International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:5, No:7, 2011

On-line and Off-line POD assisted projective integral for non-linear problems: A case study with Burgers' equation

Montri Maleewong and Sirod Sirisup

Abstract—The POD-assisted projective integration method based on the equation-free framework is presented in this paper. The method is essentially based on the slow manifold governing of given system. We have applied two variants which are the "on-line" and "off-line" methods for solving the one-dimensional viscous Bergers' equation. For the on-line method, we have computed the slow manifold by extracting the POD modes and used them on-the-fly along the projective integration process without assuming knowledge of the underlying slow manifold. In contrast, the underlying slow manifold must be computed prior to the projective integration process for the off-line method. The projective step is performed by the forward Euler method. Numerical experiments show that for the case of non-periodic system, the on-line method is more efficient than the off-line method. Besides, the online approach is more realistic when apply the POD-assisted projective integration method to solve any systems. The critical value of the projective time step which directly limits the efficiency of both methods is also shown.

Keywords—Projective integration, POD method, Equation-free.

I. 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 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 [1], [13], [16]–[18], [24], for example.

Montri Maleewong is with the Department of Mathematics, Kasetsart University, Bangkok, 10900 Thailand e-mail: Montri.M@ku.ac.th.
Sirod Sirisup is with the Large-Scale Simulation Research Laboratory, National Electronics and Computer Technology Center, 12120 Thailand e-mail:Sirod.Sirisup@nectec.or.th

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 [2]) and utilizes these structures to build a robust low-dimensional model. Used with the method of snapshots, first proposed in [23] 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 both the on-line and the off-line POD-assisted projective integration methodology that employs "equation-free" projective integration frameworks, pioneered by I.G. Kevrekidis *et al.* [7]. This framework has been applied to a variety of problems, ranging from the bifurcation analysis of complex systems to the homogenization of random media [8], [11], [12], [19], [20], [25]. 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 [6].

A similar POD-assisted projective integration approach has been successfully applied to solve the Navier-Stokes equation [22]. 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 while we are marching the numerical solution in time. This is called the "on-line" method. Full details of the on-line method are given in Section II-B. Moreover, we have presented the projective integral method based on [22], which is called the "off-line" method since the POD modes are known prior to integration process. From the application point of view, the on-line method is more practical than the off-line method. It can be applied to solve any complex systems with some modifications of the existing computer code. Moreover,

the approach can also be used to study the dynamics and interactions of the underlying slow manifold of any given systems as well.

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 if the POD modes are not available a priori for the projective integration process.
- An investigation of solution dynamic represented by each POD mode.
- A comparison of 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 of the projective forward Euler method is demonstrated in Section 3. Numerical results from the on-line and the off-line methods are presented in Section 4, where the one-dimensional viscous Burgers' equation is used as the illustrative prototype. We provide a brief discussion and summary in Section 5.

II. Methodologies

A. 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 [2], we can represent a flow field $\mathbf{u}(t,\mathbf{x})$ as follows:

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

where $\{\boldsymbol{\phi}_k(\mathbf{x})\}$ 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

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

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 $a_k =< \mathbf{u}(t,\mathbf{x}), \boldsymbol{\phi}_k(\mathbf{x}) >$. Here $<,>$ denotes the inner product operator in the L^2 sense.

B. 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* $\mathcal{R}$ and a *lifting operator* $\mathcal{L}$ such that

$$\mathbf{a}(t) \;=\; \mathcal{R}\,\mathbf{u}(t,\mathbf{x}) \equiv \;\{< \mathbf{u}(t,\mathbf{x}), \boldsymbol{\phi}_k(\mathbf{x}) >,\; t\in A, \forall k\}, \tag{5}$$

and

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

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

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

and the *truncated restriction* and *truncated lifting* operators are defined as $\mathcal{R}_K$ and $\mathcal{L}_K$, 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 [3], [4], [9], [10], [15], [21].

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 $\mathbf{u}_K(t^n,\mathbf{x}) = \mathcal{L}_K\mathbf{a}(t^n)$ 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.

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 [23]. 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

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

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) 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.

C. 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 [5], 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 $n_f - 1$.

III. Accuracy of the PFE method

A. 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 [22]. 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:

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

The total error ϵ_T is composed of four error terms on the RHS of (14). $\epsilon_f \sim O(\delta t^p, h^q)$ 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 the (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.

