INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2006; **50**:531–554
Published online 7 September 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1053

Reduced-order controllers for control of flow past an airfoil

S. S. Ravindran*,†

Department of Mathematical Sciences, The University of Alabama in Huntsville, Huntsville, AL 35899, U.S.A.

SUMMARY

Reduced-order controller design by means of reduced-order model for control of a wake flow is presented. Reduced-order model is derived by combining the Galerkin projection with proper orthogonal decomposition (POD) or with other related reduced-order approaches such as singular value decomposition or reduced-basis method. In the present investigation, we discuss the applicability of the reduced-order approaches for fast computation of the optimal control for control of vortex shedding behind a thin airfoil through unsteady blowing on the airfoil surface. Accuracy of the reduced-order model is quantified by comparing flow fields obtained from the reduced-order models with those from the full-order simulations under the same free-stream conditions. A control of vortex shedding is demonstrated for Reynolds number 100. It is found that downstream directed blowing on the upper surface of the airfoil near the leading edge is more efficient in mitigating flow separation and suppressing the vortex shedding. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: POD; nonlinear model reduction; optimal control; vortex shedding

1. INTRODUCTION

Design and control of fluid dynamical systems require fast computational fluid dynamic solvers. Reduced-order modelling provides a way to dramatically increase the speed of flow solvers by reducing the number of degrees of freedom. In this paper we will investigate reduced-order controller design by means of reduced-order model for control of a wake flow. Control methods for fluid dynamics have attracted substantial interest in recent years due to their applications in aero/hydrodynamics, combustion and MHD; see References [1–9] and references therein. An interesting aspect of recent developments in computational methods for control of fluids is the design of reduced-order controllers. The design of reduced-order controllers for fluid system is essential for real-time implementation. A key step in the design is the low-dimensional description of the flow model. Although there are linear model reduction approaches such as balanced truncation [10], Hankel norm approximation [11], Lanczos [12]

*Correspondence to: S. S. Ravindran, Department of Mathematical Sciences, The University of Alabama in Huntsville, Huntsville, AL 35899, U.S.A.
†E-mail: ravindra@ultra.uah.edu

Received 16 August 2004
Revised 2 June 2005
Accepted 3 June 2005

procedure and the Arnoldi [13] procedure, they are not suitable for large-scale nonlinear fluid flow systems.

For large-scale nonlinear systems such as the one discussed here, the most popular reduced-order modelling approach is the proper orthogonal decomposition (POD). The POD is a model reduction technique in which one may systematically extract the most energetic modes from a set of realizations from the flow model. Using these modes as basis functions in a Galerkin projection one can reduce the Navier–Stokes model in a complex geometry to a small finite-dimensional system by retaining only a small number of POD modes. The POD has been extensively discussed in the literature of the past decade as a model reduction tool for simulation [14–18] and control [17, 19–21]. Unlike the traditional Galerkin methods which uses piecewise polynomials or other special functions that have very little connection to the characteristics of the model, the reduced-order modelling techniques discussed here use global basis functions which not only reflects the characteristics of the model and complexity of the geometry well but also reduces the Navier–Stokes model to a low-dimensional model.

The viscous flow around airfoil has been studied extensively due to its simple geometry and its prototypical behaviour of bluff body wake flows. Control of flow past airfoil has applications in lift enhancement and, noise and vibration control. Experimental and computational investigations of active control of flow past airfoils at high angles of attack is an active area of research as extending the usable angles of attack has many important applications. There is also an extensive literature showing the effectiveness of flow control for airfoils. For example, in Reference [22] leading edge suction is investigated for transition delay, in Reference [23] jet flaps were employed for lift increase and in Reference [24] surface suction/blowing was used to rapidly change lift and drag on rotary wing aircraft. In Reference [25] experimental results related to the moving surface boundary layer control for airfoils are presented. However, the control techniques considered in these works and others in the past are *ad hoc* or experimental in nature. This paper presents the optimal control technique to resolve the problem of controlling the vortex shedding behind a thin airfoil using unsteady blowing control. The reduced-order modelling is used for fast computation of the optimal control. It is shown that the nature of the vortex shedding process is significantly altered by unsteady blowing control near the leading edge of the airfoil.

The paper is organized as follows. Section 2 presents the flow model for the flow past airfoil and the methodology employed to obtain the detailed numerical solution databases. Section 3 then briefly describes the POD and other related approaches, and the construction of the reduced-order model. Section 4 then presents the computational results for the numerical simulation of the flow past airfoil using the finite element method and compare them with that of the reduced-order models. In Section 5, we formulate an optimal control problem for the control of vortex shedding for the flow past the thin airfoil. In Section 6, we present computational results for control showing the effectiveness of the present approaches. Section 7 concludes the paper.

2. THE GOVERNING EQUATIONS

Let Ω denote a two-dimensional domain occupied by an incompressible viscous fluid. Let $\mathbf{u}(\mathbf{x},t)$ and $p(\mathbf{x},t)$ denote the velocity and pressure fields, respectively, and $\mathbf{u}_0$ the given velocity. Moreover, let $\mathbf{b}$ denote a specified boundary velocity. The Navier–Stokes equations

are then given by

$$\begin{aligned} \frac{\partial \mathbf{u}}{\partial t} - \frac{1}{Re}\nabla^2\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u} + \nabla p = 0 \quad & \text{in } \Omega\times(0,T] \\ \nabla\cdot\mathbf{u} = 0 \quad & \text{in } \Omega\times(0,T] \\ \mathbf{u} = \mathbf{b} \quad & \text{on } \Gamma_D\times(0,T] \\ -p\mathbf{n} + \frac{1}{Re}\frac{\partial \mathbf{u}}{\partial \mathbf{n}} = 0 \quad & \text{on } \Gamma_N\times(0,T] \\ \mathbf{u}(\mathbf{x},0) = \mathbf{u}_0(\mathbf{x}) \quad & \text{in } \Omega \end{aligned} \tag{1}$$

These equations are nondimensional and the only nondimensional parameter is the Reynolds number defined as $Re = U_0 c/\nu$, where ν is the kinematic viscosity and c is the characteristic length.

2.1. Finite element spatial discretization

We use a weak formulation and finite element method to approximate governing equations (1), but other methods can also be used in the reduced-order modelling context.

A weak formulation of problem (1) is the following: Find $\mathbf{u}\in L^2(0,T;\mathbf{V}_b)$ and $p\in L^2(0,T;L_0^2(\Omega))$ such that

$$\begin{aligned} \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}, \mathbf{v}\right) + \frac{1}{Re}(\nabla\mathbf{u},\nabla\mathbf{v}) - (p,\nabla\cdot\mathbf{v}) = 0 \quad & \forall \mathbf{v}\in\mathbf{H}_0^1(\Omega) \\ (\nabla\cdot\mathbf{u}, q) = 0 \quad & \forall q\in L_0^2(\Omega) \\ \mathbf{u}(\mathbf{x},0) = \mathbf{u}_0(\mathbf{x}) \quad & \text{for } \mathbf{x}\in\Omega \end{aligned} \tag{2}$$

where $\mathbf{V}_b = \{\mathbf{u}\in\mathbf{H}^1(\Omega) : \mathbf{u} = \mathbf{b} \text{ on } \Gamma_D,\ \mathbf{b}\in\mathbf{H}^{1/2}(\partial\Omega)\}$ and $\mathbf{H}_0^1 = \{\mathbf{u}\in\mathbf{H}^1(\Omega) : \mathbf{u} = \mathbf{0} \text{ on } \Gamma_D\}$ and $(\cdot,\cdot)$ denotes the L^2-inner-product.

A typical finite element approximation of (2) is to seek solutions $\mathbf{u}_h(\cdot,t)\in\mathbf{V}_h^b\subset\mathbf{V}_b$ and $p_h\in S_h^0\subset L_0^2(\Omega)$ such that

$$\begin{aligned} \left(\frac{\partial \mathbf{u}_h}{\partial t} + \mathbf{u}_h\cdot\nabla\mathbf{u}_h, \mathbf{v}_h\right) + \frac{1}{Re}(\nabla\mathbf{u}_h,\nabla\mathbf{v}_h) - (p_h,\nabla\cdot\mathbf{v}_h) = 0 \quad & \forall \mathbf{v}^h\in\mathbf{V}_h^0 \\ (\nabla\cdot\mathbf{u}_h, q_h) = 0 \quad & \forall q_h\in S_h^0 \\ \mathbf{u}_h(\mathbf{x},0) = \mathbf{u}_{0h}\in\mathbf{V}_h^0 & \end{aligned} \tag{3}$$

where $\mathbf{V}_h^0\subset\mathbf{H}_0^1(\Omega)$ and $S_h^0\subset L_0^2(\Omega)$.

2.2. Time discretization and solution techniques

For the time discretization of the Navier–Stokes equations we have implemented Crank–Nicholson scheme and a second-order hybrid explicit/implicit scheme. Let M be the number of time steps and $\Delta t = T/M$. Also, let $\mathbf{u}_h^k = \mathbf{u}_h(k\Delta t)$ and $p_h^k = p_h(k\Delta t)$.

2.2.1. Crank–Nicholson scheme. We introduce the second-order Crank–Nicholson time discretization as follows: Seek $\mathbf{u}_h^k \in \mathbf{V}_h^b$ and $p_h^k \in S_h^0$ such that
$\mathbf{u}_h^0 = \mathbf{u}_{0h}$
for $k = 1, \ldots, M$

$$\begin{aligned} \left(\frac{\mathbf{u}_h^k - \mathbf{u}_h^{k-1}}{\Delta t}, \mathbf{v}_h\right) + \frac{1}{2}(\mathbf{u}_h^k \cdot \nabla \mathbf{u}_h^k + \mathbf{u}_h^{k-1} \cdot \nabla \mathbf{u}_h^{k-1}, \mathbf{v}_h) - (p_h^k, \nabla \cdot \mathbf{v}_h) \\ + \frac{1}{2Re}(\nabla(\mathbf{u}_h^{k-1} + \mathbf{u}_h^k), \nabla \mathbf{v}_h) = 0 \quad \forall \mathbf{v}^h \in \mathbf{V}_h^0 \\ (\nabla \cdot \mathbf{u}_h^k, q_h) = 0 \quad \forall q_h \in S_h^0 \end{aligned} \tag{4}$$

If second-order accuracy in time is to be achieved, we need to iteratively solve a nonlinear algebraic system at each time step in (4). Our approach to this is to treat the nonlinear convective term by linear extrapolation in time, that is we approximate $\mathbf{u}_h^k \cdot \nabla \mathbf{u}_h^k$ by $(2\mathbf{u}_h^{k-1} - \mathbf{u}_h^{k-2}) \cdot \nabla \mathbf{u}_h^k$. The corresponding time discretization is second order in time. Alternatively one can treat the convective term explicitly using Adams–Bashforth method and obtain a noniterative scheme which is also second-order accurate in time.

