Journal of Computational Physics 259 (2014) 598–616

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A numerical investigation of velocity–pressure reduced order models for incompressible flows

Alfonso Caiazzo [a], Traian Iliescu [b,1], Volker John [a,c,*], Swetlana Schyschlowa [a]

[a] Weierstrass Institute for Applied Analysis and Stochastics, Leibniz Institute in Forschungsverbund Berlin e. V. (WIAS), Mohrenstr. 39, 10117 Berlin, Germany
[b] Department of Mathematics, Virginia Tech, 456 McBryde Hall, Blacksburg, VA 24061-0123, USA
[c] Free University of Berlin, Department of Mathematics and Computer Science, Arnimallee 6, 14195 Berlin, Germany

ARTICLE INFO

Article history:
Received 31 January 2013
Received in revised form 20 November 2013
Accepted 2 December 2013
Available online 11 December 2013

Keywords:
Navier–Stokes equations
Proper orthogonal decomposition
Velocity–pressure reduced order models
Snapshot accuracy

ABSTRACT

This report has two main goals. First, it numerically investigates three velocity–pressure reduced order models (ROMs) for incompressible flows. The proper orthogonal decomposition (POD) is used to generate the modes. One method computes the ROM pressure solely based on the velocity POD modes, whereas the other two ROMs use pressure modes as well. To the best of the authors' knowledge, one of the latter methods is novel. The second goal is to numerically investigate the impact of the snapshot accuracy on the results of the ROMs. Numerical studies are performed on a two-dimensional laminar flow past a circular obstacle. Comparing the results of the ROMs and of the simulations for computing the snapshots, it turns out that the latter results are generally well reproduced by the ROMs. This observation is made for snapshots of different accuracy. Both in terms of reproducing the results of the underlying simulations for obtaining the snapshots and of efficiency, the two ROMs that utilize pressure modes are superior to the ROM that uses only velocity modes.

1. Introduction

Let $\Omega \subset \mathbb{R}^d$, $d \in \{2, 3\}$, be a bounded domain and let $[0, T]$ be a finite time interval. Incompressible flows are modeled by the incompressible Navier–Stokes equations (in dimensionless form) for the velocity $\mathbf{u} : [0, T] \times \Omega \to \mathbb{R}^d$ and the pressure $p : (0, T] \times \Omega \to \mathbb{R}$

$$\begin{aligned} &\partial_t \mathbf{u} - \nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p = \mathbf{f} \quad \text{in } (0, T] \times \Omega, \\ &\nabla \cdot \mathbf{u} = 0 \quad \text{in } [0, T] \times \Omega, \end{aligned} \tag{1}$$

where $\mathbf{f}$ models body forces acting on the flow and ν is the inverse of the Reynolds number. System (1) has to be equipped with an initial velocity $\mathbf{u}(0, \mathbf{x}) = \mathbf{u}_0(\mathbf{x})$ and with appropriate boundary conditions on the boundary $\partial\Omega$ of Ω. For the concrete flow problem considered in this report, there is no forcing term ($\mathbf{f} = \mathbf{0}$) and the boundary can be decomposed as

* Corresponding author at: Weierstrass Institute for Applied Analysis and Stochastics, Leibniz Institute in Forschungsverbund Berlin e. V. (WIAS), Mohrenstr. 39, 10117 Berlin, Germany.
E-mail addresses: alfonso.caiazzo@wias-berlin.de (A. Caiazzo), iliescu@vt.edu (T. Iliescu), volker.john@wias-berlin.de (V. John), Swetlana.Schyschlowa@wias-berlin.de (S. Schyschlowa).

[1] Partially supported by the National Science Foundation (DMS-0513542 and OCE-0620464) and the Air Force Office for Scientific Research (FA9550-08-1-0136).

0021-9991/$ – see front matter
http://dx.doi.org/10.1016/j.jcp.2013.12.004

$\partial\Omega = \Gamma_{\text{in}} \cup \Gamma_0 \cup \Gamma_{\text{out}}$, where the boundary parts are mutually disjoint. Problem (1) is completed with the following boundary conditions:

$$\mathbf{u}(t,\mathbf{x}) = \mathbf{g}(\mathbf{x}) \quad \text{at } [0,T]\times\Gamma_{\text{in}} \text{ inlet},$$
$$\mathbf{u}(t,\mathbf{x}) = \mathbf{0} \quad \text{at } [0,T]\times\Gamma_0 \text{ solid walls},$$
$$(\nu\nabla\mathbf{u} - pI)\mathbf{n} = \mathbf{0} \quad \text{at } [0,T]\times\Gamma_{\text{out}} \text{ outlet},$$

where $\mathbf{n}$ denotes the outer normal unit vector on $\partial\Omega$.

In order to compute a numerical approximation of the solution of (1) with a finite element method, (1) can be transformed into a time-continuous variational formulation, using the spaces

$$\mathbf{V} = \left\{\mathbf{v}\in H^1(\Omega):\ \mathbf{v} = \mathbf{0} \text{ on } \Gamma_{\text{in}}\cup\Gamma_0\right\}, \qquad Q = L^2(\Omega).$$

Furthermore, let $(\cdot,\cdot)$ denote the standard inner product in $L^2(\Omega)$ and let $\mathbf{u}_{\mathbf{g}}\in H^1(\Omega)$ be an extension of $\mathbf{g}$ into Ω for all t. Then, the time-continuous variational formulation reads: find $\mathbf{u}:[0,T]\to H^1(\Omega)$, such that $\mathbf{u}-\mathbf{u}_{\mathbf{g}}\in\mathbf{V}$ for all t, and $p:(0,T]\to Q$ such that

$$\begin{aligned}
&(\partial_t\mathbf{u},\mathbf{v}) + (\nu\nabla\mathbf{u},\nabla\mathbf{v}) + \big((\mathbf{u}\cdot\nabla)\mathbf{u},\mathbf{v}\big) - (\nabla\cdot\mathbf{v},p) = 0 \quad \forall\mathbf{v}\in\mathbf{V},\\
&-(\nabla\cdot\mathbf{u},q) = 0 \quad \forall q\in Q,
\end{aligned} \tag{2}$$

and $\mathbf{u}(0,\mathbf{x}) = \mathbf{u}_0(\mathbf{x})$.

In finite element methods, the spaces $(\mathbf{V}, Q)$ in (2) are replaced by finite-dimensional spaces $(\mathbf{V}^h, Q^h)$ consisting of piecewise polynomials with respect to a triangulation $\mathcal{T}^h$ of Ω. Usually, $(\mathbf{V}^h, Q^h)$ are equipped with a local basis, i.e., with a basis where each basis function has a small support such that an easy construction of the spaces $(\mathbf{V}^h, Q^h)$ is possible.