IV. 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

$$u_t + \nu u_{xx} + u u_x = 0\,,\ 0 \le x \le L \tag{15}$$

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

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

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

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

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

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

In discrete form, (15) can be written as

$$\begin{aligned}u_t(x_j,t) &= -F^{-1}\left\{F(u^2)ik/2\right\}\\ &\quad + F^{-1}\left\{\nu k^2 F(u)\right\}\,,\\ &\quad 0 \le j \le 2p-1.\end{aligned} \tag{16}$$

Let $\mathbf{u} = [u(x_0,t), u(x_1,t), ..., u(x_{2p-1},t)]^T$, and (16) can be written as the system of ODEs at the collocation points:

$$\mathbf{u}_t = \mathbf{R}(\mathbf{u})\,. \tag{17}$$

To reduce the computational time, we apply the discrete fast Fourier transform (FFT) algorithm to the RHS of (17). We march forward in time by using the classical 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

$$u(x,1) = \frac{x}{1+\exp[\frac{1}{4\nu}(x^2 - \frac{1}{4})]}\,,\quad 0 < x < L\,. \tag{18}$$

This problem has an exact solution (see [14]) in the form of

$$u(x,t) = \frac{x/t}{1+(t/t_0)^{1/2}\exp(x^2/4\nu t)}\,,\quad t \ge 1\,, \tag{19}$$

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. The viscosity effect is set at $\nu = 0.005$ for all cases of our simulations. It is noted that our current investigtion focuses on non-periodic system. We will apply the PFE method and check its accuracy by comparing the numerical solutions with the exact solutions at various times in the next section.

A. Numerical results of the PFE method

The evolution of traveling wave profiles $u(x,t)$ is shown in Figure 1. 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 from $K = 1$ to $K = 4$, the inner time step at $n_f = 5$, and the outer time step at $n_c = 5$. Cases of larger outer time step $n_c = 10$ are shown in Figure 2.

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.

In Figure 1, we apply relatively small outer time step, $n_c = 5$. The numerical results at each time step are in good agreement with the exact solutions when we use the number of POD mode $K = 3, 4$. When $K = 1$, the numerical results are clearly incorrect. This impiles that we use not enough number of POD mode in the online PFE method in order detect the dynamics of solution in time and also means that the first POD mode responsibles for the actual mean flow, and the second POD mode responsibles for the flow speed. In our simulations, we need to increase more number of POD mode to increase accuracy of numerical solutions. We found that it is enough to use $K = 3$, 4 and 5 in the simulations run by the PFE method.

In Figure 2, we apply relatively large outer time step, $n_c = 10$. The numerical results at each time step are in good

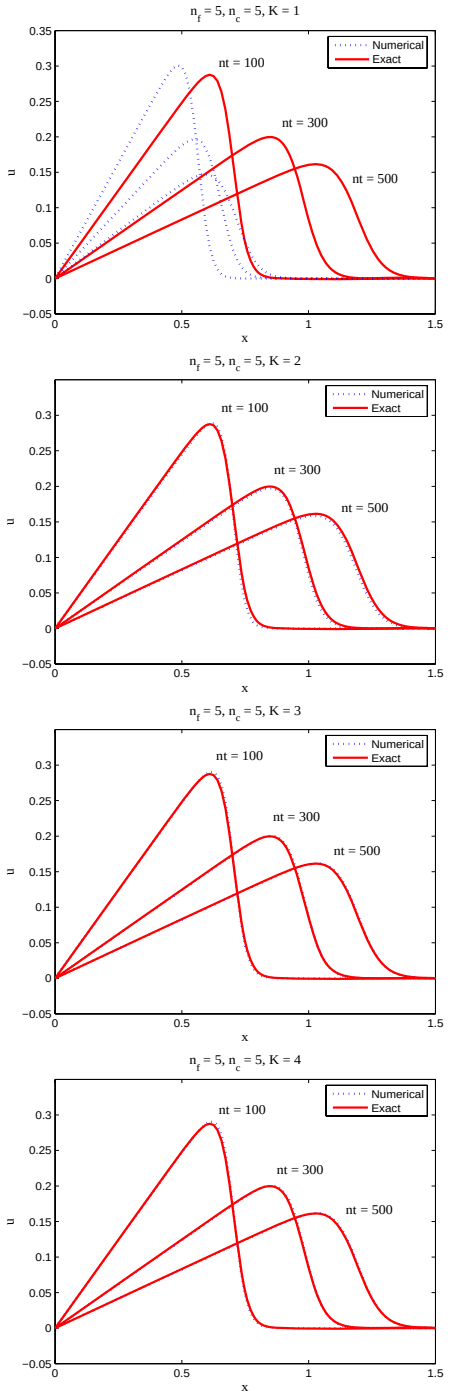

Fig. 1. Evolution of traveling wave profiles $u(x,t)$ by PFE method with POD modes $K = 1$ to 4 and $n_c = 5$.