2.2.2. Hybrid Adams scheme. In the hybrid Adams scheme, the convective term is treated explicitly using Adams–Bashforth method while the diffusion term is treated implicitly using an Adams–Moulton method as follows. Seek $\mathbf{u}_h^k \in \mathbf{V}_h^b$ and $p_h^k \in S_h^0$ such that
$\mathbf{u}_h^0 = \mathbf{u}_{0h}$
for $k = 1, \ldots, M$

$$\begin{aligned} \left(\frac{\mathbf{u}_h^k - \mathbf{u}_h^{k-1}}{\Delta t}, \mathbf{v}_h\right) + \left(\frac{3}{2}\mathbf{u}_h^{k-1} \cdot \nabla \mathbf{u}_h^{k-1} - \frac{1}{2}\mathbf{u}_h^{k-2} \cdot \nabla \mathbf{u}_h^{k-2}, \mathbf{v}_h\right) \\ -(p_h^k, \nabla \cdot \mathbf{v}_h) + \frac{1}{2Re}(\nabla(\mathbf{u}_h^{k-1} + \mathbf{u}_h^k), \nabla \mathbf{v}_h) = 0 \quad \forall \mathbf{v}^h \in \mathbf{V}_h^0 \\ (\nabla \cdot \mathbf{u}_h^k, q_h) = 0 \quad \forall q_h \in S_h^0 \end{aligned} \tag{5}$$

Because the convective terms are evaluated at previous time steps, no iteration is required to achieve the second-order accuracy of the scheme. The hybrid Adams method requires special starting values and we found that starting with a lower order method did not reduce the long-term accuracy of the scheme. However, the best results were achieved with a Crank–Nicholson solution for the first step.

3. POD, SVD AND REDUCED-ORDER MODEL

3.1. Proper orthogonal decomposition

The POD is a procedure for extracting, from a given set of data $\{\mathbf{x}_i\}_{i=1}^N$ of vectors in $\mathbb{R}^n$, an optimal basis (a smaller set of vectors) also belonging to $\mathbb{R}^n$ [26, 27]. Applications of this procedure are extensive in turbulence modelling [16, 14] and image processing [28], and it is now emerging as a tool in the field of control of fluids; see for e.g. References [17, 19, 29].

Let the scalar functions $u(\mathbf{x}, t_i)$, $i = 1, \ldots, N$, be a sequence of numerical or experimental observations (data set) where the parameter t_i is time. The time average of the sequence, defined by

$$u_m(\mathbf{x}) = \langle u(\mathbf{x}, t_i) \rangle = \frac{1}{N} \sum_{i=1}^{N} u(\mathbf{x}, t_i)$$

is, without loss of generality, assumed to be zero. The POD extracts orthonormal basis functions $\Phi_l(\mathbf{x})$, such that the reconstruction

$$u(\mathbf{x}, t_i) = \sum_{l=1}^{N} \alpha_l(t_i) \Phi_l(\mathbf{x}), \quad i = 1, \ldots, N$$

is optimal in the sense that the average least squares truncation error

$$e_i = \left\langle \left\| u(\mathbf{x}, t_i) - \sum_{l=1}^{N} \alpha_l(t_i) \Phi_l(\mathbf{x}) \right\|^2 \right\rangle \tag{6}$$

is a minimum for any given number $i \leqslant N$ of basis functions over all possible sets of orthogonal functions. Here $\| \cdot \|$ is the L^2 norm. The functions $\Phi_l(\mathbf{x})$ are called POD modes. Optimality property (6) is equivalent to finding functions Φ that maximizes the normalized average projection of $u(\mathbf{x}, t_i)$ onto Φ

$$\max_{\Phi \in L^2} \frac{\langle |(u(\mathbf{x}, t_i), \Phi)|^2 \rangle}{\|\Phi\|^2}$$

The maximization problem can be reduced to the following integral eigenvalue problem:

$$\int_{\Omega} K(\mathbf{x}, \mathbf{x}') \Phi(\mathbf{x}') \, \mathrm{d}x' = \lambda \Phi(\mathbf{x}) \tag{7}$$

where

$$K(\mathbf{x}, \mathbf{x}') = \frac{1}{N} \sum_{i=1}^{N} u(\mathbf{x}, t_i) u(\mathbf{x}', t_i)$$

But in practice the observations that form the data are only available at discrete spatial grid points and thus they are vectors of the form $\mathbf{u}_i = [u(x_1, t_i), \ldots, u(x_n, t_i)]^{\mathrm{T}}$. In this discrete case, the kernal $K(\mathbf{x}, \mathbf{x}')$ is replaced with

$$G = \begin{bmatrix} K(x_1, x_1) \cdots & \cdots K(x_1, x_n) \\ \vdots & \vdots \\ K(x_n, x_1) \cdots & \cdots K(x_n, x_n) \end{bmatrix}$$

where

$$G_{i,j} = K(x_i, x_j) = \frac{1}{N} \sum_{k=1}^{N} u(x_i, t_k) u(x_j, t_k), \quad i, j = 1, \ldots, n$$

and the eigenvalue problem (7) reduces to finding the eigensolution of the matrix G. There are at least two ways to solve this eigenvalue problem, the direct method and the method of

snapshots. The direct method attempts to solve the eigenvalue problem involving the $n \times n$ matrix G directly using the standard numerical techniques. This can be computationally intensive if the number of grid-points n is larger than the number of observations N and will not be discussed here.

3.1.1. The method of snapshots. The method of snapshots [15] is based on the fact that the data vectors $\mathbf{u}_i$ and the POD modes Φ_l span the same linear space. Thus the POD mode is assumed to be a linear combination of the data vectors

$$\Phi_l = \sum_{i=1}^{N} w_i^l \mathbf{u}_i, \quad l = 1, \ldots, N \tag{8}$$

After substituting (8) into the eigenvalue problem

$$G\Phi = \lambda\Phi$$

the coefficients w_i^l are computed by solving the eigenvalue problem

$$C\mathbf{w} = \lambda\mathbf{w}$$

where

$$C_{ij} = \frac{1}{N}(\mathbf{u}_i, \mathbf{u}_j)$$

The correlation matrix C has dimension $N \times N$ and often the number of snapshots N is much smaller than the number of grid-points n. Therefore the corresponding eigenvalue problem can be computed inexpensively when $N \ll n$. The solution of the eigenvalue problem associated with C yields N eigenfunctions $\{\Phi_l\}_{l=1}^N$ and eigenvalues $\{\lambda_l\}_{l=1}^N$, where $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_N$. The eigenfunctions Φ_l have the property that they form a complete orthonormal set when properly normalized. The eigenvalues λ_l satisfy

$$\lambda_l = \frac{1}{N} \sum_{i=1}^{N} (\Phi_l, \mathbf{u}_i)$$

and if we define the average kinetic energy as

$$E = \frac{1}{N} \sum_{i=1}^{N} (\mathbf{u}_i, \mathbf{u}_i)$$

we can use the fact that $\mathbf{u}_j = \sum_{k=1}^{N} (\mathbf{u}_j, \Phi_k)\Phi_k$ to show $E = \sum_{i=1}^{N} \lambda_i$. The velocity at any instant can be expanded in terms of these eigenvectors as $\mathbf{u}(\mathbf{x}, t) = \sum_{i=1}^{N} \alpha_i(t)\Phi_i(\mathbf{x})$. If however one computes the m dominant eigenfunctions using the percentage energy captured criterion, $E_m = 100(\sum_{i=1}^{m} \lambda_i)/(\sum_{i=1}^{N} \lambda_i)$ and uses them in the expansion, one obtains a reduced-order solution

$$\mathbf{u}(\mathbf{x}, t) \approx \sum_{i=1}^{m} \alpha_i(t)\Phi_i(\mathbf{x})$$

In most situations, the number of dominant eigenvalues m is much less than N, and hence, not all eigenfunctions need to be computed. To save computational time, we compute only

the first m dominant eigenvalues using the DEVESF subroutine from IMSL [30]. Routine DEVESF first reduces the matrix to an equivalent symmetric tridiagonal matrix and then uses the rational QR algorithm with Newton corrections to compute the extreme eigenvalues of the tridiagonal matrix. The eigenvectors of the original matrix are computed by first computing the eigenvectors of the tridiagonal matrix. This routine is based on EISPACK [31].

3.2. Singular-value decomposition

For any real $m \times n$ matrix A, there exists a real factorization

$$A = U\Sigma V^{\mathrm{T}} \tag{9}$$

where U is an $m \times m$ orthogonal matrix whose columns form the left singular vectors. Σ is an $m \times n$ pseudo-diagonal and semi-positive definite matrix with diagonal entries containing the singular values in decreasing order, i.e. $\sigma_1 \geqslant \sigma_2 \geqslant \cdots \geqslant \sigma_n$. V is an $n \times n$ orthogonal matrix and its columns form the right singular vectors.

One way to compute the SVD of A is to simply calculate the eigenvalues and eigenvectors of $A^{\mathrm{T}}A$ and AA^{T}. Indeed,

$$AA^{\mathrm{T}} = U\Sigma^2 U^{\mathrm{T}}, \quad A^{\mathrm{T}}A = V\Sigma^2 V^{\mathrm{T}}$$

and the singular values of A are found to be the square roots of the eigenvalues of AA^{T} or $A^{\mathrm{T}}A$. The left and right singular vectors of A are the eigenvectors of AA^{T} and $A^{\mathrm{T}}A$, respectively. Consider now the $(N \times n)$ snapshot matrix

$$A = \begin{bmatrix} u_1(x_1) \cdots & \cdots u_1(x_n) \\ \vdots & \vdots \\ u_N(x_1) \cdots & \cdots u_N(x_n) \end{bmatrix}$$

It is clear that $(1/N)A^{\mathrm{T}}A = G$ and the corresponding eigenvalues are the proper orthogonal eigenvalues. Therefore, POD can be carried out by computing the SVD of the matrix A. To see this, let $D = U\Sigma$, then we can write the SVD of A as

$$A = \sum_{l=1}^{n} \mathbf{d}_l \mathbf{v}_l^{\mathrm{T}} \tag{10}$$

where $\mathbf{d}_l$ and $\mathbf{v}_l$ are the lth columns of the matrices D and V, respectively. By (10), one can set desired energy percentage E_m, determine the required number of modes m and identify the corresponding columns of D and V to obtain a reduced-order solution

$$\mathbf{u}(\mathbf{x}, t) \approx \sum_{i=1}^{m} \alpha_i(t)\Phi_i(\mathbf{x})$$

where $\alpha_i(t)$ is a function of time, whose value at time $t = k\Delta t$ is equal to the kth entry of $\mathbf{d}_l$, where $k = 1, \ldots, N$. Also, $\Phi_i(\mathbf{x})$ is a function of $\mathbf{x}$, and it corresponds to the entries seen in $\mathbf{v}_l^{\mathrm{T}}$ at every spatial grid point $x = j\Delta x$, where $j = 1, \ldots, n$.

