Copyright © 2010 Tech Science Press CMES, vol.65, no.2, pp.193-215, 2010

Investigation on an Accelerated Scheme for Solving Time-Dependent Systems

Montri Maleewong[1] **and Sirod Sirisup**[2]

Abstract: In this paper, we describe our investigation of an "on-line" POD-assisted projective integration method for solving a nonlinear PDE. Using the on-line method, we have computed the representative POD modes without assuming knowledge of the underlying slow manifold along the integration process. This approach is based on the "equation-free" framework where the governing PDE does not need to be projected onto the POD bases in order to build a reduced-order model. The main objectives of this study were to investigate the effectiveness of the method in reducing the computational time required for numerically solving a nonlinear PDE. Here, the one-dimensional viscous Burgers' equation is chosen as the time-dependent illustrative prototype. The numerical results from this method are in good agreement with both the exact solutions and the full DNS results, while the computational effort has been reduced by up to 72%. Linear stability analysis is presented in order to predict the stability of the method. Moreover, study of the interplay between the POD modes reveals the role of each dominant POD mode in dynamically representing the exact solution, and pinpoints the POD mode accounting for the instability in the approach.

Keywords: Projective integration, Proper orthogonal decomposition, equation-free methods

1 Introduction

Today, computer simulation via differential equation models has become a very useful part of research areas such as engineering, physics, chemistry, biology, social sciences, and the economics of human systems. The results of computer simulation do not only allow researchers to gain insights into the operation of those systems, but also provide a visualization of fundamental behavior regarding problems of

[1] Department of Mathematics, Faculty of Science Kasetsart University, Bangkok, Thailand
[2] Large-Scale Simulation Research Laboratory National Electronics and Computer Technology Center, Thailand, Corresponing author: sirod.sirisup@nectec.or.th

interest. Although it can provide us with many good predictions of complex phenomena, the computational time required to achieve this is, in general, very large. Thus, most of the calculations are usually performed on high-performance computers, parallel or grid computing. Nevertheless, it remains impractical in many practical problems from both computing and data-handling viewpoints. It is necessary to employ a simpler model that also embraces all key phenomena from the original models. The benefits of inventing these reduced-order models are twofold. First, there would be the ability to perform simulations and accurately predict complex phenomena with much lower computing needs. The second benefit would be the ability to directly comprehend complex phenomena from these reduced-order models without mining data sets obtained from traditional simulations. There are many developments trying to address this issue. Currently, many reduced-order models or low-dimensional models have been proposed by researchers in many research fields; see Ravindran (2000); Rambo and Joshi (2007); Samimy, Debiasi, Caraballo, Serrani, amd J. Little, and Myatt (2007); Missoffe, Juillard, and Aubry (2007); Arifin, Noorani, and Kilicman (2007); Rempfer (2003), for example.

The proper orthogonal decomposition (POD) method is one of the well-known methods used for creating a low-dimensional model. It is a powerful tool based on statistical analysis. It is able to identify low-dimensional descriptions (both on a spatial/temporal dominant basis or structures) for multidimensional systems (see Bekooz, Holmes, and Lumley (1993)) and utilizes these structures to build a robust low-dimensional model. Used with the method of snapshots, first proposed in Sirovich (1987) for flow systems, the POD method becomes particularly effective and easy to implement. Moreover, the POD method has been successfully implemented in conjunction with both experimental and numerical studies for a wide range of applications.

In this paper, we explore a POD-assisted projective integration methodology that employs "equation-free" projective integration frameworks, pioneered by Kevrekidis, Gear, and Hummer (2004). This framework has been applied to a variety of problems, ranging from the bifurcation analysis of complex systems to the homogenization of random media, see Kevrekidis, Gear, Hyman, Kevrekidis, Runborg, and Theodoropoulos (2003); Rico-Martinez, Gear, and Kevrekidis (2004); Makeev, Maroudas, Panagiotopoulos, and Kevrekidis (2002); Makeev, Maroudas, and Kevrekidis (2002); Xiu, Kevrekidis, and Ghanem (2005); Russo, Siettos, and Kevrekidis (2007). The equation-free framework is designed for the efficient coarse-grained computational study of complex, multi-scale problems. The basic idea operates at two levels:

(a) design and perform short-time numerical experiments with "the best available" microscopic model, then

(b) use the numerical results of such microscopic computations to estimate quantities (residuals, action of Jacobians) required in numerical computations of the macroscopic equations for the coarse-grained system behavior, see Gear and Kevrekidis (2004).

A similar POD-assisted projective integration approach has been successfully applied to solve the Navier-Stokes equation, see Sirisup, Karniadakis, Xiu, and Kevrekidis (2005). However, in this current work, we perform POD-assisted projective integration without assuming knowledge of the underlying slow manifold in the integration process. We need to compute the underlying slow manifold for every large projective integration step (see a full definition in the Methodologies section). Thus the POD modes must be computed on the fly during which we march the numerical solution in time. This is called the "on-line" method. This method is completely different from that presented in Sirisup, Karniadakis, Xiu, and Kevrekidis (2005), which is called the "off-line" method since the POD modes are already computed before the numerical solutions are marched in time. Full details of the on-line method are given in Section 2.2. We can observe that this approach is more practical than the off-line method. It can be used to solve any complex systems where the original computer code can still be applied, just merging the new on-line method code with the original code. This is sometimes referred to as the concept of building a legacy code. Moreover, the dynamics and interactions of each underlying manifold or each representative POD mode can also be revealed through this approach. We will present the functions of each POD mode, including the first few POD modes, through the on-line method in Section 4.

In summary, in this current work, we focus on three aspects of the POD-assisted projective integration approach:

- An examination of the effectiveness of the approach in numerically solving a nonlinear PDE while the POD modes are not available a priori, and an analysis of the stability and accuracy of the on-line method.
- An investigation of the dynamic behavior of solutions represented by each POD mode during the time marching of the method. This will provide us with information on how the projected solution is formed as well as information on the sensitivity of each POD mode in the projective integration process.
- A comparison of the numerical results between the on-line and off-line methods.

The paper is organized as follows. In Section 2, we present the algorithms of the POD-assisted projective integration and related methodologies. The analysis

of the accuracy and stability of the projective forward Euler (PFE) method are demonstrated in Section 3. Numerical results follow in Section 4, where the one-dimensional viscous Burgers' equation is used as the illustrative prototype. We summarize the results and provide a brief discussion in Section 5.