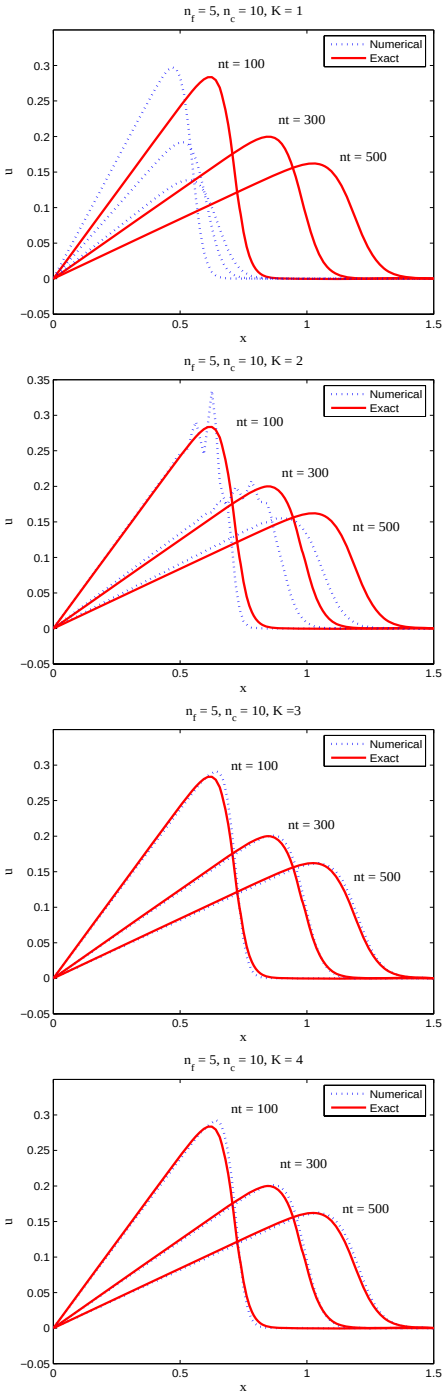

Fig. 2. Evolution of traveling wave profiles $u(x,t)$ by PFE method with POD modes $K = 1$ to 4 and $n_c = 10$.

agreement with the exact solutions when we use the number of POD mode $K = 3, 4$ but relative error still appears. Using $K = 2$ can improve dynamic solutions but with large error. In this case, the dominant error is contributed from the coarse scale time step in the PFE method. Moreover, this error term gets large when we continue to increase n_c. Figure 3 shows the numerical results when $n_c = 13$ with $K = 5$, the numerical solution is clearly incorrect at the early time step but the PFE method can recover inaccurate solution to be accurate solution when longer time step calculated. This is an important feature of the PFE method studied in our work that it can adapt the solution in time during marching. So, this method is sometimes

called 'on-line' method or projective integration performed on the fly.

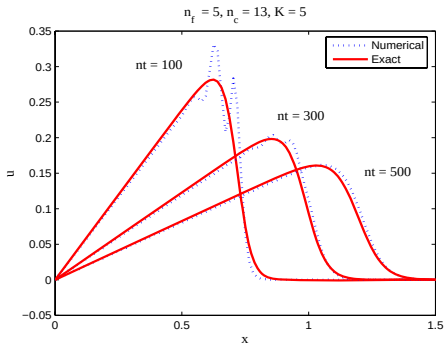

Fig. 3. Evolution of traveling wave profiles $u(x,t)$ by PFE method with POD modes $K = 5$ and $n_c = 13$.

The relationships between the number of POD modes K and the errors are shown in Figure 4. 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).

