Math. Nachr. **278,** No. 12–13, 1629 – 1642 (2005) / **DOI** 10.1002/mana.200410328

Two perspectives on reduction of ordinary differential equations

Antonios Zagaris[*1], **Hans G. Kaper**[**2], and **Tasso J. Kaper**[***1]

[1] Dept. of Mathematics and Center for BioDynamics, Boston University, Boston, Massachusetts, USA
[2] Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, Illinois, USA

Received 12 November 2004, revised 26 January 2005, accepted 15 March 2005
Published online 8 September 2005

Key words Nonlinear differential equations, kinetic equations, multiple time scales, dimension reduction, slow manifold, normal form, computational singular perturbation, zero derivative principle

MSC (2000) 34C20, 34E13, 34E15, 80A30, 80A25, 92C45

Dedicated to the memory of F. V. Atkinson

This article is concerned with general nonlinear evolution equations $x' = g(x)$ in $\mathbf{R}^N$ involving multiple time scales, where fast dynamics take the orbits close to an invariant low-dimensional manifold and slow dynamics take over as the state approaches the manifold. Reduction techniques offer a systematic way to identify the slow manifold and reduce the original equation to an autonomous equation on the slow manifold. The focus in this article is on two particular reduction techniques, namely, computational singular perturbation (CSP) proposed by Lam and Goussis [Twenty-Second Symposium (International) on Combustion, The University of Washington, Seattle, Washington, August 14–19, 1988 (The Combustion Institute, Pittsburgh, 1988), pp. 931–941] and the zero-derivative principle (ZDP) proposed recently by Gear and Kevrekidis [Constraint-defined manifolds: A legacy-code approach to low-dimensional computation, SIAM J. Sci. Comput., to appear]. It is shown that the tangent bundle to the state space offers a unifying framework for CSP and ZDP. Both techniques generate coordinate systems in the tangent bundle that are natural for the approximation of the slow manifold. Viewed from this more general perspective, both CSP and ZDP generate, at each iteration, approximate normal forms for the system under examination.

1 Introduction

This article is concerned with reduction techniques for systems of nonlinear ordinary differential equations (ODEs) involving fast and slow dynamics, where "fast" dynamics take the orbits close to an invariant low-dimensional manifold and "slow" dynamics take over as the state approaches the manifold.

If the dimension of the slow manifold is much less than that of the state space, one can achieve significant savings in the study of the long-term behavior of such systems by considering the dynamics on the slow manifold. Many techniques have been proposed for this purpose, especially in the chemical kinetics literature; see the references in [8, 22, 23]. Generally, they involve the construction of a coordinate system where the slow manifold has a simple functional representation—the ideal system being one where a number of components (the "fast" components) of the state vector are identically zero, as in the Fenichel normal form [7]. In this article we focus on two such techniques, namely, computational singular perturbation (CSP) and the zero-derivative principle (ZDP). We restrict ourselves to a discussion of the principles of these techniques; details that may vary from one implementation to another, although important, fall outside the scope of the present article.

Computational singular perturbation was first proposed by Lam and Goussis [13] and subsequently developed in [5, 12, 14]. It is being used widely, for example, in combustion modeling [6, 15, 16, 17, 20, 21] and atmospheric science [18]. It has been analyzed mathematically, and its approximation properties have been established rigorously for fast-slow systems [22, 23]. In its original formulation, CSP is essentially an iterative technique to generate a coordinate system for the vector field where the system of differential equations reduces to normal

* e-mail: azagaris@math.bu.edu, Phone: +1 617 353 2560, Fax: +1 617 353 8100
** Corresponding author: e-mail: kaper@mcs.anl.gov, Phone: +1 630 252 7160, Fax: +1 630 252 5986
*** e-mail: tasso@math.bu.edu, Phone: +1 617 353 2560, Fax: +1 617 353 8100

form. Each iteration involves two steps, one to approximate the slow manifold and another to approximate the tangent spaces of the fast fibers to the slow manifold [23]. Since the emphasis in the present article is on the approximation of the slow manifold, we restrict the discussion to a one-step CSP, where the second step is omitted. This variant of CSP was introduced in [22].

The zero-derivative principle is based on the bounded-derivative method of Kreiss [11] and is due to Gear and Kevrekidis [2]. Its principle was developed in the context of the equation-free method for multiscale computations [3, 9, 10]. Its approximation properties have been analyzed recently in [4]. It yields successive approximations to the slow manifold by setting derivatives of successively higher order of the subset of "fast variables" of the state variables equal to zero; see [3, 4] for details.

In this article, we establish two results for CSP and one for ZDP, showing at the same time how the two techniques are related. For CSP, we first show that the update rules, by which the fast and slow components of the vector field become decoupled, can be streamlined. We find that, along with the original vector field and a basis to initiate the iterative process, only the Jacobian of the current vector field is needed to compute the next iterate of the vector field. Second, we show that the update rule for the vector field can induce a sequence of coordinate changes in the state space (the space of the dependent variables in the ODEs). Thus, the CSP reduction in the space of the vector field induces a corresponding reduction in the state space (via a sequence of successively improved approximations to the slow manifold), and this induced reduction is similar to that attained by ZDP. For ZDP, we establish an analogous result, but one that goes in the opposite direction because ZDP works directly in the state space. In particular, we show that the iteration scheme for finding successive approximations of the slow manifold in the state space naturally induces a sequence of local bases and coordinate changes in the space of the vector fields. This induced sequence, in turn, decouples the dynamics of the fast and slow components of the vector field, in much the same way CSP functions.

Therefore, although CSP and ZDP approach the problem of approximating slow manifolds from opposite directions, they are similar when considered as algorithms to generate coordinate systems in the tangent bundle of the ODE, which provides the framework to simultaneously track the dynamics in the state space and in the space of the vector field. In fact, from the joint perspective offered by the tangent bundle, CSP and ZDP can also be understood in a more traditional manner as generating, at each iteration, approximate and effectively equivalent normal forms for the system under examination.

This article is organized as follows. In Section 2, we discuss various coordinates associated with Eq. (2.1) and their dynamics. In Section 3, we present some principles underlying reduction techniques. Details for CSP are given in Section 4 and for ZDP in Section 5. A brief Section 6 summarizes our results.

2 Coordinates and their dynamics

We consider a general system of nonlinear ordinary differential equations

$$x' \;=\; g(x)\,, \tag{2.1}$$

for a function x with values in $\mathbf{R}^N$. We refer to $\mathbf{R}^N$ as the state space and think of $x(t)$ as describing the state of a physical system at time t. The function g represents a smooth vector field in $\mathbf{R}^N$, and a prime $'$ denotes differentiation with respect to t. We assume that there exists a low-dimensional manifold in $\mathbf{R}^N$ that is slow, invariant, and normally attracting. We denote this manifold by $\mathcal{M}$ and its dimension by N_s, where $N = N_f + N_s$ and $N_s < N$.

