Periodic Orbit Continuation in Multiple Time Scale Systems

John Guckenheimer and M. Drew LaMar

Mathematics Department, Cornell University, USA.

Continuation methods utilizing boundary value solvers are an effective tool for computing unstable periodic orbits of dynamical systems. AUTO [1] is *the* standard implementation of these procedures. Unfortunately, the collocation methods used in AUTO often require very fine meshes for convergence on problems with multiple time scales. This inconvenience prompts the search for alternative methods for computing such periodic orbits; we introduce here new multiple-shooting algorithms based on geometric singular perturbation theory.

1 Mathematical setting

The systems that we study are *slow-fast* systems of the form

$$\begin{cases} \varepsilon \dot{x} = f(x, y), \\ \dot{y} = g(x, y), \end{cases} \tag{1}$$

where $f : \mathbb{R}^m \to \mathbb{R}^m$ and $g : \mathbb{R}^n \to \mathbb{R}^n$ are at least C^1 and $\varepsilon > 0$ is a small parameter determining the ratio of time scales between the fast variable $x \in \mathbb{R}^m$ and the slow variable $y \in \mathbb{R}^n$. The limit $\varepsilon = 0$ is a system of differential algebraic equations in which motion is constrained to the *critical manifold* C defined by $f = 0$. Rescaling time to the "slow time" $\tau = \varepsilon t$ yields the system

$$\begin{cases} x' = f(x, y), \\ y' = \varepsilon g(x, y). \end{cases} \tag{2}$$

Here, the limit $\varepsilon = 0$ is the family of *layer* equations in y, also called the fast subsystems of (1). We make two standing assumptions about (1) that further constrain the context for our analysis:

1. The critical manifold C of (1) is indeed a manifold and its projection Π onto the space of slow variables is generic in the sense of singularity theory;

2. The limit sets of all trajectories for the layer equations are equilibria, i.e., points of C.

At regular points of the projection Π, the manifold C can be represented locally as the graph of a function $x = h(y)$. Substitution of this expression into g yields the *slow flow* on the regular points of C. We shall denote the set of singular points of Π by S.

Trajectories of (1) are typically approximated by *candidates*, concatenations of trajectories of the slow flow and the layer equations that form continuous curves. Periodic orbits that contain both segments close to trajectories of the slow flow and segments close to trajectories of the layer equations are called *relaxation oscillations*. Trajectory segments close to an unstable sheet of the critical manifold are called *canards*. Numerical computation of canards by forward solution of an initial value problem is not feasible when ε is sufficiently small due to the instability on the fast time scale [11]. Thus, even stable periodic orbits containing canards cannot be computed by forward numerical integration from initial points in the basin of attraction of these orbits. As mentioned earlier, tracking such orbits with collocation methods also seems to require very fine meshes. Our goal in this paper is to re-examine the computation of relaxation oscillations, including those with canards. We propose a multiple-shooting approach, in which different segments of a periodic orbit are computed differently and then matched with suitably chosen cross-sections.

The next two sections lay out the general framework that we investigate. Section 4 presents two numerical examples, comparing the methods introduced here with AUTO computations of the same orbits. Finally, Sec. 5 comments on further extension and improvement of these methods.

2 Simple relaxation oscillations

We consider first the simplest relaxation oscillations, namely, those with a fast-slow decomposition that makes them readily amenable to analysis. We define a relaxation oscillation Γ^ε to be simple if it is approximated by a candidate Γ^0 that satisfies the following conditions:

S1: Γ^0 consists of slow segments α_i and fast segments β_i, $i = 1 \dots k$, in the order $\alpha_1, \beta_1, \dots, \alpha_k, \beta_k$. The initial and terminal points of α_i are p_i and q_i. The initial and terminal points of β_i are q_i and $p_{(i+1) \bmod k}$.

S2: Each α_i lies on a stable sheet of the critical manifold.

S3: The points q_i are saddle-node bifurcations of the layer equations and none of their eigenvalues have positive real parts. This assumption implies that there are unique solutions of the layer equations emanating from the points q_i and all nearby fold points of C.

S4: The slow flow satisfies the normal crossing conditions [20] at q_i.

S5: Denote by $\omega(S)$ the forward limits of trajectories of the layer equations emanating from the fold points of S satisfying condition **S3**. We require that $\omega(S)$ is transverse to the slow flow on C at the points p_i.

Note that we have not required that a simple relaxation oscillation be stable or even hyperbolic, although eigenvalues of a return map in the fast directions are stable and, indeed, approach zero as $\varepsilon \to 0$.

We want to establish that well-conditioned multiple-shooting methods can be formulated for the computation of simple relaxation oscillations. Our strategy is to create cross-sections Σ_i to each of the fast segments β_i of a simple relaxation oscillation Γ^ε, compute the flow maps Φ_i^ε from Σ_i to $\Sigma_{(i+1) \bmod k}$ and then solve the multiple-shooting equations

$$z_{(i+1) \bmod k} = \Phi_i^\varepsilon(z_i)$$

for points $z_i \in \Sigma_i$.

