SIAM J. Applied Dynamical Systems
Vol. 4, No. 3, pp. 711–732

Projecting to a Slow Manifold: Singularly Perturbed Systems and Legacy Codes*

C. W. Gear†, T. J. Kaper‡, I. G. Kevrekidis§, and A. Zagaris‡

Abstract. We consider dynamical systems possessing an attracting, invariant "slow manifold" that can be parameterized by a few observable variables. We present a procedure that, given a process for integrating the system step by step and a set of values of the observables, finds the values of the remaining system variables such that the state is close to the slow manifold to some desired accuracy. It should be noted that this is not equivalent to "integrating down to the manifold" since the latter process may significantly change the values of the observables. We consider problems whose solution has a singular perturbation expansion, although we do not know what it is nor have any way to compute it (because the system is not necessarily expressed in a singular perturbation form). We show in this paper that, under some conditions, computing the values of the remaining variables so that their $(m+1)$st time derivatives are zero provides an estimate of the unknown variables that is an mth-order approximation to a point on the slow manifold in a sense to be defined. We then show how this criterion can be applied approximately when the system is defined by a legacy code rather than directly through closed form equations. This procedure can be valuable when one wishes to start a simulation of the detailed model on the slow manifold with particular values of observable variables characterizing the slow manifold.

Key words. initialization, DAEs, singular perturbations, legacy codes, inertial manifolds

AMS subject classifications. 35B25, 35B42, 37M99, 65P99

DOI. 10.1137/040608295

1. Introduction.

The derivation of reduced dynamic models for many chemical and physical processes hinges on the existence of a low-dimensional, attracting, invariant "slow manifold" characterizing the long-term process dynamics. This manifold is parameterized by a small number of system variables (or "observables," functions of the system variables); when the dynamics have approached this manifold, knowing these observables suffices to approximate the full system state. A reduced dynamic model for the evolution of the observables can then in principle be deduced; the resulting simplification in complexity and in size can be vital in understanding and modeling the full system behavior.

We will consider systems that we know have a slow manifold, and for which we know a set of variables that are sufficient to parameterize that manifold, but we do not know, and

*Received by the editors May 14, 2004; accepted for publication (in revised form) by M. Golubitsky February 8, 2005; published electronically August 31, 2005. This work was partially supported by AFOSR (Dynamics and Control) and NSF/ITR grant (C.W.G., I.G.K.), and by NSF grant 0306523, Div. Math Sci. (T.J.K., A.Z.).
http://www.siam.org/journals/siads/4-3/60829.html

†Department of Chemical Engineering, A-217 Engineering Quadrangle, Princeton University, Princeton, NJ 08544 (wgear@princeton.edu).

‡Department of Mathematics and Center for BioDynamics, Boston University, Boston, MA 02215 (tasso@math.bu.edu, azagaris@math.bu.edu).

§Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544 (yannis@princeton.edu).

have no way of deriving, a reduced description. We do, however, have access to a computer model for the full system. We will call this the *microscopic model* and the computation using it *microscopic computation*. We want a procedure that, given the values of the variables that parameterize the slow manifold, lets us approximate the values of all system variables on the slow manifold. Such a procedure will be important for applications to be described at the end of this introduction.

Reduced dynamic models have been the subject of intense study from the theoretical, practical, and computational points of view. Low-dimensional center-unstable manifolds are crucial in the study of normal forms and bifurcations in dynamical systems (e.g., [15]); the theory of inertial manifolds and approximate inertial manifolds [6, 33] has guided model reduction in dissipative partial differential equations; the study of fast/slow dynamics in systems of ODEs is the subject of geometric singular perturbation theory (e.g., [7]). On the modeling side, the Bodenstein "quasi-steady state" approximation has long been the basis for the reduction of complex chemical mechanisms, described by large sets of ODEs [2] (see also the discussion by Turanyi, Tomlin, and Pilling [35]). More recently, an array of computational approaches have been proposed that bridge singular perturbation theory with large-scale scientific computation for such problems; they include the computational singular perturbation (CSP) approach of Lam and Goussis [23, 24, 25] and the intrinsic low-dimensional manifold approach of Maas and Pope [26]. The mathematical underpinnings of these methods have also been studied [17, 36]. Lower-dimensional manifolds arise naturally also in the context of DAEs, where the initialization problem has attracted considerable attention (e.g., [3]).

Remarkably, the same concept of separation of time scales and low-dimensional long-term dynamics underpins the derivation of "effectively simple" descriptions of complex systems. In this context, the detailed model is a collection of agents (molecules, cells, and individuals) interacting with each other and their environment; the entire distribution of these agents evolves through atomistic or stochastic dynamic rules. In many problems of practical interest it is possible to write macroscopic equations for the large-scale, coarse-grained dynamics of the evolving distribution in terms of only a few quantities, typically lower moments of the evolving distribution. In the case of isothermal Newtonian flow, for example, we can write closed evolution equations for the density and the momentum fields, the zeroth and first moments of the particle distribution over velocities. This is again a singularly perturbed problem; only in this case the higher moments of the evolving distribution have become quickly slaved to the lower ones (in this case stresses, after a few collisions, have become functionals of velocity gradients). Newton's law of viscosity therefore represents a similar type of "slow manifold" as in the ODE case discussed above; fast variables (stresses) become functionals of velocity gradients, and the slow manifold is embodied in the *closure*—Newton's law of viscosity. The use of slow manifolds in nonequilibrium thermodynamics, and more generally in the study of complex systems, is also a subject of intense current research (see the work of Gorban, Karlin, Oettinger, and coworkers [13, 14], as well as [20, 21] for some recent computational studies). In this context, one has an "inner simulator" at the microscopic or stochastic level for evolving the detailed distributions; a separation of time scales does not arise at this level but rather at the level of the evolution of the *statistics* or *moments* of these distributions. Typically the lower moments of the evolving distributions parameterize the slow manifold, while the higher moments quickly evolve to functionals of the lower ones. Since we do not have explicit

formulas for the equations at the coarse-grained, macroscopic level, the following interesting question arises: Can we benefit from singular perturbation, when no closed form evolution equations are available, and the only tool at our disposal is a "black-box" dynamic simulator of the detailed problem? This is the problem we will address in this paper.

We will assume that we are given an evolutionary system which can be described by

$$u' = p(u, v), \tag{1.1}$$

$$v' = q(u, v), \tag{1.2}$$

where prime designates differentiation w.r.t. t, the dimensions of u and v are N_u and N_v, respectively, and values, $u(0)$, are specified only for u. In the case of a legacy dynamic code, we may not even be given the formulas for these equations explicitly; instead, we may be given a time-stepper of the above system as a black-box subroutine—a code that, provided an initial condition for u and v, will return an accurate approximation of u and v after a time interval (a reporting horizon) ΔT. We wish to find $v(0)$ so that the solution is "close to a slow manifold." This statement is deliberately vague because in practice we are proceeding on the belief that there exists a slow, attracting, invariant manifold that can be parameterized by u and that the variables v in some sense "contain" the fast variables so that their values *on the slow manifold* can be computed from the values of u at any time. (Note that we do not need to *know* which are the slow variables—only to be able to identify a set of variables sufficient to parameterize the slow manifold.) This implies that the manifold is the graph of a function $v = v(u)$ over the observables u. As we proceed, we will make these statements more precise. A mathematically rigorous analysis will be the subject of a future report. Out intent here is to present a new method, discuss implementation issues, and demonstrate its behavior.

We assume that the system can be expressed in terms of some other variables, x and y, of the same dimension as u and v, respectively, and that in terms of x and y the system can be written in the usual singular perturbation form,

$$x' = f(x, y), \tag{1.3}$$

$$\epsilon y' = g(x, y), \tag{1.4}$$

where x and y are also of dimensions N_u and N_v, respectively. We will assume that their initial values are specified as $x(0)$ and $y(0)$, independent of ϵ. The singular perturbation parameter, ϵ, is associated with the gap, or ratio, between the "active" (slow) eigenvalues and the rightmost of the negative, inactive eigenvalues of the linearized problem, locally. We stress that we *do not* assume that we know how to express the equations in the form of (1.3) and (1.4), nor do we have any knowledge of the transformation

$$u = u(x, y), \quad x(0, \epsilon) = x(0), \tag{1.5}$$

$$v = v(x, y), \quad y(0, \epsilon) = y(0), \tag{1.6}$$

other than that we assume that it is well conditioned and does not have large derivatives. We will consider singular perturbation expansions in ϵ, even though the parameter is not identified (and cannot be varied). The functions f and g could also involve the parameter ϵ, but that serves little in this presentation other than to complicate the algebra.

