Research Article

Online Projective Integral with Proper Orthogonal Decomposition for Incompressible Flows Past NACA0012 Airfoil

Sirod Sirisup¹ and Montri Maleewong²,³

¹ Large-Scale Simulation Research Laboratory, National Electronics and Computer Technology Center, Prathum Thani 12120, Thailand
² Department of Mathematics, Faculty of Science, Kasetsart University, Bangkok 10900, Thailand
³ Centre of Excellence in Mathematics, CHE, Si Ayutthaya Road, Bangkok 10400, Thailand

Correspondence should be addressed to Montri Maleewong, montri.m@ku.ac.th

Received 27 April 2012; Revised 29 August 2012; Accepted 20 September 2012

Abstract

The projective integration method based on the Galerkin-free framework with the assistance of proper orthogonal decomposition (POD) is presented in this paper. The present method is applied to simulate two-dimensional incompressible fluid flows past the NACA0012 airfoil problem. The approach consists of using high-accuracy direct numerical simulations over short time intervals, from which POD modes are extracted for approximating the dynamics of the primary variables. The solution is then projected with larger time steps using any standard time integrator, without the need to recompute it from the governing equations. This is called the online projective integration method. The results by the projective integration method are in good agreement with the full scale simulation with less computational needs. We also study the individual function of each POD mode used in the projective integration method. It is found that the first POD mode can capture basic flow behaviors but the overall dynamic is rather inaccurate. The second and the third POD modes assist the first mode by correcting magnitudes and phases of vorticity fields. However, adding the fifth POD mode in the model leads to some incorrect results in phase-shift forms for both drag and lift coefficients. This suggests the optimal number of POD modes to use in the projective integration method.

1. Introduction

The projective integration methodology based on the “equation-free” framework has been successfully applied to analyze various kinds of engineering problems [1–3]. The key idea involves using short-duration initial simulations to estimate the time dynamics of the primary variables, without explicitly solving the governing equations either in full or reduced form. The data set obtained from the short-duration simulations can be effectively used to extract proper orthogonal decomposition (POD) basis set, which can then be used to estimate the time dynamics of the needed variables. The most efficient way to extract such a basis set is to employ the snapshots method proposed by Sirovich [4]. This “equation-free” framework has many advantages over the typical Galerkin approach. First, the reduced systems resulting from the Galerkin approach may result in spurious asymptotic states [5, 6]. Also it can be difficult to obtain an explicit form of the reduced system dynamics for a given set of governing equations. Due to these limitations, the equation-free framework of POD methods presents an attractive alternative for solving the problem, since the governing equations are not needed in explicit form during time marching.

The approach presented in this paper is called the “online” method which means that POD bases used in projective integration with “equation-free” framework are constructed from the snapshot data set collected from the full direct numerical simulation (DNS) during each projective integration step. Thus, the approach presented in this paper is different from the “offline” approach presented previously [2, 7]. In the offline approach, the global underlying POD modes are known prior to the projective integration. The online method is more practical than the offline method.
because we cannot know in advance the underlying POD basis as well as the appropriate number of POD modes used in the projective integration approach. Besides, in order to obtain that information, the representations of each POD mode should be known so that the number of POD modes can be chosen appropriately. To this end, we also propose a method and provide such information for the current projective integration.

In this paper, we apply the POD-assisted projective integration based on the online approach as an accelerated scheme for simulating incompressible fluid flows past the NACA0012 airfoil. This is one of the most popular two-dimensional flow benchmark problems. There are many numerical and experimental results [8, 9] that provide us data to investigate the accuracy and efficiency of the present scheme. Recently, Siegel et al. [10] use an alternative approach of artificial neural network to study the problem. A similar problem of NACA0015 airfoil has also been studied by the POD based method [11]. The primary objective of our studies is to investigate numerically the efficiency of the POD-assisted projective integration based on the online method. Another objective is to explore the functions of each POD mode employed in the POD-assisted projective integration.

This paper is organized as follows. Data acquisition and the methodology of the projective integration method are presented in Section 2. Numerical results from the application of the method to incompressible flows past the NACA0012 airfoil for both transient and periodic cases are shown in Section 3. The functions for each dominant POD mode in the online method are demonstrated in the POD mode interplay in Section 4. Finally, conclusions are made in Section 5.

