Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 196 (2006) 337–355
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

POD and CVT-based reduced-order modeling of Navier–Stokes flows

John Burkardt [a], Max Gunzburger [a,*], Hyung-Chun Lee [b,1]

[a] School of Computational Science, Florida State University, Tallahassee, FL 32306-4120, USA
[b] Department of Mathematics, Ajou University, Suwon 442-749, Republic of Korea

Received 25 August 2005; received in revised form 22 December 2005; accepted 11 April 2006

Abstract

A discussion of reduced-order modeling for complex systems such as fluid flows is given to provide a context for the construction and application of reduced-order bases. Reviews of the POD (proper orthogonal decomposition) and CVT (centroidal Voronoi tessellation) approaches to reduced-order modeling are provided, including descriptions of POD and CVT reduced-order bases, their construction from snapshot sets, and their application to the low-cost simulation of the Navier–Stokes system. Some concrete incompressible flow examples are used to illustrate the construction and use of POD and CVT reduced-order bases and to compare and contrast the two approaches to reduced-order modeling.

Keywords: Reduced-order modeling; Proper orthogonal decomposition; Centroidal Voronoi tessellations; Navier–Stokes equations

1. Introduction

Even with the use of good mesh generators, discretization schemes, and solution algorithms, the computational simulation of solutions of the Navier–Stokes system still remains a formidable endeavor. For the accurate simulations, typical finite element codes may require many thousands of degrees of freedom. The situation is even worse for optimization problems for which multiple solutions are usually required or for feedback control problems for which real-time solutions are needed. The types of reduced-order models that we study are those that attempt to determine accurate approximate solutions using very few degrees of freedom. To do so, such models have to use basis functions that are in some way intimately connected to the problem being approximated.

There have been many reduced-order modeling techniques proposed; see, e.g., [28–31,39] for some early examples. The most popular one, at least for the applications we have in mind, is based on proper orthogonal decomposition (POD) analysis. POD begins with a set of snapshots that are generated by evaluating the computational solution of transient problems at several instants of time or by evaluating the computational solution for several values of the parameters appearing in the problem description or by a combination of the two. The computational solutions that are used to determine the snapshot set are determined using costly, large-scale, high-fidelity, e.g., finite element, codes. The POD basis is then given by the left singular vectors corresponding to the most dominant singular values of the matrix having the snapshot vectors as its columns. The POD basis is then used, usually by applying a projection procedure, to determine an approximate solution for different values of the system parameters. POD-based model reduction has been applied with some success to several problems, notable including fluid mechanics settings. For detailed discussions, one may consult [1–5,16,21,22,24–26,33–38,40–46,50,51].

Centroidal Voronoi tessellations (CVTs) have been successfully used in several applications [6–8,10–14,8,32], including data compression settings, e.g., in image processing

* Corresponding author.
E-mail addresses: burkardt@csit.fsu.edu (J. Burkardt), gunzburg@csit.fsu.edu (M. Gunzburger), hclee@ajou.ac.kr (H.-C. Lee).
[1] Partially supported by KOSEF R01-2000-000-00008-0.

0045-7825/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.cma.2006.04.004

and the clustering of data. Reduced-order modeling of the Navier–Stokes systems is another data compression setting, i.e., one replaces high-dimensional approximations with low-dimensional ones. CVTs can be used for this purpose as well. In CVT reduced-order modeling, we start with a snapshot set just as is done in a POD-based setting. We then construct a special Voronoi clustering of the snapshot set for which the means of the clusters are also the generators of the corresponding Voronoi clusters. The generators of the Voronoi clustering constitute the CVT reduced-order basis. We then use the CVT-based basis in just the same way as one uses a POD-based basis to determine a very low-dimensional approximation to the solution of the Navier–Stokes system.

The efficiency of POD and CVT reduced bases depend on their dimension, i.e., if a reduced basis is low-dimensional and can still approximate the state well, then accurate approximations of the solution of the Navier–Stokes system can be inexpensively determined. However, the ability of a reduced basis to approximate the state of a system is dependent on the information contained in the snapshot set used to generate the basis. Certainly, a reduced basis cannot contain more information than that contained in the snapshot set. Thus, crucial to the success of reduced-order modeling approaches to model reduction is the generation of "good" snapshot sets.

The plan for the rest of the paper is as follows. In the remainder of this section, we briefly review finite element approximations of solutions of the Navier–Stokes equations. Then, in Section 2, we show how POD and CVT bases are defined and constructed and how they are used to determine very low-dimensional approximations. In Section 3, we use some concrete examples to show how snapshots sets can be generated and to compare and contrast the performance of POD and CVT-based reduced-order modeling. We also discuss several practical issues related to the implementation of reduced-order models. Finally, in Section 4, we provide some concluding remarks.

1.1. The Navier–Stokes system and its finite element approximation

Let Ω denote a bounded region in $\Re^2$ whose boundary is denoted by $\partial\Omega = \Gamma_D \cup \Gamma_N$ and let T denote a positive constant. Let ν denote the given (constant) kinematic viscosity of the fluid, $\mathbf{f}(t,\mathbf{x})$ a given body force per unit mass, $\mathbf{u}_0$ a given (solenoidal) initial velocity, and $\mathbf{b}$ a specified boundary velocity. Furthermore, let $\mathbf{u}(t,\mathbf{x})$ and $p(t,\mathbf{x})$ denote the velocity and pressure fields, respectively, that are required to satisfy the Navier–Stokes system:

$$\begin{cases} \mathbf{u}_t - 2\nu D(\mathbf{u}) + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } \Omega\times(0,T], \\ \nabla\cdot\mathbf{u} = 0 \quad \text{in } \Omega\times(0,T], \\ \mathbf{u} = \mathbf{b} \quad \text{on } \Gamma_D\times(0,T], \\ -p\mathbf{n} + \nu D(\mathbf{u})\cdot\mathbf{n} = \mathbf{0} \quad \text{on } \Gamma_N\times(0,T], \\ \mathbf{u}(0,\mathbf{x}) = \mathbf{u}_0(\mathbf{x}) \quad \text{in } \Omega, \end{cases} \tag{1}$$

where $D(\mathbf{u}) = \frac{1}{2}(\nabla\mathbf{u} + (\nabla\mathbf{u}))^{\mathrm{T}}$.

We use a variational formulation to help define a finite element method to approximate (1), but other methods can be also used in the context of reduced-order modeling. A variational formulation of the 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

$$\begin{cases} \displaystyle\int_\Omega \mathbf{u}_t\cdot\mathbf{v}\,d\Omega + 2\nu\int_\Omega D(\mathbf{u}) : D(\mathbf{v})\,d\Omega + \int_\Omega (\mathbf{u}\cdot\nabla)\mathbf{u}\cdot\mathbf{v}\,d\Omega, \\ \quad -\displaystyle\int_\Omega p\nabla\cdot\mathbf{v}\,d\Omega = \int_\Omega \mathbf{f}\cdot\mathbf{v}\,d\Omega \quad \text{for all } \mathbf{v}\in\mathbf{H}_0^1(\Omega), \\ \displaystyle\int_\Omega q\nabla\cdot\mathbf{u}\,d\Omega = 0 \quad \text{for all } q\in L_0^2(\Omega), \\ \mathbf{u}(0,\mathbf{x}) = \mathbf{u}_0(\mathbf{x}) \quad \text{in } \Omega, \end{cases} \tag{2}$$

where $\mathbf{V}_b = \{\mathbf{u}\in\mathbf{H}^1(\Omega) : \mathbf{u} = \mathbf{b} \text{ on } \Gamma_D, \mathbf{b}\in\mathbf{H}^{1/2}(\Gamma_D)\}$, $\mathbf{H}_0^1 = \{\mathbf{u}\in\mathbf{H}^1(\Omega) : \mathbf{u} = \mathbf{0} \text{ on } \Gamma_D\}$, and $L_0^2(\Omega) = \{q\in L^2(\Omega) : \int_\Omega q\,d\Omega = 0\}$.

A typical finite element approximation of (2) is defined as follows: we first choose conforming finite element subspaces $\mathbf{V}^h \subset \mathbf{H}^1(\Omega)$ and $S^h \subset L^2(\Omega)$ and then define $\mathbf{V}_0^h = \mathbf{V}^h \cap \mathbf{H}_0^1(\Omega)$, $\mathbf{V}_b^h = \{\mathbf{v}^h\in\mathbf{V}^h : \mathbf{v}^h = \mathbf{b}^h \text{ on } \Gamma_D\}$, and $S_0^h = S^h \cap L_0^2(\Omega)$, where $\mathbf{b}^h(t,\cdot)$ is an approximation to $\mathbf{b}(t,\cdot)$, e.g., $\mathbf{b}^h(t,\cdot)\in\mathbf{V}^h|_{\Gamma_D}$ can be a nodal interpolant of $\mathbf{b}(t,\cdot)$. One then seeks $\mathbf{u}^h(t,\cdot)\in\mathbf{V}_b^h$ and $p^h\in S_0^h$ such that

$$\begin{cases} \displaystyle\int_\Omega \mathbf{u}_t^h\cdot\mathbf{v}^h\,d\Omega + 2\nu\int_\Omega D(\mathbf{u}^h) : D(\mathbf{v}^h)\,d\Omega + \int_\Omega (\mathbf{u}^h\cdot\nabla)\mathbf{u}^h\cdot\mathbf{v}^h\,d\Omega, \\ \quad -\displaystyle\int_\Omega p^h\nabla\cdot\mathbf{v}^h\,d\Omega = \int_\Omega \mathbf{f}\cdot\mathbf{v}^h\,d\Omega \quad \text{for all } \mathbf{v}^h\in\mathbf{V}_0^h, \\ \displaystyle\int_\Omega q^h\nabla\cdot\mathbf{u}^h\,d\Omega = 0 \quad \text{for all } q^h\in S_0^h, \\ \mathbf{u}^h(0,\mathbf{x}) = \mathbf{u}_0^h(\mathbf{x}) \quad \text{in } \Omega, \end{cases} \tag{3}$$

where $\mathbf{u}_0^h(\mathbf{x})\in\mathbf{V}_b^h$ is an approximation, e.g., a projection, of $\mathbf{u}_0(\mathbf{x})$. Discretization is completed by choosing a time-marching method such as the backward-Euler scheme. See, e.g., [15,18] for details about finite element discretizations of the Navier–Stokes system.

2. Reduced-order modeling for the Navier–Stokes equations

In this section, we briefly describe two reduced-order models for the Navier–Stokes system. In Section 3, we will use concrete examples to compare the relative accuracy and efficiency of the two approaches. Both approaches rely on first generating a set of snapshots; see Section 3.1 for a discussion of snapshots, including how they are generated and how they are modified to satisfy zero boundary conditions.

2.1. POD reduced-order bases

Given a discrete set of snapshot vectors $W = \{\vec{w}_n\}_{n=1}^N$ belonging to $\Re^J$, where $N < J$, we form the $J\times N$ snapshot matrix $\mathbb{A}$ whose columns are the snapshot vectors $\vec{w}_n$:

$$\mathbb{A} = (\,\vec{w}_1 \quad \vec{w}_2 \quad \cdots \quad \vec{w}_N\,).$$

Let

$$\mathbb{U}^{\mathrm{T}}\mathbb{A}\mathbb{V} = \begin{pmatrix} \Sigma & 0 \\ 0 & 0 \end{pmatrix},$$

where $\mathbb{U}$ and $\mathbb{V}$ are $J\times J$ and $N\times N$ orthogonal matrices, respectively, and $\Sigma = \mathrm{diag}(\sigma_1,\ldots,\sigma_{\widetilde{N}})$ with $\sigma_1 \geqslant \sigma_2 \geqslant \cdots \geqslant \sigma_{\widetilde{N}}$ be the singular value decomposition of $\mathbb{A}$. Here, $\widetilde{N}$ is the rank of $\mathbb{A}$, i.e., the dimension of the snapshot set W, which would be less than N whenever the snapshot set is linearly dependent. It is well known [17] that if

$$\mathbb{U} = (\,\vec{\phi}_1 \quad \vec{\phi}_2 \quad \cdots \quad \vec{\phi}_J\,) \quad \text{and} \quad \mathbb{V} = (\,\vec{\psi}_1 \quad \vec{\psi}_2 \quad \cdots \quad \vec{\psi}_N\,),$$

then

$$\mathbb{A}\vec{\psi}_i = \sigma_i\vec{\phi}_i \quad \text{and} \quad \mathbb{A}^{\mathrm{T}}\vec{\phi}_i = \sigma_i\vec{\psi}_i \quad \text{for } i = 1,\ldots,\widetilde{N}$$

so that also

$$\mathbb{A}^{\mathrm{T}}\mathbb{A}\vec{\psi}_i = \sigma_i^2\vec{\psi}_i \quad \text{and} \quad \mathbb{A}\mathbb{A}^{\mathrm{T}}\vec{\phi}_i = \sigma_i^2\vec{\phi}_i \quad \text{for } i = 1,\ldots,\widetilde{N}$$

so that σ_i^2, $i = 1,\ldots,\widetilde{N}$, are the nonzero eigenvalues of $\mathbb{A}^{\mathrm{T}}\mathbb{A}$ (and also of $\mathbb{A}\mathbb{A}^{\mathrm{T}}$) arranged in nondecreasing order. Note that the matrix $\mathbb{C} = \mathbb{A}^{\mathrm{T}}\mathbb{A}$ is simply the correlation matrix for the set of snapshot vectors $W = \{\vec{w}_n\}_{n=1}^N$, i.e., we have that $\mathbb{C}_{mn} = \vec{w}_m^{\mathrm{T}}\vec{w}_n$.