The standard singular perturbation expansion for the solution of (1.3) and (1.4) takes the form

$$x(t,\epsilon) = \sum_{n=0}^{\infty} \epsilon^n X_n(t) + \epsilon \sum_{n=0}^{\infty} \epsilon^n \xi_n(\tau), \tag{1.7}$$

$$y(t,\epsilon) = \sum_{n=0}^{\infty} \epsilon^n Y_n(t) + \sum_{n=0}^{\infty} \epsilon^n \eta_n(\tau). \tag{1.8}$$

The new time variable $\tau = t/\epsilon$ is the stretched time. This involves an outer solution $(X(t), Y(t))$ that is smooth in the sense that its time derivatives are modest, and an inner solution $(\xi(\tau), \eta(\tau))$ that captures the fast boundary layer where the solution typically changes like $e^{-t/\epsilon} = e^{-\tau}$. Both outer and inner solutions are expressed as power series in ϵ. The inner solution is fast in the sense that each differentiation by t introduces a multiplier of $1/\epsilon$.

We define the slow manifold as the manifold that contains all solutions of (1.3) and (1.4) of the form (1.7) and (1.8) with the inner solution identically zero. This is an invariant manifold. We say that a *solution* is an mth-order approximation to a slow manifold solution if it has the form in (1.7) and (1.8) with the first $m+1$ terms of the inner solution expansion identically zero. A *point* (u, v) is an mth-order approximation to the slow manifold, or mth-order close to the slow manifold, if it lies on an mth-order approximate solution.

We want to stress that we are not proposing a technique for finding the singular perturbation expansion. Rather, we are using the ideas as a scaffold for the theoretical justification of the proposed computational method. It is possible that the method will provide answers even when the singular perturbation expansions do not converge (although in that case we have no justification other than an intuitive one). The procedure we propose for finding the v values that are close to the slow manifold given the u values is to find values of v that approximately solve the N_v-dimensional nonlinear equation

$$\frac{d^{m+1}v}{dt^{m+1}} = 0, \tag{1.9}$$

which we call "the $(m+1)$st derivative condition." Compare this with the "bounded derivative principle" [22] which requires the first m time derivatives to be of order 1. That condition can be used to select nonoscillating solutions from a family of solutions, almost all of which have fast oscillating components. Ours cannot, but it is simpler to apply to the types of problems we consider. Girimaji [12] has discussed a related idea[1]—that of minimizing the first derivative of the full system vector (u and v in our notation). He also suggests getting a higher-order accuracy by minimizing a higher derivative.

Note that (1.9) is a *local condition*—that is, it is applied at a single time, which we will take to be $t = t_0 = 0$—to determine a value of v corresponding to a given value of u. A solution of (1.1) and (1.2) starting from these values of u and v will not satisfy (1.9) for $t > 0$, but we do expect the solution to be close to the slow manifold. Intuitively the condition in (1.9) finds a point close to the slow manifold because differentiation "amplifies" rapidly

[1]We are grateful to a referee for pointing out this citation.

varying components more than slowly varying components, so (1.9) seeks a region where the fast components are small. We will suggest ways in which (1.9) can be solved approximately in practical codes, even those based on a legacy code for the integration of (1.1) and (1.2).

While the approach will be presented and implemented for singularly perturbed sets of ODEs, the "legacy code" formulation is appropriate also in cases where the inner simulator is not a differential equation solver, but rather a microscopic/stochastic simulator. In the *equation-free approach* we have been developing for the computer-assisted study of certain classes of complex systems, the variables u correspond to macroscopic observables of a microscopic simulation (typically, moments of a stochastically or deterministically evolving distribution). In this case v corresponds to statistics of the evolving distribution (e.g., higher moments) that become quickly slaved to (become functionals of) the observables u; thus the analogy of a slow manifold persists in moment space for the evolving distribution.

In previous work [11, 20] we have shown how brief bursts of microscopic computation can be used to effectively compute on the slow manifold. For example, with projective integration [8] we can use microscopic computation to compute the values of *macroscopic* variables (ones that parameterize the slow manifold) over a sequence of time intervals and then effectively perform a large time step, explicit integration using extrapolation of the macroscopic variables. After the projective step, it is necessary to *lift* from the values of the macroscopic variables to compute suitable values of all variables to initiate a further microscopic simulation. One use of the procedure described in this paper would be to perform that lifting.

In earlier work we assumed that we could lift in any reasonable way and that the system would quickly equilibrate the values of the other variables under microscopic simulation with little change in the macroscopic variables; in other words, the microscopic simulation would bring us back to the slow manifold quickly. This is sometimes called the brute force approach. In many cases it is effective, but let us consider the following extremely simple example in which it fails:

$$\begin{aligned} \epsilon u' &= -(u+w)/2, \\ v' &= -1, \\ \epsilon w' &= -(u+w)/2 \end{aligned}$$

with $0 < \epsilon \ll 1$. By making the transformation $x = (u-w)$, $y = v$, $z = (u+w)$ we see that the slow manifold is $z = u + w = 0$ and that solutions starting away from this manifold approach it approximately along normals to the manifold. The manifold can be parameterized by u and v and the corresponding point on the manifold is given by $w = -u$. However, if we were given an initial value u_0, v_0 and we found w_0 by the "brute force" approach of guessing a value of w_0 and integrating until we were on the slow manifold (assuming we had a computational mechanism to determine when we had almost reached the slow manifold) we would arrive close to the point $([u_0 - w_0]/2, v_0, [w_0 - u_0]/2)$, a distance of $(u_0 - w_0)^2/\sqrt{2}$ from the desired point. Since this example has been deliberately chosen so that these errors are persistent, it is clear that we need a mechanism to find an approximation to the "correct" point on the slow manifold in some cases. This simple example clearly illustrates the situation in which the brute force method will fail; if the hyperplane corresponding to the observable variables (u and v in this example) is not approximately orthogonal to the fast flow some distance from

the slow manifold, integrating the system to the slow manifold will change the observable variables if we start far from the slow manifold.

The paper is organized as follows: In the next section we will outline the theory that motivates the derivative condition. In section 3 we discuss ways in which the derivative condition can be approximately applied as a difference condition and used with legacy codes. Section 4 presents some simple examples of its application. We conclude with a brief summary and outline of the scope of the method and some of the challenges we expect to arise in its wider application.

2. Outline of the theoretical basis. In this section we will indicate how the application of the condition in (1.9) leads to an mth-order approximation to the slow manifold under suitable smoothness and smallness conditions. We will start by sketching the parts of singular perturbation expansion theory that we need by paraphrasing the presentation in O'Malley's monograph [28, pp. 46–52]. We will use the same notation to make it easier for the reader who wishes to get more details from that book.

In the following we will write X_n, Y_n, ξ_n, and η_n to mean $X_n(t)$, $Y_n(t)$, $\xi_n(\tau)$, and $\eta_n(\tau)$. We recall from [28] that the t and τ dependencies are treated separately, and that the terms in the outer expansion, $\{X_n, Y_n\}$, are obtained term by term by substituting (1.7) and (1.8) into (1.3) and (1.4), and equating each outer term in ϵ^n to zero, starting with $n = 0$. For $n = 0$ we obtain the DAE

$$X_0' = f(X_0, Y_0), \quad X_0(0) = x(0), \tag{2.1}$$

$$0 = g(X_0, Y_0). \tag{2.2}$$

The existence of a smooth solution of this equation for any $x(0)$ requires the assumption that g_y is nonsingular. (The existence of asymptotic expansions for the inner and outer components requires the stronger assumption that g_y is a stable matrix.) Then, Y_0 is specified uniquely in terms of X_0 by (2.2), say, as

$$Y_0 = \phi(X_0).$$

The nth term in the power series yields the DAE

$$X_n' = f_x(X_0, Y_0)X_n + f_y(X_0, Y_0)Y_n + \tilde{f}_n, \tag{2.3}$$

$$0 = g_x(X_0, Y_0)X_n + g_y(X_0, Y_0)Y_n + \tilde{g}_n, \tag{2.4}$$

where the $\tilde{f}_n$ and $\tilde{g}_n$ are defined in terms of earlier terms in the outer expansion, $\{X_j, Y_j\}$, $j = 0, \ldots, n-1$. The initial condition, $X_j(0)$, has to be specified. Since we have set $X(0) = x(0)$, $X_j(0)$ is obtained from (1.7) by requiring that the ϵ^j term vanishes at $t = 0$, which gives