2 Methodologies

2.1 Proper orthogonal decomposition

The Proper Orthogonal Decomposition (POD) procedure extracts empirical orthogonal features from any ensemble of data. This linear procedure produces a useful reduced basis set that is optimal in the L^2 sense. In the POD framework for continuous problems, see Bekooz, Holmes, and Lumley (1993), we can represent a flow field $\mathbf{u}(t,\mathbf{x})$ as follows:

$$\mathbf{u}(t,\mathbf{x}) \;=\; \sum_{k=0}^{\infty} a_k(t)\phi_k(\mathbf{x}), \tag{1}$$

where $\{\phi_k(\mathbf{x})\}$ is the set of POD bases determined by first determining the $\{a_k(t)\}$ from the eigenvalue problem

$$\int_A C(t,t')a_k(t')dt' \;=\; \hat{\lambda}_k a_k(t)\,, \quad t \in A\,, \tag{2}$$

where $\{a_k(t)\}$ is the set of temporal modes, A is a specified time interval, and $C(t,t')$ is the correlation function defined by

$$C(t,t') \;=\; \int_\Omega \mathbf{u}(t,\mathbf{x})\cdot\mathbf{u}(t',\mathbf{x})d\mathbf{x}. \tag{3}$$

The POD basis is thus defined by

$$\phi_k(\mathbf{x}) \;=\; \int_A a_k(t)\mathbf{u}(t,\mathbf{x})dt, \quad \forall k. \tag{4}$$

The non-negative definiteness of the correlation function (3) allows us to order the eigenvalues and the corresponding POD modes by $\hat{\lambda}_k \geq \hat{\lambda}_{k+1}$. The POD expansion coefficients for (1) can be found from $a_k =< \mathbf{u}(t,\mathbf{x}), \phi_k(\mathbf{x}) >$. Here $<,>$ denotes the inner product operator in the L^2 sense.

2.2 POD-assisted projective integration

The projective integration technique allows us to integrate numerical solutions forward in time using only two processes: restriction and lifting. We introduce the

definitions of these processes by using two operators: a *restriction operator* $\mathscr{R}$ and a *lifting operator* $\mathscr{L}$ such that

$$\mathbf{a}(t) \;=\; \mathscr{R}\,\mathbf{u}(t,\mathbf{x}) \qquad\qquad \equiv\; \{<\mathbf{u}(t,\mathbf{x}),\phi_k(\mathbf{x})>,\; t\in A,\forall k\}, \tag{5}$$

and

$$\mathbf{u}(t,\mathbf{x}) \;=\; \mathscr{L}\,\mathbf{a}(t) \;\equiv\; \sum_{k=0}^{\infty} a_k(t)\phi_k(\mathbf{x})\,. \tag{6}$$

In a discrete computation, we can approximate (1) using K terms of POD expansion. The representation can be expressed as

$$\mathbf{u}_K(t,\mathbf{x}) = \sum_{k=1}^{K} a_k(t)\phi_k(\mathbf{x}), \tag{7}$$

and the *truncated restriction* and *truncated lifting* operators are defined as $\mathscr{R}_K$ and $\mathscr{L}_K$, respectively. The convergence of the K-terms POD expansion is assumed to be in the form of

$$\|\mathbf{u}-\mathbf{u}_K\| \to K^{-\gamma}, \quad \text{as } K\to\infty, \tag{8}$$

where the convergence rate, $\gamma>0$, is sufficiently large.

In general, we can write the evolution of the POD coefficient $\mathbf{a}(t)$ using

$$\frac{d\mathbf{a}}{dt} \;=\; \mathbf{g}(\mathbf{a}(t))\,, \tag{9}$$

where the explicit form of $\mathbf{g}$ may remain unknown. Thus, the derivative of the POD coefficients must be approximated rather than explicitly evaluated, in order to march forward in time. Note that we can find an explicit form of $\mathbf{g}$ by projecting the governing PDEs onto the POD modes, see Cazemier, Verstappen, and Veldman (1998); Deane, Kevrekidis, Karniadakis, and Orszag (1991); Ma, Karamanos, and Karniadakis (2000); Ma and Karniadakis (2002); Sirisup and Karniadakis (2004); Noack, Afanasiev, Morzyński, Tadmor, and Thiele (2003).

In this study, the "fine-scale" simulator gives a fully resolved solution of Burgers' equation using the standard Fourier spectral method. The "coarse-grained" model is that of solution dynamics (from initial conditions) on the slow manifold; the dynamics are observed on only the first few POD modes that parametrize this manifold.

In general, one large POD-assisted projective integration step to march the system from $t=t^n$ to $t=t^{n+1}$ consists of the following substeps:

1. *Fine-scale computation*: Solve Burgers' equation for a short period of time for $t^n \leq t \leq t_c^n = t^n + n_f \delta t$. The computation is conducted via the standard Fourier spectral method with a small time step δt. Here, the local relaxation time ($n_f \delta t$) is assumed to be shorter than the typical coarse-grained flow time scale (substep 3, below).

2. *Restriction*: Derive the POD coefficients using the previously saved solutions from the previous step, i.e., solve the eigenvalue problem (2) and estimate the time derivatives $d\mathbf{a}/dt$ at $t = t_c^n$.

3. *Projective integration*: March $\mathbf{a}(t)$ from t^n to t^{n+1} using any standard ODE technique to obtain $\mathbf{a}(t^{n+1})$. The time step here is $\Delta t_c \equiv n_c \delta t = t^{n+1} - t_c^n$, where $n_c \geq n_f \geq 1$.

4. *Lifting*: At $t = t^{n+1}$, reconstruct the solution $\mathbf{u}_K(t^n, \mathbf{x}) = \mathscr{L}_K \mathbf{a}(t^n)$ for a specific number of POD modes, K.

5. Return to substep 1. Note that the solution from substep 4 is set to an initial condition for the next fine-scale computation. Repeat the computation until the final time is reached.

Further details of substeps 2, 3, and 4 are given below.

2.2.1 Restriction and lifting

We employ the *snapshot method* to extract the set of POD bases $\{\phi_k(\mathbf{x})\}$ from the ensemble of previously saved solutions, see Sirovich (1987). In the fine-scale time interval, the solution snapshots $\mathbf{u}(t_i, \mathbf{x})$ at time t_i are obtained by solving Burgers' equation using an accurate spectral method where $t^n \leq t_i \leq t_c^n$, $i = 1, .., n_f$. From (4), the POD bases are then determined discretely by

$$\phi_k(\mathbf{x}) \;=\; \sum_{i=1}^{n_f} a_k(t_i)\, \mathbf{u}(t_i, \mathbf{x})\, dt, \quad \forall k. \tag{10}$$

where $\{a_k\}$ are obtained by solving the correlation matrix (2). Once the POD basis functions are determined from (10), we can restrict any solution $\mathbf{u}(t, \mathbf{x})$ for any given t to obtain the corresponding POD coefficients a_k from (5). The derivative of POD coefficients can then be approximated and used to march forward in time via the projective integration technique (see below). The lifting procedure is the reverse process of restriction, i.e., for a given set of computed POD coefficients at time t, we can reconstruct the corresponding solution by using (6).