The use of finite element methods for the numerical solution of (2) allows to compute more and more details of the flow field by increasing the dimension of the finite element spaces. However, the number of basis functions can become very large, yielding large linear or nonlinear systems to be solved in the simulations. Consequently, the numerical simulation of the flow can be very time-consuming. In addition, the finite element basis is generally defined independently of the solution, and it only depends on the structure of the computational mesh. In the case that a priori information on the solution is available, one could transfer this knowledge to the finite element space by pre-adapting the triangulation of Ω.

Reduced order models (ROMs) aim at reducing the computational cost of full finite element, finite difference, or finite volume simulations by drastically reducing the dimension of the solution space. The key idea of ROMs consists in utilizing basis functions that already represent the most important features of the solution. In contrast to finite element bases, ROM bases are global bases. In this report, we focus on ROMs in which the basis functions are obtained through a proper orthogonal decomposition (POD) of a set of snapshots, see, e.g. [3,5,10–15,19,20,22,34,40,46]. Here, the snapshots will be obtained from detailed numerical simulations. It is worth noticing that generally the snapshots might even come from experimental data [4,20].

This report has the following two main goals. First, it investigates three different types of ROMs that compute, besides the velocity, also the pressure, called here for shortness vp-ROMs. One of these vp-ROMs is, to the best of the authors' knowledge, new. Second, this report investigates the impact of the accuracy of the simulations for computing the snapshots, shortly denoted by snapshot accuracy, on the vp-ROM results. The motivation and background for these two numerical investigations are presented next.

To motivate the use of vp-ROMs, we note that although most ROMs for incompressible flows do not include a pressure component, there are important settings in which vp-ROMs are appropriate. From the practical point of view, the pressure is needed in many computational fluid dynamics applications, e.g., for the simulation of fluid–structure interaction problems and for the computation of relevant quantities, such as drag and lift coefficients on solid bodies, and for ROM simulations of shear flows [36]. Other reasons for including the pressure are connected to the definition of ROMs. Using only a velocity ROM leads to a comparatively simple model that can be simulated very efficiently. The rationale behind velocity ROMs, as it can be found in the literature, is that all snapshots are divergence-free, hence all basis functions are divergence-free and consequently the ROM velocity is divergence-free, such that the pressure (which acts as a Lagrange multiplier of the divergence-free constraint) is not needed. As it will be clarified in Section 3.1, the same rationale can be applied in the context of finite element methods and discretely divergence-free velocity fields. In this case, only the integrals of the product of the velocity divergence and all test functions from the discrete pressure space vanish. In fact, many numerical methods for computing the snapshots do not provide pointwise divergence-free flow fields. Even for finite element methods, the discretely divergence-free property does not hold for many popular discretizations of the Navier–Stokes equations. Such examples include the case of using the same finite element spaces for velocity and pressure, where a numerical stabilization becomes necessary, or pressure-correction schemes without reconstructing the discretely divergence-free solution. Experimental data will generally not be divergence-free as well. Altogether, the violation of the divergence-free constraint on the snapshots is another reason for incorporating the pressure into ROMs for incompressible flow simulations. Moreover, as already pointed out in [8], the availability of the pressure enables the computation of the residual of the strong form of

the Navier–Stokes equations (1). Strong residuals are often needed in stabilized discretizations, e.g., for stabilization with respect to the violation of the inf-sup condition or with respect to dominating convection.

One can find in the literature different proposals for incorporating the pressure, or an approximation of the pressure, into the ROM. One class of vp-ROMs consists in defining a ROM for the pressure that only uses the velocity POD modes [36]. A vp-ROM from this class, denoted by VMB-ROM, will be investigated in our numerical studies. A second class of vp-ROMs employs pressure POD basis functions in addition to the velocity POD basis functions. The pressure POD basis functions can be computed separately from the velocity POD basis functions (i.e., the decoupled approach) [36], or together with them (i.e., the coupled, monolithic approach) [8,47]. In this study, we utilize the decoupled approach. Two vp-ROMs that employ a pressure POD basis will be investigated in this report. The first vp-ROM in this class, here denoted by PMB-ROM, is based on the approach proposed in [1]. The second vp-ROM, called SM-ROM, is, to the best of the authors' knowledge, novel. This new vp-ROM uses a residual-based stabilization mechanism for the incompressible Navier–Stokes equations. It is based on a mathematically well understood method [9]. The advantage over the two other vp-ROMs consists in the fact that its derivation requires the snapshots to be only discretely divergence-free (but not pointwise), and it does not need any ad hoc treatment of external forces and pressure boundary conditions. Overall, three vp-ROMs will be considered in the numerical studies. VMB-ROM and PMB-ROM solve the same equation for the pressure but in different finite-dimensional spaces. PMB-ROM and SM-ROM work in the same space, but in these methods different equations for the pressure are solved. All vp-ROMs can be considered as a postprocessing step to a velocity ROM.

The second main goal of this paper is to investigate the impact of the accuracy of the snapshots, and therefore of the resulting POD basis, on the numerical results of the vp-ROMs. In order to exploit the tremendous gain in efficiency that is provided by using the POD basis, a straightforward idea consists in using efficient numerical methods for the ROM simulations. For example, one can avoid to solve nonlinear problems by utilizing an explicit or semi-implicit time stepping scheme like in, e.g., [1,6,12,31,46]. On the other hand, the generation of the snapshots might be time consuming. Considering, e.g., a turbulent flow, then one can perform a direct numerical simulation (DNS), if the Reynolds number is sufficiently small for this approach to be feasible, or one can apply more or less advanced turbulence models on more or less fine meshes for this purpose. All approaches (should) give reasonable approximations of the large and important flow structures. The main differences will be in the resolved details of the flow. However, the DNS has to be performed on a very fine mesh and its computing time is usually orders of magnitude higher than that of a simulation with a turbulence model on a coarser grid. And even simulations with a simple turbulence model, like the Smagorinsky model, might be much faster than simulations with an advanced model, like a variational multiscale method. Since ROMs aim to compute only the most important features of the solution, and since ROMs generally utilize a simple numerical method, the following question naturally emerges: "*How strong are the impacts of the snapshot accuracy, on the one hand, and of the* (*simple*) *numerical method used in the ROMs, on the other hand, onto the ROM results?*"

In this report, we will perform the first step in numerically investigating this question. To the best of the authors' knowledge, this report presents the first study of this topic. To construct snapshot data of different accuracies, two approaches can be considered. The first approach uses the same numerical method, but different discretization parameters, e.g., different mesh sizes and/or different time steps. The second approach uses the same discretization parameters, but different numerical methods. In this study, we utilize the second approach.

