0(x))$ since their difference is $\mathcal{O}(\epsilon)$. In the more general case, where the slow variables have not been properly identified, then $-\partial_x\eta_0\partial_y X^\epsilon(x, \eta_0) = \mathcal{O}(1)$, and replacing A_0 by $\partial_y Y(x, \eta_0(x))$ will not work. We will focus on this in greater detail in our paper [12].

Remark 2. The error ρ_0 is given by

$$\rho_0(x) = -\partial_x\zeta_0(x)X^\epsilon(x, \zeta_0(x)) + Y(x, \zeta_0(x)).$$

If $\zeta_0 = \eta_0$, then ρ_0 vanishes at any equilibrium of the form $(x, y) = (x^e, \eta_0(x^e))$ where $X^\epsilon(x^e, \eta_0(x^e)) = 0$ and $Y(x^e, \eta_0(x^e)) = 0$. Proceeding by induction on n using (2.13), it easily follows that the modified SO method will preserve this property so that $\rho_{n-1}(x^e) = 0$, and hence all of the approximations η^n will include equilibria.

Remark 3. The iterative method cannot be used to compute the canard orbits like those in [17] that appear as the intersection of an attracting slow manifold with a repelling one. This is because near the intersection the condition (1.5) is violated.

The following section describes the SOF method, which will be used to approximate the fibers.

2.2. The SOF method: Approximation of fibers. Let M_h be a slow manifold of saddle type, with a stable manifold $W^s(M_h)$ of dimension $n_s + n_f^s$ and an unstable manifold $W^u(M_h)$ of dimension $n_s + n_f^u$ ($n_f = n_f^s + n_f^u$). Then Fenichel's theory shows that there exists a local transformation $(u, v, w) \mapsto (x, y)$, with $\dim\{v\} = n_f^s$ and $\dim\{w\} = n_f^u$, mapping (1.1) into the *Fenichel normal form* [38],

$$\begin{aligned} \dot{u} &= \epsilon(U_0(u) + U_1(u, v, w)vw), \\ \dot{v} &= V(u, v, w)v, \\ \dot{w} &= W(u, v, w)w. \end{aligned} \tag{2.16}$$

Here $U_1(u, v, w) : \{v\} \times \{w\} \to \mathbb{R}^{n_s}$ is a bilinear function of v and w. The slow manifold is then given by $\{v = 0, w = 0\}$ with stable manifold $\{w = 0\}$ and unstable manifold $\{v = 0\}$. Note, in particular, that the slow vector-field is independent of the fast variables to linear order. The SOF method approaches this ideal. To explain this, first assume that the SO method has been applied for an approximation of the slow manifold $y = \eta(x)$. Then introduce y_0 by $y = \eta(x) + y_0$ so that

$$\begin{aligned} \dot{x} &= \Lambda^\epsilon(x) + \mu_0(x)y_0 + T(x, y_0), \\ \dot{y}_0 &= A(x_0)y_0 + R(x_0, y_0), \end{aligned}$$

neglecting the exponentially small terms. Here

$$\Lambda^\epsilon(x) = X^\epsilon(x, \eta(x)) = \mathcal{O}(\epsilon), \tag{2.17}$$

$$\mu_0(x) = \partial_y X^\epsilon(x, \eta(x)), \tag{2.18}$$

and

$$A(x) = -\partial_x \eta \partial_y X^\epsilon(x, \eta(x)) + \partial_y Y(x, \eta(x)), \tag{2.19}$$

while $R = \mathcal{O}(y_0^2), T = \mathcal{O}(\epsilon y_0^2)$. We then seek a transformation of the slow variables of the form

$$x = x_0 + \phi_0^\epsilon(x_0) y_0, \tag{2.20}$$

pushing the error $\gamma_0 = \|\mu_0\| = \mathcal{O}(\epsilon)$ to higher order in ϵ. Here $\phi_0^\epsilon \in \mathbb{R}^{n_s \times n_f}$, and the superscript ϵ is, as above, used to highlight that ϕ_0^ϵ will be $\mathcal{O}(\epsilon)$ if the slow-fast system is written in the canonical slow-fast form (1.1). Applying the transformation in (2.20) gives

$$\dot{x}_0 = \Lambda^\epsilon(x_0) + \{\partial_x \Lambda^\epsilon(x_0)\phi_0^\epsilon(x_0) + \mu_0(x_0) - \phi_0^\epsilon(x_0)A(x_0) + \mu_1(x_0)\}\, y_0 + \mathcal{O}(\epsilon y_0^2), \tag{2.21}$$

where μ_1 is

$$\mu_1(x_0) = -\partial_x \phi_0^\epsilon(x_0)\Lambda^\epsilon(x_0). \tag{2.22}$$

Here $\partial_x \phi_0^\epsilon$ times Λ is understood columnwise. In the SOF method one is looking for a solution $\phi_0^\epsilon = \phi^\epsilon$ that makes the curly brackets (2.21) vanish:

$$\partial_x \Lambda^\epsilon \phi^\epsilon + \mu_0 - \phi^\epsilon A + \mu_1 = 0. \tag{2.23}$$

As with the SO method, this is then approached iteratively, letting first $\phi_0^\epsilon(x) \approx \mu_0(x)A(x)^{-1} = \mathcal{O}(\gamma_0)$ solve the linear equation

$$\partial_x \Lambda^\epsilon \phi_0^\epsilon + \mu_0 - \phi_0^\epsilon A = 0. \tag{2.24}$$

Then the new error is

$$\gamma_1 \equiv \|\mu_1\| = \mathcal{O}(\epsilon\gamma_0),$$

using that $\Lambda^\epsilon = \mathcal{O}(\epsilon)$; cf. (2.17) in (2.22). This error is smaller than the previous one $\gamma_0 = \mathcal{O}(\epsilon)$. Iterating this procedure, one obtains the full SOF method.

Proposition 2.2 (the SOF method [64]). *Provided ϵ is sufficiently small, the function $\phi^\epsilon = \sum_{n=1}^{N(\epsilon)} \phi_n^\epsilon$, $N(\epsilon) = \mathcal{O}(\epsilon^{-1})$, where the ϕ_n^ϵ's satisfy the linear equations*

$$\begin{aligned} \partial_x \Lambda^\epsilon \phi_n^\epsilon + \mu_n - \phi_n^\epsilon A &= 0, \\ \mu_n &= -\partial_x \phi_{n-1}^\epsilon \Lambda^\epsilon, \\ \phi_{-1}^\epsilon &\equiv 0, \end{aligned}$$

with Λ^ϵ and A as in (2.17) and (2.19), respectively, solves (2.23) up to exponentially small error,

$$\partial_x\Lambda^\epsilon\phi^\epsilon+\mu_0-\phi^\epsilon Ay_0-\partial_x\phi^\epsilon\Lambda^\epsilon=\mathcal{O}(e^{-c/\epsilon}), \tag{2.25}$$

for some constant $c>0$ independent of ϵ.

As for the modified SO method, here it also suffices to replace $A=A_0+a$ in (2.24) by A_0 since $\|a\|=\mathcal{O}(\epsilon\delta_0)$ is small and therefore, along with $\partial_x\Lambda^\epsilon\phi_n^\epsilon$, can be combined into the error at the following step $\mu_{n+1}=-\partial_x\phi_n^\epsilon\Lambda^\epsilon+\partial_x\Lambda^\epsilon\phi_n^\epsilon-\phi_n^\epsilon a$. Indeed, the last two error terms are by Cauchy estimates subordinate to the first error, and the exponential estimates therefore also can be obtained in this case.

Proposition 2.3 (the modified SOF method). *Provided ϵ is sufficiently small, the function $\phi^\epsilon=\phi_0^\epsilon+\sum_{n=1}^{N(\epsilon)}\phi_n^\epsilon$, $N(\epsilon)=\mathcal{O}(\epsilon^{-1})$, where*

$$\begin{aligned}\phi_n^\epsilon&=\mu_nA_0^{-1},\\ \mu_n&=-\partial_x\phi_{n-1}^\epsilon\Lambda^\epsilon+\partial_x\Lambda^\epsilon\phi_{n-1}^\epsilon-\epsilon\phi_{n-1}^\epsilon a,\\ a(x)&=-\partial_x\eta\partial_yX^\epsilon(x,\eta)+\partial_yY_0(x,\eta)-A_0(x)\end{aligned} \tag{2.26}$$

for $n\geq 1$ and A_0 and μ_0 as in (2.8) and (2.18), respectively, solves (2.23) up to exponentially small error,

$$\partial_x\Lambda^\epsilon\phi^\epsilon+\mu_0-\phi^\epsilon Ay_0-\partial_x\phi^\epsilon\Lambda^\epsilon=\mathcal{O}(e^{-c/\epsilon}), \tag{2.27}$$

for some constant $c>0$ independent of ϵ.

Geometrically, the function ϕ^ϵ gives, through

$$\mathrm{Rg}\left(\begin{pmatrix}\phi^\epsilon(x)\\ I_f+\partial_x\eta(x)\phi^\epsilon(x)\end{pmatrix}+\mathcal{O}(e^{-c_2/\epsilon})\right), \tag{2.28}$$

an exponentially accurate approximation of the tangent spaces to the fibers at $y=\eta(x)$ [64]. Here $I_f=\text{identity}\in\mathbb{R}^{n_f\times n_f}$.

The following section combines the two iterative methods to obtain the SO-SMST method for the approximation of trajectories near a saddle-type slow manifold.

3. The SO-SMST method.

The outcome of SO and SOF consists of the functions η and ϕ^ϵ, respectively. The properties of these functions are such that if the transformation

$$(x_0,y_0)\mapsto(x=x_0+\phi^\epsilon(x_0)y_0,y=\eta(x)+y_0) \tag{3.1}$$

is applied to (1.2), then one obtains the equations of motion

$$\begin{aligned}\dot{x}_0&=\Lambda^\epsilon(x_0)+\mathcal{O}(\epsilon y_0^2),\\ \dot{y}_0&=A(x_0)y_0+\mathcal{O}(y_0^2).\end{aligned} \tag{3.2}$$

Recall that $\Lambda^\epsilon(x_0)=X^\epsilon(x_0,\eta(x_0))=\mathcal{O}(\epsilon)$; cf. (2.17). The case where $X^\epsilon=\epsilon X$ will again be the primary focus. In (3.2) the exponentially small terms have been ignored. The remainder $\mathcal{O}(\epsilon y_0^2)$ in (3.2) shall also be ignored, and here we will just consider

$$\dot{x}_0=\Lambda^\epsilon(x_0) \tag{3.3}$$

or

$$x_0' = \Lambda(x_0) \tag{3.4}$$

in terms of the slow time $\tau = \epsilon t$ and where $\Lambda(x_0) = X(x_0, \eta(x_0))$. This formally decouples the slow variables from the fast ones. Compare with (2.28), which corresponds to projecting along the tangent space of the fibers based at $(x, \eta(x))$. A simpler but less accurate approach to obtain a formal decoupling of the equations would be to base the projection on $(1.1)_{\epsilon=0}$ and the tangent spaces of the fibers at $\epsilon = 0$. This corresponds to ignoring ϕ^ϵ in the transformation above and instead just considering

$$\begin{aligned}&(x_0, y_0) \mapsto (x = x_0, y = \eta(x) + y_0) \text{ and decouple the}\\ &\text{equations by ignoring a remainder of the form } \mathcal{O}(\epsilon y_0).\end{aligned} \tag{3.5}$$

This approach is, as highlighted by the orders $\mathcal{O}(\epsilon y_0^2)$ and $\mathcal{O}(\epsilon y_0)$ in (3.2) and (3.5), respectively, less accurate. It assumes that the fibers are vertical. See also Figure 4.1 in [64]. The approach (3.5) is therefore particularly inaccurate in comparison with (3.1) if the slow and fast variables have not been properly identified. The error in (3.5) is then $\mathcal{O}(y_0)$ rather than $\mathcal{O}(\epsilon y_0)$. See [64, section 8] and section 5.5 below.

The error from replacing (3.2) with (3.3) will be further quantified in Appendix A. However, within this approximation, the fast variables can be solved using

$$\dot{y} = Y(x, y), \tag{3.6}$$

$$x = x_0 + \phi^\epsilon(x_0)(I_f + \partial_x \eta(x_0)\phi^\epsilon(x_0))^{-1}(y - \eta(x_0)), \tag{3.7}$$

which is a nonautonomous system once $x_0 = x_0(\tau)$ has been obtained from (3.4). The equation for $x = x(x_0, y)$ (see (3.7)) has been obtained by inserting $x = x_0 + \phi^\epsilon(x_0)y_0$ into $y_0 = y - \eta(x)$ and Taylor expanding about $y_0 = 0$. This introduces an error of $\mathcal{O}(\epsilon^3 y_0^2)$, which is subordinate to the remainder $\mathcal{O}(\epsilon y_0^2)$, which was ignored in (3.2).

It will also be useful to invert $x = x_0 + \phi^\epsilon(x_0)y_0$ for x_0 (see section 5.4). By Taylor expansion about $y_0 = 0$ the approximation

$$x_0 = x - \phi^\epsilon(x)y_0 + \mathcal{O}(\epsilon^2 y_0^2) \tag{3.8}$$

is obtained, where $\phi^\epsilon = \mathcal{O}(\epsilon)$.