Theorem 1. *Let Γ^ε be a hyperbolic simple relaxation oscillation, and Σ_i and Φ_i^ε as described above. For $\varepsilon \geq 0$ sufficiently small, the system of equations*

$$z_{(i+1) \bmod k} = \Phi_i^\varepsilon(z_i), \qquad z_i \in \Sigma_i,$$

is regular and has an isolated solution with $z_i = \Sigma_i \cap \Gamma^\varepsilon$.

Proof. Fenichel theory [6] and a theorem of Levinson [17] imply that the flow maps Φ_i^ε from Σ_i to $\Sigma_{(i+1) \bmod k}$ are smooth maps that converge to smooth maps Φ_i^0 of rank $n-1$ as $\varepsilon \to 0$. We remark that the convergence is continuous but not smooth in ε due to several phenomena; for example, asymptotic expansions of trajectories near the folds involve fractional powers of ε [25]. The point $\Phi_i^0(z_i)$ is obtained by a three-step process:

(1) follow the trajectory of the layer equations with initial condition (z_i) to its limit on the critical manifold;
(2) follow this limit point to its first intersection with a fold of the critical manifold; and
(3) follow the unstable separatrix of the layer equations from this fold point to its intersection with $\Sigma_{(i+1) \bmod k}$.

Denote the image of $\Phi^0_{(i-1) \bmod k}$ by W_i. The dimension of W_i is $n-1$, the dimension of folds of the critical manifold. Condition **S5** implies that the restriction of Φ_i^0 to W_i has full rank $n-1$. The restriction of the equations $z_{(i+1) \bmod k} = \Phi_i^\varepsilon(z_i)$ to W_i has the same structure as the set of equations for a multiple-shooting method based upon cross-sections to the flow. The Jacobian of this system has the block-cyclic structure

$$\begin{pmatrix} -D\Phi_1^0|_{W_1} & I & & & & \\ & -D\Phi_2^0|_{W_2} & I & & & \\ & & \ddots & \ddots & & \\ & & & -D\Phi^0_{k-1}|_{W_{k-1}} & I & \\ & & & & -D\Phi_k^0|_{W_k} & \\ I & & & & & \end{pmatrix}.$$

As shown by Guckenheimer and Meloon [12], this matrix has maximal rank if and only if 1 is not an eigenvalue of $\mathrm{diag}(D\Phi_k^0|_{W_k}, \ldots, D\Phi_1^0|_{W_1})$, that is, Γ^0 is a hyperbolic fixed point of the composition $\Phi_k^0|_{W_k} \circ \cdots \circ \Phi_1^0|_{W_1}$. On a complementary set of coordinates to the W_i the equations $z_{(i+1) \bmod k} = \Phi_i^\varepsilon(z_i)$ reduce to $z_{(i+1) \bmod k} = 0$ because $\Phi_i^\varepsilon(z_i)$ vanishes in these directions by definition. Thus, the full system of equations on the product of the Σ_i is regular if and only if Γ^0 is hyperbolic. The equations change continuously in the C^1 topology as $\varepsilon \to 0$ [13], so hyperbolicity of Γ^0 implies that the equations are regular and that Γ^ε is hyperbolic for $\varepsilon > 0$ sufficiently small. This proves the theorem. □

For a hyperbolic simple relaxation oscillation a multiple-shooting algorithm based upon the cross-sections described above yields a regular system of equations. In many cases, these equations will be well conditioned uniformly for small ε. If they are not, additional cross-sections can be inserted. The effectiveness of the multiple shooting algorithm will be largely determined by the numerical integration method used to compute the Φ_i^ε.

3 Degenerate slow-fast decompositions

The multiple-shooting algorithm described above for locating simple relaxation oscillations can be implemented within a standard continuation framework. The procedure may break down when a family of periodic orbits encounters degenerate slow-fast decompositions resulting from the failure of one of the requirements for the orbit to be simple. Here, we examine modifications of the multiple-shooting algorithm that cope with the instability of canards in the context of two specific examples of degenerate decompositions [8].

3.1 Hopf bifurcation and canard explosions

The lowest-dimensional example of a degenerate slow-fast decomposition occurs at Hopf bifurcations of a systems with one slow and one fast direction ($n = m = 1$). The *canard explosion* of the resulting orbits has been studied extensively, especially in the system

$$\begin{cases} \varepsilon \dot{x} = y - \frac{1}{3}x^3 + x, \\ \dot{y} = a - x \end{cases} \tag{3}$$

near $a = 1$; see, for example, [10, **?**]. (Note that in several studies coordinates have been used that place the point $(1, -2/3)$ of (3) at the origin [5].) It has been proven that the periodic orbits of this system grow monotonically as a decreases from 1, "exploding" in size from $O(\varepsilon)$ to $O(1)$ over a range of a that is $O(\exp(-c/\varepsilon))$ for a suitable $c > 0$. The trajectories in the middle of this explosion contain canards that follow the unstable branch of the critical manifold given by $y = \frac{1}{3}x^3 - x$ for an $O(1)$ distance before jumping right or left to a

stable branch of the critical manifold. Trajectories along the canard segments of these trajectories diverge from one another at a rate $\exp(-t(x^2-1)/\varepsilon)$. For small values of ε, this divergence effectively prevents accurate computation of a trajectory for times that are larger than $O(\varepsilon)$. However, backward integration along these canards is highly stable.