The report is organized as follows. Section 2 gives a short review of the way a basis of the ROM is obtained with POD. Several vp-ROMs are discussed in Section 3. Section 4 presents numerical studies that compare these vp-ROMs. These studies are performed for a two-dimensional (2D) laminar flow around a cylinder. This example is on the one hand sufficiently simple to allow the computation of accurate reference solutions to compare with. On the other hand, the flow is laminar such that it is possible to focus on the two main goals of this report without interference of additional aspects, e.g., like turbulence modeling. The report concludes with a summary and an outlook in Section 5.

2. Computation of a ROM basis with POD

For the report to be self-contained, this section briefly presents the computation of a basis for ROMs with POD. For more details, the reader is referred to [20,35,41,44].

Consider a function $u(t,\mathbf{x}) : [0,T] \times \Omega \to \mathbb{R}^d$ and let $R \in \mathbb{N}$. Then, the goal of POD consists in finding two sets $\{\alpha_r(t), \alpha_r : [0,T] \to \mathbb{R}\}_{r=1}^R$, $\{\phi_r(\mathbf{x}), \phi_r : \Omega \to \mathbb{R}^d\}_{r=1}^R$ which deliver the best approximation

$$\arg\min_{(\alpha_r,\phi_r)} \left\| u(t,\mathbf{x}) - \sum_{r=1}^{R} \alpha_r(t)\phi_r(\mathbf{x}) \right\|$$

according to a norm which determines in which sense the best approximation is sought. For flow problems, usually the $L^2(0,T;L^2(\Omega))$ norm is used, since it is directly related to the kinetic energy of the flow field.

In the framework of the numerical solution of partial differential equations, u is usually given at a finite number of times $t_1,\dots,t_M$, the so-called snapshots. For the sake of simplifying the presentation, and without loss of generality, we assume that the snapshots are computed at equidistant time steps τ with a finite element method. The numerical studies will use this approach. Then, usually an approximation of the error in the (square of the) $L^2(0,T)$ norm is considered, e.g., by (a modification of) the composite trapezoidal rule

$$\arg\min_{(\alpha_r,\phi_r)} \sum_{m=1}^{M} \tau \left\| u(t_m,\mathbf{x}) - \sum_{r=1}^{R} \alpha_r(t_m)\phi_r(\mathbf{x}) \right\|^2. \tag{3}$$

In this section, it will be assumed that the norm is induced by an inner product $(\cdot,\cdot)$, and $\{\phi_r\}_{r=1}^R$ are orthonormal with respect to this inner product. The functions $u(t_m,\mathbf{x})$ can be represented by a finite number of degrees of freedom, since they were computed from a discretized equation. Likewise, $\phi_r(\mathbf{x})$ will be represented by the same degrees of freedom. Let $\{\mathbf{x}_n\}_{n=1}^N$ be the nodes, $\{v_n(\mathbf{x})\}_{n=1}^N$ be the nodal basis with $v_n(\mathbf{x}_k)=\delta_{nk}$, and $V^h=\operatorname{span}\{v_n\}_{n=1}^N$. Then, the representations have the form

$$u(t_m,\mathbf{x}) = \sum_{n=1}^{N} u(t_m,\mathbf{x}_n)v_n(\mathbf{x}), \qquad \phi_r(\mathbf{x}) = \sum_{n=1}^{N} \phi_r(\mathbf{x}_n)v_n(\mathbf{x}). \tag{4}$$

Thus, the data can be collected into the so-called snapshot matrix $U\in\mathbb{R}^{N\times M}$ with $(U)_{nm}=u(t_m,\mathbf{x}_n)$. It will be required that $R\leqslant \operatorname{rank}(U)$. In practice, it is usually $M\ll N$, which will be assumed also here. It is clear that the result of (3) does not depend on τ, consequently without loss of generality one can consider $\tau=1$.

From Hilbert space theory, it is known that

$$\alpha_r(t_m) = \big(u(t_m,\cdot),\phi_r\big). \tag{5}$$

Inserting this expression into (3) and using the orthonormality of $\{\phi_r\}_{r=1}^R$, problem (3) can be reformulated as

$$\begin{aligned}&\arg\min_{(\alpha_r,\phi_r),\ \|\phi_r\|=1} \sum_{m=1}^{M} \left(u(t_m,\cdot) - \sum_{r=1}^{R}\big(u(t_m,\cdot),\phi_r\big)\phi_r, u(t_m,\cdot) - \sum_{r=1}^{R}\big(u(t_m,\cdot),\phi_r\big)\phi_r \right)\\ &= \arg\min_{(\alpha_r,\phi_r),\ \|\phi_r\|=1} \sum_{m=1}^{M} \left[\big\|u(t_m,\cdot)\big\|^2 - \sum_{r=1}^{R}\big(u(t_m,\cdot),\phi_r\big)^2 \right].\end{aligned}$$

Since the first term is a given constant, (3) becomes equivalent to maximizing the second term. The Lagrangian functional of this optimization problem has the form

$$\mathcal{L}(\phi_1,\dots,\phi_R;\lambda_1,\dots,\lambda_R) = \sum_{m=1}^{M}\sum_{r=1}^{R}\big(u(t_m,\cdot),\phi_r\big)^2 - \sum_{r=1}^{R}\lambda_r\big[(\phi_r,\phi_r)-1\big].$$

Optimal values can be obtained only at the stationary points

$$\begin{aligned}0 &= \partial_{\phi_r}\mathcal{L} = 2\sum_{m=1}^{M}\big(u(t_m,\cdot),\phi_r\big)\big(u(t_m,\cdot),\psi\big) - 2\lambda_r(\phi_r,\psi), \quad \forall\psi\in V^h,\\ 0 &= \partial_{\lambda_r}\mathcal{L} = (\phi_r,\phi_r)-1,\end{aligned}$$

$r=1,\dots,R$. The first condition can be reformulated as follows

$$\sum_{m=1}^{M}\big(u(t_m,\cdot),\phi_r\big)\big(u(t_m,\cdot),\psi\big) = \left(\sum_{m=1}^{M}\big(u(t_m,\cdot),\phi_r\big)u(t_m,\cdot),\psi\right) = (\lambda_r\phi_r,\psi)$$

for all $\psi\in V^h$. This equality holds if and only if

$$\sum_{m=1}^{M}\big(u(t_m,\cdot),\phi_r\big)u(t_m,\cdot) = \lambda_r\phi_r, \quad r=1,\dots,R. \tag{6}$$

For a function $z\in V^h$, denote by $\underline{z}$ the vector of its coefficients with respect to the basis $\{v_n\}_{n=1}^N$. Then, the inner product can be written as

$$(z,w) = \underline{z}^T S\underline{w}, \quad \forall z,w\in V^h, \tag{7}$$

where the matrix $S\in\mathbb{R}^{N\times N}$ with $S_{ln}=(v_n,v_l)$ is symmetric and positive definite.