$$X_j(0) = -\xi_{j-1}(0). \tag{2.5}$$

We are most interested in the way in which the inner terms are defined, since we wish to annihilate the first $m+1$ of these to get an mth-order approximation. These are obtained by considering the change in the inner terms as τ varies for arbitrarily small ϵ—in other words,

with $t = 0$ and the outer solutions fixed at their initial values. Following [28], we consider terms in successive powers of ϵ and find that the ϵ^0 terms satisfy

$$\dot{\xi}_0 = f(x(0), Y_0(0) + \eta_0) - f(x(0), Y_0(0)), \tag{2.6}$$

$$\dot{\eta}_0 = g(x(0), Y_0(0) + \eta_0) - g(x(0), Y_0(0)), \tag{2.7}$$

where a dot represents differentiation w.r.t. $\tau = t/\epsilon$. If we have an initial value for $\eta_0(0)$ we can solve (2.7) for η_0. Equation (2.6) gives ξ_0 as an indefinite integral, so it is determined by specifying ξ_0 at any point. This is normally done at $\tau = \infty$—in other words, at the end of the boundary layer. However, in our development here we will be showing that η_j and ξ_j are identically zero for $j \le m$, so we will actually choose $\xi_j(0) = 0$ (so that we also have $X_{j+1}(0) = 0$ from (2.5)). Subsequent inner terms satisfy

$$\dot{\xi}_n = f_y(x(0), Y_0(0) + \eta_0)\eta_n + \hat{f}_n, \tag{2.8}$$

$$\dot{\eta}_n = g_y(x(0), Y_0(0) + \eta_0)\eta_n + \hat{g}_n, \tag{2.9}$$

where $\hat{f}_n$ and $\hat{g}_n$ are functions of the earlier terms, $\{\xi_j, \eta_j\}$, $j = 0, \ldots, n-1$. In particular, if all of these terms are zero, then $\hat{f}_n$ and $\hat{g}_n$ are zero. Equation (2.9) can be solved if an initial value is known for $\eta_n(0)$. Once again, (2.8) gives ξ_n as an indefinite integral.

In the following we are going to show, one by one, that $\eta_j(0) = 0$ for $j = 0, 1, \ldots, m$, and that we can then choose $\xi_j(0) = 0$. Note that once we have shown that $\eta_0(0) = 0$, then (2.6) and (2.7) indicate that $\eta_0(\tau) = 0$ and that $\xi_0(\tau)$ is constant, which we can make zero by choosing $\xi_0(0) = 0$. Then it follows that $\hat{f}_1$ and $\hat{g}_1$ are identically zero. Repeating this argument, we see that if $\eta_j(0) = 0$, $\xi_j(0) = 0$, $j = 0, \ldots, m$, then $\xi_j(\tau)$ and $\eta_j(\tau)$ are identically zero for $j \le m$. This provides a solution that is an mth-order approximation.

Now we return to the original problem phrased in terms of u and v. If we knew the transformation to x and y we would do better to work in that space, but our assumption is that, although a transformation exists, it is unknown and we have to work with u and v. We want to show that if the $(m+1)$st derivative of v is zero, then the point is mth-order close to the slow manifold. All terms below are evaluated at $t = 0$ or $\tau = 0$, the time at which we are attempting to solve (1.9). We will simplify the notation and write η_j for $\eta_j(0)$ and similarly for other terms in the following. We have from (1.6)

$$\frac{d^{m+1}v}{dt^{m+1}} = v_y \frac{d^{m+1}y}{dt^{m+1}} + \text{other terms}, \tag{2.10}$$

where the other terms involve either partial derivatives of v w.r.t. x and/or multiple derivatives of v w.r.t. y and products of derivatives of y.

Substituting from (1.8) into (2.10) and extracting the lowest-order term in ϵ (ϵ^{-m-1}), we find that at $t = 0$

$$\frac{d^{m+1}v}{dt^{m+1}} = \epsilon^{-m-1} v_y \frac{d^{m+1}\eta_0}{d\tau^{m+1}} + \text{other terms} + \mathrm{O}(\epsilon^{-m}), \tag{2.11}$$

where now the other terms include products of a higher-order partial derivative of v w.r.t. y with more than one τ-derivative of η_0 such that the sum of the levels of differentiation is $m+1$,

that is, terms like

$$v_{yy}\frac{d^k\eta_0}{d\tau^k}\frac{d^{m+1-k}\eta_0}{d\tau^{m+1-k}}$$

and terms with higher partial derivatives and more derivatives of η_0 in the product. Note that whenever η_0 appears, it is always differentiated w.r.t. τ at least once. Also note that we do not get any terms involving ξ_0 because of the additional ϵ appearing in front of the inner solution expansion for x in (1.7).

Now we use (2.7) to find the higher-order derivatives of η_0 w.r.t. τ. We get for $p > 1$

$$\frac{d^p\eta_0}{d\tau^p} = g_y^{p-1}\dot{\eta}_0 + \text{other terms}, \tag{2.12}$$

where the other terms involve $\dot{\eta}_0^j$ with $j > 1$. Substituting (2.12) in (2.11), we arrive at

$$\frac{d^{m+1}v}{dt^{m+1}} = \epsilon^{-m-1}\left[v_y g_y^m\dot{\eta}_0 + \sum_{j=2}^{m+1} v_z g_z \dot{\eta}_0^j\right] + \mathrm{O}(\epsilon^{-m}), \tag{2.13}$$

where the notation $v_z g_z \dot{\eta}_0^j$ stands for the evaluation of j-linear forms involving the partial derivatives of v and g on the vector $\dot{\eta}_0$. Equating the leading term of the right-hand side of (2.13) to zero, we now have a polynomial equation for $\dot{\eta}_0$ as

$$v_y g_y^m\dot{\eta}_0 + \sum_{j=2}^{m+1} v_z g_z \dot{\eta}_0^j = 0. \tag{2.14}$$

One solution of this is

$$\dot{\eta}_0 = 0,$$

and it is an isolated root as long as $v_y g_y$ is nonsingular. Since we have assumed that g_y is a stable matrix for the existence of a singularly perturbed solution (and hence a slow manifold) and that v in some sense spans the fast variables (meaning that v_y is nonsingular), this is no problem. If all other partial derivatives involved in (2.14) are "of order 1," then other solutions are also of order 1, i.e., well separated from the zero solution. We will delay discussion of how to avoid the "wrong" solutions for the moment and assume that we find the zero solution. (If the problem is linear, these other terms are null, so there are no other solutions, and it is only in the case of high nonlinearity when the partial derivatives are large that these other solutions can become small and cause problems.)

If $\dot{\eta}_0 = 0$, then (2.7) tells us that $\eta_0 = 0$ because we have assumed that g_y is a stable matrix (in the domain of interest). This immediately implies that $\xi_0 = 0$ (or is a constant that can be absorbed into the outer solution, thus making $\xi_0 = 0$).

As discussed following (2.9), the vanishing of η_0 and ξ_0 means that the last terms of (2.8) and (2.9) are zero for $n = 1$, making them look similar to (2.6) and (2.7). Therefore, the above argument can now be applied to show that η_1 and ξ_1 are zero. This argument can be

repeated for higher-order terms as long as the power of ϵ in (2.13) remains negative—in other words, until we have shown that

$$\eta_j = \xi_j = 0, \qquad j = 0, \ldots, m.$$

Note that when we have made the $(m+1)$st derivative zero, the lower-order derivatives will not be zero, or even small. This is because a small movement away from the slow manifold can make large changes in the derivatives of the inner solution. However, the difference between successive v values as we make m successively larger is small (of order ϵ^{m+1}) as we go from the mth to the $(m+1)$st derivative condition as the v values are converging to the slow manifold. Hence one way to solve for zeros of high-order derivatives would be to start by finding the zeros of the first-order derivative, then repeating for successively higher-order derivatives, each step requiring smaller and smaller changes to v, until we have found the zeros of the $(m+1)$st derivatives using whatever computational process is appropriate. (The computational process is addressed in the next section.) This procedure helps address the issue of finding the smallest of multiple roots of (2.14) since, for $m = 0$, there is only one root so that the iteration for $m = 1$ and larger m starts with a good approximation. If the zero root is well separated from the others, we will converge to it.

