Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 214 (2006) 316–346
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

An efficient iterative algorithm for the approximation of the fast and slow dynamics of stiff systems

Dimitris A. Goussis [a,*], Mauro Valorani [b]

[a] *Agiou Georgiou 49, 26500 Patra, Greece*
[b] *Dipartimento di Meccanica e Aeronautica, Via Eudossiana 18, 00184 Rome, Italy*

Received 14 July 2005; received in revised form 23 September 2005; accepted 23 September 2005
Available online 9 November 2005

Abstract

The relation between the iterative algorithms based on the computational singular perturbation (CSP) and the invariance equation (IE) methods is examined. The success of the two methods is based on the appearance of fast and slow time scales in the dynamics of stiff systems. Both methods can identify the low-dimensional surface in the phase space (slow invariant manifold, SIM), where the state vector is attracted under the action of fast dynamics. It is shown that this equivalence of the two methods can be expressed by simple algebraic relations. CSP can also construct the simplified non-stiff system that models the slow dynamics of the state vector on the SIM. An extended version of IE is presented which can also perform this task. This new IE version is shown to be exactly similar to a modified version of CSP, which results in a very efficient algorithm, especially in cases where the SIM dimension is small, so that significant model simplifications are possible.

Keywords: Invariant manifolds; Model reduction; Multiple time scales; Asymptotic analysis; Singular perturbation analysis

1. Introduction

Stiff dynamical systems often evolve according to the fastest time scales during a short period only, as in boundary layers, shocks, etc. In the remaining much longer period the solution evolves on a low-dimensional surface in the phase space (slow invariant manifold, SIM) according to the slower time scales [1,2]; the fastest time scales, being exhausted, restrain the solution on this surface.

The existence of the fast time scales in the long periods where the solution is characterized by the slow ones generates the numerical difficulties, typical of stiff systems.

The accurate identification of the SIM allows the simplification of the stiff dynamical system (i.e., smaller number of unknowns and a vector field free of the fast scales), while the generation of the simplified non-stiff system governing the evolution of the state vector on the SIM allows the use of computationally efficient (e.g.,

* Corresponding author. Tel.: +302610990619.
E-mail address: macdagou@otenet.gr (D.A. Goussis).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2005.09.019

explicit) time integration methods. In addition, the availability of the SIM and the simplified system can shed light on the physics of the problem under examination, by identifying the processes in equilibrium (responsible for the development of the SIM) and the driving ones (dominating the simplified system). For these reasons this topic has lately attracted considerable attention. Among the most established algorithms are the invariance equation (IE) algorithm by Fraser and Roussel [3–15], the computational singular perturbation (CSP) method by Lam and Goussis [16–28] and the intrinsic low dimensional method (ILDM) approach by Maas and Pope [29–40].

The IE, CSP and ILDM methods can all generate approximations of the SIM. For nonlinear problems, IE and CSP, both iterative algorithms, can in principle provide approximate SIM's of unlimited precision, their accuracy being specified by the number of iterations carried out [41,42,45]. In contrast, ILDM provides second order accuracy [17,42–45], recovering the exact description of the SIM only when the original problem is linear [17,41].

CSP and ILDM provide directly the simplified non-stiff system, whereas IE does not. For nonlinear systems, the accuracy of the solution of the simplified problem is determined by the number of iterations when CSP is employed and is just leading order when ILDM is used [17,45].

The IE iterative algorithm is based on the solution of the so-called invariance equation, for which the attracting low-dimensional SIM is a fixed point. The algorithm requires the partitioning of the unknowns (or a linear combination of them) in two sets, one of which parameterizes the SIM. However, no rule is provided for the identification of these two sets of unknowns nor the possibility that the size or content of these two sets might be different in various regions of the phase space is faced. The Jacobian of the vector field, say **J**, is not explicitly required, but it might be needed when the invariance equation is solved during the iterations. Convergence problems were reported [3,5,9] and different methodologies were proposed to overcome this problem [9,12–15].

The CSP and ILDM methods are based on the decomposition of the tangent space into fast and slow subspaces; i.e., the subspaces where the fast and slow time scales act, respectively. The two methods provide approximations of the two subspaces, by producing the appropriate basis vectors. According to CSP, these vectors are produced by applying a refinement procedure; each refinement resulting in a better approximation of the fast/slow subspaces. For a nonlinear problem, leading order accuracy requires the Jacobian **J** and either (i) one block-power method iteration or (ii) the solution of the eigenvalue problem. If higher than leading order accuracy is sought, CSP requires a number of additional iterations involving time derivatives of **J** [45]. According to ILDM, the basis vectors are the eigenvectors of **J**. ILDM does not propose a procedure for improving the leading order accuracy provided by the eigenvectors.

Here, the iterative IE and CSP methods will be considered and their relation will be established. The focus of this study is the development of an accurate and efficient algorithm for the identification of the SIM and the construction of the simplified non-stiff system.

After stating the problem, the IE and CSP methods will be briefly reviewed. Then, the extension of the former to produce simplified non-stiff systems, as CSP does, will be presented. The conditions that establish the equivalence of the two algorithms will then be stated and analyzed. A new version of CSP will finally be presented, which results in an algorithm that is much more efficient than the original one, especially in the cases where SIM's of very small dimensions emerge and, therefore, large model simplifications are possible. These developments will be validated using a model problem describing the kinetics of air association/dissociation processes.

2. Statement of the problem

Consider the N-dimensional stiff system:

$$\frac{\mathrm{d}\underline{y}}{\mathrm{d}t} = \underline{g}(\underline{y}), \tag{1}$$

where the state vector $\underline{y}$ and the vector field $\underline{g}$ are N-dimensional column vectors; $\underline{y} = (y^1, \ldots, y^N)^{\mathrm{T}}$ and $\underline{g} = (g^1, \ldots, g^N)^{\mathrm{T}}$. It is assumed that Eq. (1) is characterized by a wide range of time scales. This feature is manifested in a large spectral radius of the Jacobian, **J**, of $\underline{g}$, and by the largest eigenvalues of **J** (in magnitude)

having a dominant negative real part. We are interested in the time period where the solution evolves on an $(N-M)$-dimensional SIM, according to the $N-M$ slower time scales; the M fastest time scales being exhausted. The stiffness of Eq. (1) is guaranteed as long as a wide range of time scales exists and the system is evolving according to slow ones. It is desired to derive the equation defining the SIM and the simplified non-stiff system governing the evolution of the system on the SIM.

3. Description of the SIM

Suppose that the SIM is an $(N-M)$-dimensional surface in the N-dimensional phase space. By definition, the M fastest time scales associated with all state vectors belonging to this SIM are exhausted. Let this SIM be parameterized by some $N-M$, say s^j $(j=1,N-M)$, smooth functions of $\underline{y}$:

$$s^j = s^j(\underline{y}) = s^j(y^1,\ldots,y^N), \quad j=1,N-M \tag{2}$$

so that the state vector on the SIM can be computed from:

$$y^i = y^i(\underline{s}) = y^i(s^1,\ldots,s^{N-M}), \quad i=1,N, \tag{3}$$

where $\underline{s}=(s^1,\ldots,s^{N-M})^{\mathrm{T}}$. Differentiating Eq. (3) with time yields:

$$\frac{\mathrm{d}\underline{y}}{\mathrm{d}t} = \mathbf{Y}_s \frac{\mathrm{d}\underline{s}}{\mathrm{d}t} = \underline{g}(\underline{y}), \tag{4}$$

where

$$\frac{\mathrm{d}\underline{s}}{\mathrm{d}t} = \mathbf{S}_y \frac{\mathrm{d}\underline{y}}{\mathrm{d}t} = \mathbf{S}_y \underline{g}(\underline{y}) \tag{5}$$

and $\mathbf{Y}_s$ and $\mathbf{S}_y$ are $N\times(N-M)$ and $(N-M)\times N$ matrices:

$$\mathbf{Y}_s = \begin{bmatrix} \frac{\partial y^1}{\partial s^1} & \cdots & \frac{\partial y^1}{\partial s^{N-M}} \\ \vdots & & \vdots \\ \frac{\partial y^N}{\partial s^1} & \cdots & \frac{\partial y^N}{\partial s^{N-M}} \end{bmatrix}, \quad \mathbf{S}_y = \begin{bmatrix} \frac{\partial s^1}{\partial y^1} & \cdots & \frac{\partial s^1}{\partial y^N} \\ \vdots & & \vdots \\ \frac{\partial s^{N-M}}{\partial y^1} & \cdots & \frac{\partial s^{N-M}}{\partial y^N} \end{bmatrix}$$

satisfying the relation:

$$\mathbf{S}_y \mathbf{Y}_s = \mathbf{I}_s^s,$$

where $\mathbf{I}_s^s$ is the $(N-M)\times(N-M)$ unit matrix. Substituting Eq. (5) in Eq. (4) yields the N-dimensional system of algebraic equations:

$$\left[\mathbf{I}_N^N - \mathbf{Y}_s\mathbf{S}_y\right]\underline{g}(\underline{y}) = \underline{0}. \tag{6}$$

Only M components of Eq. (6) are linearly independent, enough to describe the SIM. It follows that the solution on the SIM is governed by the N-dimensional system:

$$\frac{\mathrm{d}\underline{y}}{\mathrm{d}t} = \mathbf{Y}_s\mathbf{S}_y\underline{g}(\underline{y}) \tag{7}$$

of which only $N-M$ components are linearly independent. Different forms of Eqs. (6) and (7) are encountered in CSP and ILDM, as well as in the MIM algorithm for constructing low-dimensional manifolds [46,47].

4. The IE algorithm for identifying the SIM

Let the smooth functions s^j be the $N-M$ last components of $\underline{y}$:

$$\underline{s} = (y^{M+1},\ldots,y^N)^{\mathrm{T}} \tag{8}$$

and define the M-dimensional vector $\underline{z}$ as:

$$\underline{z} = \left(y^1(\underline{s}), \ldots, y^M(\underline{s})\right)^{\mathrm{T}} = \underline{z}(\underline{s}). \tag{9}$$

Eq. (6) reduces to:

$$\underline{g}^r(\underline{z},\underline{s}) - \mathbf{G}_s^r(\underline{z},\underline{s})\underline{g}^s(\underline{z},\underline{s}) = \underline{0}, \tag{10}$$

where $\underline{g}^r = (g^1,\ldots,g^M)^{\mathrm{T}}$ and $\underline{g}^s = (g^{M+1},\ldots,g^N)^{\mathrm{T}}$ are M- and $(N-M)$-dimensional vectors consisting of the first M and last $N-M$ elements, respectively, of the vector field $\underline{g}$. The $M\times(N-M)$ matrix $\mathbf{G}_s^r$ involves the partial derivatives of the M components of $\underline{y}$ in $\underline{z}$ with respect to the remaining $N-M$ components of $\underline{y}$ in $\underline{s}$:

$$\mathbf{G}_s^r(\underline{z},\underline{s}) = \frac{\partial \underline{z}}{\partial \underline{s}} = \begin{bmatrix} \dfrac{\partial z^1}{\partial s^1} & \cdots & \dfrac{\partial z^1}{\partial s^{N-M}} \\ \vdots & & \vdots \\ \dfrac{\partial z^M}{\partial s^1} & \cdots & \dfrac{\partial z^M}{\partial s^{N-M}} \end{bmatrix}. \tag{11}$$

Eq. (10) is the "invariance equation", consisting of M equations for M unknowns (the components of $\underline{z}$). According to IE, the $(n+1)$th iterate $\underline{z}_{n+1}$ can be computed, for a given $\underline{s}$ and an initial guess $\underline{z}_o$, from the implicit equation:

$$\underline{g}^r(\underline{z}_{n+1},\underline{s}) - \mathbf{G}_s^r(\underline{z}_n,s)\underline{g}^s(\underline{z}_{n+1},\underline{s}) = \underline{0} \tag{12}$$

or, by virtue of the implicit function theorem, from an explicit functional equation of the general form:

$$\underline{z}_{n+1} = \underline{H}(\underline{z}_n,\underline{s}), \tag{13}$$

where the derivatives in the matrix $\mathbf{G}_s^r(\underline{z}_n,\underline{s})$ can be approximated numerically, possibly by finite differences [12,15]. In principle, with every iteration, Eq. (13) provides a description of the $(N-M)$-dimensional SIM with increased accuracy [41]. However, the convergence of this process is not guaranteed or might not be monotonic [3,9,10,12,15]. Of course, convergence depends on the magnitude of the Jacobian:

$$\mathbf{J}_H = \frac{\partial \underline{H}(\underline{z}_n,\underline{s})}{\partial \underline{z}_n}.$$

The iterative scheme (13) converges if $|\mathbf{J}_H| < 1$, the convergence being monotonic when $0 < \mathbf{J}_H < 1$ and oscillatory when $-1 < \mathbf{J}_H < 0$ [48]. Different methods were proposed to achieve and stabilize the convergence of the iterative process, the most used one involving a pseudo-time stepping procedure [12,14,15]:

$$\frac{\mathrm{d}\underline{z}}{\mathrm{d}\tau} = \underline{H}(\underline{z},\underline{s}) - \underline{z} \tag{14}$$

that either stabilizes the iteration steps (13) [12,14] or provides directly the SIM as the fixed point of Eq. (14) [15]. Apparently, one can infer that possible causes of the convergence problems are (i) the stiffness of the iterative equation (13), which is also manifested in the dynamics of Eq. (14) [12,15], (ii) a poor condition number of the matrix $\mathbf{G}_s^r$ generated by an incorrect identification of the fast and slow variables [49], yielding a non-optimal partitioning of $\underline{y}$ in $\underline{s}$ and $\underline{z}$, (iii) the existence of multiple solutions of the invariance equation and (iv) the specific form of the function $\underline{H}(\underline{z}_n,\underline{s})$ employed, not satisfying the condition for convergence $|\mathbf{J}_H| < 1$.

It is noted here that, to this point, the IE algorithm provides no rules on how to partition the state vector $\underline{y}$ in $\underline{z}$ and $\underline{s}$, i.e., in a fast and a slow component, respectively, nor suggests the appropriate form of the function $\underline{H}(\underline{z}_n,\underline{s})$ so that convergence can be obtained.

5. An algorithm for constructing the simplified system using the $\mathbf{G}_s^r$ matrix

The fact that the invariance equation (10) can be cast as:

$$[\mathbf{I}_r^r, -\mathbf{G}_s^r]\begin{bmatrix} \underline{g}^r(\underline{z},\underline{s}) \\ \underline{g}^s(\underline{z},\underline{s}) \end{bmatrix} = \underline{0} \tag{15}$$

suggests the introduction of the matrices:

$$\mathbf{A}_r = \begin{bmatrix} \mathbf{I}_r^r - \mathbf{G}_s^r \mathbf{R}_r^s \\ -\mathbf{R}_r^s \end{bmatrix}, \quad \mathbf{A}_s = \begin{bmatrix} \mathbf{G}_s^r \\ \mathbf{I}_s^s \end{bmatrix}, \tag{16}$$

$$\mathbf{B}^r = [\mathbf{I}_r^r, -\mathbf{G}_s^r], \quad \mathbf{B}^s = [\mathbf{R}_r^s, \mathbf{I}_s^s - \mathbf{R}_r^s \mathbf{G}_s^r], \tag{17}$$

where the $M \times (N - M)$ matrix $\mathbf{G}_s^r$ was defined by Eq. (11) and the $(N - M) \times M$ matrix $\mathbf{R}_r^s$ will be defined later when the condition for the non-stiffness of the simplified system is stated. The dimensions of the matrices $\mathbf{A}_r$, $\mathbf{A}_s$, $\mathbf{B}^r$ and $\mathbf{B}^s$ are $N \times M$, $N \times (N - M)$, $M \times N$ and $(N - M) \times N$, respectively. They satisfy the orthogonality relations:

$$[\mathbf{A}_r \mathbf{A}_s] \begin{bmatrix} \mathbf{B}^r \\ \mathbf{B}^s \end{bmatrix} = \begin{bmatrix} \mathbf{B}^r \\ \mathbf{B}^s \end{bmatrix} [\mathbf{A}_r \mathbf{A}_s] = \mathbf{I}_N^N.$$

With the definitions (16) and (17), the original problem (1) can be cast as:

$$\frac{\mathrm{d}\underline{y}}{\mathrm{d}t} = \mathbf{A}_r \underline{F}^r + \mathbf{A}_s \underline{F}^s = \begin{bmatrix} \mathbf{I}_r^r - \mathbf{G}_s^r \mathbf{R}_r^s \\ -\mathbf{R}_r^s \end{bmatrix} \underline{F}^r + \begin{bmatrix} \mathbf{G}_s^r \\ \mathbf{I}_s^s \end{bmatrix} \underline{F}^s, \tag{18}$$

where the amplitudes $\underline{F}^r$ and $\underline{F}^s$ are defined as:

$$\underline{F}^r = \mathbf{B}^r \underline{g} = [\mathbf{I}_r^r, -\mathbf{G}_s^r] \begin{bmatrix} \underline{g}^r \\ \underline{g}^s \end{bmatrix} = (\underline{g}^r - \mathbf{G}_s^r \underline{g}^s), \tag{19}$$

$$\underline{F}^s = \mathbf{B}^s \underline{g} = [\mathbf{R}_r^s, \mathbf{I}_s^s - \mathbf{R}_r^s \mathbf{G}_s^r] \begin{bmatrix} \underline{g}^r \\ \underline{g}^s \end{bmatrix} = \mathbf{R}_r^s \underline{g}^r + \left(\mathbf{I}_s^s - \mathbf{R}_r^s \mathbf{G}_s^r\right) \underline{g}^s. \tag{20}$$

When the solution is on the manifold, the invariance equation (10) is satisfied and ensures that $\underline{F}^r$ is identically zero, that is:

$$\underline{F}^r = \underline{g}^r - \mathbf{G}_s^r \underline{g}^s = \underline{0} \tag{21}$$

and therefore, only the "slow" term involving $\underline{F}^s$ survives in Eq. (18):

$$\frac{\mathrm{d}\underline{y}}{\mathrm{d}t} = \begin{bmatrix} \mathbf{G}_s^r \\ \mathbf{I}_s^s \end{bmatrix} (\mathbf{R}_r^s \underline{g}^r + (\mathbf{I}_s^s - \mathbf{R}_r^s \mathbf{G}_s^r) \underline{g}^s). \tag{22}$$

Eqs. (21) and (22) define the SIM and the simplified non-stiff equation governing the motion of the solution on the SIM, respectively. Notice that by employing Eq. (21), Eq. (22) can be further simplified to:

$$\frac{\mathrm{d}\underline{y}}{\mathrm{d}t} = \begin{bmatrix} \mathbf{G}_s^r \underline{g}^s \\ \underline{g}^s \end{bmatrix}, \tag{23}$$

i.e., to a form independent of $\mathbf{R}_r^s$. However, as it will be demonstrated next, this way of handling Eq. (21) provides no guarantee that the simplified system is non-stiff. For example, in the case where both $\underline{g}^r$ and $\underline{g}^s$ contribute to the development of the fast time scales, preference of Eq. (23) instead of Eq. (22) will cause the stiffness of the original problem to persist in the simplified system.