Using (4) and (7), problem (6) can be rewritten as the following eigenvalue problem in $\mathbb{R}^N$

$$UU^T S\underline{\phi}_r = \lambda_r\underline{\phi}_r, \quad r=1,\dots,R. \tag{8}$$

Multiplying (8) from the left-hand side by $S^{1/2}$, it can be readily seen that the eigenvalue problem can be reformulated as an eigenvalue problem with the symmetric, positive semi-definite matrix $S^{1/2}UU^TS^{T/2}$. Hence, all eigenvalues λ_r are

real and non-negative. In particular, the largest R eigenvalues are positive because of $R \leqslant \operatorname{rank}(U)$. These are exactly the eigenvalues whose corresponding eigenfunctions ϕ_r are sought.

The solution of (8) is generally very expensive since N is usually very large. Eq. (5) can be written in the form $\underline{\alpha}_r = U^T S \underline{\phi}_r$. Using this equation and multiplying (8) by $U^T S$ from the left-hand side leads to an eigenvalue problem in R^M

$$U^T S U \underline{\alpha}_r = \lambda_r \underline{\alpha}_r, \quad U^T S U \in \mathbb{R}^{M\times M}, \tag{9}$$

whose solution is generally much cheaper than the solution of (8). Thus, solving (9) gives $(\lambda_r, \underline{\alpha}_r)$, $r = 1, \ldots, R$, with orthogonal eigenvectors $\underline{\alpha}_r$. Multiplying (9) with $\underline{\alpha}_r^T$ from the left-hand side, it follows that $\|U\underline{\alpha}_r\| = \lambda_r^{1/2}(\underline{\alpha}_r^T \underline{\alpha}_r)^{1/2}$. Setting

$$\underline{\phi}_r = \frac{U\underline{\alpha}_r}{\|U\underline{\alpha}_r\|} = \frac{U\underline{\alpha}_r}{(\underline{\alpha}_r^T U^T S U \underline{\alpha}_r)^{1/2}} = \frac{U\underline{\alpha}_r}{\lambda_r^{1/2}(\underline{\alpha}_r^T \underline{\alpha}_r)^{1/2}}, \quad r = 1, \ldots, R, \tag{10}$$

one obtains with (9)

$$UU^T S\underline{\phi}_r = \frac{1}{\|U\underline{\alpha}_r\|} U\big(U^T S U \underline{\alpha}_r\big) = \frac{1}{\|U\underline{\alpha}_r\|} U\lambda_r \underline{\alpha}_r = \lambda_r \underline{\phi}_r.$$

Thus, $(\lambda_r, \underline{\phi}_r)$ with $\underline{\phi}_r$ given by (10) solves (8). The approach of computing the eigenvalues λ_r by solving (9) and the eigenvectors or modes $\underline{\phi}_r$ by (10) is called method of snapshots. It was first proposed in [41].

In practice, in case of stationary Dirichlet boundary conditions, the POD is often not applied to the function $u(t, \mathbf{x})$ itself but to the fluctuations of that function. To this end, one has to define a temporal mean value, e.g., by

$$\overline{u}(\mathbf{x}) = \frac{1}{M}\sum_{m=1}^{M} u(t_m, \mathbf{x}),$$

which is subtracted from the snapshots, to obtain the fluctuations

$$u'(t_m, \mathbf{x}) = u(t_m, \mathbf{x}) - \overline{u}(\mathbf{x}), \quad m = 1, \ldots, M.$$

Now, the POD is computed starting from $u'(t_m, \mathbf{x})$ instead of $u(t_m, \mathbf{x})$. Then, the basic ansatz for the ROM has the form

$$u_{\mathrm{ro}}(t, \mathbf{x}) = \overline{u}(\mathbf{x}) + \sum_{r=1}^{R} \alpha_r'(t)\phi_r'(\mathbf{x}) = \sum_{r=0}^{R} \alpha_r'(t)\phi_r'(\mathbf{x}), \tag{11}$$

with $\phi_0'(\mathbf{x}) = \overline{u}(\mathbf{x})$ and $\alpha_0'(t) = 1$. Generally, there is no orthogonality condition between $\overline{u}(\mathbf{x})$ and any of the functions $\phi_r'(\mathbf{x})$. In the numerical studies presented in Section 4, the POD was applied to the fluctuations.

3. ROMs for incompressible flows

ROM for incompressible flows is meanwhile widely used and it is an active field of research, see, e.g., [2,5,6,42,46] for recent publications.

In the case of the Navier–Stokes equations, the solution of the problem consists of two components, velocity and pressure. Thus, the considerations of Section 2 apply to $u = (\mathbf{u}, p)$, where here $(\mathbf{u}, p)$ are discrete approximations of the velocity and the pressure, respectively. For simplicity of presentation, the discrete character of $(\mathbf{u}, p)$ is not emphasized in the notation below.

The standard procedure for deriving ROMs for incompressible flows employs the POD basis together with a Galerkin projection. Let $\{\phi_r'\}_{r=1}^R = \{(\boldsymbol{\varphi}_r', \psi_r')\}_{r=1}^R$ be the POD basis of the fluctuations, where $\boldsymbol{\varphi}_r'$ are the velocity basis functions and ψ_r' are the pressure basis functions. To simplify the presentation, the same number of velocity and pressure modes is used. Then, the Galerkin projection of the Navier–Stokes equations (1) yields the following ROM: Find $(\mathbf{u}_{\mathrm{ro}}, p_{\mathrm{ro}})$ with $\mathbf{u}_{\mathrm{ro}} - \overline{\mathbf{u}} : (0, T] \to \operatorname{span}\{\boldsymbol{\varphi}_r'\}_{r=1}^R$ and $p_{\mathrm{ro}} - \overline{p} : (0, T] \to \operatorname{span}\{\psi_r'\}_{r=1}^R$ such that for $r = 1, \ldots, R$,

$$\begin{aligned}
&\big(\partial_t \mathbf{u}_{\mathrm{ro}}, \boldsymbol{\varphi}_r'\big) + \big(\nu\nabla\mathbf{u}_{\mathrm{ro}}, \nabla\boldsymbol{\varphi}_r'\big) + \big((\mathbf{u}_{\mathrm{ro}}\cdot\nabla)\mathbf{u}_{\mathrm{ro}}, \boldsymbol{\varphi}_r'\big) - \big(p_{\mathrm{ro}}, \nabla\cdot\boldsymbol{\varphi}_r'\big) \\
&\quad = \big(\mathbf{f}, \boldsymbol{\varphi}_r'\big) + \int_{\Gamma_{\mathrm{out}}} (\nu\nabla\mathbf{u}_{\mathrm{ro}} - p_{\mathrm{ro}} I)\mathbf{n}\cdot\boldsymbol{\varphi}_r'\, d\mathbf{s}, \\
&\big(\nabla\cdot\mathbf{u}_{\mathrm{ro}}, \psi_r'\big) = 0,
\end{aligned} \tag{12}$$

and $\mathbf{u}_{\mathrm{ro}}(0, \mathbf{x})$ is an approximation of the initial condition with the POD modes. In this report, it will be assumed that the external force $\mathbf{f}$ in (12) vanishes.