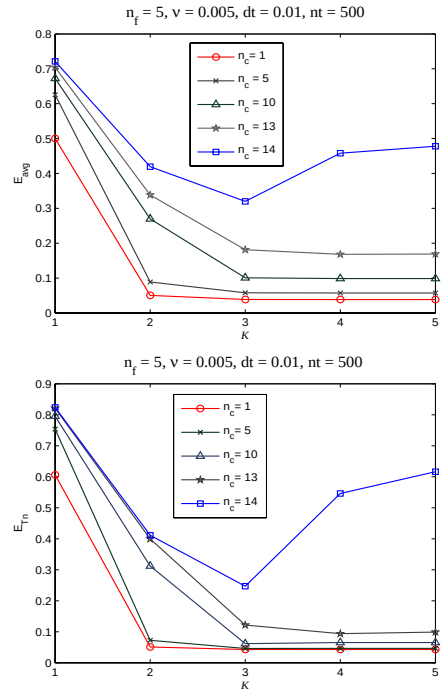

Fig. 4. Relationship between the number of POD modes K and averaged error E_{avg}, and E_{T_n}

We can see the effect of K by fixing both n_c and n_f and varying K. It can be seen from Figure 4 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.

B. Numerical results of off-line projective integration

In this section, we apply equation-free projective integration based on [22] to solve the same problem. This method is referred to as "off-line" because the POD modes are known prior to the computation. In this method, we need to employ a large number of snapshots to make sure that the POD modes have been fully resolved by the method of snapshots [23]. POD modes are extracted from the solution ensemble and will be later used in the restriction and lifting processes. In the current study, we use 80 snapshots in the solution ensemble. These POD modes govern simulation dynamics in entire integration period.

The concept of the off-line projective integration method is different from the on-line method that in the latter POD modes are computed on-the-fly. This means that POD modes are obtained by extracting n_f snapshots from the DNS simulations for each large projective integration step. However, off-line projective integration method uses the POD modes computed prior to the integration process. 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 chosen to be the same values as in the numerical experiments of the on-line method. The evolution of traveling wave profiles $u(x,t)$ is shown in Figure 5. Here, the number of POD modes are $K = 1, 5, 10$ and 20, the inner time step is $n_f = 5$, and the outer time step is $n_c = 5$. Similar to the study of on-line PFE method, the off-line projective integration with only the first POD mode is not enough to represent the solution accurately. The accuracy of the solution is obtained as we have increase K to $K = 20$. However, increasing number of POD modes used in the off-line projective integration to be more than $K = 20$ can result in unstability in the method eventhough by including more higher modes can lead to a more accurate solution. This is because the approximation to derivatives in projection step of highly fluctuated mode is very poor and the approximation error is also greatly amplified. Thus using these high modes in POD-assisted projective integration is not feasible in practice.

The case of a larger time step with $n_c = 13$ with $K = 40$ is shown in Figure 6. For the numerical experiments we have found that this value of n_c is the possible highest possible value for the current off-line projective integration and it is close to that of on-line one.

The relationship between averaged error E_{avg} and the number of POD modes K of the off-line method is shown in Figure 7. 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