The main purpose of this paper is to use this principle near a saddle-type slow manifold to construct the type of trajectories that are computed by the SMST algorithm. Consider, e.g., a base trajectory $x_0 = x_0(\tau)$ solving (3.4) with $x_0(0) = x_{00}$ and $x_0(T) = x_{0T}$. This will be obtained by applying a quadrature to (3.4). We will return to this in section 4.1. The aim is then to compute an approximation of a trajectory connecting to such a base trajectory, in the sense that it decays to the base trajectory exponentially fast at one end and escapes from it exponentially fast at the other end. This is done as in the SMST algorithm [35] by specifying the stable components $y_{s0} = \pi_s^z(y - \eta(x(0)))$ at $t = 0$ and unstable ones $y_{uT} = \pi_u^z(y - \eta(x(T)))$ at the other end $t = T/\epsilon$. In particular, the approximation

$$y - \eta(x) = (I_f + \partial_x \eta(x_0)\phi^\epsilon(x_0))^{-1}(y - \eta(x_0)) + \mathcal{O}(\epsilon^2 y_0^2),$$

also used in (3.7), is used to write these components as

$$y_{s0} = \pi_s^z (I_f + \partial_x \eta(x_{00}) \phi^\epsilon(x_{00}))^{-1} (y - \eta(x_{00}))$$

and

$$y_{uT} = \pi_u^z (I_f + \partial_x \eta(x_{0T}) \phi^\epsilon(x_{0T}))^{-1} (y - \eta(x_{0T})),$$

respectively. Recall here the definitions of $\pi_{s,u}^z$ in (1.8). In contrast to the SMST algorithm, however, collocation is only performed on the fast y-space, as the base trajectory $x_0 = x_0(\tau)$ solving (3.4) has been obtained by direct integration. Moreover, one need only consider time intervals of order $t = \mathcal{O}(1)$ in each end. This means that the vector-field in this collocation problem has no ϵ^{-1} factor, and hence the Jacobian will be well-conditioned. The length of the time intervals can be estimated through the eigenvalues of $\partial_y Y(x, \eta(x))$. Suppose that $r_0 = \|y_{s0}\|$ is small and that $\lambda_s > 0$ is a lower estimate of the absolute values of the real parts of the eigenvalues of $\partial_y Y(x, \eta(x))$ with negative real parts; then

$$t_0 = -\lambda_s^{-1} \log\left(\frac{\text{tol}}{r_0}\right)$$

is an estimate of how long it takes y_0 to decrease below a given tolerance tol ($\ll \epsilon r_0^2$; cf. (A.1) in Appendix A). At $t = t_0$ we then enforce the condition that the "unstable components" of $y = y(t_0)$ vanish. That is,

$$\pi_u^z (y - \eta(x_0(\epsilon t_0))) = 0.$$

At the other end, we then let $r_T = \|y_{uT}\|$ and suppose that $\lambda_u > 0$ is a lower estimate of the real parts of the eigenvalues of $\partial_y Y(x, \eta(x))$ with positive real part. Then

$$t_1 = -\lambda_u^{-1} \log\left(\frac{\text{tol}}{r_T}\right)$$

is an estimate of how long it takes y_0 to decrease below, now in backwards time, a given tolerance tol ($\ll \epsilon r_T^2$). At $t = T/\epsilon - t_1$ one therefore enforces the condition that the "stable components" of $y = y(T/\epsilon - t_1)$ vanish. That is,

$$\pi_s^z (y - \eta(x_0(T - \epsilon t_1))).$$

This defines two boundary value problems on the fast space. They are solved by the same collocation principle as used in the SMST algorithm on the full space using divisions of the fast time intervals $t \in [0, t_0]$ and $t \in [T/\epsilon - t_1, T/\epsilon]$ by a time step Δt. For $t \in (t_0, T/\epsilon - t_1)$, one sets $y(t) = \eta(x_0(t))$, that is, $y_0 = 0$. Finally, $x = x(\tau)$ is obtained from (3.7).

Remark 4. A consequence of the analysis in Appendix A (see also Remark 8) is that for the trajectories computed in [35], where the stable and unstable components are taken from the critical manifold, setting $(1.9)_{\epsilon=0}$ and $(1.10)_{\epsilon=0}$, respectively, to 0, the error of the SO-SMST method is of order $\mathcal{O}(\epsilon^3)$.

4. Numerical implementation of the iterative methods. If the nonlinear equation for the critical manifold $y = \eta_0(x)$ can be solved explicitly, then SO and SOF can be implemented into a computer algebra system (CAS), such as Maple or Mathematica, to obtain very accurate closed-form approximations of the slow manifold and the tangent spaces to the fibers. There are other methods that could also be used to achieve this. If an explicit small parameter ϵ can be identified, then such accurate closed-form approximations are even obtainable using direct asymptotic expansions. But whereas closed-form approximations could potentially be useful in some specific cases, they have clear disadvantages in general. First, the number of terms to include to obtain a desired accuracy depends in a nontrivial way on the position in phase space. Second, the expressions are typically very lengthy, and just the evaluation of such expressions will involve many operations which, if combined with numerical integration, could be costly. Finally, closed-form approximations are highly inflexible: if the model is slightly modified, then one needs to redo the CAS computations.

A numerical implementation of the SO and SOF methods circumvents the highlighted issues of a CAS implementation. The only obstacle is the fact that one needs to approximate derivatives of the approximations (see $\partial_x \eta_{n-1}$ in (2.13) and $\partial_x \phi^\epsilon_{n-1}$ in (2.26)) to obtain improved approximations. For this, the differential operator ∂_x that appears in these expressions can be replaced by a finite difference operator δ^h_x satisfying

$$(\partial_x - \delta^h_x) f = \mathcal{O}(h^p) \tag{4.1}$$

for all smooth f. As an example, one could take

$$\delta^h_{x_i} f(x) = \frac{f(x + he_i) - f(x - he_i)}{2h}, \quad i = 1, \dots, n_s, \tag{4.2}$$

with $(e_i)_j = \delta_{ij}$, δ_{ij} being Kronecker's delta, and set

$$\delta^h_x f = (\delta^h_{x_1} f \ \cdots \ \delta^h_{x_{n_s}} f).$$

Then $p = 2$ since

$$\|(\partial_{x_i} - \delta^h_{x_i}) f(x)\| \le \frac{\sup_{t\in[0,1]} \|\partial^3_{x_i} f(x + the_i)\|}{6} h^2. \tag{4.3}$$

Cauchy-type estimates also apply to δ^h_x in the sense that

$$\|\delta^h_x f\|_{\chi-\xi} \le \frac{C_p \|f\|_\chi}{\xi}, \quad C_p \ge 1, \tag{4.4}$$

provided $h < \xi$ is sufficiently small and that f is analytic. For (4.2), for example, with $h \le \xi/2$ we have

$$\|\delta^h_{x_i} f\|_{\chi-\xi} \le \frac{3\|f\|_\chi}{\xi},$$

using (4.3) and a Cauchy estimate of $\|\partial^3_{x_i} f\|_{\chi-\xi+h}$. Therefore,

$$\|\delta^h_x f\|_{\chi-\xi} \le \frac{3n_s \|f\|_\chi}{\xi},$$

and $C_p = 3n_s > 1$ in this case. The discretized version of the invariance equation

$$-\epsilon\delta_x^h\eta^h X(x,\eta^h) + Y(x,\eta^h) = 0 \tag{4.5}$$

can then be solved by the SO principle to obtain an approximate solution η^h with exponential small error:

$$-\epsilon\delta_x^h\eta^h(x)X(x,\eta^h(x)) + Y(x,\eta^h(x)) = \mathcal{O}(e^{-c_p/\epsilon}). \tag{4.6}$$

Here c_p is independent of ϵ and h.

Proposition 4.1. *Consider a finite difference operator δ_x^h satisfying (4.1) and (4.4). Provided ϵ is sufficiently small, applying the* SO *method to (4.5) gives an approximate solution η^h that satisfies*

$$-\partial_x\eta^h X^\epsilon(x,\eta^h(x)) + Y(x,\eta^h(x)) = \mathcal{O}(\epsilon\|(\partial_x - \delta_x^h)\eta^h\| + e^{-c_p/\epsilon}) = \mathcal{O}(\epsilon h^p + e^{-c_p/\epsilon}). \tag{4.7}$$

Proof. We write

$$\partial_x\eta^h = \delta_x^h\eta^h - (\partial_x^h - \delta_x^h)\eta^h$$

in (4.7) and use (4.1) and (4.6) to estimate the error. ∎

From $Y(x,\eta_0(x)) = 0$ one can obtain $\partial_x\eta_0 = -(\partial_y Y)^{-1}\partial_x Y$ in the first step of the iteration. The error from replacing ∂_x with δ_x^h then does not appear before the second step. This gives rise to the improved order $\mathcal{O}(\epsilon^2 h^p)$ in (4.7).

Corollary 4.2. *Suppose $\eta_0 = \eta_0(x)$ is known. Then, provided ϵ is sufficiently small, applying the procedure*

$$\eta_n^h = -A_0^{-1}\rho_{n-1}, \tag{4.8}$$

$$\begin{aligned}\rho_{n-1} &= -(\partial_x\eta_0(x) + \delta_x^h(\eta^{n-1,h}(x) - \eta_0(x)))X^\epsilon(x,\eta^{n-1,h}) + Y(x,\eta^{n-1,h}),\\ A_0(x) &= -\partial_x\eta_0\partial_y X^\epsilon(x,\eta_0) + \partial_y Y(x,\eta_0),\\ \eta^{n,h} &= \eta_0 + \sum_{k=1}^{n}\eta_k^h\end{aligned} \tag{4.9}$$

generates an approximate solution $\eta^h = \eta_0 + \sum_{n=1}^{N(\epsilon)}\eta_n^h$, $N(\epsilon) = \mathcal{O}(\epsilon^{-1})$, satisfying

$$-\partial_x\eta^h X^\epsilon(x,\eta^h) + Y(x,\eta^h) = \mathcal{O}(\epsilon\|(\partial_x - \delta_x^h)(\eta^h - \eta_0)\| + e^{-c_p/\epsilon}) = \mathcal{O}(\epsilon^2 h^p + e^{-c_p/\epsilon}). \tag{4.10}$$

The derivative $\partial_x\eta_0$ is obtained through $Y(x,\eta_0) = 0$:

$$\partial_x\eta_0 = -(\partial_y Y(x,\eta_0))^{-1}\partial_x Y(x,\eta_0).$$

Note how $\partial_x\eta^{n-1,h}$ is approximated as $\partial_x\eta_0 + \delta_x^h(\eta^{n-1,h} - \eta_0)$ in (4.9). We further stress the simplicity of this method: it only requires the first partial derivatives of the vector-field.

It is easy to obtain a similar result for the discretization of the SOF method.

Proposition 4.3. *Assume that the conditions* (4.1) *and* (4.4) *hold true, and let* η^h *be given as in Proposition* 4.1. *Then provided* ϵ *is sufficiently small, the function* $\phi^{\epsilon,h} = \sum_{n=0}^{N(\epsilon)} \phi_n^{\epsilon,h}$, $N(\epsilon) = \mathcal{O}(\epsilon^{-1})$, *where*

$$\begin{aligned}
\phi_n^{\epsilon,h} &= \mu_n A_0^{-1},\\
\mu_n &= -\delta_x^h \phi_{n-1}^{\epsilon} X^{\epsilon}(x,\eta^h)\\
&\quad + \left(\partial_x X^{\epsilon}(x,\eta^h) + \partial_y X^{\epsilon}(x,\eta^h(x))\delta_x^h \eta^h(x)\right)\phi_{n-1}^{\epsilon,h}\\
&\quad - \phi_{n-1}^{\epsilon,h} a,\\
A_0(x) &= -\partial_x \eta_0 \partial_y X^{\epsilon}(x,\eta_0) + \partial_y Y_0(x,\eta_0),\\
a(x) &= -\delta_x^h \eta^h(x)\partial_y X_0^{\epsilon}(x,\eta^h(x)) + \partial_y Y_0(x,\eta^h) - A_0(x),
\end{aligned}$$

solves (2.23) *up to the error*

$$\mathcal{O}(\epsilon \|(\partial_x - \delta_x^h)\phi^{\epsilon,h}\| + e^{-c_p/\epsilon}). \tag{4.11}$$

Proof. One can proceed as in Proposition 4.1. Note that

$$\partial_x \Lambda^{\epsilon} = \partial_x X^{\epsilon}(x,\eta^h) + \partial_y X^{\epsilon}(x,\eta^h)\partial_x \eta^h(x)$$

has been replaced by

$$\partial_x X^{\epsilon}(x,\eta^h) + \partial_y X_0(x,\eta^h)\delta_x^h \eta^h. \qquad \blacksquare$$

Remark 5. If $X^{\epsilon} = \epsilon X$ and the slow and fast variables have been properly identified, then $\|\phi^{\epsilon}\| = \mathcal{O}(\epsilon)$ (cf. (2.24)) and the order in (4.11) will be $\mathcal{O}(\epsilon^2 h^p + e^{-c/\epsilon})$ as in Corollary 4.2. If $X^{\epsilon} = \mathcal{O}(1)$ and only satisfies (2.4), then $\phi^{\epsilon} = \mathcal{O}(1)$ and the error in (4.11) is therefore only $O(\epsilon h^p + e^{-c/\epsilon})$, which is slightly less accurate. To improve it by a factor of ϵ, one could do as in Corollary 4.2 and replace $\delta_x^h \phi^{\epsilon,h}$ by $\partial_x \phi_0^{\epsilon} + \delta_x^h(\phi^{\epsilon,h} - \phi_0^{\epsilon})$ and use the fact that $\partial_x \phi_0^{\epsilon}$ can be obtained analytically from (2.24).

The SO-SMST method requires the propagation of the slow variables on the slow manifold. For this the discretized SO method will be integrated into a Runge–Kutta quadrature scheme as explained in the following section.