3.1. Velocity ROM

In many, probably even most, published reports on ROMs for incompressible flows, only velocity models are considered. This approach is based on the argument that, if the snapshots are divergence-free, then also each POD basis function $\boldsymbol{\varphi}'_r$ is divergence-free, which follows from (10). This argument also holds in the case of a POD basis for the fluctuations, as the average velocity $\bar{\mathbf{u}}(\mathbf{x})$, being a linear combination of snapshots, is divergence-free. In fact, if $\mathbf{u}_{\mathrm{ro}}$ and $\{\boldsymbol{\varphi}'_r\}_{r=1}^R$ are divergence-free, the pressure term on the left-hand sides of the momentum equation and the continuity equation in (12) drop out. Moreover, assuming natural boundary conditions at the open boundaries, which is the case in the numerical studies in Section 4, also the boundary term in (12) vanishes. The ROM for the velocity, based on the Galerkin projection has the following form: Find

$$\mathbf{u}_{\mathrm{ro}} = \bar{\mathbf{u}}(\mathbf{x}) + \sum_{r=1}^{R} \alpha'_r(t)\boldsymbol{\varphi}'_r(\mathbf{x}),$$

such that, for $r = 1, \ldots, R$,

$$\left(\partial_t \mathbf{u}_{\mathrm{ro}}, \boldsymbol{\varphi}'_r\right) + \left(\nu\nabla\mathbf{u}_{\mathrm{ro}}, \nabla\boldsymbol{\varphi}'_r\right) + \left((\mathbf{u}_{\mathrm{ro}}\cdot\nabla)\mathbf{u}_{\mathrm{ro}}, \boldsymbol{\varphi}'_r\right) = \left(\mathbf{f}, \boldsymbol{\varphi}'_r\right). \tag{13}$$

Note that (13) requires only a POD for the velocity. Furthermore, if the usual $L^2(0,T;L^2(\Omega))$ norm is used in performing the POD, the mass matrix $(\boldsymbol{\varphi}'_s, \boldsymbol{\varphi}'_r)$, $s, r \geqslant 1$, becomes the identity.

As already mentioned in the introduction, the assumption of divergence-free snapshots is idealized. For instance, in the context of inf-sup stable finite element discretizations there are only very few divergence-free pairs of spaces, like the Scott–Vogelius element on barycentric refined grids [48]. Most of the inf-sup stable pairs, in particular the most popular ones like the Taylor–Hood finite elements, are only discretely divergence-free. The magnitude of the divergence of the finite element solution can be even large [32]. Indeed, the standard finite element convergence theory shows that the $L^2(\Omega)$ norm of the divergence has the same order of convergence as the error in the $L^2(\Omega)$ norm of the velocity gradient [23].

The reduction from (12) to (13), however, can be achieved in certain situations by using the argument that the snapshots are only discretely divergence-free. This situation holds if the finite element continuity equation is not perturbed by any additional term. Moreover, the modes $\{\boldsymbol{\varphi}_r, \psi_r\}_{r=1}^R$ and the mean values should belong to the velocity and pressure finite element spaces, respectively. In this case, the pressure term in the ROM (12) drops out and (12) reduces to the velocity ROM (13). The above argument does not apply if the continuity equation is perturbed by additional terms, as in the case of finite element pairs that do not fulfill a discrete inf-sup condition, e.g., equal finite elements for velocity and pressure, which require additional stabilizations introducing a control on the pressure through a modification of the continuity equation.

An essential motivation for developing ROMs is the computational efficiency. For this reason, one usually prefers to avoid complex and time-consuming time discretization methods in combination with a ROM, see, e.g., [45,42]. In the numerical studies in Section 4, the Crank–Nicolson scheme for the time discretization in combination with the IMEX scheme for the linearization of (13) was used. Denoting the discrete times by t^k, the functions at those times with a corresponding superscript, and the length of the equidistant time step by τ, the linearized and time-discretized velocity ROM (13) reads: find

$$\mathbf{u}_{\mathrm{ro}}^{k+1} = \bar{\mathbf{u}} + \sum_{r=1}^{R} \left(\alpha'_r\right)^{k+1} \boldsymbol{\varphi}'_r,$$

such that for $r = 1, \ldots, R$, and $k = 0, 1, \ldots$

$$\begin{aligned}
&\left(\mathbf{u}_{\mathrm{ro}}^{k+1}, \boldsymbol{\varphi}'_r\right) + \frac{1}{2}\tau\left(\nu\nabla\mathbf{u}_{\mathrm{ro}}^{k+1}, \nabla\boldsymbol{\varphi}'_r\right) + \frac{1}{2}\tau\left(\left(\mathbf{u}_{\mathrm{ro}}^{k}\cdot\nabla\right)\mathbf{u}_{\mathrm{ro}}^{k+1}, \boldsymbol{\varphi}'_r\right) \\
&\quad = \left(\mathbf{u}_{\mathrm{ro}}^{k}, \boldsymbol{\varphi}'_r\right) + \frac{1}{2}\tau\left(\mathbf{f}^{k+1}, \boldsymbol{\varphi}'_r\right) + \frac{1}{2}\tau\left(\mathbf{f}^{k}, \boldsymbol{\varphi}'_r\right) - \frac{1}{2}\tau\left(\nu\nabla\mathbf{u}_{\mathrm{ro}}^{k}, \nabla\boldsymbol{\varphi}'_r\right) - \frac{1}{2}\tau\left(\left(\mathbf{u}_{\mathrm{ro}}^{k}\cdot\nabla\right)\mathbf{u}_{\mathrm{ro}}^{k}, \boldsymbol{\varphi}'_r\right).
\end{aligned} \tag{14}$$

The initial condition $\{(\alpha'_r)^0\}_{r=1}^R$ for (14) can be computed by

$$\left(\alpha'_r\right)^0 = \left(\mathbf{u}^0 - \bar{\mathbf{u}}, \boldsymbol{\varphi}'_r\right),$$

where $\mathbf{u}^0$ is a finite element approximation of the initial condition.