2.1 Standard coordinates

A useful framework for the study of Eq. (2.1) is the tangent bundle $\mathcal{T}\,\mathbf{R}^N$ of $\mathbf{R}^N$—that is, the collection of elements (x, v), where both x and v are arbitrary elements of $\mathbf{R}^N$ [19]. To introduce a frame of reference in $\mathcal{T}\,\mathbf{R}^N$, we must first select a basis in $\mathbf{R}^N$. The standard basis of $\mathbf{R}^N$ consists of the unit vectors $\{e_1, \ldots, e_N\}$, collectively represented by the unit matrix I. In this basis, the vectors x and $g(x)$ are represented by the N-tuples of their coordinates,

$$x \;=\; \big(x^1, \ldots, x^N\big)^{\mathrm{t}} \in \mathbf{R}^N\,, \quad g(x) \;=\; \big(g^1(x), \ldots, g^N(x)\big)^{\mathrm{t}} \in \mathbf{R}^N.$$

The standard basis of $\mathcal{T}\,\mathbf{R}^N$ is a combination of the standard basis of $\mathbf{R}^N$ with itself, so the ordered pair $(x, g(x))$ is represented by

$$(x, g(x)) \;=\; \big(x^1, \ldots, x^N, g^1(x), \ldots, g^N(x)\big)^{\mathrm{t}} \in \mathcal{T}\,\mathbf{R}^N.$$

To find the equations governing the evolution of $(x, g(x))$ in $\mathcal{T}\,\mathbf{R}^N$, we first consider the dynamics of (the coordinates of) x and $g(x)$ separately. The evolution of x in $\mathbf{R}^N$ is governed by Eq. (2.1). Using this equation and the chain rule of differentiation, we obtain the evolution equation for $g(x)$ in $\mathbf{R}^N$,

$$g' \;=\; (D_x g)g\,, \tag{2.2}$$

where $D_x g \;=\; (\partial g/\partial x)$ is the Jacobian of g. Combining these results, we obtain the evolution equation for $(x, g) = (x, g(x))$ in $\mathcal{T}\,\mathbf{R}^N$,

$$(x, g)' \;=\; (g, (D_x g)g)\,. \tag{2.3}$$

This equation tracks both the change of the state variable x with time and the concurrent change of the vector field g along orbits $x(t)$. It provides a unifying framework for the study of reduction methods that have been developed in the context of either the space of vector fields (as is the case for CSP, Section 4) or the space of state variables (as is the case for ZDP, Section 5). Of course, when it comes to the practical implementation, it is sufficient to work in only one context, and one can solve either Eq. (2.1) or Eq. (2.2).

2.2 General coordinates

In Eq. (2.3), both the state variable and the vector field are represented in the standard basis of $\mathbf{R}^N$. In this section, we consider a general coordinate system in $\mathbf{R}^N$, which is related to the standard one via a smooth map of $\mathbf{R}^N$ onto itself, and derive the corresponding evolution equations for the state variable and the vector field in the tangent bundle.

Let the state of the system at time t be represented by a new vector, $y(t) = y(x(t))$; inversely, $x(t) = x(y(t))$. The components of y are

$$\big(y^1, \ldots, y^N\big)^{\mathrm{t}} \;=\; \big(y^1(x), \ldots, y^N(x)\big)^{\mathrm{t}} \in \mathbf{R}^N. \tag{2.4}$$

If the new coordinates are curvilinear (and they will be in most of the systems one wants to reduce), then they are not associated with any fixed global vector basis that can be used to express $g(x)$ for all x. Rather, at each point $y = y(x)$ of the state space, they are associated with a *local* basis that generally varies from point to point and that consists of the columns of the Jacobian $(D_y x)(y)$. In that basis, we have

$$g \;=\; (D_y x)f\,, \tag{2.5}$$

for some f. Differentiation of both members of $y(x(y)) = y$ with respect to y yields the identity $(D_x y)(D_y x) = I$, which we can use to invert Eq. (2.5),

$$f \;=\; (D_x y)g\,. \tag{2.6}$$

The evolution equation for y follows from Eqs. (2.1), (2.4), and (2.6),

$$y' \;=\; f(y)\,. \tag{2.7}$$

This equation is the analog of Eq. (2.1). The analog of Eq. (2.2) is

$$f' \;=\; (D_y f)f\,. \tag{2.8}$$

We express the Jacobian $D_y f$ in terms of the local basis and its dual. Using Eq. (2.6) and the chain rule, we have

$$D_y f \;=\; D_x((D_x y)g)(D_y x) \;=\; (D_x y)(D_x g)(D_y x) + \big(\big(D_x^2 y\big)g\big)(D_y x)\,.$$

Differentiation of both members of $(D_x y)(D_y x) = I$ with respect to t along solutions of Eq. (2.1) yields the identity $\big(\big(D_x^2 y\big)g\big)(D_y x) + (D_x y)(D_x(D_y x)g) = 0$, which we use to rewrite the expression for $D_y f$ as

$$D_y f \;=\; (D_x y)\,[(D_x g)(D_y x) - (D_x(D_y x))g]\,.$$

The expression inside the brackets is the Lie bracket of $D_y x$ (taken column by column) and g. (The Lie bracket of any two vector fields a and b is $[a,b] = (D_x b)a - (D_x a)b$.) Therefore, Eq. (2.8) can also be written as

$$f' \;=\; \Lambda f\,, \tag{2.9}$$

where

$$\Lambda \;=\; \Lambda(y,f) \;=\; D_y f \;=\; (D_x y)\,[D_y x, g]\,. \tag{2.10}$$

Combining Eqs. (2.7) and (2.9), we obtain the evolution equation for $(y,f) = (y, f(y))$ in $\mathcal{T}\,\mathbf{R}^N$,

$$(y,f)' \;=\; (f, \Lambda f)\,. \tag{2.11}$$

The task now is to construct a suitable basis for $\mathcal{T}\,\mathbf{R}^N$ where fast and slow variables can be distinguished. We address this issue in the following section.

3 Reduction to the slow manifold

An ideal coordinate system is one in which $\mathcal{M}$ is the locus of points where N_f components of the state vector are identically zero. If the state vector y, defined in Eq. (2.4), corresponds to such an ideal coordinate system, we can partition its components into two groups,

$$y \;=\; \begin{pmatrix} y^f \\ y^s \end{pmatrix}, \tag{3.1}$$

with $y^f = \big(y^1, \ldots, y^{N_f}\big)^{\mathrm{t}}$ and $y^s = \big(y^{N_f+1}, \ldots, y^N\big)^{\mathrm{t}}$, and define $\mathcal{M}$ implicitly by the equation

$$\mathcal{M} \;=\; \big\{x \in \mathbf{R}^N : y^f(x) = 0\big\} \tag{3.2}$$

or in parametric form by the equation

$$\mathcal{M} \;=\; \big\{x(0, y^s) : y^s \in \mathbf{R}^{N_s}\big\}\,. \tag{3.3}$$

Eq. (3.2), a condition in the state space, can be combined with the invariance of $\mathcal{M}$ to yield a condition in the tangent bundle. First, we note that the decomposition (3.1) induces decompositions for the local basis $D_y x$ and the coordinates f,[1]

$$D_y x \;=\; (D_f x, D_s x)\,, \quad f \;=\; \begin{pmatrix} f^f \\ f^s \end{pmatrix} \;=\; \begin{pmatrix} \big(D_x y^f\big)g \\ (D_x y^s)g \end{pmatrix}. \tag{3.4}$$

Given these decompositions, we can rewrite Eq. (2.5) as

$$g \;=\; (D_f x) f^f + (D_s x) f^s\,. \tag{3.5}$$

This equation represents a decomposition of the vector field into a component in the subspace spanned by the columns of $D_f x$ and a component in the subspace spanned by the columns of $D_s x$. Because $\mathcal{M}$ is invariant, the vector field $g(p)$ at any point $p \in \mathcal{M}$ lies in the tangent space $\mathcal{T}_p\mathcal{M}$. Furthermore, since the variables y^s parameterize $\mathcal{M}$ (see Eq. (3.3)), $\mathcal{T}_p\mathcal{M} = \mathrm{span}(\mathrm{cols}(D_s x)(p))$. Therefore, $g(p) \in \mathrm{span}(\mathrm{cols}(D_s x)(p))$, and $\mathcal{M}$ is the locus of points where g has no component along $\mathrm{span}(\mathrm{cols}(D_f x))$,

$$\mathcal{M} \;=\; \big\{x \in \mathbf{R}^N : f^f(y(x)) = \big(\big(D_x y^f\big)g\big)(x) = 0\big\}\,. \tag{3.6}$$

[1] Throughout this section, we use the shorthand notation D_f for D_{y^f} and D_s for D_{y^s}.