To compute the periodic orbits with canards in this family, we adopt a shooting strategy that shoots forward and backward from one cross-section of the flow to another cross-section. The cross-sections are chosen so that forward trajectories do not contain canard segments and backward trajectories do not contain segments that track the stable part of the slow manifold. The initial cross-section depends upon where we are in the family, in particular on the direction of the jump away from the canard segment in a periodic orbit. Over part of the family, the jump is to the right and there is a single stable slow segment in the periodic orbit. Over another part of the family, the jump is to the left and there are two stable slow segments and two fast segments in the slow-fast decomposition of the trajectory. The behavior that occurs between these two possibilities is that there is a *maximal canard* that extends over the entire length of the unstable branch of the critical manifold. When the jumps from the canards are to the right, we choose the initial cross-section to be the line $\{x = a\}$ where the vector field is horizontal. For jumps to the left, we choose the initial cross-section $\{x = -1\}$, which is crossed by all trajectories that flow left from the unstable branch of the critical manifold to the stable branch of the critical manifold. In both cases, we take the final target cross-section to be $\{x = a\}$.

The shooting problem that we seek to solve is $\Phi_a^+(y) = \Phi_a^-(y)$ where $\Phi_a^+(y)$ is the flow map from the initial to the final cross-section in the forward time direction and $\Phi_a^-(y)$ is the flow map from the initial to the final cross-section in the backward time direction. There are three remarks that we make about this problem:

1. The derivatives of both $\Phi_a^+(y)$ and $\Phi_a^-(y)$ are small for members of the canard family, but the derivative of $\Phi_a^+(y)$ is much smaller [5, 9], so the periodic orbits are stable;
2. The derivative of $\Phi_a^+(y) - \Phi_a^-(y)$ with respect to a is $O(1)$, so the small derivative of $\Phi_a^+(y) - \Phi_a^-(y)$ with respect to y yields an extreme sensitivity of the solution y to the shooting problem as a function of the parameter a. Therefore, in the middle of the canard family, we fix the initial point on the first cross-section and vary a to locate a periodic orbit passing through the initial point instead of fixing a and trying to locate the solution y of the shooting equation. Alternatively, we regard the shooting equation as a continuation problem in the variables (y, a) and use pseudo-arclength or other continuation strategies to find the curve of solutions to the equation;
3. If we move beyond the range of parameter values for which there are canards, we are likely to find trajectories that do not reach the target cross-section. This happens when there is a stable equilibrium point for

parameters $a > 1$. When a is large enough so that the vector field has a stable node, trajectories that follow the right-hand branch of the critical manifold accumulate at the stable equilibrium point without reaching $x = a$. Implementations of the shooting algorithms need to test for this possibility and take appropriate action if the shooting equation is not defined.

3.2 Folded saddles

The forced Van der Pol system

$$\begin{cases} \varepsilon\dot{x} = y + x - \frac{x^3}{3}, \\ \dot{y} = -x + a\sin(2\pi\theta), \\ \dot{\theta} = \omega \end{cases} \tag{4}$$

is a slow-fast system with two slow variables and one fast variable ($n = 2$, $m = 1$). Cartwright and Littlewood studied this system in their seminal work on chaos in dynamical systems [2, 18, 19]. Recently, Haiduc [14] has extended the classical results of Cartwright and Littlewood by using geometric singular perturbation theory; these methods have also been used to investigate the dynamics of this system numerically [7, 11]. Throughout these studies, *folded saddles* play a prominent role in the analysis. Folded saddles are points where the rescaled slow flow equations

$$\begin{cases} \theta' = \omega(x^2 - 1), \\ x' = -x + a\sin(2\pi\theta) \end{cases} \tag{5}$$

have a saddle point. In the original system they are points on the fold curves where the normal crossing conditions fail. At the folded saddles the slow flow changes direction from pointing toward the fold to pointing away from the fold. Canards emanate from the folded saddles along stable manifolds of the saddles of (5) (lifted back to the unstable sheet of the critical manifold of (4)).

Bold et al. [7] used AUTO to track families of periodic orbits in the forced van der Pol system that contain canards emanating from folded saddles. These computations required fine meshes when applied to the system with $\varepsilon = 10^{-3}$ and even more so with $\varepsilon = 10^{-4}$. We develop here modifications similar to the ones described above for Hopf bifurcations to compute these trajectories with multiple-shooting methods that use a small number of cross-sections. We place cross-sections to the flow at the beginning of canard segments and in the middle of jumps that leave the canard segments. Backward integration between these cross-sections is stable since there is a single fast variable and the unstable sheet of the critical manifold is stable for the reversed time flow. At folded saddles of a system with two slow and one fast variables, the flow is parallel to the fold curve. For the forced Van der Pol system this suggests that we choose the cross-section defined by $\theta = \theta_{fs}$, where θ_{fs} is the value of θ at the folded saddle. All trajectories that jump from the unstable sheet of

the critical manifold to a stable sheet intersect one of the planes $\{x = \pm 1\}$, so we choose these planes as the cross-sections for the trajectories that jump from canards. Periodic orbits of different periods yield different sequences of intersections with the cross-sections and different defining equations. Unlike the canard explosions discussed in Sec. 3.1, many of the periodic orbits are embedded in chaotic invariant sets and are unstable.