3.2. Velocity–pressure ROMs

To our best knowledge, the ROMs with a pressure component can be divided into two classes, depending on whether they use pressure POD modes or not. If pressure modes are employed, there are again two principal approaches. In the decoupled approach, the velocity and pressure snapshots are considered separately. Choosing the velocity POD modes with the highest kinetic energy and the pressure POD modes with the largest $L^2(\Omega)$ norm, one obtains two separate bases. For

this approach, it is straightforward to choose a different number of POD modes for velocity and pressure, based on the corresponding distribution of their eigenvalues. In the coupled approach, each snapshot, and thus, each POD mode, has both a velocity and the corresponding pressure component. This approach naturally yields the same number of velocity and pressure modes. In this report, the decoupled approach will be considered.

3.2.1. A velocity–pressure ROM based on the velocity modes (VMB-ROM)

If the ROM uses only a POD basis for the velocity, the pressure field must be reconstructed a posteriori using the ROM velocity solution. In this report, the approach proposed in [36] will be considered. It utilizes the pressure Poisson equation

$$-\Delta p = \nabla \cdot \big((\mathbf{u}\cdot\nabla)\mathbf{u}\big) \quad \text{in } \Omega, \tag{15}$$

which is obtained by taking the divergence of the momentum equation of the Navier–Stokes equations (1). Eq. (15) is equipped with Neumann boundary conditions on Γ_0 and Γ_{in} (as in [36]) and Dirichlet boundary conditions on Γ_{out}, to ensure the well-posedness of problem (15). The main idea used in [36] consists in approximating $\mathbf{u}$ on the right-hand side of (15) by $\mathbf{u}_{\text{ro}}$ defined by (11). Assuming that all functions in (11) are divergence-free, one obtains the equation

$$-\Delta p_{\text{ro}} = \sum_{r=0}^{R}\sum_{s=0}^{R}\alpha_r'(t)\alpha_s'(t)\left(\sum_{i=1}^{d}\sum_{j=1}^{d}\partial_{x_i}\big(\boldsymbol{\varphi}_r'\big)_j\partial_{x_j}\big(\boldsymbol{\varphi}_s'\big)_i\right) \quad \text{in } \Omega. \tag{16}$$

Problem (16) is an equation in space, in which the functions $\alpha_r'(t), \alpha_s'(t)$ act as constants. Hence, the solution of (16) has the form

$$p_{\text{ro}}(t,\mathbf{x}) = \sum_{r=0}^{R}\sum_{s=0}^{R}\alpha_r'(t)\alpha_s'(t)\,p_{rs}(\mathbf{x}), \tag{17}$$

with $p_{rs}(\mathbf{x})$ solving

$$-\Delta p_{rs} = \sum_{i=1}^{d}\sum_{j=1}^{d}\partial_{x_i}\big(\boldsymbol{\varphi}_r'\big)_j\partial_{x_j}\big(\boldsymbol{\varphi}_s'\big)_i \quad \text{in } \Omega. \tag{18}$$

In what follows, the ROM (14) together with $p_{\text{ro}}(\mathbf{x})$ given by (17), will be referred to as the VMB-ROM (velocity-modes-based ROM).

Note that $p_{rs}(\mathbf{x}) = p_{sr}(\mathbf{x})$ and, thus, system (18) has only $(R+1)R/2$ unknowns. The functions $p_{rs}(\mathbf{x})$ can be computed in a preprocessing step. In this way, the ROM pressure $p_{\text{ro}}(\mathbf{x})$ can be efficiently computed at each time step by using (17). It should be noted that this preprocessing approach does not work if the Navier–Stokes equations have a time-dependent body force which is not divergence-free.

In [36], the term $(\nabla p_{\text{ro}}, \boldsymbol{\varphi}_r')$ was even introduced into the momentum equation (13) to improve the results of ROMs for shear flows. If only the term $(\nabla p_{\text{ro}}, \boldsymbol{\varphi}_r')$ is of interest, while the explicit computation of p_{ro} is not required, it was proposed in [16,36] to approximate this term using linear models in $\alpha_r'(t)$, resulting in additional minimization problems to be solved for determining the coefficients in the ansatz.

3.2.2. A velocity–pressure ROM based on pressure modes (PMB-ROM)

A second approach for defining a pressure ROM a posteriori consists in discretizing the pressure equation (15) using a pressure POD basis [1].

Assuming that the reduced velocity $\mathbf{u}_{\text{ro}}^{k+1}$ has already been computed, and assuming the velocity POD space to be divergence-free, one obtains the pressure Poisson equation

$$-\Delta p_{\text{ro}}^{k+1} = \nabla\cdot\big(\big(\mathbf{u}_{\text{ro}}^{k+1}\cdot\nabla\big)\mathbf{u}_{\text{ro}}^{k+1}\big) \quad \text{in } \Omega. \tag{19}$$

In [1], it was suggested to compute the pressure by applying the Galerkin projection to (19) on the pressure POD modes $\{\psi_r'\}_{r=1}^{R}$. This suggestion leads to the following method: find

$$p_{\text{ro}}^{k+1} = \overline{p} + \sum_{r=1}^{R}\big(\beta_r'\big)^{k+1}\psi_r', \tag{20}$$

such that for $r = 1,\dots,R$,

$$\big(\nabla p_{\text{ro}}^{k+1}, \nabla\psi_r'\big) = \big(\nabla\cdot\big(\big(\mathbf{u}_{\text{ro}}^{k+1}\cdot\nabla\big)\mathbf{u}_{\text{ro}}^{k+1}\big), \psi_r'\big) \tag{21}$$

with a homogeneous Dirichlet boundary condition on Γ_{out} and a Neumann boundary condition on $\partial\Omega\setminus\Gamma_{\text{out}}$, the same as used in the VMB-ROM. In the numerical studies presented in Section 4, the ROM (14) together with (21), will be referred to as PMB-ROM (pressure-modes-based ROM).

Note that, although VMB-ROM and PMB-ROM are based on the same equation, the respective pressures are computed using different discrete spaces. In the VMB-ROM, the pressure is represented in terms of the functions computed from the derivatives of the velocity POD modes, see (18), whereas in the PMB-ROM the pressure is represented in terms of the pressure POD modes, cf. (20).