2.2.2 Projective integration

The projective integration procedure is described as follows:

- Approximate the RHS of (9) at $t = t_c^n$ via

$$\mathbf{g}(t_c^n) = \sum_{j=0}^{n_e} \alpha_j \mathbf{a}(t_j) = \frac{d\mathbf{a}}{dt}(t_c^n) + O\left(\delta t^{J_f}\right), \tag{11}$$

  where $1 \leq n_e \leq n_f$, $t_j = t_c^n - j\delta t$, and J_f denotes the order of the approximation. Here, $\{\alpha_j\}_{j=0}^{n_e}$ is a set of consistent coefficients such that $\sum \alpha_j f(t_j) = df/dt(t_c^n) + O(\delta t^{J_f})$.

- Once the RHS of the typical reduced-order model (9) is estimated numerically, we can effectively integrate it via standard ODE solvers. For instance, given a coarse time step $\Delta t_c \equiv n_c \delta t$ where $n_c \geq 1$, such that $t^{n+1} = t_c^n + \Delta t_c = t^n + (n_f + n_c)\delta t$, the single-step forward Euler projective integrator takes the form

$$\mathbf{a}(t^{n+1}) = \mathbf{a}(t_c^n) + \Delta t_c \cdot \mathbf{g}(t_c^n) + O(\Delta t_c^2). \tag{12}$$

It should be noted that other higher-order explicit integration schemes (possibly implicit ones) can be used as well. For instance, we can use the following scheme:

$$\mathbf{a}(t^{n+1}) = \mathbf{a}(t_c^n) + \sum_{k=1}^{J_c} \frac{(\Delta t_c)^k}{k!} \frac{\partial^{(k-1)}}{\partial t^{k-1}} \mathbf{g}(t_c^n) + O(\Delta t_c^{J_c+1}). \tag{13}$$

The higher-order temporal derivatives of $\mathbf{g}(t)$ are approximated in a way similar to (11). Note that (13) is a high-order single-step method.

2.3 Projective Forward Euler Method (PFE)

The global time for projective integrators are composed of two types of integrators: fine-scale integrator and coarse-scale integrator. We start the computations via fine-scale integration with n_f time steps and then perform coarse-scale integration with n_c time steps. In this study, fine-scale integration is performed using the Fourier spectral method, whereas coarse-scale integration is carried out using the single-step forward Euler method. Here, we apply the Euler method in order to check the stability of the PFE method by comparing it with some predictions from linear stability analysis. Details of the analysis will be presented in the next section.

Following Gear and Kevrekidis (2003b), we divide the computational stages in the PFE method into several steps, as follows.

1. Use a suitable fine-scale integrator to integrate the solutions for n_f time steps, say from t^n to t_c^n.

2. Approximate $d\mathbf{a}/dt$ at $t = t_c^n$.

3. Perform outer integration with n_c steps using dy/dt at time $t = t_c^n$ via

$$\mathbf{a}_{n+n_f+n_c} = (n_c+1)\mathbf{a}_{n+n_f} - n_c\mathbf{a}_{n+n_f-1}\,.$$

Here, we approximate $d\mathbf{a}/dt$ at time $t = t_c^n$ via the Euler method at points n_f and $n_f - 1$.

3 Accuracy and Linear Stability of the PFE method

3.1 Accuracy of the PFE method

Recently, a detailed analysis of the consistency and accuracy of the "off-line" POD-assisted projective integration method has been presented in Sirisup, Karniadakis, Xiu, and Kevrekidis (2005). The resultant analysis can be applied to the proposed ("on-line") method. The main results are summarized here as follows.

Let $\mathbf{v}^{n+1}$ and $\mathbf{u}^{n+1}$ be an exact solution and numerical solution, respectively, at time t^{n+1} . Suppose that we employ one step of the PFE method with K POD modes to approximate the exact solution in one global time step $\Delta t = \Delta t_c + \Delta t_f$. The error from the approximation $\mathbf{u}_K^{n+1}$ against the exact flow field $\mathbf{v}$ at any time t^{n+1} can be written as:

$$\begin{aligned}
\varepsilon_T &= \left\|\mathbf{u}_K^{n+1} - \mathbf{v}(t^{n+1})\right\| \\
&\leq \left\|\mathbf{u}_K^{n+1} - \mathbf{v}_K(t^{n+1})\right\| + \left\|\mathbf{v}_K(t^{n+1}) - \mathbf{v}(t^{n+1})\right\| \\
&\sim \Delta t_f\,\varepsilon_f + O\left(\Delta t_c^2\right) + O\left(\delta t^{J_f}\right) + O\left(K^{-\gamma}\right) \qquad (14)
\end{aligned}$$

The total error ε_T is composed of four error terms on the RHS of (14). $\varepsilon_f \sim O(\delta t^p, h^q)$ is the error from the fine-scale computation. The second term is the error from the coarse-scale computation. The third term is the error due to the approximation of $\mathbf{g}(\mathbf{a}(t))$, which is not known exactly in closed-form formulae, and the last term is the error from the convergence of POD representations.

For very accurate fine-scale computation, we obtain $O(\Delta t_f) \ll O(\Delta t)$. Thus, the dominant error terms arise from the last three terms. In the case of a highly effective method, n_c must be large, so the error term $O(\Delta t_c^2)$ dominates other error terms, and

it grows very rapidly when we march the numerical solution in time. However, for a large K, the error from approximation of (9) at $t = t_c^n$ can dominate other error terms because of aliasing. A technique like non-uniform sampling near $t = t_c^n$ could be used as a remedy for this situation. In practice, the appropriate values of Δt_c and Δt_f providing the most efficient PFE method are not known in advance. One way to analyze the relation between Δt_c and Δt_f is to use the concept of linear stability. A detailed analysis of this method will be given in the next subsection.