6. The CSP algorithm

Eq. (18) is similar to the CSP form, according to which Eq. (1) is cast as:

$$\frac{\mathrm{d}\underline{y}}{\mathrm{d}t} = \mathbf{a}_r (\mathbf{b}^r \underline{g}) + \mathbf{a}_s (\mathbf{b}^s \underline{g}), \tag{24}$$

where $\mathbf{a}_r$, $\mathbf{a}_s$ are $N \times M$, $N \times (N - M)$, respectively, matrices containing the fast and slow column CSP vectors, while the $\mathbf{b}^r$ and $\mathbf{b}^s$ matrices are $M \times N$ and $(N - M) \times N$ dimensional and contain the related dual (orthogonal) row vectors. The matrices containing the CSP basis vectors are introduced here in the form:

$$\mathbf{a}_r = \begin{bmatrix} \mathbf{a}_r^r \\ \mathbf{a}_r^s \end{bmatrix}, \quad \mathbf{a}_s = \begin{bmatrix} \mathbf{a}_s^r \\ \mathbf{a}_s^s \end{bmatrix}, \tag{25}$$

$$\mathbf{b}^r = [\mathbf{b}_r^r, \quad \mathbf{b}_s^r], \quad \mathbf{b}^s = [\mathbf{b}_r^s, \quad \mathbf{b}_s^s], \tag{26}$$

where $\mathbf{a}_r^r$ and $\mathbf{b}_r^r$ are $M \times M$ matrices, $\mathbf{a}_r^s$ and $\mathbf{b}_r^s$ are $(N-M)\times M$ matrices, $\mathbf{a}_s^r$ and $\mathbf{b}_s^r$ are $M \times (N-M)$ matrices and $\mathbf{a}_s^s$ and $\mathbf{b}_s^s$ are $(N-M)\times(N-M)$ matrices. Eq. (24) can then be cast as:

$$\frac{\mathrm{d}\underline{y}}{\mathrm{d}t} = \begin{bmatrix} \mathbf{a}_r^r \\ \mathbf{a}_r^s \end{bmatrix} \underline{f}^r + \begin{bmatrix} \mathbf{a}_s^r \\ \mathbf{a}_s^s \end{bmatrix} \underline{f}^s, \tag{27}$$

where

$$\underline{f}^r = \mathbf{b}^r \underline{g} = [\mathbf{b}_r^r, \quad \mathbf{b}_s^r] \begin{bmatrix} \underline{g}^r \\ \underline{g}^s \end{bmatrix} = (\mathbf{b}_r^r \underline{g}^r + \mathbf{b}_s^r \underline{g}^s), \tag{28}$$

$$\underline{f}^s = \mathbf{b}^s \underline{g} = [\mathbf{b}_r^s, \quad \mathbf{b}_s^s] \begin{bmatrix} \underline{g}^r \\ \underline{g}^s \end{bmatrix} = (\mathbf{b}_r^s \underline{g}^r + \mathbf{b}_s^s \underline{g}^s). \tag{29}$$

The equations describing the SIM and the simplified non-stiff system are then:

$$\mathbf{b}_r^r \underline{g}^r + \mathbf{b}_s^r \underline{g}^s = \underline{0}, \tag{30}$$

$$\frac{\mathrm{d}\underline{y}}{\mathrm{d}t} = \begin{bmatrix} \mathbf{a}_s^r \\ \mathbf{a}_s^s \end{bmatrix} (\mathbf{b}_r^s \underline{g}^r + \mathbf{b}_s^s \underline{g}^s). \tag{31}$$

The solution on the SIM can be computed either from the M algebraic equations (30) and $N-M$ components of the differential equation (31) or from all the N components of Eq. (31) [26]. If the former approach is employed, the M components of $\underline{y}$ that will be computed from Eq. (30) are identified by the CSP pointer [18,19], i.e., the M largest components of the N-dimensional vector $\underline{q}^r$:

$$\underline{q}^r = \mathrm{diag}\left[\frac{1}{M} \mathbf{a}_r \mathbf{b}^r\right]. \tag{32}$$

Such a choice guarantees that the algebraic equation (30) provides a solution of the highest accuracy [28]. According to CSP, the basis vectors that provide an exact representation of the fast and slow subspaces obey the evolution equations [17]:

$$\frac{\mathrm{d}\mathbf{b}^r}{\mathrm{d}t} + \mathbf{b}^r \mathbf{J} = \boldsymbol{\lambda}_r^r \mathbf{b}^r, \tag{33}$$

$$-\frac{\mathrm{d}\mathbf{a}_r}{\mathrm{d}t} + \mathbf{J}\mathbf{a}_r = \mathbf{a}_r \boldsymbol{\lambda}_r^r, \tag{34}$$

$$\frac{\mathrm{d}\mathbf{b}^s}{\mathrm{d}t} + \mathbf{b}^s \mathbf{J} = \boldsymbol{\lambda}_s^s \mathbf{b}^s, \tag{35}$$

$$-\frac{\mathrm{d}\mathbf{a}_s}{\mathrm{d}t} + \mathbf{J}\mathbf{a}_s = \mathbf{a}_s \boldsymbol{\lambda}_s^s, \tag{36}$$

where

$$\boldsymbol{\lambda}_r^r = \left(\frac{\mathrm{d}\mathbf{b}^r}{\mathrm{d}t} + \mathbf{b}^r \mathbf{J}\right) \mathbf{a}_r, \quad \boldsymbol{\lambda}_s^s = \left(\frac{\mathrm{d}\mathbf{b}^s}{\mathrm{d}t} + \mathbf{b}^s \mathbf{J}\right) \mathbf{a}_s. \tag{37}$$

The CSP basis vectors are computed from Eqs. (33)–(36) using two refinement processes, the $\mathbf{b}^r$-refinement [17]:

$$\begin{aligned} \mathbf{b}^r(k_1+1, m_1) &= \boldsymbol{\tau}_r^r(k_1, m_1) \left(\mathbf{b}^r(k_1, m_1)\mathbf{J} + \frac{\mathrm{d}\mathbf{b}^r(k_1, m_1)}{\mathrm{d}t}\right), \\ \mathbf{a}_r(k_1+1, m_1) &= \mathbf{a}_r(k_1, m_1), \\ \mathbf{b}^s(k_1+1, m_1) &= \mathbf{b}^s(k_1, m_1), \\ \mathbf{a}_s(k_1+1, m_1) &= \left[\mathbf{I}_r^r - \mathbf{a}_r(k_1+1, m_1)\mathbf{b}^r(k_1+1, m_1)\right] \mathbf{a}_s(k_1, m_1) \end{aligned} \tag{38}$$

and the $\mathbf{a}_r$-refinement:

$$\begin{aligned}
&\mathbf{b}^r(k_2, m_2+1) = \mathbf{b}^r(k_2, m_2),\\
&\mathbf{a}_r(k_2, m_2+1) = \left(\mathbf{J}\mathbf{a}_r(k_2, m_2) - \frac{\mathrm{d}\mathbf{a}_r(k_2, m_2)}{\mathrm{d}t}\right)\boldsymbol{\tau}_r^r(k_2, m_2),\\
&\mathbf{b}^s(k_2, m_2+1) = \mathbf{b}^s(k_2, m_2)\left[\mathbf{I}_r^r - \mathbf{a}_r(k_2, m_2+1)\mathbf{b}^r(k_2, m_2+1)\right],\\
&\mathbf{a}_s(k_2, m_2+1) = \mathbf{a}_s(k_2, m_2),
\end{aligned} \tag{39}$$

where

$$\boldsymbol{\tau}_r^r(k_i, m_i) = \left[\left(\mathbf{b}^r(k_i, m_i)\mathbf{J} + \frac{\mathrm{d}\mathbf{b}^r(k_i, m_i)}{\mathrm{d}t}\right)\mathbf{a}_r(k_i, m_i)\right]^{-1}.$$

The refinement (38) increases the accuracy in the description of the SIM and the solution of the simplified non-stiff problem, while the refinement (39) guarantees the non-stiffness of the simplified system [17–21,26,42,43]. The subscripts "1" and "2" in the indices "k" and "m" denote the fact that the two refinements are independent of each other [17,26,45], preserving the orthogonality of the basis vectors:

$$[\mathbf{a}_r(i,j)\mathbf{a}_s(i,j)]\begin{bmatrix}\mathbf{b}^r(i,j)\\ \mathbf{b}^s(i,j)\end{bmatrix} = \begin{bmatrix}\mathbf{b}^r(i,j)\\ \mathbf{b}^s(i,j)\end{bmatrix}[\mathbf{a}_r(i,j)\mathbf{a}_s(i,j)] = \mathbf{I}_N^N, \tag{40}$$

where $(i,j) = (k_1{+}1, m_1)$ or $(i,j) = (k_2, m_2+1)$.

For completeness, it is noted here that according to ILDM, the basis vectors are the eigenvectors of the Jacobian matrix of the vector field:

$$\begin{bmatrix}\mathbf{b}^r\\ \mathbf{b}^s\end{bmatrix}\mathbf{J} = \begin{bmatrix}\boldsymbol{\mu}_r^r & \mathbf{0}_s^r\\ \mathbf{0}_r^s & \boldsymbol{\mu}_s^s\end{bmatrix}\begin{bmatrix}\mathbf{b}^r\\ \mathbf{b}^s\end{bmatrix}, \quad \mathbf{J}[\mathbf{a}_r \quad \mathbf{a}_s] = [\mathbf{a}_r \quad \mathbf{a}_s]\begin{bmatrix}\boldsymbol{\mu}_r^r & \mathbf{0}_s^r\\ \mathbf{0}_r^s & \boldsymbol{\mu}_s^s\end{bmatrix},$$

where $\boldsymbol{\mu}_r^r$ and $\boldsymbol{\mu}_s^s$ are $M\times M$ and $(N-M)\times(N-M)$, respectively, diagonal matrices, with the M largest and $N-M$ smallest eigenvalues of $\mathbf{J}$. These vectors satisfy Eqs. (38) and (39), if the time derivative terms are neglected, and provide second and leading order accuracy in the description of the SIM and the solution of the simplified non-stiff problem, respectively [17,41,42,44,45].

7. The equivalence of the two algorithms for the construction of the simplified system

It is straightforward to show that the two forms of the original equation (1), i.e., Eqs. (18) and (27), are identical if:

$$\mathbf{G}_s^r = \mathbf{a}_s^r\left(\mathbf{a}_s^s\right)^{-1} = -\left(\mathbf{b}_r^r\right)^{-1}\mathbf{b}_s^r, \tag{41}$$

$$\mathbf{R}_r^s = \mathbf{a}_s^s\mathbf{b}_r^s = -\mathbf{a}_r^s\mathbf{b}_r^r \tag{42}$$

provided that $(\mathbf{b}_r^r)^{-1}$ and $(\mathbf{a}_s^s)^{-1}$ exist. It is then easy to show that the equations describing the SIM and the simplified non-stiff problem generated by Eqs. (21), (22) and (30), (31):

$$\underline{g}^r - \mathbf{G}_s^r\underline{g}^s = \underline{0}, \tag{43}$$

$$\frac{\mathrm{d}\underline{y}}{\mathrm{d}t} = \begin{bmatrix}\mathbf{G}_s^r\\ \mathbf{I}_s^s\end{bmatrix}\left(\mathbf{R}_r^s\underline{g}^r + \left(\mathbf{I}_s^s - \mathbf{R}_r^s\mathbf{G}_s^r\right)\underline{g}^s\right), \tag{44}$$

$$\mathbf{b}_r^r\underline{g}^r + \mathbf{b}_s^r\underline{g}^s = \underline{0}, \tag{45}$$

$$\frac{\mathrm{d}\underline{y}}{\mathrm{d}t} = \begin{bmatrix}\mathbf{a}_s^r\\ \mathbf{a}_s^s\end{bmatrix}\left(\mathbf{b}_r^s\underline{g}^r + \mathbf{b}_s^s\underline{g}^s\right) \tag{46}$$

are also identical. In order to explore the meaning of the two transformations, Eqs. (41) and (42), it is noted that use of Eq. (41) yields the following relations between the new basis vectors introduced here, Eqs. (16) and (17), and the CSP basis vectors:

$$\mathbf{B}^r = \left(\mathbf{b}^r_r\right)^{-1}\mathbf{b}^r, \quad \mathbf{A}_r = \mathbf{a}_r\mathbf{b}^r_r + \mathbf{a}_s\mathbf{N}^s_r,$$
$$\mathbf{B}^s = \mathbf{M}^s_r\mathbf{b}^r + \mathbf{a}^s_s\mathbf{b}^s, \quad \mathbf{A}_s = \mathbf{a}_s\left(\mathbf{a}^s_s\right)^{-1},$$

where

$$\mathbf{N}^s_r = -\left(\mathbf{a}^s_s\right)^{-1}\left[\mathbf{R}^s_r + \mathbf{a}^s_r\mathbf{b}^r_r\right], \quad \mathbf{M}^s_r = \left[\mathbf{R}^s_r + \mathbf{a}^s_r\mathbf{b}^r_r\right]\left(\mathbf{b}^r_r\right)^{-1},$$

i.e., the vectors in $\mathbf{B}^r$ and $\mathbf{A}_s$ span the same subspaces as those in $\mathbf{b}^r$ and $\mathbf{a}_s$ do, respectively; the latter two sets being the ones affected by the CSP $\mathbf{b}^r$-refinement, Eq. (38). However, the vectors in $\mathbf{B}^s$ and $\mathbf{A}_r$ do not span the subspaces spanned by the vectors in $\mathbf{b}^s$ and $\mathbf{a}_r$, respectively; the latter two sets being the ones affected by the CSP $\mathbf{a}_r$-refinement, Eq. (39).

On the other hand, by replacing the matrix $\mathbf{R}^s_r$ as defined by Eq. (42) into the definitions of the new set of basis vectors, Eqs. (16) and (17), we have:

$$\mathbf{B}^r = \left[\mathbf{I}^r_r, -\mathbf{G}^r_s\right], \quad \mathbf{A}_r = \mathbf{a}_r\mathbf{b}^r_r + \begin{bmatrix}\left(\mathbf{a}^r_s - \mathbf{G}^r_s\mathbf{a}^s_s\right)\mathbf{b}^s_r \\ \mathbf{0}^s_r\end{bmatrix},$$
$$\mathbf{B}^s = \left[\mathbf{0}^s_r, \mathbf{a}^s_r\left(\mathbf{b}^r_r\mathbf{G}^r_s + \mathbf{b}^r_s\right)\right] + \mathbf{a}^s_s\mathbf{b}^s, \quad \mathbf{A}_s = \begin{bmatrix}\mathbf{G}^r_s \\ \mathbf{I}^s_s\end{bmatrix},$$

i.e., no set of vectors in $\mathbf{B}^r$, $\mathbf{A}_s$, $\mathbf{B}^s$ and $\mathbf{A}_r$ spans the same subspace as the sets of vectors in $\mathbf{b}^r$, $\mathbf{a}_s$, $\mathbf{b}^s$ and $\mathbf{a}_r$, respectively, do.