This equation identifies $\mathcal{M}$ in terms of the coordinates f of the vector field.

In an ideal coordinate system, the fast and slow dynamics decouple naturally. Indeed, take any point $p \in \mathcal{M}$ and consider the decomposition

$$\mathcal{T}_p\mathbf{R}^N \;=\; \mathcal{T}_p\mathcal{F} \oplus \mathcal{T}_p\mathcal{M} \tag{3.7}$$

of $\mathcal{T}_p\mathbf{R}^N$, the tangent space to $\mathbf{R}^N$ at p, into the direct sum of the tangent space at p to the fast fiber through p, $\mathcal{T}_p\mathcal{F}$, and $\mathcal{T}_p\mathcal{M}$ (which was defined earlier). This decomposition is unique if $\mathcal{M}$ is normally hyperbolic and compact, because the two linear spaces intersect transversally—that is, they intersect at the single point p, and the sum of their dimensions equals the dimension N of $\mathbf{R}^N$. (Note that $\dim\mathcal{T}_p\mathcal{F} = N_f$ and $\dim\mathcal{T}_p\mathcal{M} = N_s$.) The decompositions (3.4) induce a decomposition for the operator Λ,

$$\Lambda \;=\; \begin{pmatrix} \Lambda_f^f & \Lambda_s^f \\ \Lambda_f^s & \Lambda_s^s \end{pmatrix}, \tag{3.8}$$

where

$$\Lambda_f^f \;=\; D_f f^f = \big(D_x y^f\big)[D_f x, g]\,, \quad \Lambda_s^f \;=\; D_s f^f = \big(D_x y^f\big)[D_s x, g]\,,$$
$$\Lambda_f^s \;=\; D_f f^s = (D_x y^s)[D_f x, g]\,, \quad \Lambda_s^s \;=\; D_s f^s = (D_x y^s)[D_s x, g]\,.$$

In [22, Lemma 3.1], we have shown that, if

$$\mathrm{span}(\mathrm{cols}(D_f x)(p)) \;=\; \mathcal{T}_p\mathcal{F} \quad \text{and} \quad \mathrm{span}(\mathrm{cols}(D_s x)(p)) \;=\; \mathcal{T}_p\mathcal{M}\,, \tag{3.9}$$

then Λ reduces to a block-diagonal form at p,

$$\Lambda_p \;=\; \begin{pmatrix} D_f f^f & 0 \\ 0 & D_s f^s \end{pmatrix}, \quad p \in \mathcal{M}\,.$$

Thus, a transformation $y = y(x)$ yielding a basis $D_y x(y)$ that respects Eq. (3.7) reduces Λ and separates the fast and slow subspaces along trajectories on $\mathcal{M}$.

In the same coordinate system, the dynamics of the components of y decouple. On $\mathcal{M}$, the fast variables satisfy the trivial equation, $\big(y^f\big)' = f^f = 0$, and the slow variables satisfy the equation

$$(y^s)' \;=\; f^s(0, y^s)\,. \tag{3.10}$$

This is an equation in $\mathbf{R}^{N_s}$. Since $N_s < N$, its dimension is less than the dimension of the original equation (2.1), and a reduction has been achieved.

4 Computational singular perturbation

Computational singular perturbation focuses on the components of the vector field g and is, in essence, an iterative process to reduce the system (2.1) to normal form. At each step of the iterative process, one constructs a transformation $y = y(x)$ and corresponding local bases $D_y x(y)$ and $D_x y(x)$ by requiring that the off-diagonal blocks of the then current approximation of the operator Λ be zero. An approximate slow manifold is then defined as the locus of all points in state space where the vector field lies entirely in the then current approximation of the slow subspace, as in Eq. (3.6).

The dimensions of the fast and slow subspaces are not known a priori because the fast and slow variables can, in general, not be identified without further knowledge of the system. To emphasize this fact, we will use the generic sub- and superscripts "1" and "2", rather than "f" and "s", as labels for potentially fast and slow categories. In practice, it is essential to monitor the progress of the algorithm and check whether the operator Λ indeed converges to block-diagonal form. As we have shown in our paper [23], asymptotic convergence of CSP is guaranteed for *fast-slow systems*, where the fast and slow variables are identified a priori and the ratio of their characteristic times is measured by an arbitrarily small parameter ε.

In Section 4.1 we describe CSP in some detail. Our main results are contained in Sections 4.2 and 4.3. In Section 4.2 we present a streamlined version of the CSP update rules, and in Section 4.3 we show how the CSP reduction of the vector field induces a reduction in the state space.

4.1 The principle of CSP

In the framework of CSP, it is customary to use the notation

$$A \;=\; D_y x\,, \quad B \;=\; D_x y \tag{4.1}$$

and consider A as a matrix of column vectors (vectors in $\mathbf{R}^N$) and B as a matrix of row vectors (functionals on $\mathbf{R}^N$). The matrices are partitioned following the partition (3.1) of y,

$$A \;=\; (A_1, A_2)\,, \quad B \;=\; \begin{pmatrix} B^1 \\ B^2 \end{pmatrix}, \tag{4.2}$$

where $A_1 = D_{y^1}x$, $A_2 = D_{y^2}x$, $B^1 = D_x y^1$, and $B^2 = D_x y^2$.

Initialized with two matrices $A^{(0)}$ and $B_{(0)}$, CSP generates two sequences of $N \times N$ matrices,

$$\big\{A^{(m)} : m = 0, 1, \dots\big\}\,, \quad \big\{B_{(m)} : m = 0, 1, \dots\big\}\,. \tag{4.3}$$

The matrices are partitioned as in Eq. (4.2) and normalized,

$$B_{(m)}A^{(m)} \;=\; \begin{pmatrix} B^1_{(m)} \\ B^2_{(m)} \end{pmatrix}\Big(A_1^{(m)}, A_2^{(m)}\Big) \;=\; \begin{pmatrix} I_1 & 0 \\ 0 & I_2 \end{pmatrix} \;=\; I\,, \quad m \;=\; 0\,,\, 1\,,\, \dots\,. \tag{4.4}$$

Here, I_1 and I_2 are the identity matrices of order N_1 and N_2, respectively.