In the reduced-order modeling context, given a set of snapshots $W = \{\vec{w}_n\}_{n=1}^N$ belonging to $\mathfrak{R}^J$, the POD reduced-basis of dimension $K \leqslant N < J$ is the set $\{\vec{\phi}_k\}_{k=1}^K$ of vectors also belonging to $\mathfrak{R}^J$ consisting of the first K left singular vectors of the snapshot matrix $\mathbb{A}$. Thus, one can determine the POD basis by computing the (partial) singular value decomposition of the $J\times N$ matrix $\mathbb{A}$. Alternately, one can compute the (partial) eigensystem $\{\sigma_k^2,\vec{\psi}_i\}_{i=1}^K$ of the $N\times N$ correlation matrix $\mathbb{C} = \mathbb{A}^{\mathrm{T}}\mathbb{A}$ and then set $\vec{\phi}_k = \frac{1}{\sigma_k}\mathbb{A}\vec{\psi}_k$, $k = 1,\ldots,K$.

The K-dimensional POD basis has the obvious property of orthonormality. It also has several other important properties which we now mention. Let $\{\vec{s}_k\}_{k=1}^K$ be an arbitrary set of K orthonormal vectors in $\mathfrak{R}^J$ and let $\Pi\vec{w}$ denote the projection of a vector $\vec{w}\in\mathfrak{R}^J$ onto the subspace spanned by that set. Further, let

$$\mathscr{E}(\vec{s}_1,\ldots,\vec{s}_K) = \sum_{n=1}^N |\vec{w}_n - \Pi\vec{w}_n|^2,$$

i.e., $\mathscr{E}$ is the sum of the squares of the error between each snapshot vector $\vec{w}_n$ and its projection $\Pi\vec{w}_n$ onto the span of $\{\vec{s}_k\}_{k=1}^K$. Then, it can be shown that

$$\begin{cases} \text{the POD basis } \{\vec{\phi}_k\}_{k=1}^K \text{ minimizes } \mathscr{E} \text{ over all possible} \\ K\text{-dimensional orthonormal sets in } \mathfrak{R}^J. \end{cases} \tag{4}$$

In fact, often the POD basis corresponding to a set of snapshots $W = \{\vec{w}_n\}_{n=1}^N$ is defined by (4) and then its relation to the singular value decomposition of the matrix $\mathbb{A}$ or to the eigenvalue decomposition of $\mathbb{A}^{\mathrm{T}}\mathbb{A}$ are derived properties. We note that $\mathscr{E}(\vec{\phi}_1,\ldots,\vec{\phi}_K)$ is referred to as the "POD energy" or "error in the POD basis". Also, it can be shown that

$$\mathscr{E}(\vec{\phi}_1,\ldots,\vec{\phi}_K) = \sum_{k=K+1}^{\widetilde{N}} \sigma_k^2, \tag{5}$$

i.e., the error in the POD basis is simply the sum of the squares of the singular values corresponding to the neglected POD modes.

Another property of the POD basis is given as follows:

$$\begin{cases} \text{the POD basis } \{\vec{\phi}_k\}_{k=1}^K \text{ solves the sequence of problems:} \\ \quad \text{for } k = 1,\ldots,K, \quad \max\limits_{\vec{s}_k\in\mathfrak{R}^J} \sum\limits_{n=1}^N \left(\vec{w}_n^{\mathrm{T}}\vec{s}_k\right)^2 \\ \text{subject to } |\vec{s}_k| = 1 \quad \text{and} \quad \vec{s}_j^{\mathrm{T}}\vec{s}_k = 0 \text{ for } j = 1,\ldots,k-1. \end{cases} \tag{6}$$

Again, (6) is often used to define the POD basis and then its relation to the singular value decomposition and (4) are noted as derived properties.

The singular values of the snapshot matrix may be used to determine a practical value for the dimension K of the POD basis. Indeed, it is a simple matter to show that is one wants the error in the POD basis to be less than some prescribed tolerance δ, i.e., that

$$\mathscr{E}(\vec{\phi}_1,\ldots,\vec{\phi}_K) \leqslant \delta,$$

then one need only

choose K to be the smallest integer such that

$$\frac{\sum_{k=1}^K \sigma_k^2}{\sum_{k=1}^{\widetilde{N}} \sigma_k^2} \geqslant 1-\delta.$$

The perceived usefulness of POD-based reduced-order modeling is derived from the observation that in many settings one finds that, even if δ is chosen to be very small, e.g., 0.01, one can still choose K to be relatively small, e.g., K is much smaller than $\widetilde{N}$ and often can be of order 10 or so.

For reduced-order modeling applications, the snapshot vectors are coefficient vectors in the expansion of the finite element approximation of the velocity field evaluated at different instants in time. Thus, to each snapshot vector $\vec{w}_n$, $n = 1,\ldots,N$, there corresponds a finite element function

$$\mathbf{w}_n(\mathbf{x}) = \sum_{j=1}^J w_{j,n}\boldsymbol{\xi}_j(\mathbf{x}) \in \mathbf{V}^h, \tag{7}$$

where $w_{j,n}$ denotes the jth component of the vector $\vec{w}_n$ and $\boldsymbol{\xi}_j(\mathbf{x})\in\mathbf{V}^h$ denotes the jth finite element basis function. One can define a POD basis with respect to functions instead of vectors, i.e., we could start with a snapshot set $\mathbf{W} = \{\mathbf{w}_n(\mathbf{x})\}_{n=1}^N$ consisting of finite element functions belonging to $\mathbf{V}^h$. Then, instead of (6), one could define the POD basis $\{\boldsymbol{\phi}_k(\mathbf{x})\in\mathbf{V}^h\}_{k=1}^K$ to be the solution of the sequence of problems: *for* $k = 1,\ldots,K$,

$$\max_{\mathbf{s}_k(\mathbf{x})\in\mathbf{V}^h} \sum_{n=1}^N (\mathbf{w}_n,\mathbf{s}_k)_0^2$$

subject to $\|\mathbf{s}_k(\mathbf{x})\|_0 = 1$ and $(\mathbf{s}_j,\mathbf{s}_k)_0 = 0$ for $j = 1,\ldots,k-1$. Equivalently, one could define the POD basis to be the solution of the problem: *minimize*

$$\mathscr{E}(\mathbf{s}_1,\ldots,\mathbf{s}_K)=\sum_{n=1}^{N}\|\mathbf{w}_n-\Pi\mathbf{w}_n\|_0^2$$

over all possible K-dimensional $L^2(\Omega)$-orthonormal sets $\{\mathbf{s}_k(\mathbf{x})\}_{k=1}^K$ in $\mathbf{V}^h$, where $\Pi\mathbf{w}_n$ is the $L^2(\Omega)$-projection of $\mathbf{w}_n$ onto the span of the functions $\{\mathbf{s}_k(\mathbf{x})\}_{k=1}^K$. Equivalently again, one can determine the POD basis by first solving the $N\times N$ eigenvalue problem: for $k=1,\ldots,\widetilde{N}$,

$$\mathbb{C}\vec{a}_k=\sigma_k^2\vec{a}_k,\quad |\vec{a}_k|=1,\quad \vec{a}_\ell^{\mathrm{T}}\vec{a}_k=0 \text{ if } k\neq\ell,$$
$$\text{and}\quad \sigma_k\geqslant\sigma_{k-1}>0, \tag{8}$$

then setting

$$\boldsymbol{\phi}_k(\mathbf{x})=\sum_{n=1}^{N}\frac{1}{\sigma_k}a_{k,n}\mathbf{w}_n(\mathbf{x})\quad\text{for } k=1,\ldots,K.$$

Here, we now have that the rank $\widetilde{N}\leqslant N$ correlation matrix $\mathbb{C}$ is defined by $\mathbb{C}_{mn}=(\mathbf{w}_m,\mathbf{w}_n)_0$, and $a_{k,n}$ is the nth component of the eigenvector $\vec{a}_k$. Note that in terms of the snapshot matrix $\mathbb{A}$ and the mass matrix $\mathbb{M}$ for the finite element basis, i.e., for $\mathbb{M}_{ij}=(\boldsymbol{\xi}_i,\boldsymbol{\xi}_j)_0$, we now have that $\mathbb{C}=\mathbb{A}^{\mathrm{T}}\mathbb{M}\mathbb{A}$. This fact allows us to again use the singular value decomposition to determine the POD basis function. To this end, let $\mathbb{M}=\mathbb{S}^{\mathrm{T}}\mathbb{S}$, where the $J\times J$ matrix $\mathbb{S}$ could be chosen to be a symmetric, positive definite square root of $\mathbb{M}$, i.e., $\mathbb{S}=\mathbb{M}^{1/2}$, or $\mathbb{S}$ could be a Cholesky factor, i.e., $\mathbb{S}^{\mathrm{T}}=\mathbb{L}$. Then, we let $\widetilde{\mathbb{A}}=\mathbb{S}\mathbb{A}$ so that $\mathbb{C}=\mathbb{A}^{\mathrm{T}}\mathbb{M}\mathbb{A}=\widetilde{\mathbb{A}}^{\mathrm{T}}\widetilde{\mathbb{A}}$ and therefore $\vec{a}_k$, $k=1,\ldots,K$, are the first K left singular vectors of $\widetilde{\mathbb{A}}$.

2.2. CVT reduced-order bases

Given a discrete set of snapshot vectors $W=\{\vec{w}_n\}_{n=1}^N$ belonging to $\mathfrak{R}^J$, a set $\{V_k\}_{k=1}^K$ is a tessellation of W if $\{V_k\}_{k=1}^K$ is a subdivision of W into disjoint, covering subsets, i.e., $V_k\subset W$ for $k=1,\ldots,K$, $V_k\cap V_i=\emptyset$ for $k\neq i$, and $\bigcup_{k=1}^K V_k=W$. Given a set of points $\{\vec{z}_k\}_{k=1}^K$ belonging to $\mathfrak{R}^J$ (but not necessarily to W), the *Voronoi region* corresponding to the point $\vec{z}_k$ is defined by

$$V_k=\{\vec{w}\in W:|\vec{w}-\vec{z}_k|\leqslant|\vec{w}-\vec{z}_i|\text{ for } i=1,\ldots,K,\ k\neq i\},$$

where equality holds only for $k<i$. Such a set $\{V_k\}_{k=1}^K$ is called a *Voronoi tessellation* or *Voronoi diagram* of W corresponding to the set of points $\{\vec{z}_k\}_{k=1}^K$. The points $\vec{z}_k$, $k=1,\ldots,K$, are called the generators of the Voronoi diagram $\{V_k\}_{k=1}^K$ of W.

Given a density function $\rho(\vec{y})\geqslant 0$, defined for $\vec{y}\in W$, the *mass centroid* $\vec{z}^*$ of any subset $V\subset W$ is defined by

$$\sum_{\vec{y}\in V}\rho(\vec{y})|\vec{y}-\vec{z}^*|^2=\inf_{\vec{z}\in V^*}\sum_{\vec{y}\in V}\rho(\vec{y})|\vec{y}-\vec{z}|^2,$$

where the sums extend over the points belonging to V. The set V^* can be taken to be V or it can be an even larger set such as all of $\mathfrak{R}^J$. In case $V^*=\mathfrak{R}^J$, $\vec{z}^*$ is the ordinary mean

$$\vec{z}^*=\frac{\sum_{\vec{y}\in V}\rho(\vec{y})\vec{y}}{\sum_{\vec{y}\in V}\rho(\vec{y})}.$$

In this case, $\vec{z}^*\notin W$ in general.

If $\{\vec{z}_k\}_{k=1}^K=$ the set of generating points of a Voronoi tessellation $\{\widehat{V}_k^K\}$ and $\{\vec{z}_k^*\}_{k=1}^K=$ the set of mass centroids of the Voronoi regions $\{\widehat{V}_k^K\}$, then, in general, $\vec{z}_k\neq\vec{z}_k^*$ for $k=1,\ldots,K$. If it so happens that $\vec{z}_k=\vec{z}_k^*$ for $k=1,\ldots,K$, we then refer to the Voronoi tessellation as being a *centroidal Voronoi tessellation* (CVT). CVTs of discrete sets are closely related to optimal k-means clusters [19,20,27,47–49] so that Voronoi regions and centroids can be referred to as clusters and cluster centers, respectively.

There are several algorithms known for constructing centroidal Voronoi tessellations of a given set. Lloyd's method is a deterministic algorithm which is the obvious iteration between computing Voronoi diagrams and mass centroids, i.e., a given set of generators is replaced in an iterative process by the mass centroids of the Voronoi regions corresponding to those generators. MacQueen's method is a very elegant probabilistic algorithm. Other probabilistic methods have been devised that may be viewed as generalization of both the MacQueen and Lloyd methods and that are amenable to efficient parallelization. See [6,23] for detailed descriptions of algorithms for constructing CVTs.

In the reduced-order modeling context, given a set of snapshots $W=\{\vec{w}_n\}_{n=1}^N$ belonging to $\mathfrak{R}^J$, the CVT reduced-basis of dimension $K\leqslant N<J$ is the set of generators $\{\vec{z}_k\}_{k=1}^K$, also belonging to $\mathfrak{R}^J$, of a CVT of the snapshot set.