4.1. Modified Runge–Kutta scheme and h-grid. On the slow manifold, the motion of the slow variables is given in terms of the reduced system:

$$x' = \Lambda(x) \equiv X(x,\eta(x)). \tag{4.12}$$

Recall that $()'$ denotes differentiation with respect to the slow time $\tau = \epsilon t$. The solution of this reduced system can be approximated by applying a quadrature scheme. A classical 4th order Runge–Kutta scheme will be used. The modification from one scheme to another is straightforward and not important for what we will be presenting.

Starting from $x(\tau) = x_0$, the 4th order Runge–Kutta method approximates $x(\tau+\Delta\tau) = x_1$ as

$$x_1 = x_0 + \frac{1}{6}\left(\kappa_1 + 2\kappa_2 + 2\kappa_3 + \kappa_4\right),$$

where

$$\begin{aligned}\kappa_1 &= \Delta\tau X(x_0, \eta(x_0)),\\ \kappa_2 &= \Delta\tau X(x_0 + 0.5\kappa_1, \eta(x_0 + 0.5\kappa_1)),\\ \kappa_3 &= \Delta\tau X(x_0 + 0.5\kappa_2, \eta(x_0 + 0.5\kappa_2)),\\ \kappa_4 &= \Delta\tau X(x_0 + \kappa_3, \eta(x_0 + \kappa_3)).\end{aligned}$$

See, e.g., [13]. Here $\Delta\tau$ is the time step on the slow time scale. The local error is $\mathcal{O}(\Delta\tau^5)$, while the accumulated error is $\mathcal{O}(\Delta\tau^4)$. The Runge–Kutta scheme will therefore require the determination of $\eta(x)$ at the following different x-values:

$$x = x_0, \quad x_0 + 0.5\kappa_1, \quad x_0 + 0.5\kappa_2, \quad \text{and} \quad x_0 + \kappa_3. \tag{4.13}$$

This is where the discretized SO method will be used. To explain the construction of the finite difference operator δ_x^h (4.1), consider for example the determination of $\eta(x_0)$. A grid is introduced around x_0, and δ_x^h is then determined by Lagrange interpolation polynomials derived from function values at the 3^{n_s} points:

$$x_0 + e_h \quad \text{with} \quad (e_h)_i = 0, \quad \text{or} \pm h \quad \text{for} \quad i = 1, \ldots, n_s. \tag{4.14}$$

This gives $p = 2$ in (4.7) and (4.10). The following is important: since h is small, $A_0 = A_0(x)$ can be taken to be a constant on the h-grid. The error from this can be collected into $a_n = \mathcal{O}(\epsilon\delta_0)$ (cf. (2.15)) and therefore does not change the result. The LU-decomposition of A_0 therefore can be stored outside the iteration in n. Cf. (4.7); by letting $\epsilon h^2 \sim \Delta\tau^5$ or $h^2 \sim \epsilon^{-1}\Delta\tau^5$, one can match the order of the Runge–Kutta scheme to the order of the approximation of the slow manifold η. If η_0 is used explicitly as described in Corollary 4.2, then instead one can let

$$h^2 \sim \epsilon^{-2}\Delta\tau^5. \tag{4.15}$$

This quadrature scheme for the propagation will be referred to as the *modified Runge–Kutta scheme*. For moderate values of n_s, say 1, 2, or 3, the cost involved in each time step is comparable to the cost of a single step in an implicit method of the same order applied to the full system. Indeed, for both methods, the computational cost is expected to be dominated by the cost required to obtain a solution of a linear equation. The modified Runge–Kutta scheme requires the solution (4.8), while an implicit method requires the solution of another linear equation on the full space in the application of the Newton method. For larger values of n_s, the reduced quadrature suffers from having to resolve ∂_x using 3^{n_s} number of points in the h-grid.

Remark 6. As an alternative to the method outlined above, one could compute the slow manifold on a larger grid and then interpolate to obtain the values of κ_i, $i = 1, 2, 3, 4$, needed in the Runge-Kutta scheme. This, however, involves unnecessarily many computations. The direct use of the SO method in the forward integration only involves computations of the slow manifold, where it is needed for the propagation of the variables.

The following section combines several examples for testing and demonstrating the SO-SMST method.

5. Examples. This section includes five different examples.

- In section 5.1 a toy model is considered in order to test the iterative methods and demonstrate their stated properties.
- Section 5.2 includes the boundary value problem (1.11), where the SO-SMST method gives the desired solution up to exponentially small terms.
- Section 5.3 considers a nonlinear model of reciprocal inhibition. A boundary value problem with fixed boundaries is considered. The results from applying the SO-SMST method to this problem are compared to results obtained from the SMST method. It is demonstrated that there are no issues with obtaining a solution using the SO-SMST method for $\epsilon \to 0$.
- In section 5.4 the FitzHugh–Nagumo model is considered. A homoclinic solution is computed, and it is shown how the SO-SMST can be combined with other methods to compute a full orbit. The projection based on the tangent spaces to the fibers through the function ϕ^ϵ is also compared with the result of just using the tangent spaces with $\epsilon = 0$ as explained in (3.5). An improvement in accuracy by a factor of 10^{-3} is observed when the projection is based on ϕ^ϵ without any detectable difference in computational time.
- In section 5.5 the SO-SMST method is successfully applied to the Lindemann mechanism [14, 31, 64], which is an example of a system where the slow and fast variables have not been properly identified. Applying the "naive" projection as described in (3.5) gives rise to $\mathcal{O}(1)$-errors.

5.1. Testing the iterative methods: A toy example with a saddle-type slow manifold. Consider the following toy problem:

$$\dot{x} = \epsilon \begin{pmatrix} \cos x_1 + y_1 + y_2 \cos x_2 \\ -\sin x_2 + y_2 + y_1 \sin x_1 \end{pmatrix}, \tag{5.1}$$

$$\dot{y} = \begin{pmatrix} \cos x_2 - y_1 \\ -\sin x_1 + y_2 \end{pmatrix}. \tag{5.2}$$

From the Jacobian matrix $\partial_y Y = \operatorname{diag}(-1, 1)$ it follows that the slow manifold is of saddle type. Since the problem (5.1) is linear in the fast variables the SO method can then be used to compute $\eta = (\eta_1, \eta_2)$ explicitly using Maple. Terms up to and including order ϵ^8 will be used in the following. In Figure 1 such a reference CAS solution is compared with a numerical solution η^h obtained using the discretized SO method (see Proposition 4.1 and Corollary 4.2) at $x = (-0.5, -0.7)$. The finite difference operator δ_x^h was 2nd order ($p = 2$) and based on Lagrange interpolation, as explained after (4.14). In both figures $h = \epsilon$. Figure 1(a) is obtained using Proposition 4.1, whereas Figure 1(b) is obtained using Corollary 4.2 and

$$\eta_0 = \begin{pmatrix} \cos x_2 \\ \sin x_1 \end{pmatrix}$$

explicitly. From the log-log scale in Figure 1 we numerically determine the orders of the approximations to be $\mathcal{O}(\epsilon^3)$ and $\mathcal{O}(\epsilon^4)$. This is in agreement with the analysis above; see $(4.7)_{h=\epsilon}$ and $(4.10)_{h=\epsilon}$, respectively, with $p = 2$.

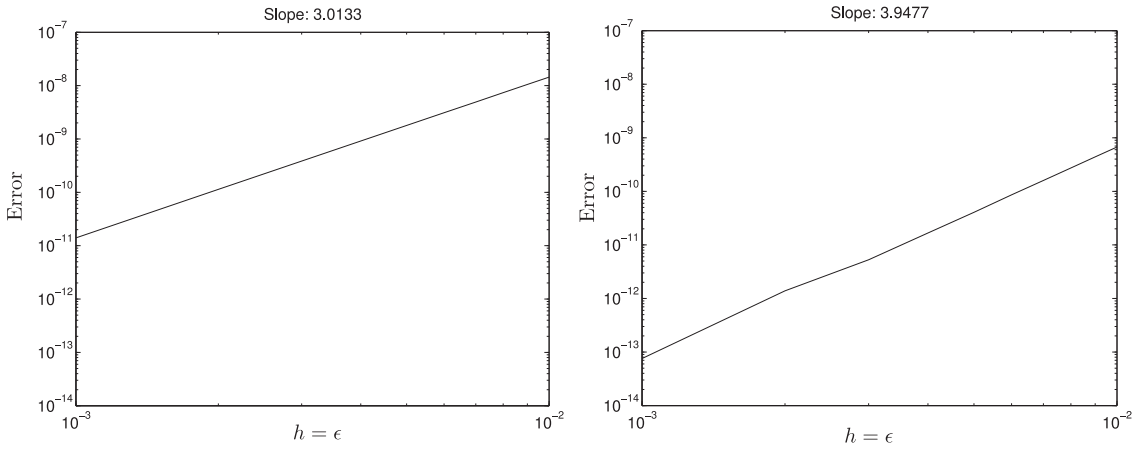


Figure 1. *The error $\|\eta^h - \eta\|$ for the problem (5.1) at $x = (-0.5, -0.7)$ for $h = \epsilon$ and as a function of ϵ. The approximation η^h in* (a) *is obtained using Proposition 4.1, while η^h in* (b) *is based on Corollary 4.2 and the explicit use of $\partial_x \eta_0$. The finite difference operator is 2nd order ($p = 2$) with respect to the grid size h. The reference solution η is obtained using Maple (accurate up to $\mathcal{O}(\epsilon^9)$). The slopes of ≈ 3 and ≈ 4 correspond to orders of $\mathcal{O}(\epsilon^3)$ and $\mathcal{O}(\epsilon^4)$, respectively, which are in agreement with the analysis.*

In Figure 2 the results of applying the modified Runge–Kutta scheme to

$$x' = \begin{pmatrix} \cos x_1 + \eta_1 + \eta_2 \cos x_2 \\ -\sin x_2 + \eta_2 + \eta_1 \sin x_1 \end{pmatrix}$$

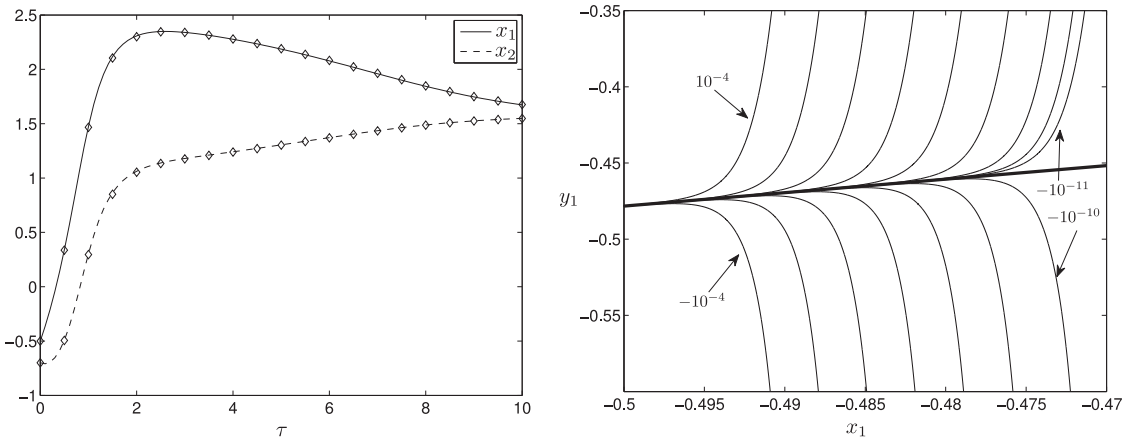
for different values of $\Delta\tau$ and $\epsilon = 10^{-3}$ are compared with a high-precision reference solution obtained using MATLAB function `ode45` applied to (4.12). The integration is initialized at $x(0) = (-0.5, -0.7)$ and integrated up until $\tau = 10$. The absolute and relative tolerances of `ode45` were set to 10^{-12}, and $\eta = (\eta_1, \eta_2)$ from the Maple computation, again including terms up to order ϵ^8, was used in the `ode45` solver to obtain an accurate reduction to the slow manifold. In the modified Runge–Kutta scheme the method described in Corollary 4.2 was used, with η_0 and $\partial_x \eta_0$ used explicitly, stopping the SO iteration when the error

$$\| - \delta_x^h \eta X^\epsilon(x, \eta) + Y(x, \eta)\| \tag{5.3}$$

had reached below a tolerance tol, which was set to be 10^{-12}. The grid size was set to be

$$h = \min\{10^{-2}, 0.1\Delta\tau^5/\epsilon^2\}.$$

The factor of 0.1 was introduced as a "safety factor" aiming to ensure that the error from the approximation of η was subordinate to the error of the Runge–Kutta scheme; see also (4.15). Figure 2(a) compares the reference `ode45` solutions x_1 $(-)$ and x_2 $(--)$ with the solutions $(\diamond)$ obtained by the modified Runge–Kutta scheme for $\Delta\tau = 0.5$. It is observed that the $\diamond$'s agree with the accurate reference solutions. The maximal deviation was 3×10^{-3} for this value of $\Delta\tau$. Figure 2(b) shows the result of direct simulation for 16 different initial conditions that



(a) (b)

Figure 2. *The results of applying the modified Runge–Kutta scheme to the problem* (5.1). *Figure* (a) *compares the solution obtained by the modified Runge–Kutta scheme* ($\diamond$) *with an accurate reference solution* $(-,--)$. *Figure* (b) *shows the result of an accurate direct simulation on the full system for* 16 *initial conditions displaced by distances of* $\pm 10^{-4}, \pm 10^{-5}, \ldots, \pm 10^{-11}$ *from the slow manifold along its unstable directions. Of all the pairs, only for the one with* $\pm 10^{-11}$ *do the trajectories jump in the same direction.*

are obtained as displacements by $\pm 10^{-4}, \pm 10^{-5}, \ldots, \pm 10^{-11}$ from the slow manifold along its unstable directions. The MATLAB function `ode15s` was used with tolerances set to 10^{-12} for the propagation on the full space. Of all the pairs, only for the one with $\pm 10^{-11}$ do the trajectories jump in the same direction. This gives reason to believe that the slow manifold is correct up to $\pm 10^{-10}$ but not more accurate than $\pm 10^{-11}$.