After m iterations, the state of the system is represented by the vector $y_{(m)} = y_{(m)}(x)$, the current local basis and its dual are $A^{(m)} = D_{y_{(m)}}x$ and $B_{(m)} = D_x y_{(m)}$, respectively, and the vector field is expressed as $g = A^{(m)} f_{(m)}$ with $f_{(m)} = B_{(m)}g$. The evolution of $y_{(m)}$ is governed by the equation

$$y'_{(m)} \;=\; f_{(m)} \tag{4.5}$$

and that of $f_{(m)}$ by the equation

$$f'_{(m)} \;=\; \Lambda_{(m)} f_{(m)}\,, \tag{4.6}$$

where $\Lambda_{(m)} = D_{y_{(m)}} f_{(m)} = B_{(m)}[A^{(m)}, g]$; see Eqs. (2.7), (2.9), and (2.10). In block form,

$$\Lambda_{(m)} \;=\; \begin{pmatrix} \Lambda^{11}_{(m)} & \Lambda^{12}_{(m)} \\ \Lambda^{21}_{(m)} & \Lambda^{22}_{(m)} \end{pmatrix} \tag{4.7}$$

with

$$\Lambda^{11}_{(m)} \;=\; D_{y^1_{(m)}} f^1_{(m)} \;=\; B^1_{(m)}\big[A_1^{(m)}, g\big]\,, \qquad \Lambda^{12}_{(m)} \;=\; D_{y^2_{(m)}} f^1_{(m)} \;=\; B^1_{(m)}\big[A_2^{(m)}, g\big]\,,$$
$$\Lambda^{21}_{(m)} \;=\; D_{y^1_{(m)}} f^2_{(m)} \;=\; B^2_{(m)}\big[A_1^{(m)}, g\big]\,, \qquad \Lambda^{22}_{(m)} \;=\; D_{y^2_{(m)}} f^2_{(m)} \;=\; B^2_{(m)}\big[A_2^{(m)}, g\big]\,.$$

The update rule for the matrices A and B is

$$A^{(m+1)} \;=\; A^{(m)}\big(I - U_{(m)}\big)\big(I + L_{(m)}\big)\,, \quad m \;=\; 0\,,\, 1\,,\, \dots\,, \tag{4.8}$$
$$B_{(m+1)} \;=\; \big(I - L_{(m)}\big)\big(I + U_{(m)}\big)B_{(m)}\,, \quad m \;=\; 0\,,\, 1\,,\, \dots\,, \tag{4.9}$$

where $U_{(m)}$ and $L_{(m)}$ are nilpotent matrices,

$$U_{(m)} \;=\; \begin{pmatrix} 0 & \big(\Lambda^{11}_{(m)}\big)^{-1}\Lambda^{12}_{(m)} \\ 0 & 0 \end{pmatrix}, \quad L_{(m)} \;=\; \begin{pmatrix} 0 & 0 \\ \Lambda^{21}_{(m)}\big(\Lambda^{11}_{(m)}\big)^{-1} & 0 \end{pmatrix}. \tag{4.10}$$

The nonzero entries in $U_{(m)}$ and $L_{(m)}$ have been defined in such a way that $\Lambda_{(m+1)}$ is closer to block-diagonal form. Note that the normalization condition (4.4) is satisfied for all m.

As can be seen from Eqs. (4.8)–(4.9), CSP performs the updating in two steps; the first step involves the matrices U, the second the matrices L. In [22] and [23], we showed that the two steps play different roles and that, if one is interested solely in approximating the slow manifold, one may as well skip the second step. The slow manifold constructed with the so-called one-step CSP [22], which involves only the matrices U, has the same asymptotic accuracy as the slow manifold constructed with CSP. Since the purpose of this article is to compare CSP and ZDP, and the latter focuses entirely on the approximation of the slow manifold, we restrict the following discussion to the one-step CSP, where the update rule (4.8)–(4.9) is reduced to

$$A^{(m+1)} \;=\; A^{(m)}\big(I - U_{(m)}\big)\,, \quad B_{(m+1)} \;=\; \big(I + U_{(m)}\big)B_{(m)}\,, \quad m \;=\; 0\,,1\,,\ldots\,, \tag{4.11}$$

with $U_{(m)}$ given by Eq. (4.10).

The current approximations to the fast and slow components of the vector field are $f^1_{(m)} = B^1_{(m)}g$ and $f^2_{(m)} = B^2_{(m)}g$, respectively, and the current approximation to the slow manifold is defined as in Eq. (3.6),

$$\mathcal{K}^{(m)} \;=\; \big\{x \in \mathbf{R}^N : f^1_{(m)}(y(x)) = 0\big\}\,. \tag{4.12}$$

4.2 CSP update rules

In this section, we focus on the vector of "fast" coordinates of g and show that, if one is interested only in the slow manifold, $f^1_{(m+1)}$ may be found recursively in terms of $f^1_{(m)}$ without reference to the basis $A^{(m)}$, its inverse $B^{(m)}$, or the operator $\Lambda_{(m)}$.

Applying the update rule (4.11) recursively and using the fact that $U_{(i)}U_{(j)} = 0$ for all indices i and j, we obtain the expressions

$$A^{(m+1)} \;=\; A^{(0)}\left(I - \sum_{i=0}^{m} U_{(i)}\right), \quad B_{(m+1)} \;=\; \left(I + \sum_{i=0}^{m} U_{(i)}\right)B_{(0)}\,, \quad m \;=\; 0\,,1\,,\ldots\,. \tag{4.13}$$

The sum $\sum_{i=0}^m U_{(i)}$ has only one nonzero block, namely, $\sum_{i=0}^m \big(\Lambda^{11}_{(i)}\big)^{-1}\Lambda^{12}_{(i)}$ in the 12-position. In the remainder of this section, we will use the same notation, $\sum_{i=0}^m U_{(i)}$, to denote this nonzero block. Note that the 12-block is rectangular; its size is $N_1 \times N_2$.

We partition $A^{(0)}$ as in Eq. (4.2), $A^{(0)} = \big(A^{(0)}_1, A^{(0)}_2\big)$, where $A^{(0)}_1$ and $A^{(0)}_2$ are rectangular matrices of size $N \times N_1$ and $N \times N_2$, respectively, and rewrite the expressions (4.13) in the form

$$A^{(m+1)} \;=\; \Big(A^{(0)}_1,\, A^{(0)}_2 - A^{(0)}_1\big(\textstyle\sum_{i=0}^m U_{(i)}\big)\Big)\,, \qquad m \;=\; 0\,,1\,,\ldots\,, \tag{4.14}$$

$$B_{(m+1)} \;=\; \begin{pmatrix} B^1_{(0)} + \big(\sum_{i=0}^m U_{(i)}\big)B^2_{(0)} \\ B^2_{(0)} \end{pmatrix}, \qquad m \;=\; 0\,,1\,,\ldots\,. \tag{4.15}$$

The sum can be expressed solely in terms of CSP quantities current at the mth iteration. To see this, start from the definition $\Lambda_{(m)} = D_{y_{(m)}} f_{(m)}$, evaluate at $x = x(y_{(m)})$, and use the chain rule to replace the derivative with respect to $y_{(m)}$ by the derivative with respect to x. Since $D_{y_{(m)}} x = A^{(m)}$, we obtain

$$\Lambda_{(m)} \;=\; D_{y_{(m)}} f_{(m)} \;=\; \big(D_x f_{(m)}\big)\big(D_{y_{(m)}} x\big) \;=\; \big(D_x f_{(m)}\big)A^{(m)}\,.$$