Similar to POD, CVTs also have an optimization characterization. Let $\{\vec{z}_k\}_{k=1}^K$ denote an arbitrary set of K vectors in $\mathfrak{R}^J$ and let $\{V_k\}_{k=1}^K$ denote a tessellation of the snapshot set $\{\vec{w}_n\}_{n=1}^N$ into K disjoint, covering subsets. Note that, at the start, we do not require $\{V_k\}_{k=1}^K$ to be a Voronoi tessellation and also, we do not require any connection between the vectors in $\{\vec{z}_k\}_{k=1}^K$ and the sets in $\{V_k\}_{k=1}^K$. Let

$$\mathscr{F}(\vec{z}_1,\ldots,\vec{z}_K;V_1,\ldots,V_K)=\sum_{k=1}^{K}\sum_{\vec{w}_n\in V_k}\rho(\vec{w}_n)|\vec{w}_n-\vec{z}_k|^2 \tag{9}$$

and pose the problem: *minimize $\mathscr{F}(\cdot;\cdot)$ over all possible tessellations of the snapshot set $\{\vec{w}_n\}_{n=1}^N$ into K clusters and all possible sets of K points in $\mathfrak{R}^J$*, where all the points and clusters can be varied independently. The solution of this problem is a CVT, i.e., $\mathscr{F}(\cdot;\cdot)$ is minimized whenever $\{V_k\}_{k=1}^K$ is a Voronoi tessellation and the vectors in $\{\vec{z}_k\}_{k=1}^K$ are simultaneously the generators of that Voronoi tessellation and the centers of mass of the corresponding Voronoi clusters. Thus, (9) is referred to as the "CVT energy". Note that it is simply a variance measure of the clustering $\{V_k\}_{k=1}^K$ of the snapshot set $\{\vec{w}_n\}_{n=1}^N$.

Again, similar to POD, one can get an indication of what is a reasonable choice for K, the number of CVT clusters, that one should use. Clearly, as K increases, the CVT energy decreases. Also, it is clear that if $K=N$, i.e., the number of CVT generators is equal to the number of snapshots that are being clustered, that the CVT energy is zero. In general, if the snapshot vectors are randomly distributed, then the energy will decrease slowly as the number of generators increases. However, if he snapshot vectors

cluster well, then the decrease in the energy can be quite rapid, at least until enough generators are used so that the data is well clustered. Increasing K further will not effect a further large decrease in the POD energy. Thus, one can monitor the CVT energy (which is a computable quantity) for increasing values of K and when it ceases to decrease appreciably, one knows that one has found a reasonable value K for the number of clusters. This effect is known as the *elbowing effect* for the CVT energy that occurs when the data is amenable to clustering.

The perceived usefulness of CVT-based reduced-order modeling is derived from the observation that in many settings one finds that even for K relatively small, e.g., K much smaller than N, the snapshot set is well clustered into K clusters and any further increase in the value of K does not result in an appreciable reduction in the CVT energy.

We have defined CVTs based on a snapshot set consisting of vectors in $\Re^J$ that, in our context, are vectors of coefficients of a finite element function. One can also define CVTs based on a snapshot set $\mathbf{W} = \{\mathbf{w}_n(\mathbf{x})\}_{n=1}^N$ consisting of the functions themselves, where the functions $\mathbf{w}_n(\mathbf{x})$ are related to the vectors $\vec{w}_n$ through (7). Then, given a set $\{\mathbf{z}_k\}_{k=1}^K$ of K functions belonging to $\mathbf{V}^h$, we can define the Voronoi tessellation $\{\mathbf{V}_k\}_{k=1}^K$ of $\mathbf{W}$ into K subsets by

$$\mathbf{V}_k = \{\mathbf{w} \in \mathbf{W} : \|\mathbf{w} - \mathbf{z}_k\|_0 \leqslant \|\mathbf{w} - \mathbf{z}_i\|_0 \text{ for } i = 1, \ldots, K,\ k \neq i\}.$$

Given any subset $\mathbf{V} \subset \mathbf{W}$, we can also define the centroid of $\mathbf{V}$ with respect to a given density function $\rho(\mathbf{y})$ defined for $\mathbf{y} \in \mathbf{W}$, e.g.,

$$\mathbf{z}^* = \frac{\sum_{\mathbf{y}\in\mathbf{V}} \rho(\mathbf{y})\mathbf{y}}{\sum_{\mathbf{y}\in\mathbf{V}} \rho(\mathbf{y})}.$$

Note that, in general, $\mathbf{z}^* \notin \mathbf{V}$ but $\mathbf{z}^* \in \text{span}(\mathbf{V})$. Then, a CVT of the snapshot set $\mathbf{W} = \{\mathbf{w}_n(\mathbf{x})\}_{n=1}^N$ is again defined to be a set of K functions $\{\mathbf{z}_k\}_{k=1}^K$ belonging to $\mathbf{V}^h$ and a Voronoi tessellation $\{\mathbf{V}_k\}_{k=1}^K$ of $\mathbf{W}$ such that the functions $\mathbf{z}_k$ are simultaneously the generators of the Voronoi diagram and centers of mass of the corresponding Voronoi cell. It can be shown that such a CVT is a minimizer of the CVT energy

$$\mathscr{F}(\mathbf{z}_1, \ldots, \mathbf{z}_K; \mathbf{V}_1, \ldots, \mathbf{V}_K) = \sum_{k=1}^K \sum_{\mathbf{w}_n \in \mathbf{V}_k} \rho(\mathbf{w}_n)\|\mathbf{w}_n - \mathbf{z}_k\|_0^2.$$

Although in this paper we will choose the density function $\rho(\cdot) = 1$, the variable density capability of CVT-based reduced-order modeling could prove to be a useful tool, e.g., for assigning a different importance to each snapshot.

2.3. The POD and CVT reduced-order models

We now show how a POD or CVT basis is used to define a reduced-order model for the Navier–Stokes system. For the sake of brevity, we only discuss the case for which the snapshot set is viewed as a set of finite element coefficient vectors; the case for which the snapshot set is a set of finite element functions proceeds in an almost identical manner.

Let

$$\vec{\psi}_k = \begin{cases} \vec{\phi}_k & \text{for a POD reduced basis} \\ \vec{z}_k & \text{for a CVT reduced basis} \end{cases} \quad \text{for } k = 1, \ldots, K,$$

where $\{\vec{\phi}_k\}_{k=1}^K$ is a K-dimensional POD basis corresponding to the snapshot set $\{\vec{w}_n\}_{n=1}^N$ and $\{\vec{z}_k\}_{k=1}^K$ is the set of generators of a CVT of that snapshot set. Corresponding to each $\vec{\psi}_k$, $k = 1, \ldots, K$, is a finite element function

$$\boldsymbol{\psi}_k = \sum_{j=1}^J \psi_{j,k}\boldsymbol{\xi}_j(\mathbf{x}) \in \mathbf{V}^h,$$

where $\psi_{j,k}$ denotes the j-component of $\vec{\psi}_k$. Let

$U_K = \text{span}\{\boldsymbol{\psi}_k\}_{i=1}^K \subset \mathbf{V}^h$.

As will be illustrated in Section 3.1, the reduced-basis functions satisfy homogeneous boundary conditions. We then seek a reduced-basis approximation of the velocity field of the form

$$\mathbf{u}^K(t, \cdot) = \mathbf{u}_p^K(t, \cdot) + \mathbf{u}_h^K(t, \cdot),$$

where $\mathbf{u}_h^K(t, \cdot) \in U_K$ satisfies homogeneous boundary conditions and $\mathbf{u}_p^K(t, \cdot) \in \mathbf{V}^h$ is a particular finite element function chosen to satisfy the boundary conditions (again, see Section 3.1). We then determine $\mathbf{u}_h^K(t, \cdot)$ from the discrete problem

$$\begin{cases} \displaystyle\int_\Omega \mathbf{u}_t^K \cdot \mathbf{v}\,\mathrm{d}\Omega + 2\nu \int_\Omega D(\mathbf{u}^K) : D(\mathbf{v})\,\mathrm{d}\Omega \\ \quad \displaystyle + \int_\Omega (\mathbf{u}^K \cdot \nabla)\mathbf{u}^K \cdot \mathbf{v}\,\mathrm{d}\Omega = \int_\Omega \mathbf{f} \cdot \mathbf{v}\,\mathrm{d}\Omega \quad \forall \mathbf{v} \in \mathbf{U}_K \\ (\mathbf{u}(0, \mathbf{x}), \mathbf{v}) = (\mathbf{u}_0(\mathbf{x}), \mathbf{v}) \quad \forall \mathbf{v} \in \mathbf{U}_K. \end{cases} \tag{10}$$

Note that the pressure does not appear in these equations and that the continuity equation in (2) is absent. This is because, as a result of the manner in which the reduced-basis functions are constructed, they are automatically discretely divergence free.

Because of the special form chosen for the inhomogeneous data $\mathbf{b}$ in our computational example, we will be able to choose the particular solution $\mathbf{u}_p^K(t, \mathbf{x})$ to have the form $\mathbf{u}_p^K = \alpha_0(t)\boldsymbol{\psi}_0(\mathbf{x})$, where $\boldsymbol{\psi}_0$ is a steady state finite element solution of the discretized Navier–Stokes equations and $\alpha_0(t)$ is chosen so that $\mathbf{u}_p^K(t, \mathbf{x})$ satisfies the time-dependent boundary condition. Then, the reduced-basis approximation of the velocity takes the form

$$\mathbf{u}^K(t) = \sum_{k=0}^K \alpha_k(t)\boldsymbol{\psi}_k$$

and (10) may be expressed as

$$\begin{cases} \displaystyle\sum_{k=0}^K \tfrac{\mathrm{d}}{\mathrm{d}t}\alpha_k(t)(\boldsymbol{\psi}_k, \boldsymbol{\psi}_\ell) + 2\nu \sum_{k=0}^K \alpha_k(t)(D(\boldsymbol{\psi}_k), D(\boldsymbol{\psi}_\ell) \\ \quad \displaystyle + \left(\sum_{m=0}^K \alpha_m(t)\boldsymbol{\psi}_m \cdot \nabla \sum_{k=0}^K \alpha_k(t)\boldsymbol{\psi}_k\boldsymbol{\psi}_\ell\right) = (\mathbf{f}, \boldsymbol{\psi}_\ell) \\ \displaystyle\sum_{k=0}^K \alpha_k(0)(\boldsymbol{\psi}_k, \boldsymbol{\psi}_\ell) = (\mathbf{u}_0, \boldsymbol{\psi}_\ell) \end{cases}$$

for $\ell = 1, \ldots, K$. Equivalently, we have the system of nonlinear ordinary differential equations that determine the coefficient functions $\{\alpha_k(t)\}_{k=1}^K$:

$$\begin{cases} \mathbb{G}\frac{\mathrm{d}}{\mathrm{d}t}\vec{\alpha}(t) + \mathbb{K}\vec{\alpha}(t) + \alpha_0(t)\mathbb{H}\vec{\alpha}(t) + (\vec{\alpha}(t))^{\mathrm{T}}\mathbb{N}\vec{\alpha}(t) = \vec{f}(t), \\ \mathbb{M}\vec{\alpha}(0) = \vec{\alpha}_0, \end{cases} \tag{11}$$

where the Gram matrix $\mathbb{G}$, stiffness matrix $\mathbb{K}$, convection matrix $\mathbb{H}$, convection tensor $\mathbb{N}$, and solution vector $\vec{\alpha}(t)$ are respectively given by

$$\mathbb{G}_{\ell k} = \int_\Omega \boldsymbol{\psi}_k \cdot \boldsymbol{\psi}_\ell \,\mathrm{d}\Omega, \quad \mathbb{K}_{\ell k} = 2\nu \int_\Omega D(\boldsymbol{\psi}_k) : D(\boldsymbol{\psi}_\ell)\,\mathrm{d}\Omega,$$

$$\mathbb{H}_{\ell k} = \int_\Omega (\boldsymbol{\psi}_0 \cdot \nabla\boldsymbol{\psi}_k \cdot \boldsymbol{\psi}_\ell + \boldsymbol{\psi}_k \cdot \nabla\boldsymbol{\psi}_0 \cdot \boldsymbol{\psi}_\ell)\,\mathrm{d}\Omega$$

$$\mathbb{N}_{\ell m k} = \int_\Omega (\boldsymbol{\psi}_m \cdot \nabla)\boldsymbol{\psi}_k \cdot \boldsymbol{\psi}_\ell \,\mathrm{d}\Omega \quad \text{and} \quad (\vec{\alpha})_k = \alpha_k(t)$$

for $k, \ell, m = 1, \ldots, K$, and the forcing vector $\vec{f}(t)$ and initial data vector $\vec{\alpha}_0$ are respectively given by

$$(\vec{f})_\ell = \int_\Omega \mathbf{f} \cdot \boldsymbol{\psi}_\ell \,\mathrm{d}\Omega - \frac{\mathrm{d}}{\mathrm{d}t}\alpha_0(t) \int_\Omega \boldsymbol{\psi}_0 \cdot \boldsymbol{\psi}_\ell \,\mathrm{d}\Omega - 2\nu\alpha_0(t) \int_\Omega \nabla\boldsymbol{\psi}_0 : \nabla\boldsymbol{\psi}_\ell \,\mathrm{d}\Omega - \alpha_0^2(t) \int_\Omega \boldsymbol{\psi}_0 \cdot \nabla\boldsymbol{\psi}_0 \cdot \boldsymbol{\psi}_\ell \,\mathrm{d}\Omega$$

and

$$(\vec{\alpha}_0)_\ell = \int_\Omega \mathbf{u}_0 \cdot \boldsymbol{\psi}_\ell \,\mathrm{d}\Omega - \int_\Omega \alpha_0(0)\boldsymbol{\psi}_0 \cdot \boldsymbol{\psi}_\ell \,\mathrm{d}\Omega$$

for $\ell = 1, \ldots, K$. Note that all of these matrices and tensors are full; however, since K will be chosen small (see Section 3.3), this does not cause any computational inefficiencies. Another important observation is that matrices $\mathbb{G}$, $\mathbb{K}$, and $\mathbb{H}$ and the tensor $\mathbb{N}$ depend only on the reduced-basis functions $\{\boldsymbol{\psi}_k\}_{k=1}^K$ so that they may be all pre-computed; thus, no finite element-type assembly is required to define the system (11).