3.2.3. A velocity–pressure ROM based on a stabilization of the coupled problem (SM-ROM)

A ROM that is based on a coupled scheme for $(\mathbf{u}_{\mathrm{ro}}, p_{\mathrm{ro}})$, like the ROM (12), raises the issue of the inf-sup condition for saddle point problems [17]. It seems to be hard to address this question for the general setting of the ROM, unlike for, e.g., finite element methods, where the approximation spaces are specified beforehand and the corresponding discrete inf-sup condition can be investigated a priori. In the POD-ROM framework, however, the approximation spaces are problem-dependent – they are known only after having performed the underlying finite element simulations, or even an actual physical experiment [4,20]. Thus, checking beforehand whether the velocity and pressure POD spaces satisfy an inf-sup condition is generally not possible. In the context of finite element methods, the discrete inf-sup condition states, loosely speaking, that the dimension of the discrete velocity space is sufficiently high compared with the dimension of the discrete pressure space. In the case of reduced basis method, several suggestions exist in the literature on how to enrich the velocity space to verify the inf-sup condition [30,37,38]. For POD-ROMs, to the authors' best knowledge, there are no results on that issue. In the framework of finite element methods, the coupled velocity–pressure problem can be stabilized by including additional terms in the variational formulation, in order to overcome a possible violation of the inf-sup condition. This aspect motivates the new ROM for the pressure introduced in this section. The idea is to define an equation for the pressure based on a stabilization approach for the coupled velocity–pressure ROM (12). Among the stabilizations for incompressible flow problems [9], the class of residual-based approaches seems to be promising in our opinion, since these methods immediately allow in addition the stabilization of dominant convection. These approaches are also the basis of residual-based variational multiscale methods [7].

A popular residual-based stabilization is the SUPG/PSPG/grad-div method, see [9] and the references therein. In this approach, the residual of the momentum equation is tested with the streamline derivative of the velocity and the gradient of the pressure. Thus, the following stabilization term is added to the momentum equation

$$s_h(\mathbf{u}, \mathbf{v}, p, q) = \sum_{K\in\mathcal{T}^h} \big(\partial_t \mathbf{u} - \nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p - \mathbf{f}, \delta_{K,\mathbf{u}}(\mathbf{u}\cdot\nabla)\mathbf{v} + \delta_{K,p}\nabla q\big)_K, \tag{22}$$

where K denotes a mesh cell of the considered triangulation $\mathcal{T}^h$ of Ω, and $\delta_{K,\mathbf{u}}$ and $\delta_{K,p}$ are the stabilization parameter functions. The so-called grad-div term is based on the residual of the continuity equation and it adds to the momentum equation the following stabilization term

$$\sum_{K\in\mathcal{T}^h} (\nabla\cdot\mathbf{u}, \mu_K \nabla\cdot\mathbf{v})_K, \tag{23}$$

where μ_K denotes the stabilization parameter function. The SUPG term in (22) accounts for stabilizing dominating convection, the grad-div term (23) accounts for improving the discrete conservation of mass, and the PSPG term in (22) accounts for stabilizing a violated inf-sup condition.

Note that the SUPG/PSPG/grad-div method has already been used in [8,47] within a ROM framework. However, in [8,47] the ROM pressure was not computed by solving a separate pressure equation.

Although an explicit treatment of (22) and (23) might be advantageous in terms of computational efficiency, the stabilization of the inf-sup condition has to appear in the system matrix in order to become effective.

In the residual for the momentum balance (22), the viscous term is generally neglected, since it is of little importance in the interesting case of small viscosity. Denote by

$$\mathbf{res}_{\mathrm{ro}}^k = \frac{\mathbf{u}_{\mathrm{ro}}^k - \mathbf{u}_{\mathrm{ro}}^{k-1}}{\tau} + \big(\mathbf{u}_{\mathrm{ro}}^k\cdot\nabla\big)\mathbf{u}_{\mathrm{ro}}^k + \nabla p_{\mathrm{ro}}^k - \mathbf{f}_{\mathrm{ro}}^k$$

an approximation of the residual at t^k. Then, the right-hand side of the momentum equation of the coupled system at t^{k+1} contains the explicit stabilization terms

$$-\sum_{K\in\mathcal{T}^h} \delta_{K,\mathbf{u}}\big(\mathbf{res}_{\mathrm{ro}}^k, \big(\mathbf{u}_{\mathrm{ro}}^k\cdot\nabla\big)\boldsymbol{\varphi}_r'\big)_K, \; -\sum_{K\in\mathcal{T}^h} \mu_K\big(\nabla\cdot\mathbf{u}_{\mathrm{ro}}^k, \nabla\cdot\boldsymbol{\varphi}_r'\big)_K, \quad r = 1,\ldots,R, \tag{24}$$

where the stabilization parameters are now assumed to be piecewise constant. In the continuity equation, the term

$$\sum_{K\in\mathcal{T}^h} \delta_{K,p}\big(\nabla p_{\mathrm{ro}}^{k+1}, \nabla\psi_r'\big)_K, \quad r = 1,\ldots,R,$$

is included in the system matrix. Moving the velocity–pressure coupling of the stabilization

Table 1
Velocity–pressure ROMs presented in Section 3. VMB-ROM and PMB-ROM use the same equation for computing the ROM pressure, but the discrete spaces in these methods are different. PMB-ROM and SM-ROM apply the same discrete space, but different equations for the ROM pressure.

Acronym	Description	Equations
VMB-ROM	velocity-modes based	(14), (18)
PMB-ROM	pressure-modes based	(14), (21)
SM-ROM	stabilization motivated	(14), (27)

$$- \sum_{K\in\mathcal{T}^h} \delta_{K,p}\big(\mathbf{res}_{\mathrm{ro}}^{k+1} - \nabla p_{\mathrm{ro}}^{k+1}, \nabla\psi_r'\big)_K, \quad r = 1,\dots,R, \tag{25}$$

to the right-hand side of the continuity equation, the matrix of the coupled problem has the form

$$\begin{pmatrix} A_{\mathrm{ro}} & B_{\mathrm{ro}}^T \\ B_{\mathrm{ro}} & C_{\mathrm{ro}} \end{pmatrix}, \tag{26}$$

where A_{ro} contains the discretization of the temporal derivative, the viscous, and the linearized convective term, and

$$(B_{\mathrm{ro}})_{sr} = \big(\nabla\cdot\boldsymbol{\varphi}_r', \psi_s'\big), \quad r,s = 1,\dots,R,$$
$$(C_{\mathrm{ro}})_{sr} = \sum_{K\in\mathcal{T}^h} \delta_{K,p}\big(\nabla\psi_r', \nabla\psi_s'\big)_K, \quad r,s = 1,\dots,R.$$

Consider now the ROM matrix (26) for the case in which the snapshots are discretely divergence-free, e.g., when they are computed with a Galerkin finite element method with inf-sup stable pairs of finite element spaces. In this case, the matrix B_{ro} vanishes. Hence, the system with matrix (26) results in two decoupled equations. After having computed the velocity, the right-hand side (25) of the continuity equation can be evaluated. If the stabilizations of dominating convection and of violating the mass conservation (24) can be neglected, as for the flow problem considered in Section 4, the velocity equation corresponding to (26) is the same as that in the velocity ROM (14).