If we now substitute $A^{(m)}$ in accordance with Eq. (4.14), we find

$$\begin{aligned} \Lambda^{11}_{(m)} \;&=\; \big(D_x f^1_{(m)}\big)A^{(m)}_1 \;=\; \big(D_x f^1_{(m)}\big)A^{(0)}_1\,, \\ \Lambda^{12}_{(m)} \;&=\; \big(D_x f^1_{(m)}\big)A^{(m)}_2 \;=\; \big(D_x f^1_{(m)}\big)A^{(0)}_2 - \big(D_x f^1_{(m)}\big)A^{(0)}_1 \sum_{i=0}^{m-1} U_{(i)}\,. \end{aligned}$$

Hence,

$$U_{(m)} \;=\; \big(\Lambda^{11}_{(m)}\big)^{-1}\Lambda^{12}_{(m)} \;=\; \Big[\big(D_x f^1_{(m)}\big)A^{(0)}_1\Big]^{-1}\big(D_x f^1_{(m)}\big)A^{(0)}_2 - \sum_{i=0}^{m-1} U_{(i)}\,.$$

It follows that $\sum_{i=0}^{m} U_{(i)}$ is given in terms of $f^1_{(m)}$ by

$$\sum_{i=0}^{m} U_{(i)} \;=\; \Big[\big(D_x f^1_{(m)}\big)A^{(0)}_1\Big]^{-1}\big(D_x f^1_{(m)}\big)A^{(0)}_2\,. \tag{4.16}$$

(Note that the matrix inside the square brackets is square ($N_1 \times N_1$), but its components are rectangular ($D_x f^1_{(m)}$ is $N_1 \times N$, $A^{(0)}_1$ is $N \times N_1$), so the inverse cannot be written as the product of the inverses of the component matrices.) Substituting this result into Eqs. (4.14)–(4.15), we obtain the following expressions for $A^{(m+1)}$ and $B_{(m+1)}$:

$$A^{(m+1)} = \left(A^{(0)}_1\,,\; A^{(0)}_2 - A^{(0)}_1\Big[\big(D_x f^1_{(m)}\big)A^{(0)}_1\Big]^{-1}\big(D_x f^1_{(m)}\big)A^{(0)}_2\right),\;\; m \;=\; 0\,,\,1\,,\,\ldots\,, \tag{4.17}$$

$$B_{(m+1)} = \begin{pmatrix} B^1_{(0)} + \Big[\big(D_x f^1_{(m)}\big)A^{(0)}_1\Big]^{-1}\big(D_x f^1_{(m)}\big)A^{(0)}_2 B^2_{(0)} \\ B^2_{(0)} \end{pmatrix}, \qquad m \;=\; 0\,,\,1\,,\,\ldots\,. \tag{4.18}$$

If we use the expression (4.18) in the identity $f^1_{(m+1)} = B^1_{(m+1)}g$, we obtain the update rule for the fast coordinates of the vector field,

$$\begin{aligned} f^1_{(m+1)} \;&=\; \left(B^1_{(0)} + \Big[\big(D_x f^1_{(m)}\big)A^{(0)}_1\Big]^{-1}\big(D_x f^1_{(m)}\big)A^{(0)}_2 B^2_{(0)}\right)g \\ &=\; \Big[\big(D_x f^1_{(m)}\big)A^{(0)}_1\Big]^{-1}\big(D_x f^1_{(m)}\big)\big(A^{(0)}_1 B^1_{(0)} + A^{(0)}_2 B^2_{(0)}\big)g\,, \quad m \;=\; 0\,,\,1\,,\,\ldots\,. \end{aligned}$$

But $A^{(0)}_1 B^1_{(0)} + A^{(0)}_2 B^2_{(0)} = A^{(0)}B_{(0)} = I$, so the update rule simplifies to

$$f^1_{(m+1)} \;=\; \Big[\big(D_x f^1_{(m)}\big)A^{(0)}_1\Big]^{-1}\big(D_x f^1_{(m)}\big)g\,, \quad m \;=\; 0\,,\,1\,,\,\ldots\,. \tag{4.19}$$

In the special case where CSP is initialized with the identity matrices $A^{(0)} = B_{(0)} = I$, the update rule (4.19) reduces further to

$$f^1_{(m+1)} \;=\; \big(D_{x^1} f^1_{(m)}\big)^{-1}\big(D_x f^1_{(m)}\big)g\,, \quad m \;=\; 0\,,\,1\,,\,\ldots\,. \tag{4.20}$$

The rule (4.19) shows that one can obtain $f^1_{(m+1)}$ solely in terms of $f^1_{(m)}$ and g. The same is then true for $\mathcal{K}^{(m+1)}$; see Eq. (4.12). In other words, one can obtain the sequence $\{\mathcal{K}^{(m)} : m = 0, 1, \ldots\}$ of approximate manifolds recursively by means of Eqs. (4.12) and (4.19), knowing only $A^{(0)}$ and $B_{(0)}$.

4.3 CSP coordinates in state space

We conclude the discussion of CSP by showing that the local basis $A^{(m)}$ can induce a set of coordinates $y_{(m)}$ in the state space.

For simplicity, and to enable a comparison with ZDP, we assume that CSP is seeded with the standard coordinate system, $y_{(0)}(x) = x$ and $A^{(0)} = B_{(0)} = I$. In this case, Eqs. (4.17)–(4.18) reduce to

$$A^{(m+1)} \;=\; \begin{pmatrix} I_1 & -\big(D_{x^1} f^1_{(m)}\big)^{-1}\big(D_{x^2} f^1_{(m)}\big) \\ 0 & I_2 \end{pmatrix}, \quad m \;=\; 0\,,\,1\,,\,\ldots\,, \tag{4.21}$$

$$B_{(m+1)} \;=\; \begin{pmatrix} I_1 & \big(D_{x^1} f^1_{(m)}\big)^{-1}\big(D_{x^2} f^1_{(m)}\big) \\ 0 & I_2 \end{pmatrix}, \qquad m \;=\; 0\,,\,1\,,\,\ldots\,. \tag{4.22}$$

Furthermore, we assume that $\mathcal{K}^{(m)}$ is the graph of a function $h_{(m)} : K \to \mathbf{R}^{N_1}$ over a compact set $K \subset \mathbf{R}^{N_2}$,

$$f^1_{(m)}\big(y\big(h_{(m)}(x^2), x^2\big)\big) \;=\; \Big(B^1_{(m)}\big(h_{(m-1)}(x^2), x^2\big) g\big(h_{(m)}(x^2), x^2\big)\Big) \;=\; 0\,, \quad x^2 \in K\,. \tag{4.23}$$

Taking the total derivative of Eq. (4.23) with respect to x^2, we obtain the identity

$$\big(D_{x^1} f^1_{(m)}\big)\big(Dh_{(m)}\big) + D_{x^2} f^1_{(m)} \;=\; 0\,,$$

where $Dh_{(m)}$ is the Jacobian of $h_{(m)}$ (a rectangular matrix of size $N_1 \times N_2$). Hence,

$$\big(D_{x^1} f^1_{(m)}\big)^{-1}\big(D_{x^2} f^1_{(m)}\big) \;=\; -Dh_{(m)} \quad \text{on} \quad \mathcal{K}^{(m)}\,. \tag{4.24}$$

Consequently, on $\mathcal{K}^{(m)}$, Eqs. (4.21)–(4.22) reduce to

$$A^{(m+1)} \;=\; \begin{pmatrix} I_1 & Dh_{(m)} \\ 0 & I_2 \end{pmatrix}, \quad B_{(m+1)} \;=\; \begin{pmatrix} I_1 & -Dh_{(m)} \\ 0 & I_2 \end{pmatrix}, \quad m \;=\; 0\,,1\,,\ldots\,. \tag{4.25}$$

We extend these expressions from points on $\mathcal{K}^{(m)}$ to a neighborhood of $\mathcal{K}^{(m)}$ by assigning the basis $A^{(m+1)}(x) = A^{(m+1)}\big(h_{(m)}(x^2), x^2\big)$ to any point $x = (x^1, x^2)$ with x^1 close to $h_{(m)}(x^2)$ and $x^2 \in K$.