2.4. The error in a reduced-order solution

At any given time t, we define the "error" $E(t)$ in a reduced-order solution, be it of the POD or CVT type, to be the $\mathbf{L}^2(\Omega)$-norm of the difference between the full finite element solution and the reduced-order solution, i.e.,

$$E(t) = \left(\int_\Omega (\mathbf{u}_{fe} - \mathbf{u}^K)^2 \,\mathrm{d}\Omega \right)^{1/2}, \tag{12}$$

where $\mathbf{u}_{fe}$ denotes the approximate velocity field determined using the full finite element simulation code and $\mathbf{u}^K$ denotes the approximate velocity field determined by either a POD or CVT reduced-order model. Also of interest is the space-time error

$$E_T = \left(\int_0^{\mathrm{T}} E^2(t)\,\mathrm{d}t \right)^{1/2} = \left(\int_0^{\mathrm{T}} \int_\Omega (\mathbf{u}_{fe} - \mathbf{u}^K)^2 \,\mathrm{d}\Omega\,\mathrm{d}t \right)^{1/2}. \tag{13}$$

It is important to note that there are two contributions to these "errors." One is due to the fact that the reduced-order model does not exactly reproduce the information contained in the snapshot set. The other contribution to the errors is due to the fact that the snapshot set itself cannot exactly represent the full finite element solution.

Recall that both the POD and CVT reduced bases are determined from a set of snapshots and that those bases are designed so that the information contained in the snapshot set can be captured by a reduced basis of dimension K much smaller than the cardinality N of the snapshot set, i.e., anything in the span of the snapshot set can be approximated well by a very low-dimensional reduced basis. Thus, a reduced-order model, be it of POD or CVT type, cannot possibly do better than what information it is given, i.e., the information contained in the snapshot set.

Even if a POD or CVT reduced-order model could exactly capture the information contained in the snapshot set, the errors (12) and (13) would not vanish because the snapshot set itself cannot exactly capture all the information contained in a full finite element solution. In the first place, a snapshot cannot exactly represent even the finite element solutions used in its construction because that set consists of a time-sampling of those solutions. More important, a snapshot set is determined using finite element solutions for system data, e.g., boundary conditions, that are different from those used in the simulations carried out using a reduced-order model. Thus, one relies on the snapshot set being able to represent solutions different from those that were used in its construction.

It has been demonstrated many times in the literature that POD-based reduced-order models are excellent at approximating the information contained in the snapshot set used in their construction; the computational experiments reported in Section 3 provide another such example. Thus, the real key to designing reduced-order models of either the POD or CVT type is to construct a snapshot set that contains sufficient information so that the full finite element solutions of the problems to be solved using the reduced-order model can be approximated well by some member of the span of the snapshot set. The POD or CVT reduced-order model merely does a good job of approximating that member in a reduced-space of much smaller dimension.

3. Computational experiments

To compare the use, efficiency, and accuracy of the POD and CVT-based reduced-order modeling techniques, one example is considered, which we denote the *T-cell problem.* This is an incompressible, viscous flow problem having boundary conditions that include an inflow shape function containing a multiplicative, time-varying parameter γ that controls the strength of the inflow.

The T-shaped region Ω is sketched in Fig. 1; its boundary $\partial\Omega$ is made up of $\Gamma_N = \Gamma_{\mathrm{o}}$ and $\Gamma_D = \Gamma_{\mathrm{i}} \cap \Gamma_d$ (see Fig. 1), where $\Gamma_{\mathrm{i}} = \{x = 0; 0.5 < y < 1.0\}$ and $\Gamma_{\mathrm{o}} = \{x =$

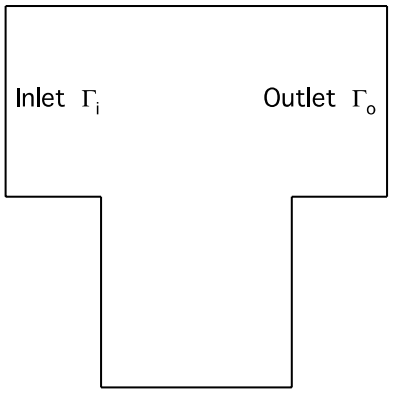

Fig. 1. The flow region for the T-cell example.

$1; 0.5 < y < 1.0\}$ are the inflow and outflow parts of the boundary, respectively, and $\Gamma_d = \partial\Omega \setminus (\overline{\Gamma}_i \cup \overline{\Gamma}_o)$. The governing equations for two dimensional incompressible, viscous flow in the T-shaped region are the Navier–Stokes system (1) with the data $\mathbf{f} = \mathbf{0}$ in Ω and

$$\mathbf{b} = \begin{cases} \mathbf{0} & \text{on } (0,T) \times \Gamma_d, \\ \begin{pmatrix} 100\gamma(t)(1-y)(0.5-y) \\ 0 \end{pmatrix} & \text{on } (0,T) \times \Gamma_i. \end{cases} \tag{14}$$

In (14), $\gamma(t)$ is a parameter function that determines the strength of the parabolic inflow velocity profile.

The $P2$–$P1$ Taylor-Hood finite element pair on a grid of 4961 nodes is used in the discrete weak formulation (2) to obtain accurate Galerkin-mixed method finite element approximations of the solutions of (1) with data given in (14); the time derivative is discretized by a backward-Euler method. Finite element solutions are used for the generation of snapshots and later for comparison with POD and CVT-based reduced-order solutions.

For the generation of snapshots, we will also solve, by the finite element method, stationary versions of our governing system for which the time derivative term and the initial condition in (1) are omitted and γ in (14) is chosen independent of t.

3.1. Generating snapshots

For all simulations involved in the snapshot generation process, we choose $\nu = 1$. We use the following procedure to determine a set of snapshot vectors. First, the finite element approximation to the stationary version of (1) and (14) with $\gamma = 1$ is obtained. Using that steady state solution as the initial data $\mathbf{u}_0$ in (1) and using $\gamma = 5$ for $0 < t < T/2 = 0.025$ and $\gamma = 1$ for $0.025 < t < T = 0.05$ in the boundary condition data (14), we then determine a finite element approximation $\sum_{j=1}^{J} w_j(t)\boldsymbol{\xi}_j(\mathbf{x})$ of the solution of (1) and (14), where J denotes the dimension of the finite element space used for the velocity and $\{\boldsymbol{\xi}_j\}_{j=1}^{J}$ denotes the basis functions for that space. This is the flow we use to generate the snapshots; it can be viewed as one for which the steady state solution for $\gamma = 1$ is suddenly jolted, at $t = 0$, by increasing the value of γ to five, and jolted again at $t = T/2 = 0.025$ by decreasing the value of γ back to one. The 500 snapshot vectors

$$\vec{w}_n = \begin{pmatrix} w_1(t_n) \\ w_2(t_n) \\ \vdots \\ w_J(t_n) \end{pmatrix}, \quad n = 1, \ldots, N = 500 \tag{15}$$

are then determined by evaluating the solution of this impulsively started problem at 500 equally spaced time values t_n, $n = 1, \ldots, 500$, ranging from $t = 0$ to $t = T = 0.05$. Note that the time interval used for sampling snapshots is a multiple of the time interval used for the time discretization of Navier–Stokes system (3). The snapshot vectors $\{\vec{w}_n\}_{n=1}^{N}$ correspond to the snapshot functions

$$\mathbf{w}_n(\mathbf{x}) = \sum_{j=1}^{J} w_j(t_n)\boldsymbol{\xi}_j(\mathbf{x}) \quad \text{for } n = 1, \ldots, N = 500.$$

For subsequent use, it is convenient to modify the 500 snapshots so that they satisfy homogeneous boundary conditions. To this end, we first obtain the reference finite element approximation $\mathbf{v}(\mathbf{x}) = \sum_{j=1}^{J} v_j \boldsymbol{\xi}_j(\mathbf{x})$ of the stationary version of (1) with $\gamma = 3$ in (14). We then modify the first 250 snapshots by

$$\mathbf{w}_n \leftarrow \left(\mathbf{w}_n - \frac{5}{3}\mathbf{v}\right) \quad \text{or} \quad \vec{w}_n \leftarrow \left(\vec{w}_n - \frac{5}{3}\vec{v}\right)$$
$$\text{for } n = 1, \ldots, 250$$

and, in the same way, the second 250 snapshots are modified

$$\mathbf{w}_n \leftarrow \left(\mathbf{w}_n - \frac{1}{3}\mathbf{v}\right) \quad \text{or} \quad \vec{w}_n \leftarrow \left(\vec{w}_n - \frac{1}{3}\vec{v}\right)$$
$$\text{for } n = 251, \ldots, 500,$$

where $(\vec{v})_j = v_j$ for $j = 1, \ldots, J$. In this way, all the snapshots satisfy homogeneous boundary conditions.

3.2. POD reduced bases

POD reduced bases corresponding to the snapshot set $\{\vec{w}_n\}_{n=1}^{N}$ are determined as described in Section 2.1. Note that since the inhomogeneous boundary conditions have been "subtracted away" from the snapshot vectors, each POD basis function satisfies zero Dirichlet boundary condition at the inlet; in the interior of the flow domain, each basis function satisfies the discretized continuity equation, i.e., it is discretely solenoidal. Also, note that the elements of a K-dimensional POD basis constitute the first K elements of all POD bases of dimension greater that K. For the snapshot set determined as described in Section 3.1, the eight-dimensional POD basis functions are displayed in Fig. 2.

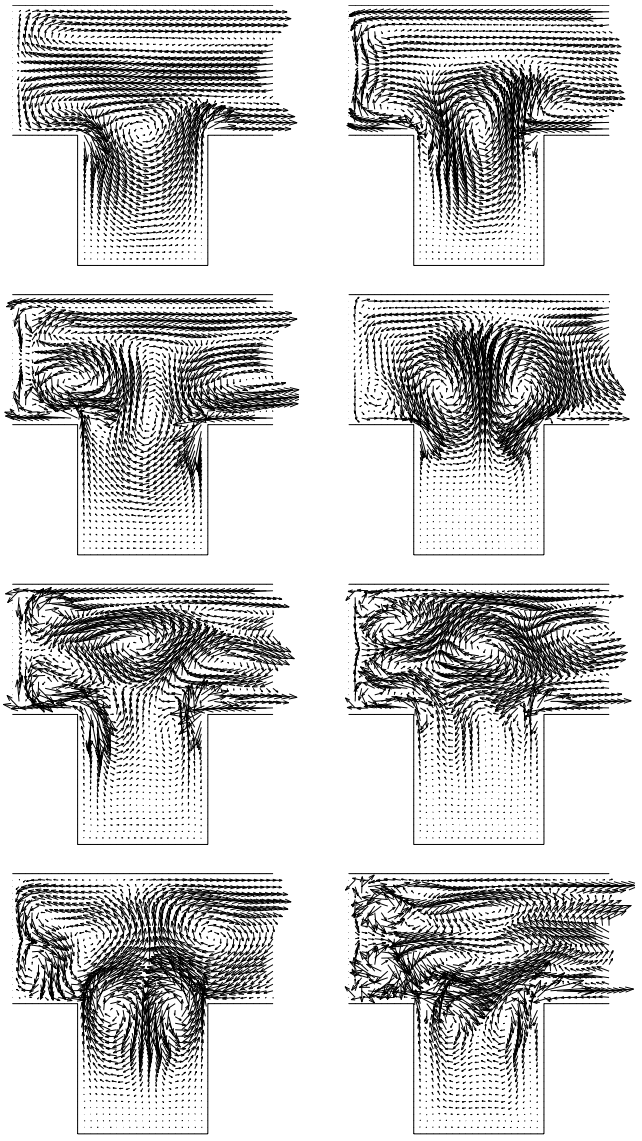

Fig. 2. The POD reduced basis of cardinality 8 for the T-cell problem.

For the snapshot set determined as described in Section 3.1, the first 16 singular values of the corresponding snapshot matrix are listed in Table 1. We see that the singular values decrease rapidly indicating that, for the T-cell problem we consider here, a "small-dimensional" POD basis can capture most of the information contained in the snapshot set. This behavior cannot, of course, be universal, but it has been observed in many other examples.

Table 1
The first 16 singular values of the snapshot matrix for the T-cell problem

1	2.7234	5	0.0866	9	0.0077	13	0.0014
2	0.6704	6	0.0349	10	0.0060	14	0.0010
3	0.2612	7	0.0168	11	0.0029	15	0.0008
4	0.1197	8	0.0151	12	0.0023	16	0.0004

3.3. CVT reduced bases

CVT reduced bases corresponding to the snapshot set $\{\vec{w}_n\}_{n=1}^{N}$ are determined as described in Section 2.2. For the snapshot set determined as described in Section 3.1, the eight-dimensional CVT basis functions are displayed in Fig. 3. As was the case for the POD basis functions, the CVT basis functions are discretely solenoidal and satisfy zero Dirichlet boundary condition at the inlet. However, unlike the POD basis functions, the CVT basis is not built by augmenting a CVT basis of smaller cardinality; in general, one observes that elements of two CVT bases generated from the same snapshot set but having different cardinalities seem to differ significantly.