4 Numerical results

We computed families of periodic orbits containing canards in the Van der Pol and forced Van der Pol systems with AUTO [1] and with our shooting methods. Our calculations follow the approach in [10] to address the complexity of periodic orbits in systems with multiple time scales. Thus, we use as many as 1000 mesh points in AUTO and error tolerances for both AUTO and shooting on the order of 10^{-10}.

For the multiple-shooting algorithms, we used PYCONT, a continuation package developed for PYDSTOOL [3]. The shooting algorithms were embedded in a Moore-Penrose continuation framework [**?**], with the differential equations numerically integrated using Radau [15], an implicit Runge-Kutta method. We set the absolute error tolerances in the calculations with Radau to 10^{-12} and the relative error tolerance to 10^{-9}. These tolerances produced trajectories with sufficient precision so that the truncation errors did not appear to be significant for the Newton iterations within the shooting method.

In the next two sections we show results for each of the two examples mentioned above. After presenting the results, we do some comparisons between AUTO and multiple-shooting with the forced Van der Pol system. Since algorithmic performance is highly dependent on implementation details, e.g., programming language used, we assess algorithmic complexity based on four properties of Newton's method: the size of Jacobians, condition numbers of Jacobians, the number of Newton iterations, and domains of convergence.

4.1 Hopf bifurcation and canard explosions

Figure 1 displays the results of our computations with multiple shooting of the canard explosion in the Van der Pol oscillator (3). Panel (a) shows the bifurcation diagram and panel (b) is a plot of representative limit cycles. Three regions are indicated on the bifurcation diagram in Fig. 1(a), where open circles denote the boundaries between them. Region A corresponds to simple relaxation oscillations, where we use the shooting methods described in Sec. 2. Regions B and C correspond to jump-left and jump-right canards, respectively; in these regions we use the shooting methods described in Sec. 3.1. The open circle that separates regions B and C corresponds to a maximal canard. Finally, the solid black circle denotes the Hopf point.

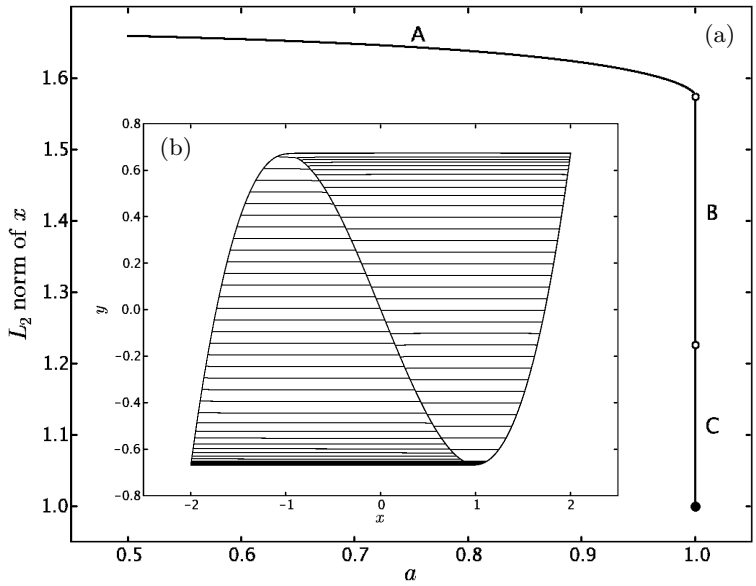

Fig. 1. The bifurcation diagram of the Van der Pol oscillator (3) near the canard explosion (a) and the corresponding limit cycles with canard segments (b) as computed with our shooting method.

In order to start the continuation of canards in the Van der Pol system with the multiple-shooting algorithm, we proceed as follows. We first find a simple stable relaxation oscillation numerically for the parameter values $(a, \varepsilon) = (0.5, 1.0)$ using Radau integration. We then continue this orbit in AUTO with ε as the free parameter. The continuation terminates at the periodic orbit with $\varepsilon = 10^{-4}$. Starting from this periodic orbit, we subsequently continue the family of orbits in AUTO with a as the free parameter. This continuation terminates at a limit cycle in the middle of region B. This limit cycle is our initial limit cycle for continuation with our shooting methods. We perform continuation in the forward and backward direction with a as the free parameter. When moving in the backward direction, we pass from region B to region A, where canards cease and simple relaxation oscillations exist. At the transition from B to A, backward integration fails, indicating that we should change shooting methods. During the continuation in the forward direction we pass from region B to region C, by encountering a maximal canard. In this situation backward integration also fails, and we must switch cross-sections from $\{x = -1\}$ to $\{x = 1\}$. Note that the tangent vector to the bifurcation curve switches: in region B both a and y are increasing, while in region C a is increasing and y is decreasing. Automated detection of periodic orbits that separate these regions (open circles) is an important topic for future work.