To see that $A^{(m+1)}$ thus extended induces coordinates in the state space, start from the identity $B_{(m+1)} = D_x y_{(m+1)}$ and substitute $B_{(m+1)}$ from Eq. (4.25),

$$B_{(m+1)} \;=\; \begin{pmatrix} D_{x^1} y^1_{(m+1)} & D_{x^2} y^1_{(m+1)} \\ D_{x^1} y^2_{(m+1)} & D_{x^2} y^2_{(m+1)} \end{pmatrix} \;=\; \begin{pmatrix} I_1 & -Dh_{(m)} \\ 0 & I_2 \end{pmatrix}, \quad m \;=\; 0\,,1\,,\ldots\,.$$

These ODEs can be integrated up to additive constants; the latter correspond to a translation of the origin and can be ignored. The result is

$$\begin{pmatrix} y^1_{(m+1)}(x) \\ y^2_{(m+1)}(x) \end{pmatrix} \;=\; \begin{pmatrix} x^1 - h_{(m)}(x^2) \\ x^2 \end{pmatrix}, \quad m \;=\; 0\,,1\,,\ldots\,. \tag{4.26}$$

Therefore, each update of the local basis from $A^{(m)}$ to $A^{(m+1)}$ induces a coordinate change in the state space from $y_{(m)}$ to $y_{(m+1)}$.

We emphasize that the above results are particular to the CSP update rules, where the basis $A^{(m)}$ is extended to a neighborhood of $\mathcal{K}^{(m)}$ as described after Eq. (4.25); otherwise, the basis $A^{(m+1)}$ is not necessarily associated with a coordinate system. (Such a basis is often referred to as a *noncoordinate basis* [19]). This can be seen already for $A^{(1)}$. If this basis were associated with some coordinate system (Y^1, Y^2), Eq. (4.22) would have to be satisfied. Since $B_{(0)} = I$, we have $f_{(0)} = g$, so Eq. (4.22) yields

$$B_{(1)} \;=\; \begin{pmatrix} D_{x^1} Y^1 & D_{x^2} Y^1 \\ D_{x^1} Y^2 & D_{x^2} Y^2 \end{pmatrix} \;=\; \begin{pmatrix} I_1 & \big(D_{x^1} g^1\big)^{-1}\big(D_{x^2} g^1\big) \\ 0 & I_2 \end{pmatrix}.$$

This is a system of $N \times N$ ODEs written in block format. The upper left block yields $Y^1 = x^1 - \omega(x^2)$, for some function ω. Substituting this expression in the upper right block, we arrive at the equation $D_{x^2}\omega = -\big(D_{x^1} g^1\big)^{-1}\big(D_{x^2} g^1\big)$. While the left member of this equation for ω depends only on x^2, its right member depends, for a general nonlinear vector field g, on both x^1 and x^2; hence, the equation has, in general, no solution. Therefore, $A^{(1)}$ does not, in general, induce a coordinate system in the state space. A similar argument applies to higher values of m.

5 The zero-derivative principle

The zero-derivative principle focuses on the state variables and is, in essence, an iterative process to generate an ideal coordinate system in the state space where the fast and slow dynamics decouple. At each step, one partitions the ZDP coordinates into "fast" and "slow" and constructs a change of coordinates $y = y(x)$ by identifying the new "fast" coordinates with the time derivatives of the old ones. An approximation to the slow manifold is defined as the locus of points in state space where the then-current "fast" variables vanish, as in Eq. (3.2).

As with CSP, the dimensions of the fast and slow subspaces—and therefore also the variables that can be classified as fast and slow—are not known a priori. Therefore, we again use the generic sub- and superscripts "1" and "2", rather than "f" and "s", as labels for potentially fast and slow categories. In the case of a fast-slow system, where the fast and slow variables can be identified a priori and the ratio of the characteristic times of the fast and slow variables is measured by an arbitrarily small parameter ε, ZDP generates the asymptotic expansions of slow manifolds order by order, iterate by iterate [4].

In Section 5.1 we describe ZDP in some detail. Our main results are contained in Sections 5.2 and 5.3. In Section 5.2 we determine the basis that is induced in the tangent space by the coordinate transformation in the state space. In Section 5.3 we derive the update rule for this basis and show that it is similar to the update rule of CSP. Throughout this section, we use the same symbols as in Section 4 to denote quantities that have a similar meaning in CSP and ZDP.

5.1 The principle of ZDP

Initialized with the standard coordinate system $y_{(0)}(x) = x$ in $\mathbf{R}^N$, ZDP generates a sequence of state space coordinates,

$$\left\{ y_{(m)}(x) : m = 0, 1, \ldots \right\}, \tag{5.1}$$

by means of the update rule

$$y_{(m+1)}(x) \;=\; \begin{pmatrix} y^1_{(m+1)}(x) \\ y^2_{(m+1)}(x) \end{pmatrix} = \begin{pmatrix} \left(y^1_{(m)}\right)'(x) \\ y^2_{(m)}(x) \end{pmatrix}, \quad m \;=\; 0\,,\, 1\,,\, \ldots\,. \tag{5.2}$$

Time derivatives are taken along solutions of Eq. (2.1), so $d\cdot/dt = (D_x\,\cdot)g$. An explicit form of the update rule is

$$y_{(m+1)}(x) \;=\; \begin{pmatrix} L_{(m)}(x) \\ x^2 \end{pmatrix}, \quad m \;=\; 0\,,\, 1\,,\, \ldots\,, \tag{5.3}$$

where

$$L_{(m)}(x) \;=\; \left(\frac{d^{m+1}x^1}{dt^{m+1}} \right)(x)\,. \tag{5.4}$$

At the $(m+1)$th iteration, an approximate slow manifold is defined by

$$\mathcal{L}^{(m)} \;=\; \left\{ x \in \mathbf{R}^N : y^1_{(m+1)}(x) = L_{(m)}(x) = 0 \right\}, \quad m \;=\; 0\,,\, 1\,,\, \ldots\,. \tag{5.5}$$

This definition, which is the analog of Eq. (3.2), shows why we refer to the present reduction technique as the "zero derivative principle."

5.2 Local bases for the ZDP coordinates

In this section, we determine the local basis induced in $\mathcal{T}\,\mathbf{R}^N$ by the coordinates $y_{(m)}$ and show that, in this basis, $\mathcal{L}^{(m)}$ coincides with the locus of points where the "fast" vector field component current at the mth iteration vanishes, as in Eq. (3.6).

We first consider the case $m = 0$. Here, $y_{(0)}(x) = x$, so $D_{y_{(0)}}x = A^{(0)} = I$ and $D_x y_{(0)} = B^{(0)} = I$. In other words, the initial coordinates $y_{(0)}$ induce the standard basis in the tangent space. In this basis, we have $f_{(0)} = g$.