3.2 Linear stability analysis of the PFE method

In this subsection, we analyze the stability of the PFE method. Following the analysis in Gear and Kevrekidis (2003b), the characteristic polynomial $\sigma(z)$ of the inner and outer integrations in one global time step can be written as:

$$\sigma(z) \;=\; [(n_c+1)\rho - n_c]\rho^{n_f-1}\,. \tag{15}$$

Here, $\rho = \rho(z)$ is the amplification factor of the inner integration method. The region of absolute stability in the complex z-plane is the set of z for which $|\sigma(z)| \leq 1$. For simplicity, when the inner integrator is the forward Euler method, the stability region is the set of z such that

$$[(n_c+1)(1+z) - n_c](1+z)^{n_f-1} \;\leq\; 1\,. \tag{16}$$

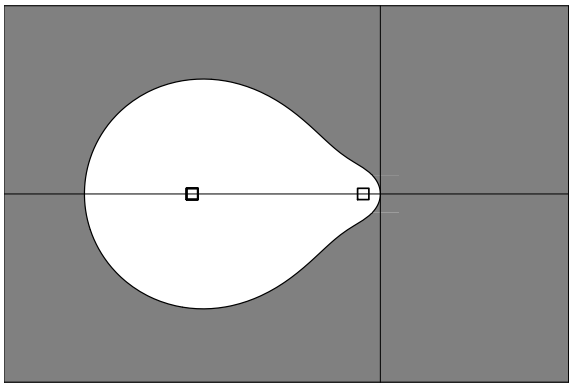

Figure 1: Stability region for $n_f = 5$ and $n_c = 10$ when the inner and outer integrators are the Euler method.

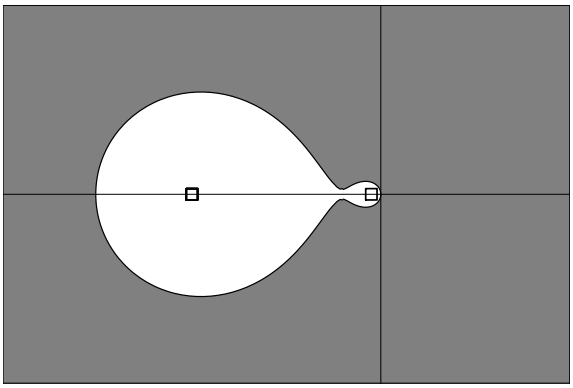

Figure 2: Stability region for $n_f = 5$ and $n_c = 19$ when the inner and outer integrators are the Euler method.

The stability regions when we use the forward Euler method for both the inner and the outer integrators are shown in Figures 1 and 2. When n_f is fixed, it is found that the stability region is divided into two regions as n_c increases. The critical value at which the stability region begins to be divided on the negative real axis is $n_c/n_f \approx 3.9$. This implies that the values of n_c and n_f must be set so that this ratio is maintained to ensure the stability of the method. Thus, one way to investigate the stability of the PFE method is to fix n_f and vary n_c. Moreover, the efficiency of the method can be observed using this variation because the maximum value of n_c can be approximately determined when n_f is specified. Some numerical results will be presented in the next section.

Next, we analyze the stability of the PFE method when we use the higher-order method as inner integration. For instance, the amplification factor when inner integration uses the fourth-order Runge–Kutta method (RK4) is

$$\rho(z) \;=\; 1 + z + \frac{z^2}{2!} + \frac{z^3}{3!} + \frac{z^4}{4!}. \tag{17}$$

Substituting (17) into (15), we obtain

$$\begin{aligned} &\left\{(n_c+1)(1+z+\tfrac{z^2}{2!}+\tfrac{z^3}{3!}+\tfrac{z^4}{4!})-n_c\right\} \\ &\quad\times\left\{1+z+\tfrac{z^2}{2!}+\tfrac{z^3}{3!}+\tfrac{z^4}{4!}\right\}^{n_f-1} \le 1 \end{aligned} \tag{18}$$

The set of z satisfying (18) in the complex plane shows the stability region when in-

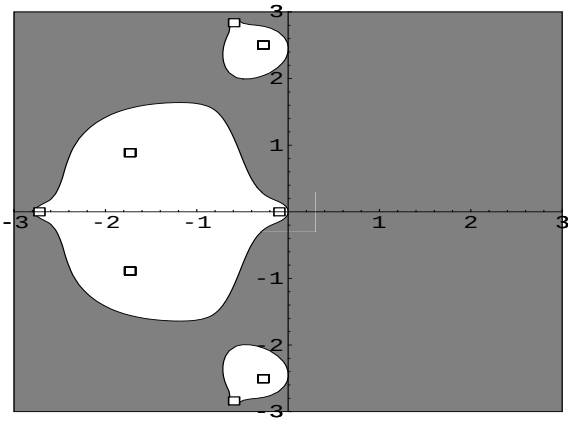

Figure 3: Stability region for $n_f = 5$ and $n_c = 10$ when the inner integrator is the RK4 method and the outer integrator is the Euler method.

ner integration is performed via the RK4 method, while outer integration is carried out by the forward Euler method. Figure 3 shows the stability region when $n_f = 5$ and $n_c = 10$ (exactly three regions), while Figure 4 shows the stability region when $n_f = 5$ and $n_c = 20$ (three smaller regions, including a point or very small region at the origin). The critical value at which the stability region starts to be divided on the negative real axis is $n_c/n_f \approx 3.12$. In this case, the critical value is smaller than that of the previous case (where the inner integrator is the first-order method). We will compare this finding of the ratio of n_c to n_f with numerical experiments in the next section.

It should be noted that the explicit and implicit multi-step methods can be used as the outer integrators. Detailed studies and applications of these methods for solving stiff ODEs can be found in Lee and Gear (2007) and Gear and Kevrekidis (2003a).

4 Numerical Results

In order to demonstrate the PFE method, Burgers' equation, which is a simple one-dimensional model of the Navier-Stokes equations, is chosen as a demonstration model. Fine-scale computation is performed by the Fourier spectral method. Some details of the method are summarized as follows.

The one-dimensional viscous Burgers' equation for unknown $u(x,t)$ can be written as

$$u_t + \nu u_{xx} + u u_x = 0, \; 0 \leq x \leq L \tag{19}$$

where L is a given computational domain, and ν is the viscosity effect.

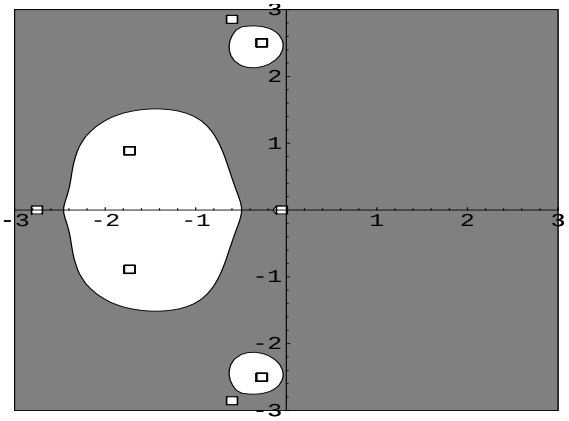

Figure 4: Stability region for $n_f = 5$ and $n_c = 20$ when the inner integrator is the RK4 method and the outer integrator is the Euler method.