Finally, the low-dimensional reduced base is defined as

$$\mathbf{V}^R \equiv \mathrm{span}\{\Phi_1, \Phi_2, \ldots, \Phi_m\}$$

3.3. Reduced-order model for state simulation

For state simulation, the reduced-order model construction proceeds as follows. First one chooses a reduced-basis set consisting of the modes $\Phi_i,\ i=1,\ldots,m$, where m is usually very small compared to the number of functions used in finite element approximations or the number of grid points used in a finite difference approximation. Next, one seeks approximation to the state as

$$\mathbf{u}(\mathbf{x},t)=\mathbf{u}_m(\mathbf{x})+\sum_{i=1}^{m}\alpha_i(t)\Phi_i(\mathbf{x}) \tag{11}$$

where $\mathbf{u}_m$ is the time average of the snapshots. To determine the coefficients $\alpha_i(t),\ i=1,\ldots,n$ we employ the Galerkin projection. The Navier–Stokes model residual $\mathbf{R}$ can be expressed as

$$\mathbf{R}(\mathbf{u},p)=\frac{\partial\mathbf{u}}{\partial t}-\frac{1}{Re}\nabla^2\mathbf{u}+\mathbf{u}\cdot\nabla\mathbf{u}+\nabla p$$

Applying the Galerkin projection which enforces the residual to be orthogonal to each of the basis functions, $(\mathbf{R},\boldsymbol{\Phi}_i)=0,\ i=1,\ldots,m$, and integrating by parts on the Laplacian term yields

$$\left(\frac{\partial\mathbf{u}}{\partial t}+\mathbf{u}\cdot\nabla\mathbf{u},\Phi_i\right)-(p,\nabla\cdot\Phi_i)+\frac{1}{Re}(\nabla\mathbf{u},\nabla\Phi_i)+\left(p\mathbf{n}-\frac{1}{Re}\frac{\partial\mathbf{u}}{\partial\mathbf{n}},\Phi_i\right)_{\Gamma_N}=0 \tag{12}$$

for all $\Phi_i\in\mathbf{V}^R$. Notice that basis elements Φ_i are divergence free as flow is incompressible and satisfy zero boundary conditions on Γ_D. Using these properties of $\boldsymbol{\Phi}_i$ and the boundary condition on Γ_N, we see that the pressure term and the boundary term vanish in (12). Then (12) reduces to

$$\left(\frac{\partial\mathbf{u}}{\partial t}+\mathbf{u}\cdot\nabla\mathbf{u},\Phi_i\right)+\frac{1}{Re}(\nabla\mathbf{u},\nabla\Phi_i)=0 \tag{13}$$

for all $\Phi_i\in\mathbf{V}^R$. On substitution of (11) into (13), we obtain the following nonlinear evolution equation for the coefficients $\boldsymbol{\alpha}(t)$:

$$\begin{aligned}\frac{\mathrm{d}\boldsymbol{\alpha}}{\mathrm{d}t}&=\mathscr{A}\boldsymbol{\alpha}+\boldsymbol{\alpha}^{\mathrm{T}}\mathscr{N}\boldsymbol{\alpha}+\mathbf{e}\\ \boldsymbol{\alpha}(0)&=\boldsymbol{\alpha}_0\end{aligned} \tag{14}$$

where

$$\begin{gathered}\mathscr{A}_{ij}=-(\Phi_j\cdot\nabla\mathbf{u}_m,\Phi_i)-(\mathbf{u}_m\cdot\nabla\Phi_j,\Phi_i)-\frac{1}{Re}(\nabla\Phi_j,\nabla\Phi_i),\quad i,j=1,\ldots,m\\ \mathscr{N}_{ikl}=-(\Phi_k\cdot\nabla\Phi_l,\Phi_i),\quad i,k,l=1,\ldots,m\\ \alpha_{0i}=(\mathbf{u}_0,\Phi_i),\quad e_i=-(\mathbf{u}_m\cdot\nabla\mathbf{u}_m,\boldsymbol{\Phi}_i)-\frac{1}{Re}(\nabla\mathbf{u}_m,\nabla\boldsymbol{\Phi}_i),\quad i=1,\ldots,m\end{gathered}$$

System (14) is the required reduced-order model of the uncontrolled Navier–Stokes model (1). The computational cost of the reduced-order model would be small if m is small, ignoring the cost of the off-line determination of the basis functions $\boldsymbol{\Phi}_i,\ i=1,\ldots,m$.

It is now clear that both POD and SVD require the generation of a snapshot set (data set). A POD or SVD base can be no better than the information contained in the snapshot set on which it is based. Therefore a necessary condition for producing effective low-dimensional reduced bases is the generation of good snapshot sets. A good snapshot set should have all the information needed to accurately carry out further simulations of the state or to effect control using the associated reduced-order models. However, the state of a system is determined by parameters in the model such as initial and boundary conditions, sources, coefficients and geometry. The state also depends on the independent variables. Therefore a snapshot set can be a steady-state solutions corresponding to several sets of design parameters [6], an unsteady state solutions for a fixed set of design parameter values evaluated at several instants in time [17, 19] or several unsteady state solutions corresponding to different sets of parameter values evaluated at several instants in time.

4. COMPUTATIONAL RESULTS FOR SIMULATION

In this section, we present the computational results for the simulation of flow around an inclined airfoil. We also compare and contrast the simulations of finite element and reduced-order models.

The geometric configuration of the flow around an inclined airfoil is shown in Figure 1. The entire flow domain extends to six-times the airfoil length c $(=1)$ in the horizontal and vertical directions. The airfoil is placed inside the domain such that the leading edge of the airfoil is at $(x, y)=(1, 1)$ and inclined at an angle of 45° to the inflow direction. A Reynolds number of $Re=100$ was chosen. No slip boundary conditions were applied on the surface (Γ_s) of the airfoil; an inflow boundary condition was applied at the left and bottom boundaries (Γ_i):

$$u=u_{\text{in}}=\cos(\pi/4) \quad \text{and} \quad v=v_{\text{in}}=\sin(\pi/4)$$

and an outflow boundary condition was applied at the top and right boundaries (Γ_o):

$$-p\mathbf{n}+\frac{1}{Re}\frac{\partial \mathbf{u}}{\partial \mathbf{n}}=0$$

All the computations were done with a 127×127 spatial grid and a time step size $\Delta t=0.025$. The spatial grid points are clustered in the vicinity of the airfoil and towards the leading and trailing edges; see Figure 2. In order to assess the convergence of the numerical solutions, 127×127 grid solution was compared with that obtained on a 354×354 grid and the variation was found to be less than 1%. In addition a coarse 64×64 grid solution was used to compute the Richardson interpolation and compared with 354×354 grid solution. The variation was again found to be less than 1%. The 127×127 spatial grid is therefore considered to provide a sufficiently accurate resolution for this flow. Both the time discretization schemes of Section 2.2 showed second-order accuracy in time and found to be comparable in efficiency. But the Crank–Nicholson scheme when implemented in iterative mode without using the extrapolation in time was inefficient. Typical computations require 20 CPU minutes per time step on a single processor SUN Ultra 60 workstation, while the time integration proceeds for approximately 1000 time steps.

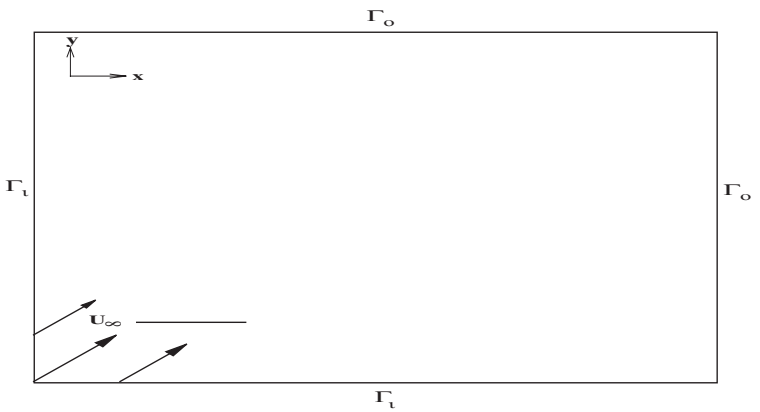

Figure 1. Sketch of the flow configuration considered.

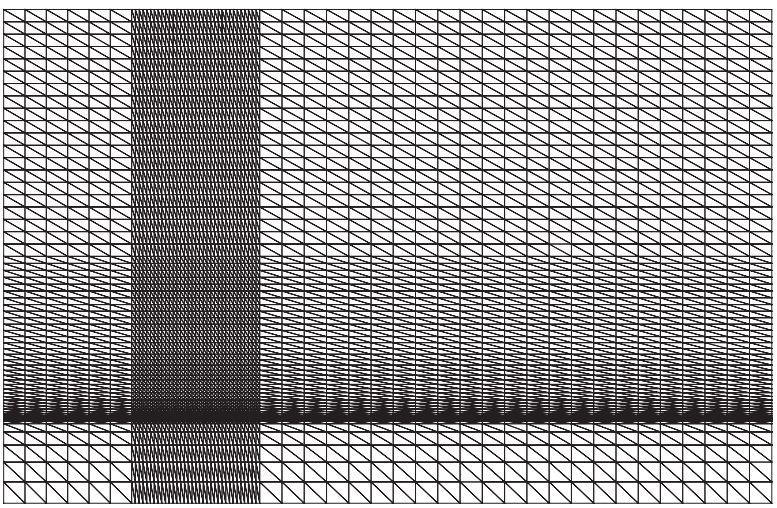

Figure 2. Computational grid for finite element simulation.