2. Methodology

2.1. Navier-Stokes Equations. In this paper, we consider two-dimensional incompressible flow past the NACA0012 airfoil. The governing equations are the set of incompressible Navier-Stokes equations in the form of

\[
\nabla \cdot \mathbf{v} = 0, \\
\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} = -\nabla p + \frac{1}{Re} \nabla^2 \mathbf{v},
\]

where \( \mathbf{v} \equiv \mathbf{v}(t, \mathbf{x}) \) is the velocity field and \( p \equiv p(t, \mathbf{x}) \) the pressure term for \( \mathbf{x} \in \Omega \subset \mathbb{R}^2 \) and \( t \in \mathbb{R}^+ \). The equations are in dimensionless forms scaled by a characteristic velocity \( U \) and a characteristic length \( L \). The Reynolds number is defined as \( Re = UL/\nu \), where \( \nu \) is the kinematic viscosity.

The computational domain is shown in Figure 1. The object is the NACA0012 airfoil positioned at an angle of attack of 20° with respect to the horizontal axis. The points labeled 1–6 are the locations of probes to check the accuracy and efficiency of our method as discussed later. We impose the boundary conditions as follows. Uniform inflow is prescribed on the left boundary while periodic boundary conditions are specified on the top and bottom boundaries. We apply outflow condition on the right boundary, and no-slip condition on the surface of the airfoil. The full direct numerical simulations (DNS) are performed by solving the Navier-Stokes equations using the spectral/hp element method [12]. We have checked that the computational domain is long enough so that there is no effect from truncated domain. To obtain high accuracy of solutions, we apply the ninth-order Jacobi polynomial basis in each element. The full DNS results can then be regarded as the exact solutions in our problem.

2.2. Proper Orthogonal Decomposition. In the equation-free or Galerkin-free framework, proper orthogonal decomposition (POD) is the main procedure used to extract the global behavior of flow fields (snapshots) during calculations in time. Some basic concepts of POD have been summarized in this subsection. The POD procedure extracts empirical orthogonal features from any ensemble of data from the full set of DNS. The POD method is linear procedure that produces a reduced basis set which is optimal in \( L^2 \) sense. For continuous
problems [13], one can assume that the flow field \( u(t, x) \) can be represented as
\[
     u(t, x) = \sum_{k=0}^{\infty} a_k(t) \phi_k(x),
\]
where \( \{\phi_k(x)\} \) is the set of POD basis determined by the eigenvalue problem
\[
     \int_A C(t, t') a_k(t') dt' = \hat{\lambda}_k a_k(t), \quad t \in A,
\]
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
\[
     C(t, t') = \int_{\Omega} u(t, x) \cdot u(t', x) \, dx.
\]

The POD basis is thus defined by
\[
     \phi_k(x) = \int_A a_k(t) u(t, x) \, dt, \quad \forall k.
\]

The nonnegative definiteness of the correlation function (4) 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 (2) can be found by \( a_k = \langle u(t, x), \phi_k(x) \rangle \). Here \( \langle , \rangle \) denotes the inner product operator in \( L^2 \) sense.

2.3. The Equation-Free POD-Assisted Model. The equation-free form of the POD-assisted model in conjunction with the projective integration technique allows us to integrate numerical solutions forward in time by using two main processes: restriction and lifting. We introduce the definitions of

Figure 4: Transient dynamics: \( Re = 400 \), drag coefficient for \( T = 0.01 \) with various number of \( M \). From top to bottom: \( M = 3, 4, \) and 5.
these processes by two operators: a restriction operator $\mathcal{R}$ and a lifting operator $\mathcal{L}$ such that

$$a(t) = \mathcal{R}u(t, x) \equiv \{\langle u(t, x), \phi_k(x) \rangle, t \in A, \forall k\}, \quad (6)$$

$$u(t, x) = \mathcal{L}a(t) \equiv \sum_{k=0}^{\infty} a_k(t) \phi_k(x). \quad (7)$$