We examine what happens to the CVT energy as the number of generators is increased. For the snapshot set determined as described in Section 3.1, the result is given in Table 2 and Fig. 4. The elbowing effect is evident. For

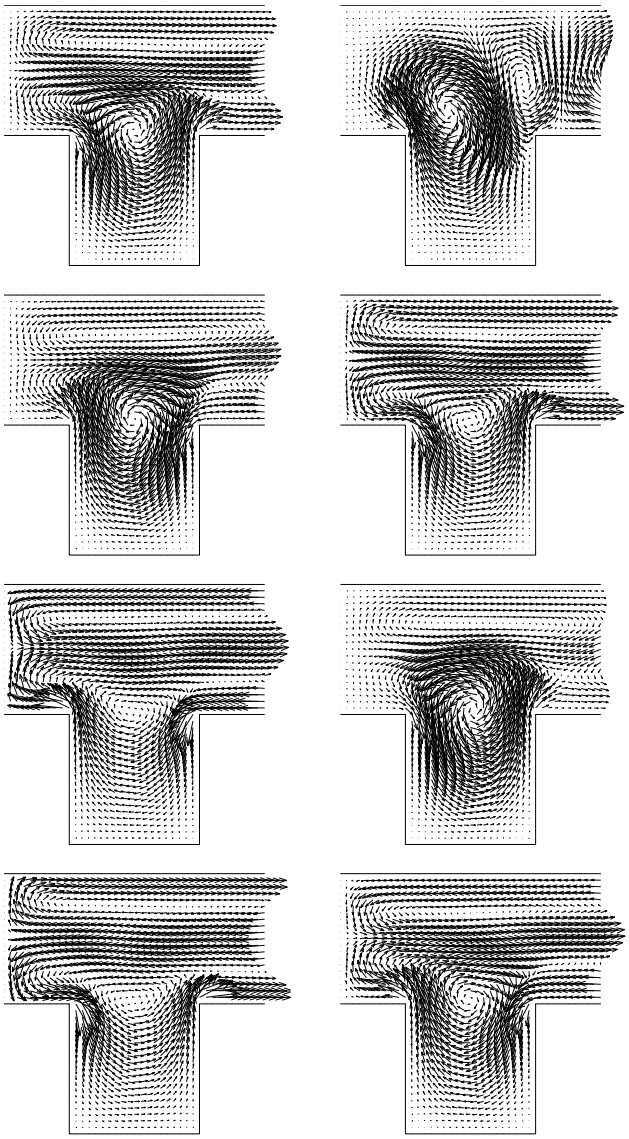

Fig. 3. The CVT basis of cardinality 8 for the T-cell problem.

Table 2
The CVT energy vs. the number of CVT generators for the T-cell problem

4	9837	8	2879	12	1442	16	833
5	6250	9	2254	13	1241	17	751
6	4857	10	1906	14	1041	18	669
7	3524	11	1697	15	936	19	626
						20	584

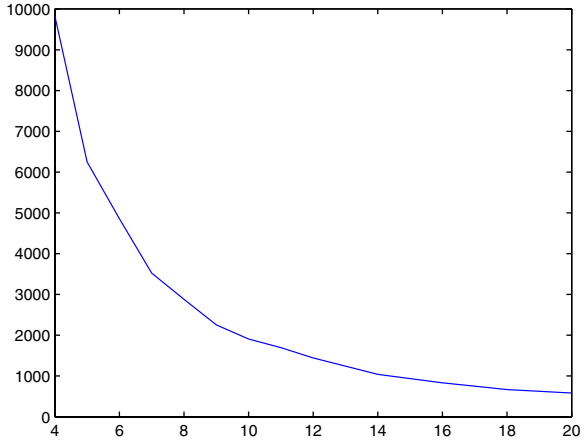

Fig. 4. The CVT energy vs. the number of CVT generators for the T-cell problem.

very few generators, increasing the number of generators effects a large decrease in the CVT energy, but as the number of generators increase, the reduction in the energy becomes much smaller. This is an indication that the snapshot set for the T-cell problem clusters well. Again, this phenomenon has been observed in many other settings, but is not universal.

It is interesting to examine some details, given in Table 3, about the CVT clustering of the snapshot set. That table is for an eight-dimensional CVT clustering of the snapshot set determined as described in Section 3.1. For each of the eight clusters, the number of elements in each cluster, the indices n of the snapshots $\vec{w}_n$ that are members of the cluster, and the contribution to the CVT energy of the cluster are given, as is the total CVT energy. In the interest of clarity, we have renumbered the cluster indices so that the cluster numbers in Table 3 do not correspond to the ordering implicit in Fig. 3. Note that, except for the fifth cluster, the clusters are formed exactly from a sequence of data points at neighboring times. This is not surprising since clusters should contain snapshots that are, in some sense, alike so that all elements in the cluster can be represented well by the cluster generator, i.e., by the mean of the cluster members. The two smallest clusters, the first and the sixth, correspond to the initial transient, and the "shock" to the system that occurs after step 250 when the system parameter α is changed abruptly. Since the solution is changing rapidly at those times, it makes sense that the clusters are small, and yet still contribute a sizable amount to the CVT energy. Also, the exceptional fifth cluster includes snapshots from two different time intervals that correspond to the relatively quiescent "tails" of the two evolution processes; again, it is not surprising that those two sets of snapshots are assigned to the same cluster by the CVT process.

Table 3
Cluster statistics for a CVT clustering of cardinality 8 for the T-cell problem

Cluster number	Cardinality of cluster	Snapshot indices	Cluster energy contribution
1	5	[1, 5]	281.1
2	10	[6, 15]	247.5
3	18	[16, 33]	236.4
4	41	[34, 74]	215.3
5	369	[75, 250] + [308, 500]	532.1
6	7	[251, 257]	523.3
7	15	[258, 272]	428.3
8	35	[273, 307]	420.1
Total	500	[1, 500]	3212.5

3.4. Determining POD and CVT reduced-order approximations

Given a boundary forcing function $\gamma(t)$, we use the K-dimensional system of nonlinear ordinary differential equations (11) to determine reduced-order solutions of the Navier–Stokes system in the T-cell configuration. Approximations of solutions of the system of ordinary differential equation (11) are determined using a fourth order Adams–Moulton method.

The matrices and tensor appearing in the system (11) depend only on the choice of reduced basis, so that once a POD or CVT basis is determined as described in Sections 3.2 and 3.3, respectively, they can be computed and used over and over again for different choices of $\gamma(t)$.

In Section 2.3, the reference solution $\boldsymbol{\psi}_0(\mathbf{x})$ was used to determine the particular solution $\mathbf{u}_p^K$ that satisfies the inhomogeneous boundary condition along Γ_D. In the context of the T-cell problem, this reference solution is chosen to be the steady state solution of the Navier–Stokes system with $\gamma = 3$, i.e., the function $\mathbf{v}$ of Section 3.1. Then, we choose

$$\alpha_0(t) = \gamma(t)/3 \tag{16}$$

so that $\alpha_0(t)\boldsymbol{\psi}_0(\mathbf{x}) = \alpha_0(t)\mathbf{v}(\mathbf{x})$ satisfies the inhomogeneous boundary condition along the inlet Γ_i of the T-cell problem. The function $\alpha_0(t)$ given by (16) is used in the system (11). In addition, we choose $\mathbf{f}(\mathbf{x}, \mathbf{t}) = \mathbf{0}$.

The specification of the system (11) is completed by choosing the boundary velocity forcing function $\gamma(t)$ and an initial condition $\mathbf{u}_0(\mathbf{x})$. To illustrate the use and effectiveness of the low-dimensional POD and CVT-based reduced-order models, we employ several shapes for the inlet velocity forcing factor $\gamma(t)$ and several different choices for the dimension of the reduced bases. The initial condition $\mathbf{u}_0(\mathbf{x})$ is chosen to be a steady state solution of the Navier–Stokes system that is compatible with the boundary data $\gamma(t)$.

3.5. Boundary velocity forcing functions and initial data

The specific choices used for $\gamma(t)$ in the reduced-order simulation using the system (11) are given as follows.

- *Case* 1 – a hat function with respect to time (solid curve in the left of Fig. 5):

$$\gamma(t) = \begin{cases} \frac{400}{3}t + 1 & \text{for } 0 \leqslant t \leqslant 0.03, \\ -\frac{400}{3}t + 9 & \text{for } 0.03 \leqslant t \leqslant 0.06. \end{cases}$$

- *Case* 2, 3, and 4 – sinusoidal functions with respect to time:

$$\gamma(t) = 3 + 2\sin(2a\pi t) \quad \text{for } 0 \leqslant t \leqslant 0.06,$$

where $a = 10$ for Case 2 (dashed curve in the left of Fig. 5), $a = 25$ for Case 3 (dotted curve in the left of Fig. 5), and $a = 50$ for Case 4 (dotted curve in the middle of Fig. 5).

- *Case* 5 – a "general" function with respect to time (dot-dashed curve in the middle of Fig. 5):

$$\gamma(t) = 3 + 2\cos(18\pi t)\sin(70\pi(t + 10)) \quad \text{for } 0 \leqslant t \leqslant 0.06.$$

- *Case* 6: a hat function with respect to time (solid curve in the right of Fig. 5):

$$\gamma(t) = \begin{cases} 180t + 1 & \text{for } 0 \leqslant t \leqslant 0.05, \\ -180t + 19 & \text{for } 0.05 \leqslant t \leqslant 0.1. \end{cases}$$

- *Case* 7: a "general" function with respect to time (dot-dashed curve in the right of Fig. 5):

$$\gamma(t) = 6 - 5\cos(9\pi t)\sin(90t + b) \quad \text{for } 0 \leqslant t \leqslant 0.1,$$

where $b = \arcsin(3/5)$.

The initial data $\mathbf{u}_0(\mathbf{x})$ is chosen to be the steady state solution of the Navier–Stokes system for $\gamma = 1$ for Cases 1 and 6 and $\gamma = 3$ for the other five cases. These choices insure that the initial and boundary data for each case are compatible.

Some remarks about the seven boundary forcing functions $\gamma(t)$ used in the computational experiments are in order. First, for all seven cases, that function is completely different from the step function used to generate the snapshots. This is, of course, how one wants to use a reduced-order model: generate a reduced-basis using snapshots determined from finite element simulations for some specific choice of data for the Navier–Stokes system, and then solve the reduced-order model for a variety of different data. Second, we note that Cases 1–5 are mostly of an "interpolatory" nature. In the first place, for those cases, we have that $1 \leqslant \gamma(t) \leqslant 5$ which is exactly the same bounds satisfied by the function $\gamma(t)$ used to generate the snapshots. In the second place, for Cases 1–5, the reduced-order simulations are carried out over the time interval $[0, 0.06]$ which is only 20% larger than the time interval $[0, 0.05]$ used to generate the snapshots. Cases 6 and 7 are much more "extrapolatory" in nature. First, we have that $1 \leqslant \gamma(t) \leqslant 10$ so that the reduced-order simulations see values of γ that are twice as large as what is seen during the generation of snapshots. Second, the time interval used for the reduced-order simulations is $[0, 0.1]$ which is twice as large as that used for the generation of snapshots.

3.6. Computational results

For all seven cases, full finite element solutions employing thousands of unknowns are determined so that they may be compared to the POD and CVT reduced-order solutions for the same data. Specifically, for the comparisons, we use the measures $E(t)$ and E_T defined in (12) and (13), respectively.

In Figs. 6–12, plots of $E(t)$ vs. the time t are provided for each of the seven test problems described in Section 3.5. The top of row of each figure is for POD-based reduced-order modeling; plots are provided for eight different choices for K, the dimension of the POD basis. The values of K used range from 4 to 16. The middle row of each figure provides similar information for CVT-based reduced-order modeling. Thus, the top two rows of each figure can be used to study the performance of each reduced-order model as the dimension of the reduced-bases is increased. The bottom row of each figure provides some selected plots of $E(t)$ for both the POD and CVT-based reduced-order

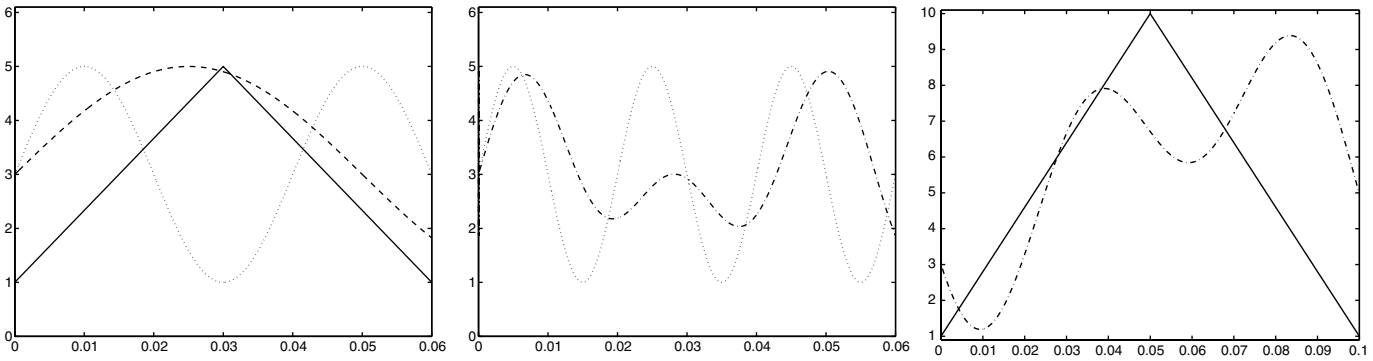

Fig. 5. The seven choices for the inflow forcing function $\gamma(t)$ used in the reduced-order simulations plotted vs. the time t.