When the transformation equation (41) is employed, the invariance equation, Eq. (43), can provide an exact identification of the SIM as the converged CSP $\mathbf{b}^r$-refinement does. However, the simplified non-stiff system constructed by CSP cannot be reproduced. This is achieved when both transformations equations (41) and (42) are employed, in which case the CSP basis vectors and the ones introduced here are related by the expressions:

$$\begin{bmatrix}\mathbf{B}^r \\ \mathbf{B}^s\end{bmatrix} = \begin{bmatrix}\left(\mathbf{b}^r_r\right)^{-1} & \mathbf{0}^r_s \\ \mathbf{0}^s_r & \mathbf{a}^s_s\end{bmatrix}\begin{bmatrix}\mathbf{b}^r \\ \mathbf{b}^s\end{bmatrix}, \tag{47}$$

$$\left[\mathbf{A}_r\mathbf{A}_s\right] = \left[\mathbf{a}_r \quad \mathbf{a}_s\right]\begin{bmatrix}\mathbf{b}^r_r & \mathbf{0}^r_s \\ \mathbf{0}^s_r & \left(\mathbf{a}^s_s\right)^{-1}\end{bmatrix} \tag{48}$$

showing that the vectors in $\mathbf{B}^r$, $\mathbf{A}_s$, $\mathbf{B}^s$ and $\mathbf{A}_r$ are just the CSP basis vectors $\mathbf{b}^r$, $\mathbf{a}_s$, $\mathbf{b}^s$ and $\mathbf{a}_r$, respectively, re-scaled as shown. These new vectors define a modified version of the CSP method, which can be employed only when the state vector $\underline{y}$ is properly partitioned into the $\underline{s}$ and $\underline{z}$ components.

For the identification of the SIM and the construction of the simplified system, Eqs. (43) and (44), this modified CSP algorithm requires the computation of the $\mathbf{G}^r_s$ and $\mathbf{R}^s_r$ matrices. These issues are discussed next.

8. The computation of the matrix $\mathbf{G}^r_s$

The matrix $\mathbf{G}^r_s$ can in principle be computed in a number of ways, which might exhibit numerical difficulties [5,9,12–15]. An alternative method for computing $\mathbf{G}^r_s$ can be obtained by differentiating the invariance equation $\mathbf{B}^r\,\underline{g} = \underline{0}$ with respect to time:

$$\mathbf{\Lambda}^r_r\underline{F}^r + \mathbf{\Lambda}^r_s\underline{F}^s = \underline{0}, \tag{49}$$

where

$$\mathbf{\Lambda}^r_r = \left(\frac{\mathrm{d}\mathbf{B}^r}{\mathrm{d}t} + \mathbf{B}^r\mathbf{J}\right)\mathbf{A}_r, \quad \mathbf{\Lambda}^r_s = \left(\frac{\mathrm{d}\mathbf{B}^r}{\mathrm{d}t} + \mathbf{B}^r\mathbf{J}\right)\mathbf{A}_s \tag{50}$$

from which follows:

$$\frac{\mathrm{d}\mathbf{B}^r}{\mathrm{d}t} + \mathbf{B}^r\mathbf{J} = \mathbf{\Lambda}^r_r\mathbf{B}^r + \mathbf{\Lambda}^r_s\mathbf{B}^s. \tag{51}$$

Since on the manifold $\underline{F}^r = \mathbf{B}^r \underline{g} = \underline{0}$ and since $\underline{F}^s \neq \underline{0}$, Eq. (49) suggests that $\mathbf{\Lambda}^r_s = \mathbf{0}^r_s$ so that Eq. (51) reduces to:

$$\frac{\mathrm{d}\mathbf{B}^r}{\mathrm{d}t} + \mathbf{B}^r \mathbf{J} = \mathbf{\Lambda}^r_r \mathbf{B}^r. \tag{52}$$

Note that this equation is similar to the one governing the evolution of the CSP vectors $\mathbf{b}^r$, Eq. (33). Using the definition for $\mathbf{B}^r$, Eq. (17), and casting the Jacobian $\mathbf{J}$ of the vector field $\underline{g}$ in the form:

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}^r_r & \mathbf{J}^r_s \\ \mathbf{J}^s_r & \mathbf{J}^s_s \end{bmatrix}. \tag{53}$$

Eq. (52) yields the vector equation:

$$\left[\mathbf{0}^r_r, -\frac{\mathrm{d}\mathbf{G}^r_s}{\mathrm{d}t}\right] + \left[\mathbf{J}^r_r - \mathbf{G}^r_s \mathbf{J}^s_r, \mathbf{J}^r_s - \mathbf{G}^r_s \mathbf{J}^s_s\right] = \mathbf{\Lambda}^r_r \left[\mathbf{I}^r_r, -\mathbf{G}^r_s\right] \tag{54}$$

the two components of which produce:

$$\mathbf{J}^r_r - \mathbf{G}^r_s \mathbf{J}^s_r = \mathbf{\Lambda}^r_r, \tag{55}$$

$$\frac{\mathrm{d}\mathbf{G}^r_s}{\mathrm{d}t} + \mathbf{G}^r_s \mathbf{J}^s_s - \mathbf{J}^r_s = \mathbf{\Lambda}^r_r \mathbf{G}^r_s \tag{56}$$

so that the following evolution equation for $\mathbf{G}^r_s$ is obtained:

$$\frac{\mathrm{d}\mathbf{G}^r_s}{\mathrm{d}t} + \mathbf{G}^r_s \mathbf{J}^s_s - \mathbf{J}^r_s = \left[\mathbf{J}^r_r - \mathbf{G}^r_s \mathbf{J}^s_r\right] \mathbf{G}^r_s. \tag{57}$$

Eq. (57), being stiff, can be solved iteratively as:

$$\mathbf{G}^r_s(n+1) = \left(\mathbf{J}^r_r - \mathbf{G}^r_s(n)\mathbf{J}^s_r\right)^{-1} \left[\mathbf{G}^r_s(n)\mathbf{J}^s_s - \mathbf{J}^r_s + \frac{\mathrm{d}\mathbf{G}^r_s(n)}{\mathrm{d}t}\right]. \tag{58}$$

Note that, starting with $\mathbf{G}^r_s(0) = \mathbf{0}^r_s$, Eq. (58) yields:

$$\mathbf{G}^r_s(1) = -\left(\mathbf{J}^r_r\right)^{-1} \mathbf{J}^r_s,$$

i.e., the outcome of one block-power method iteration on $\mathbf{J}$ for computing the left eigenvectors, when starting with $\mathbf{B}^r = [\mathbf{I}^r_r, \mathbf{0}^r_s]$ and using $\mathbf{A}_r = [\mathbf{I}^r_r, \mathbf{0}^r_s]^{\mathrm{T}}$ [50].

Having developed the evolution equation (57) for $\mathbf{G}^r_s$, a direct proof that the CSP algorithm and its variation presented here can describe the SIM with similar accuracy can be obtained as follows. Considering the fast CSP dual vectors in the form $\mathbf{b}^r = [\mathbf{b}^r_r, \mathbf{b}^r_s]$, the evolution equation (33) yields:

$$\frac{\mathrm{d}\left[\mathbf{b}^r_r, \mathbf{b}^r_s\right]}{\mathrm{d}t} + \left[\mathbf{b}^r_r, \mathbf{b}^r_s\right] J = \lambda^r_r \left[\mathbf{b}^r_r, \mathbf{b}^r_s\right]$$

the two components of which produce:

$$\frac{\mathrm{d}\mathbf{b}^r_r}{\mathrm{d}t} + \mathbf{b}^r_r \mathbf{J}^r_r + \mathbf{b}^r_s \mathbf{J}^s_r = \lambda^r_r \mathbf{b}^r_r, \tag{59}$$

$$\frac{\mathrm{d}\mathbf{b}^r_s}{\mathrm{d}t} + \mathbf{b}^r_r \mathbf{J}^r_s + \mathbf{b}^r_s \mathbf{J}^s_s = \lambda^r_r \mathbf{b}^r_s. \tag{60}$$

Pre-multiplying Eq. (60) with $\left(\mathbf{b}^r_r\right)^{-1}$ yields:

$$\frac{\mathrm{d}\left[\left(\mathbf{b}^r_r\right)^{-1}\mathbf{b}^r_s\right]}{\mathrm{d}t} + \left(\mathbf{b}^r_r\right)^{-1}\mathbf{b}^r_s \mathbf{J}^s_s + \mathbf{J}^r_s = \left(\frac{\mathrm{d}\left(\mathbf{b}^r_r\right)^{-1}}{\mathrm{d}t}\mathbf{b}^r_r + \left(\mathbf{b}^r_r\right)^{-1}\lambda^r_r \mathbf{b}^r_r\right)\left(\mathbf{b}^r_r\right)^{-1}\mathbf{b}^r_s.$$

Substituting from Eq. (59) results:

$$\frac{\mathrm{d}\left[\left(\mathbf{b}^r_r\right)^{-1}\mathbf{b}^r_s\right]}{\mathrm{d}t} + \left(\mathbf{b}^r_r\right)^{-1}\mathbf{b}^r_s \mathbf{J}^s_s + \mathbf{J}^r_s = \left(\mathbf{J}^r_r + \left(\mathbf{b}^r_r\right)^{-1}\mathbf{b}^r_s \mathbf{J}^s_r\right)\left(\mathbf{b}^r_r\right)^{-1}\mathbf{b}^r_s.$$

Introducing the transformation (41), i.e., $\mathbf{G}_s^r = -(\mathbf{b}_r^r)^{-1}\mathbf{b}_s^r$, the evolution equation for $\mathbf{G}_s^r$, Eq. (57), is recovered. Regarding the description of the SIM, this shows that both the evolution equations for $\mathbf{b}^r$ and $\mathbf{G}_s^r$, Eqs. (33) and (57), provide an exact representation. Their iterative counterparts, Eqs. (38) and (58), are expected to provide an approximate description of similar accuracy for the same number of refinements; the possible differences in accuracy related to the initial guesses for the iterative procedures.

9. The computation of the matrix $\mathbf{R}_r^s$

The evolution of the slow amplitudes $\underline{F}^s$ is governed by the equation:

$$\frac{\mathrm{d}\underline{F}^s}{\mathrm{d}t} = \mathbf{\Lambda}_r^s \underline{F}^r + \mathbf{\Lambda}_s^s \underline{F}^s, \tag{61}$$

where

$$\mathbf{\Lambda}_r^s = \left(\frac{\mathrm{d}\mathbf{B}^s}{\mathrm{d}t} + \mathbf{B}^s\mathbf{J}\right)\mathbf{A}_r, \quad \mathbf{\Lambda}_s^s = \left(\frac{\mathrm{d}\mathbf{B}^s}{\mathrm{d}t} + \mathbf{B}^s\mathbf{J}\right)\mathbf{A}_s. \tag{62}$$

In order for the fast time scales not to have an influence on the evolution of $\underline{F}^s$ is desired that $\mathbf{\Lambda}_r^s = \mathbf{0}_r^s$. In that case, in accordance with the evolution equation for $\mathbf{B}^r$, Eq. (52), the following equation for $\mathbf{A}_r$ holds:

$$-\frac{\mathrm{d}\mathbf{A}_r}{\mathrm{d}t} + \mathbf{J}\mathbf{A}_r = \mathbf{A}_r\mathbf{\Lambda}_r^r. \tag{63}$$

Note that this equation is similar to the one governing the evolution of the CSP vectors $\mathbf{a}_r$, Eq. (34). Substituting from the definition of $\mathbf{A}_r$, Eq. (16), the following evolution equation for the matrix $\mathbf{R}_r^s$ is obtained:

$$\frac{\mathrm{d}\mathbf{R}_r^s}{\mathrm{d}t} + \mathbf{J}_r^s\left(\mathbf{I}_r^r - \mathbf{G}_s^r\mathbf{R}_r^s\right) - \mathbf{J}_s^s\mathbf{R}_r^s = -\mathbf{R}_r^s\mathbf{\Lambda}_r^r, \tag{64}$$

where $\mathbf{\Lambda}_r^r$ is defined by Eq. (55). Since Eq. (64) is stiff, $\mathbf{R}_r^s$ can be computed iteratively by the expression:

$$\mathbf{R}_r^s(j+1) = \left[\mathbf{J}_s^s\mathbf{R}_r^s(j) - \mathbf{J}_r^s\left(\mathbf{I}_r^r - \mathbf{G}_s^r(n)\mathbf{R}_r^s(j)\right) - \frac{\mathrm{d}\mathbf{R}_r^s(j)}{\mathrm{d}t}\right]\left(\mathbf{\Lambda}_r^r(n)\right)^{-1}, \tag{65}$$

where $\mathbf{\Lambda}_r^r(n) = \mathbf{J}_r^r - \mathbf{G}_s^r(n)\mathbf{J}_r^s$. Note that, starting with $\mathbf{R}_r^s(0) = \mathbf{0}_r^s$, Eq. (65) yields:

$$\mathbf{R}_r^s(1) = -\mathbf{J}_r^s\left(\mathbf{J}_r^r - \mathbf{G}_s^r(n)\mathbf{J}_r^s\right)^{-1},$$

i.e., the outcome of one block-power method iteration on $\mathbf{J}$ for computing the right eigenvectors, when starting with $\mathbf{A}_r = [\mathbf{I}_r^r, \mathbf{0}_s^r]^{\mathrm{T}}$ and using $\mathbf{B}^r = [\mathbf{I}_r^r, -\mathbf{G}_s^r(n)]$ [50].

10. Partitioning of the unknowns in fast and slow components

The optimal partition of the state vector $\underline{y} = (y^1, \ldots, y^N)$ into a fast and a slow component, i.e., $\underline{z} = (y^1, \ldots, y^M)$ and $\underline{s} = (y^{M+1}, \ldots, y^N)$ must be based on the quantities:

$$|\mathbf{b}^i \cdot \mathbf{e}_k y^k|, \tag{66}$$

$$|\mathbf{e}^k \cdot \mathbf{a}_i / y^k|, \tag{67}$$

where $i = 1, M$, $\mathbf{e}_k$ and $\mathbf{e}^k$ are N-dimensional column and row, respectively, vectors with all elements zero but the kth, which is set equal to 1. It was shown that the largest the magnitude of the quantity (66) the more significant is the participation of y^k in the ith component of Eq. (30) defining the SIM [28]. In addition, it was shown that the larger the magnitude of the quantity (67) the larger the influence of the ith time scale on y^k [28]. It was concluded that the M y^ks exhibiting the strongest (i) participation in the M components of Eq. (30) and (ii) influence from the M fastest time scales are those indicated by the largest elements of the CSP pointer, Eq. (32), which can be cast as [18,19]:

$$\underline{q}^r = \mathrm{diag}\left[\frac{1}{M}\mathbf{a}_r\mathbf{b}^r\right] = \frac{1}{M}\begin{bmatrix}\mathrm{diag}\left(\mathbf{a}_r^r\mathbf{b}_r^r\right)\\ \mathrm{diag}\left(\mathbf{a}_r^s\mathbf{b}_s^r\right)\end{bmatrix}.$$

Due to the orthogonality condition, Eq. (40):

$$\frac{1}{M}\left[\sum_{i=1,M}\left(\mathbf{a}_r^r\mathbf{b}_r^r\right)_i^i + \sum_{j=1,N-M}\left(\mathbf{a}_r^s\mathbf{b}_s^r\right)_j^j\right] = 1$$

so that the sum of all N components in $\underline{q}^r$ equal unity. The CSP pointer provides a measure of how parallel the N axes of the unknowns are to the M vectors in $\mathbf{a}_r$, along which the M fastest time scales act [18,19]. The M components of $\underline{y}$ corresponding to the largest (in magnitude) elements of $\underline{q}^r$ constitute the optimal set of unknowns to be computed from the algebraic equations (30) in terms of the remaining $N-M$ ones [18,19,28].

Having assumed that it is desired to parameterize the SIM by the $N-M$ last components of $\underline{y}$, the elements in the M diagonal entries of the $M\times M$ matrix $\mathbf{a}_r^r\mathbf{b}_r^r$ are expected to be among the largest. In particular, when all N components of $\underline{y}$ are fast, $\mathrm{diag}\left(\mathbf{a}_r^r\mathbf{b}_r^r\right) = \mathrm{O}(1)$ and $\mathrm{diag}\left(\mathbf{a}_r^s\mathbf{b}_s^r\right) = \mathrm{O}(1)$. However, when only the M components of $\underline{y}$ in $\underline{z}$ are fast and the $N-M$ components in $\underline{s}$ are slow, $\mathrm{diag}\left(\mathbf{a}_r^r\mathbf{b}_r^r\right) \to 1$ and $\mathrm{diag}\left(\mathbf{a}_r^s\mathbf{b}_s^r\right) \to 0$. In any case, these estimates of the magnitude of the matrix product $\mathbf{a}_r^r\mathbf{b}_r^r$ indicate that the matrices $\mathbf{b}_r^r$ and $\mathbf{a}_s^s$ are invertible (the latter using Eq. (40)). This means that, with proper partitioning of $\underline{y}$ in $\underline{z}$ and $\underline{s}$, the transformations (41) and (42) exist and the matrix $\mathbf{G}_s^r$ can be computed.

In terms of the basis vectors of the modified CSP algorithm, Eqs. (16) and (17), the CSP pointer takes the form:

$$\underline{Q}^r = \mathrm{diag}\left[\frac{1}{M}\mathbf{A}_r\mathbf{B}^r\right] = \frac{1}{M}\begin{bmatrix}\mathrm{diag}\left(\mathbf{I}_r^r - \mathbf{G}_s^r\mathbf{R}_r^s\right)\\ \mathrm{diag}\left(\mathbf{R}_r^s\mathbf{G}_s^r\right)\end{bmatrix}, \tag{68}$$

where the diagonal elements of the matrices $\mathbf{G}_s^r\mathbf{R}_r^s$ and $\mathbf{R}_r^s\mathbf{G}_s^r$ satisfy the relation:

$$\sum_{i=1,M}\left(\mathbf{G}_s^r\mathbf{R}_r^s\right)_i^i = \sum_{j=1,N-M}\left(\mathbf{R}_r^s\mathbf{G}_s^r\right)_j^j.$$

Since $\underline{Q}^r$ and $\underline{q}^r$ are identical, via the transformations (41) and (42), is expected that $\underline{Q}^r$ will identify the M fast components of $\underline{y}$ as accurately as $\underline{q}^r$ does. As a result, in the case where only the M components of $\underline{y}$ in $\underline{z}$ are fast, it is anticipated that the diagonal elements of $\mathbf{G}_s^r\mathbf{R}_r^s$ and $\mathbf{R}_r^s\mathbf{G}_s^r$ will be negligible:

$$\left|\left(\mathbf{G}_s^r\mathbf{R}_r^s\right)_i^i\right| \ll 1, \quad i = 1, M,$$
$$\left|\left(\mathbf{R}_r^s\mathbf{G}_s^r\right)_i^i\right| \ll 1, \quad i = 1, N-M.$$

11. Example: the Lindemann mechanism

Consider the problem:

$$y_t = \frac{z}{\varepsilon}(z-y) - y, \quad y(0) = y_*, \quad z_t = -\frac{z}{\varepsilon}(z-y), \quad z(0) = z_*, \tag{69}$$

where $0 < \varepsilon \ll 1$ and the subscript "t" denotes differentiation with time. This model problem has been employed for the study of association/dissociation kinetics [51–57].