The function ϕ^ϵ can also be computed explicitly for the toy problem (5.1):

$$\phi^\epsilon = \epsilon \begin{pmatrix} -1 & \cos x_2 \\ -\sin x_1 & 1 \end{pmatrix} + \mathcal{O}(\epsilon^2).$$

Again Maple is used with terms up to order ϵ^8. In Figure 3 this is compared with a numerical solution $\phi^{\epsilon,h}$ at $x = (-0.5, -0.7)$ taking again $h = \epsilon$. In agreement with the analysis (cf. $(4.11)_{h=\epsilon}$ with $p = 2$), the slope in the log-log scale is ≈ 4.

5.2. An example where SO-SMST gives the result up to exponentially small terms.

Consider the linear, singular perturbed boundary value problem

$$\epsilon u''(\tau) + u'(\tau) = 1, \quad u(0) = 1 = u(1) \tag{5.4}$$

taken from [41]. Setting

$$x = u \quad \text{and} \quad y = u'$$

gives the slow-fast system

$$\begin{aligned} \dot{x} &= \epsilon y, \\ \dot{y} &= 1 - y, \end{aligned}$$

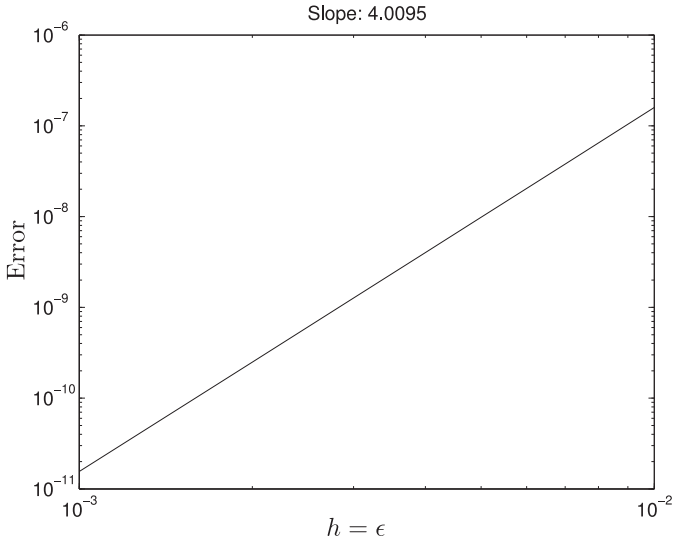


Figure 3. *Comparison of $\phi^{\epsilon,h}$ (obtained using Proposition 4.3 with δ_x^h a 2nd order finite difference) with ϕ^ϵ (obtained using Maple, accurate up to order $\mathcal{O}(\epsilon^9)$) for the toy problem (5.1). The grid size h is set equal to ϵ and the difference $\|\phi^{\epsilon,h} - \phi^\epsilon\|$ is computed for different values of ϵ. The computations are based on $x = (-0.5, -0.7)$. The slope ≈ 4 corresponds to an order of $\mathcal{O}(\epsilon^4)$ which is in agreement with Proposition 4.3 and $(4.11)_{h=\epsilon}$.*

with respect to the fast time $t = \epsilon^{-1}\tau$. Here $y = 1$ is a normally hyperbolic invariant manifold. The SOF method gives $\eta = 1$ and $\phi^\epsilon = -\epsilon$ both exact in one step. Since the problem is linear and ϕ^ϵ is independent of x, this also implies that the SO-SMST method gives

$$x(\tau) = \tau + e^{-\tau/\epsilon},$$

which agrees with the true solution of (5.4),

$$x(\tau) = \tau + \frac{e^{-\tau/\epsilon} - e^{-1/\epsilon}}{1 - e^{-1/\epsilon}},$$

up to exponentially small terms. This is not the case for the classical SMST method. See [41].

5.3. Numerical results for a model for reciprocal inhibition. To demonstrate the SO-SMST method further, we consider a model for a pair of neurons coupled by reciprocal inhibition [58],

$$\begin{aligned}
\dot{q}_1 &= \epsilon(-q_1 + sv_1),\\
\dot{q}_2 &= \epsilon(-q_2 + sv_2),\\
\dot{v}_1 &= -\left(v_1 - a\tanh\left(\frac{\sigma_1 v_1}{a}\right) + q_1 + \omega f(v_2)(v_1 - r)\right),\\
\dot{v}_2 &= -\left(v_2 - a\tanh\left(\frac{\sigma_2 v_2}{a}\right) + q_2 + \omega f(v_1)(v_2 - r)\right),
\end{aligned}$$

with

$$f(x) = (1 + \exp(-4\gamma(x - \theta)))^{-1}.$$

Here the fast variables v_1 and v_2 are interpreted as the membrane potential of two neurons coupled synaptically through the terms involving f. The slow variables q_1 and q_2 represent the gating of membrane channels in the neurons. The model does not incorporate the fast membrane currents, and in that sense it is slightly caricatural. However, further reduced models have been used to study reciprocal inhibition of a pair of neurons [61, 69]. The model was also considered in [35], the paper presenting the SMST algorithm. The parameter values

$$\omega = 0.03,\ \gamma = 10,\ r = -4,\ \theta = 0.01333,\ a = 1,\ s = 1,\ \sigma_1 = 3, \quad \text{and} \quad \sigma_2 = 1.2652372051,$$

also used in [33, 35], will be used henceforth.

Computation of base trajectory. Figure 4 shows two projections (thick lines) of a trajectory segment on the slow manifold, which includes the segment B' in Figure 6(c) of [33], which was computed using the modified Runge–Kutta scheme with the discretized SO method based on Corollary 4.2. The time T was set to 0.5. In forward time the flow is from the lower left to the upper right. Here $\Delta\tau = 10^{-2}$ and $h = 10^{-4} < \epsilon^{-1}\Delta\tau^{5/2} = 0.01$. To compute such trajectories using the SMST algorithm it is expected that one has to introduce some sort of continuation to *pull out the transitions at the ends* [45]. The segment computed here is much longer than the one in [33]. To realize this one can, for example, compare Figure 4(b) with Figure 6(c) of [33]. It took 0.01 seconds to compute the trajectory in MATLAB on an Intel Core i7-3520M 2.90 GHz processor. This time includes the computation of ϕ^ϵ which will be used in the following subsection. Trajectories, with initial conditions that are displayed from the slow manifold by distances of $\pm 10^{-10}$ along the stable and unstable manifolds, are displayed using thinner lines at the tip of this base trajectory. These were obtained from direct integration using MATLAB function `ode15s` with tolerances set to tol $= 10^{-10}$.

Computation of transients. Next, trajectories that connect to the trajectory $z = z(\tau)$ in Figure 4(b) near its starting point,

$$(q_1, q_2)(0) = (-0.51723351869, -0.73434299772),$$
$$(v_1, v_2)(0) = (-0.27894449516, 1.71095643157),$$

and leave it near its end point,

$$(q_1, q_2)(T) = (-0.39340933174, 0.00310289762),$$
$$(v_1, v_2)(T) = (-0.15410414452, 0.72034762953),$$

were computed using the SO-SMST method described in section 3. An example is shown in Figure 5(a) as a projection onto the (v_1, v_2, q_1)-space. The trajectory was obtained using the SO-SMST method with $\Delta t = \Delta\tau = 0.01$. The value of v_2 is fixed at $\tau = 0$ to 1.71095643157, while v_1 is fixed to be -0.025410414452 at $\tau = T$. This gives a distance of $r \approx 0.1$ from the slow manifold at both ends. In Figure 5(b) this is compared with an accurate reference solution obtained using the SMST algorithm by plotting the Euclidean norm of the difference

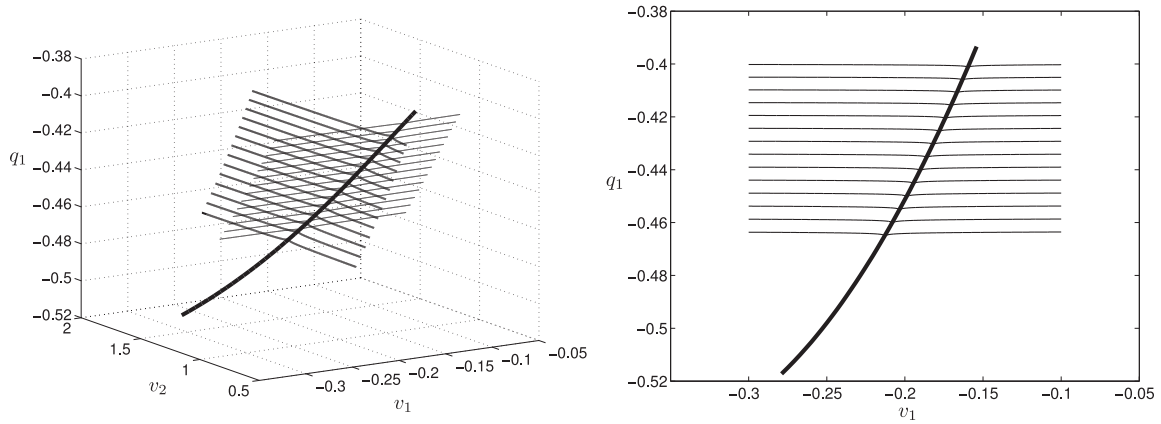


Figure 4. *The thick lines in* (a) *and* (b) *represent two projections of a trajectory on the slow manifold for the model of reciprocal inhibition. These trajectories were obtained using the modified Runge–Kutta scheme with the* SO *method based on Corollary* 4.2. *Trajectories, with initial conditions that are displaced from the slow manifold by distances of* $\pm 10^{-10}$ *along the stable and unstable manifolds, are displayed using thinner lines at the tip of this base trajectory. The thicker of the two sets of thinner lines, going in the* v_2*-direction, corresponds to trajectories on the stable manifold. The set of lines going in the* v_1*-direction corresponds to trajectories on the unstable manifold. These are obtained by direct backwards and forward integration, respectively.*

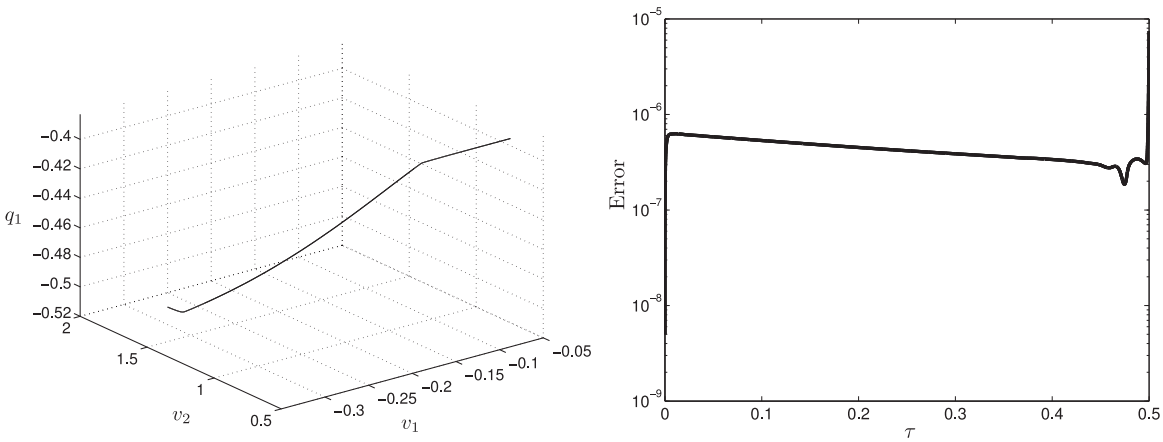


Figure 5. (a) *A trajectory computed using the* SO-SMST *principle in section* 3. *The fast connections are clearly visible.* (b) *The accuracy of the trajectory in* (a) *is analyzed by computing an accurate reference solution using the* SMST *algorithm.*

of the two solutions as a function of time. There is a good agreement between the two trajectories, the maximal error being $\approx 7.5 \times 10^{-6}$ at $\tau = T$. This value is also consistent with Proposition A.1: here $\epsilon = 10^{-3}$ and $r = 0.1$, so $\epsilon r^2 = 10^{-5}$. The computation of the approximation using the principle in section 3, which is visualized using the projections in

Figure 5(a), took

(5.5) 1.6 seconds.

The 1.6 seconds include the time required for the propagation of the base trajectory and the time for the collocation on the fast space. Figure 6 displays $v_1 = v_1(\tau)$ in (a) and $v_2 = v_2(\tau)$ in (b). Here the fast transients are clearly visible. Finally, if the resulting time mesh from the SO-SMST method, t-fine at the ends, τ-fine in-between, is used in the SMST collocation method, then one obtains an accuracy of 3.0×10^{-8}, but it took about twice as long (3.1 seconds to be precise). If one continues in this way for smaller values of ϵ while fixing $r = 0.1$, computing trajectories using the SO-SMST method, and then using the resulting time mesh in the SMST collocation method, the time used for the collocation method was still about twice as long. But more importantly, the SMST method did not converge for smaller values of ϵ than $\epsilon = 5 \times 10^{-6}$. The two methods used the same MATLAB collocation code. In contrast to the considerations in [36], the distance r has been fixed from the slow manifold while decreasing ϵ. It would be interesting to perform a more detailed comparison of the two methods in future research.