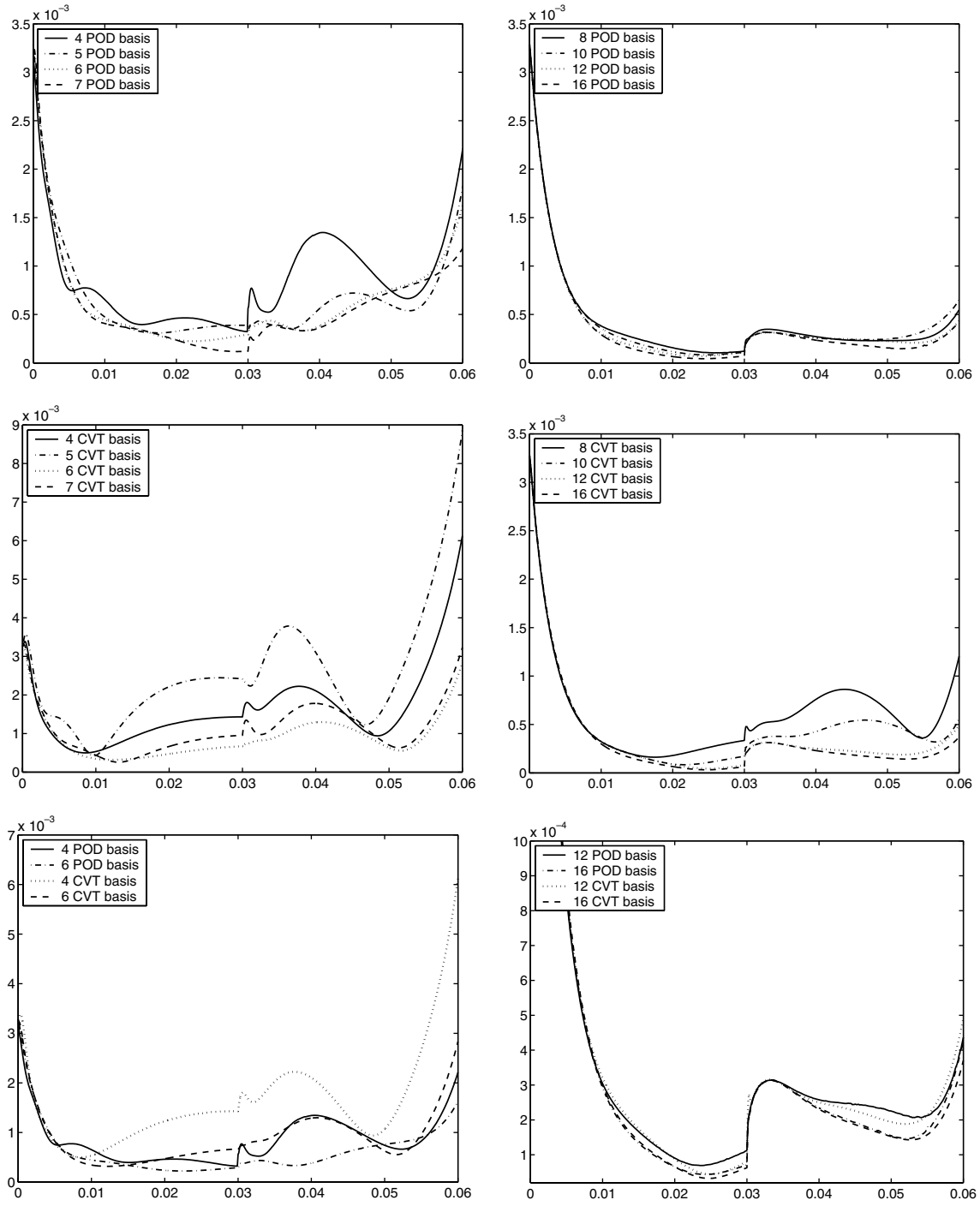

Fig. 6. $E(t)$ for the POD and CVT-based reduced-order models vs. time t for Case 1.

solutions and can be used to compare the relative performance of the two approaches.

An examination of Figs. 6–12 shows, at least for the T-cell example we consider, that very low-dimensional POD and CVT-based reduced-order models are both quite effective at approximating full finite element solutions; one sees that even for bases of dimension less than 10, the "error" $E(t)$ is small. One also sees that if the dimension K of the basis is about 12, increasing that dimension effects very little improvement in the performance of the POD and CVT reduced-order models. This is an indication that POD and CVT reduced-order models of dimension about 12 already effectively capture all of the information contained in the snapshot set that, in our example, has cardinality 500. This conclusion can also be inferred from Table 4 where the space-time "error" E_T vs. K is listed. We see that there is almost no reduction in this error if K is increased from 12 to 16.

Comparing Figs. 6–10 and Figs. 11, 12 that respectively correspond to "interpolatory" and "extrapolatory"

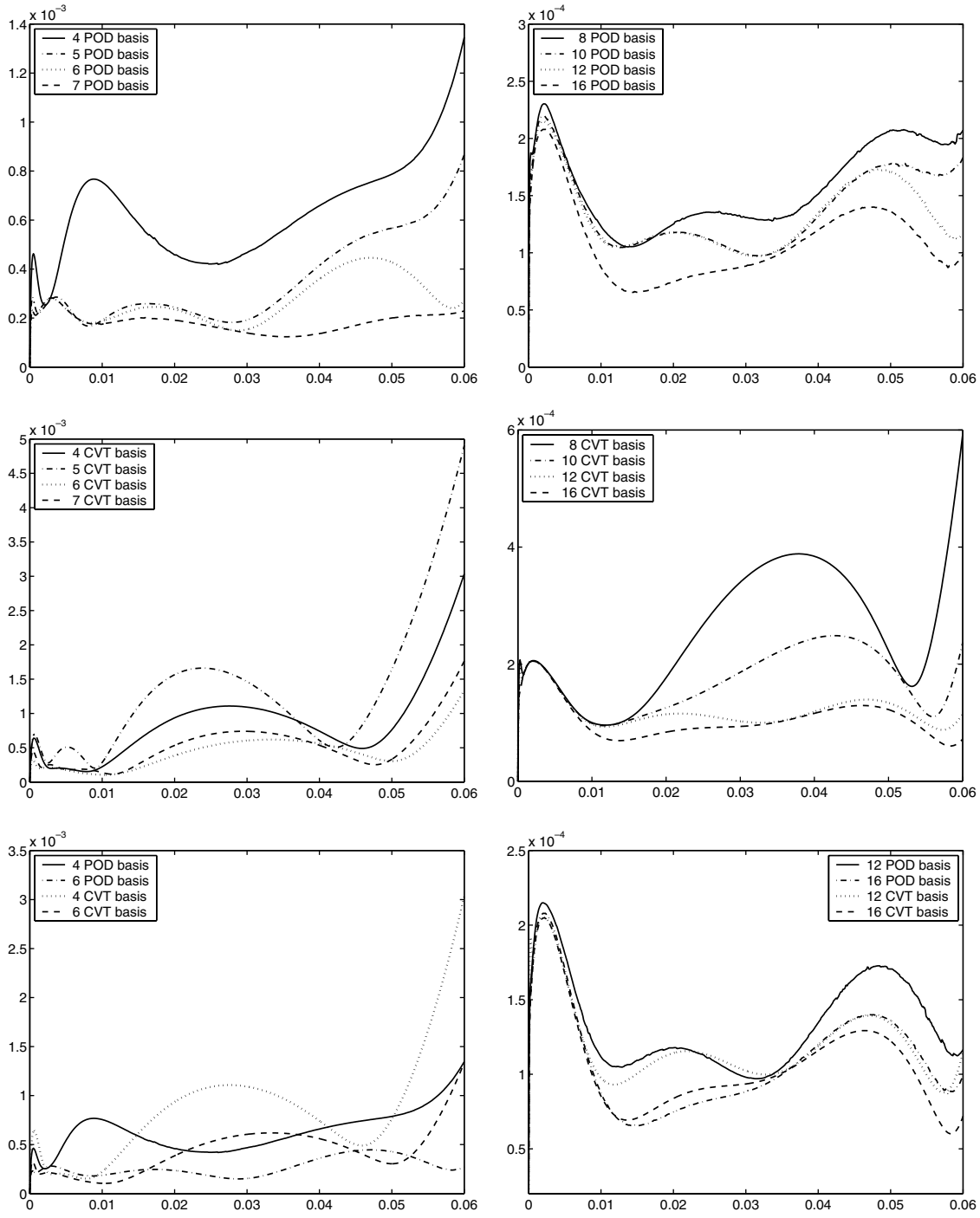

Fig. 7. $E(t)$ for the POD and CVT-based reduced-order models vs. time t for Case 2.

uses of the reduced-order models, we see that the "errors" in the latter case are not much worse that those for the former, although there is some deterioration evident in the performance of the reduced-order models near the end of the extrapolated simulation time interval $[0, 0.1]$.

Finally, Figs. 6–10 and Table 4 show that, as far as the ability of POD and CVT-based reduced-order models to approximate full finite element solutions, there is very little to choose between them.

4. Concluding remarks

We saw in Sections 2.1 and 2.2 that snapshots can be viewed either as finite element coefficient vectors or as finite element functions. The net effect of taking the latter view is the appearance of the finite element mass matrix $\mathbb{M}$ in, e.g., (8). Although handling the appearance of the mass matrix does not add a significant cost to the determination of a reduced basis, it also does not appreciably affect the effectiveness of the reduced-order model. For this reason, in

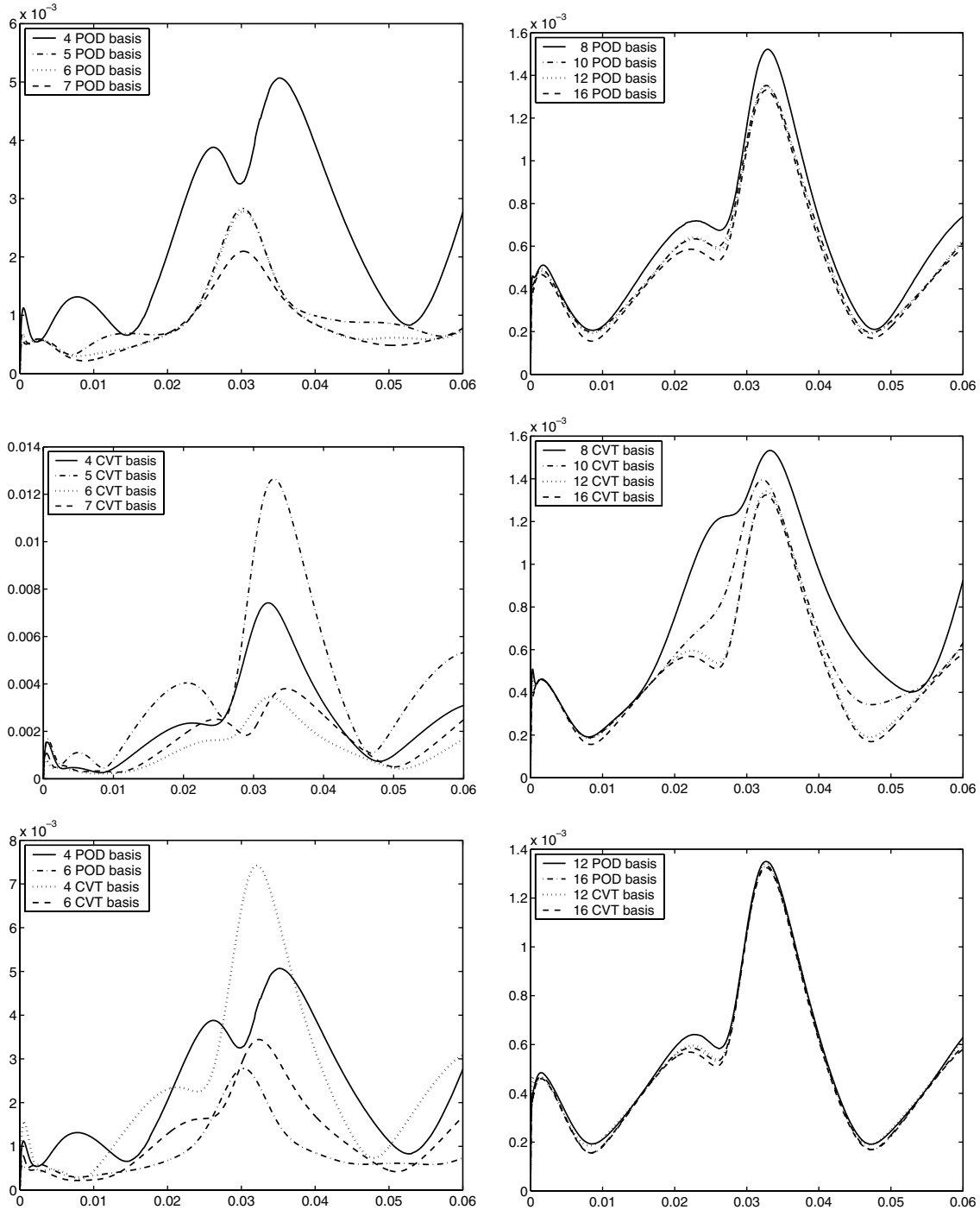

Fig. 8. $E(t)$ for the POD and CVT-based reduced-order models vs. time t for Case 3.

Section 3.6, we only provided results based on viewing snapshots as coefficient vectors.

The results given in Section 3.6 show that POD and CVT-based reduced-order modeling seem to be equally effective with regards to their ability to approximate solutions of the Navier–Stokes system for low values of the Reynolds number. POD-based reduced-order modeling has also been shown to be effective for some high Reynolds number problems but CVT-based reduced-order modeling has not been extensively applied, either in this or other papers, for this case. Thus, no conclusive comparisons of the merits of the two approaches can be made until CVT-based reduced-order modeling is applied to high Reynolds number problems.

The costs of determining a single POD and CVT reduced-order basis are pretty much the same. Certainly, once a reduced basis has been determined, the costs of carrying out a POD or CVT reduced-order simulations are the same. So, the only cost difference between the two approaches is what it costs to determine a reduced basis.