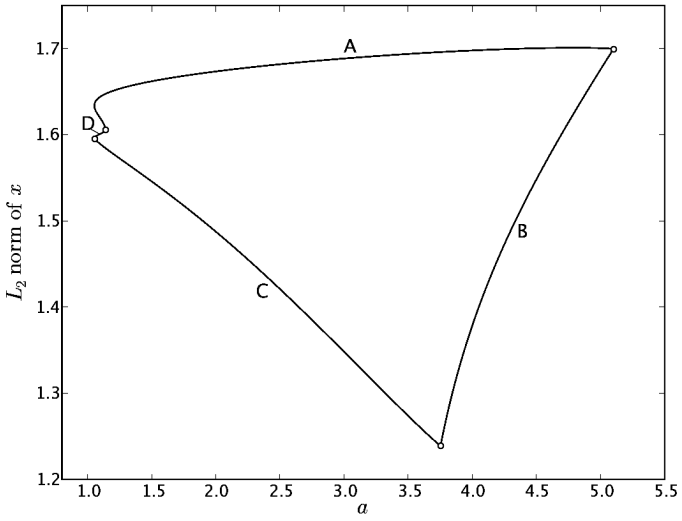

Fig. 2. Bifurcation diagram of period-three orbits in the forced Van der Pol system (4) computed with shooting. In this example, $\varepsilon = 10^{-4}$ and $\omega = 1.55$.

4.2 Folded saddles

For the continuation of periodic orbits with canards initiated at folded saddles in the forced Van der Pol system, we start with a period-three simple stable relaxation oscillation for the parameter values $(\varepsilon, \omega, a) = (10^{-4}, 1.55, 2.5)$. A bifurcation diagram, calculated with shooting and a as the free parameter, is shown in Fig. 2. During the computation of the periodic orbits we take advantage of the symmetry $(\theta, x, y) \mapsto (\theta + 1.5, -x, -y)$, which means that we need only integrate over half the period. We apply the symmetry transformation to the endpoint at the section $\{\theta = \theta_{fs} + 1.5\}$, and compare this point to the endpoint at $\{\theta = \theta_{fs}\}$, giving our matching condition. Our initial cross-sections are $\{x = \pm 1\}$, and thus the matching conditions for shooting occur in a hyperplane with coordinates (y, θ, a). The sign of x on the initial cross-section depends on the location of the periodic orbit in the bifurcation diagram. There are four regions. Region A corresponds to no canards, where we start from the cross-section $\{x = 1\}$ and integrate forward to the cross-section $\{x = -1^+\}$ (the superscript denotes crossing in the increasing x-direction). Regions B and D correspond to jump-right canards (right relative to the fast variable x), where we start from the cross-section $\{x = 1\}$, shooting forward to the cross-section $\{\theta = \theta_{fs} + 1.5\}$ and backward to the cross-section $\{\theta = \theta_{fs}\}$. Finally, region C corresponds to jump-left canards, where we start from the cross-section $\{x = -1\}$, shooting forward to the cross-section $\{\theta = \theta_{fs} + 1.5\}$ and backward to the cross-section $\{\theta = \theta_{fs}\}$. Transitions between regions are again denoted by open circles.

4.3 Comparisons

Let us now discuss the convergence of the Newton iterations for each method. In AUTO, a collocation boundary value method is used to find periodic orbits. With this method, the Jacobian is of dimension $mnN+b+q+1$, where N is the number of subintervals, m the number of collocation points per subinterval, n the dimension of the vector field, b the number of boundary conditions and q the number of integral conditions. We have already mentioned that, due to the slow-fast structure of the systems, the number of subintervals must be large for accuracy and convergence (we use $N = 1000$). The number of collocation points used per subinterval is set at $m = 4$ in our calculations. Thus, we expect the size of the Jacobian to be approximately $8,000 \times 8,000$ in the Van der Pol system and approximately $12,000 \times 12,000$ in the forced Van der Pol system. In the multiple-shooting methods, the Jacobian is of dimension $(n-1)s+1$, where again n denotes the dimension of the vector field and s is the number of cross-sections that are used. In the forced Van der Pol system the Jacobian is 3×3. Note that we will require more cross-sections when tracking periodic orbits with multiple canard segments.

Computationally, most of the effort in the shooting methods is in the integration, while in collocation most of the computational effort is in solving a large, sparse matrix. The construction of the Jacobian in the multiple-shooting method involves numerical integration of the vector field, which is an efficient and speedy process. The Jacobian for collocation is much larger because it is also used to determine a suitable orbit segment (while in shooting this is done with an integrator whose accuracy must be controlled separately). It should be noted that, although the Jacobian is considerably larger with AUTO, efficient numerical techniques are used to invert the Jacobian in two stages by taking advantage of the sparsity structure of the matrix. The first stage uses a method known as condensation of parameters to perform independent eliminations in N blocks of size $nm \times n(m+1)$. The second stage produces a *reduced* Jacobian of size $(n+b+q+1) \times (n+b+q+1)$; see also Chapter 1. This inversion still requires much more computation than Gaussian elimination on the matrix of size $((n-1)s+1) \times ((n-1)s+1)$ used in our shooting method.