Let $\hat{u}$ be the discrete Fourier transform of u, defined by

$$\hat{u}(k,t) = F(u) = \frac{1}{N}\sum_{j=0}^{N-1} u(x_j,t)\exp(-ikx_j)\,,\ -p \le k \le p-1\,,$$

where p and N are the number of Fourier modes and discretization points of x, respectively. Applying the discrete inverse Fourier transform, we obtain

$$u(x_j,t) = F^{-1}(u) = \sum_{k=-p}^{p-1} \hat{u}(k,t)\exp(ikx_j)\,,\ 0 \le j \le 2p-1$$

where F and F^{-1} respectively denote the discrete Fourier transform and the inverse Fourier transform.

In discrete form, (19) can be written as

$$u_t(x_j,t) \ = \ -F^{-1}\left\{F(u^2)ik/2\right\} + \ F^{-1}\left\{\nu k^2 F(u)\right\}\,,\ 0 \le j \le 2p-1. \tag{20}$$

Let $\mathbf{u} \ = \ [u(x_0,t), u(x_1,t), ..., u(x_{2p-1},t)]^T$, and (20) can be written as the system of ODEs at the collocation points:

$$\mathbf{u}_t \ = \ \mathbf{R}(\mathbf{u})\,. \tag{21}$$

To reduce the computational time, we apply the discrete fast Fourier transform (FFT) algorithm to the RHS of (21). We march forward in time by using the clas-

sical Runge–Kutta method. Hence, the problem can now be solved numerically subject to appropriate boundary conditions.

The solution $u(x,t)$ represents a traveling wave along a flat horizontal bottom in the domain $0 < x < L$. The boundary conditions can be approximated as $u(0,t) = u(L,t) = 0$ for $t > 1$, whereas the initial condition is given by

$$u(x,1) = \frac{x}{1+\exp[\frac{1}{4\nu}(x^2-\frac{1}{4})]}, \quad 0 < x < L. \tag{22}$$

This problem has an exact solution (see Nguyen and Reynen (1982)) in the form of

$$u(x,t) = \frac{x/t}{1+(t/t_0)^{1/2}\exp(x^2/4\nu t)}, \quad t \geq 1, \tag{23}$$

where $t_0 = \exp(1/8\nu)$. The solution represents a nonlinear wave propagating to the right with decreasing amplitude due to the viscosity effect. We will apply the PFE method and check its accuracy by comparing the numerical solutions with these exact solutions at various times in the next section.

4.1 Numerical results of the PFE method

The evolution of traveling wave profiles $u(x,t)$ is shown in Figure 5. The horizontal axis represents the x domain, while the vertical axis represents the wave amplitude. The solid and the dashed lines depict the exact solutions and the numerical solutions, respectively. Here, we set the number of POD modes at $K = 3$, the inner time steps at $n_f = 5$, and the outer time steps at $n_c = 10$. The viscosity effect is set at $\nu = 0.005$ for all presented results. The numerical results and the exact solutions are in very good agreement. These results ensure the accuracy and stability of the PFE method.

To investigate the accuracy of the PFE method, we define three forms of error as follows.

$$\begin{aligned} E_{t_j} &= \frac{1}{N}\left\{\sum_{i=1}^{N}\left|\frac{q_{\text{exact}}(i,t_j)-q_{\text{approx}}(i,t_j)}{q_{\text{exact}}(i,t_j)}\right|\right\} \\ E_{\text{avg}} &= \frac{1}{T_n}\left\{\sum_{j=1}^{nt} E_{t_j}\Delta t\right\}, \end{aligned}$$

$$\text{and} \quad E_{T_n} = \frac{1}{N}\left\{\sum_{i=1}^{N}\left|\frac{q_{\text{exact}}(i,T_n)-q_{\text{approx}}(i,T_n)}{q_{\text{exact}}(i,T_n)}\right|\right\},$$

where N is the number of mesh points, nt is the number of time steps, and $T_n = nt \cdot \delta t$ is the final time. Thus, E_{avg} is the accumulation of averaged error measured at the final time step, and E_{t_j} is the averaged error at an arbitrary time t_j.

The relationships between the number of POD modes K and the errors are shown in Figure 6. When fixing K, it was found that the errors E_{avg} and E_{Tn} increase as n_c increases. That is, the accuracy of the PFE method decreases as the coarse time step increases. In this case, the dominant error term is the truncation error of $O(\Delta t_c^2)$ on the RHS of (14).

We can see the effect of K by fixing both n_c and n_f and varying K. It can be seen from Figure 6 that the errors decrease as K increases, or equivalently, that we have used a sufficient number of POD modes for the expected convergence and desired accuracy to be obtained. This conclusion can be made only if $n_c \leq 13$. We can see in the case of a relatively large value of n_c ($n_c \geq 14$) that the error is very large even if we use many K. In this case, the error term from the coarse-scale computation is much larger than the error term from the POD convergence. This then directly affects all computations, resulting in a divergence of the numerical solutions.

4.2 Linear stability results for the PFE method

In numerical experiments, we investigated the stability of the PFE method by fixing $n_f = 5$, while the value of n_c was varied. It was found that n_c could be dramatically increased until $n_c = 14$. Referring back to the linear stability analysis, we showed that the critical value n_c/n_f is approximately 3.14. In our numerical experiments, we found that the maximum value of n_c is 14, or $n_c/n_f = 2.8$. Thus, the linear stability analysis shows good prediction. Instability of the PFE method usually occurs before the predicted critical value n_c/n_f is reached, because nonlinear effects are taken into account in the calculations. In addition, we found that the error in $O(\Delta t_c^2)$ increases as Δt_c increases. Thus, this error term is a direct factor in the stability of the PFE method.