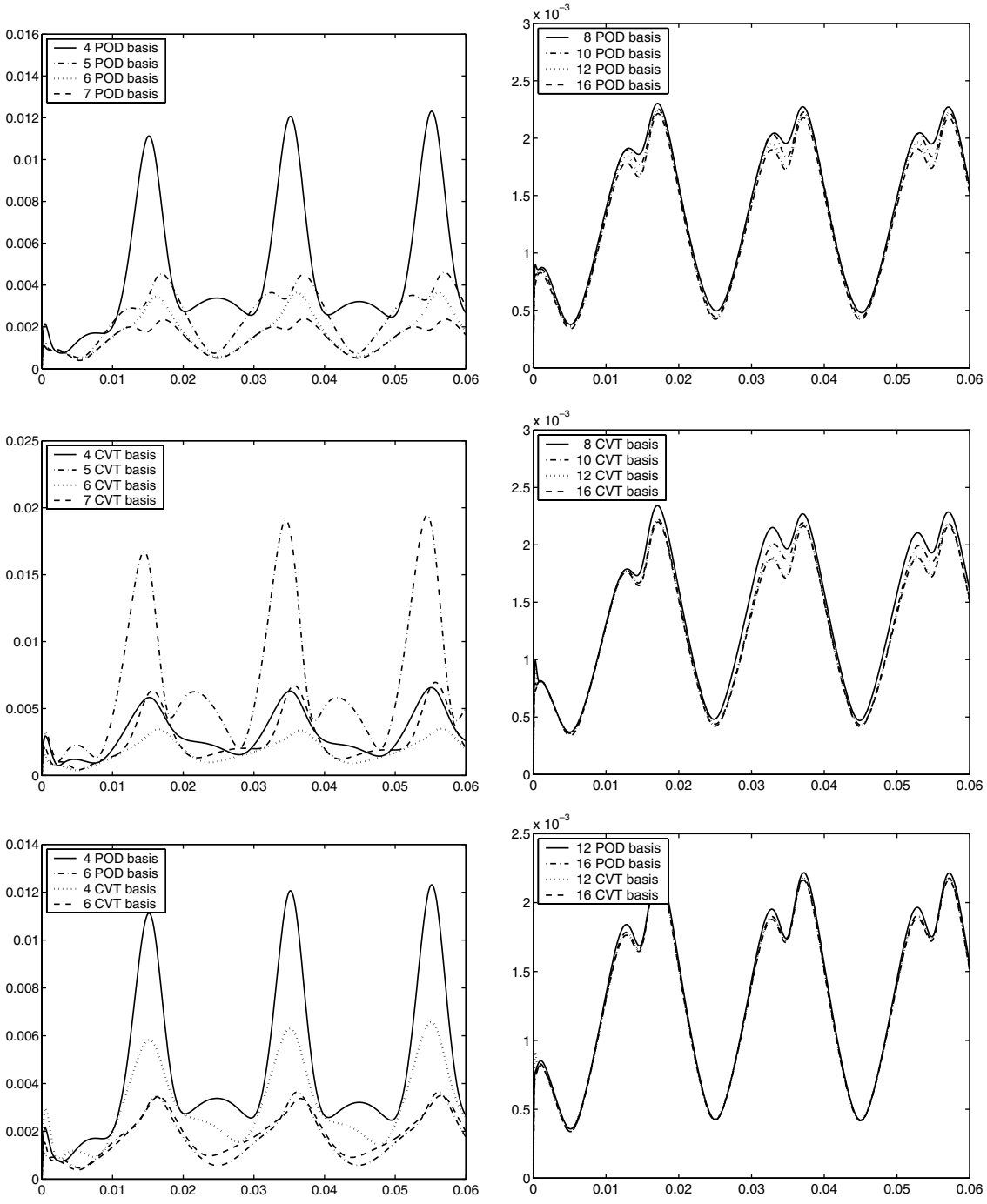

Fig. 9. $E(t)$ for the POD and CVT-based reduced-order models vs. time t for Case 4.

There are two contributions to the latter cost. First is the cost incurred for generating the snapshot set; since the construction of POD and CVT bases both use the same snapshot set, this cost is the same for the two approaches. Furthermore, although the costs of generating the POD and CVT bases from a given set of snapshots may be different, for both approaches that cost is miniscule compared to the cost of generating the snapshot set itself. For example, for the T-cell problem, the cost of determining a reduced-order basis from the snapshot set is less than one-half of one percent of the cost of determining the snapshot set itself. POD does have one advantage over CVT, namely that the POD bases are nested, i.e., a POD basis contains all the POD bases of lower dimension; this it not true about CVT bases. Again, this advantage is not huge since the costs of determining a second basis, should one be needed, from the snapshot set is very small compared to the cost of determining the snapshot set itself.

As has already been mentioned, at this point in time, one cannot give a definitive answer to the question of

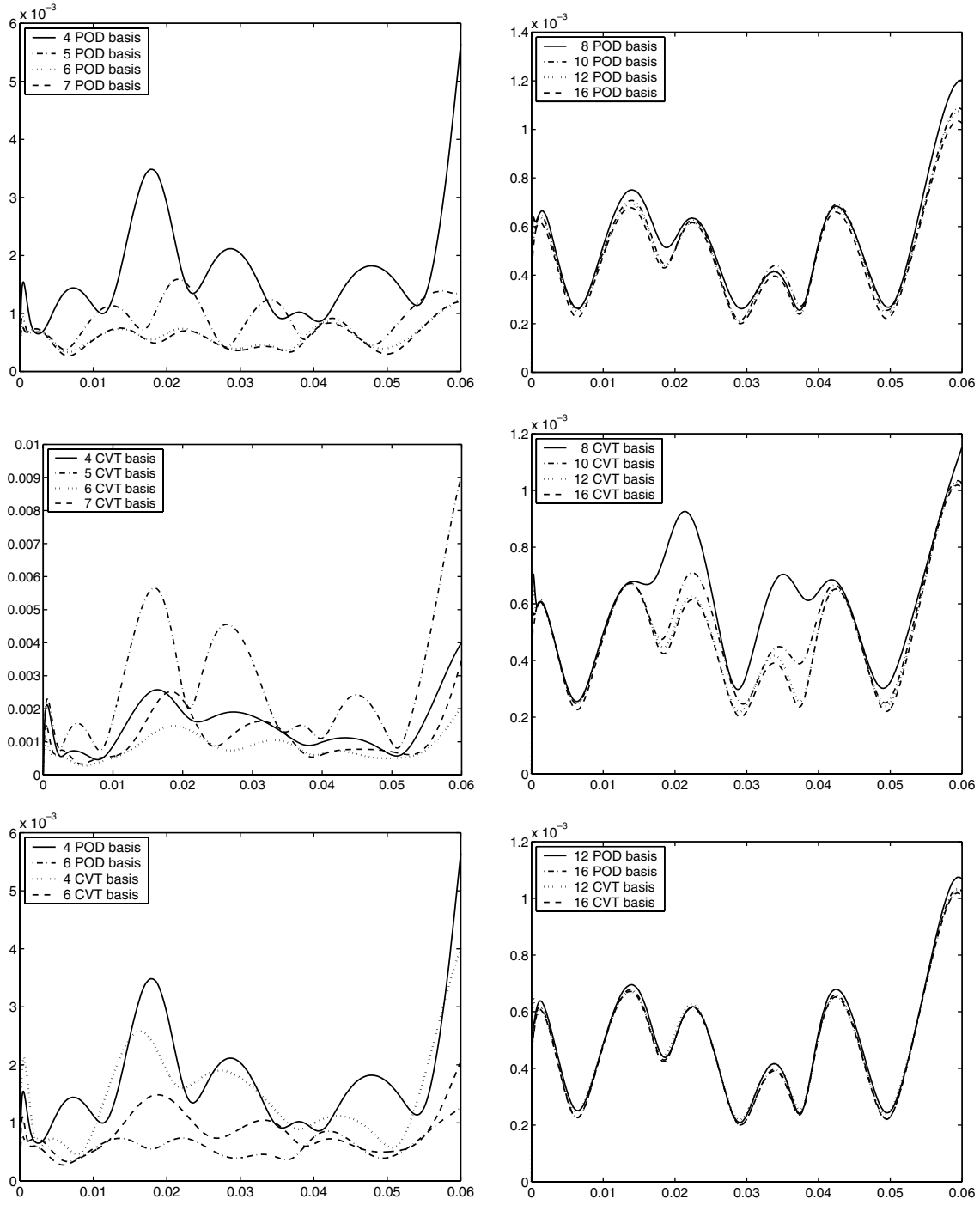

Fig. 10. $E(t)$ for the POD and CVT-based reduced-order models vs. time t for Case 5.

which reduced-order modeling approach is better. There have been many variations on the POD theme already developed in the literature, but the usefulness of these variations and especially what improvements they effect over plain-vanilla POD has not yet been clearly demonstrated. There are also several improvements possible in CVT-based reduced-order modeling. For example, it is possible that a judicious choice for the CVT-density function $\rho(\cdot)$ (see Section 2.2) can result in better performance. Perhaps after "improved" POD and CVT-based reduced-order models have been developed and thoroughly tested and after CVT-based models have been applied to high Reynolds number problems an answer to the question of which approach is better will become apparent. Certainly, at this time, one cannot say that CVT-based reduced-order modeling has definite advantages over the POD alternative.

In the end, it may not be necessary to choose between the POD and CVT approaches to reduced-order modeling since it is possible to develop combinations of the two approaches that hopefully would take advantage of the

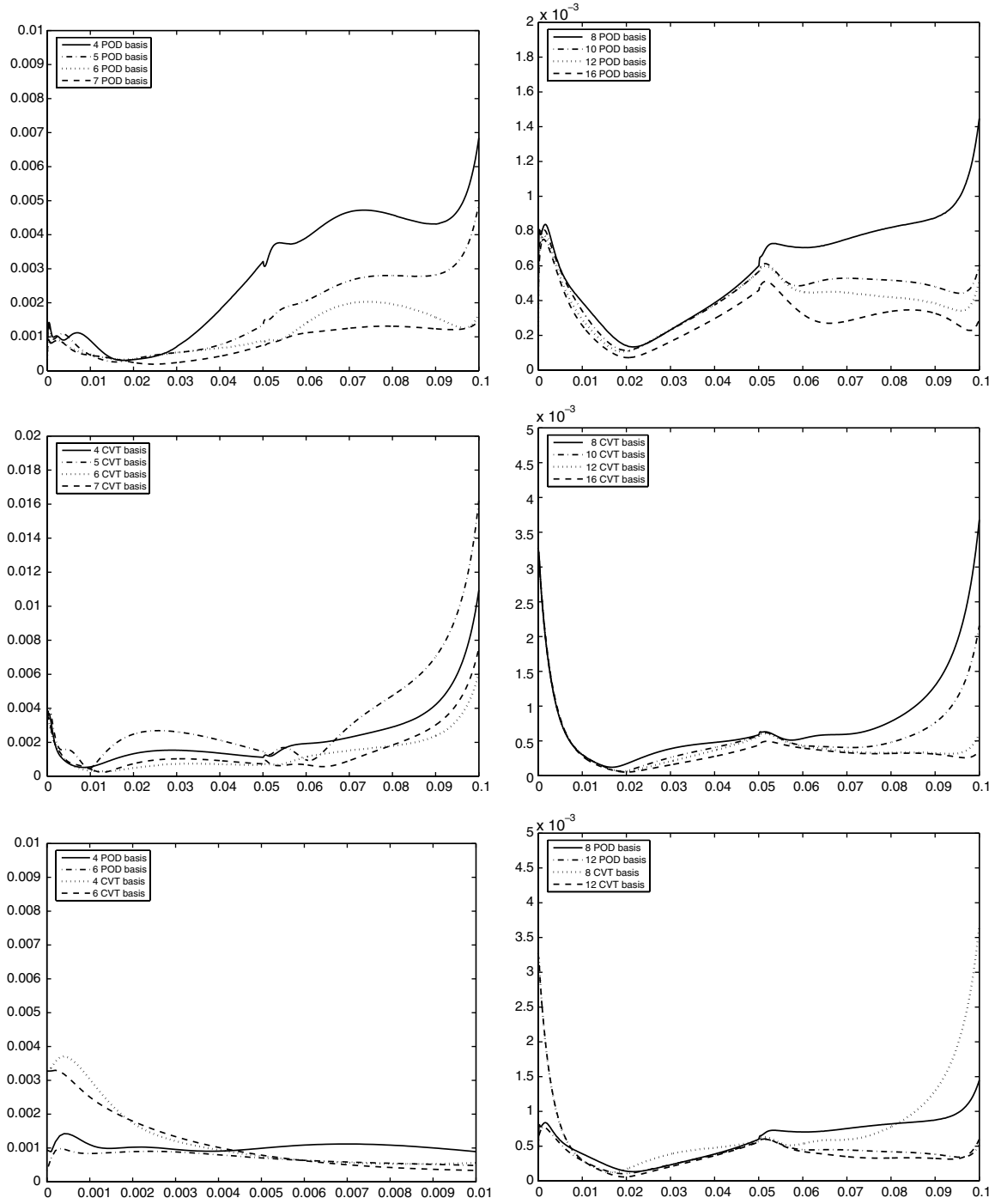

Fig. 11. $E(t)$ for the POD and CVT-based reduced-order models vs. time t for Case 6.

best features of both. For example, one could use a CVT algorithm to cluster the snapshots and then apply a POD analysis to each cluster separately instead of to the whole snapshot set. The final reduced basis would consist of the collection of POD basis vectors for all the clusters. More sophisticated POD-CVT combinations are discussed in [9]. The development and testing of such combined reduced-order models is currently under way.