Eqs. (69) are stiff, characterized by two time scales; one fast and one slow. A typical solution for y and z is shown in Fig. 1 for $\varepsilon = 10^{-3}$. The trajectory starts at $(y_*, z_*) = (1, 4)$, heading towards the fixed point $(y, z) = (0, 0)$. It can be shown that, after a fast initial transient which lasts for a period $\mathrm{O}(\varepsilon)$, under the action of the fast time scale the solution is attracted to a manifold defined by the relation $y \approx z - \varepsilon/2 + \varepsilon^2/(4z)$. On the manifold, the fast time scale is exhausted and the solution is directed towards the fixed point according to the slow scale. However, the manifold degenerates as y and z become $\mathrm{O}(\varepsilon)$, re-emerging further in time as $\varepsilon y \approx z^2 + z^3/\varepsilon - 5z^5/\varepsilon^3$ and extending in this form to the fixed point.

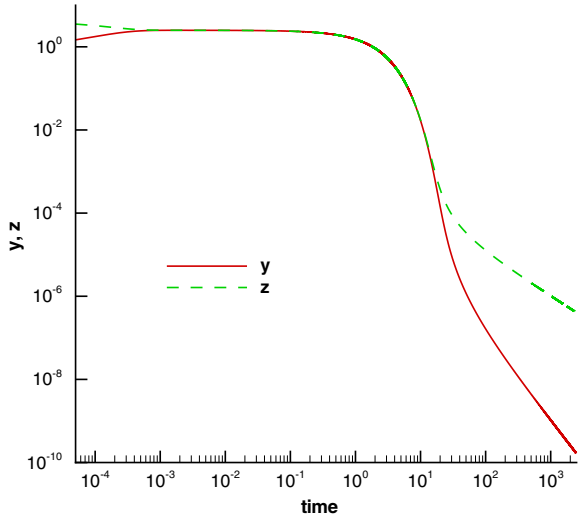

Fig. 1. The solution for $\varepsilon = 10^{-3}$, $y_* = 1$ and $z_* = 4$.

The two different representations of the manifold and their validity are clearly displayed in Fig. 2, where the ratios:

$$R_1 = \frac{y}{z}, \quad R_2 = \frac{\varepsilon y}{z^2}$$

are plotted. It is shown that after an initial short period of $O(\varepsilon)$ and for a period that lasts up to $t = O(10^1)$, $R_1 \to 1$ and $R_2 \ll 1$, signaling the fact that the manifold has the $y \approx z$ representation. From about $t = O(5 \times 10^1)$ and on, $R_1 \ll 1$ and $R_2 \to 1$, indicating that the manifold has now the $\epsilon y \approx z^2$ representation. In the intermediate period, neither of the representations hold; a natural consequence due to the degeneracy of the manifold.

The reasons for the generation of the manifold in the first part of the outer region, its degeneration and subsequent re-generation in the second part are all manifested in the evolution of the two time scales of the system. Approximating the time scales with the inverse of the eigenvalues of the Jacobian of the RHS of Eq. (69) yield:

$$\tau_1 = \frac{1}{|\lambda_1|} = \frac{2\varepsilon}{d + \sqrt{b}}, \quad \tau_2 = \frac{1}{|\lambda_2|} = \frac{2\varepsilon}{d - \sqrt{b}},$$

where the subscripts "1" and "2" denote the fast and slow, respectively, time scales and where:

$$b = (y - z + \varepsilon)^2 + 8z^2 - 4zy, \quad d = 3z - y + \varepsilon.$$

Fig. 3 shows that, in the first part of the outer region where y and z are $O(1)$ and the $y \approx z$ manifold emerges, the time scale gap developed:

$$\tau_1 \to \varepsilon/2z, \quad \tau_2 \to 2 \tag{70}$$

disappears during the period where y and z become smaller than $O(\varepsilon)$ and the manifold degenerates, yielding:

$$\tau_1 = O(1), \quad \tau_2 = O(1) \tag{71}$$

after which period, as y and z become much smaller than $O(\varepsilon)$ and the $\epsilon y \approx z^2$ manifold emerges, a new time scale gap is established tending to:

$$\tau_1 \to 1, \quad \tau_2 \to \varepsilon/2z. \tag{72}$$

In the following, Eqs. (69) will be analyzed using the classical singular perturbation analysis, the original CSP algorithm and its modified version presented above.

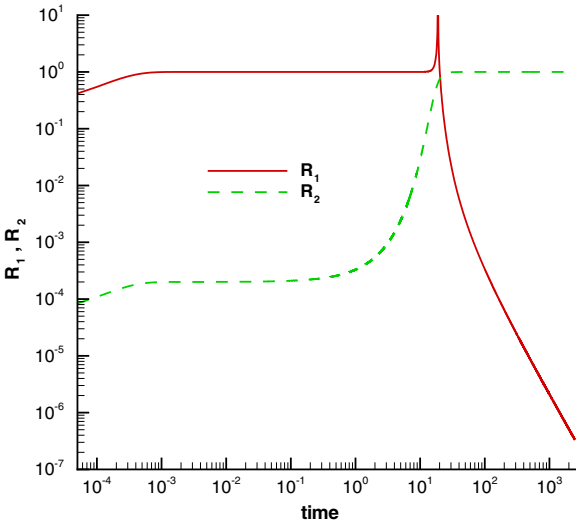

Fig. 2. The two types of manifolds generated in the outer region; $R_1 = 1$ for $O(\epsilon) < t < O(10^1)$ and $R_2 = 1$ for $O(2 \times 10^1) < t$.

11.1. Singular perturbation analysis

Assuming y_* and z_* are both O(1), the fast and slow time scales, Eqs. (70), are $O(\varepsilon)$ and O(1) respectively, throughout the fast initial transient and the first part of the outer region. In that period, both variables are affected by the fast time scale; i.e., they are both "fast". However, their sum $w = y + z$ is affected by the slow time scale only and a transformed system can be stated as:

$$w_t = -w + z, \quad z_t = -\frac{z}{\varepsilon}(2z - w) \tag{73}$$

so that only variable z is associated with the fast time scale. For the particular problem examined here, classical singular perturbation analysis fails if the original form (69) of the problem is considered, but works if the transformed one (73) is employed instead. By doing so and assuming that the variables can be expanded in terms of the time scale ratio $\tau_1/\tau_2 = O(\varepsilon)$, as:

$$y = y_0 + \varepsilon y_1 + O(\varepsilon^2), \quad z = z_0 + \varepsilon z_1 + O(\varepsilon^2) \tag{74}$$

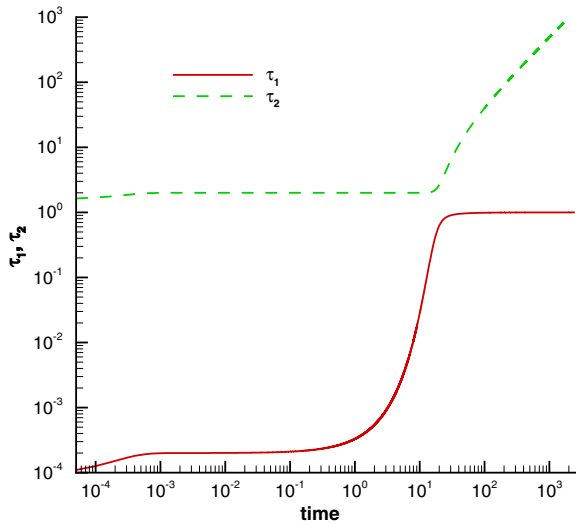

Fig. 3. The two time scales; $\varepsilon = 10^{-3}$.

yields:

$$y_0 - z_0 = 0, \quad y_{0t} = -\frac{1}{2}y_0, \quad z_{0t} = -\frac{1}{2}z_0, \tag{75}$$

$$y_1 - z_1 + \frac{1}{2} = 0, \quad y_{1t} = -\frac{1}{2}z_1 + \frac{1}{4}, \quad z_{1t} = -\frac{1}{2}z_1 + \frac{1}{4}. \tag{76}$$

From Eqs. (74)–(76), follows that the SIM and the simplified non-stiff problem in the outer region are described to $O(\varepsilon^1)$ accuracy by the expressions:

$$y - z \approx 0, \tag{77}$$

$$\begin{bmatrix} y \\ z \end{bmatrix}_t \approx \begin{bmatrix} 1 \\ 1 \end{bmatrix} \left(-\frac{1}{2}z \right) \tag{78}$$

and to $O(\varepsilon^2)$ accuracy by the expressions:

$$y - z + \frac{\varepsilon}{2} \approx 0, \tag{79}$$

$$\begin{bmatrix} y \\ z \end{bmatrix}_t \approx \begin{bmatrix} 1 \\ 1 \end{bmatrix} \left(-\frac{1}{2}z + \frac{\varepsilon}{4} \right). \tag{80}$$

However, the expansion (74) fails when the variables y and z become small, as the system approaches the fixed point. In fact, in the period where $y \approx z = O(\varepsilon)$ there is no time scale separation, i.e., $\tau_1/\tau_2 = O(1)$, Eq. (71). In that period, the scalings $y = \varepsilon\bar{y}$ and $z = \varepsilon\bar{z}$ make the small parameter ε disappear from the governing equations, the problem is not stiff and the SIM degenerates. Further in time, when y and z become smaller than $O(\varepsilon)$ and a new time scale gap appears, Eq. (72), a different representation of the SIM emerges. This can be found, by introducing the scalings:

$$y = \frac{\alpha^2}{\varepsilon}Y, \quad z = \alpha Z, \quad t = \frac{\varepsilon}{\alpha}\tau, \tag{81}$$

where $0 < \alpha \ll \varepsilon$ is a small parameter indicative of the magnitude of z. With these transformations, the given system (69) is cast as:

$$Y_\tau = \frac{Z}{\omega}(Z - \omega Y) - \frac{1}{\omega}Y, \quad Z_\tau = -Z(Z - \omega Y), \tag{82}$$

where $\omega = \alpha/\varepsilon \ll 1$. Assuming that the new variables can be expanded as:

$$Y = Y_0 + \omega Y_1 + O(\omega^2), \quad Z = Z_0 + \omega Z_1 + O(\omega^2) \tag{83}$$

and substituting in Eqs. (82) yields:

$$Y_0 = Z_0^2, \quad Y_{0\tau} = -2Z_0^3, \quad Z_{0\tau} = -Z_0^2, \tag{84}$$

$$Y_1 = Z_0\left(Z_0^2 + 2Z_1\right), \quad Y_{1\tau} = -Z_0^2\left(Z_0^2 + 6Z_1\right), \quad Z_{1\tau} = Z_0\left(Z_0^2 - 2Z_1\right). \tag{85}$$

From Eqs. (83)–(85), follows that the SIM and the simplified non-stiff problem in the second part of the outer region are described to $O(\omega^1)$ accuracy by the expressions:

$$Y \approx Z^2, \tag{86}$$

$$\begin{bmatrix} Y \\ Z \end{bmatrix}_\tau \approx \begin{bmatrix} -2Z^3 \\ -Z^2 \end{bmatrix} \tag{87}$$

and to $O(\omega^2)$ accuracy by the expressions:

$$Y \approx Z^2 + \omega Z^3, \tag{88}$$

$$\begin{bmatrix} Y \\ Z \end{bmatrix}_\tau \approx \begin{bmatrix} -2Z^3 - \omega Z^4 \\ -Z^2 + \omega Z^3 \end{bmatrix}. \tag{89}$$

In terms of the original variables, Eqs. (88) and (89) become:

$$\varepsilon y \approx z^2 + \frac{1}{\varepsilon} z^3, \tag{90}$$

$$\begin{bmatrix} y \\ z \end{bmatrix}_t \approx \begin{bmatrix} -\frac{2}{\varepsilon^2} z^3 - \frac{1}{\varepsilon^3} z^4 \\ -\frac{1}{\varepsilon} z^2 + \frac{1}{\varepsilon^2} z^3 \end{bmatrix}. \tag{91}$$

When compared to Eqs. (79) and (80), Eqs. (90) and (91) show that completely different representations of the manifold and the simplified system result in the first and second parts of the outer region.

11.2. The CSP method

According to CSP, the system (69) is cast in the form:

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{bmatrix} y \\ z \end{bmatrix} = \mathbf{a}_1(k,m) f^1(k,m) + \mathbf{a}_2(k,m) f^2(k,m),$$

where

$$f^1(k,m) = \mathbf{b}^1(k,m)\underline{g}, \quad f^2(k,m) = \mathbf{b}^2(k,m)\underline{g}$$

and (k,m) are the number of $\mathbf{b}^r$- and $\mathbf{a}_r$-refinements. The SIM and the simplified non-stiff problem are then described by the equations:

$$f^1(k,m) = 0, \tag{92}$$

$$\frac{\mathrm{d}}{\mathrm{d}t} \begin{bmatrix} y \\ z \end{bmatrix} = \mathbf{a}_2(k,m) f^2(k,m). \tag{93}$$

Starting the CSP refinements with the following sets of basis vectors:

$$\mathbf{a}_1(0,0) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \mathbf{a}_2(0,0) = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

$$\mathbf{b}^1(0,0) = [1,0], \quad \mathbf{b}^2(0,0) = [0,1]$$

one $\mathbf{b}^r$- and one $\mathbf{a}_r$-refinement yield:

$$\mathbf{a}_1(1,1) = \begin{bmatrix} (z+\varepsilon)q \\ -zq \end{bmatrix}, \quad \mathbf{a}_2(1,1) = \begin{bmatrix} p \\ 1 \end{bmatrix}, \tag{94}$$

$$\mathbf{b}^1(1,1) = [1,-p], \quad \mathbf{b}^2(1,1) = [zq, 1-zqp], \tag{95}$$

$$f^1(1,1) = \frac{z}{\varepsilon}(z-y)(1+p) - y, \tag{96}$$

$$f^2(1,1) = -\frac{z^2(z+\varepsilon)}{(z+\varepsilon)^2 + z(2z-y)}, \tag{97}$$

where $p = (2z-y)/(z+\varepsilon)$, $q = (z+\varepsilon)/((z+\varepsilon)^2 + z(2z-y))$. An additional $\mathbf{b}^r$-refinement yields:

$$\mathbf{a}_1(2,1) = \begin{bmatrix} (z+\varepsilon)q \\ -zq \end{bmatrix}, \quad \mathbf{a}_2(2,1) = \begin{bmatrix} sv \\ su \end{bmatrix}, \tag{98}$$

$$\mathbf{b}^1(2,1) = \frac{u}{w}[u,-v], \quad \mathbf{b}^2(2,1) = [zq, 1-zqp], \tag{99}$$

$$f^1(2,1) = \frac{u}{w}\left(\frac{z}{\varepsilon}(z-y)(u+v) - uy\right), \tag{100}$$

$$f^2(2,1) = -\frac{z^2(z+\varepsilon)}{(z+\varepsilon)^2 + z(2z-y)}, \tag{101}$$

where

$$u = z(3z - y) + 2\varepsilon z + \varepsilon^2,$$
$$v = (z^2(7z - 5y) + \varepsilon(y^2 - 10yz + 11z^2) + 2\varepsilon^2(z - y))/(z + \varepsilon),$$
$$w = 2z^3(5z - 3y) + \varepsilon z(y^2 - 12yz + 19z^2) + \varepsilon^2 z(10z - 3y) + 4\varepsilon^3 z + \varepsilon^4,$$
$$s = 1/(zqv + (1 - zqp)u).$$