For both AUTO and shooting, we performed a full Newton's method with a maximum of eight iterations and error tolerances on the order of 10^{-10}. During the calculations, we kept track of the number of iterations in the convergence of each step of Newton's method, as well as the condition number of the Jacobian. In the forced Van der Pol system the Jacobians for the shooting methods had $O(10)$ condition numbers, while for AUTO the condition numbers for the reduced Jacobians were $O(10^6)$.

Domains of convergence for the family $\mathcal{F}_\Gamma$ of limit cycles Γ along the branch B of Fig. 2 are displayed in Fig. 3; panel (a) shows the domain of convergence for the multiple-shooting method and panel (b) that for an AUTO computation (with $N = 1000$ mesh intervals). The thick (approximately) horizontal black line through the origin represents the a-dependent family of limit

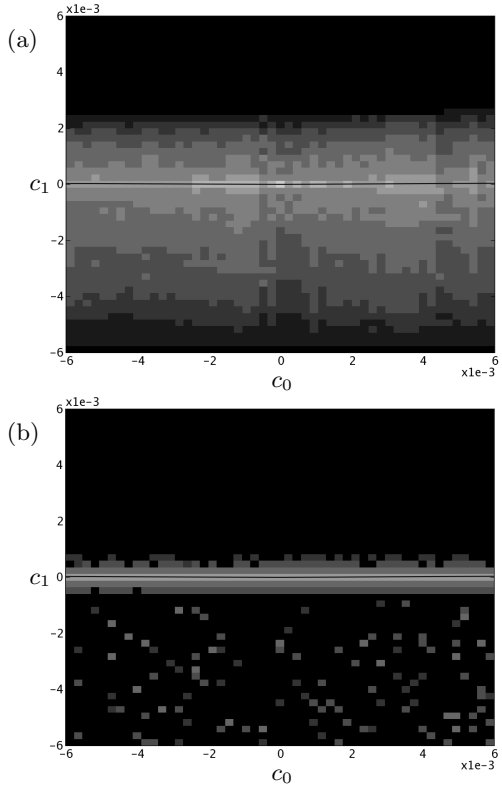

Fig. 3. Domains of convergence for the computation of limit cycles Γ along branch B of Fig. 2 with the shooting method (a) and with AUTO for $N = 1000$ mesh intervals.

cycles Γ. The figures are shaded according to the number of iterates needed for convergence, where shades white to black represent convergence after one to eight iterations, respectively. In fact, black denotes no convergence of the method using our error tolerances and choice of maximally eight iterates. Figure 2(a) contains and additional darkest shade of gray that is used for those points that do not converge in eight iterations but show signs of converging. Specifically, such points are marked as converging if the function values and differences between variable values for the last three iterates are decreasing.

The plots were obtained by starting with a specific limit cycle Γ_0 and its intersection γ_0 with the cross-section Σ defined by $x = 1$. The section Σ is three dimensional with coordinates (y, θ, a). We compute orthonormal vectors ($\boldsymbol{v}_0$ and $\boldsymbol{v}_1$) in Σ at γ_0 so that $\boldsymbol{v}_0$ is tangent to $\mathcal{F}_\Gamma \cap \Sigma$ and $\boldsymbol{v}_1$ lies in the plane spanned by $\boldsymbol{v}_0$ and $(0, 0, 1)$. The coordinates $\boldsymbol{c} = (c_0, c_1)$ in the figure correspond to the points $p_c = \gamma_0 + c_0\boldsymbol{v}_0 + c_1\boldsymbol{v}_1 \in \Sigma$ and, thus, the origin represents γ_0. The horizontal black line in the figure is the projection of $\mathcal{F}_\Gamma \cap \Sigma$ onto the plane spanned by $(\boldsymbol{v}_0, \boldsymbol{v}_1)$. Note that $\mathcal{F}_\Gamma \cap \Sigma$ appears