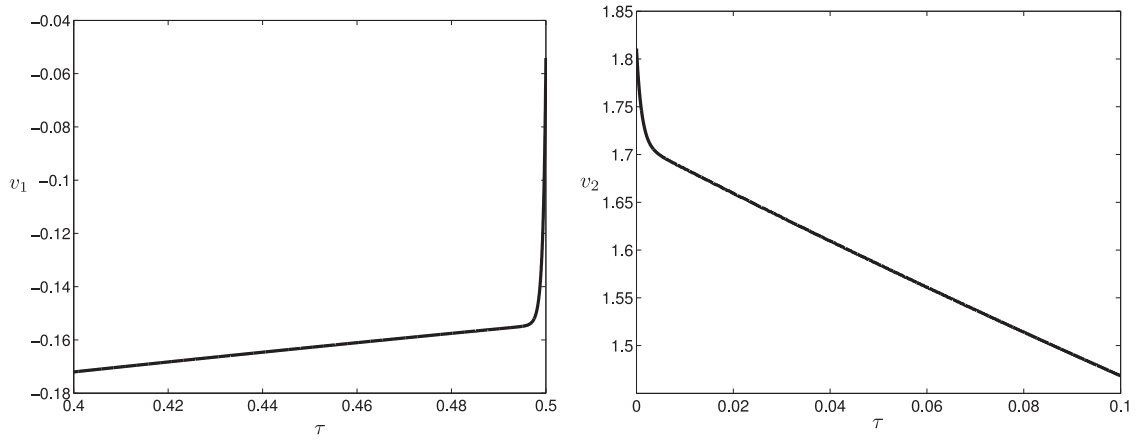
Figure 7(a) shows a comparison of solutions obtained using the SO-SMST method with accurate solutions obtained using the SMST algorithm for three different values of $r = 0.1$, 0.5, and 1. Only the last part of each trajectory is visualized using a projection onto the (q_1, v_1)-plane. The thick lines are the SO-SMST solutions while the thinner ones are those obtained using the SMST method. The error increases with increasing r. In Figure 7(b) the square of r in (A.1) is verified by computing the slope ≈ 2 of the maximal error as a function of r on a logarithm scale. Here the maximal error is understood as the maximum over $\tau \in [0, T]$ of the Euclidean distances between the trajectories.

By applying the SO-SMST procedure, the computation of trajectories near a saddle-type slow manifold, has been split into two nonstiff subproblems and as such the singular nature of the original problem has been removed. Therefore, no numerical issues appear when ϵ becomes extremely small. On the contrary, the solution becomes more accurate. Figure 8 shows the result of computing trajectories similar to the ones above for extremely small values of ϵ. The distance to the slow manifold has been fixed to $r = 0.1$ in both ends. In Figure 8(a) the exit trajectories are shown in the (v_1, q_1)-plane, while Figure 8(b) shows the time required for different values of ϵ. Notice the ≈ 1.6 seconds from (5.5) for $\epsilon = 10^{-3}$, and furthermore that all the time is taken up by the collocation on the fast space (compare $\diamond$'s with $\circ$'s). The time (≈ 0.01 seconds) required for the computation of the base trajectory ($\diamond$'s in Figure 8) is not visible on this scale.

5.4. Numerical results for the FitzHugh–Nagumo model. The FitzHugh–Nagumo model is a PDE model for the membrane potential of a nerve axon which is derived as a simplification of the Hodgkin–Huxley model,

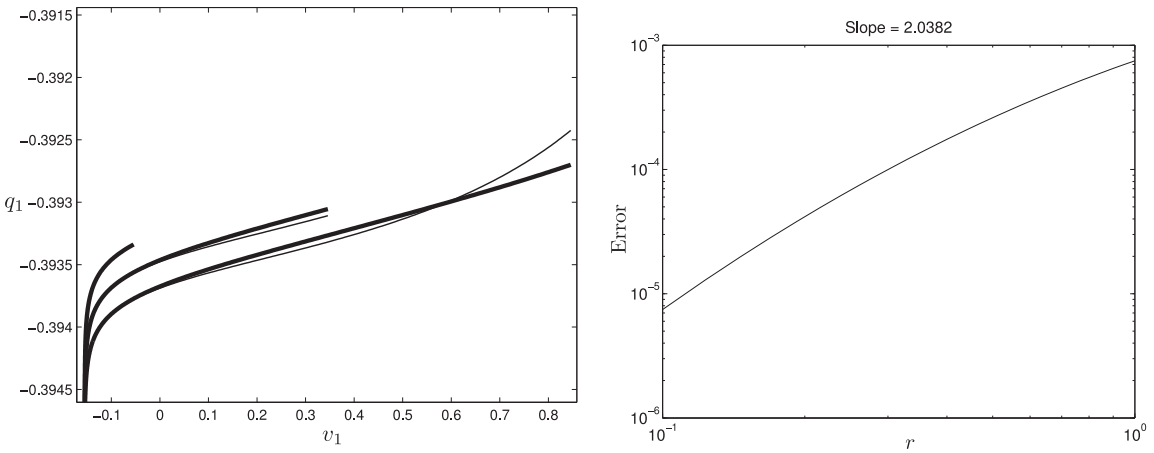
$$\partial_t u = \epsilon(v - \gamma u), \quad \partial_t v = d\partial_s^2 v + f_a(v) - u + p, \quad s \in \mathbb{R}^3,$$

with $f_a(u) = u(u-a)(1-u)$ and parameters p, γ, d, and a. When looking for traveling wave solutions of the form $u(t,s) = x(s+ct)$, $v(t,s) = y_1(s+ct)$, $y_2 = y_1'$, one obtains the following



(a) (b)

Figure 6. (a) $v_1 = v_1(\tau)$ *and* (b) $v_2 = v_2(\tau)$ *for the trajectory shown in Figure* 10(a). *The* v_2*-direction is the stable direction, whereas the* v_1*-direction is the unstable direction. The fast connection to the slow manifold is clearly visible in these diagrams. The variable* v_2 *decreases quickly initially, whereas* v_1 *grows fast near the end* $\tau = 0.5$.



(a) (b)

Figure 7. (a) *Transients for three different values of* $r = 0.1$, 0.5, *and* 1. *The thick lines are obtained using the* SO-SMST *principle described in section* 3, *while the thinner lines are due to the* SMST *algorithm.* (b) *The maximal error as a function of* r. *The slope on the logarithm scale is* ≈ 2.0.

finite dimensional slow-fast system:

$$\begin{aligned} \dot{x} &= \epsilon(y_1 - \gamma x), \\ \dot{y}_1 &= y_2, \\ \dot{y}_2 &= \frac{1}{d}(cy_2 - f_a(y_1) + x - p). \end{aligned}$$

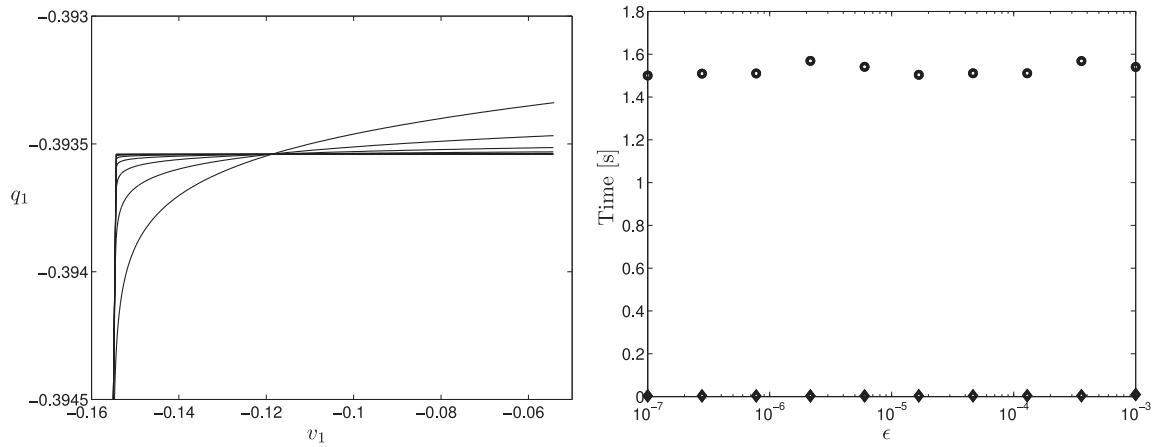


Figure 8. *The results of computing canards and their transients by the* SO-SMST *method for extremely small values of* ϵ. *Figure* (a) *shows the different exit trajectories projected onto the* (v_1, q_1)*-plane. In* (b) *the time used by the* SO-SMST *method is illustrated for different values of* ϵ. *The circles* ($\circ$) *give the total time, while the diamonds* ($\diamond$) *give the time required to compute the base trajectory.*

Here c is the wave speed. Geometric singular perturbation theory has been successfully used to analyze this system; see, e.g., [38, 36, 42] and references therein. In particular, the exchange lemma has been applied to prove the existence of homoclinic orbits including both fast and slow segments. Homoclinic orbits correspond, by the traveling wave ansatz, to traveling pulse solutions of the PDEs. Such trajectories will be computed in this section using the SO-SMST method. In this section it will be illustrated how the SO-SMST method can be combined with direct integration for computation of a full orbit. As in [35], attention is restricted to $a = 1/10$, $d = 5$, and $f \equiv f_{1/10}$ for simplicity.

To explain an example of a homoclinic orbit, it is first pointed out that the critical manifold is 1D and of the form

$$M_0 = \{y_2 = 0,\, x = f(y_1) + p\} = M_0^l \cup \{z_{lm}\} \cup M_0^m \cup \{z_{mr}\} \cup M_0^r.$$

It has three different normal hyperbolic components M_0^l, M_0^m, and M_0^r that are separated by two fold points $z_{lm} \approx (-0.0024 + p, 0.049, 0)$ and $z_{mr} \approx (0.13 + p, 0.68, 0)$. These objects are all contained within the plane $y_2 = 0$. An example for $p = 0$ is shown in Figure 9(a). Both M_0^l and M_0^r are of saddle type, whereas M_0^m is repelling. For ϵ sufficiently small, Fenichel's theory implies that $M_0^l \backslash B_\rho(z_{lm})$, $M_0^m \backslash \{B_\rho(z_{lm}) \cup B_\rho(z_{mr})\}$ and $M_0^r \backslash B_\rho(z_{mr})$ all perturb to some M^l, M^m, and M^r. Small neighborhoods $B_\rho(z_{lm})$ and $B_\rho(z_{rm})$ of the fold points z_{lm} and z_{mr}, respectively, have been removed from M_0^l and M_+^r since normal hyperbolicity is violated there.

For $p = 0$ the point $q = (0, 0, 0)$ is the unique equilibrium, and the results of, e.g., [38, 42], show that for ϵ sufficiently small there exists c_*, z_r^{approach}, z_r^{exit}, and z_l^{approach} so that for $c = c_*$ there is a homoclinic connection to q composed of four segments as follows:

(i) a fast segment along the strong unstable manifold of q connecting to M^r close to $z_r^{\text{approach}} \in M^r$.

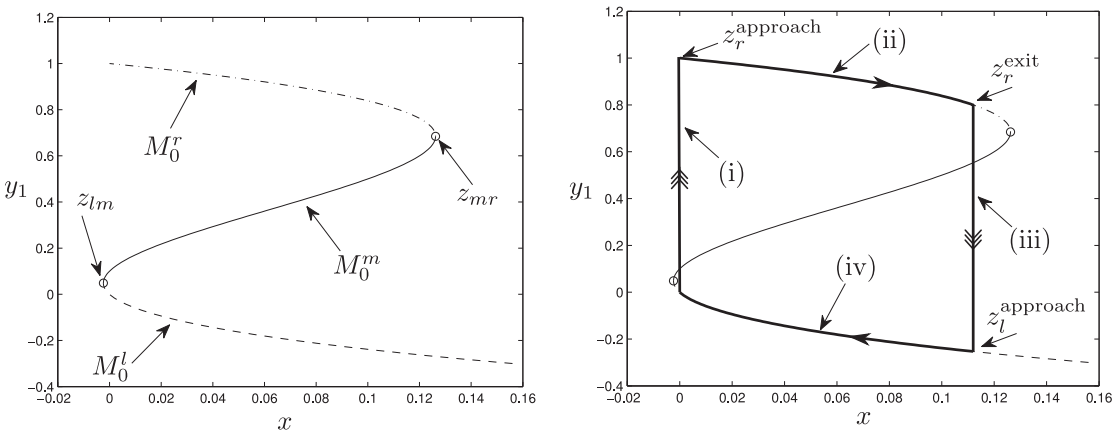


Figure 9. (a) *The critical manifold* $M_0 \subset \{(x, y_1, y_2)|y_2 = 0\}$ *and its three hyperbolic components* M_0^l, M_0^m, *and* M_0^r *for* $p = 0$, *within the plane* (x, y_1). *The points* z_{lm} *and* z_{rm} *are fold points.* (b) *The singular homoclinic orbit is contained within the* (x, y_1)*-plane and composed of segments* (i), (ii), (iii), *and* (iv). *A true homoclinic connection can be established for* $\epsilon > 0$ *but small by transversality from this singular homoclinic orbit.*

(ii) a slow segment on M^r initiated near z_r^{approach} and terminated near $z_r^{\text{exit}} \in M^r$.
(iii) a fast segment leaving M^r near z_r^{exit} and approaching M^l near $z_l^{\text{approach}} \in M^l$.
(iv) a slow segment on M^l initiated near z_l^{approach} and eventually terminating at $q = (0, 0, 0)$.