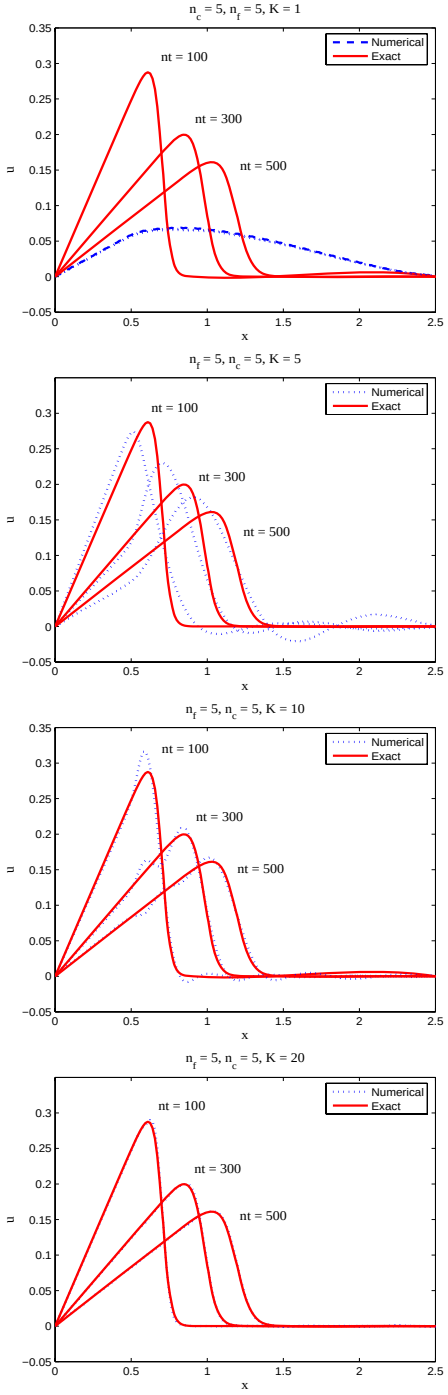

Fig. 5. Evolution of traveling wave profiles $u(x,t)$ by PFE method with POD modes $K = 1, 5, 10$ and 20 with $n_c = 5$

of POD modes increases. This is similar to what we have observed in the results of the on-line projective integration. 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

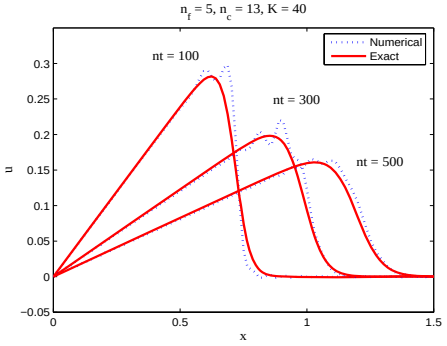

Fig. 6. Evolution of traveling wave profiles $u(x,t)$ by PFE method with POD modes $K = 40$ and $n_c = 13$

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 larger step with $n_c = 15$, the averaged error is relatively large compared to the previous cases. The averaged error does not decrease with increasing K. Moreover, the numerical solution diverges as K becomes too large. This is becuase 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 $n_c = 20$.

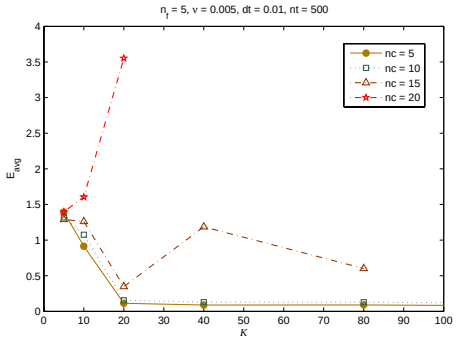

Fig. 7. Relationship between POD modes K and averaged error E_{avg} for the off-line projective integration.

TABLE I
AVERAGED ERRORS OF ON-LINE AND OFF-LINE

n_c	On-line	Off-line
5	0.01587	0.05062
8	0.02451	0.06337
10	0.04387	0.06887
12	0.04011	0.07564
13	0.04481	0.10307

Table I shows the comparison of averaged errors between the on-line and the off-line PFE methods. For the same value of outer time step, the on-line projective integration provides results with higher accuracy compared to those of the off-line projective integration. The on-line projective integration can

precisely detect solution dynamics while the off-line projective integration may miss some features if such features have not been observed in solution ensemble. In the on-line projective integration, the numerical solutions can be updated at the fine step and then used to construct the corresponding POD modes during the time integration. From this results and the implementation point of view we can see that the on-line projective integration method is rather efficient than the off-line projective integration method. However, the costs of repeatingly computed in on-line projective integration method should be assess before applying such aprroach to solve any given systems.

V. Conclusions

In this paper, we have applied both the on-line and off-line POD-assisted projective integration methods to solve numerically the one-dimensional viscous Burgers' equation. The methods compose 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 of the on-line projective integration method as well as compare that to the off-line projective integration method. Various sets of numerical experiments have been carried out. 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.

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. We have investigated the efficiency of the presented methods by changing from the first-order forward Euler method to be the second-order forward Euler method. It is found that averaged errors decreases but it is not significant and still be the same order of accuracy. When we increase the number of POD modes for both the on-line and off-line projective integration methods, the second-order methods are unstable because we have included high POD modes having high frequency for approximating time derivative at the projective step. Thus, using just first-order method would be efficient method in case of combinations of many POD modes.

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.

Acknowledgment

This research work was financially supported by Centre of Excellence in Mathematics (CEM) under the grant number RS-2-53-01-1.