Since $g^1 = (x^1)' = L_{(0)}$, we can write the vector $f_{(0)}$ as

$$f_{(0)} \;=\; \begin{pmatrix} L_{(0)} \\ g^2 \end{pmatrix}. \tag{5.6}$$

For higher values of m, we use the definition (5.3) to find the dual basis,

$$B_{(m)} \;=\; \begin{pmatrix} B^1_{(m)} \\ B^2_{(m)} \end{pmatrix} = \begin{pmatrix} D_x y^1_{(m)} \\ D_x y^2_{(m)} \end{pmatrix} = \begin{pmatrix} D_{x^1} L_{(m-1)} & D_{x^2} L_{(m-1)} \\ 0 & I_2 \end{pmatrix}. \tag{5.7}$$

The corresponding basis $A^{(m)}$ is found by inversion,

$$A^{(m)} \;=\; \big(B_{(m)}\big)^{-1} \;=\; \begin{pmatrix} \big(D_{x^1} L_{(m-1)}\big)^{-1} & -\big(D_{x^1} L_{(m-1)}\big)^{-1} D_{x^2} L_{(m-1)} \\ 0 & I_2 \end{pmatrix}. \tag{5.8}$$

In this basis, the vector field is given by $g = A^{(m)} f_{(m)}$, with

$$f_{(m)} \;=\; B_{(m)} g \;=\; \begin{pmatrix} \big(D_x L_{(m-1)}\big) g \\ g^2 \end{pmatrix}. \tag{5.9}$$

Since

$$\big(D_x L_{(m-1)}\big) g \;=\; \big(L_{(m-1)}\big)' \;=\; \left(\frac{d^m x^1}{dt^m}\right)' \;=\; \frac{d^{m+1} x^1}{dt^{m+1}} \;=\; L_{(m)}\,, \tag{5.10}$$

it follows that

$$f_{(m)} \;=\; \begin{pmatrix} L_{(m)} \\ g^2 \end{pmatrix}, \quad m \;=\; 0\,,\, 1\,,\, \dots\,. \tag{5.11}$$

In particular, $f^1_{(m)} = L_{(m)}$, for $m = 0, 1, \dots$.

Combining this result with the definition (5.5), we obtain yet a third definition of $\mathcal{L}^{(m)}$:

$$\mathcal{L}^{(m)} \;=\; \big\{x \in \mathbf{R}^N : f^1_{(m)}(x) = 0\big\}\,, \quad m \;=\; 0\,,\, 1\,,\, \dots\,. \tag{5.12}$$

This is the ZDP analog of Eq. (4.12).

We emphasize that the above calculation of $A^{(m)}$ and $B_{(m)}$ and the alternative characterization (5.12) of $\mathcal{L}_{(m)}$ are not part of ZDP; they are introduced here for the sole purpose of comparing ZDP and CSP. In practical implementations of ZDP, the sequence $\{\mathcal{L}^{(m)} : m = 0, 1, \dots\}$ is obtained by working strictly in the state space, as explained in Section 5.1.

Given the alternative characterization (5.12) of $\mathcal{L}_{(m)}$, one can recursively update f^1 without recourse to computing the bases. Indeed, it follows from Eqs. (5.10) and (5.11) that

$$f^1_{(m+1)} \;=\; \big(f^1_{(m)}\big)' \;=\; \big(D_x f^1_{(m)}\big) g\,. \tag{5.13}$$

Hence, the "fast" vector field component current at the $(m+1)$th iteration, $f^1_{(m+1)}$, can be obtained in terms of its counterpart at the mth iteration, $f^1_{(m)}$, just as for CSP, see Eq. (4.19). Notice that the update rule (5.13) differs from its CSP analog (4.19) by a scaling factor of Λ^{11}.

Remark 5.1 Eqs. (5.3) and (5.11) show that the update rule (5.2) for the ZDP coordinates can be recast as

$$y_{(m+1)}(x) \;=\; \begin{pmatrix} y^1_{(m+1)}(x) \\ y^2_{(m+1)}(x) \end{pmatrix} = \begin{pmatrix} f^1_{(m)}(x) \\ x^2 \end{pmatrix}, \quad m \;=\; 0\,,\, 1\,,\, \dots\,. \tag{5.14}$$

That is, at each iteration, the new "fast" coordinates in the state space are obtained as the old "fast" coordinates of the vector field.

5.3 Local basis updates

In this section, we derive the update rules for the ZDP bases $A^{(m)}$ and $B_{(m)}$. They are similar to those of CSP, Eq. (4.11), and reduce $\Lambda_{(m)}$ to block-diagonal form. The latter is defined as before, $\Lambda_{(m)} = D_{y_{(m)}} f_{(m)} = \big(D_x f_{(m)}\big) A^{(m)} = B_{(m)}\big[A^{(m)}, g\big]$, and governs the evolution of $f_{(m)}$.

Let $C_{(m)}$ be the invertible matrix updating $A^{(m)}$ to $A^{(m+1)}$,

$$A^{(m+1)} \;=\; A^{(m)} C_{(m)}\,, \quad B^{(m+1)} \;=\; \big(C_{(m)}\big)^{-1} B^{(m)}\,. \tag{5.15}$$

Applying $B_{(m)}$ to both sides of the first equation and using the identity $B_{(m)} A^{(m)} = I$, we obtain an expression for the update matrix,

$$C_{(m)} \;=\; B_{(m)} A^{(m+1)}\,. \tag{5.16}$$

Expressions for $B_{(m)}$ and $A^{(m)}$ are given in Eqs. (5.7) and (5.8), respectively. Carrying out the matrix multiplication, we obtain the following expressions for the blocks of $C_{(m)}$:

$$\begin{aligned}
C^{11}_{(m)} &= \big(D_{x^1} L_{(m-1)}\big)\big(D_{x^1} L_{(m)}\big)^{-1}\,,\\
C^{12}_{(m)} &= D_{x^2} L_{(m-1)} - \big(D_{x^1} L_{(m-1)}\big)\big(D_{x^1} L_{(m)}\big)^{-1}\big(D_{x^2} L_{(m)}\big)\,,\\
C^{21}_{(m)} &= 0\,,\\
C^{22}_{(m)} &= I_2\,.
\end{aligned} \tag{5.17}$$

These expressions hold for $m = 0, 1, \ldots$, provided we take $L_{(-1)}(x) = x^1$.

We now express the blocks of $C_{(m)}$ in terms of the blocks of $\Lambda_{(m)}$. We calculate the latter using the formula $\Lambda_{(m)} = (D_x f_{(m)}) A^{(m)}$, substituting $f_{(m)}$ from Eq. (5.11) and $A^{(m)}$ from Eq. (5.8). Thus, we obtain

$$\begin{aligned}
\Lambda^{11}_{(m)} &= \big(D_{x^1} L_{(m)}\big)\big(D_{x^1} L_{(m-1)}\big)^{-1}\,,\\
\Lambda^{12}_{(m)} &= D_{x^2} L_{(m)} - \big(D_{x^1} L_{(m)}\big)\big(D_{x^1} L_{(m-1)}\big)^{-1}\big(D_{x^2} L_{(m-1)}\big)\,,\\
\Lambda^{21}_{(m)} &= \big(D_{x^1} g^2\big)\big(D_{x^1} L_{(m-1)}\big)^{-1}\,,\\
\Lambda^{22}_{(m)} &= D_{x^2} g^2 - \big(D_{x^1} g^2\big)\big(D_{x^1} L_{(m-1)}\big)^{-1}\big(D_{x^2} L_{(m-1)}\big)\,.
\end{aligned}$$