In the case of $n_f \ll n_c$, the error term $O(\Delta t_c^2/\Delta t) = O(n_c^2 \delta t/(n_f + n_c)) \sim O(n_c \delta t)$. The truncation error is an approximate product of n_c and δt. Thus, we can set n_c to be very large provided that δt is very small. It can be seen from the linear stability analysis that the stability region can be separated into four regions when $n_c/n_f \approx 3.14$. The smallest stability region appears near the origin in the complex plane. In fact, this region shrinks to a stable point at the origin as n_c becomes very large (see Figure 4). This result from the linear stability analysis again confirms the stability of the PFE method. That is, for very small δt ($\delta t = 0.0001$), it is still possible to use a very large coarse time step ($n_c = 200$ and $n_f = 100$). Although this can be done, it is inefficient in practice. Note that it is very difficult to obtain the derivation of the relationship between these parameters to a desired order of accuracy. In addition,

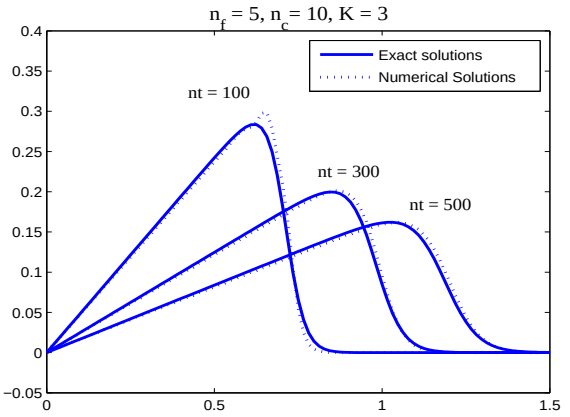

Figure 5: Traveling waves at various time steps

the method of selecting the number of POD modes in the PFE method is also not clear and is not easy to clarify. However, the notion of how to choose the number of POD modes in the PFE method should be observed. Fortunately, the behavior of each POD mode can be individually investigated, which can be achieved only by the "on-line" method. We will observe the function of each POD mode in the next section.

4.3 Interplay of POD modes

In this part of the paper, we will focus on the interplay between POD modes in the "on-line" POD-assisted projective integration method. The traditional low-dimensional POD system obtained from Galerkin projection will be inconsistent with the original systems for long-term model integration, see Jolly, , Kevrekidis, and Titi (1990); Sirisup and Karniadakis (2004). In some cases, the method requires many POD modes in order to achieve stability during the model integration, see Deane, Kevrekidis, Karniadakis, and Orszag (1991). This makes it impossible to understand the roles of each individual POD mode in the integration process. Thus, one way to individually study the characteristics of each POD mode, especially of low modes and the interplay of those modes, is to use this approach.

We will focus our study on the first four modes as determined by their energy distribution. We have checked that only the first four modes are sufficient in our simulations. The appropriate number of POD modes, in general, should depend on the dynamics and characteristics for each problem. In order to determine the role of each individual POD mode, we perform the "on-line" POD-assisted projective integration with only those modes of interest. The wave profiles $u(x,t)$ of numerical

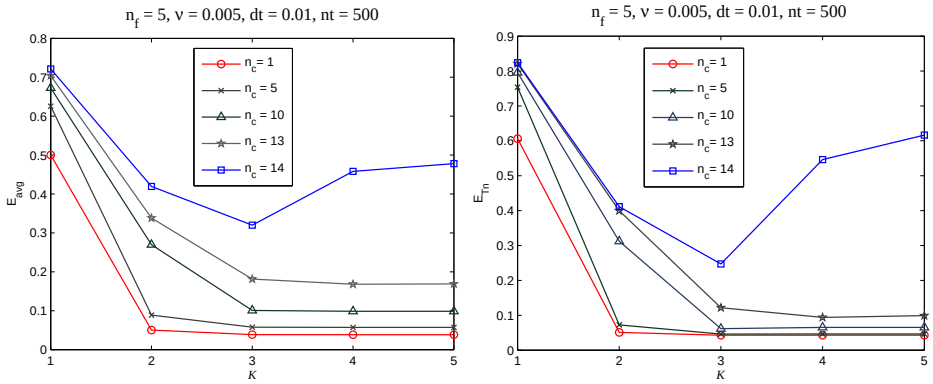

Figure 6: Relationship between the number of POD modes K and averaged error E_{avg}, and E_{T_n}

results and exact solutions at $t = 3.01$ and 5.02 are shown in Figures 7–9. Important findings for each POD mode are summarized as follows:

- The first POD mode is responsible for providing a damped solution with no propagation speed. The wave profiles are shown in Figure 7 (left). The wave amplitude matches well with the exact solution. Moreover, we found that the damped solution maintains its profile even if the value of n_c is doubled (large projective integration time step).
- The second POD mode is responsible for providing the propagation speed

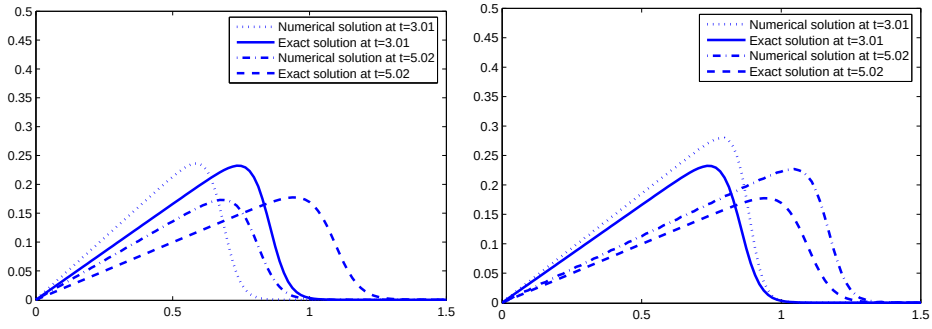

Figure 7: Comparison of the exact solutions and numerical solutions obtained from the PFE method with only the first POD mode (left), and only the second POD mode (right).

of the wave solution; see Figure 7 (right). The wave speed of the numerical solution agrees well with the speed of the exact solution. However, the wave profile has a high oscillation near the peak when the value of n_c is doubled.

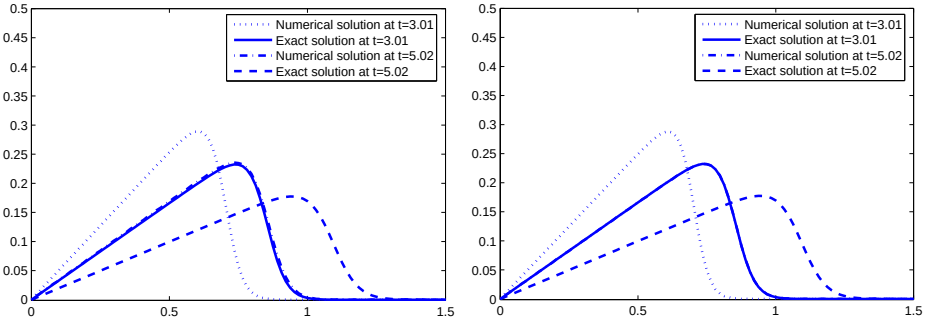

Figure 8: Comparison of the exact solutions and numerical solutions obtained from the PFE method with only the third POD mode (left), and only the fourth POD mode (right).

- The third POD mode or the fourth POD mode: These modes are responsible for a damped traveling wave solution analogous to the exact solution. However, both the wave speed and the wave amplitude are slower than those of the exact solution; see Figure 8. The profile of the damped traveling wave is maintained as the value of n_c is doubled.