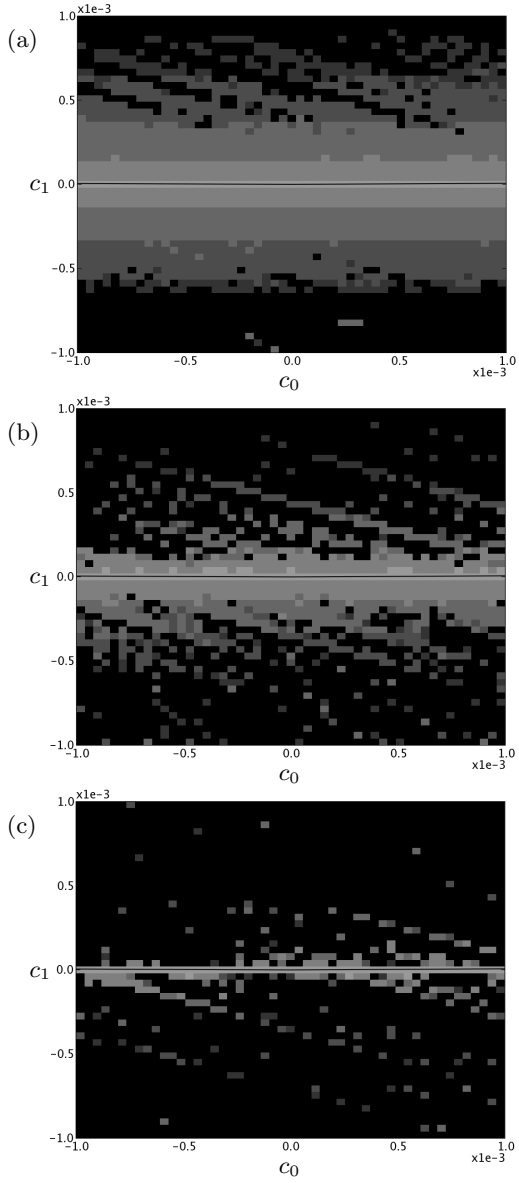

Fig. 4. The domains of convergence for the computation of limit cycles Γ along branch B of Fig. 2 with AUTO depend on the number of mesh intervals; panel (a) to (c) are for $N = 1000$, $N = 500$ and $N = 250$ mesh intervals, respectively.

horizontal because $\boldsymbol{v}_0$ is tangent to $\mathcal{F}_\Gamma \cap \Sigma$ and the domain represented in the figure is small.

For the shooting algorithm, we use the point p_c for each grid point $\boldsymbol{c} = (c_0, c_1)$ in the figure as initial point for the algorithm and shade the pixel according to the number of iterates required for convergence as described above. Unfortunately, initialization of AUTO requires an entire curve. To obtain a curve from each of the points p_c, we computed trajectories forward and backward as in the shooting algorithm and used the concatenation of these two trajectory segments to initialize AUTO. (AUTO computes a mesh of specified size from the trajectories provided.) Except on $\mathcal{F}_\Gamma$, these curves are not closed: the final points of the forward and backward trajectory segments do not match. Since AUTO is based upon solving the differential equations within the space of closed curves, it might be preferable to initialize AUTO with closed curves. Lacking a natural way to produce closed curves at different distances from $\mathcal{F}_\Gamma$, we did not pursue such a comparison here.

We also tested the dependence of AUTO's domain of convergence on the number of mesh intervals N. Figure 4 displays the results of three computations. Panel (a) shows an enlargement of Fig. 3(b) (with $N = 1000$). Panels (b) and (c) are for computations with $N = 500$ and 250, respectively. The domain of convergence for AUTO decreases with N, and values of N smaller than 1000 have very small domains of convergence.

5 Towards a general theory

The examples presented above demonstrate that multiple-shooting algorithms adapted to the slow-fast decomposition of trajectories in multiple time scale dynamical systems can be effective for computing periodic orbits. These algorithms are able to exploit the advantages of numerical integration methods for stiff systems to compute canards in vector fields with a single fast variable, where we use integration backward in time. The strategy presented here relies upon two ingredients. First, one must choose cross-sections for shooting that isolate the trajectory segments to be computed in forward time and those to be computed in backward time and, second, the computation must be placed in a continuation setting in which the periodic orbits vary at a moderate rate with respect to the continuation parameter. The rapid change of periodic orbits containing canards with respect to system parameters requires that the root finding procedure performed by the continuation algorithm is able to vary a system parameter as well as phase space variables.

Development of algorithms that make suitable choices of cross-sections and continuation strategy is likely to require good methods for automatically computing aspects of the slow-fast decomposition of trajectories in order to base locating suitable cross-sections upon this information. In particular, such methods should determine where degenerate decompositions are encountered and use information about the types of canards that are associated with these decompositions.

Canards in slow-fast systems with more than one fast variable typically lie along sheets of the critical manifold that consist of saddle points for the layer equations. Accurate computation of these canards cannot be done with either forward or backward numerical integration. Instead, two-point boundary value solvers, methods that are designed for computing normally hyperbolic manifolds (see Chapter 4) or methods for shadowing of trajectories of vector fields [4, 24] will need to be incorporated into shooting methods when one wants to compute relaxation oscillations that contain these canards. These algorithms will require more computation than numerical integration, but they still are likely to provide a good alternative to collocation methods for these problems.

Acknowledgments

This research was partially supported by grants from the National Institutes of Health, the Department of Energy and the National Science Foundation.