3. Practical implementation. It is often not practical to work with higher-order derivatives of a differential equation, either because they are algebraically complicated or because the equations are defined by a "legacy code," that is, as an implementation of a step-by-step integrator that effectively cannot be changed or analyzed. (The same would be true if part of the derivative calculations involved table look-up functions that could be difficult to differentiate.) Therefore, we are interested in methods that do not require direct access to the mathematical functions constituting the differential equation. The same rationale applies when the "inner simulator" simulates the system at a different level (i.e., in the form of an evolving microscopic or stochastic distribution). In this case we have only a time-T map for the macroscopic observables that results from running the microscopic simulator and monitoring the evolution of the observables (e.g., particle densities) in time [11, 34].

The obvious alternative is to use a forward difference approximation to the derivative, noting that if

$$\Delta^{m+1} v(t) \equiv \Delta^m v(t+H) - \Delta^m v(t) \tag{3.1}$$

with $\Delta^0 v(t) = v(t)$, then

$$\Delta^{m+1} v(t) = H^{m+1} \frac{d^{m+1} v}{dt^{m+1}} + \mathrm{O}(H^{m+2}).$$

It turns out that there is a straightforward way to implement a functional iteration to find a zero of the $(m+1)$st forward difference, even if we have only a "black-box" code that integrates (1.1) and (1.2). Suppose we have operators, ϕ and θ, that, given values of $u(t)$ and $v(t)$, yield approximations to $u(t+H)$ and $v(t+H)$, namely,

$$u(t+H) \approx \phi(u(t), v(t)), \tag{3.2}$$

$$v(t+H) \approx \theta(u(t), v(t)). \tag{3.3}$$

Letting $t_j = t_0 + jH$ and applying (3.2) and (3.3) $m+1$ times starting from $t = t_0$, we can compute approximations

$$u_{j+1} = \phi(u_j, v_j) \approx u(t_{j+1}), \tag{3.4}$$

$$v_{j+1} = \theta(u_j, v_j) \approx v(t_{j+1}) \tag{3.5}$$

for $j = 0, \ldots, m$. The functional iteration to find a v_0 for a given u_0 such that the $(m+1)$st difference is zero (we call this the *constraint iteration*) consists of the following steps:

1. Start with the given u_0 and for v_0 any guess.[2]
2. Set the iteration number $p = 1$.
3. Set the current iterate $v_0^{(1)} = v_0$.
4. Apply (3.4) and (3.5) $m+1$ times starting from $u_0, v_0^{(p)}$ to generate $v_1^{(p)}, v_2^{(p)}, \ldots, v_{m+1}^{(p)}$.
5. Compute $\delta = (-1)^m \Delta^{m+1} v_0^{(p)}$.
6. If δ is sufficiently small, the iteration has converged.
7. If δ is not sufficiently small,

$$v_0^{(p+1)} = v_0^{(p)} + \delta. \tag{3.6}$$

8. Increment p and return to step 4.

The size of the smallness threshold in step 6 will typically decrease with m; in fact, one could argue it should be proportional to H^{m+1}. On the other hand, if one can get quick convergence, it makes sense to set it in the vicinity of machine precision. This point is discussed further in the section on examples. If the black-box integrator provides a good integration (that is, it does not introduce spurious oscillations due to near instability) and H is small enough, this process will converge on a zero of the difference.

As an illustration we consider the case $m = 0$ and assume that the integrator is simply forward Euler with a step size such that its product with the magnitude of the largest eigenvalue is less than one. We see that the process for v consists of computing

$$\delta = v_1 - v_0 \approx H \frac{dv}{dt}(t_0).$$

If this is insufficiently small, we replace v_0 with $v_0 + \delta = v_1$. This is just the stationary projection process used in [9] and is related to the "reverse time" projective integration method in [10]. In the case of $m = 1$ we compute δ as

$$\delta = -v_2 + 2v_1 - v_0.$$

[2]Obviously the initial value v_0 has to be in the basin of attraction of the iterative process. The context in which this procedure is used may provide natural ways to compute a good guess. For example, if it is being used in projective integration, the final value of v computed in the previous projective step might provide a good initial value in the next one. Because our process for approximating the slow manifold involves several integration steps of the microscopic system, one can also test to see if the rate of change seems to be decreasing, indicating an approach to an attracting slow manifold, or increasing, suggesting that perhaps the initial value has strayed too far. In the end, one has to face all of the problems of selecting a good initial value of any iterative process. However, if we get convergence, the final result is not dependent on the choice of the initial value in the sense that a brute force integration to the slow manifold would be (although both methods could get to another branch of the stable manifold).

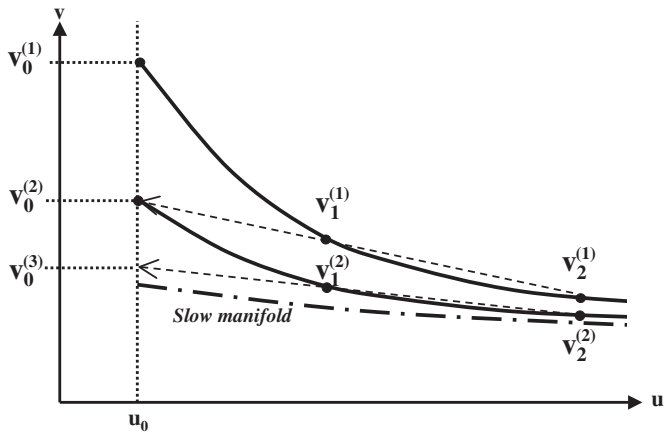

Figure 1. *Iterative process for $m = 1$.*

If this is insufficiently small, we add it to v_0 to get a new v_0 given by

$$v_0 = 2v_1 - v_2.$$

This is precisely the linear interpolant through v_1 and v_2 back to the starting point. It is illustrated in Figure 1. (This figure may be a little confusing because it is drawn in the u-v plane to emphasize that u is being held constant from iteration to iteration. The backward interpolant, however, is really taking place in the t-v plane.) The general iteration is equivalent to the obvious extension of that: an mth-degree interpolant is passed through $v_1, v_2, \ldots, v_{m+1}$ to compute a new approximation to v_0. This is conveniently done using differences as described above.

Why does this iteration equation (3.6) converge for small H under reasonable circumstances? *Intuitively, the forward integration exponentially damps the fast components more than they are amplified by the polynomial extrapolation backward.* In more mathematical terms, the iteration takes the form

$$v_0^{new} = v_0 + \delta. \tag{3.7}$$

If $\partial\delta/\partial v_0$ is negative definite and small, this will converge. We have

$$\partial\delta/\partial v_0 = -(-H)^{m+1}\frac{\partial(\frac{d^{m+1}v}{dt^{m+1}})}{\partial v} + \mathrm{O}(H^{m+2}).$$

The term

$$\frac{\partial(\frac{d^{m+1}v}{dt^{m+1}})}{\partial v}$$

is dominated by

$$\epsilon^{m+1} g_y^{m+1}$$

in powers of ϵ. Since we have assumed that g_y is a negative definite matrix for the existence of a singular perturbation expansion, convergence follows for small enough H.

In the above discussion we have used the attractivity of the manifold and, in effect, successive substitution in order to converge to a fixed point of our mapping. One can accelerate this computation by using fixed point algorithms, like Newton's method; clearly, since no equations and Jacobians are available, the problem lends itself to matrix-free fixed point implementations like the recursive projection method by Shroff and Keller [30] or Newton–Krylov implementations (see [18] and, for a GMRES-based implementation for time-steppers, see [19]).

4. Examples. We will consider three examples to illustrate the method. The Michaelis–Menten enzyme kinetics model is a classic example for singular perturbation (see [1] for a brief discussion on the history of the model and its analysis). Since it is a simple system we can make a direct comparison with an easily computable singular perturbation expansion. The second example is a realistic five-dimensional chemical reaction system with a one-dimensional slow manifold. As in many real examples, we do not know the slow manifold and can only show "plausibility" of our solution. The final example is a contrived five-dimensional nonlinear system with a known two-dimensional slow manifold so that we can compute the "errors" as the distance from the slow manifold.

4.1. Michaelis–Menten equation. This is given in a singular perturbation form in [28] as

$$x' = -x + (x + \kappa - \lambda)y, \tag{4.1}$$

$$\epsilon y' = x - (x + \kappa)y \tag{4.2}$$

with $x(0) = 1$. For some simple tests we will take $\kappa = 1$, $\lambda = 0.5$, and $\epsilon = 0.1$ and 0.01. (These are larger figures than typical for reactions, but we wish to show that the method works even for problems with a relative small gap, and also to have problems where the higher-order initializations are visibly different from the lowest-order ones.)