References

[1] N.M. Arifin, M.S.M Noorani, and A. Kilicman. Modelling of marangoni convection using proper orthogonal decomposition. *Nonlinear Dynamics*, 48(3):331–337, 2007.

[2] G. Bekooz, P. Holmes, and J.L. Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Ann. Rev. Fluid Mech.*, 25:539–575, 1993.

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

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

[5] C.W. Gear and I.G. Kevrekidis. Telescopic projective methods for parabolic differential equations. *J. Comput. Phys.*, 187(1):95–109, 2003.

[6] C.W. Gear and I.G. Kevrekidis. Constraint-defined manifolds: a legacy code approach to low-dimensional computation. *J. Sci. Comput.*, 25(1):17–28, 2004.

[7] I.G. Kevrekidis, C.W. Gear, and G. Hummer. Equation-free: The computer-aided analysis of complex multiscale systems. *AIChE Journal 50 (7), pp. 1346-1355*, 50(7):1346–1355, 2004.

[8] I.G. Kevrekidis, C.W. Gear, J.M. Hyman, P.G. Kevrekidis, O. Runborg, and C. Theodoropoulos. Equation-free coarse-grained multiscale computation: enabling microscopic simulators to perform system-level analysis. *Comm. Math. Sci.*, 1(4):715–762, 2003.

[9] X. Ma, G.S. Karamanos, and G.E. Karniadakis. Dynamics and low-dimensionality of the turbulent near-wake. *J. Fluid Mech.*, 410:29–65, 2000.

[10] X. Ma and G.E. Karniadakis. A low-dimensional model for simulating 3d cylinder flow. *J. Fluid Mech.*, 458:181–190, 2002.

[11] A. Makeev, D. Maroudas, and I.G. Kevrekidis. Coarse stability and bifurcation analysis using stochastic simulators: kinetic Monte Carlo examples. *J. Chem. Phys.*, 116:10083–10091, 2002.

[12] A. Makeev, D. Maroudas, A. Panagiotopoulos, and I.G. Kevrekidis. Coarse bifurcation analysis of kinetic Monte Carlo simulations: a lattice-gas model with lateral interactions. *J. Chem. Phys.*, 117:8229–8240, 2002.

[13] A. Missoffe, J. Juillard, and D. Aubry. Reduced-order modelling of the reynolds equation for flexible structures. In *Technical Proceedings of NSTI Nanotech 2007*, pages 137–140, 2007.

[14] H. Nguyen and J. Reynen. A space-time finite element approach to burgers equation. In *Numerical Methods for Non-Linear Problems, Vol.2*. Pineridge Publisher, Swansea, 1982.

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

[16] J. Rambo and Y. Joshi. Reduced-order modeling of turbulent forced convection with parametric conditions. *International Journal of Heat and Mass Transfer*, 50(3-4):539–551, 2007.

[17] S.S. Ravindran. A reduced-order approach for optimal control of fluids using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids*, 34(5):425–448, 2000.

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

[19] R. Rico-Martinez, C.W. Gear, and I.G. Kevrekidis. Coarse projective kMC integration: forward/reverse initial and boundary value problems. *J. Comput. Phys.*, 196:474–489, 2004.

[20] L. Russo, C.I. Siettos, and I.G. Kevrekidis. Reduced computations for nematic-liquid crystals: A timestepper approach for systems with continuous symmetries. *Journal of Non-Newtonian Fluid Mechanics*, 146(1-3):51–58, 2007.
[21] S. Sirisup and G.E. Karniadakis. A spectral viscosity method for correcting the long-term behavior of POD models. *J. Comput. Phys.*, 194(1):92–116, 2004.
[22] S. Sirisup, G.E. Karniadakis, D. Xiu, and I.G. Kevrekidis. Equation-free/galerkin-free pod-assisted computation of incompressible flows. *J. Comput. Phys.*, 207(2):568–587, 2005.
[23] L. Sirovich. Turbulence and the dynamics of coherent structures, Parts I, II and III. *Quart. Appl. Math.*, XLV:561–590, 1987.
[24] M. Samimy *et. al*. Feedback control of subsonic cavity flows using reduced-order models. *J. Fluid Mech.*, 579:315–346, 2007.
[25] D. Xiu, I.G. Kevrekidis, and R. Ghanem. An equation-free, multiscale approach to uncertainty quantification. *Computing in Science and Engineering*, 7(3):16–23, 2005.