In a discrete computation, we can approximate (7) by $K$-term of POD expansion. The representation can be expressed as

$$u_K(t, x) = \sum_{k=1}^{K} a_k(t) \phi_k(x), \quad (8)$$

and the truncated restriction and truncated lifting operators are defined as $\mathcal{R}_K$ and $\mathcal{L}_K$, respectively. The convergence of the $K$-term POD expansion is the form

$$\|u - u_K\| \to K^{-\gamma}, \quad \text{as} \ K \to \infty, \quad (9)$$

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

In general, we can write the evolution of the POD coefficient $a(t)$ by

$$\frac{da}{dt} = g(a(t)), \quad (10)$$

where the explicit form of $g$ remains unknown. Thus, the derivative of POD coefficients must be approximated, rather than explicitly evaluated, to march forward in time. It has been noted that one can find an explicit form of $g$ by projecting the governing PDEs onto the POD modes [6, 14–18]. It is a traditional Galerkin projection approach that is

Figure 5: Transient dynamics: Re = 400, lift coefficient for $T = 0.01$ with various number of $M$. From top to bottom: $M = 3, 4, \text{and} 5$. 
Figure 6: Transient dynamics: $Re = 800$, drag coefficient for $T = 0.01$ with various number of $M$. From top to bottom: $M = 3$, 4, and 5.

different from our present method. We will provide more details of restriction and lifting processes for solving the incompressible flows in the next subsections.

2.3.1. Restriction and Lifting. We employ the snapshot method [4] to extract the set of POD bases, $\{\phi_k(x)\}$, from the ensemble of previously saved solutions. In our study, these solutions or snapshots of flow fields at each time step are obtained from the full DNS in a small finite time interval. We refer to this computation as a fine-scale computation. Then we project this set of snapshots to the POD bases and compute forward in time on the slow manifold space. We call this step a projective step that enables us to perform relatively large time step on the manifold space when comparing with the time step of the fine scale computation. Hence, the projective step is sometimes called a coarse-scale computation.

In the fine-scale time interval, the solutions or snapshots $u(t_i, x)$ at time $t_i$ are obtained by solving numerically the incompressible Navier-Stokes equations through the spectral/hp element method, where $t^n \leq t_i \leq t^n$, $i = 1, \ldots, n_f$, where $t_c$ and $n_f$ denote the final time value and the number of time steps of fine-scale computations, respectively. From (5), the POD bases are then determined discretely by

$$\phi_k(x) = \sum_{i=1}^{n_f} a_k(t_i) u(t_i, x) \, dt, \quad \forall k, \quad (11)$$
where \( \{a_k\} \) are obtained by solving the correlation matrix (3). Once the POD basis functions are determined from (11), we can restrict any solution \( u(t, x) \) for any given \( t \) to obtain the corresponding POD coefficients \( a_k \) by (6). The derivative of the POD coefficients can then be approximated and used to march forward in time by the projective integration technique (coarse-scale). Details are summarized in the next subsection. After the projective step is completed, the lifting procedure is required to return the computations back to a fine-scale resolution. It is the reverse process of the restriction, that is, for a given set of computed POD coefficients at time \( t \), we can reconstruct the corresponding solution by using (7).

2.3.2. Projective Integration. The projective integration procedure is described as follows.

(i) Approximate the RHS of (10) at \( t = t^n \) using

\[
g(t^n) = \sum_{j=0}^{n_e} \alpha_j a(t_j) = \frac{da}{dt}(t^n) + O(\delta t^f),
\]

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

(ii) Once the RHS of the typical reduced-order model (10) is estimated numerically, one can use any standard ODE solver to integrate the numerical solution in time. For

\[...\]
instance, given a coarse time step $\Delta t_c \equiv n_c \delta t$, where $n_c \geq 1$, such that $t^{n+1} = t^n + \Delta t_c = t^n + (n_f + n_c) \delta t$, the single step forward Euler method takes the form

$$a(t^{n+1}) = a(t^n) + \Delta t_c \cdot g(t^n) + O(\Delta t^2). \quad (13)$$

It should be noted that other higher-order explicit integration schemes (and possibly implicit ones) can be used to achieve better accuracy and/or stability properties. For instance, one can use the following scheme:

$$a(t^{n+1}) = a(t^n) + \frac{n_c}{k!} (\Delta t_c)^k \frac{\partial^{(k-1)} g(t^n)}{\partial t^{k-1}} + O(\Delta t^2). \quad (14)$$

The higher order temporal derivatives of $g(t)$ are approximated in a way similar to (12). It can be seen that (14) is a high-order single-step method. In this work, we perform the projective step using forward Euler method. This scheme provides smaller oscillations than using higher order scheme such as (14).