The effect of a linear combination of the first four POD modes was investigated; see Figure 9. The numerical results and exact solutions are in good agreement. We also investigated the role of the higher POD mode. It was found that the higher POD modes do not contribute any significant characteristics to the solution. The reason for this is that the energy distribution of the higher POD modes is less than the energy of the first few POD modes. Thus, the combination of the first four POD modes alone is sufficient for this problem. This finding is also confirmed by the results presented in Figure 6 in that the results for $K = 3$, 4, and 5 have small error when $n_c \leq 13$.

Through the studies of our POD mode interplay, we revealed the role of each POD mode in representing the numerical solution of the 1D Burgers' equation. We found that the fourth POD mode does not play any significant roles from the third POD mode. This shows why it is sufficient to represent the exact solution with the combination of only the first four POD modes in the PFE method. However, in more

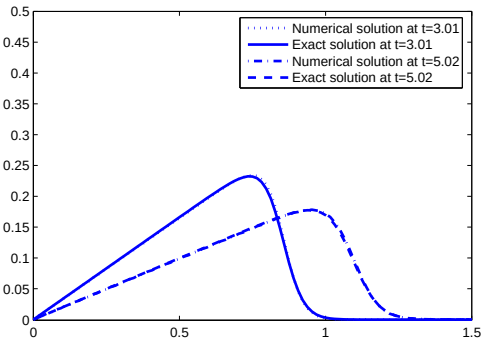

Figure 9: Comparison of the exact solutions and numerical solutions obtained from the PFE method with a combination of the first through fourth POD modes.

complicated phenomena such as turbulence flows, the number of representative POD modes usually require many POD modes. However, we do not know in advance the appropriate number of POD modes, since it depends on the characteristics of the problem being considered. However, the role of each POD mode in representing flow-field behavior can be analyzed in a similar manner as presented by our approach.

4.4 Numerical results of off-line projective integration

In this subsection, we apply equation-free projective integration, see Sirisup, Karniadakis, Xiu, and Kevrekidis (2005), to solve the same problem. This method is referred to as "off-line" because the POD modes are known by a priori computation to invoke the whole integration process. In this method, we need to employ a large number of snapshots in order to maintain accuracy and the characteristics of the dynamic behavior. POD modes are then extracted from the ensemble of data and used in the restriction and lifting processes. Here, we obtained 80 snapshots from the priori DNSs. These "off-line" POD modes govern the entire simulation dynamics.

The off-line projective integration method is different in concept from the on-line method (the present method) because in the later the POD modes are computed on-the-fly, which means that the POD modes are obtained by n_f snapshots from the DNS simulations for each large time-step projective integration. This is the main difference between the on-line and the off-line projective integration processes.

To investigate the efficiency of the off-line method, all parameters ν, dt, and n_f are then set to be the same values as in the numerical experiments of the on-line method. The relationship between averaged error E_{avg} and the number of POD

modes K of the off-line method is shown in Figure 10. We have shown four cases of n_c: 5, 10, 15, and 20. For $n_c = 5$ (relatively small jump in the projective step), we can see that the averaged error decreases as the number of POD modes increases. This is similar to what we have observed in the results of the current method. As K increases, the error does not decrease, because the higher POD modes ($K > 40$) contribute very little to the accuracy of the solution, or equivalently, they possess very small energy when compared to the lower modes. The accuracy of the numerical results in the case of $n_c = 10$ is similar to that in the case of $n_c = 5$. However, in the case of $n_c = 15$, the averaged error is relatively large compared with the previous cases. The averaged error does not decrease with increasing K. Moreover, the numerical solution diverges as K becomes large. We can conclude that the truncation error of the coarse time-scale computation dominates in the case of large n_c, leading to the instability of the method. As we can see, the numerical solution diverges very rapidly in the case of $n_c = 20$.

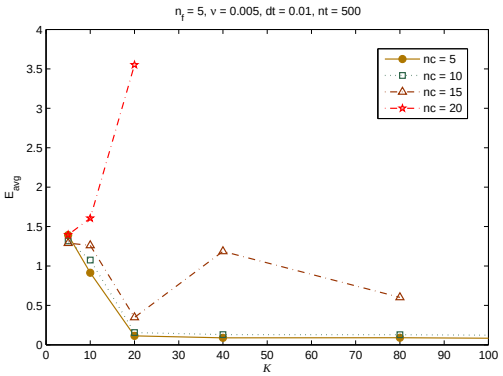

Figure 10: Relationship between POD modes K and averaged error E_{avg} for the off-line projective integration.

5 Conclusions

In this paper, we have applied the "on-line" POD-assisted projective integration method to solve numerically the one-dimensional viscous Burgers' equation. The method composes of two time-scale computations: fine-scale and coarse-scale. The fine-scale computation is performed by the Fourier spectral method, and the coarse-scale computation or the projective integration is carried out using the first-order forward Euler method.

The main objective of this study was to investigate the efficiency and stability of the proposed method. Various sets of numerical experiments have been carried

out. Efficiency in our work was measured by the percentage of the ratio of n_c to $n_c + n_f$, or equivalently, the ratio of the number of coarse-time steps to the number of time steps in one large projective integration. We have found that the method can produce very accurate results, while the computational effort was reduced by up to 72%. The stability of the method was investigated numerically. It was found that the method is stable when the coarse time step is small, while it is unstable when the coarse time step is large. This results in a large truncation error in the projective integration where the increment of representative POD modes cannot reduce the total error. We also found that the results from the linear stability analysis agree well with the numerical results. This finding provides the maximum value of the number of coarse time steps in order to obtain the greatest efficiency. In general, we can also apply the higher-order integrator as the coarse-scale integrator; for instance, we can use the fourth-order single-step Runge–Kutta method or the multi-step predictor-corrector method. This would result in a more effective method. Accuracy and stability analysis by these methods remain open issues. The numerical study of POD mode interplay was also presented. It revealed the characteristics and functions of each POD mode in representing the dynamics of traveling wave solutions. It was found that the second POD mode causes more oscillations in the solution than other POD modes do. The role of the third and the fourth mode in representing the wave solution is the same. These numerical observations provide important ideas for choosing a suitable number of POD modes. In general, each POD mode interplay cannot be studied individually by using traditional low-dimensional modeling, and here we have presented one method that can be applied. Moreover, this concept can be extended to the study of other complex problems, including higher-dimensional problems. In the current work, although we have applied a method for solving the prototype viscous Burgers' equation, the method can be extended to any dissipative PDEs. The important limitation of the method is that the projected domain must rely on the existence of a relatively low-dimensional model or attracting slow manifolds parametrized by the representative POD modes. In complex systems such as atmospheric or ocean system, time spent in the fine scale computation step can necessarily be much longer which directly affects the overall efficiency of the proposed method. More research which includes time required in fine scale computation step, longest projection time step as well as optimal number of POD modes is still needed in order to achieve the highest efficiency and stability regarding this method as it is being applied to solve such systems.