If leading order accuracy is desired, the basis vectors (94) and (95), obtained with one $\mathbf{a}_r$- and one $\mathbf{b}^r$-refinements, must be employed along with the relation $f^1(1,1) = 0$. In that case, when $y = O(1)$ and $z = O(1)$ and the solution lies on the SIM expressed by the relation $y = z - \varepsilon/2 + \varepsilon^2/(4z) + O(\varepsilon^3 K)$, where the term $O(\varepsilon^3 K)$ includes all higher order terms, the CSP vectors, the CSP amplitudes and the simplified problem are:

$$\mathbf{a}_1(1,1) = \frac{1}{2}\begin{bmatrix} 1 + \frac{3\varepsilon}{4z} - \frac{5\varepsilon^2}{16z^2} \\ -1 + \frac{\varepsilon}{4z} + \frac{\varepsilon^2}{16z^2} \end{bmatrix} + O(\varepsilon^3), \quad \mathbf{a}_2(1,1) = \begin{bmatrix} 1 - \frac{\varepsilon}{2z} + \frac{\varepsilon^2}{4z^2} \\ 1 \end{bmatrix} + O(\varepsilon^3),$$
$$\mathbf{b}^1(1,1) = \left[1, -1 + \frac{\varepsilon}{2z} - \frac{\varepsilon^2}{4z^2}\right] + O(\varepsilon^3),$$
$$\mathbf{b}^2(1,1) = \frac{1}{2}\left[1 - \frac{\varepsilon}{4z} - \frac{\varepsilon^2}{16z^2}, 1 + \frac{3\varepsilon}{4z} - \frac{5\varepsilon^2}{16z^2}\right] + O(\varepsilon^3),$$
$$f^1(1,1) = -\frac{\varepsilon}{8} + O(\varepsilon^2), \quad f^2(1,1) = -\frac{z}{2} + \frac{\varepsilon}{8} + \frac{\varepsilon^2}{32z} + O(\varepsilon^3),$$
$$\frac{d}{dt}\begin{bmatrix} y \\ z \end{bmatrix} = \begin{bmatrix} -\frac{z}{2} + \frac{3\varepsilon}{8} - \frac{5\varepsilon^2}{32z} \\ -\frac{z}{2} + \frac{\varepsilon}{8} + \frac{\varepsilon^2}{32z} \end{bmatrix} + O(\varepsilon^3).$$

When $z = \alpha Z$, $y = \alpha^2 Y/\varepsilon$, $t = \varepsilon\tau/\alpha$, $\omega = \alpha/\varepsilon \ll 1$ and the solution lies on the SIM expressed by the relation $Y = Z^2 + \omega Z^3 + O(\omega^3\Lambda)$, where the term $O(\omega^3\Lambda)$ includes all higher order terms, the related CSP quantities become:

$$\mathbf{a}_1(1,1) = \begin{bmatrix} 1 - 2\omega^2 Z^2 \\ -\omega^2 Z \end{bmatrix} + O(\omega^3), \quad \mathbf{a}_2(1,1) = \begin{bmatrix} 2Z - 3\omega Z^2 + 2\omega^2 Z^3 \\ 1 \end{bmatrix} + O(\omega^3),$$
$$\mathbf{b}^1(1,1) = \left[1, -2Z + 3\omega Z^2 - 2\omega^2 Z^3\right] + O(\omega^3),$$
$$\mathbf{b}^2(1,1) = \left[\omega^2 Z, 1 - 2\omega^2 Z^2\right] + O(\omega^3),$$
$$f^1(1,1) = -6\omega Z^4 + 8\omega^2 Z^5 + O(\omega^3), \quad f^2(1,1) = -Z^2 + \omega Z^3 + \omega^2 Z^4 + O(\omega^3),$$
$$\frac{d}{d\tau}\begin{bmatrix} Y \\ Z \end{bmatrix} = \begin{bmatrix} -2Z^3 + 5\omega Z^4 - 3\omega^2 Z^5 \\ -Z^2 + \omega Z^3 + \omega^2 Z^4 \end{bmatrix} + O(\omega^3).$$

It is seen that CSP recovers the singular perturbation analysis leading order results for both parts in the outer region, Eqs. (77), (78) and (86), (87).

If second order accuracy is desired, the basis vectors (98) and (99), obtained with one $\mathbf{a}_r$- and two $\mathbf{b}^r$-refinements, must be employed along with the relation $f^1(2,1) = 0$. In that case, when $y = O(1)$, $z = O(1)$ and the solution lies on the SIM $y = z - \varepsilon/2 + \varepsilon^2/(4z) + O(\varepsilon^3 K)$, the CSP vectors, the CSP amplitudes and the simplified problem become:

$$\mathbf{a}_1(2,1) = \begin{bmatrix} 1 + \frac{3\varepsilon}{4z} - \frac{5\varepsilon^2}{16z^2} \\ -1 + \frac{\varepsilon}{4z} + \frac{\varepsilon^2}{16z^2} \end{bmatrix} + O(\varepsilon^3), \quad \mathbf{a}_2(2,1) = \begin{bmatrix} 1 - \frac{\varepsilon}{4z} + \frac{\varepsilon^2}{8z^2} \\ 1 - \frac{\varepsilon}{4z} + \frac{3\varepsilon^2}{8z^2} \end{bmatrix} + O(\varepsilon^3),$$
$$\mathbf{b}^1(2,1) = \frac{1}{2}\left[1 - \frac{\varepsilon}{4z} + \frac{3\varepsilon^2}{8z^2}, -1 + \frac{\varepsilon}{4z} - \frac{\varepsilon^2}{8z^2}\right] + O(\varepsilon^3),$$
$$\mathbf{b}^2(2,1) = \frac{1}{2}\left[1 - \frac{\varepsilon}{4z} - \frac{\varepsilon^2}{16z^2}, 1 + \frac{3\varepsilon}{4z} - \frac{5\varepsilon^2}{16z^2}\right] + O(\varepsilon^3),$$

$$f^1(2,1) = -(\frac{3}{z} + Kz)\varepsilon^2 + \mathrm{O}(\varepsilon^3), \quad f^2(1,1) = -\frac{z}{2} + \frac{\varepsilon}{8} + \frac{\varepsilon^2}{32z} + \mathrm{O}(\varepsilon^3),$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} y \\ z \end{bmatrix} = \begin{bmatrix} -\frac{z}{2} + \frac{\varepsilon}{4} + \frac{\varepsilon^2}{16z} \\ -\frac{z}{2} + \frac{\varepsilon}{4} - \frac{3\varepsilon^2}{16z} \end{bmatrix} + \mathrm{O}(\varepsilon^3).$$

When $z = \alpha Z$, $y = \alpha^2 Y/\varepsilon$, $t = \varepsilon\tau/\alpha$, $\omega = \alpha/\varepsilon \ll 1$ and the solution lies on the SIM $Y = Z^2 + \omega Z^3 + \mathrm{O}(\omega^3\Lambda)$, the related CSP quantities yield:

$$\mathbf{a}_1(2,1) = \begin{bmatrix} 1 - 2\omega^2 Z^2 \\ -\omega^2 Z \end{bmatrix} + \mathrm{O}(\omega^3), \quad \mathbf{a}_2(2,1) = \begin{bmatrix} 2Z + 3\omega Z^2 - 24\omega^2 Z^3 \\ 1 \end{bmatrix} + \mathrm{O}(\omega^3),$$

$$\mathbf{b}^1(2,1) = \left[1, -2Z - 3\omega Z^2 + 24\omega^2 Z^3\right] + \mathrm{O}(\omega^3),$$

$$\mathbf{b}^2(2,1) = \left[\omega^2 Z, 1 - 2\omega^2 Z^2\right] + \mathrm{O}(\omega^3),$$

$$f^1(2,1) = -24\omega^2 Z^5 + \mathrm{O}(\omega^3), \quad f^2(2,1) = -Z^2 + \omega Z^3 + \omega^2 Z^4 + \mathrm{O}(\omega^3),$$

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\begin{bmatrix} Y \\ Z \end{bmatrix} = \begin{bmatrix} -2Z^3 - \omega Z^4 + 29\omega^2 Z^5 \\ -Z^2 + \omega Z^3 + \omega^2 Z^4 \end{bmatrix} + \mathrm{O}(\omega^3).$$

Again, it is seen that CSP recovers the singular perturbation analysis second order results for both parts in the outer region, Eqs. (79), (80) and (88), (89).

The evolution of the amplitudes $f^1(1,1)$ and $f^1(2,1)$ with time, computed from Eqs. (96) and (100), respectively, is displayed in Fig. 4, where the improvement in approximating the SIM with an additional $\mathbf{b}^r$-refinement is clearly demonstrated. In particular, it is shown that better accuracy is achieved when employing the relation $f^1(2,1) = 0$ than when employing the relation $f^1(1,1) = 0$, since the value of $f^1(2,1)$ is indeed smaller than that of $f^1(1,1)$, in accordance to the existing time scale separation. The fuzziness in $f^1(2,1)$ in the first part of the outer region, where y and z are both O(1), is due to the truncation error caused by large cancellations among the different terms in the expression of f^1. In the second part of the outer region the cancellations are milder; the decay of the magnitude of the different terms making-up f^1 contributing to its demise.

Regarding the CSP pointer, Eq. (32), the CSP basis vectors (98) and (99) provide:

$$\underline{q}^1 = \mathrm{diag}[\mathbf{a}_1\mathbf{b}^1] = [d^1, d^2] = \left[\frac{(z+\varepsilon)qu^2}{w}, \frac{zquv}{w}\right], \tag{102}$$

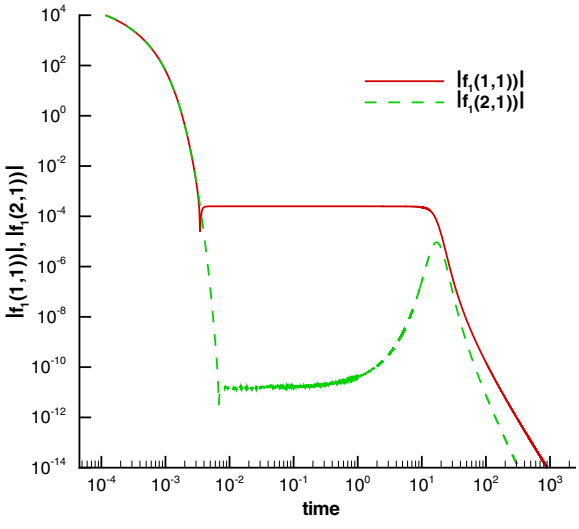

Fig. 4. The CSP fast amplitude f^1 (1 or 2 $\mathbf{b}^r$- and 1 $\mathbf{a}_r$-refinements); $\varepsilon = 10^{-3}$.

where $d^1 + d^2 = 1$. In the first part of the outer region, where the SIM takes the form $z \approx y + \varepsilon/2$, Eq. (102) reduces to:

$$\underline{q}^1 \approx \left[\frac{1}{2}, \frac{1}{2}\right]. \tag{103}$$

Noting that $\underline{y} = [y, z]^T$, Eq. (103) indicates that both variables, y and z, can be identified as fast. However, in the second part of the outer region, where the SIM takes the form $\varepsilon y \approx z^2 + z^3/\varepsilon$, Eq. (102) reduces to:

$$\underline{q}^1 \approx [1, 0] \tag{104}$$

indicating that only variable y can be identified as fast. The evolution of the two components of $\underline{q}^1 = [d^1, d^2]$ is displayed in Fig. 5, where the adjustment of the CSP pointer in the first and the second parts of the outer region, as noted by Eqs. (103) and (104), is clearly displayed.

The problems that might arise when a wrong identification for the fast variable is made can be illustrated by considering the second part of the outer region where y is the fast variable, z is the slow, $\varepsilon y \approx z^2 + z^3/\varepsilon$ and $y \ll \varepsilon$, $z \ll \varepsilon$. In terms of the original variables, if y is chosen as the fast variable Eqs. (92) and (93) reduce to:

$$y \approx \frac{z^2}{\varepsilon}, \tag{105}$$

$$\frac{dz}{dt} \approx -y \tag{106}$$

while if z is chosen, instead, they reduce to:

$$z \approx \sqrt{\varepsilon y}, \tag{107}$$

$$\frac{dy}{dt} \approx -\frac{2yz}{\varepsilon}. \tag{108}$$

Introducing an error, say ζ:

$$z = z_o + \zeta, \quad y = y_o + \zeta$$

in the computation of z and y, originating from the differential equations (106) and (108), respectively, will produce the following perturbations in y and z, as computed from the related algebraic equations (105) and (107):

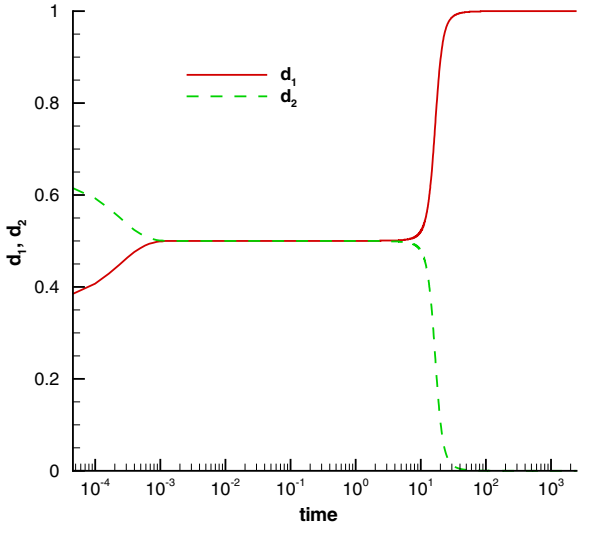

Fig. 5. The two components of the CSP pointer $\underline{q}^1 = [d^1, d^2]$; $\varepsilon = 10^{-3}$.

$$y = y_o\left(1+\frac{2}{z_o}\zeta\right), \quad z = z_o\left(1+\frac{1}{2y_o}\zeta\right),$$

where $\varepsilon y_o \approx z_o^2$. Since $y_o << z_o$, it is clear that the same error from the differential equations (106) and (108) will be amplified much more in the algebraic equation (107) than in Eq. (105); i.e., when the wrong choice is made for the fast variable and z is selected instead of y.

11.3. The modified CSP algorithm

According to the modified CSP algorithm, a fast and a slow variable selection must be made, prior to its application. Here, the cases where each of the two variables, y or z, is identified as the fast one will be considered, in order to display the problems that might arise from a wrong selection.

A key ingredient of the modified CSP algorithm is the computation of $\mathbf{G}_s^r$ either from the differential equation (57) or the iterative equation (58). Note that the correct expressions in the first part of the outer region, where the SIM is described by the relation $y = z - \varepsilon/2 + \varepsilon^2/(4z) + \mathrm{O}(\varepsilon^3)$, are:

$$G_y = 1 + \left(\frac{\varepsilon}{2z}\right)^2 + \mathrm{O}(\varepsilon^3), \tag{109}$$

$$G_z = 1 - \left(\frac{\varepsilon}{2z}\right)^2 + \mathrm{O}(\varepsilon^3), \tag{110}$$

where $G_y = \partial z/\partial y$ and $G_z = \partial y/\partial z$. In the second part of the outer region, where the SIM is described by the relation $Y = Z^2 + \omega Z^3 + \mathrm{O}(\omega^3)$, the correct expressions are:

$$G_Y = \frac{1}{2Z} - \frac{3\omega}{4} + \mathrm{O}(\omega^2), \tag{111}$$

$$G_Z = 2Z + 3\omega Z^2 + \mathrm{O}(\omega^2), \tag{112}$$

where $G_Y = \partial Z/\partial Y$, $G_Z = \partial Y/\partial Z$, $G_Y = \omega G_y$ and $G_Z = G_z/\omega$.

Assuming that z is the fast variable and y is the slow one, the differential equation (57) and the iterative relation (58) for G_y take the form:

$$\frac{\mathrm{d}G_y}{\mathrm{d}t} + \left(\frac{-z-\varepsilon}{\varepsilon}\right)G_y - \frac{z}{\varepsilon} = -\left(\frac{2z-y}{\varepsilon}(1+G_y)\right)G_y, \tag{113}$$

$$G_y(n+1) = \frac{\varepsilon}{(2z-y)(1+G_y(n))}\left(\left(\frac{z+\varepsilon}{\varepsilon}\right)G_y(n) + \frac{z}{\varepsilon} - \frac{\mathrm{d}G_y(n)}{\mathrm{d}t}\right). \tag{114}$$

Starting with $G_y(0) = 0$, Eq. (114) yields:

$$G_y(1) = \frac{z}{2z-y}, \quad G_y(2) = \frac{z}{2z-y}\left(1 - \frac{p_1}{q_1}\right), \tag{115}$$

where $p_1 = z^2 - y^2 - 2\varepsilon z$ and $q_1 = (2z-y)(3z-y)$.

In the first part of the outer region, where $y \approx \mathrm{O}(1)$, $z \approx \mathrm{O}(1)$ and $y = z - \varepsilon/2 + \varepsilon^2/(4z) + \mathrm{O}(\varepsilon^3)$, Eqs. (115) yield:

$$G_y(1) = 1 - \frac{\varepsilon}{2z} + \mathrm{O}(\varepsilon^2), \tag{116}$$

$$G_y(2) = 1 + \left(\frac{\varepsilon}{2z}\right)^2 + \mathrm{O}(\varepsilon^3). \tag{117}$$

When compared to the correct expression, Eq. (109), Eqs. (116) and (117) are $\mathrm{O}(\varepsilon^1)$ and $\mathrm{O}(\varepsilon^2)$ accurate; each iteration with Eq. (114) adding one degree of accuracy. In the second part of the outer region, where $y = \varepsilon\omega^2 Y$, $z = \varepsilon\omega Z$ and $Y = Z^2 + \omega Z^3 + \mathrm{O}(\omega^3)$, Eqs. (115) yield:

$$G_Y(1) = \frac{\omega}{2} + \frac{\omega^2 Z}{4} + \mathrm{O}(\omega^3), \tag{118}$$

$$G_Y(2) = \frac{1}{6Z} + \frac{23}{36}\omega + \mathrm{O}(\omega^2) \tag{119}$$

showing that two iterations with Eq. (114) fail to produce even leading order accuracy. However, more iterations with Eq. (114) produce convergence:

$$G_Y(3) = \frac{1}{2Z} - \frac{11}{4}\omega + \mathrm{O}(\omega^2), \tag{120}$$

$$G_Y(4) = \frac{1}{2Z} - \frac{3}{4}\omega + \mathrm{O}(\omega^2), \tag{121}$$

where $G_Y(3)$ and $G_Y(4)$ being $\mathrm{O}(\omega^1)$ and $\mathrm{O}(\omega^2)$ accurate when compared to the correct value, Eq. (111).