For this set of parameters and without boundary control, the flow pattern displays the expected evolution as the Reynolds number increases. For highly viscous fluids, the flow practically remains attached. At low viscosities, the flow becomes unsteady, and typical Karman vortex streets emerge. Figure 3 shows the evolution of the flow for $Re = 100$. Soon after the impulsive start, both the front and rear stagnation points begins to move. The movement of the rear stagnation point is towards the trailing edge whereas that of the front stagnation point is towards the leading edge. After initial separation, the lift increases owing to enlargement of the clockwise separation bubble. This bubble extends beyond the trailing edge and breaks up. The stalling characteristic of the airfoil begins with the break up of the clockwise bubble. Afterwards, new counterclockwise bubbles are formed near the upper surface of the

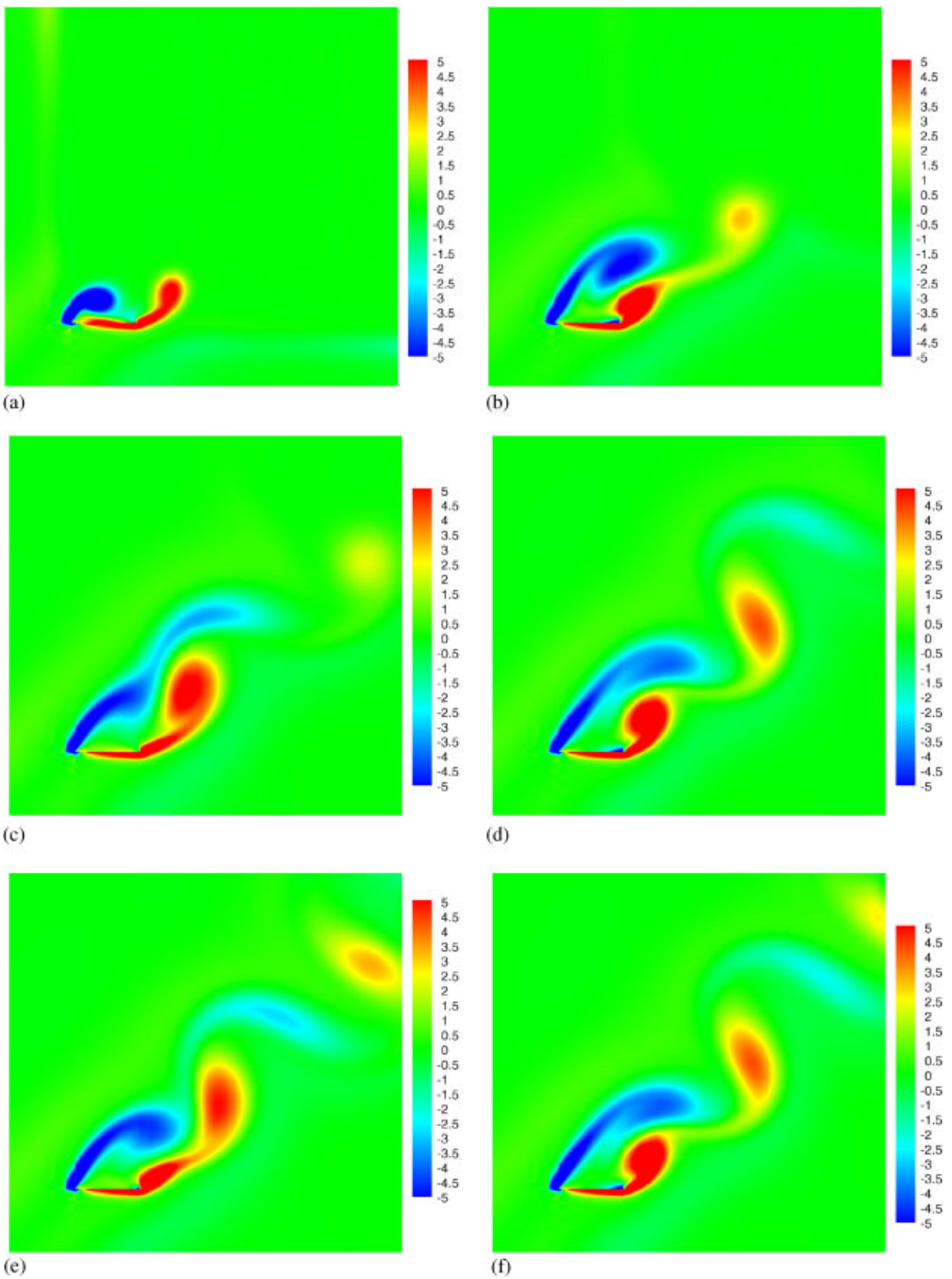

Figure 3. Snapshots of vorticity at various time instants: (a) $t = 2.5$; (b) $t = 7.5$; (c) $t = 12.5$; (d) $t = 17.5$; (e) $t = 22.5$; and (f) $t = 25$.

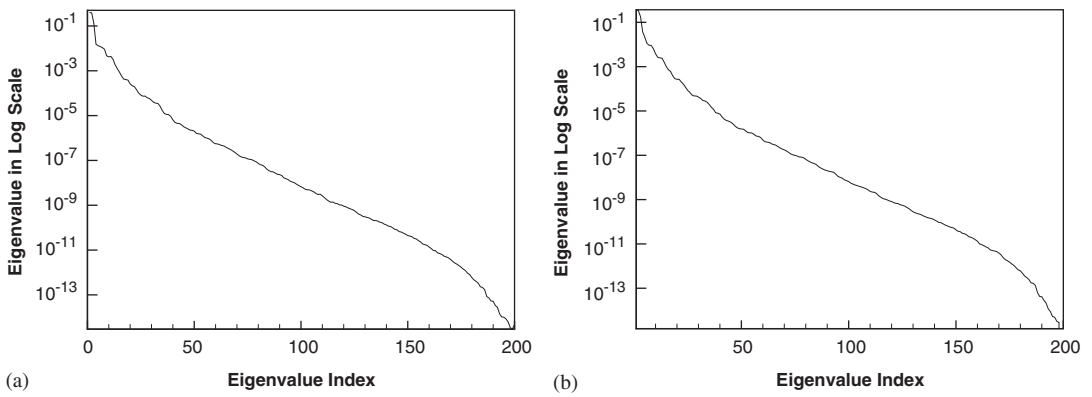

Figure 4. (a) Normalized eigenvalues of the correlation matrix C. (b) Normalized eigenvalues of the matrix G.

airfoil. An increase in the strength of a clockwise (or counterclockwise) bubble is associated with lower (or higher) pressures along the airfoil surface of contact. The lift increases when attached clockwise bubble grow and when counterclockwise bubbles are shed. It decreases when counterclockwise bubbles grow and clockwise bubbles are shed.

In the rest of this section, we focus on the reduced-order model construction and compare its results with finite element model results. For the reduced-order model construction for uncontrolled flow, the snapshots of the total flow field are obtained by solving the Navier–Stokes equations (1) with the above boundary conditions and zero initial condition. The transient flow fields are recorded at constant time interval Δt^* ($\Delta t^* = 5\Delta t$, $\Delta t = 0.025$) to obtain the snapshot set. A total of 200 snapshots are recorded and these snapshots are then made a set of homogeneous velocity fields by subtracting the mean velocity field $\mathbf{u}_m$ from each of these snapshots of the flow field. The correlation matrix C was formed with the aid of the finite element scheme used to solve the Navier–Stokes model and the eigenvalue problem associated with the matrix C was solved using the IMSL subroutine DEVESF [30] for the first m dominant eigenvalues. The eigenvalue spectrum from the correlation matrix is shown in Figure 4(a). For the reduced-order model based on SVD, the singular values of the snapshot matrix were computed using the LAPACK subroutine DGESVD [32] which implements in double precision the SVD via bidiagonalization followed by computing the SVD of the bidiagonal matrix [33].

The eigenvalues of the matrix G are shown in Figure 4(b). Table I lists the normalized eigenvalues of the matrices C and G, and their cumulative contribution to the energy. The convergence of the modes for both POD and SVD is good with the first 12 modes representing approximately 99% of the total energy albeit the convergence for SVD is slightly better. Also the POD convergence is slower than that for the separated flow in a channel reported in Reference [19]. The POD eigenvalue and SVD singular-value computations were also performed with different numbers of snapshots and it was found that the magnitudes of the first few modes were not changed significantly. For numerically solving the reduced-order model, Crank–Nicholson method was used with the time step $\Delta t = 0.025$ and the resulting nonlinear algebraic system was solved using the Newton iterative method.

Table I. Normalized eigenvalues of the 12 most energetic modes and their cumulative contribution to the energy.

	POD		SVD	
	Eigenvalues	Total energy	Eigenvalues	Total energy
Index	$\overline{\lambda_i^C} = \frac{\lambda_i^C}{\sum_{i=1}^N \lambda_i^C}$	$\sum_{i=1}^{n_i} \overline{\lambda_i^C}\%$	$\overline{\lambda_i^G} = \frac{\lambda_i^G}{\sum_{i=1}^N \lambda_i^G}$	$\sum_{i=1}^{n_i} \overline{\lambda_i^G}\%$
1	0.402244	40.22	0.369202	36.92
2	0.383402	78.56	0.337569	70.67
3	0.130067	91.57	0.180963	88.77
4	0.015489	93.12	0.037513	92.52
5	0.013117	94.43	0.021036	94.62
6	0.012087	95.64	0.011333	95.76
7	0.010793	96.72	0.009263	96.68
8	0.009085	97.62	0.009057	97.59
9	0.004641	98.09	0.006415	98.23
10	0.004217	98.51	0.003837	98.61
11	0.004362	98.94	0.002672	98.98
12	0.003221	99.27	0.002496	99.33

In order to highlight some of the other features of the POD reduced-order model, let us next compare it with another reduced-order model based on the so-called *reduced-basis method* (RBM); see References [6, 34]. In Reference [6] several ways to choose reduced-basis subspaces were discussed. Here we consider the so-called Lagrange subspace and the basis elements in this subspace are snapshots of the solutions of the Navier–Stokes model. Supposing $\{\Psi_i\}_{i=1}^N$ denote the snapshots, the reduced-order subspace is defined as $V^{\mathrm{RBM}} = \mathrm{span}\{\Phi_i\}_{i=1}^N = \mathrm{span}\{\Psi_{i+1} - \Psi_i\}_{i=1}^N$ and the reduced-order solution is defined as $\mathbf{u} = \mathbf{u}_m + \sum_{i=1}^N \alpha_i \Phi_i$. Once we have a reduced-order subspace V^{RBM}, the Navier–Stokes model is projected onto V^{RBM} to obtain a reduced-order model as in Section 3.3. For the reduced-basis simulations, we simply took 100 snapshots in the time interval $[0, 25]$ and applied Galerkin projection to obtain the RBM reduced-order model.

The number of modes m employed was determined by comparing the result of the reduced-order model with the numerical solution of the Navier–Stokes model. The error in the reduced-order model decreases as we increase the number of eigenfunctions m and the best value for m is found to be 20 for both POD and SVD. Therefore we will use only 20 modes in our calculations reported in the forthcoming sections. Figures 5 and 6 show the temporal variation of the u and v components of the velocity field at the spatial location $(x, y) = (1.824, 1.616)$ with 20 modes for the POD and SVD reduced-order models, and 50 basis elements for the RBM-based reduced-order model. It reveals that the error takes the largest value at the initial stage and then it reduces to a small value as the system reaches a periodic state. Among the three approaches the SVD-based model was the most accurate when the number of modes was small. It was noted for the RBM that the condition number of the mass matrix can increase dramatically with increasing basis elements deteriorating convergence. However, the POD and SVD reduced-order models do not generate such bad condition numbers. This is due to the fact that snapshot sets often contain 'redundant' information and therefore they must be post-processed to remove as much of the redundancy as possible before they can be used for

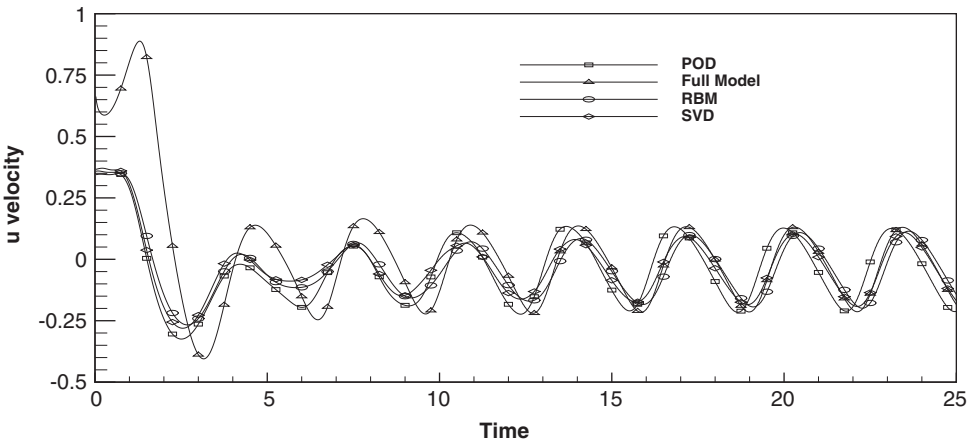

Figure 5. Time evolution of u velocity at spatial point $(1.824, 1.616)$ for POD, RBM and SVD models. Full model velocity is shown for comparison.