2.4. Equation-Free POD-Assisted Projective Integration. The overall steps of equation-free POD-assisted projective integration are summarized in this section.

Generally, one global time step of this method is composed of two types of time scales which are fine-scale and coarse-scale time steps, respectively. We start calculations by using fine-scale integrator with $n_f$ time steps and then perform coarse-scale integration for $n_c$ steps. The fine-scale integrator can be performed by any accurate methods whereas the coarse-scale integrator is chosen to be the
forward Euler’s method in this study. A single step of the equation-free POD-assisted projective integration method, from $t = t^n$ to $t = t^{n+1}$, consists of the following main steps.

(i) **Fine-scale computation**: we perform this step by the DNS. The DNS provides numerical solutions at each time step $\delta t$, and so at $t^n = t^n + \Delta t_f$, where $\Delta t_f = n_f \delta t$, and $n_f$ is the number of fine-scale time steps.

(ii) **Restriction**: derive the POD coefficients using the previously saved solutions. Here, we use the method of snapshot for deriving POD modes and their corresponding coefficients. We also estimate the time derivatives of each POD mode coefficient at this stage.

(iii) **Projective integration**: time integration using the Euler method is performed on the POD hyperspace with time step $T$. This step is called the coarse-scale computation because we can use a relatively large $T$ when comparing with $\Delta t_f$. Thus the efficiency of the method depends directly on how large coarse-time step $T$ can be.

(iv) **Lifting**: the projected solution at time $t^{n+1} = \Delta t_f + T$ is lifted from POD hyperspace to the physical domain.

This completes the single time step of the equation-free POD-assisted projective integration method. Finally, we repeat the whole processes until the final time is reached.

### 3. Numerical Results

In this section, we investigate the accuracy and efficiency of the projective integration method for solving incompressible
flows. The purpose is to explore the representations of online POD modes. So, we restrict our investigations to two cases of Reynolds number, Re = 400 and 800. For each case, we separate flow pattern into two types: transient and quasi-periodic flows. We have checked the accuracy of our DNS results and found that they are in good agreement with the experimental results reported by Williamson et al. [19]. Thus the DNS results in our simulations can be regarded as the exact solutions which will be used to validate the accuracy of the numerical results obtained from the equation-free POD-assisted projective integration method.

For all simulations, there are two time scales: fine time scale, $\delta t$, and coarse time scale, $T$. The total time step of one projective step is $n_f \delta t + T$, where $n_f$ is the number of fine-scale steps. We have set five snapshots for each projective step, thus the maximum number of POD modes in one projective time step is five as well. This number is enough to capture the dynamics of solutions because flow dynamics change rather slightly in one projective step. It is different from the offline method where the number of POD modes used to describe the entire flow dynamics must be much greater.

We calculate time dependent solutions from $t = 0$ to 30. The unsteady flows can be classified into two cases: transient case when $0 \leq t < 15$, and periodic case with Karman vortex street when $15 \leq t \leq 30$. Note that periodic flow in both cases of Re occurs before $t = 15$, but we classify the flow pattern by this value to make sure that the flow is entirely

---

**Figure 10:** Periodic case: Re = 800. Computed drag coefficient for $T = 0.01$ with various number of $M$. From top to bottom: $M = 3, 4,$ and $5$. 

![Figure 10](image-url)
periodic. Vorticity plots of DNS results when \( t = 13 \) for \( \text{Re} = 400 \) and 800 are shown in Figures 2 and 3, respectively. Efficiency of the numerical scheme for both transient and periodic flows will be shown in the next sections.