Alternatively, assuming that y is the fast variable and z is the slow one, the differential equation (57) and the iterative relation (58) for G_z take the form:

$$\frac{\mathrm{d}G_z}{\mathrm{d}t} - \frac{2z-y}{\varepsilon}(1+G_z) = -\left(\frac{z+\varepsilon}{\varepsilon} + \frac{z}{\varepsilon}G_z\right)G_z, \tag{122}$$

$$G_z(n+1) = \frac{\varepsilon}{z(1+G_z(n))+\varepsilon}\left(\frac{(2z-y)}{\varepsilon}(1+G_z(n)) - \frac{\mathrm{d}G_z(n)}{\mathrm{d}t}\right). \tag{123}$$

Starting with $G_z(0) = 0$, Eq. (123) yields:

$$G_z(1) = \frac{2z-y}{z+\varepsilon}, \quad G_z(2) = \frac{2z-y}{z+\varepsilon}\left(1+\frac{p_2}{q_2}\right), \tag{124}$$

where

$$p_2 = z(z^2-y^2) + \varepsilon(7z^2 - 8yz + y^2) + \varepsilon^2 y,$$

$$q_2 = (2z-y)(z(3z-y) + 2\varepsilon z + \varepsilon^2).$$

In the first part of the outer region, where $y \approx \mathrm{O}(1)$, $z \approx \mathrm{O}(1)$ and $y = z - \varepsilon/2 + \varepsilon^2/(4z) + \mathrm{O}(\varepsilon^3)$, Eqs. (124) yield:

$$G_z(1) \approx 1 - \frac{\varepsilon}{2z} + \mathrm{O}(\varepsilon^2), \tag{125}$$

$$G_z(2) = 1 - \left(\frac{\varepsilon}{2z}\right)^2 + \mathrm{O}(\varepsilon^3) \tag{126}$$

showing that each iteration with Eq. (123) adds one degree of accuracy, towards the correct expression given by Eq. (110). In the second part of the outer region, where $y = \varepsilon\omega^2 Y$, $z = \varepsilon\omega Z$ and $Y = Z^2 + \omega\, Z^3 + \mathrm{O}(\omega^3 \mathbf{\Lambda})$, Eqs. (124) yield:

$$G_Z(1) = 2Z - \omega(2Z^2 - Y) + \mathrm{O}(\omega^2), \tag{127}$$

$$G_Z(2) = 2Z + \omega(5Z^2 - 2Y) + \mathrm{O}(\omega^2) \tag{128}$$

each iteration with Eq. (123) adding again one degree of accuracy; $G_Z(1)$ and $G_Z(2)$ being within $\mathrm{O}(\omega^1)$ and $\mathrm{O}(\omega^2)$ the correct value, Eq. (112).

The analytical results presented above were corroborated by computational ones based on the numerical (4-th order accuracy) and approximate solution of the pertinent differential equations; namely Eqs. (113) and (114) for the case where the variable z is considered the fast variable or Eqs. (122) and (123) for the case where the variable y is considered the fast one.

Fig. 6 shows the evolution of G_y computed numerically from the differential equation (113), denoted as $G_{y,\mathrm{exact}}$, and approximately from the iterative equation (114) for two levels of accuracy; $G_y(1)$ and $G_y(2)$ from Eqs. (115). It is shown that, within a period of $\mathrm{O}(\varepsilon)$, all three quantities converge to the appropriate value, set by Eq. (109), for the first part of the outer region, where $y \approx z - \varepsilon/2$. However, when the solution evolves in the second part of the outer region, where $\varepsilon y \approx z^2$, only $G_{y,\mathrm{exact}}$ follows the correct value, set by Eq. (111). In contrast, $G_y(1)$ diverges, while $G_y(2)$ follows the right trend, staying far from the expected accuracy.

Fig. 7 shows the evolution of G_z computed numerically from the differential equation (122), denoted as $G_{z,\mathrm{exact}}$, and approximately from the iterative equation (123) for two levels of accuracy; $G_z(1)$ and $G_z(2)$ from Eqs. (124). Again, it is shown that, within a period of $\mathrm{O}(\varepsilon)$, all three quantities converge to the appropriate

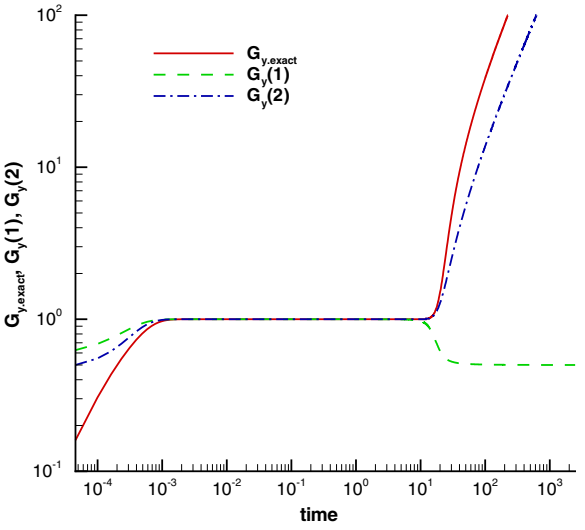

Fig. 6. G_y vs. time as computed from the differential equation (113), $G_{y,\text{exact}}$, and the iterative expression (114), $G_y(1)$ and $G_y(2)$; $\varepsilon = 10^{-3}$.

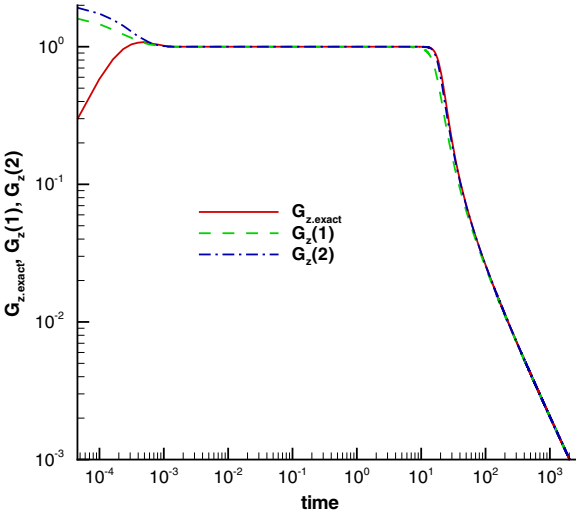

Fig. 7. G_z vs. time as computed from the differential equation (122), $G_{z,\text{exact}}$, and the expression (123), $G_z(1)$ and $G_z(2)$; $\varepsilon = 10^{-3}$.

value, set by Eq. (110), for the first part of the outer region. This agreement persists when the solution evolves in the second part of the outer region, where the appropriate value is set by Eq. (112).

Apparently, in the first part of the outer region where both variables are fast, it does not matter whether the G_y or the G_z formulation is employed. As a result, both iterative expressions, Eqs. (114) and (123), provide equally valid results. In contrast, in the second part where only y is a fast variable, the G_y formulation needs more iterations than the G_z formulation does to obtain similar order accuracy.

These developments are reflected in the evolution of the fast amplitude, defined according to Eq. (19) as $F^1 = \mathbf{B}^1 \underline{g} = (g^r - G_s^r g^s)$. In the case where z is taken as the fast variable, the first two levels of G_y approximation produce:

$$F^1_{z,\text{fast}}(1) = \left(z_t - G_y(1) y_t\right), \quad F^1_{z,\text{fast}}(2) = \left(z_t - G_y(2) y_t\right), \tag{129}$$

while in the case where y is taken as the fast variable, the first two levels of G_z approximation produce:

$$F^1_{y,\text{fast}}(1) = (y_t - G_z(1) z_t), \quad F^1_{y,\text{fast}}(2) = (y_t - G_z(2) z_t). \tag{130}$$

The evolution of $F^1_{z,\mathrm{fast}}(1)$ and $F^1_{z,\mathrm{fast}}(2)$ is displayed in Fig. 8, while that of $F^1_{y,\mathrm{fast}}(1)$ and $F^1_{y,\mathrm{fast}}(2)$ is displayed in Fig. 9. It is shown that during the first part of the outer region, where $y \approx z - \varepsilon/2$ holds, both G_y and G_z formulations produce similar results; i.e., same rate of decay for both $F^1(1)$ and $F^1(2)$ (the decay of the latter lasting for a longer time), and similar accuracy improvement from $F^1(1)$ to $F^1(2)$.

This agreement is not true in the second part of the outer region, where $\varepsilon y \approx z^2$ holds. Now, only the G_z formulation allows for the proper decay for both $F^1(1)$ and $F^1(2)$ and for the accuracy improvement from F^1 (1) to $F^1(2)$. In comparison, the G_y formulation allows for slower decay and little accuracy improvement.

Figs. 10 and 11 compare the leading and second order accurate fast amplitude as computed from the CSP algorithm, $f^1(1,1)$ and $f^1(2,1)$, and from the modified CSP, $F^1_{z,\mathrm{fast}}(1)$ and $F^1_{z,\mathrm{fast}}(2)$ when z is considered the fast variable and $F^1_{y,\mathrm{fast}}(1)$ and $F^1_{y,\mathrm{fast}}(2)$ when y is considered the fast variable. It is shown that in the first part of the outer region, where both y and z are fast, there exists a full agreement between the amplitudes of similar accuracy; i.e., $F^1_{z,\mathrm{fast}}(1)$ and $F^1_{y,\mathrm{fast}}(1)$ agree very well with $f^1(1,1)$, while $F^1_{z,\mathrm{fast}}(2)$ and $F^1_{y,\mathrm{fast}}(2)$ agree very well with $f^1(2,1)$.

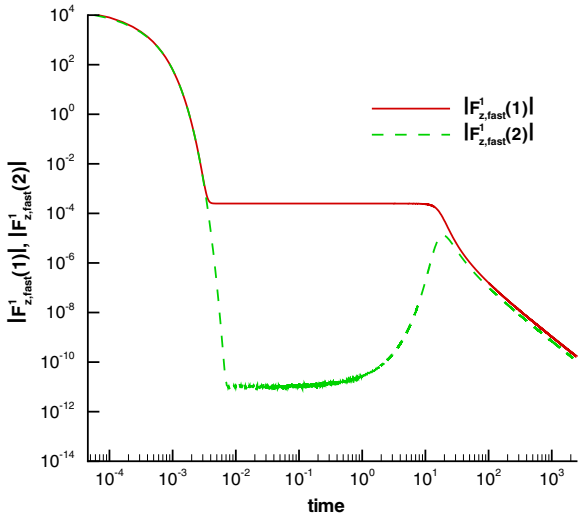

Fig. 8. The fast amplitude $F^1_{z,\mathrm{fast}}(1)$ and $F^1_{z,\mathrm{fast}}(2)$, computed on the basis of G_y (1) and $G_y(2)$, Eqs. (129), in the case where z is taken as the fast variable; $\varepsilon = 10^{-3}$.

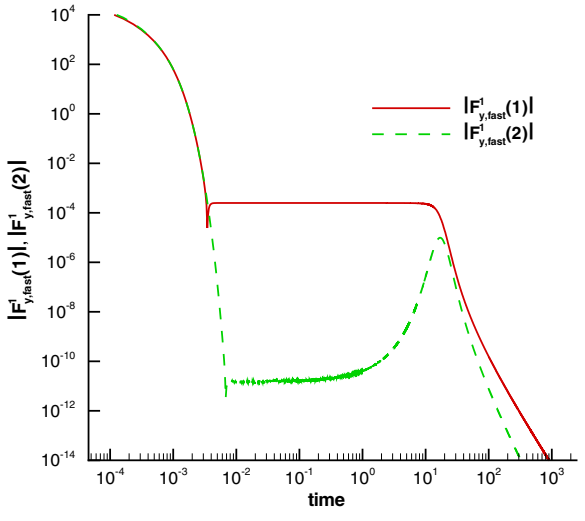

Fig. 9. The fast amplitude $F^1_{y,\mathrm{fast}}(1)$ and $F^1_{y,\mathrm{fast}}(2)$, computed on the basis of $G_z(1)$ and $G_z(2)$, Eqs. (130), in the case where y is taken as the fast variable; $\varepsilon = 10^{-3}$.

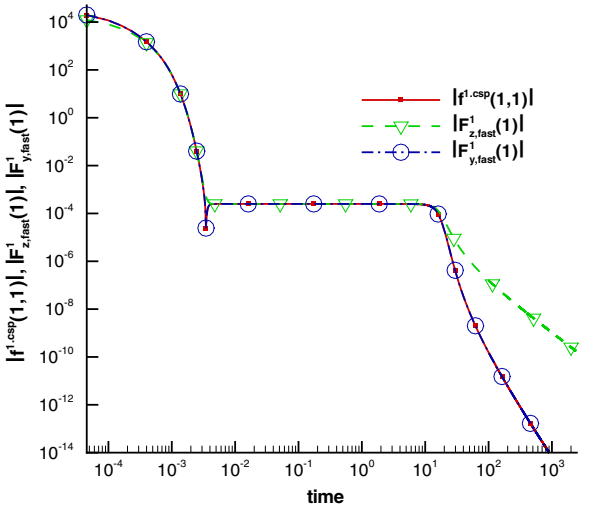

Fig. 10. The fast amplitude computed with one $\mathbf{b}^r$- and one $\mathbf{a}_r$-CSP refinements $f^1(1,1)$ and with first order accuracy in $\mathbf{G}^r_s$, the latter by considering as the fast variable both z and y, $F^1_{z,\mathrm{fast}}(1)$ and $F^1_{y,\mathrm{fast}}(1)$; $\varepsilon = 10^{-3}$.

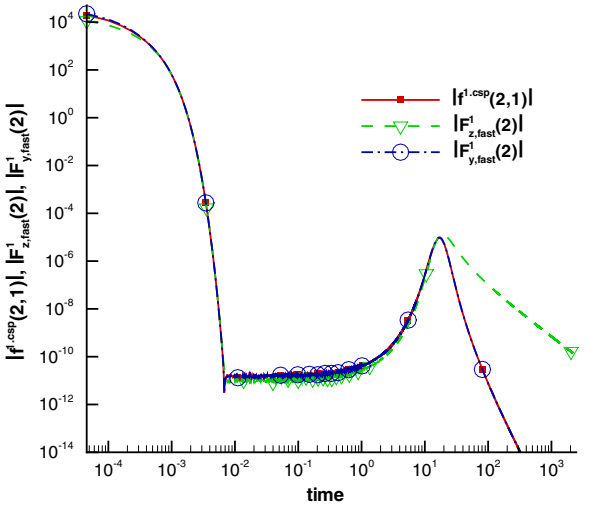

Fig. 11. The fast amplitude computed with two $\mathbf{b}^r$- and one $\mathbf{a}_r$-CSP refinements $f^1(2,1)$ and with second order accuracy in $\mathbf{G}^r_s$, the latter by considering as the fast variable both z and y, $F^1_{z,\mathrm{fast}}(2)$ and $F^1_{y,\mathrm{fast}}(2)$; $\varepsilon = 10^{-3}$.

However, in the second part of the outer region, where only y is fast, only $F^1_{y,\mathrm{fast}}(1)$ and $F^1_{y,\mathrm{fast}}(2)$ agree with $f^1(1,1)$ and $f^1(2,1)$, respectively; the quantities $F^1_{z,\mathrm{fast}}(1)$ and $F^1_{z,\mathrm{fast}}(2)$ exhibiting significant deviations.

11.4. Fast variables

The identification of the fast variables can be made on the basis of the pointer $\underline{\underline{Q}}^r$, Eq. (68), formulated with quantities from the modified CSP algorithm. In particular, when z is considered the fast variable, Eq. (65) yields for $R^s_r(1)$ ($m = 1$ and $R^s_r(0) = 0$) in the first and second parts in the outer region:

$$R^s_r(1) = \frac{1}{1+G_y}, \quad R^s_r(1) = \frac{1}{\omega + G_Y}.$$

The pointer $\underline{Q}^1$ yields:

$$\underline{Q}^1_{1st} = \begin{bmatrix} (G_y+1)^{-1} \\ G_y(G_y+1)^{-1} \end{bmatrix}, \quad \underline{Q}^1_{2nd} = \begin{bmatrix} \omega(G_Y+\omega)^{-1} \\ G_Y(G_Y+\omega)^{-1} \end{bmatrix},$$

where the subscripts "1st" and "2nd" refer to the first ($y \approx z - \varepsilon/2$) and second ($Y \approx Z^2 + \omega Z^3$) parts in the outer region. Substituting for G_y from Eqs. (109) and (111) yields:

$$\underline{Q}^1_{1st} = \begin{bmatrix} 0.5 + O(\varepsilon) \\ 0.5 + O(\varepsilon) \end{bmatrix}, \quad \underline{Q}^1_{2nd} = \begin{bmatrix} O(\omega) \\ 1 + O(\omega) \end{bmatrix}. \tag{131}$$

When z is considered the fast variable, then $\underline{y} = [z, y]^T$. Eq. (131) then confirms that is correct to consider z a fast variable in the first part of the outer region (since $d^1 = O(1)$) but wrong in the second (since $d^1 = O(\omega)$).