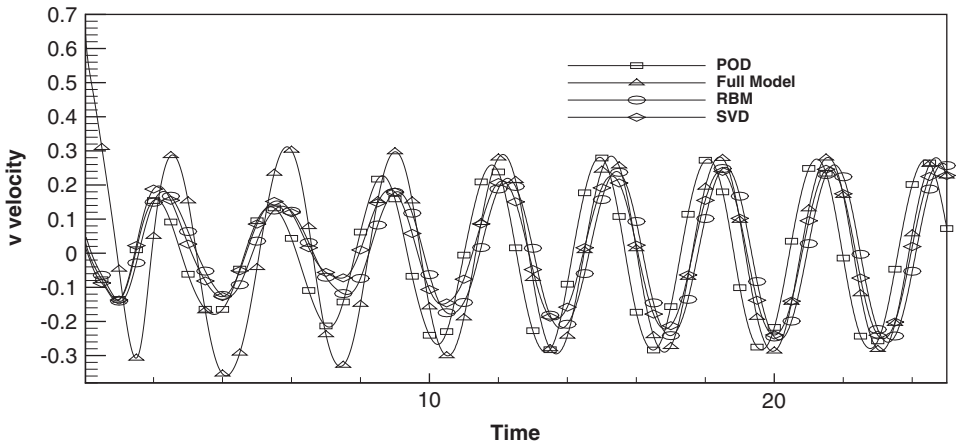

Figure 6. Time evolution of v velocity at spatial point $(1.824, 1.616)$ for POD, RBM and SVD models. Full model velocity is shown for comparison.

reduced-order modelling. POD and SVD may be viewed as a way to post-process snapshots. Figure 7 shows the time evolution of the first seven amplitude coefficients associated with the 20 mode POD reduced-order model. These time series reveal some well-defined structures, which also support the existence of low-dimensional dynamics. To gain insight into the actual structure of the attractor, one can use the time series to plot phase-space projections. For example, Figure 8 shows six projections for the amplitude coefficients. The relatively closed nature of the resulting curves further illustrate the presence of a low-dimensional attractor. We close this section with the observation that the reduced-order models should work in an interpolatory setting.

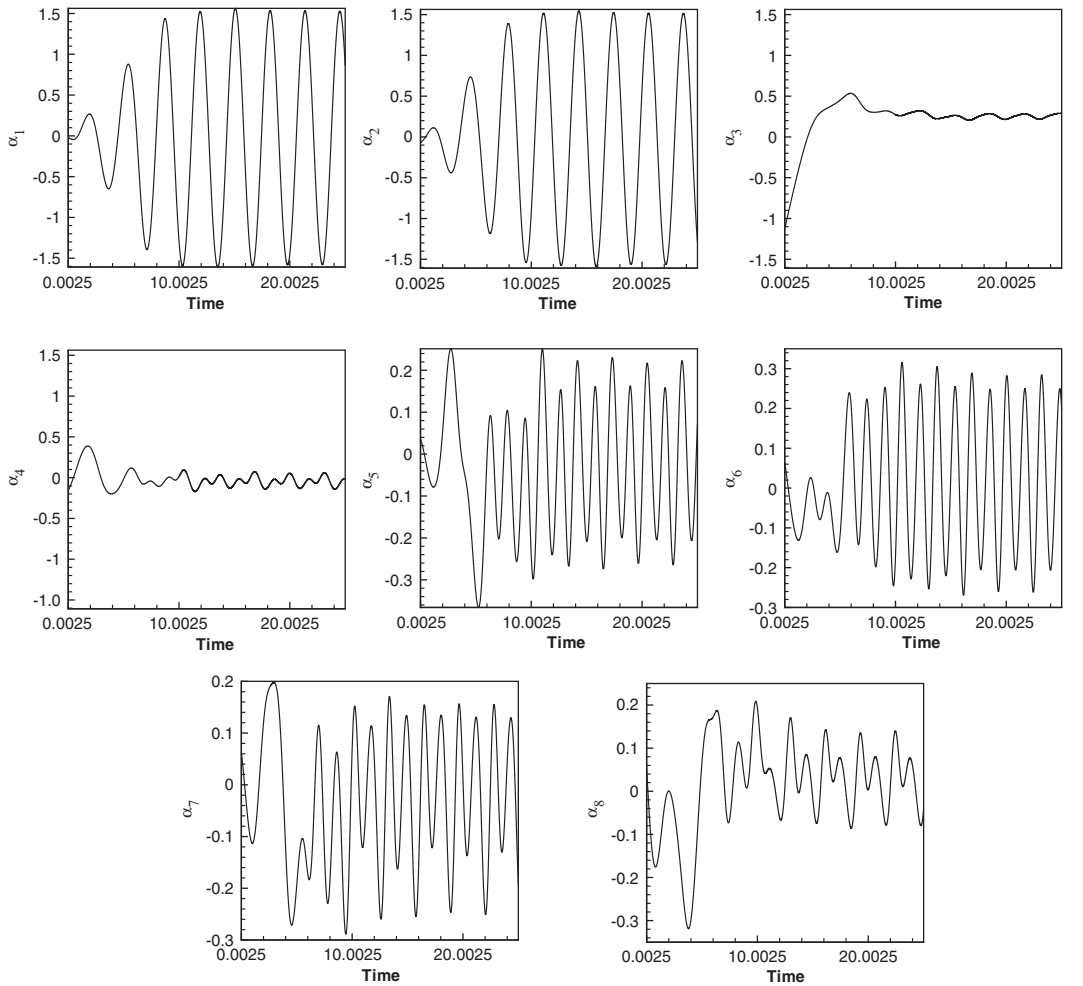

Figure 7. Time evolution of POD amplitude coefficients in the 20-mode reduced-order model.

5. REDUCED-ORDER OPTIMAL CONTROL APPROACH

5.1. *Control problem for airfoil*

In this section, we will formulate an optimal control problem for the control of vortex shedding for the flow past the thin airfoil. The thin airfoil has simple geometry and is a representative of high-lift aerodynamic flows with massive separation region. The optimal control problem we consider is that of suppressing the vortex shedding behind the airfoil by unsteady blowing control on the boundary of the airfoil using the least amount of control effort. Denoting the control actuator location on the boundary by Γ_c, we seek the control in the form

$$\mathbf{u}(x,t) = c(t)\mathbf{g}(\mathbf{x}) \quad \text{on } \Gamma_c \times [0,T]$$

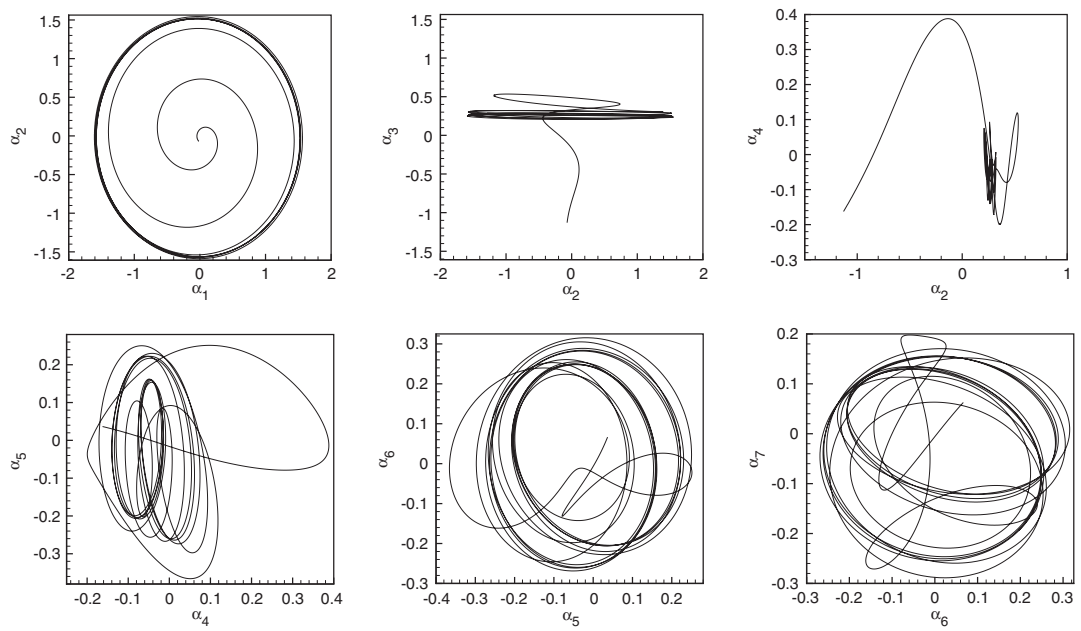

Figure 8. Phase space projections of amplitude coefficients obtained through POD.