The matrix $C_{(m)}$ can therefore be rewritten as

$$C_{(m)} \;=\; \begin{pmatrix} \big(\Lambda^{11}_{(m)}\big)^{-1} & -\big(\Lambda^{11}_{(m)}\big)^{-1}\Lambda^{12}_{(m)} \\ 0 & I_2 \end{pmatrix}, \quad m \;=\; 0\,,\,1\,,\,\ldots\,, \tag{5.18}$$

and the update rule for the bases becomes

$$A^{(m+1)} \;=\; A^{(m)} \begin{pmatrix} \big(\Lambda^{11}_{(m)}\big)^{-1} & -\big(\Lambda^{11}_{(m)}\big)^{-1}\Lambda^{12}_{(m)} \\ 0 & I_2 \end{pmatrix}, \quad m \;=\; 0\,,\,1\,,\,\ldots\,, \tag{5.19}$$

$$B_{(m+1)} \;=\; \begin{pmatrix} \Lambda^{11}_{(m)} & \Lambda^{12}_{(m)} \\ 0 & I_2 \end{pmatrix} B_{(m)}\,, \qquad\qquad m \;=\; 0\,,\,1\,,\,\ldots\,. \tag{5.20}$$

A comparison of Eqs. (5.19)–(5.20) with their CSP analog, Eq. (4.11), reveals that they differ only by a scaling factor of Λ^{11}. The same was true for the update rules for $f^1_{(m)}$, Eqs. (4.19) and (5.13).

6 Summary

In this article, we have developed a unifying framework for two reduction techniques, namely, computational singular perturbation (CSP) and the zero-derivative principle (ZDP). Both are designed to identify slow manifolds

where the long-term dynamics of the system are played out. The unifying framework arises naturally from the duality between the system dynamics in the state space and that in the space of the vector field (the tangent bundle).

CSP operates in the space of the vector field. At each iteration, it yields a coordinate system for this space in such a manner that the fast and slow components of the vector field become separated to successively higher orders. In this article, we have shown that each such coordinate system induces a coordinate system in the state space and that slow manifolds can be represented naturally in terms of the induced coordinates. We have also discovered and presented ways to streamline CSP, eliminating some intermediate steps at each iteration.

ZDP operates in the state space. At each iteration, it yields a coordinate system for this space in which the fast and slow dynamics are decoupled to successively higher orders. Slow manifolds can be represented naturally in terms of these coordinates. In this article, we have shown that each such coordinate system induces a coordinate system in the space of the vector field, and we have compared the induced coordinate systems with the ones constructed by CSP.

Acknowledgements The work of H. K. was supported by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract W-31-109-Eng-38. The work of T. K. and A. Z. was supported in part by the Division of Mathematical Sciences of the National Science Foundation via grant NSF-0306523. The authors thank Yannis Kevrekidis (Princeton) whose probing questions about reduction methods led to the present investigation.

References

[1] J. Carr, Applications of Centre Manifold Theory, Applied Mathematical Sciences Vol. 35 (Springer-Verlag, New York, 1981).
[2] C. W. Gear and I. G. Kevrekidis, Constraint-defined manifolds: A legacy-code approach to low-dimensional computation, SIAM J. Sci. Comput., to appear.
[3] C. W. Gear, T. J. Kaper, I. G. Kevrekidis, and A. Zagaris, Projecting to a slow manifold: Singularly perturbed systems and legacy codes, SIAM J. Appl. Dyn. Syst., to appear.
[4] C. W. Gear, T. J. Kaper, I. G. Kevrekidis, and A. Zagaris, Projecting to a slow manifold: Singularly perturbed systems and legacy codes, Part 2 (working title), in preparation.
[5] D. A. Goussis and S. H. Lam, A study of homogeneous methanol oxidation kinetics using CSP, in: Proceedings of the Twenty-Fourth Symposium (International) on Combustion, Sydney, July 5–10, 1992 (The Combustion Institute, Pittsburgh, 1992), pp. 113–120.
[6] M. Hadjinicolaou and D. A. Goussis, Asymptotic solutions of stiff PDEs with the CSP method: The reaction diffusion equation, SIAM J. Sci. Comput. **20**, 781–810 (1999).
[7] C. K. R. T. Jones, Geometric singular perturbation theory, in: Dynamical Systems, Montecatini Terme, edited by L. Arnold, Lecture Notes in Mathematics Vol. 1609 (Springer-Verlag, Berlin, 1994), pp. 44–118.
[8] H. G. Kaper and T. J. Kaper, Asymptotic analysis of two reduction methods for systems of chemical kinetics, Phys. D **165**, 66–93 (2002).
[9] I. G. Kevrekidis, C. W. Gear, and G. Hummer, Equation-free: The computer-aided analysis of complex multiscale systems, AIChE J. **50**, 1438–1452 (2004).
[10] 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**, 715–762 (2003).
[11] H.-O. Kreiss, Problems with different time scales, in: Multiple Time Scales, edited by J. H. Brackbill and B. I. Cohen, (Academic Press, Orlando, Florida, 1985), pp. 29–57.
[12] S. H. Lam, Using CSP to understand complex chemical kinetics, Combust. Sci. Tech. **89**, 375–404 (1993).
[13] S. H. Lam and D. A. Goussis, Understanding complex chemical kinetics with computational singular erturbation, in: Proceedings of the Twenty-Second Symposium (International) on Combustion, Seattle, Washington, August 14–19, 1988 (The Combustion Institute, Pittsburgh, 1988), pp. 931–941.
[14] S. H. Lam and D. A. Goussis, The CSP method for simplifying kinetics, Internat. J. Chem. Kin. **26**, 461–486 (1994).
[15] T. F. Lu, Y. G. Ju, and C. K. Law, Complex CSP for chemistry reduction and analysis, Combustion and Flame **126**, 1445–1455 (2001).
[16] A. Massias, D. Diamantis, E. Mastorakos, and D. Goussis, Global reduced mechanisms for methane and hydrogen combustion with nitric oxide formation constructed with CSP data, Combust. Theory Model. **3**, 233–257 (1999).
[17] A. Massias and D. A. Goussis, On the manifold of stiff reaction-diffusion PDE's: The effects of diffusion, preprint (2001).

[18] M. K. Neophytou, D. A. Goussis, M. van Loon, and E. Mastorakos, Reduced chemical mechanisms for atmospheric pollution using computational singular perturbation analysis, Atmospheric Environment **38**, 3661–3673 (2004).
[19] B. Schutz, Geometrical Methods of Mathematical Physics (Cambridge University Press, Cambridge, U.K., 1980).
[20] M. Valorani and D. A. Goussis, Explicit time-scale splitting algorithm for stiff problems: auto-ignition of gaseous mixtures behind a steady shock, J. Comput. Phys. **169**, 44–79 (2001).
[21] M. Valorani, H. M. Najm, and D. A. Goussis, CSP analysis of a transient flame-vortex interaction: Time scales and manifolds, Combustion and Flame **134**, 35–53 (2003).
[22] A. Zagaris, H. G. Kaper, and T. J. Kaper, Analysis of the computational singular perturbation reduction method for chemical kinetics, J. Nonlinear Sci. **14**, 59–91 (2004).
[23] A. Zagaris, H. G. Kaper, and T. J. Kaper, Fast and slow dynamics for the computational singular perturbation method, Multiscale Model. Simul. **2**, 613–638 (2004).