In the following tests, we implemented the operators ϕ and θ in (3.2) and (3.3) using n steps of forward Euler with step size h to simulate a microscopic integrator with step size $H = nh$. In all cases (3.7) was iterated until δ was less than 10^{-14} (which is rather excessive, but we did not want any errors from premature termination to color the results).

We ran with $m = 0, \ldots, 4$ and compared the results with values believed to be good to 10 digits (0.5030315116 when $\epsilon = 0.1$ and 0.5003115178 when $\epsilon = 0.01$).[3] In this example, we set the initial value of the iterate $y_0 = 0$ when $m = 0$ and then used the solution for case m when initializing case $m + 1$. Two different combinations of h and n were used to illustrate the impact of changing $H = nh$, $h = \epsilon/100$, and $n = 1$ for a small H and $h = \epsilon/10$ and $n = 4$

[3]To compute the values "believed to be good to 10 digits," we numerically solved for the distinguished slow manifold that is a heteroclinic orbit from the degenerate saddle fixed point at infinity ($x \to \infty$, $y = 1$) to the origin. To obtain this representation, we applied Poincaré compactification to the Michaelis–Menten vector field and studied the induced dynamics on the Poincaré sphere using a chart in which the entire heteroclinic connection lies. Also we used a three-term expansion of the center-unstable manifold of the degenerate saddle at infinity in order to obtain a highly accurate initial segment of the manifold and integrated forward with a series of successively smaller step sizes to ensure convergence.

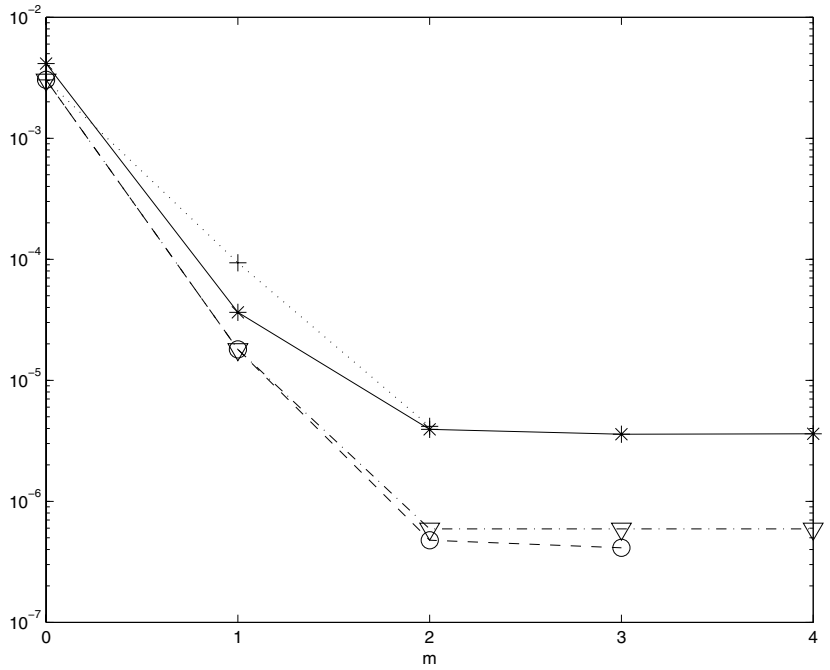

Figure 2. *Log error versus m. Asterisk is $H = 0.04$. Circle and triangle are $H = 0.001$ (triangle uses convergence acceleration). Plus is asymptotic expansion.*

for a large H. For $\epsilon = 0.1$, the logarithms of the errors versus m are shown in Figure 2. The circles indicate the results for small H and the asterisks indicate the results for large H. (The small H results for $m = 3$ and $m = 4$ were identical so the latter is not plotted.) As can be seen from the figure, the accuracy is better for larger m when H is smaller because of the poorer estimate of the derivative with large H.

We also show the value of the first $m + 1$ terms of asymptotic expansion of the outer solution with the + signs for $m = 0, \ldots, 2$. This solution is

$$y = \frac{x}{x+\kappa} + \frac{\kappa\lambda x}{(x+\kappa)^4}\epsilon + \frac{\kappa\lambda x(2\kappa\lambda - 3x\lambda - x\kappa - \kappa^2)}{(x+\kappa)^7}\epsilon^2 + \mathrm{O}(\epsilon^3). \tag{4.3}$$

The triangles indicate the errors in a solution obtained by convergence acceleration discussed below.

A larger H gives more rapid convergence. This is illustrated in Figure 3 which shows the number of iterations versus m. (In this figure the number of iterations shown for each m is the sum of the number of iterations used for that and each lower m since the result from each m is used to initialize the iteration for the next m.) As can be seen, the number of iterations is very much larger for the small H case (circles versus asterisks). However, this is normally a linearly convergent process and can easily be accelerated. The triangles and the stars show the number of iterations needed when the simplest acceleration technique is used on the circle and asterisk results, respectively. The numerical results are slightly different and are shown in Figure 2 for the triangle case. (The differences in the asterisk/star cases are not distinguishable

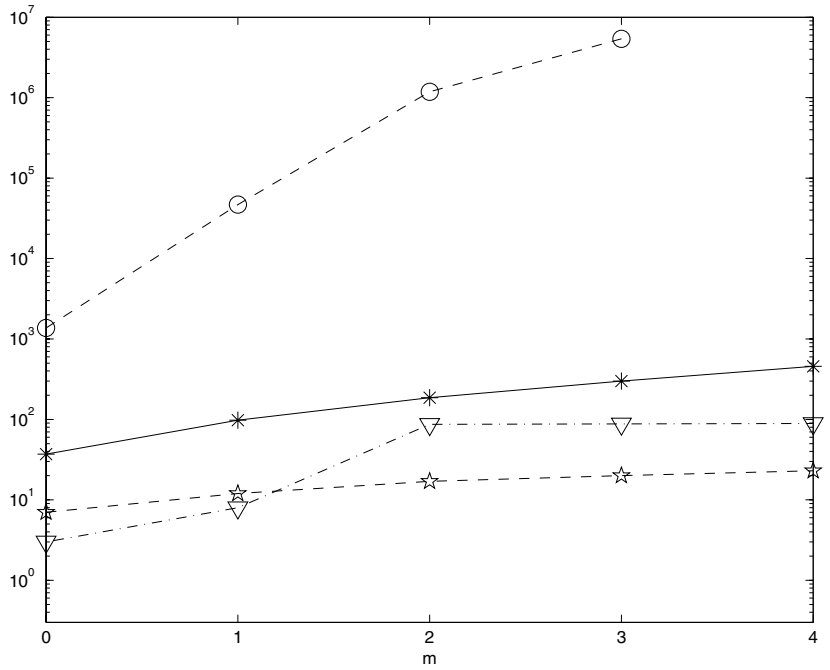

Figure 3. *Numbers of iterations versus m. Asterisk and star are $H = 0.04$. Circle and triangle are $H = 0.001$ (star and triangle have convergence acceleration).*

to plot accuracy, so they are not shown.) This figure clearly shows the improvement in convergence for larger H but also shows that with simple convergence acceleration, the rate of convergence is not really an issue. The results obtained for convergence rates and improvement by acceleration in this example were similar to those in the other examples and will not be further discussed.

The next case, shown in Figure 4, uses an ϵ smaller by a factor of 10. Because the higher-order terms in ϵ are now smaller, the convergence is faster with m. The computation was run with the small H case only and shows an improvement of about 2.5 orders of magnitude for each increase in m (versus about 1.5 when ϵ was ten times larger in Figure 2). Larger values of m gave no improvement in this case, so they are not shown. For comparison, the values of the asymptotic expansion are also shown.