Acknowledgement: This research work was financially supported by Kurdi.

References

Arifin, N.; Noorani, M.; Kilicman, A. (2007): Modelling of marangoni convection using proper orthogonal decomposition. *Nonlinear Dynamics*, vol. 48, no. 3, pp. 331–337.

Bekooz, G.; Holmes, P.; Lumley, J. (1993): The proper orthogonal decomposition in the analysis of turbulent flows. *Ann. Rev. Fluid Mech.*, vol. 25, pp. 539–575.

Cazemier, W.; Verstappen, R.; Veldman, A. (1998): Proper orthogonal decomposition and low-dimensional models for driven cavity flows. *Phys. Fluids*, vol. 10 (7), pp. 1685–1699.

Deane, A.; Kevrekidis, I.; Karniadakis, G.; Orszag, S. (1991): Low-dimensional models for complex geometry flows: Application to grooved channels and circular cylinders. *Phys. Fluids A*, vol. 3 (10), pp. 2337–2354.

Gear, C.; Kevrekidis, I. (2003a): Projective methods for stiff differential equations: problems with gaps in their eigenvalue spectrum. *SIAM J. Sci. Comput.*, vol. 24, pp. 1091–1106.

Gear, C.; Kevrekidis, I. (2003b): Telescopic projective methods for parabolic differential equations. *J. Comput. Phys.*, vol. 187, no. 1, pp. 95–109.

Gear, C.; Kevrekidis, I. (2004): Constraint-defined manifolds: a legacy code approach to low-dimensional computation. *J. Sci. Comput.*, vol. 25, no. 1, pp. 17–28.

Jolly, M.; ; Kevrekidis, I.; Titi, E. (1990): Approximate inertial manifolds for the Kuramoto-Sivashinsky equation: Analysis and computations. *Phys. D*, vol. 44, pp. 36–60.

Kevrekidis, I.; Gear, C.; Hummer, G. (2004): Equation-free: The computer-aided analysis of complex multiscale systems. *AIChE Journal 50 (7), pp. 1346-1355*, vol. 50, no. 7, pp. 1346–1355.

Kevrekidis, I.; Gear, C.; Hyman, J.; Kevrekidis, P.; Runborg, O.; Theodoropoulos, C. (2003): Equation-free coarse-grained multiscale computation: enabling microscopic simulators to perform system-level analysis. *Comm. Math. Sci.*, vol. 1, no. 4, pp. 715–762.

Lee, S.; Gear, C. (2007): Second-order accurate projective integrators for multiscale problems. *Journal of Computational and Applied Mathematics*, vol. 201, no. 1, pp. 258–274.

Ma, X.; Karamanos, G.; Karniadakis, G. (2000): Dynamics and low-dimensionality of the turbulent near-wake. *J. Fluid Mech.*, vol. 410, pp. 29–65.

Ma, X.; Karniadakis, G. (2002): A low-dimensional model for simulating 3d cylinder flow. *J. Fluid Mech.*, vol. 458, pp. 181–190.

Makeev, A.; Maroudas, D.; Kevrekidis, I. (2002): Coarse stability and bifurcation analysis using stochastic simulators: kinetic Monte Carlo examples. *J. Chem. Phys.*, vol. 116, pp. 10083–10091.

Makeev, A.; Maroudas, D.; Panagiotopoulos, A.; Kevrekidis, I. (2002): Coarse bifurcation analysis of kinetic Monte Carlo simulations: a lattice-gas model with lateral interactions. *J. Chem. Phys.*, vol. 117, pp. 8229–8240.

Missoffe, A.; Juillard, J.; Aubry, D. (2007): Reduced-order modelling of the reynolds equation for flexible structures. In *Technical Proceedings of NSTI Nanotech 2007*, pp. 137–140.

Nguyen, H.; Reynen, J. (1982): A spacetime finite element approach to burgerséquation. In *Numerical Methods for Non-Linear Problems, Vol.2*. Pineridge Publisher, Swansea.

Noack, B.; Afanasiev, K.; Morzyński, M.; Tadmor, G.; Thiele, F. (2003): A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *J. Fluid Mech.*, vol. 497, pp. 335–363.

Rambo, J.; Joshi, Y. (2007): Reduced-order modeling of turbulent forced convection with parametric conditions. *International Journal of Heat and Mass Transfer*, vol. 50, no. 3-4, pp. 539–551.

Ravindran, S. (2000): A reduced-order approach for optimal control of fluids using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids*, vol. 34, no. 5, pp. 425–448.

Rempfer, D. (2003): Low-dimensional modeling and numerical simulation of transition in simple shear flows. *Annual Review of Fluid Mechanics*, vol. 35, pp. 229–265.

Rico-Martinez, R.; Gear, C.; Kevrekidis, I. (2004): Coarse projective kMC integration: forward/reverse initial and boundary value problems. *J. Comput. Phys.*, vol. 196, pp. 474–489.

Russo, L.; Siettos, C.; Kevrekidis, I. (2007): Reduced computations for nematic-liquid crystals: A timestepper approach for systems with continuous symmetries. *Journal of Non-Newtonian Fluid Mechanics*, vol. 146, no. 1-3, pp. 51–58.

Samimy, M.; Debiasi, M.; Caraballo, E.; Serrani, A.; amd J. Little, X. Y.; Myatt, J. H. (2007): Feedback control of subsonic cavity flows using reduced-order models. *J. Fluid Mech.*, vol. 579, pp. 315–346.

Sirisup, S.; Karniadakis, G. (2004): A spectral viscosity method for correcting the long-term behavior of POD models. *J. Comput. Phys.*, vol. 194, no. 1, pp. 92–116.

Sirisup, S.; Karniadakis, G.; Xiu, D.; Kevrekidis, I. (2005): Equation-free/galerkin-free pod-assisted computation of incompressible flows. *J. Comput. Phys.*, vol. 207, no. 2, pp. 568–587.

Sirovich, L. (1987): Turbulence and the dynamics of coherent structures, Parts I, II and III. *Quart. Appl. Math.*, vol. XLV, pp. 561–590.

Xiu, D.; Kevrekidis, I.; Ghanem, R. (2005): An equation-free, multiscale approach to uncertainty quantification. *Computing in Science and Engineering*, vol. 7, no. 3, pp. 16–23.