### 3.1. Transient Dynamics

In this section, we apply the equation-free POD-assisted projective integration method to numerically solve unsteady flow problems on \( 0 \leq t < 15 \). We set \( \delta t = 0.0005 \). The accuracy and efficiency of the present method can be observed by varying the values of coarse time step and the number of POD modes, \( M \).

Figures 4 and 5 show drag and lift coefficients for \( \text{Re} = 400 \) using various numbers of POD modes in the simulations. It can be seen that flows become periodic when \( t \approx 8 \). The label of AIM (approximated inertial manifold) in these figures refers to the results obtained by the projective integration method. We have set coarse time step as \( T = 0.01 \). This time step is very large (twenty times as much) when comparing with the fine-scale. To extract POD bases, we have used five snapshots from DNS where each snapshot is collected at 0.01 time step. So, we run on fine-scale system 0.05 time steps. The last two snapshots are used in the projective step by the forward Euler method. Thus, we have run on the fine-scale system by the DNS with 0.05 time steps and on the slow manifold (coarse-scale) with 0.01 time steps.

Next, we investigate the appropriate number of POD modes by trial and error. It is found that solutions by using \( M = 3 \) or 4 are in good agreement with the DNS results. However, some phase shifts in both drag and lift coefficients appear for \( M = 5 \). Thus, the appropriate number of POD modes should be 3 or 4. Adding just one more
We have checked for the cases of smaller coarse time step, $T = 0.005$ and $0.007$, are also investigated. Numerical results are in good agreement with the DNS results for $M = 3$ and 4. The profiles of drag and lift coefficients are the same as those for the case of $T = 0.01$. But the results of using $M = 5$ and $T = 0.005$ still produce some phase shifts. This numerical error comes from the POD expansion part. The numerical results are clearly incorrect when we use larger coarse-scale step, $T = 0.02$. The numerical solutions diverge very rapidly for $M = 4$ and 5. When we set $T = 0.015$, the numerical method still provides accurate solutions for $M = 4$. Thus, the highest efficiency for simulating this problem occurs when we set $0.01 < T < 0.015$ and $M = 4$.

Figures 6 and 7 show drag and lift coefficients for $\text{Re} = 800$ with various number of POD modes. In this case, these coefficients are higher than the values in the case of $\text{Re} = 400$. Flows become periodic when $t \approx 6$. We have set coarse time step to $T = 0.01$. Accurate numerical results are obtained by using only $M = 4$. Using not enough POD modes, for example $M = 3$, produces some oscillations. On the other hand, increasing just one more POD mode in solution expansion has affected the phase-shift pattern. We have checked for the cases of smaller $T$. It is found that $M = 4$ provides accurate results. But as we set $T = 0.015$, numerical solution diverges rapidly for any number of POD modes constructed. This shows the effects of truncation error in the forward Euler method that becomes a dominant factor causing the instability of the numerical scheme. For $\text{Re} = 800$, we conclude that the projective integration method has the highest efficiency when we set $0.01 < T < 0.0125$ and $M = 4$.

3.2. Periodic Dynamics. In this section, periodic flows when $15 < t < 30$ are investigated. The main propose is to explore the efficiency of the online Galerkin-free POD-assisted projective integration method in order to capture the periodic pattern of vorticity downstream. For $\text{Re} = 400$, computed drag coefficients when $T = 0.015$ are shown in Figure 8. The projective integration results (AIM) are in good agreement with the DNS results for $M = 4$ and 5. Using not enough POD modes, for example $M = 3$, results directly to the stability of numerical solutions. Similar results of computed lift coefficients are shown in Figure 9. Only using $M = 4$ and 5 provides accurate results. These observations are relatively different from the transient case where increment of $M$ results in some phase-shift patterns. This suggests computations in the case of periodic flow stay within some range of the attractor, in constrast to the case of transient flow.

For $\text{Re} = 800$, computed drag and lift coefficients when $T = 0.01$ are shown in Figures 10 and 11. Only using $M = 5$, provides accurate results for both drag and lift coefficients. The numerical scheme is more stable as $M$ increased. But, solutions diverge rapidly as we increase time step to $T = 0.015$. Increment of $M$ cannot improve the stability of the numerical scheme. Truncation error from coarse-scale time step is now a dominant factor destroying the stability.