where $c(t)$ and $\mathbf{g}(\mathbf{x})$ represent the undetermined control function and a prescribed velocity field, respectively. The first step in solving the optimal control problem is to represent the problem of interest as a cost functional to be minimized. The choice of cost functional to be minimized to achieve our goal is not trivial. Since our goal is to suppress the vortex shedding in the flow, a straight forward choice is to regulate the square of the vorticity (the enstrophy) over the time horizon $(0, T)$. Alternatively, one can minimize the wake unsteadiness or the H^1-norm of the velocity field or track a target velocity field. The other less direct choice is the minimization of viscous dissipation. We therefore consider five different cost functionals namely the cost functional for the enstrophy regulation

$$\mathscr{J}_{\mathrm{Ens}}(\mathbf{u}) = \frac{1}{2}\int_0^T \int_\Omega |\nabla \times \mathbf{u}|^2 \,\mathrm{d}\mathbf{x}\,\mathrm{d}t + \frac{\gamma}{2}\int_0^T |U|^2 \,\mathrm{d}t$$

where $U = \mathrm{d}c/\mathrm{d}t$, the cost functional for the viscous dissipation which for incompressible flow is equivalent to the drag on the surface

$$\mathscr{J}_{\mathrm{Dra}}(\mathbf{u}) = \frac{1}{2}\int_0^T \int_\Omega |\nabla \mathbf{u} + \nabla \mathbf{u}^{\mathrm{T}}|^2 \,\mathrm{d}\mathbf{x}\,\mathrm{d}t + \frac{\gamma}{2}\int_0^T |U|^2 \,\mathrm{d}t$$

the cost functional for tracking a target velocity field

$$\mathscr{J}_{\mathrm{Track}}(\mathbf{u}) = \frac{1}{2}\int_0^T \int_\Omega |\mathbf{u} - \mathbf{u}_d|^2 \,\mathrm{d}\mathbf{x}\,\mathrm{d}t + \frac{\gamma}{2}\int_0^T |U|^2 \,\mathrm{d}t$$

the cost functional for reducing the wake unsteadiness

$$\mathscr{J}_{\mathrm{Amp}}(\mathbf{v}) = \frac{1}{2}\int_0^T \int_\Omega |\mathbf{v}|^2 \,\mathrm{d}\mathbf{x}\,\mathrm{d}t + \frac{\gamma}{2}\int_0^T |U|^2 \,\mathrm{d}t$$

where $\mathbf{v} = \mathbf{u} - \mathbf{u}_m$ is the wake unsteadiness and the cost function for the kinetic energy in H^1-norm

$$\mathscr{J}_{\mathrm{H1KE}}(\mathbf{u}) = \frac{1}{2}\int_0^T \int_\Omega |\mathbf{u}|^2 + |\nabla\mathbf{u}|^2 \,\mathrm{d}\mathbf{x}\,\mathrm{d}t + \frac{\gamma}{2}\int_0^T |U|^2 \,\mathrm{d}t$$

In the above cost functionals, the first term is a measure of exactly that quantity we would like to minimize and the second term is necessary since we will not impose any *a priori* constraints on the controls. The parameter γ adjusts the relative weights of the two terms in the functional. The controlled Navier–Stokes system we consider is

$$\begin{aligned}
\frac{\partial\mathbf{u}}{\partial t} - \frac{1}{Re}\nabla^2\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u} + \nabla p = 0 \quad &\text{in } \Omega\times(0,T] \\
\nabla\cdot\mathbf{u} = 0 \quad &\text{in } \Omega\times(0,T] \\
\mathbf{u} = c(t)\mathbf{g}(\mathbf{x}) \quad &\text{on } \Gamma_{\mathrm{c}}\times(0,T] \\
\mathbf{u} = (\cos(\pi/4),\sin(\pi/4)) \quad &\text{on } \Gamma_{\mathrm{i}}\times(0,T] \\
-p\mathbf{n} + \frac{1}{Re}\frac{\partial\mathbf{u}}{\partial\mathbf{n}} = 0 \quad &\text{on } \Gamma_{\mathrm{o}}\times(0,T] \\
\mathbf{u}(\mathbf{x},0) = \mathbf{u}_0(\mathbf{x}) \quad &\text{in } \Omega
\end{aligned} \tag{15}$$

where Γ_{o}, Γ_{c} and Γ_{i} are the outflow, actuator and inflow parts of the boundary, respectively.

The optimal control problem we will consider can be described in a general manner as follows:
Find $U(t)$ that minimizes the cost $\mathscr{J}(\mathbf{u},U) = \frac{1}{2}\int_0^T [\mathscr{F}(\mathbf{u}) + \gamma|U|^2]\,\mathrm{d}t$ subject to the constraint that the flow fields satisfy the controlled Navier–Stokes system (15).

Solution of this nonlinear optimal control problem requires large-scale numerical approximation involving hundreds of thousands of variables; see for example References [3, 4, 7, 35] and [36] for adjoint equation approach to solve continuous optimal flow control problems. Below we discuss a controller design using reduced-order model which allows fast computations of control.

5.2. *Reduced-order control problem*

As noted before, a good snapshot set should have all the information needed to effect control using the associated reduced-order model. Obviously, some *a priori* knowledge about the types of states to be controlled using the reduced-order model is very useful. Our computational strategy for solving the above optimal control problem is to replace the Navier–Stokes system (15) by a reduced-order model and solve the resulting reduced-order control problem by an iterative minimization method. Therefore, we need a set of POD eigenfunctions that span the solution space of the system for various trajectories of $c(t)$, which include not only the optimal trajectory but also other trajectories appearing during the iterative minimization of the cost

functional. The snapshot set appropriate for this purpose is prepared as follows. We apply a time-varying profile $c(t) = t/12.5$ in the boundary condition on Γ_c and solve the Navier–Stokes model over the time interval $[0, 25]$ to record the snapshots. The spatial grid, time step size, Reynolds number and boundary conditions were all the same as in the simulations reported in Section 4. Two hundred snapshots were recorded at constant time interval Δt^* ($\Delta t^* = 5\Delta t$, $\Delta t = 0.025$). Another interesting approach for this purpose is discussed in Reference [29] that involves adaptively updating the POD eigenfunctions.

As described in Reference [19], it is necessary to remove the inhomogeneities on the boundary. A convenient way to do this is to introduce a reference flow field $(\mathbf{u}_r, p_r, \phi_r)$ that satisfies the steady state version of the Navier–Stokes system (15) with a fixed control profile on Γ_c, i.e. $\Gamma_c : \mathbf{u}(x) = c_r g(\mathbf{x})$. Let $(\mathbf{u}_{r_1}, p_{r_1})$ and $(\mathbf{u}_{r_0}, p_{r_0})$ be two flow fields corresponding to $c_r = 1$ and 0, respectively. Set $(\mathbf{u}_r, p_r) = (\mathbf{u}_{r_1}, p_{r_1}) - (\mathbf{u}_{r_0}, p_{r_0})$. Then each flow field in the modified snapshot set

$$\{(\mathbf{u}(\mathbf{x}, t^k), p(\mathbf{x}, t^k)) - c(t^k)(\mathbf{u}_r(\mathbf{x}), p_r(\mathbf{x}))\}, \quad k = 1, \ldots, n$$

satisfies homogeneous boundary conditions on Γ_c. Finally, we let $(\mathbf{u}_m, p_m)$ be the mean flow field of these modified fields and define a new snapshot set

$$\{(\mathbf{u}(\mathbf{x}, t^k), p(\mathbf{x}, t^k)) - c(t^k)(\mathbf{u}_r, p_r) - (\mathbf{u}_m, p_m)\}, \quad k = 1, \ldots, n$$

which satisfies homogeneous boundary conditions on all the boundaries. To these n snapshots, we apply the POD to obtain the basis functions in descending order according to the information content of the system.

We employ the Galerkin projection on the Navier–Stokes model with the above POD basis functions to derive the reduced-order model to be used in the optimization algorithm to solve the optimal control problem. The flow field is decomposed as follows:

$$(\mathbf{u}, p) = (\mathbf{u}_m, p_m) + c(t)(\mathbf{u}_r, p_r) + \sum_{i=1}^{m} \alpha_i(t)\boldsymbol{\Phi}_i \tag{16}$$

where $\boldsymbol{\Phi}_i$ is the ith POD basis function, $\alpha_i(t)$ the corresponding coefficient and m the total number of POD basis functions. The reduced-order optimal control problem is obtained by inserting expansion (16) into (13) and the cost functional $\mathscr{J}(\mathbf{u}, U)$:

$$\begin{aligned} &\text{Minimize} \quad \mathscr{J}(\mathbf{X}, U) = \int_0^T \left[\ell(\mathbf{X}) + \frac{\gamma}{2} U^2\right] \mathrm{d}t \\ &\text{subject to} \quad \frac{\mathrm{d}\mathbf{X}}{\mathrm{d}t} = f(\mathbf{X}) + BU \\ &\qquad\qquad \mathbf{X}(0) = \mathbf{X}_0 \end{aligned} \tag{17}$$

where $\mathbf{X} = (1, \alpha, c)^{\mathrm{T}}$, $f(\mathbf{X}) = -\mathbf{A}\mathbf{X} - \mathbf{N}(\mathbf{X})$, $i, j = 0, \ldots, m+1$. In order to solve it numerically, we first discretize it in time by using Crank–Nicholson method for the time derivative and

trapezoidal rule for the time integral, and rewrite it as

$$\begin{aligned} &\text{Minimize} \quad \mathscr{J}(\mathbf{Z}) = \sum_{k=1}^{M} [\tfrac{1}{2}(\ell(\mathbf{X}^{k-1}) + \ell(\mathbf{X}^{k})) + h(U^{k})]\Delta t \\ &\text{subject to} \quad F(\mathbf{Z}) = \begin{bmatrix} \dfrac{\mathbf{X}^1 - \mathbf{X}^0}{\Delta t} - \dfrac{1}{2}(f(\mathbf{X}^1) + f(\mathbf{X}^0)) + BU^1 \\ \vdots \\ \dfrac{\mathbf{X}^M - \mathbf{X}^{M-1}}{\Delta t} - \dfrac{1}{2}(f(\mathbf{X}^M) + f(\mathbf{X}^{M-1})) + BU^M \end{bmatrix} = 0 \end{aligned} \tag{18}$$

where $\mathbf{Z} = (\mathbf{X}, U)$ and $\Delta t = T/M$. This is a finite-dimensional nonlinear optimal control problem or nonlinear programming problem:

$$\begin{aligned} &\text{Minimize} \quad \mathscr{J}(\mathbf{Z}) \\ &\text{subject to} \quad F(\mathbf{Z}) = 0 \end{aligned} \tag{19}$$

An efficient method for the solution of (19) is the sequential quadratic programming method (SQP); see References [3, 4, 35, 36] for SQP method developments in solving related optimal control problems in infinite-dimensional setting. The SQP method solves the nonlinear optimal control problem (18) by a sequence of linear quadratic control problems.