The cases above "constrained" the derivative of the singularly perturbed "fast" variable, y. Usually we cannot isolate this variable. To see the effect of having a different variable set, we transform x and y into

$$u = x + y, \tag{4.4}$$

$$v = y - x \tag{4.5}$$

and work with u and v assuming that we do not know this transformation. (We chose this transformation because in some sense it puts equal parts of the slow and fast variables, x and y, in u and v, illustrating the fact that we need only know variables that *parameterize* the slow

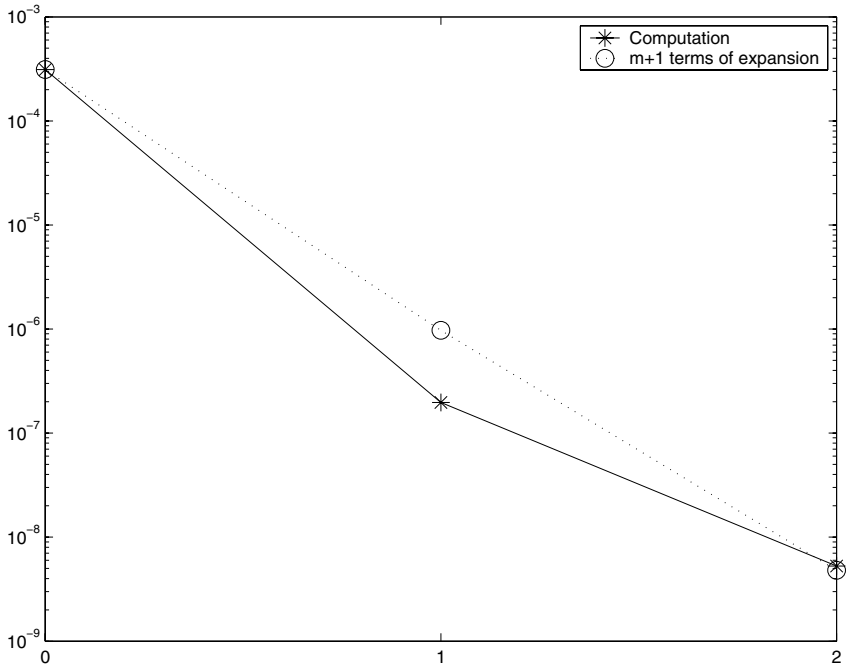

Figure 4. *Log error versus m for $\epsilon = 0.01$. Asterisk is computation with small H, and circle is asymptotic expansion.*

manifold (u in this case)—not ones that are in some sense dominated by the slow manifold.) We assumed that we were given a value of u and computed the approximation to $v(0)$ using our method. We chose a value for u corresponding to the correct solution with $x = 1$ used above. This was run with $h = \epsilon/10$ and $\epsilon = 0.1$, in both cases with the large H choice. In Figure 5 we show the log error in each case. Only $m = 0, \ldots, 2$ are shown because larger values of m yielded no improvement. The errors are the L_2 norms of the differences between the computed solution in x, y coordinates and the known (to ten digits) true solution.

4.2. A simplified hydrogen-oxygen reaction system. We will use an example from Lam and Goussis [25]. It contains seven radicals, O_2, H, OH, O, H_2, H_2O, and HO_2 which we will group in that order as the vector $\mathbf{y}$. The differential equations are

$$\begin{aligned}
\frac{dy_1}{dt} &= -k_{1f}y_1y_2 + k_{1b}y_3y_4 + k_{4f}y_3y_7 - \mu k_{5f}y_1y_2,\\
\frac{dy_2}{dt} &= -k_{1f}y_1y_2 + k_{1b}y_3y_4 + k_{2f}y_4y_5 - k_{2b}y_2y_3 + k_{3f}y_3y_5 - k_{3b}y_2y_6 - \mu k_{5f}y_1y_2,\\
\frac{dy_3}{dt} &= k_{1f}y_1y_2 - k_{1b}y_3y_4 + k_{2f}y_4y_5 + k_{2b}y_2y_3\\
&\quad - k_{3f}y_3y_5 + k_{3b}y_2y_6 - k_{4f}y_3y_7 - 2k_{8f}y_3^2 + 2k_{8b}y_4y_6,\\
\frac{dy_4}{dt} &= k_{1f}y_1y_2 - k_{1b}y_3y_4 - k_{2f}y_4y_5 + k_{2b}y_2y_3 + k_{8f}y_3^2 - k_{8b}y_4y_6,\\
\frac{dy_5}{dt} &= -k_{2f}y_4y_5 + k_{2b}y_2y_3 - k_{3f}y_3y_5 + k_{3b}y_2y_6,
\end{aligned}$$

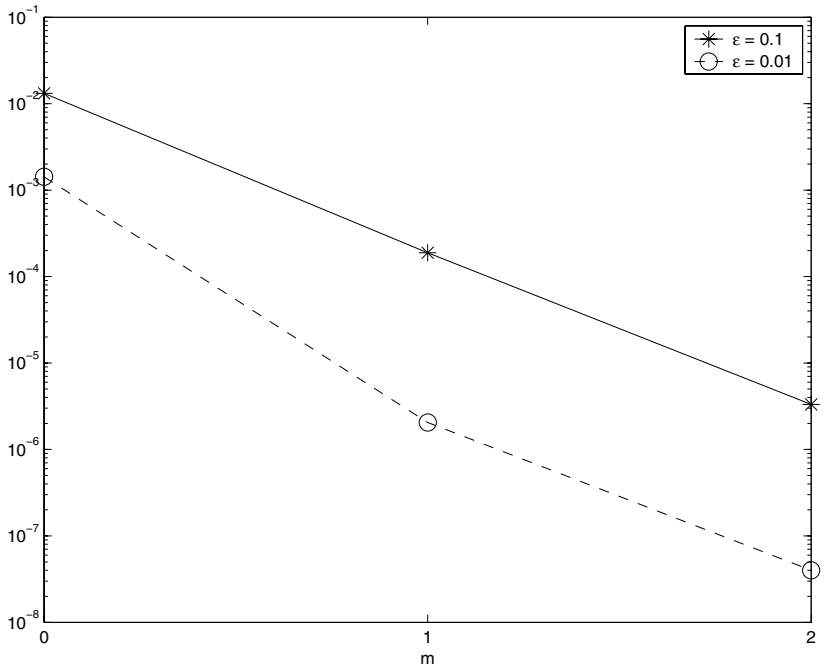

Figure 5. *Log error versus m for mixed observable case.*

$$\frac{dy_6}{dt} = k_{3f}y_3y_5 - k_{3b}y_2y_6 + k_{4f}y_3y_7 + k_{8f}y_3^2 - k_{8b}y_4y_6,$$
$$\frac{dy_7}{dt} = -k_{4f}y_3y_7 + \mu k_{5f}y_1y_2.$$

The values of the coefficients are taken from the cited paper and shown in Table 1. The parameter μ represents pressure and is 4.5×10^{-4}. The differential system has two constants of integration representing mass balance for oxygen and hydrogen atoms, so it is really a five-dimensional system. The eigenvalues of a local linearization in the region of operation for this example are approximately as shown in Table 2. From these we see that after an interval of order one millisecond, the system is effectively one-dimensional.

In the test below we chose y_5 (O_2) as the observable variable. The system was run from the starting conditions given in [25] until $t = 6.41 \times 10^{-4}$ (one of the reporting times in their paper). Then our process was applied, fixing y_5 to its current value, and choosing all other variables so that their $(m+1)$st forward difference is zero. To emulate a "legacy code" situation, we integrated the equations using a standard integrator (**ode23s** in MATLAB) over $m+1$ intervals of length H. In this example, we used $H = 1 \times 10^{-5}$. The relative and absolute error tolerances for **ode23s** were set to 10^{-12} and 10^{-14}, respectively. To maintain the mass balance relationship, radical concentrations of H and OH are computed directly from the mass balance relations. (These two were chosen because their concentrations remain reasonably nonzero. If a radical whose concentration gets close to zero is used, there is some danger of roundoff errors causing the concentration to become negative. This will often make the system unstable as well as physically unrealistic.)

Table 1
Reaction rates for hydrogen oxygen system.

i	k_{if}	k_{ib}
1	1.0136×10^{12}	1.1007×10^{13}
2	3.5699×10^{12}	3.2105×10^{12}
3	4.7430×10^{12}	1.8240×10^{11}
4	6.0000×10^{13}	
5	6.2868×10^{15}	
8	6.5325×10^{12}	3.1906×10^{11}

Table 2
Eigenvalues.

-2.5×10^{6}
-1.4×10^{6}
-4.0×10^{4}
-8.3×10^{3}
-4.0×10^{-3}

Table 3
Radical concentrations prior to constraining to slow manifold.

y_1	$4.2783465727 \times 10^{-13}$
y_2	$3.9878034748 \times 10^{-8}$
y_3	$1.3883748623 \times 10^{-10}$
y_4	$1.1300067412 \times 10^{-11}$
y_5	$4.4019256520 \times 10^{-7}$
y_6	$3.9848995981 \times 10^{-8}$
y_7	$5.3981503775 \times 10^{-15}$

Table 4
Difference between starting y_4 and constrained value.

m	Difference
0	$-2.0767748211 \times 10^{-17}$
1	$-2.0157837979 \times 10^{-17}$
2	$-2.0157711737 \times 10^{-17}$
3	$-2.0157697197 \times 10^{-17}$
4	$-2.0157429383 \times 10^{-17}$

The procedure was run with $m = 0, 1, \ldots, 4$. The starting values of the concentrations were as shown in Table 3. (These are given for the sake of completeness should anyone wish to compare them with our results.) The constrained results for each m differ from the starting value by no more than 10^{-14} and from each other by less, so they are not particularly revealing to study directly. (Larger changes from the starting value could be obtained by starting from a different point with the same mass balance values but would not give any further insight.) Since it is difficult to compute the slow manifold (often a problem when real examples are used) we do not have a good way to characterize errors, but we can examine two features to see if the method appears to work.