It is of interest to compare the CPU times for effecting a simulation of the T-cell problem using a POD or CVT-based reduced-order model with that needed for a full finite element simulation. Such a comparison is given in Table 5 for the input data of Case 2. All computations were carried out on a Dell Precision Workstation 650 with dual 3.02 GHz CPUs. Note that 32,855 s were required to obtain the 500 snapshots that recall were generated by a calculation over the time interval $(0, 0.05)$ which was shorter than the interval $(0, 0.06)$ used in the simulations of Table 5. We see from the table the tremendous reduction in computing time that is effected by using a reduced-order

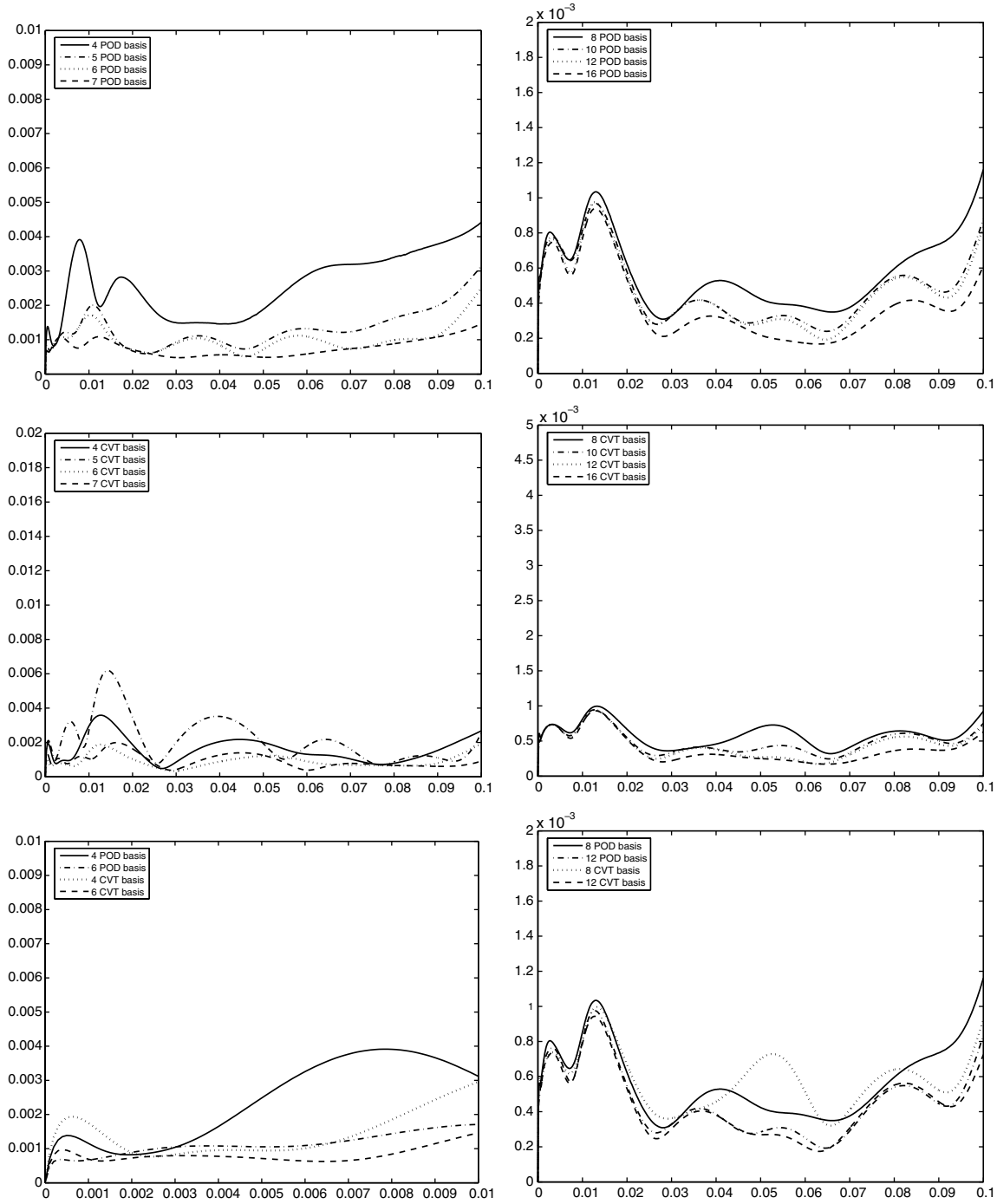

Fig. 12. $E(t)$ for the POD and CVT-based reduced-order models vs. time t for Case 7.

model. For example, once the reduced-bases have been determined (presumably this is considered an "off-line" calculation), one can do 2553 12-dimensional POD or CVT simulations for the same cost as a single full finite element simulation; one can do 28,673 six-dimensional reduced-order simulations for the same price! It is exactly this tremendous decrease in the costs of simulations that makes reduced-order modeling a subject of such intense interest.

In fact, the results given in Table 5 are *pessimistic* because the reduced-order simulations were carried out using a finite element-type assembly of the reduced-order model (10). If that model is pre-assembled, i.e., if we use (11) with a pre-assembly of the matrices and tensors involved, a substantial reduction in the CPU times for reduced-order modeling can be realized.

Finally, we want to emphasize that both POD and CVT reduced-order modeling seem to do a good job at extracting all of the useful information contained in snapshot sets. Thus, more attention than has so far been expended should be paid to developing methodologies for determining good

Table 4
E_T for the POD and CVT-based reduced-order models vs. the dimension K of the reduced-basis space for Case 5

K	POD	CVT
4	6.125e−02	5.264e−02
5	3.255e−02	1.012e−01
6	2.192e−02	2.976e−02
7	2.097e−02	4.038e−02
8	1.914e−02	2.147e−02
10	1.830e−02	1.857e−02
12	1.787e−02	1.761e−02
16	1.736e−02	1.721e−02

Table 5
For Case 2, the CPU times in seconds for POD or CVT simulations vs. the dimension K of the reduced basis and the CPU time for the full finite element simulation

K	CPU time
4	0.641
5	0.719
6	1.375
7	2.078
8	3.515
10	8.093
12	15.438
16	52.438
Full FEM	39,427

snapshot sets, i.e., to determine snapshot sets that contain the information needed to accurately approximate the results of the problems one wishes to apply a reduced-order model to.

References

[1] N. Aubry, W. Lian, E. Titi, Preserving symmetries in the proper orthogonal decomposition, SIAM J. Sci. Comput. 14 (1993) 483–505.

[2] G. Berkooz, P. Holmes, J. Lumley, The proper orthogonal decomposition in the analysis of turbulent flows, Ann. Rev. Fluid Mech. 25 (1993) 539–575.

[3] G. Berkooz, E. Titi, Galerkin projections and the proper orthogonal decomposition for equivariant equations, Phys. Lett. A 174 (1993) 94–102.

[4] E. Christensen, M. Brons, J. Sorensen, Evaluation of proper orthogonal decomposition-based decomposition techniques applied to parameter-dependent nonturbulent flows, SIAM J. Sci. Comput. 21 (2000) 1419–1434.

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

[6] Q. Du, V. Faber, M. Gunzburger, Centroidal Voronoi tessellations: applications and algorithms, SIAM Rev. 41 (1999) 637–676.

[7] Q. Du, M. Gunzburger, Model reduction by proper orthogonal decomposition coupled with centroidal Voronoi tessellation, in: Proc. Fluids Engineering Division Summer Meeting, FEDSM2002-31051, ASME, 2002.

[8] Q. Du, M. Gunzburger, Grid generation and optimization based on centroidal Voronoi tessellations, Appl. Math. Comput. 133 (2002) 591–607.

[9] Q. Du, M. Gunzburger, Centroidal Voronoi tessellation based proper orthogonal decomposition analysis, in: Proc. 8th Conference on Control of Distributed Parameter Systems, Birkhauser, Basel, 2002, pp. 137–150.

[10] Q. Du, M. Gunzburger, L. Ju, Meshfree, probabilistic determination of points sets and support regions for meshless computing, Comput. Methods Appl. Mech. Engrg. 191 (2002) 1349–1366.

[11] Q. Du, M. Gunzburger, L. Ju, Constrained CVTs on general surfaces, SIAM J. Sci. Comput. 24 (2003) 1488–1506.

[12] Q. Du, M. Gunzburger, L. Ju, Voronoi-based finite volume methods, optimal Voronoi meshes, and PDEs on the sphere, Comput. Methods Appl. Mech. Engrg. 192 (2003) 3933–3957.

[13] A. Faulds, Centroidal Voronoi Decompositions, Algorithms and Applications, M.S. Thesis, Department of Mathematics, Penn State University, 2002.

[14] A. Faulds, B. King, Sensor location in feedback control of partial differential equation systems, in: Proc. 2000 IEEE CCA/CACSD, IEEE, Washington, 2000, pp. 536–541.

[15] P.-A. Raviart, V. Girault, Finite Element Methods for Navier–Stokes Equations, Springer, Berlin, 1986.

[16] M. Graham, I. Kevrekidis, Pattern analysis and model reduction: some alternative approaches to the Karhunen–Lóeve decomposition, Comput. Chem. Engrg. 20 (1996) 495–506.

[17] G. Golub, C. van Loan, Matrix Computations, Johns Hopkins University, Baltimore, 1996.

[18] M. Gunzburger, Finite Element Methods for Viscous Incompressible Flows: A Guide to Theory, Practice and Algorithms, Academic, Boston, 1989.

[19] J. Hartigan, Clustering Algorithms, Wiley, 1975.

[20] J. Hartigan, M. Wong, Algorithm AS 136: A K-Means Clustering Algorithm, Appl. Stat. 28 (1979) 100–108.

[21] P. Holmes, J. Lumley, G. Berkooz, Turbulence, Coherent Structures, Dynamical Systems and Symmetry, Cambridge University, Cambridge, 1996.

[22] P. Holmes, J. Lumley, G. Berkooz, J. Mattingly, R. Wittenberg, Low-dimensional models of coherent structures in turbulence, Phys. Rep. 287 (1997) 337–384.

[23] L. Ju, Q. Du, M. Gunzburger, Probabilistic methods for centroidal Voronoi tessellations and their parallel implementations, J. Parallel Comput. 28 (2002) 1477–1500.

[24] K. Kunisch, S. Volkwein, Control of Burger's equation by a reduced order approach using proper orthogonal decomposition, JOTA 102 (1999) 345–371.

[25] K. Kunisch, S. Volkwein, Galerkin proper orthogonal decomposition methods for parabolic problems, Spezialforschungsbereich F003 Optimierung und Kontrolle, Projektbereich Kontinuierliche Optimierung und Kontrolle, Bericht Nr. 153, Graz, 1999.

[26] J. Lumley, Stochastic Tools in Turbulence, Academic, New York, 1971.

[27] W. Martinez, A. Martinez, Computational Statistics Handbook with MATLAB, Chapman and Hall/CRC, 2002.

[28] D. Nagy, Modal representation of geometrically nonlinear behavior by the finite element method, Comput. Struct. 10 (1979) 693.

[29] A. Noor, Recent advances in reduction methods for nonlinear problems, Comput. Struct. 13 (1981) 31–44.

[30] A. Noor, C. Anderson, J. Peters, Reduced basis technique for collapse analysis of shells, AAIA J. 19 (1981) 393.

[31] A. Noor, J. Peters, Tracking post-linit-paths with reduced basis technique, Comput. Methods Appl. Mech. Engrg. 28 (1981) 217.

[32] A. Okabe, B. Boots, K. Sugihara, S. Chiu, Spatial Tessellations: Concepts and Applications of Voronoi Diagrams, Wiley, Chichester, 2000.

[33] H. Park, D. Cho, Low dimensional modeling of flow reactors, Int. J. Heat Mass Transfer 39 (1996) 3311–3323.

[34] H. Park, J. Chung, A sequential method of solving inverse natural convection problems, Inverse Prob. 18 (2002) 529–546.

[35] H. Park, Y. Jang, Control of Burgers equation by means of mode reduction, Int. J. Engrg. Sci. 38 (2000) 785–805.

[36] H. Park, J. Lee, Solution of an inverse heat transfer problem by means of empirical reduction of modes, Z. Angew. Math. Phys. 51 (2000) 17–38.

[37] H. Park, W. Lee, An efficient method of solving the Navier–Stokes equations for flow control, Int. J. Numer. Meth. Engrg. 41 (1998) 1133–1151.
[38] H. Park, W. Lee, A new numerical method for the boundary optimal control problems of the heat conduction equation, Int. J. Numer. Meth. Engrg. 53 (2002) 1593–1613.
[39] J. Peterson, The reduced basis method for incompressible viscous flow calculations, SIAM J. Sci. Stat. Comput. 10 (1989) 777.
[40] M. Rathinam, L. Petzold, A new look at proper orthogonal decomposition, in press.
[41] M. Rathinam, L. Petzold, Dynamic iteration using reduced order models: a method for simulation of large scale modular systems, in press.
[42] S. Ravindran, Proper orthogonal decomposition in optimal control of fluids, Int. J. Numer. Meth. Fluids 34 (2000) 425–448.
[43] S. Ravindran, Reduced-order adaptive controllers for fluid flows using POD, SIAM J. Sci. Comput. 15 (2000) 457–478.
[44] J. Rodríguez, L. Sirovich, Low-dimensional dynamics for the complex Ginzburg–Landau equations, Physica D 43 (1990) 77–86.
[45] L. Sirovich, Turbulence and the dynamics of coherent structures, I–III, Quart. J. Appl. Math. 45 (1987) 561–590.
[46] N. Smaoui, D. Armbruster, Symmetry and the Karhunen–Loève analysis, SIAM J. Sci. Comput. 18 (1997) 1526–1532.
[47] H. Späth, Cluster Analysis Algorithms for Data Reduction and Classification of Objects, Ellis Horwood, 1980.
[48] H. Späth, Cluster Dissection and Analysis, Theory, FORTRAN Programs, Examples, Ellis Horwood, 1985.
[49] D. Sparks, Algorithm AS 58: Euclidean cluster analysis, Appl. Stat. 22 (1973) 126–130.
[50] S. Volkwein, Optimal control of a phase field model using the proper orthogonal decomposition, ZAMM 81 (2001) 83–97.
[51] S. Volkwein, Proper orthogonal decomposition and singular value decomposition, Spezialforschungsbereich F003 Optimierung und Kontrolle, Projektbereich Kontinuierliche Optimierung und Kontrolle, Bericht Nr. 153, Graz, 1999.