Suppose a current estimate $\mathbf{Z}^{(j)}$ to the minimizer $\mathbf{Z}^*$ is known, a search direction $\boldsymbol{\delta}^{(j)}$ is computed by solving the quadratic programming problem

$$\begin{aligned} &\text{Minimize} \quad \tfrac{1}{2}\boldsymbol{\delta}^{(j)\mathrm{T}} B^{(j)} \boldsymbol{\delta}^{(j)} + \boldsymbol{\delta}^{(j)\mathrm{T}} \nabla \mathscr{J}(\mathbf{Z}^{(j)}) \\ &\text{such that} \quad \nabla F(\mathbf{Z}^{(j)})^{\mathrm{T}} \boldsymbol{\delta}^{(j)} + F(\mathbf{Z}^{(j)}) = 0 \end{aligned} \tag{20}$$

where $\nabla F(\mathbf{Z}^{(j)}) = \{\nabla F_1(\mathbf{Z}^{(j)}), \ldots, \nabla F_M(\mathbf{Z}^{(j)})\}$, the matrix of the constraint normals evaluated at $\mathbf{Z}^{(j)}$, $F(\mathbf{Z})$ is the vector $(F_1(\mathbf{Z}), \ldots, F_M(\mathbf{Z}))^{\mathrm{T}}$, and $B^{(j)}$ is a positive definite approximation to the Hessian, with respect to $\mathbf{Z}$, of the Lagrangian

$$\mathscr{L}(\mathbf{Z}, \boldsymbol{\zeta}) = \mathscr{J}(\mathbf{Z}) + \boldsymbol{\zeta}^{\mathrm{T}} F(\mathbf{Z})$$

where $\boldsymbol{\zeta} = (\zeta_1, \ldots, \zeta_M)^{\mathrm{T}}$ is the vector of Lagrange multipliers, and $B^{(j)}$ is an estimate to $\mathscr{L}_{ZZ}(\mathbf{Z}^{(j)}, \boldsymbol{\zeta}^{(j)})$ for an estimate $\boldsymbol{\zeta}^{(j)}$ to the optimal Lagrange multiplier $\boldsymbol{\zeta}^*$. The solution to quadric programming problem (20) is obtained by solving the associated first-order necessary condition of optimality [4, 36].

To continue the iterative procedure a new estimate to the minimizer $\mathbf{Z}^*$ is chosen by

$$\mathbf{Z}^{(j+1)} = \mathbf{Z}^{(j)} + \eta^{(j)} \boldsymbol{\delta}^{(j)}$$

where $\eta^{(j)}$ is an appropriately chosen scalar step length, and new estimates $B^{(j+1)}$ and $\boldsymbol{\zeta}^{(j+1)}$ are computed. Quadratic programming problem (20) is solved again with j replaced with $j+1$, and the process is repeated until the pair $(\mathbf{Z}^{(j)}, \boldsymbol{\zeta}^{(j)})$ satisfy the Karush–Kuhn–Tucker conditions [4, 36] for a minimizer. The SQP requires satisfaction of only a linear approximation of the state constraints, avoiding the need to converge fully. Thus, the state equations are satisfied as the control values converge to their optimal values and the convergence is quadratic.

6. COMPUTATIONAL RESULTS FOR CONTROL

In this section, we present the results for the computation of optimal control using the reduced-order approaches. Effective placement of control actuator is crucial for flow control. We address this problem by the following criterion: Carry out simulations with blowing at different locations and see which one has the 'maximum' influence. We first choose to place the actuator on the upper surface of the airfoil. This choice is motivated by the heuristic that if one wants maximum influence in the flow, then the control has to be applied in that vicinity. Furthermore, we define the prescribed velocity function $\mathbf{g}$ in the control definition to be $\mathbf{g}(\mathbf{x}) = (\cos(\pi/9), \sin(\pi/9))$ so that the blowing is directed downstream and into the boundary layer and wake. Such an unsteady blowing is known to energize the low momentum fluid inside the boundary layer and mitigate flow separation. We next find where exactly on the upper surface the actuator has to be placed for best performance. To find this, we divided this portion of the airfoil into 10 equal parts of size $\Delta x = 1/10$ and computed the control for each of these cases. For the above criterion, the best position for the actuator was found to be $\Gamma_c : 1 \leqslant x \leqslant 1.1$ which is in the vicinity of the leading edge separation point. For the computational results presented in the rest of this section, we placed the actuator at this optimal position. The penalty parameter γ plays a crucial role in the control design and we select it to be $10 \leqslant \gamma \leqslant 100$ to avoid under/overshooting.

We have performed control simulations with all three reduced-order modelling approaches. As in the case of uncontrolled simulations reported in Section 4, POD and SVD-based reduced-order models required only 20 modes to converge and converged to comparable optimal controls whereas the one based on RBM did not converge with the same number of modes. Therefore only results for the 20-mode POD model case is reported here. The choice of the cost functional to be minimized is crucial to achieve the control objective. Figures 9, 10, 11, 12 and 13 show the performance of the optimal controls based on $\mathcal{J}_{\text{Amp}}$ and $\mathcal{J}_{\text{Track}}$, respectively. Shown in the figures are the contours of speed of the flow at time $T = 25$. The control for the tracking cost function is significantly different from the other cases display the distribution of the optimal control $(c(t))$ on the actuator for the cost functions $\mathcal{J}_{\text{Track}}$, $\mathcal{J}_{\text{H1KE}}$, $\mathcal{J}_{\text{Dra}}$, $\mathcal{J}_{\text{Ens}}$ and $\mathcal{J}_{\text{Amp}}$, respectively. For the tracking functional $\mathcal{J}_{\text{Track}}$, we chose the desired velocity $\mathbf{u}_d$ to be the steady velocity field for $Re = 1$ with zero boundary conditions

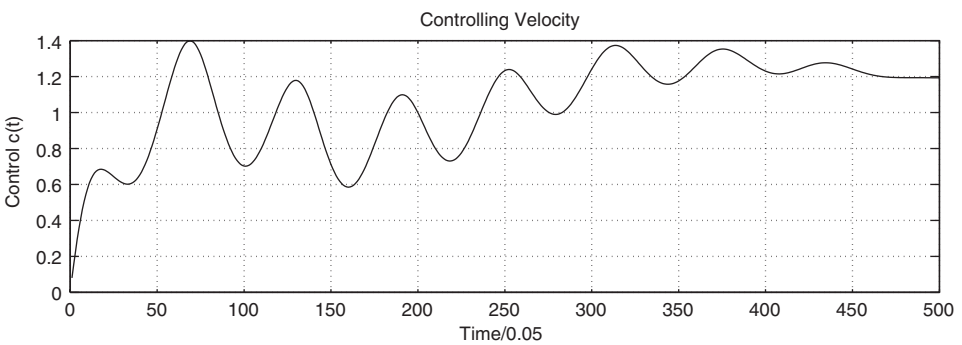

Figure 9. Computed control as a function of time for cost-function $\mathcal{J}_{\text{Track}}$.

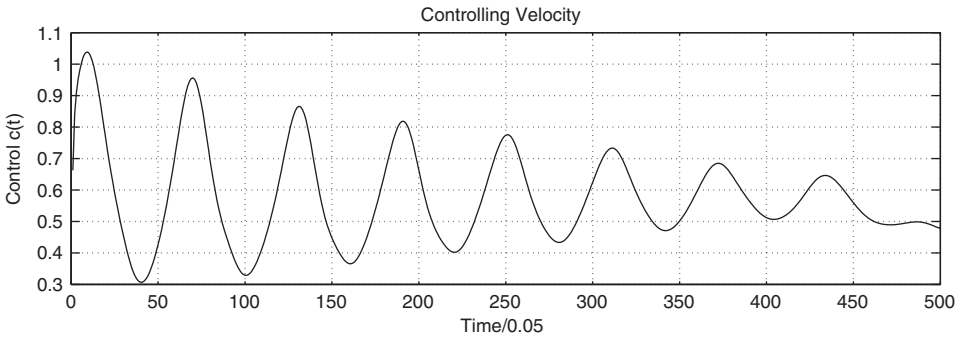

Figure 10. Computed control as a function of time for cost-function $\mathcal{J}_{\mathrm{H1KE}}$.

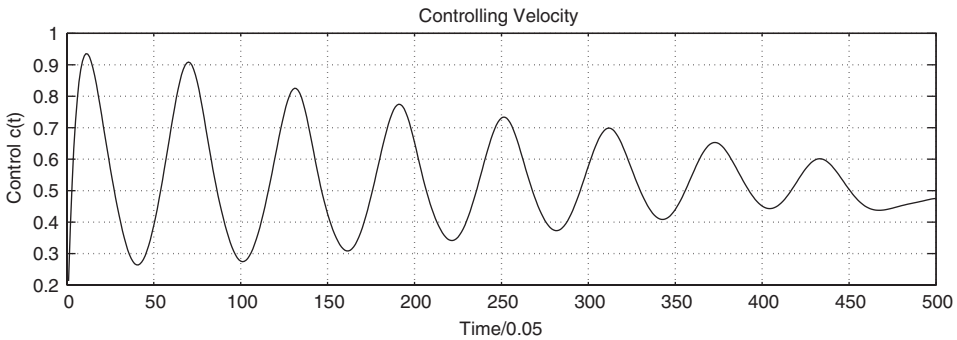

Figure 11. Computed control as a function of time for cost-function $\mathcal{J}_{\mathrm{Dra}}$.

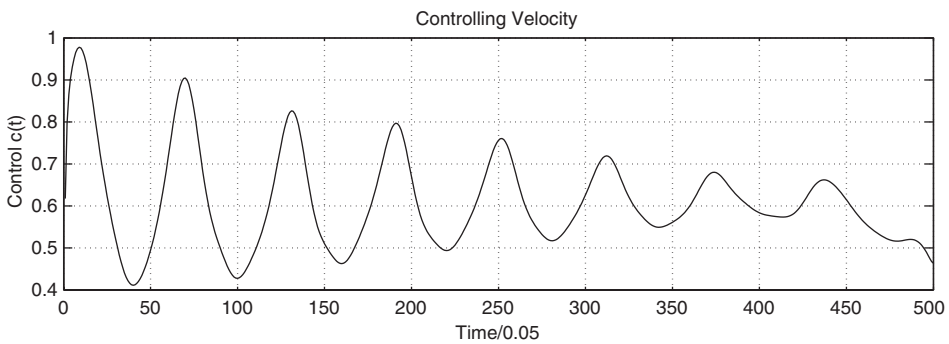

Figure 12. Computed control as a function of time for cost-function $\mathcal{J}_{\mathrm{Ens}}$.

on Γ_c. Figure 14 and gave the best performance in terms of suppressing the vortex shedding. The performance for the other three controls were about the same. As indicated by the controlled flow fields, the vortex shedding has been effectively eliminated by the optimal unsteady blowing control.

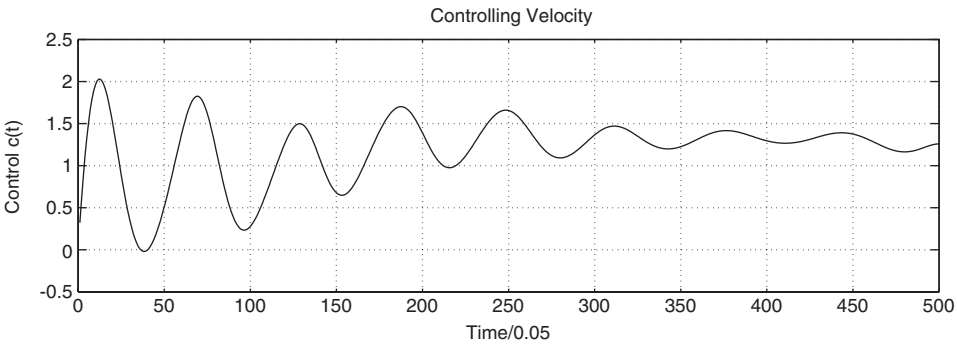

Figure 13. Computed control as a function of time for cost-function $\mathcal{J}_{\text{Amp}}$.