4. Interplay of POD Modes

In this section, the representations of each individual POD mode in the online POD-assisted projective integration method are investigated. Because it is an online process, it implies that we must investigate the POD modes during time integration. This approach is different from the traditional reduced-order model by the Galerkin projection. This shows one advantage of the present scheme allowing one to gain more understanding of the dynamical representations for each POD mode.

In our problem, the normalized energy for each POD mode is shown in Figure 12. These normalized values are calculated from the set of eigenvalues extracted from the restriction stage of one projective time step. Log-scale plot shows dynamically the POD energy in time. The first POD mode has the maximum energy of order $O(1)$ whereas the fifth POD mode has the minimum energy of $O(10^{-15})$. The first POD mode captures the most energy in every projective step. Higher POD modes have functions for improving solution accuracy. The energy plot shows clearly that, in energy sense, the combination of the first four POD modes is enough to build up the online model. The energy of the fifth POD mode is very small. Also, this POD mode produces some oscillations in the simulations. It is shown by phase-shift appearance of the drag and lift coefficients in Figures 4 and 5.

In order to understand the representations for each POD mode in time, the interplay among the first five POD modes is investigated. We proceed as follows. We compute solutions using only $M = 1$, for example, only the first POD mode included in the projective integration. Next, we calculate...
solutions using $M = 2$. The difference between the first and second calculations is regarded as the interplay of the first two POD modes. Similar processes can be performed for investigating the interplay of other higher POD modes. We have set $T = 0.005$ in all calculations to minimize numerical errors. Observed results can be summarized as follows.

4.1. The First and Second POD Modes. Drag and lift calculations for the case of projective integration with $M = 1$ and $M = 2$ are shown in Figure 13. Comparing with the DNS results in Figures 4 and 5 shows that using only the first POD mode produces incorrect result. It can be seen that the second POD mode assists the first POD mode by recovering the frequency and amplitude of drag and lift forces. The differences of individual components in the lift and drag coefficients (i.e., pressure and viscous components) are also shown. The different values are relatively large especially at the early stage of the transient case ($0 < t < 5$).

Apart from investigating the overall variables like the lift and drag coefficients, we also investigate the local differences in the primary results obtained from the projective integration method with $M = 1$ and $M = 2$. Plots of time series for $u$, $v$, and $p$ at various probe numbers, #1, #2, and #3 (see Figure 1) are presented in Figure 14. The second POD mode assists the first POD mode by increasing frequency but representing the same order of amplitude similar to that observed in the lift and drag coefficients. The representations of the second POD mode are similar in all of these three probes.

4.2. The Third POD Mode. We now focus on the third POD mode. Comparing the results in Figure 15 with those in Figure 13, we see that inclusion of the third mode brings both phase and amplitude of the lift and drag coefficients in good agreement with DNS results. We see that the third POD mode does provide a significant contribution to the solution in terms of further correcting phase and amplitude. The differences in drag and lift coefficients between $M = 2$ and $M = 3$ are smaller than those values in the cases of $M = 1$ and $M = 2$. The first three POD modes are very important in the online method because they can capture all basic flow behaviors.

4.3. The Fourth and the Fifth POD Modes. We perform the same investigations on higher POD modes. It is found that adding one more POD mode which is the fourth POD mode into the solution expansion of $M = 3$ does not provide any significant improvement. The total drag and lift by applying $M = 3$ and $M = 4$ are shown in Figure 16. Sets 1 and 2 refer to using $M = 3$ and 4, respectively. Because all major characteristics of flow solution are largely captured by the first three POD modes, including too many POD modes in the projective method does not guarantee the improvement in solution accuracy. This can be seen in Figure 17. The inconsistency can be explained by the fact that the approximations of the higher mode derivatives are inaccurate. They exhibit small-amplitude high-frequency oscillatory behavior, which results in the phase-shift pattern seen in the solution when we include the fifth POD mode.