In Table 4 we show the differences between the starting value and the constrained value

Table 5
Norm ratio $||Jv||/||v||$ at constrained solution.

m	R
0	4.44973316×10^5
1	5.85785391×10^1
2	6.18695075×10^1
3	2.06227270×10^2
4	1.50929245×10^2

of y_4 for each order of constraint. We see that these differences show signs of "converging," but this is certainly not irrefutable evidence of convergence. As a second test, we considered the relationship of the local derivative of the solution at the result of the constraint iteration. If it were very far from the slow manifold, we would expect it to have relatively large components of the eigenvectors corresponding to the large eigenvalues. (In general, even on the slow manifold it will not have zero components in the large eigendirections except for linear problems.) To estimate the amount of large eigencomponents present we computed $v = dy/dt = f(y)$ and the local Jacobian $J = \partial f/\partial y$ at the solution, y, of the constraint iteration. Then we computed the norm ratio

$$R = \frac{||Jv||}{||v||}.$$

Its upper bound is the magnitude of the largest eigenvalue, and a value significantly less than this is an indicator that u is deficient in the largest eigendirection. Hence, the norm ratio gives some indication of the amount of the largest eigencomponents present. Its values are shown in Table 5.

Since the largest eigenvalue is around 2.5×10^6, it is clear that the $m = 0$ case contains no more than around 20% of the large eigendirections, but this is drastically reduced for $m = 1$. From this particular starting point and choice of H, larger m gave no further improvement, but other choices of starting points or H yield norm ratios that reduce with each m increase, although by relatively small amounts.

4.3. A five-dimensional system. Because of the difficulty of determining whether the method is getting better approximations as m increases, our final example is an artificial nonlinear five-dimensional problem with a two-dimensional attracting invariant manifold. We start with the loosely coupled differential equations

$$\frac{dx_1}{dt} = -x_2,$$

$$\frac{dx_2}{dt} = x_1,$$

$$\frac{dw}{dt} = L(x_1^2 + x_2^2 - w),$$

$$\frac{du_1}{dt} = \beta_1 u_1 + u_1^2,$$

$$\frac{du_2}{dt} = \beta_2 u_2 + u_2^2$$

with $L = 1000$, $\beta_1 = 800$, and $\beta_2 = 1200$. The solutions of these are

$$\begin{aligned} x_1 &= A\cos(t+\phi), \\ x_2 &= A\sin(t+\phi), \\ w &= A^2(1+be^{-Lt}), \\ u_i &= -\beta_i/(1+c_i e^{-\beta_i t}). \end{aligned}$$

For any starting conditions, $w \to x_1^2(0)+x_2^2(0)$, and, if $u_i(0)$ is chosen appropriately, $u_i \to -\beta_i$ and the system goes to a closed orbit at each point of which the eigenvalues of the system Jacobian are $\pm i$, -800, $-1{,}000$, and $-1{,}200$. Thus $w = x_1^2 + x_2^2$ is an attracting, two-dimensional invariant manifold. The above system is now subject to the unitary linear transformation given by $y = Qv$, where $v = [x^T, w, u^T]^T$ and Q is

$$Q = \frac{1}{5}\begin{bmatrix} -3 & 2 & 2 & 2 & 2 \\ 2 & -3 & 2 & 2 & 2 \\ 2 & 2 & -3 & 2 & 2 \\ 2 & 2 & 2 & -3 & 2 \\ 2 & 2 & 2 & 2 & -3 \end{bmatrix}.$$

As in the first example, this is chosen to "mix up" the slow and fast components. We applied the constraint method using y_1 and y_2 as the fixed "observables." They were set to the values -791.2 and -792.2, respectively. The subspace $y_1 = -791.2$, $y_2 = -792.2$ intersects with the invariant manifold at four points (the defining system is a pair of quadratic equations). The intersection in the neighborhood of the solution has the values

$$x_1 = -3.559434800714, \qquad x_2 = -2.559434800714.$$

Integration of y was performed using forward Euler with step size h for $m+1$ steps, and iterating until the $(m+1)$st forward difference was less than a tolerance with $m = 0$, 1, and 2. The norms of the residuals (differences between the constraint calculations and the known solution) are shown for three values of h in Figure 6. We see that the residuals decrease by just over two orders of magnitude for each increase in m until the error from the step size h begins to dominate. In a practical application, it might be wise to use a large h to get an initial approximation and then refine with a smaller h (although it might still be wise to use some convergence acceleration technique).

5. Discussion and conclusions. We presented a "computational wrapper" approach for the approximation of a low-dimensional slow manifold using a legacy simulator. The approach effectively constitutes a protocol for the design and processing of brief computational experiments with the legacy simulator, which converge to an approximation of the slow manifold; in the spirit of CSP, one can think of it as "singular perturbation through computational experiments." It is interesting that, if one could initialize a laboratory experiment at will,

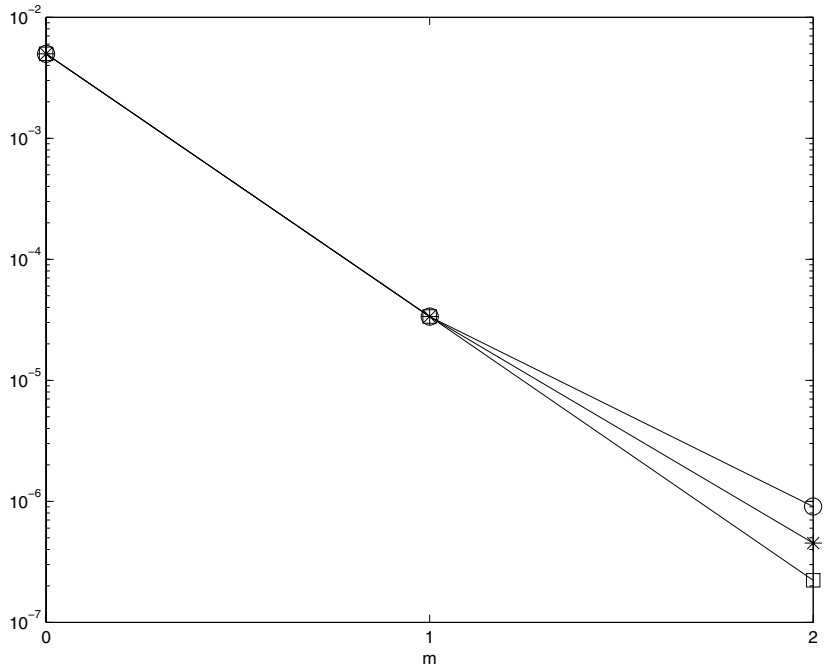

Figure 6. *Log norm of residual versus m for five-dimensional example. Circle, $h = 0.0004$, asterisk $h = 0.0002$, square $h = 0.0001$.*

our "computer experiment" protocol could become a laboratory experiment protocol for the approximation of a slow manifold.

The approach can be enhanced in many ways; we already mentioned the possible use of matrix-free fixed point algorithms for the acceleration of its convergence. Here we used the "simplest possible" estimation (through finite difference interpolation) of the trajectory from the results of the simulation. Better estimation techniques (e.g., maximum likelihood) can be linked with the data processing part of the approach; this will be particularly important when the results of the detailed integration are noisy, as will be the case in the observation of the evolution of statistics of complex evolving distributions.

It is also important to notice that, upon convergence of the procedure, one can implement a matrix-free, time-stepper–based computational approximation of the leading eigenvalues of the local linearization of the dynamics (e.g., through a time-stepper–based Arnoldi procedure; see [5, 31]). As the evolution progresses, or as the parameters change, this test can be used to adaptively adjust the local *dimension* of the slow manifold; we can detect whether a slow mode is starting to become fast, or when a mode that used to be fast is now becoming slow. The *eigenvectors* of these modes constitute good *additional observables* for the parameterization of a faster slow manifold. One of the important features of the approach is that one does not need to a priori know what the so-called slow variables are—*any* set of observables that can parameterize the slow manifold (i.e., over which the manifold is the graph of a function) can be used for our approach. If data analysis [16, 32] suggests good observables that are nonlinear combinations of the obvious state variables, the approach can still be implemented;

the knowledge of good order parameters can thus be naturally incorporated in this approach.