This orbit is obtained by transversality (using the exchange lemma and Fenichel's theory) from a singular orbit, whose projection onto the (x, y_1)-plane is shown in Figure 9(b).

We consider $p = 0$ and compute the homoclinic connection to the equilibrium at $(0, 0, 0)$ as follows:

1° Determination of c and the strong unstable manifold of the equilibrium $(0, 0, 0)$: As in [36] we use the fact that the stable manifold $W^s(M^r)$ acts as a separatrix in phase space. The resulting trajectory is terminated at $y_2 = 0$. This fast segment is denoted by γ_1.
2° Computation of M^r and the connection of γ_1 to M^r: Using the function ϕ^ϵ, the end point of γ_1 is projected onto M^r using (3.8) neglecting terms of order $\mathcal{O}(\epsilon^2 y_0^2)$, with y_0 measuring the deviation from M^r. The modified Runge–Kutta scheme is then used to compute $M^r = \{y = \eta_r(x)\}$. The connection from the end of γ_1 to M^r is computed using the SO-SMST algorithm. This slow segment is denoted by γ_2.
3° Computation of M^l: It is obtained as a graph $y = \eta_l(x)$ by using the modified Runge–Kutta scheme in backwards integration of $q = (0, 0, 0)$.
4° Computation of $W^u(M^r) \cap W^s(M^l)$: For this the Newton's method is used to obtain a root of the function

$$F(x_b^r, x_b^l) = (x^r, y_2^r)(x_b^r) - (x^l, y_2^l)(x_b^l),$$

with (x^r, y_2^r) being the intersection of a trajectory on $W^u(M^r)$, obtained by forward

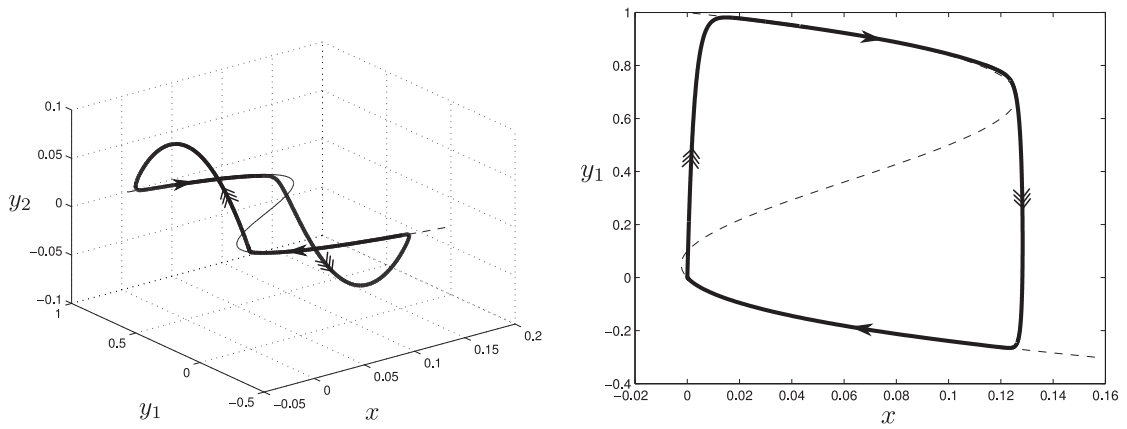


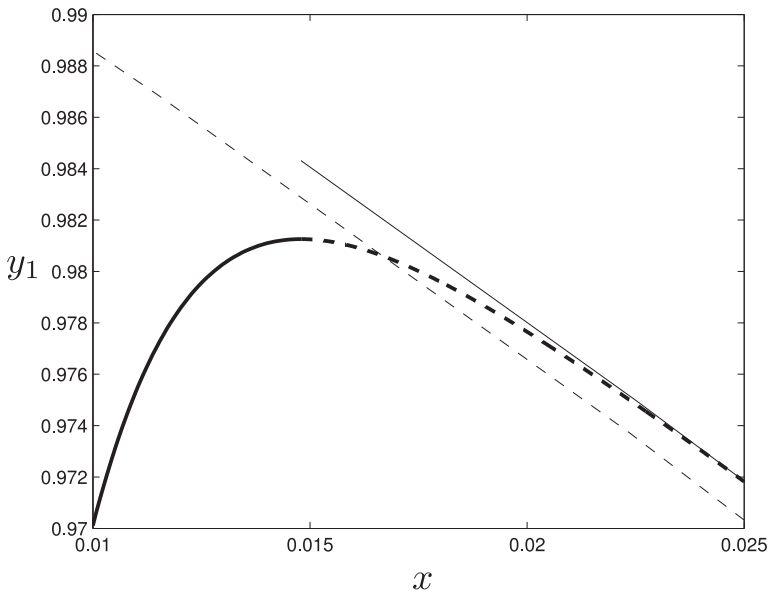
Figure 10. *The homoclinic orbit for $p = 0$, $c = 1.2462875$, and $\epsilon = 10^{-3}$ computed using the method described by $1°, 2°, \ldots, 5°$. Figure* (b) *shows the projection of the homoclinic onto the (x, y_1)-plane. The thin dotted lines indicate the critical manifold.*

integration, that was initiated at a point that was displayed from $(x_b^r, \eta_r(x_b^r))$ on M^r by an amount of 10^{-6} along the unstable direction, with the plane $y_1 = 1/2$. Similarly, (x^l, y_2^l) is the intersection of a trajectory on $W^s(M^l)$, obtained by backwards integration, that was initiated at a point that was displayed from $(x_b^l, \eta_l(x_b^l))$ on M^l by an amount of 10^{-6} along the stable direction, with the plane $y_1 = 1/2$. The Jacobian is computed through the variational equations. The derivatives $\partial_x \eta_r$ and $\partial_x \eta_l$ are obtained from the SO method. The resulting trajectory segment is denoted by γ_3. It connects γ_2 from the point of departure $(x_b^r, \eta(x_b^r))$ with M^l through the entrance $(x_b^l, \eta_l(x_b^l))$.

5° The final slow segment γ_4 is taken from M^l from the entrance point $(x_b^l, \eta(x_b^l))$ to $(0, 0, 0)$.

The union of the segments $\gamma_1, \gamma_2, \gamma_3$, and γ_4 forms a homoclinic orbit. The result is shown in Figure 10. From here it is also clear that the homoclinic has segments near the end of segment (iv) and near the end of segment (ii) that are relatively close to the fold points z_{lm} and z_{mr}. We obtain $c = 1.2462875$ for $p = 0$. The result of step 2° for $\epsilon = 10^{-3}$ is shown in Figure 11 using a close-up. There is an error in the connection with γ_1 to γ_2 due to (3.8) that is not visible in this diagram. It is too small: 5×10^{-9}.

A simpler alternative to the projection method used here that is based on the determination of the function ϕ^ϵ would be to use the "naive" fiber projection: $(x, y) \mapsto (x, \eta(x))$. See also (3.5). In general this projection is $\mathcal{O}(\epsilon r)$-close to the correct one. The number r again measures the deviation from the slow manifold. If this naive projection is applied here, then one obtains a slightly larger error of 2×10^{-6} in the connection. There is an improvement by factor of 10^{-3} using the more accurate SOF projection without any detectable increase in time.



(a)

Figure 11. *The result of step* 2° *for* $\epsilon = 10^{-3}$ *and* $p = 0$. *The full thick line shows the incoming trajectory* γ_1 *from step* 1°. *The thin line shows the base trajectory on* M^r *obtained by projection through* ϕ. *The thick dotted line is* γ_2, *the result of applying the* SO-SMST *algorithm to connect* γ_1 *to* M^r. *There is a discrepancy at the connection of* γ_1 *with* γ_2 *due to the errors associated with the projection described in* (3.8). *However, it is small and not visible, being only* 5×10^{-9}.

5.5. The Lindemann mechanism: An example not in the canonical slow-fast form. In this section we finally consider the Lindemann mechanism,

$$\begin{aligned} \dot{x} &= X^\epsilon(x,y) = -x(x-y), \\ \dot{y} &= Y(x,y) = x(x-y) - \epsilon y, \end{aligned} \tag{5.6}$$

also considered in [31, 64]. Here x, $y \geq 0$. It is an example of a slow-fast system where the slow and fast variables have not been properly identified, and it is used as a caricature of an ϵ-free system. Setting $(w,z) = (x+y, 2y)$ gives a system in the canonical slow-fast form:

$$\begin{aligned} \dot{w} &= \epsilon W(w,z) = -\frac{1}{2}\epsilon z, \\ \dot{z} &= Z(w,z) = 2w^2 - (3w+\epsilon)z + z^2. \end{aligned} \tag{5.7}$$

The graph $z = w$, which corresponds to $y = x$ in the original variables, is then a normally attracting critical manifold. Using the original variables in (5.6) it is easy to realize the existence of a unique equilibrium at $(x,y) = 0$. This equilibrium is nonhyperbolic even for $\epsilon > 0$: the eigenvalues are 0 and $-\epsilon$. In [14] it is nevertheless shown that the origin attracts all of the first quadrant x, $y \geq 0$ for all $\epsilon > 0$.

In [64] it was shown that the two iterative methods, SO and SOF, are both successful in

approximating the slow manifold and the tangent spaces of the fibers. What proves crucial to this are the following:

(a) SO and SOF make no explicit reference to ϵ. These methods only involve the vector-fields X^ϵ and Y and their first partial derivatives.

(b) The variable x can still parametrize the critical manifold. In contrast to (1.1), where normally hyperbolicity always implies that the critical manifold can be written as a graph over the slow variables, this does not need to hold true if the slow and fast variables have not been properly identified.

It was demonstrated in [64] that SO and SOF performed better than the alternative CSP method when applied to (5.6). In particular, Figure 10 in [64] shows that n applications of the CSP method and the SOF method give approximations of the tangent spaces of the fibers accurate to orders $\mathcal{O}(\epsilon^{n-1})$ and $\mathcal{O}(\epsilon^{n+1})$, respectively.

The system (5.6) is not an example with a saddle-type slow manifold. Nevertheless, the SO-SMST method will still be applied in order to demonstrate its use on ϵ-free systems. Figure 12 shows a comparison of accurate closed-form solutions for $\eta = \eta(x)$ and $\phi^\epsilon = \phi^\epsilon(x)$ obtained using Maple with solutions η^h and $\phi^{\epsilon,h}$ obtained using the discretized iterative methods in Corollary 4.2 and Proposition 4.3. The comparison was made for $x = 1$, grid size $h = \epsilon$, and varying values of ϵ. The operator δ_x^h was again based on classical 2nd order finite differences. The errors are seen to give approximately straight lines in the log-log scale. The slopes were ≈ 5 and ≈ 4 for the approximations of η and ϕ^ϵ, respectively. Since $p = 2$ one would expect from $(4.7)_{h=\epsilon}$ a slope of ≈ 4 for the determination of η. The improved slope of ≈ 5 is due to the fact that the $\mathcal{O}(\epsilon)$-term in the asymptotic expansion for η,

$$\eta(x) = x - \frac{1}{2}\epsilon + \mathcal{O}(\epsilon^2),$$

is constant. The error $(\partial_x - \delta_x^h)(\eta^h - \eta_0)$ is therefore $\mathcal{O}(\epsilon^2)$, and in this case the error in $(4.7)_{h=\epsilon}$ with $p = 2$ should be $\mathcal{O}(\epsilon^3 h^2)$, ignoring the exponentially small terms. Moreover, the order of ≈ 4 for the determination of ϕ^ϵ is not in agreement with $(4.11)_{h=\epsilon}$ since

$$\phi^\epsilon = -\frac{1}{2} + \mathcal{O}(\epsilon) \tag{5.8}$$

is not small. See also Remark 5. However, since the zeroth order term in the expansion of ϕ^ϵ in (5.8) is constant, the error from replacing ∂_x by δ_x^h is therefore $(\partial_x - \delta_x^h)\phi^{\epsilon,h} = \mathcal{O}(\epsilon)$, and the total error is therefore $\mathcal{O}(\epsilon^2 h^2)$, again ignoring the exponentially small terms, which is in agreement with Figure 12(b).