On the other hand, when y is considered the fast variable, Eq. (65) yields for $R^s_r(1)$ ($m = 1$ and $R^s_r(0) = 0$) in the first and second parts in the outer region:

$$R^s_r(1) = \frac{z}{(1+G_z)z + \varepsilon}, \quad R^s_r(1) = \frac{\omega^2 Z}{(1+\omega G_Z)\omega Z + 1}.$$

The pointer $\underline{Q}^1$ yields:

$$\underline{Q}^1_{1st} = \begin{bmatrix} 1 - G_z z[(1+G_z)z+\varepsilon]^{-1} \\ G_z z[(1+G_z)z+\varepsilon]^{-1} \end{bmatrix}, \quad \underline{Q}^1_{2nd} = \begin{bmatrix} 1 - \omega^2 G_Z Z[1+(1+\omega G_Z)\omega Z]^{-1} \\ \omega^2 G_Z Z[1+(1+\omega G_Z)\omega Z]^{-1} \end{bmatrix}.$$

Substituting for G_y from Eqs. (110) and (112) yields:

$$\underline{Q}^1_{1st} = \begin{bmatrix} 0.5 + O(\varepsilon^2) \\ 0.5 + O(\varepsilon^2) \end{bmatrix}, \quad \underline{Q}^1_{2nd} = \begin{bmatrix} 1 + O(\omega^2) \\ O(\omega^2) \end{bmatrix}. \tag{132}$$

Since the fast variable is now assumed to be y, then $\underline{y} = [y, z]^T$. Accordingly, Eqs. (132) indicate that this assumption is correct in both parts of the outer region (since $d^1 = O(1)$).

11.5. The simplified non-stiff problem

The correct or wrong identification of the fast variable has an effect on the accuracy of the simplified non-stiff problem, Eq. (22). In particular, when z is considered the fast variable, the simplified non-stiff system in the first and second parts of the outer region, where $y \approx z - \varepsilon/2$ or $Y \approx Z^2 + \omega Z^3$ hold, respectively, are:

$$\frac{d}{dt}\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} G_y \\ 1 \end{bmatrix}\left(\frac{-y}{1+G_y}\right), \tag{133}$$

$$\frac{d}{d\tau}\begin{bmatrix} Z \\ Y \end{bmatrix} = \begin{bmatrix} G_Y \\ 1 \end{bmatrix}\left(\frac{-Y}{G_Y+\omega}\right). \tag{134}$$

For leading order accuracy, employing Eq. (116) for G_y and Eq. (118) for G_Y in Eqs. (133) and (134), respectively, yields:

$$\frac{d}{dt}\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}\left(-\frac{z}{2}\right) + O(\varepsilon), \tag{135}$$

$$\frac{d}{d\tau}\begin{bmatrix} Z \\ Y \end{bmatrix} = \frac{1}{3}\begin{bmatrix} -Z^2 \\ -2Z^2/\omega - 5Z^3/3 \end{bmatrix} + O(\omega) \tag{136}$$

while for second order accuracy, substituting Eq. (117) for G_y and Eq. (119) for G_Y in Eqs. (133) and (134), respectively, yields:

$$\frac{d}{dt}\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}\left(-\frac{z}{2} + \frac{\varepsilon}{4}\right) + O(\varepsilon^2), \tag{137}$$

$$\frac{d}{d\tau}\begin{bmatrix} Z \\ Y \end{bmatrix} = \begin{bmatrix} -Z^2 + 5\omega Z^3 \\ -6Z^3 + 53\omega Z^4 \end{bmatrix} + O(\omega^2). \tag{138}$$

On the other hand, when y is considered the fast variable, the simplified non-stiff system in the two parts of the outer region are:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} y \\ z \end{bmatrix} = \begin{bmatrix} G_z \\ 1 \end{bmatrix}\left(\frac{-z^2}{(1+G_z)z+\varepsilon}\right), \tag{139}$$

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\begin{bmatrix} Y \\ Z \end{bmatrix} = \begin{bmatrix} G_Z \\ 1 \end{bmatrix}\left(-\frac{Z^2}{1+\omega(1+\omega G_Z)Z}\right). \tag{140}$$

For leading order accuracy, employing Eqs. (125) for G_z and (127) for G_Z in Eqs. (139) and (140), respectively, yields:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}\left(-\frac{z}{2}\right) + \mathrm{O}(\varepsilon), \tag{141}$$

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\begin{bmatrix} Y \\ Z \end{bmatrix} = \begin{bmatrix} -2Z^3 \\ -Z^2 \end{bmatrix} + \mathrm{O}(\omega), \tag{142}$$

while for second order accuracy, substituting Eqs. (126) for G_z and (128) for G_Z in Eqs. (139) and (140), respectively, yields:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}\left(-\frac{z}{2}+\frac{\varepsilon}{4}\right) + \mathrm{O}(\varepsilon^2), \tag{143}$$

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\begin{bmatrix} Y \\ Z \end{bmatrix} = \begin{bmatrix} -2Z^3-\omega Z^4 \\ -Z^2+\omega Z^3 \end{bmatrix} + \mathrm{O}(\omega^2). \tag{144}$$

The different simplified non-stiff systems above should be compared with those obtained with the singular perturbation analysis and the CSP method in the first and second parts of the outer region for leading order accuracy, Eqs. (78) and (87), and second order accuracy, Eqs. (80) and (89). It is seen that in the first part of the outer region, where both y and z are fast, the new algorithm produces the expected accuracy when selecting either y or z as the fast one; compare Eqs. (135), (137) and (141), (143) with (78), (80). However, in the second part, where only y is fast, the new algorithm failed to produce the correct results when z was selected as the fast variable, compare Eqs. (136), (138) with (87), (89), but succeeded when y was identified as the fast variable, compare Eqs. (142), (144) with (87), (89).

Finally, the problems that arise when the form of Eq. (23) of the simplified system is employed, instead of the correct form of Eq. (22) can now be discussed. It was shown by Eqs. (133), (134) and (139), (140) that use of Eq. (22) resulted in non-stiff simplified problems, the accuracy level of which depended on whether the correct choice was made for the fast variable. However, if Eq. (23) is employed in the first part of the outer region, the following simplified systems are produced:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}\left(\frac{z}{\varepsilon}(z-y)-y\right) + \mathrm{O}(\varepsilon), \tag{145}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} y \\ z \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}\left(-\frac{z}{\varepsilon}(z-y)\right) + \mathrm{O}(\varepsilon) \tag{146}$$

in the case where z or y, respectively, is selected as the fast variable. It is seen that, although on the SIM expressed by the relation $y = z - \varepsilon/2 + \varepsilon^2/(4z) + \mathrm{O}(\varepsilon^3)$, Eqs. (145) and (146) reduce to the correct asymptotic limit displayed by Eq. (137), both are stiff. Similar conclusions for the stiffness of Eq. (23) can be drawn if the second part of the outer region is considered.

12. Discussion

The new CSP algorithm presented here resulted in the set of basis vectors:

$$\mathbf{A}_r(n,j) = \begin{bmatrix} \mathbf{I}_r^r - \mathbf{G}_s^r(n)\mathbf{R}_r^s(j) \\ -\mathbf{R}_r^s(j) \end{bmatrix}, \quad \mathbf{A}_s(n) = \begin{bmatrix} \mathbf{G}_s^r(n) \\ \mathbf{I}_s^s \end{bmatrix},$$
$$\mathbf{B}^r(n) = [\mathbf{I}_r^r, -\mathbf{G}_s^r(n)], \quad \mathbf{B}^s(n,j) = \left[\mathbf{R}_r^s(j), \mathbf{I}_s^s - \mathbf{R}_r^s(j)\mathbf{G}_s^r(n)\right].$$

It was shown that these vectors are the CSP basis vectors, properly scaled. The accuracy they provide for the identification of the SIM and the solution of the simplified non-stiff system is similar to that provided by the original CSP algorithm. The equivalence of the two sets of basis vectors is stated in Appendix A.

The modified CSP algorithm is shown to be much more efficient, especially in the cases where the SIM extends to a very small number of dimensions ($M \to N$), so that significant simplifications are practically possible. An estimate of the increased efficiency can be obtained by estimating the number of operations for (i) the evaluation of the M vectors in $\mathbf{b}^r$ and $\mathbf{B}^r$ after one refinement, which are needed for the identification of the SIM through the equations $\mathbf{b}^r\,\underline{g} \approx \mathbf{0}$ or $\mathbf{B}^r\underline{g} \approx \mathbf{0}$, and (ii) the evaluation of the projection matrices $\mathbf{a}_s\mathbf{b}^s$ and $\mathbf{A}_s\mathbf{B}^s$, which are needed for the computation of the vector fields $(\mathbf{a}_s\mathbf{b}^s)\underline{g}$ or $(\mathbf{A}_s\mathbf{B}^s)\underline{g}$ of the simplified non-stiff problems, according to the original and the new CSP algorithms, respectively. In the following, it is assumed that the solution of M N-dimensional linear systems $\mathbf{A}\underline{x} = \underline{b}$ and the inversion of a $N \times N$ matrix require $(N/3+M)N^2$ and $4N^3/3$ number of operations, respectively.

In particular, the $\mathbf{b}^r$-refinement is completed by solving NM-dimensional linear systems, as specified by Eq. (38), while the $\mathbf{B}^r$-refinement according to Eq. (58) is completed by solving $(N-M)$ M-dimensional linear systems. Neglecting the computational load for the evaluation of the time derivative terms, the $\mathbf{b}^r$- and $\mathbf{B}^r$-refinements require:

$$C_1 = MN^2(1 + 2\delta + \delta^2/3)$$

and

$$C_2 = MN^2(1 - 2\delta^2/3)$$

number of operations, respectively, where $\delta = M/N$. When $\delta \ll 1$ both algorithms need the same number of operations. In contrast, when $\delta \to 1$ the new CSP algorithm requires O(10) less operations; i.e., $C_1 \approx 10C_2$.

The computation of the projection matrices $\mathbf{a}_s\mathbf{b}^s$ and $\mathbf{A}_s\mathbf{B}^s$, given $\mathbf{a}_r$, $\mathbf{b}^r$, $\mathbf{G}_s^r$, and $\mathbf{R}_s^r$, requires:

$$D_1 = MN^2 + N^2$$

and

$$D_2 = MN^2(1-\delta)^2 + N^2(1-\delta^2)$$

number of operations, respectively. Both algorithms need the same number of operations when $\delta \ll 1$, while the new algorithm becomes much faster in the limit $\delta \to 1$; i.e., $D_2 \ll D_1$.

The superior efficiency of the modified CSP algorithm, thus established in the case where the time derivative terms are neglected, becomes more pronounced when the time derivative terms are taken into account. First, the $\mathbf{b}^r$-refinement requires the evaluation of MN derivatives, while the $\mathbf{B}^r$-refinement requires the evaluation of only $M(N-M)$ ones. Appendix B displays the necessary operations for the time derivative evaluation of $\mathbf{b}^r(1,m_1)$ needed for two $\mathbf{b}^r$-refinements, when starting from constant $\mathbf{b}^r(0,m_1)$ with the algorithm developed in [45]. Also shown are the operations for the time derivative evaluation of $\mathbf{B}^r(1)$ needed for two $\mathbf{B}^r$-refinements, with the same algorithm. It is demonstrated that $\partial\mathbf{b}^r(1,m_1)/\partial t$ and $\partial\mathbf{B}^r(1)/\partial t$ require:

$$E_1 = MN^2(4 + 3\delta + 2\delta^2/3)$$

and

$$E_2 = MN^2((1-\delta)(2+3\delta) + 2\delta^2/3)$$

number of operations, respectively. Again, both algorithms need a similar number of operations when $\delta \ll 1$, while when $\delta \to 1$ the new algorithm requires less operations; i.e., $E_1 \approx 10E_2$.

By employing Eqs. (41) and (42), the methodologies employed by CSP for declaring the fast amplitudes negligible and for correcting the omission of the fast amplitudes during the integration along the SIM can also be employed in the modified CSP algorithm presented here.

In particular, according to Eq. (47), the amplitudes of the modified CSP version presented here are related to the ones of the original version by the relations:

$$\underline{F}^r = (\mathbf{b}_r^r)^{-1}\underline{f}^r, \quad \underline{F}^s = (\mathbf{a}_s^s)\underline{f}^s. \tag{147}$$

Given the discussion in Section 10 on the magnitude of $\mathbf{b}_r^r$ and $\mathbf{a}_s^s$, it follows that the amplitudes $\underline{F}^r$ will remain negligible when the M fastest time scales are exhausted, exactly as the amplitudes $\underline{f}^r$ do. According to the CSP method the M fastest modes were declared exhausted when the corresponding amplitudes satisfy the condition:

$$|\tau(M+1)\mathbf{a}_r\underline{f}^r| < e_{\mathrm{rel}}\underline{y} + e_{\mathrm{abs}}, \tag{148}$$

where $\tau(M+1)$ denotes the fastest of the $N-M$ slow time scales, $|\underline{v}|$ denotes the absolute values of the elements of the vector $\underline{v}$ and $0(e_{\mathrm{rel}}) \ll 1$ and $0(e_{\mathrm{abs}}) \ll 1$ denote the allowed relative and absolute error committed when the fast amplitudes are neglected during the integration of the simplified non-stiff system, Eqs. (46) [18–20,22,26]. Due to the transformations (41) and (42), the condition equation (148) is cast as:

$$|\tau(M+1)\mathbf{A}_r\underline{F}^r| < e_{\mathrm{rel}}\underline{y} + e_{\mathrm{abs}} \tag{149}$$

and can be employed for identifying the negligible fast amplitudes, Eq. (43), in the revised CSP algorithm.

In addition, according to the CSP algorithm a correction to the effects of omitting the fast amplitudes when solving the simplified non-stiff system along the SIM was applied at the end of each time step, according to the relation:

$$\underline{y} \leftarrow \underline{y} - \mathbf{a}_r\tau_r^r\underline{f}^r \tag{150}$$

introduced as the "homogeneous correction" [17,19]. Employing the relations (41) and (42), this correction can also be employed in the modified CSP algorithm as:

$$\underline{y} \leftarrow \underline{y} - \mathbf{A}_r\left(\mathbf{J}_r^r - \mathbf{G}_s^r(n)\mathbf{J}_r^s\right)^{-1}\underline{F}^r. \tag{151}$$

Finally, it should be noted that the ease of computing $\mathbf{G}_s^r$ allows the incremental changes in the M fast components of the state vector $\underline{y}$ to be directly related to those of the remaining $N-M$ ones. This is achieved by the definition of the matrix $\mathbf{G}_s^r$, Eq. (11), which can be cast as:

$$\partial\underline{z} = \mathbf{G}_s^r\partial\underline{s}, \tag{152}$$

where $\underline{z}$ contains the M fastest components of $\underline{y}$ and $\underline{s}$ contains the remaining $N-M$ components parameterizing the manifold. For the Lindemann problem examined previously and for the second part of the outer region where $\epsilon y \approx z^2$, substituting the leading order approximation for $\mathbf{G}_s^r$ in Eq. (152) yields:

$$\partial y = \frac{2z}{\varepsilon}\partial z \tag{153}$$

when y is considered the fast variable and

$$\partial z = \frac{\varepsilon}{2z}\partial y \tag{154}$$

when z is considered the fast variable. Noting that in the period under consideration $2z/\varepsilon \ll 1$, Eqs. (153) and (154) provide a clear demonstration of the possible benefits one can gain from Eq. (152), as well as the pitfalls that might arise from a wrong selection of the fast variables.

13. Conclusions

The invariance equation method for the construction of SIM was extended to produce the simplified system that governs the evolution of the solution on the SIM. It was shown that this extended IE method is nothing but the computational singular perturbation method, its basis vectors properly scaled. It was further shown that this new (rescaled) version of CSP produces the expected accuracy only when the number and identity of the fast variables, equal to the dimensions of the fast subdomain, are correctly defined.

It was also demonstrated that the modified CSP algorithm exhibits superior convergence properties in the description of the manifold (in comparison to the original IE method) and a much higher efficiency in

constructing the simplified non-stiff problem (in comparison to the original CSP method). The modified CSP method offers the basis for developing a very promising scheme of numerical integration, along the lines of the time-scale splitting, explicit algorithm presented in [26], especially when significant reductions are allowed by the physics of the problem under consideration; i.e., $1 \ll N$ and $M \approx N$.

These very appealing characteristics follow mainly from (i) the development of Eq. (58) for the iterative computation of the matrix $\mathbf{G}_s^r$ and (ii) the modification of the CSP method in a way that allows both the manifold and the simplified non-stiff system to be constructed on the basis of $\mathbf{G}_s^r$ and $\mathbf{R}_r^s$.

The matrix $\mathbf{G}_s^r$ is smaller than its counterpart in the original CSP method $\mathbf{b}^r$; the former being $M \times (N - M)$-dimensional and the latter $M \times N$-dimensional. $\mathbf{G}_s^r$ must be updated every time step, since it determines the accuracy of both the manifold and the solution of the simplified non-stiff problem. Similarly, the matrix $\mathbf{R}_r^s$ is smaller than its counterpart $\mathbf{a}_r$; the former being $(N - M) \times M$-dimensional and the later $N \times M$-dimensional. However, since it determines the non-stiffness of the simplified problem, the matrix $\mathbf{R}_r^s$ needs updating only when significant rotation of the fast/slow subspaces is taking place.

The CSP version presented here is very promising in addressing problems modeled by differential equations where large scale reductions are possible, such as those arising in the simulation of physical problems requiring an efficient prediction of coarse-grained or macroscopic quantities by considering mathematical models based on the detailed or microscopic level [58–62].