Overall, this approach provides us with a *good initial condition* for the full problem, consistent with a set of observables—an initial condition that lies close to the slow manifold, sometimes referred to as a "mature" or "bred" initial condition. Such initial conditions are essential for the implementation of equation-free algorithms—algorithms that solve the reduced problem *without* ever deriving it in closed form [4, 27, 29]. Indeed, short bursts of appropriately initialized simulations can be used to perform long-term prediction (projective and coarse projective integration) for the reduced problem, its stability and bifurcation analysis, as well as tasks like control and optimization. We expect this approach to become a vital component of the lifting operator in equation-free computation.

REFERENCES

[1] R. Aris, *Mathematical Modeling: A Chemical Engineer's Perspective*, Academic Press, San Diego, 1999.

[2] M. Bodenstein, *Eine Theorie der photochemischen Reaktionsgeschwindigkeiten*, Z. Phys. Chem., 85 (1913), pp. 329–397.

[3] P. N. Brown, A. C. Hindmarsh, and L. R. Petzold, *Consistent initial condition calculation for differential-algebraic systems*, SIAM J. Sci. Comput., 19 (1998), pp. 1495–1512.

[4] L. Chen, P. G. Debenedetti, C. W. Gear, and I. G. Kevrekidis, *From molecular dynamics to coarse self-similar solutions: A simple example using equation-free computation*, J. Non-Newtonian Fluid Mech., 120 (2004), pp. 215–223.

[5] K. N. Christodoulou and L. E. Scriven, *Finding leading modes of a viscous free surface flow: An asymmetric generalized eigenproblem*, J. Sci. Comput., 3 (1988), pp. 355–406.

[6] P. Constantin, C. Foias, B. Nicolaenko, and R. Temam, *Integral Manifolds and Inertial Manifolds for Dissipative Partial Differential Equations*, Springer-Verlag, New York, 1988.

[7] N. Fenichel, *Geometric singular perturbation theory for ordinary differential equations*, J. Differential Equations, 31 (1979), pp. 53–98.

[8] C. W. Gear and I. G. Kevrekidis, *Projective methods for stiff differential equations: Problems with gaps in their eigenvalue spectrum*, J. Sci. Comput., 24 (2004), pp. 1091–1106; longer version at physics/0312094 at http://arXiv.org.

[9] C. W. Gear and I. G. Kevrekidis, *Constraint-defined manifolds: A legacy-code approach to low-dimensional computation*, J. Sci. Comput., 25 (2006), to appear; also at physics/0312094 at http://arXiv.org.

[10] C. W. Gear and I. G. Kevrekidis, *Computing in the past with forward integration*, Phys. Lett. A, 321 (2004), pp. 335–343; also at nlin.CD/0302055 at http://arXiv.org.

[11] C. W. Gear, I. G. Kevrekidis, and C. Theodoropoulos, *"Coarse" integration/bifurcation analysis via microscopic simulators: Micro-Galerkin methods*, Comp. Chem. Engng., 26 (2002), pp. 941–963.

[12] S. S. Girimaji, *Reduction of large dynamical systems by minimization of evolution rate*, Phys. Rev. Lett., 82 (1999), pp. 2282–2285.

[13] A. N. Gorban and I. V. Karlin, *Geometry of irreversibility: The film of nonequilibrium states*, in Invariant Manifolds for Physical and Chemical Kinetics, A. N. Gorban and I. V. Karlin, eds., Lecture Notes in Phys. 660, Springer-Verlag, Berlin, 2005, pp. 325–366.

[14] A. N. Gorban, I. V. Karlin, and A. Yu. Zinovyev, *Constructive methods of invariant manifolds for kinetic problems*, Phys. Rep., 396 (2004), pp. 197–403; also at cond-mat/0311017 at http://arXiv.org.

[15] J. Guckenheimer and P. Holmes, *Nonlinear Oscillations, Dynamical Systems and Bifurcations of Vector Fields*, Springer-Verlag, New York, 1983.

[16] I. T. Jolliffe, *Principal Component Analysis*, Springer-Verlag, New York, 1986.

[17] H. G. Kaper and T. J. Kaper, *Asymptotic analysis of two reduction methods for systems of chemical reactions*, Phys. D, 65 (2002), pp. 66–93.

[18] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, Frontiers Appl. Math. 16, SIAM, Philadelphia, 1995.

[19] C. T. Kelley, I. G. Kevrekidis, and L. Qiao, *Newton–Krylov solvers for time-steppers*, SIAM J. Appl. Dyn. Syst., submitted; also at math.DS/0404374 at http://arXiv.org.

[20] I. G. Kevrekidis, C. W. Gear, J. M. Hyman, P. G. Kevrekidis, O. Runborg, and K. Theodoropoulos, *Equation-free coarse-grained multiscale computation: Enabling microscopic simulators to perform system-level tasks*, Commun. Math. Sci., 1 (2003), pp. 715–762; original version at physics/0209043 at http://arXiv.org.

[21] I. G. Kevrekidis, C. W. Gear, and G. Hummer, *Equation-free: The computer-aided analysis of complex multiscale systems*, AIChE J., 50 (2004), pp. 1346–1354.

[22] H.-O. Kreiss, *Problems with different time scales*, in Multiple Time Scales, J. H. Brackbill and B. I. Cohen, eds., Academic Press, New York, 1985, pp. 29–57.

[23] S. H. Lam, *Using CSP to understand complex chemical kinetics*, Combust. Sci. Technol., 89 (1993), pp. 375–404.

[24] S. H. Lam and D. A. Goussis, *The CSP method for simplifying chemical kinetics*, Int. J. Chem. Kin., 26 (1994), pp. 461–486.

[25] S. H. Lam and D. A. Goussis, *Understanding complex chemical kinetics with computational singular perturbation*, in 22nd Symposium on Combustion, The Combustion Institute, Pittsburgh, PA, 1988, pp. 931–941.

[26] U. Maas and S. B. Pope, *Simplifying chemical kinetics: Intrinsic low-dimensional manifolds in composition space*, Comb. Flame, 88 (1992), pp. 239–264.

[27] A. G. Makeev, D. Maroudas, A. Z. Panagiotopoulos, and I. G. Kevrekidis, *Coarse bifurcation analysis of kinetic Monte Carlo simulations: A lattice gas model with lateral interaction*, J. Chem. Phys., 117 (2002), pp. 8229–8240.

[28] R. E. O'Malley, *Singular Perturbation Methods for Ordinary Differential Equations*, Appl. Math. Sci. 89, Springer-Verlag, Berlin, 1991.

[29] R. Rico-Martinez, C. W. Gear, and I. G. Kevrekidis, *Coarse projective KMC integration: Forward/reverse initial and boundary value problems*, J. Comput. Phys., 196 (2004), pp. 474–489.

[30] G. M. Shroff and H. B. Keller, *Stabilization of unstable procedures: The recursive projection method*, SIAM J. Numer. Anal., 30 (1993), pp. 1099–1120.

[31] C. Siettos, M. D. Graham, and I. G. Kevrekidis, *Coarse Brownian dynamics for nematic liquid crystals: Bifurcation, projective integration and control via stochastic simulation*, J. Chem. Phys., 118 (2003), pp. 10149–10157.

[32] A. J. Smola, O. L. Mangasarian, and B. Schoelkopf, *Sparse kernel feature analysis*, in 24th Annual Conference of Gesselschaft für Klassifikation (University of Passau, Passau, Germany, 2000); Technical Report 99-04, Data Mining Institute, Computer Science Department, University of Wisconsin, Madison, WI, 1999.

[33] R. Temam, *Infinite Dimensional Dynamical Systems in Mechanics and Physics*, Springer-Verlag, New York, 1988.

[34] K. Theodoropoulos, Y.-H. Qian, and I. G. Kevrekidis, *"Coarse" stability and bifurcation analysis using timesteppers: A reaction diffusion example*, Proc. Natl. Acad. Sci., 97 (2000), pp. 9840–9843.

[35] T. Turanyi, A. S. Tomlin, and M. J. Pilling, *On the error of the quasi-steady state approximation*, J. Phys. Chem., 97 (1993), pp. 163–172.

[36] A. Zagaris, H. G. Kaper, and T. J. Kaper, *Analysis of the computational singular perturbation reduction method for chemical kinetics*, J. Nonlinear Sci., 14 (2004), pp. 59–91.