Figure 13(a) shows a trajectory computed using the modified Runge–Kutta scheme (thick line) for $\Delta\tau = 0.01$, $\epsilon = 0.1$, and $h = 10^{-3}$. The thinner lines show the result of accurate backwards integration of initial conditions that were displayed by an amount of $\pm 10^{-4}$, $\pm 10^{-5}$, $\ldots$, $\pm 10^{-9}$ from the slow manifold along the stable direction. Of all the pairs, only for the one with $\pm 10^{-9}$ do the trajectories jump in the same direction. The slow manifold is therefore expected to be correct up to $\pm 10^{-8}$ but not more accurate than $\pm 10^{-9}$. Figure 13(b) shows a connection (full thick line) to the base trajectory in Figure 13(a) (dotted line in Figure 13(b)) obtained using the SO-SMST method with $\Delta t = \Delta\tau = 0.01$. Using $\diamond$'s, this solution is compared with a solution obtained by direct forward integration. There is a good agreement

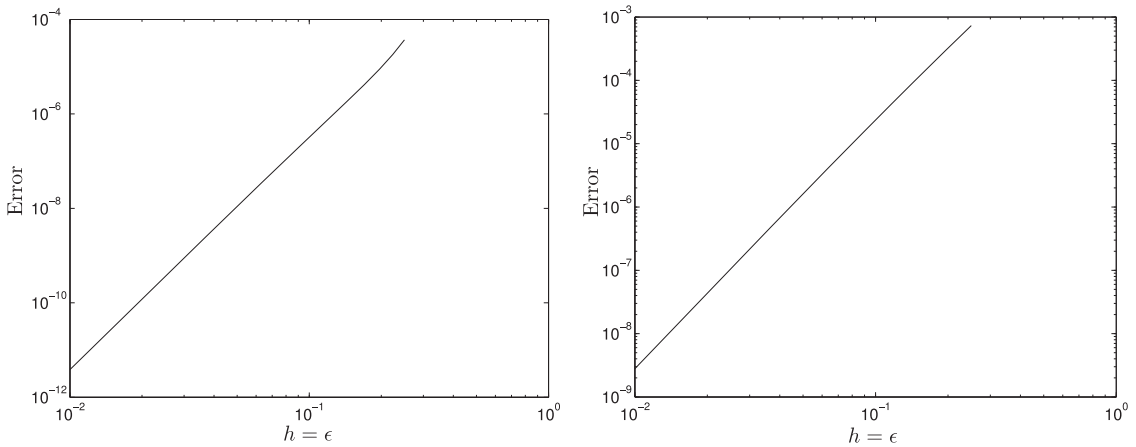


Figure 12. *The errors* $\|\eta^h - \eta\|$ (a) *and* $\|\phi^{\epsilon,h} - \phi^\epsilon\|$ (b) *for the Lindemann mechanism* (5.6) *at* $x = 1$ *for* $h = \epsilon$ *and as a function of* ϵ. *The slopes are* ≈ 5 *and* ≈ 4. *There is an improvement with respect to the estimates in* $(4.7)_{h=\epsilon}$ *and* $(4.11)_{h=\epsilon}$ *for this example. This is due to the fact that for this example the finite difference operator* δ_x^h *resolves the derivatives of the first terms in the asymptotic expansions of* η *and* ϕ^ϵ *exactly.*

between the two solutions. Figure 13(c) shows the maximal error between accurate reference solutions, obtained using accurate forward integration, and trajectories computed using the SO-SMST method as a function of the distance r from the slow manifold. The slope of the straight line in the logarithmic scale is ≈ 2 in agreement with (A.1) and (A.2). The "naive" projection described in (3.5) assumes that the fast fibers are vertical. Applying this principle to this example will therefore lead to an $\mathcal{O}(1)$-error with no improvement for $\epsilon \to 0$.

Remark 7. The system (5.6) also exemplifies the importance of the property described in Remark 2. To explain this, first note that the critical manifold $y = x$ of (5.6) (or $z = w$ of (5.7)) is nonhyperbolic at $x = 0$. This manifests itself in the fact that if one approximates the slow manifold using, e.g., asymptotic expansions, then the accuracy of the approximation will deteriorate for $x \to 0^+$. But since $(x, y) = 0$ is actually an equilibrium of the system, and the SO approximation always includes equilibria of the system, the SO approximation for the slow manifold of (5.6) goes through $(x, y) = 0$, and the error of the approximation therefore improves near $x = 0$. The SOF approximation is also well-defined up to $(x, y) = 0$. See [64, section 8].

6. Conclusion.

This paper has presented an alternative method for the computation of trajectories on saddle-type slow manifolds using iterative methods to approximate the slow manifold and its fiber projections. This included a numerical implementation of a modified SO method (also known as the iterative method of Fraser and Roussel) in a classical Runge–Kutta quadrature scheme for the computation of these unstable trajectories on the slow manifold. This part applies to other types of slow manifolds, even normally elliptic ones. For the computation of transients the SOF method was augmented to this quadrature scheme, and a basic principle of splitting the problem into two nonstiff subproblems was outlined and demonstrated in several examples, including a model of reciprocal inhibition and

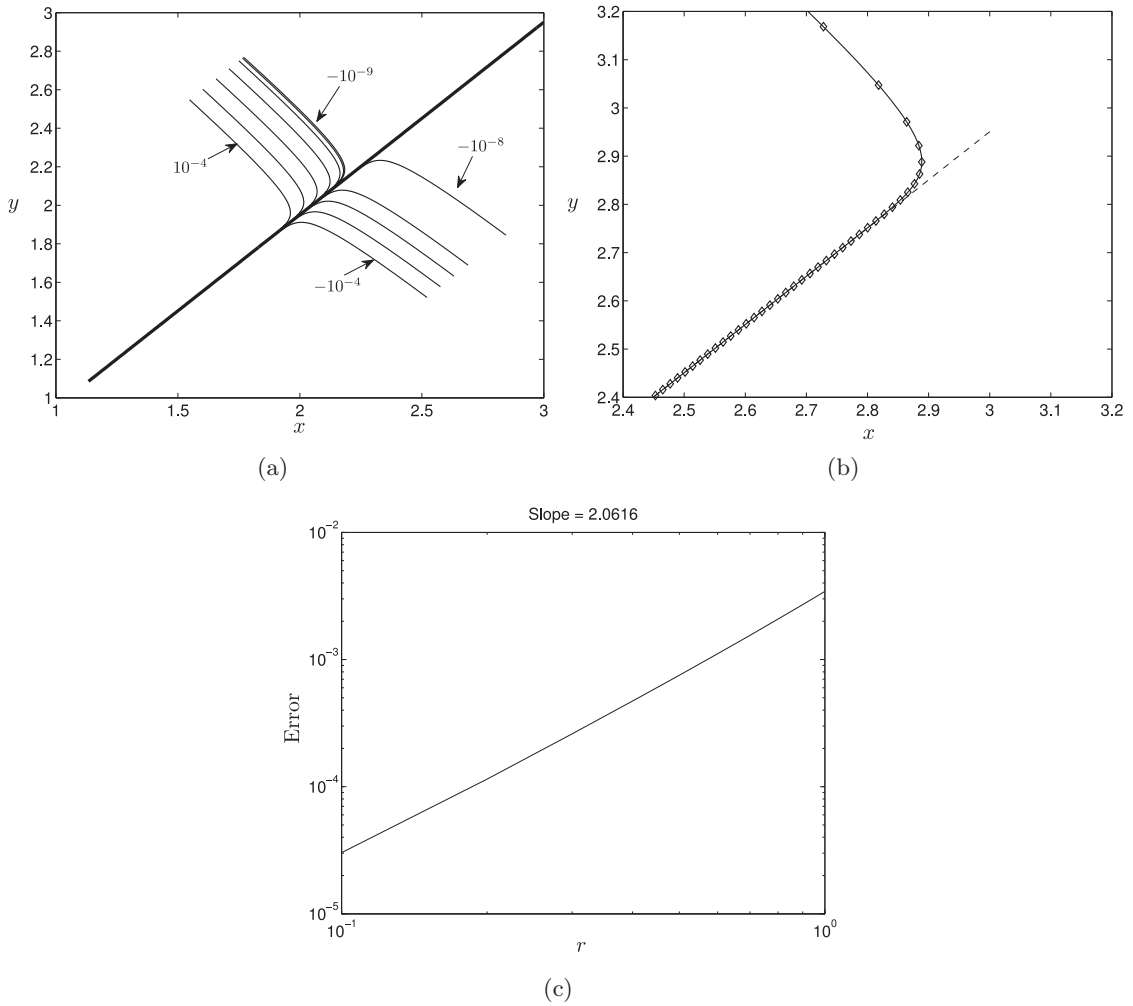


Figure 13. *Figure* (a) *shows the result of accurate backwards integration of* (5.6) *with* $\epsilon = 0.1$ *for pairs of initial conditions displayed by an amount* $\pm 10^{-4}, \pm 10^{-5}, \ldots, \pm 10^{-9}$ *from the slow manifold, computed using Corollary* 4.2*, along the unstable direction. Of all the pairs, only for the one with* $\pm 10^{-9}$ *do the trajectories jump in the same direction. The slow manifold is therefore expected to be correct up to* $\pm 10^{-8}$*. Figure* (b) *shows the result of applying the* SO-SMST *method (thick line) for the computation of a trajectory asymptotic to the slow manifold. The dotted line shows the slow manifold, and the* $\diamond$*'s show the result of applying accurate direct forward integration. There is a good agreement with the* $\diamond$*'s and the thick line. Figure* (c) *shows the result of comparing direct integration with the result of applying the* SO-SMST *method for different initial distances* r *from the slow manifold. In agreement with* (A.1) *and* (A.2) *the slope is* ≈ 2.

the FitzHugh–Nagumo model. This principle, which was named SO-SMST, benefits from the fact that the singular nature of the problem has been removed. On the other hand, the SO-SMST method is disadvantaged by the fact that its accuracy is determined by ϵ alone.

Future research should further explore the use of the proposed method in applications. A promising area is believed to be ϵ-free systems. In a "real-life" slow-fast system one will typically not expect there to be an explicit small parameter (such as the Olsen model [57])

and it may be very difficult (if not impossible) to write the system in the canonical form (1.1); see, e.g., [11, 10, 12, 40]. The method presented here applies to such systems, as demonstrated in section 5.5, and hence it could potentially provide a useful tool for numerical exploration of such systems.

Appendix A. Error estimates for SO-SMST. In this appendix, the error introduced by replacing (3.2),

$$\begin{aligned}\dot{x}_0 &= \Lambda^\epsilon(x_0) + \mathcal{O}(\epsilon y_0^2),\\ \dot{y}_0 &= A(x_0)y_0 + \mathcal{O}(y_0^2),\end{aligned}$$

with (3.3),

$$\dot{x}_0 = \Lambda^\epsilon(x_0),$$

in the SO-SMST is quantified. The set $y_0 = 0$ is here a saddle-type slow manifold.

Proposition A.1. *Suppose that $y_0 = y_0(t)$ decays exponentially fast to the slow manifold $y_0 = 0$ in one end and escapes it exponentially fast at the other end. That is, assume that there exists a positive constant λ so that*

$$\|y_0(t)\| \leq r\left(\exp(-\lambda t) + \exp(-\lambda(T/\epsilon - t))\right)$$

for $t \in [0, T/\epsilon]$, where $r = \max\{\|y_0(0)\|, \|y_0(T/\epsilon)\|\}$. Then the error $\Delta x_0 = \Delta x_0(\tau)$, taking $\Delta x_0(0) = 0$, from replacing (3.2) by (3.3) is

$$\mathcal{O}(\lambda^{-1}\epsilon r^2) \quad \text{for all } \tau \in [0, T]. \tag{A.1}$$

Proof. Given that $\Delta x_0(0) = 0$ and (3.2), Δx_0 satisfies

$$\begin{aligned}\Delta x_0(\tau) &\leq L\int_0^\tau \Delta x_0(s)ds + Cr^2\int_0^\tau \left(\exp(-2\lambda\epsilon^{-1}s) + \exp(-2\lambda\epsilon^{-1}(T-s))\right)ds\\ &\leq L\int_0^\tau \Delta x_0(s)ds + C\lambda^{-1}\epsilon r^2,\end{aligned}$$

with $L = \sup_x \|\partial_x \Lambda\|$ and some $C > 0$, for ϵ and r sufficiently small. Applying Gronwall's inequality in integral form [5] then gives

$$\Delta x_0(\tau) \leq C\lambda^{-1}\epsilon r^2 \exp(LT) \quad \text{for all} \quad \tau \in [0, T],$$

from which the result follows. ∎

The error Δx_0 from replacing (3.2) by (3.3) gives rise to an error Δy_0 in y_0. This error is described in the following proposition.

Proposition A.2. *Let $\tilde{y}_0$ be the solution of (3.6) obtained using (3.4) and (3.7), and set $\Delta y_0 = \tilde{y}_0 - y_0$. Then there exist constants C_1 and C_2 independent of ϵ and r so that*

$$\|\Delta y_0(t)\| \leq \left(\|\Delta y_0(0)\| + C_1\epsilon r^2 t_0\right)\exp(C_2 t_0) \tag{A.2}$$

for all $t \in [0, t_0]$.

Proof. It directly follows that

$$\|\Delta y_0(t)\| \leq \|\Delta y_0(0)\| + \int_0^t \int_0^1 \|\partial_x Y(x + s\Delta x_0, y_0 + s\Delta y_0)\| \|\Delta x_0\| ds dt$$
$$+ \int_0^t \int_0^1 \|\partial_y Y(x + s\Delta x_0, y_0 + s\Delta y_0)\| \|\Delta y_0\| ds dt.$$

Now use (A.1) and Gronwall's inequality in integral form to obtain (A.2). ∎

Remark 8. For the trajectories computed in [35], where the stable and unstable components are taken from the critical manifold, setting $(1.9)_{\epsilon=0}$, respectively, $(1.10)_{\epsilon=0}$, to 0, this gives $r = \mathcal{O}(\epsilon)$ and errors in (A.1) and (A.2) (supposing in the latter case that $\Delta y_0(0) = 0$) of order $\mathcal{O}(\epsilon^3)$. This is the proof of the statement in Remark 4.

Acknowledgments. I would like to thank M. Brøns and S. J. Hogan for helpful discussions and providing valuable feedback in the preparation of this document. I also thank an anonymous referee for suggestions leading to an improved manuscript.