Acknowledgments

M.V. acknowledges the Italian Ministry of University and Research (MIUR) for the support to this research. D.A.G. acknowledges MIUR who supported his stay in Rome the fall of 2004. This work was partially supported by the US Department of Energy, SciDAC Computational Chemistry Program.

Appendix A. Similarity of the original and new CSP basis vectors

The $\mathbf{b}^r$-refinement of the CSP basis vectors, Eq. (38), can be expressed in matrix form as:

$$\begin{bmatrix} \mathbf{b}^r(k_1+1,m_1) \\ \mathbf{b}^s(k_1+1,m_1) \end{bmatrix} = \begin{bmatrix} \mathbf{I}_r^r & \mathbf{p}_s^r \\ \mathbf{0}_r^s & \mathbf{I}_s^s \end{bmatrix} \begin{bmatrix} \mathbf{b}^r(k_1,m_1) \\ \mathbf{b}^s(k_1,m_1) \end{bmatrix}, \tag{A.1}$$

$$[\mathbf{a}_r(k_1+1,m_1)\mathbf{a}_s(k_1+1,m_1)] = [\mathbf{a}_r(k_1,m_1)\mathbf{a}_s(k_1,m_1)] \begin{bmatrix} \mathbf{I}_r^r & -\mathbf{p}_s^r \\ \mathbf{0}_r^s & \mathbf{I}_s^s \end{bmatrix}, \tag{A.2}$$

where $\mathbf{p}_s^r = \boldsymbol{\tau}_r^r(k_1,m_1)\boldsymbol{\lambda}_s^r(k_1,m_1)$.

Similarly, the $\mathbf{a}_r$-refinement of the CSP basis vectors, Eq. (39), can be stated as:

$$\begin{bmatrix} \mathbf{b}^r(k_2,m_2+1) \\ \mathbf{b}^s(k_2,m_2+1) \end{bmatrix} = \begin{bmatrix} \mathbf{I}_r^r & \mathbf{0}_s^r \\ -\mathbf{q}_r^s & \mathbf{I}_s^s \end{bmatrix} \begin{bmatrix} \mathbf{b}^r(k_2,m_2) \\ \mathbf{b}^s(k_2,m_2) \end{bmatrix}, \tag{A.3}$$

$$[\mathbf{a}_r(k_2,m_2+1)\mathbf{a}_s(k_2,m_2+1)] = [\mathbf{a}_r(k_2,m_2)\mathbf{a}_s(k_2,m_2)] \begin{bmatrix} \mathbf{I}_r^r & \mathbf{0}_s^r \\ \mathbf{q}_r^s & \mathbf{I}_s^s \end{bmatrix}, \tag{A.4}$$

where $\mathbf{q}_r^s = \boldsymbol{\lambda}_r^s(k_2,m_2)\boldsymbol{\tau}_r^r(k_2,m_2)$.

After k $\mathbf{b}^r$-refinements and m $\mathbf{a}_r$-refinements, the accuracy provided by the original CSP vectors is equivalent to that provided by the new basis vectors, as determined by the expressions:

$$\begin{bmatrix} \mathbf{B}^r(n) \\ \mathbf{B}^s(n,j) \end{bmatrix} = \begin{bmatrix} \left(\mathbf{b}_r^r(k,m)\right)^{-1} & \mathbf{0}_s^r \\ \mathbf{0}_r^s & \mathbf{a}_s^s(k,m) \end{bmatrix} \begin{bmatrix} \mathbf{b}^r(k,m) \\ \mathbf{b}^s(k,m) \end{bmatrix}, \tag{A.5}$$

$$[\mathbf{A}_r(n,j)\mathbf{A}_s(n)] = [\mathbf{a}_r(k,m)\mathbf{a}_s(k,m)] \begin{bmatrix} \mathbf{b}_r^r(k,m) & \mathbf{0}_s^r \\ \mathbf{0}_r^s & \left(\mathbf{a}_s^s(k,m)\right)^{-1} \end{bmatrix}, \tag{A.6}$$

where the values of the indices n and j depend on the initial guess employed for the new refinement procedure.

Appendix B. The time derivative of the vectors in $\mathbf{b}^r$ and $\mathbf{B}^r$

In [45] it was shown that starting with constant $\mathbf{b}^r(0,m_1)$ and $\mathbf{a}_r(0,m_1)$, the evaluation of $\mathbf{b}^r(1,m_1)$ and its derivative can be accomplished through the sequence of operations:

$$\boldsymbol{\lambda}_r^r(0,m_1) = \mathbf{b}^r(0,m_1)\mathbf{J}\mathbf{a}_r(0,m_1), \tag{B.1}$$

$$\boldsymbol{\lambda}_r^r(0,m_1)\mathbf{b}^r(1,m_1) = \mathbf{b}^r(0,m_1)\mathbf{J}, \tag{B.2}$$

$$\boldsymbol{\lambda}_r^r(0,m_1)\frac{\partial \mathbf{b}^r(1,m_1)}{\partial t} = \mathbf{b}^r(0,m_1)\frac{\partial \mathbf{J}}{\partial t}\left[\mathbf{I}_N^N - \mathbf{a}_r(0,m_1)\mathbf{b}^r(1,m_1)\right]. \tag{B.3}$$

Similarly, starting with constant $\mathbf{B}^r(0) = [\mathbf{I}_r^r, -\mathbf{G}_s^r(0)]$, the evaluation of $\mathbf{B}^r(1)$ and its derivative can be accomplished through the sequence of operations:

$$\boldsymbol{\Lambda}_r^r(0) = \mathbf{J}_r^r - \mathbf{G}_s^r(0)\mathbf{J}_r^s, \tag{B.4}$$

$$\boldsymbol{\Lambda}_r^r(0)\mathbf{G}_s^r(1) = \left[\mathbf{G}_s^r(0)\mathbf{J}_s^s - \mathbf{J}_s^r\right], \tag{B.5}$$

$$\mathbf{B}^r(1) = [\mathbf{I}_r^r, -\mathbf{G}_s^r(1)], \tag{B.6}$$

$$\boldsymbol{\Lambda}_r^r(0)\frac{\partial \mathbf{G}_s^r(1)}{\partial t} = -\mathbf{B}^r(0)\frac{\partial \mathbf{J}}{\partial t}\mathbf{A}_s(1), \tag{B.7}$$

$$\frac{\partial \mathbf{B}^r(1)}{\partial t} = \left[\mathbf{0}_r^r, -\frac{\partial \mathbf{G}_s^r(1)}{\partial t}\right]. \tag{B.8}$$

References

[1] N. Fenichel, Geometric singular perturbation theory for ordinary differential equations, J. Diff. Eqs. 31 (1979) 53–98.
[2] P. Constantin, C. Foias, B. Nicolaenko, R. Temam, Integral manifolds and inertial manifolds for dissipative partial differential equations, Appl. Math. Sci. 70 (1989).
[3] S.J. Fraser, The steady state and equilibrium approximations: a geometrical picture, J. Chem. Phys. 88 (1988) 4732–4738.
[4] A.H. Nguyen, S.J. Fraser, Geometrical picture of reaction in enzyme kinetics, J. Chem. Phys. 91 (1989) 186–193.
[5] M.R. Roussel, S.J. Fraser, Geometry of the steady-state approximation: perturbation and accelerated convergence methods, J. Chem. Phys. 93 (1990) 1072–1081.
[6] M.R. Roussel, S.J. Fraser, On the geometry of transient relaxation, J. Chem. Phys. 94 (1991) 7106–7113.
[7] M.R. Roussel, S.J. Fraser, Accurate steady-state approximations-implications for kinetics experiments and mechanism, J. Phys. Chem. 95 (1991) 8762–8770.
[8] M.R. Roussel, S.J. Fraser, Global analysis of enzyme-inhibition kinetics, J. Phys. Chem. 97 (1993) 8316–8327.
[9] M.R. Roussel, Forced convergence iterative schemes for the approximation of invariant manifolds, J. Math. Chem. 21 (1997) 385–393.
[10] M.R. Roussel, S.J. Fraser, Invariant manifold for metabolic reduction, Chaos 11 (2001) 196–206.
[11] S.J. Fraser, Slow manifold for a bimolecular association mechanism, J. Chem. Phys. 120 (2004) 3075–3085.
[12] M.J. Davis, R.T. Skodje, Geometric investigation of low-dimensional manifolds in systems approaching equilibrium, J. Chem. Phys. 111 (1999) 859–874.
[13] M.J. Davis, R.T. Skodje, Geometric approach to multiple-time-scale kinetics: a nonlinear master equation describing vibration-to-vibration relaxation, Z. Phys. Chem. 215 (2001) 233–252.
[14] R.T. Skodje, M.J. Davis, Geometrical simplification of complex kinetic systems, J. Phys. Chem. A 105 (2001) 10356–10365.
[15] J. Nafe, U. Maas, A general algorithm for improving ILDMs, Combust. Theory Model 6 (2002) 697–709.
[16] S.H. Lam, D.A. Goussis, Understanding complex chemical kinetics with computational singular perturbation, Proc. Combust. Inst. 22 (1988) 931–941.
[17] S.H. Lam, D.A. Goussis, Conventional asymptotic and computational singular perturbation for simplified kinetics modelling, in: M.O. Smooke (Ed.), Reduced Kinetic Mechanisms and Asymptotic Approximations for Methane–Air Flames, Springer Lecture Notes, Springer, Berlin, 1991, pp. 227–242.
[18] D.A. Goussis, S.H. Lam, A study of homogeneous methanol oxidation kinetics using CSP, Proc. Combust. Inst. 24 (1992) 113–120.
[19] S.H. Lam, Using CSP to understand complex chemical kinetics, Combust. Sci. Technol. (1993) 375–404.
[20] S.H. Lam, D.A. Goussis, The CSP method for simplifying kinetics, Int. J. Chem. Kinet. 26 (1994) 461–486.
[21] D.A. Goussis, On the construction and use of reduced chemical kinetics mechanisms produced on the basis of given algebraic relations, J. Comput. Phys. 128 (1996) 261–273.
[22] M. Hadjinicolaou, D.A. Goussis, Asymptotic solution of Stiff PDE's – the reaction diffusion equation, SIAM J. Sci. Comput. 20 (1999) 781–810.

[23] A. Massias, D. Diamantis, E. Mastorakos, D.A. Goussis, An algorithm for the construction of global reduced mechanisms with CSP data, Combust. Flame 117 (1999) 685–708.
[24] G.E. Frouzakis, K. Boulouchos, Analysis and reduction of the CH_4–air mechanism at lean conditions, Combust. Sci. Tech. 159 (2000) 281–303.
[25] T.F. Lu, Y.G. Ju, C.K. Law, Complex CSP for chemistry reduction and analysis, Combust. Flame 126 (2001) 1445–1455.
[26] M. Valorani, D.A. Goussis, Explicit time-scale splitting algorithms for Stiff problems: auto-ignition of gaseous mixtures behind a steady shock, J. Comput. Phys. 169 (2001) 44–79.
[27] M. Valorani, F. Creta, D.A. Goussis, H.N. Najm, Local and global manifolds in stiff reaction-diffusion systems, in: 2nd MIT Conf. on Comput. Fluid and Solid Mechanics, Boston, June, 2003, pp. 1548–1551.
[28] M. Valorani, H.N. Najm, D.A. Goussis, CSP analysis of a transient flame vortex interaction: time scales and manifolds, Combust. Flame 134 (2003) 35–53.
[29] U. Maas, S.B. Pope, Simplifying chemical kinetics – intrinsic low dimensional manifolds in composition space, Combust. Flame 88 (1992) 239–264.
[30] U. Maas, S.B. Pope, Implementation of simplified chemical kinetics based on intrinsic low dimensional manifolds, Proc. Combust. Inst. 24 (1992) 103–112.
[31] D. Schmidt, U. Maas, Warnatz, Simulation of nonequilibrium hypersonic flows, Comput. Fluids 22 (1993) 285–294.
[32] U. Maas, S.B. Pope, Laminar flame calculations using simplified chemical kinetics based on intrinsic low dimensional manifolds, Proc. Combust. Inst. 25 (1994) 1349–1356.
[33] R.L.G.M. Eggels, L.P.H. DeGoey, Mathematically reduced reaction mechanisms applied to adiabatic flat hydrogen–air flames, Combust. Flame 100 (1995) 559–570.
[34] R.L.G.M. Eggels, L.P.H. DeGoey, Modeling of burner-stabilized hydrogen/air flames using mathematically reduced reaction schemes, Combust. Sci. Tech. 107 (1995) 165–180.
[35] D. Schmidt, T. Blasenderey, U. Maas, Intristic low-dimensional manifold for strained and unstrained flames, Combust. Theory Model 2 (1998) 135–152.
[36] K. Xiao, D. Schmidt, U. Maas, Pdf simulation of turbulent non-premixed CH_4/H_2–air flames using automatically simplified kinetics, Proc. Combust. Inst. 27 (1998) 1073–1080.
[37] T. Blasenderey, U. Maas, ILDMs of higher hydrocarbons and the hierarchy of chemical kinetics, Proc. Combust. Inst. 28 (2000) 1623–1630.
[38] S. Singh, Y. Rastigejev, S. Paolucci, J.M. Powers, Viscous detonation in H_2–O_2–Ar using intristic low-dimensionals and wavelet adaptive multilevel representation, Combust. Theory Model. 5 (2001) 163–184.
[39] S. Singh, J.M. Powers, S. Paolucci, On slow manifolds of chemically reactive systems, J. Chem. Phys. 117 (2002) 1482–1496.
[40] I. Goldfarb, V. Goldshtein, U. Maas, Comparative analysis of two asymptotic approaches based on integral manifolds, IMA J. Appl. Math. 69 (2004) 353–374.
[41] H.G. Kaper, T.J. Kaper, Asymptotic analysis of two reduction methods for systems of chemical reactions, Physica D 165 (2002) 66–93.
[42] A. Zagaris, H.G. Kaper, T.J. Kaper, Analysis of the computational singular perturbation reduction method for chemical kinetics, J. Nonlinear Sci. 14 (2004) 59–91.
[43] A. Zagaris, H.G. Kaper, T.J. Kaper, Fast and slow dynamics for the CSP method, SIAM J. Multiscale Model. 2 (2004) 613–638.
[44] D.A. Goussis, M. Valorani, F. Creta, H.N. Najm, Inertial manifolds with CSP, in: 2nd MIT Conference on Computational Fluid and Solid Mechanics, Boston, June, 2003, pp. 1951–1954.
[45] M. Valorani, D.A. Goussis, F. Creta, H.N. Najm, Higher order corrections in the approximation of inertial manifolds and the construction of simplified problems with the CSP method, J. Comput. Phys. 209 (2005) 754–786.
[46] A.N. Gorban, I.V. Karlin, Method of invariant manifold for chemical kinetics, Chem. Eng. Sci. 58 (2003) 4751–4768.
[47] A.N. Gorban, I.V. Karlin, Invariant manifolds for physical and chemical kineticsLectutre Notes in Physics, vol. 660, Springer, Berlin, 2004.
[48] O. Axelsson, Iterative Solution Methods, Cambridge University Press, England, 1994.
[49] G. Strang, Linear Algebra and its Applications, third ed., Saunders, Philadelphia, PA, 1988.
[50] A. Jennings, J.J. McKeown, Matrix Computations, Wiley, England, 1992.
[51] W. Richardson, J. Volk, K.H. Lau, S.H. Lin, H. Eyring, Application of the singular perturbation method to reaction kinetics, Proc. Natl. Acad. Sci. USA 70 (1973) 1588–1592.
[52] K.J. Laidler, Chemical Kinetics, McGraw-Hill, New York, 1950, pp. 76–85.
[53] S.M.T. De La Selva, E. Pina, Some mathematical properties of the Lindemann mechanism, Rev. Mex. Fis. 42 (1996) 431–448.
[54] M.J. Davis, S.J. Klippenstein, Geometric investigation of association dissociation kinetics with an application to the master equation for $CH_3 + CH_3 \lessgtr C_2H_6$, J. Phys. Chem. A 106 (2002) 5860–5879.
[55] R.Y. Qian, A new formulation of the Lindemann mechanism of unimolecular reactions, Chin. J. Chem. 21 (2003) 1562–1564.
[56] K.R.A. Kumar, A.C. McIntosh, J. Brindley, et al., Effect of two-step chemistry on the critical extinction-pressure drop for pre-mixed flames, Combust. Flame 134 (2003) 157–167.
[57] K. Saito, Effect of different choices of the Boltzmannized flux operator on rate constants in the Lindemann mechanism, Chem. Phys. Lett. 383 (2004) 282–287.
[58] A.N. Gorban, I.V. Karlin, A.Y. Zinovyev, Constructive methods of invariant manifolds for kinetic problems, Phys. Rep. 396 (2004) 197–403.

[59] D. Givon, R. Kupferman, A. Stuart, Extracting macroscopic dynamics: model problems and algorithms, Nonlinearity 17 (2004) 55–127.
[60] C.W. Gear, I.G. Kevrekidis, Telescopic projective methods for parabolic differential equations, J. Comput. Phys. 187 (1993) 95–109.
[61] E. Pesheck, N. Boivin, P. Christophe, Nonlinear modal analysis of structural systems using multi mode invariant manifolds, Nonlinear Dyn. 25 (2001) 183–205.
[62] C.W. Gear, T.J. Kaper, I.G. Kevrekidis, A. Zagaris, Projecting to a slow manifold: singularly perturbed systems and legacy codes, SIAM J. Appl. Dyn. Syst. 4 (2005) 711–732.