References

1. E. J. Doedel, R. C. Paffenroth, A. R. Champneys, T. F. Fairgrieve, Yu. A. Kuznetsov, B. E. Oldeman, B. Sandstede, and X. J. Wang. Auto2000: Continuation and bifurcation software for ordinary differential equations. available via `http://cmvl.cs.concordia.ca/`.
2. M. Cartwright and J. Littlewood 1947 On nonlinear differential equations of the second order: II the equation $\ddot{y} - kf(y, \dot{y})\dot{y} + g(y, k) = p(t) = p_1(t) + kp_2(t)$, $k > 0, f(y) \geq 1$. *Ann. Math.* 48:472–94, 1947. [Addendum 1949 **50** 504–5
3. R. Clewley, M. D. LaMar and E. Sherwood. "PyDSTool" available via `http://sourceforge.net/projects/pydstool`
4. B. Coomes, H. Koçak and K. Palmer. Rigorous computational shadowing of orbits of ordinary differential equations. *Numer. Math.* 69:401–421 1995.
5. F. Dumortier and R. Roussarie. Canard cycles and center manifolds. With an appendix by Cheng Zhi Li. *Mem. Amer. Math. Soc.* 121, no. 577, x+100 pp, 1996.
6. N. Fenichel, Geometric Singular Perturbation Theory. *J. Diff Eq.* 31:53–98, 1979
7. K. Bold, C. Edwards, J. Guckenheimer, S. Guharay, K. Hoffman, J. Hubbard, R. Oliva and W. Weckesser. The forced van der Pol equation II: Canards in the reduced system. *SIAM J. Appl. Dyn. Syst.* 2:570–608, 2003.
8. J. Guckenheimer. Bifurcation and degenerate decomposition in multiple time scale dynamical systems. In J. Hogan, A. Champneys, B. Krauskopf, M. di Bernardo, E. Wilson, H. Osinga, and M. Homer, editors, *Nonlinear Dynamics and Chaos: where do we go from here?'*, pages 1–21. Institute of Physics Publishing, Bristol, 2002.
9. J. Guckenheimer J 2004 Bifurcations of relaxation oscillations. In *Normal forms, bifurcations and finiteness problems in differential equations*, pages 295–316, NATO Sci. Ser. II Math. Phys. Chem., 137, Kluwer Acad. Publ., Dordrecht, 2004.

10. J. Guckenheimer, K. Hoffman and W. Weckesser. Numerical Computation of Canards. *International Journal of Bifurcation and Chaos*, 10:2669–2687, 2000.
11. J. Guckenheimer, K. Hoffman and W. Weckesser. The Forced van der Pol Equation I: The Slow Flow and its Bifurcations. *SIAM J. App. Dyn. Sys.* 2:1–35, 2003.
12. J. Guckenheimer and B. Meloon. Computing Periodic Orbits and their Bifurcations with Automatic Differentiation. *SIAM J. Sci. Comp.*, 22:951–985, 2000.
13. J. Guckenheimer, M. Wechselberger, L.-S. Young. Chaotic attractors of relaxation oscillators. *Nonlinearity*, 19:701–720, 2006.
14. R. Haiduc. Horseshoes in the forced van der Pol equation, PhD dissertation, Cornell University, 2005.
15. E. Hairer, S. P. Norsett and G. Wanner. *Solving Ordinary Differential Equations I, 2nd. ed.* Springer-Verlag, 1993.
16. C. K. R. T. Jones. Geometric singular perturbation theory. In *Dynamical Systems*, *Springer Lecture Notes Math 1609*, pages 44–120, 1995.
17. N. Levinson. Perturbations of discontinuous solutions of non-linear systems of differential equations. *Acta Math.*, 82:71–106, 1950.
18. J. Littlewood. On nonlinear differential equations of the second order: III the equation $\ddot{y} - k(1 - y^2)\dot{y} + y = bk\cos(\lambda t + a)$ for large k and its generalizations. *Acta math.*, 97:267–308, 1957. [Errata at end of 1957, 2.]
19. J. Littlewood. On nonlinear differential equations of the second order: III the equation $\ddot{y} - k(1 - y^2)\dot{y} + y = bk\cos(\lambda t + a)$ for large k and its generalizations. *Acta math.*, 98:1–110, 1957.
20. E. F. Mishchenko, Yu. S. Kolesov, A. Yu. Kolesov and N. Kh. Rhozov. *Asymptotic methods in singularly perturbed systems. Monographs in Contemporary Mathematics*, (Consultants Bureau, New York, A Division of Plenum Publishing Corporation), 1994.
21. P. Szmolyan and M. Wechselberger. Canards in $\mathbb{R}^3$. *J. Diff. Eq.*, 177:419–453, 2001.
22. P. Szmolyan and M. Wechselberger. Relaxation Oscillations in $\mathbb{R}^3$. *J. Diff. Eq.*, 200:69–104, 2004.
23. B. van der Pol. On relaxation oscillations, *Phil. Mag.*,2:978–992, 1926.
24. E. Van Vleck. Numerical shadowing near hyperbolic trajectories. *SIAM J. Sci. Comput.* 16:1177–1189, 1995.
25. A. B. Vasil′eva. Asymptotic behaviour of solutions of certain problems for ordinary non-linear differential equations with a small parameter multiplying the highest derivatives. *Russian Mathematical Surveys*, 18:13–84, 1963. (Russian) *Uspehi Mat. Nauk*, 18:15–86, 1963.