REFERENCES

[1] G. L. Alfimov, V. M. Eleonsky, and L. M. Lerman, *Solitary wave solutions of nonlocal sine-Gordan equations*, Chaos, 8 (1998), pp. 257–271.
[2] C. J. Amick and K. Kirchgässer, *A theory of solitary water-waves in the presence of surface tension*, Arch. Rational Mech. Anal., 105 (1989), pp. 1–49.
[3] V. I. Arnold, E. Khukhro, V. V. Kozlov, and A. I Neishtadt, *Mathematical Aspects of Classical and Celestial Mechanics* [*Dynamical Systems.* III], 3rd ed., translated from the Russian original by E. Khukhro, Encyclopaedia Math. Sci. 3, Springer-Verlag, Berlin, 2006.
[4] U. M. Ascher, R. M. M. Mattheij, and R. D. Russell, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations*, Classics Appl. Math. 13, SIAM, Philadelphia, 1987.
[5] R. Bellmann, *The stability of solutions of linear differential equations*, Duke Math. J., 10 (1943), pp. 643–647.
[6] E. Benoît, *Chasse au canard.* II. *Tunnels–entonnoirs–peignes.* [*Duck hunt.* II. *Tunnels–funnels–combs*], Collect. Math., 32 (1981), pp. 77–97.
[7] J.-L. Callot, *Chasse au canard.* III. *Les canards ont la vie brève.* [*Duck hunt.* III. *Ducks have a short life*], Collect. Math., 32 (1981), pp. 99–114.
[8] E. Benoît and J.-L. Callot, *Chasse au canard.* IV. *Annexe numérique.* [*Duck hunt.* IV. *Numerical appendix*], Collect. Math., 32 (1981), pp. 115–119.
[9] P. Braza and T. Erneux, *Singular Hopf bifurcation to unstable periodic solutions in an NMR laser*, Phys. Rev. A, 40 (1989), pp. 2539–2542.
[10] M. Brøns, *Canard explosion of limit cycles in templator models of self-replication mechanisms*, J. Chem. Phys., 134 (2011), 144105.
[11] M. Brøns, *An Iterative Method for the Canard Explosion in General Planar Systems*, preprint, arXiv:1209.1109v1 [math.DS], 2012.
[12] M. Brøns and K. U. Kristiansen, *On the Approximation of the Canard Explosion Point in Epsilon-free Systems*, preprint, arXiv:1504.07752v1 [math.DS], 2015. Available online at http://adsabs.harvard.edu/cgi-bin/bib_query?arXiv:1504.07752.
[13] J. Butcher, *Numerical Methods for Ordinary Differential Equations*, John Wiley & Sons, Chichester, UK, 2003.
[14] M. S. Calder and D. Siegel, *Properties of the Lindemann mechanism in phase space*, Electron. J. Qual. Theory Diff. Equ., 2011 (2011), 8.
[15] J. Carr, *Applications of Centre Manifold Theory*, Appl. Math. Sci. 35, Springer-Verlag, New York, 1981.
[16] M. Desroches, J. Guckenheimer, B. Krauskopf, C. Kuehn, H. M. Osinga, and M. Wechselberger, *Mixed-mode oscillations with multiple time scales*, SIAM Rev., 54 (2012), pp. 211–288.

[17] M. Desroches, B. Krauskopf, and H. M. Osinga, *Numerical continuation of canard orbits in slow-fast dynamical systems*, Nonlinearity, 23 (2010), pp. 739–765.
[18] F. Diener and M. Diener, *Chasse au canard.* I. *Les canards.* [*Duck hunt.* I. *The ducks*], Collect. Math., 32 (1981), pp. 37–74.
[19] E. J. Doedel, R. C. Paffenroth, A. R. Champneys, T. F. Fairgrieve, Yu. A. Kuznetsov, B. E. Oldemann, B. Sandstede, and X. Wang, *AUTO2000: Continuation and Bifurcation Software for Ordinary Differential Equations (with HomCont)*, Division of Applied Mathematics, Brown University, Providence, RI, 2002. Available online at http://www.dam.brown.edu/people/sandsted/publications/auto2000.pdf
[20] M. Domijan, R. Murray, and J. Sneyd, *Dynamical probing of the mechanism underlying calcium oscillations*, J. Nonlinear Sci., 16 (2006), pp. 438–506.
[21] J. L. Dubbeldam and B. Krauskopf, *Self-pulsations in laser with saturable absorber: Dynamics and bifurcations*, Opt. Comm., 159 (1999), pp. 325–338.
[22] J. P. England, B. Krauskopf, and H. M. Osinga, *Computing one-dimensional stable manifolds and stable sets of planar maps without the inverse*, SIAM J. Appl. Dyn. Syst., 3 (2004), pp. 161–190.
[23] T. Erneux, *Q-switching bifurcation in a laser with a saturable absorber*, J. Opt. Soc. Am. B, 5 (1988), pp. 1063–1069.
[24] T. Erneux and P. Mandel, *Bifurcation phenomena in a laser with saturable absorber.* I, Z. Phys. B, 44 (1981), pp. 353–363.
[25] T. Erneux and P. Mandel, *Bifurcation phenomena in a laser with saturable absorber.* II, Z. Phy. B, 44 (1981), pp. 365–374.
[26] N. Fenichel, *Persistence and smoothness of invariant manifolds for flows*, Indiana Univ. Math. J., 21 (1971), pp. 193–226.
[27] N. Fenichel, *Asymptotic stability with rate conditions*, Indiana Univ. Math. J., 23 (1974), pp. 1109–1137.
[28] C. W. Gear, T. J. Kaper, I. G. Kevrekidis, and A. Zagaris, *Projecting to a slow manifold: Singularly perturbed systems and legacy codes*, SIAM J. Appl. Dyn. Syst., 4 (2005), pp. 711–732.
[29] V. Gelfreich and L. Lerman, *Almost invariant elliptic manifold in a singularly perturbed Hamiltonian system*, Nonlinearity, 15 (2002), pp. 447–557.
[30] V. Gelfreich and L. Lerman, *Long-periodic orbits and invariant tori in a singularly perturbed Hamiltonian system*, Phys. D, 176 (2003), pp. pp 125–146.
[31] D. A. Goussis and M. Valorani, *An efficient iterative algorithm for the approximation of the fast and slow dynamics of stiff systems*, J. Comput. Phys., 214 (2006), pp. 316–346.
[32] J. Guckenheimer and R. Haiduc, *Canards at folded nodes*, Moscow Math. J., 5 (2005), pp. 91–103.
[33] J. Guckenheimer, K. Hoffman, and W. Weckesser, *Numerical computation of canards*, Int. J. Bifur. Chaos Appl. Sci. Engrg., 4 (2000), pp. 84–97.
[34] J. Guckenheimer, K. Hoffman, and W. Weckesser, *The forced van der Pol equation* I*: The slow flow and its bifurcations*, SIAM J. Appl. Dyn. Syst., 2 (2003), pp. 1–35.
[35] J. Guckenheimer and C. Kuehn, *Computing slow manifolds of saddle type*, SIAM J. Appl. Dyn. Syst., 8 (2009), pp. 854–879.
[36] J. Guckenheimer and C. Kuehn, *Homoclinic orbits of the FitzHugh-Nagumo equation: The singular limit*, Discrete Contin. Dyn. Syst. Ser. S, 2 (2009), pp. 851–872.
[37] J. Guckenheimer and C. Kuehn, *Homoclinic orbits of the FitzHugh–Nagumo equation: Bifurcations in the full system*, SIAM J. Appl. Dyn. Syst., 9 (2010), pp. 138–153.
[38] C. K. R. T. Jones, *Geometric Singular Perturbation Theory, Lecture Notes in Mathematics, Dynamical Systems (Montecatini Terme)*, Springer, Berlin, 1995.
[39] H. G. Kaper and T. J. Kaper, *Asymptotic analysis of two reduction methods for systems of chemical reactions*, Phys. D, 165 (2002), pp. 66–93.
[40] I. G. Kevrekidis, C. W. Gear, J. M. Hyman, P. G. Kevrekidis, O. Runborg, and C. Theodoropoulos, *Equation-free, coarse-grained multiscale computation: Enabling microscopic simulators to perform system-level analysis*, Comm. Math. Sci., 1 (2003), pp. 715–762.
[41] N. Kopteva and E. O'Riordan, *Shishkin meshes in the numerical solution of singularly perturbed differential equations*, Int. J. Numer. Anal. Model., 7 (2010), pp. 393–415.
[42] M. Krupa, B. Sandstede, and P. Szmolyan, *Fast and slow waves in the FitzHugh-Nagumo equations*, J. Differential Equations, 133 (1997), pp. 49–97.

[43] S. H. Lam, *Using CSP to understand complex chemical kinetics*, Combust. Sci. Tech., 89 (1993), pp. 375–404.
[44] S. H. Lam and D. A. Goussis, *Understanding complex chemical kinetics with computational singular perturbation*, in Proceedings of the 22nd International Symposium on Combustion, Seattle, WA, 1988, pp. 931–941.
[45] P. Langfield, B. Krauskopf, and H. M. Osinga, *Solving Winfree's puzzle: The isochrons in the FitzHugh-Nagumo model*, Chaos, 24 (2014), 013131.
[46] J. Laskar, *Large scale chaos in the solar system*, Astronom. and Astrophys., 287 (1994), pp. 9–12.
[47] J. Laskar and M. Gastineau, *Existence of collisional trajectories of Mercury, Mars and Venus with the Earth*, Nature, 459 (2009), pp. 817–819.
[48] E. N. Lorenz, *The slow manifold—what is it?*, J. Atmospheric Sci., 49 (1992), pp. 2449–2451.
[49] E. N. Lorenz, *Existence of a slow manifold*, J. Atmospheric Sci., 43 (1986), pp. 1547–1557.
[50] E. N. Lorenz and V. Krishnamurty, *On the non-existence of a slow manifold*, J. Atmospheric Sci., 44 (1987), pp. 2940–2950.
[51] U. Maas and S. B. Pope, *Simplifying chemical kinetics: Intrinsic low-dimensional manifolds in composition space*, Combust. Flame, 88 (1992), pp. 239–264.
[52] R. S. MacKay, *Slow manifolds*, in Energy Localisation and Transfer, T. Dauxois, A Litvak-Hinenzon, R. S. MacKay, and A. Spanoudaki, eds., World Scientific, River Edge, NJ, 2004, pp. 149–192.
[53] D. A. McQuarrie, *Physical Chemistry: A Molecular Approach*, University Science Books, Sausalito, 1997.
[54] L. Michaelis and M. Menten, *Die Kinetik der Invertinwirkung*, Biochem. Z., 49 (1913), pp. 333–369.
[55] A. Neishtadt, *Persistence of stability loss for dynamical bifurcation,* I, Differential Equations, 23 (1987), pp. 1385–1390.
[56] K. Nipp, *Numerical integration of stiff ODEs of singular perturbation type*, Z. Angew. Math. Phys., 42 (1991), pp. 53–79.
[57] L. F. Olsen, *An enzyme reaction with a strange attractor*, Phys. Lett. A, 94 (1983), pp. 454–457.
[58] P. F. Rowat and A. I. Selverston, *Modeling the gastric mill central pattern generator of the lobster with a relaxation-oscillator network*, J. Neurophys., 70 (1993), pp. 1030–1053.
[59] J. Rubin and D. Terman, *Geometric singular perturbation analysis of neuronal dynamics*, in Handbook of Dynamical Systems, vol. 2, B. Fieldler, ed., North-Holland, Amsterdam, 2002, pp. 93–146.
[60] J. Rubin and M. Wechselberger, *Giant squid - hidden canard: The 3D geometry of the Hodgkin–Huxley model*, Biol. Cybernet., 97 (2007), pp. 5–32.
[61] F. K. Skinner, N. Kopell, and E. Marder, *Mechanisms for oscillation and frequency control in reciprocally inhibitory model neural networks*, J. Comput. Neurosci., 1 (1994), pp. 69–87.
[62] R. Temam, *Inertial Manifolds*, Math. Intell., 12 (1990), pp. 68–74.
[63] A. N. Tihonov, *Systems of differential equations containing small parameters in the derivatives*, Mat. Sb. N.S., 31 (1952), pp. 575–586 (in Russian).
[64] K. U. Kristiansen, M. Brøns, and J. Starke, *An iterative method for the approximation of fibers in slow-fast systems*, SIAM J. Appl. Dyn. Syst., 13 (2014), pp. 861–900.
[65] K. U. Kristiansen, P. Palmer, and M. Roberts, *A unification of models of tethered satellites*, SIAM J. Appl. Dyn. Syst., 10 (2011), pp. 1042–1069.
[66] K. U. Kristiansen, P. Palmer, and R. M. Roberts, *The persistence of a slow manifold with bifurcation*, SIAM J. Appl. Dyn. Syst., 11 (2012), pp. 661–683.
[67] K. U. Kristiansen and C. Wulff, *Exponential Estimates of Slow Manifolds*, preprint, arXiv:1208.4219v1 [math.DS], 2012.
[68] J. Vanneste, *Asymptotics of a slow manifold*, SIAM J. Appl. Dyn. Syst., 7 (2008), pp. 1163–1190.
[69] X. J. Wang and J. Rinzel, *Alternating and synchronous rhythms in reciprocally inhibitory model neurons*, Neural Comput., 4 (1992), pp. 84–97.
[70] M. Wechselberger, *Existence and bifurcation of canards in $\mathbb{R}^3$ in the case of a folded node*, SIAM J. Appl. Dyn. Syst., 4 (2005), pp. 101–139.
[71] A. Zagaris, C. W. Gear, T. J. Kaper, and I. G. Kevrekidis, *Analysis of the accuracy and convergence of equation-free projection to a slow manifold*, ESAIM Math. Model. Numer. Anal., 43 (2009), pp. 757–784.
[72] A. Zagaris, H. G. Kaper, and T. J. Kaper, *Fast and slow dynamics for the computational singular perturbation method*, Multiscale Model. Simul., 2 (2004), pp. 613–638.