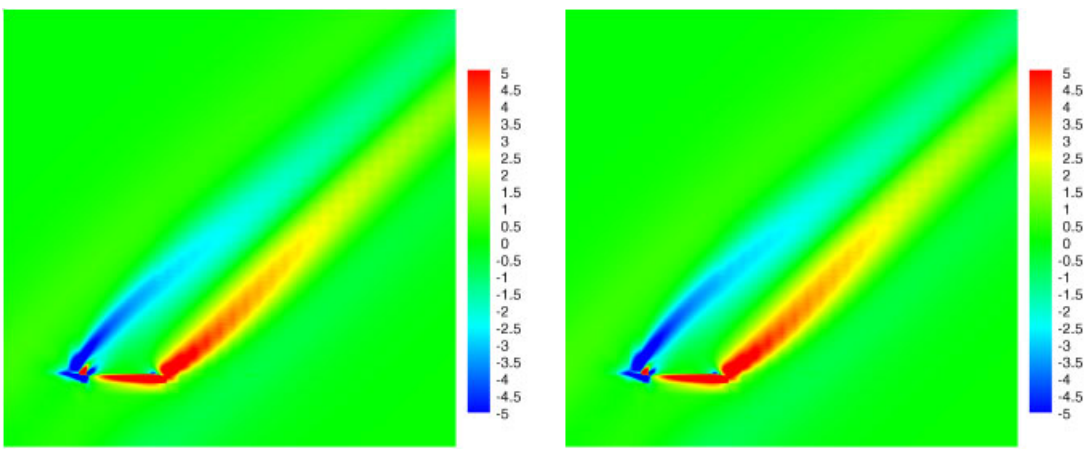

Figure 14. Vorticity of controlled flow for cost-function $\mathcal{J}_{\text{Amp}}$ (left) and $\mathcal{J}_{\text{Track}}$ (right).

Let us next comment on the CPU time required for each step in the reduced-order model construction and compare the total CPU time with that of the finite element method when simulating the Navier–Stokes equations during a certain period of time, say $[0, 25]$. When SUN Ultra 60 workstation is used, it requires about 333 h to obtain 200 snapshots and 5 min (or 7 min) to obtain the POD modes (or SVD) from these snapshots. As expected the consumption of CPU time for the reduced-order control computation is much less than that for the case of the full-order control computation since the degree of freedom of the former is only about $5/3571$ of the latter. In the control computation, one iteration of the reduced-order SQP requires about 10 min. The number of iterations needed to reach the converged optimal control is 6. The remaining steps in the POD (or SVD) takes about 6 min (or 8 min). The total CPU time requirement for the reduced-order control computation is about 334 h. This is a substantial savings in computational time since the control computation with full-order Navier–Stokes model using the conjugate-gradient method (CGM) (for e.g.) requires solving the Navier–Stokes equations and the adjoint Navier–Stokes equations each time the control

is updated and the CGM typically takes about 40–60 iterations before giving a converged optimal control. Therefore, the construction of the low-dimensional model consumes about that required in just one iteration of the CGM. This savings in computational time is mainly due to the fact that the adjoint equation of the Navier–Stokes equations, which is computationally as costly as the Navier–Stokes equation itself, is computed using the reduced-order model in the reduced-order control computation. The reduction in computational time with the use of the reduced-order models in three-dimensional complex geometries will be much more significant as the difference in the degrees of freedom between reduced- and full-order models will become much larger. However our reduced-order model design does not introduce any additional difficulties in three dimension as it employs Galerkin projection.

7. CONCLUDING REMARKS

We have presented a fast computational approach for controller design using reduced-order models for control of vortex shedding behind a thin airfoil. The reduced-order models are derived by combining Galerkin projection and POD or other related reduced-order approaches. The fast computational approach for controller design was used to compute the optimal control that suppresses the vortex shedding. The control was effected through blowing on the airfoil. It was found that downstream directed blowing on the upper surface of the airfoil near the leading edge is more efficient in mitigating the flow separation and suppressing vortex shedding.

REFERENCES

1. Gunzburger MD. *Perspectives in Flow Control and Optimization*. SIAM: Philadelphia, PA, 2003.
2. Sritharan SS (ed.). *Optimal Control of Viscous Flows*. SIAM: Philadelphia, PA, 1998.
3. Hou LS, Ravindran SS. Computations of boundary optimal control problems for an electrically conducting fluid. *Journal of Computational Physics* 1996; **128**(2):319–330.
4. Hou LS, Ravindran SS. Numerical approximation of optimal flow control problems by a penalty method: error estimates and numerical results. *SIAM Journal on Scientific Computing* 1999; **20**(5):1753–1777.
5. Fattorini HO, Sritharan SS. Existence of optimal controls for viscous flow problems. *Proceedings of the Royal Society of London*, *Series A* 1992; **439**:81–102.
6. Ito K, Ravindran SS. A reduced order method for simulation and control of fluid flows. *Journal of Computational Physics* 1998; **143**(2):403–425.
7. Ito K, Ravindran SS. Optimal control of thermally convected fluid flows. *SIAM Journal on Scientific Computing* 1998; **19**(6):1847–1869.
8. Barbu V, Sritharan SS. H^{∞}-control theory of fluid dynamics. *Proceedings of the Royal Society of London*, *Series A*, *Mathematical*, *Physical and Engineering Sciences* 1998; **454**:3009–3033.
9. Bewley TR, Liu S. Optimal and robust control and estimation of linear paths to transition. *Journal of Fluid Mechanics* 1998; **365**:305–349.
10. Moore BC. Principal component analysis in linear systems: controllability, observability and model reduction. *IEEE Transactions on Automatic Control* 1981; **26**(1):17–32.
11. Adamjan VM, Arov DZ, Krein MG. Analytic properties of Schmidt pairs for a Hankel operator and the generalized Schur-Takagi problem. *Mathematics of the USSR-Sbornik* 1971; **15**:31–73.
12. Lanczos C. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *Journal of Research of the National Bureau of Standards* 1950; **45**:255–282.
13. Arnoldi WE. The principle of minimized iterations in the solution of the matrix eigenvalue problems. *Quarterly of Applied Mathematics* 1951; **9**:17–29.
14. Holmes P, Lumley J, Berkooz G. *Turbulence*, *Coherent Structures*, *Dynamical Systems and Symmetry*. Cambridge, 1996.
15. Sirovich L. Turbulence and the dynamics of coherent structures: part I–III. *Quarterly of Applied Mathematics* 1987; **45**(3):561–590.

16. Lumley JL. The structure of inhomogeneous turbulent flows. In *Atmospheric Turbulence and Radio Wave Propagation*, Yaglom AM, Tatarski VI (eds). Nauka: Moscow, 1967; 167–178.
17. Ravindran SS. Proper orthogonal decomposition in optimal control of fluids. *NASA Technical Memorandum*, *NASA TM 1999-209113*, 1999.
18. Graham MD, Kevrekidis IG. Alternative approaches to the Karhunen-Lo'eve decomposition for model reduction and data analysis. *Computer and Chemical Engineering* 1996; **20**:495–506.
19. Ravindran SS. A reduced order approach to optimal control of fluids using proper orthogonal decomposition. *International Journal for Numerical Methods in Fluids* 2000; **34**(5):425–448.
20. Atwell JA, King BB. Proper orthogonal decomposition for reduced basis feedback controllers for parabolic equations. *Mathematical and Computer Modelling* 2001; **33**(1–3):1–19.
21. Ly HV, Tran, Hien T. Proper orthogonal decomposition for flow calculations and optimal control in a horizontal CVD reactor. *Quarterly of Applied Mathematics* 2002; **60**(4):631–656.
22. Maddalon DV, Collier FS, Montoya FS, Land CK. Transition flight experiments on a swept wing with suction. *AIAA Paper 89-1893*, 1989.
23. Yoshihara H, Zonars D. The transonic jet flap—a review of recent results. *Proceedings*, *SAE National Aerospace and Manufacturing Meeting*, 1975.
24. Hassan AA, JanakiRam RD. Effects of zero-mass synthetic jets on the aerodynamics of the NACA-0012 airfoil. *AIAA 97-2326*, 1997.
25. Modi VJ, Mokhtarian F, Fernando M, Yokomizo T. Moving surface boundary-layer control as applied to two-dimensional airfoils. *Journal of Aircraft* 1991; **28**:104–112.
26. Karhunen K. Zur Spekraltheorie Stochasticher. *Annales Academiae Scientiarum Fennicae Series A* 1946; **1**:34.
27. Loeve M. *em Probability Theory*. New York, 1955.
28. Hilai R, Rubinstein J. Recognition of rotated images by invariant Karhunen–Loeve expansion. *Journal of the Optical Society of America A—Optics Image Science and Vision* 1994; **11**(N5):1610–1618.
29. Ravindran SS. Reduced-order adaptive controllers for fluid flows using POD. *Journal of Scientific Computing* 2000; **15**(4):457–478.
30. *International Mathematical and Statistical Library* (IMSL). *MATH/LIBRARY*, Chapters 1–2, vol. 1, Houston, 1989; 1–398.
31. Smith B, Boyle JM, Dongarra JJ, Garbow BS, Ikebe Y, Klema VC, Moler CB. *Matrix Eigensystem Routines–EISPACK Guide*. Springer: New York, 1976.
32. Anderson E, Bai Z, Bischof C, Demmel J, Dongarra J, Du Croz J, Greenbaum A, Hammarling S, McKenney A, Ostrouchov S, Sorensen D. *LAPACK Users' Guide*. SIAM: Philadelphia, PA, 1992. http://www.netlib.org/lapack/
33. Demmel JW, Kahan WH. Accurate singular values of didiagonal matrices. *SIAM Journal on Scientific Computing* 1990; **11**:873–912.
34. Grepl MA, Patera AT. A posteriori error bounds for reduced-basis approximations of parameterized parabolic partial differential equations. *M2AN Mathematical Modelling and Numerical Analysis* 2005; **39**(1):157–181.
35. Ravindran SS. Numerical approximation of optimal flow control problems by SQP method. In *Optimal Control of Viscous Flows*, Sritharan SS (ed.), SIAM Proceedings in Applied Mathematics. SIAM: Philadelphia, PA, 1998; 181–198.
36. Ravindran SS. Numerical approximation of optimal control of unsteady flows using SQP and time decomposition. *International Journal for Numerical Methods in Fluids* 2004; **45**(1):21–42.
37. Hou LS, Ravindran SS. A penalized Neumann control approach for solving an optimal Dirichlet control problem for the Navier–Stokes equations. *SIAM Journal on Control and Optimization* 1997; **36**(5):1795–1814.