To reveal the representations of each POD mode by the online method, we also show some transient simulations when using various POD modes. Results are presented in Figures 18–21. The vorticity plots in the case of $Re = 400$ at time $t = 1.9, 6.3, 10.7, \text{ and } 12.9$ are shown in Figure 18.
It can be seen that the first POD mode can capture only basic flow behaviors, but its representation is inaccurate. It represents as the mean mode due to its highest contained energy (see Figure 12). Using the first two POD modes is shown in Figure 19. The second POD mode assists the first POD mode by improving the strength and frequency of vorticity field downstream. When comparing with the DNS results in Figure 2, flow patterns are nearly the same. There are some small differences in vorticity magnitude. However, if we increase the number of POD modes to $M = 3$, the results from the online projective method are in good agreement to those results from the DNS (see Figures 20 and 2 at approximately $t = 13$). The third POD mode can correct the magnitude of vorticity field. Hence, using the combination of the first three POD modes is enough in our simulations. The results by the online method when using the first four POD modes are shown in Figure 21. The results are the same as those by the first three POD modes.

The appropriate number of extracted POD modes should be 3 or 4 in our simulations. However, the results by the online method are not accurate when using the combination of the first five POD modes (see Figures 4 and 5). It is due to the fact that the fifth POD mode has very high frequency with even smaller amplitude. Including this high POD mode in the projective method produces some noise. These observations also show the difference of reduced model structure between the online and offline methods. For the online method, the first few POD modes are enough to represent flow behaviors whereas a large number of POD modes is needed in the offline method.

5. Conclusions

The POD-assisted projective integration method based on the Galerkin-free approach is presented in this paper. We apply the approach to simulate the two-dimensional incompressible fluid flows past the NACA0012 airfoil. The present approach is referred to as the online method. POD bases used in solution expansions are extracted during time integration from the data set of direct numerical simulation. The present approach is different from the offline method in which refered POD based are extracted from a priori known
data set in the whole time integration process, not during time integration process. We have applied the online method for simulating the incompressible flows in two cases of Re, 400 and 800, for time $0 < t < 30$. There are two time scales which are fine and coarse scales. Fine ($\delta t$) and coarse ($T$) scales are the time scales for the DNS and the projective integration, respectively. We have set $\delta t = 0.0005$ in all simulations. It is observed that the maximum values of $T$ are 0.015 and 0.0125 for $Re = 400$ and 800, respectively. This shows the efficiency of the present method that can use very large jump in the projective step when comparing with the DNS time step. The number of POD modes used in the projective method is also an important issue. We found that using only the first few POD modes in the online method is enough for representing flow behavior. We have also revealed the representations of each dominant POD mode. The first mode has the highest energy. The basic flow behavior can be represented by using this mode but
Figure 17: Total drag and total lift when using $M = 4$ and $M = 5$.

Figure 18: Vorticity plots of $Re = 400$ when $M = 1$ for $t = 1.9$ (a), 6.3 (b), 10.7 (c), and 12.9 (d).

Figure 19: Vorticity plots of $Re = 400$ when $M = 2$ for $t = 1.9$ (a), 6.3 (b), 10.7 (c), and 12.9 (d).
the overall results are inaccurate. Including the second and third POD modes can improve solution accuracy because the higher POD modes contain high energy and degree of fluctuations. The derivative approximations for temporal mode are less accurate as well. Thus, including higher POD mode can produce some incorrect results in simulations. Although, there are many cases of Re that have not been investigated. Hopefully our observations may provide some insights of how to choose discretized parameters and the number of POD modes employed in other simulations.

Extensions of this work would be to study and compare the efficiency of the online and offline approaches, as well as the relationship of POD modes. Our preliminary studies show that the first POD mode in the online method is almost the same as the first POD mode in the offline method. However, it is not the case for the second POD mode in the online method. It is represented by a combination of higher offline POD modes.

Acknowledgment

The authors would like to thank Professor A. L. Pardhanani at the Earlham College for some useful comments and proof-reading throughout this paper.

References


[8] Y. Hoarau, M. Braza, Y. Ventikos, and D. Faghani, “First stages of the transition to turbulence and control in the