Altogether, we propose to combine the ROM velocity equation (14) with

$$\sum_{K\in\mathcal{T}^h} \delta_{K,p}\big(\nabla p_{\mathrm{ro}}^{k+1}, \nabla\psi_r'\big)_K$$
$$= - \sum_{K\in\mathcal{T}^h} \delta_{K,p}\left(\frac{\mathbf{u}_{\mathrm{ro}}^{k+1} - \mathbf{u}_{\mathrm{ro}}^{k}}{\tau} + \big(\mathbf{u}_{\mathrm{ro}}^{k+1}\cdot\nabla\big)\mathbf{u}_{\mathrm{ro}}^{k+1} - \mathbf{f}_{\mathrm{ro}}^{k+1}, \nabla\psi_r'\right)_K, \quad r = 1,\dots,R. \tag{27}$$

Below, the ROM (14) together with (27), will be referred to as SM-ROM (stabilization-motivated ROM). The SM-ROM (14), (27) is, to the best of the authors' knowledge, new.

The matrix for the pressure equation in (26) corresponds to the discretization of a scaled Laplacian. In (27), the stabilization parameters $\{\delta_{K,p}\}$ have to be chosen. Since there is no numerical analysis for this choice in the context of ROMs, we used the guidance provided by the standard finite element theory. In the numerical studies in Section 4, the same number of velocity and pressure modes were used. For this case, following the finite element theory, an optimal stabilized method is obtained with $\delta_{K,p} = C\,h_K$ in (27), where C is a generic constant and h_K is the diameter of the mesh cell K, [9]. Note that the value of the constant C has no effect on the SM-ROM, since it appears on both sides of (27). Thus, without loss of generality, we used $\delta_{K,p} = h_K$.

It is worth emphasizing that one of the advantages of SM-ROM is that its derivation requires the velocity snapshots to be only discretely divergence-free but not pointwise divergence-free, as needed for the derivation of the VMB-ROM and PMB-ROM. Furthermore, being based on a general formulation of the Navier–Stokes equations, the SM-ROM does not need any ad hoc treatment of external forces and any specification of additional pressure boundary conditions.

4. Numerical studies

First, this section presents numerical results for the three vp-ROMs introduced in Section 3, which are summarized in Table 1. Second, it investigates the impact of the snapshot accuracy on the results of the vp-ROMs. The effect of the dimension of the POD basis on the numerical results is also monitored.

4.1. The laminar flow around a cylinder

To allow a detailed discussion of the results, the numerical studies were carried out for the well understood example of a 2D laminar flow around a circular cylinder defined in [39]. This problem is given in

$$\Omega = \big\{(0, 2.2)\times(0, 0.41)\big\} \setminus \big\{\mathbf{x}\colon \big(\mathbf{x} - (0.2, 0.2)\big)^2 \leqslant 0.05^2\big\},$$

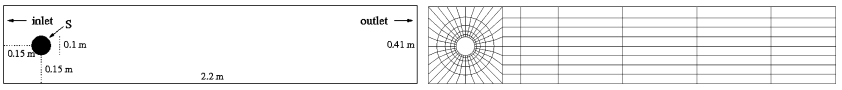

Fig. 1. The flow domain (left) and the coarse grid (right).

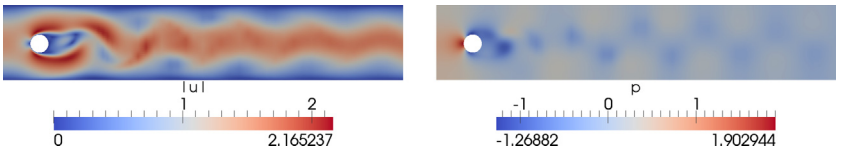

Fig. 2. Snapshots of the finite element solution.

see Fig. 1. At the boundary $x = 0$ the steady-state inflow condition $\mathbf{u}(x, 0) = (0.41^{-2}(6y(0.41 - y)), 0)^T$ is used, at the boundary $x = 2.2$ the outflow (do nothing) condition $(\nu\nabla\mathbf{u} - pI)\mathbf{n} = \mathbf{0}$ is applied, while no-slip boundary conditions are prescribed elsewhere. The kinematic viscosity of the fluid is given by $\nu = 10^{-3}$ m^2/s. The initial condition is a fully developed flow field that has to be computed in a preprocessing step. Based on the mean inflow velocity $U = 1$ m/s, the diameter of the cylinder $L = 0.1$ m and the kinematic viscosity, the Reynolds number of the flow is $Re = 100$. In the fully developed periodic regime, a vortex shedding (von Kármán vortex street) can be observed behind the obstacle, see Fig. 2.

Quantities of interest are the drag and lift coefficients at the cylinder and corresponding reference intervals were defined in [39], see Table 2. In the presented numerical studies, these quantities were computed as volume integrals, e.g., for the drag coefficient,

$$c_d = -\frac{2}{LU^2}\big[(\partial_t\mathbf{u}, \mathbf{v}_d) + (\nu\nabla\mathbf{u}, \nabla\mathbf{v}_d) + \big((\mathbf{u}\cdot\nabla)\mathbf{u}, \mathbf{v}_d\big) - (\nabla\cdot\mathbf{v}_d, p)\big], \tag{28}$$

for a function $\mathbf{v}_d \in H^1(\Omega)$ such that $\mathbf{v}_d = (1, 0)^T$ on the boundary of the cylinder and $\mathbf{v}_d = (0, 0)^T$ on all other boundaries. The lift coefficient c_l was computed in a similar way, using a test function $\mathbf{v}_l$ such that $\mathbf{v}_l = (0, 1)^T$ on the boundary of the cylinder. Since $\mathbf{v}_d$ is not discretely divergence-free, the last term in (28) does not vanish and the pressure is needed for computing the drag (and the lift) coefficient. In the numerical studies presented below, the approach described in [24,25] was used for evaluating (28). In the periodic regime, another important quantity of interest is the Strouhal number $\mathrm{St} = L\nu_f/U$, which is correlated to the frequency of the vortex shedding ν_f. We are not aware of any relation between the kinetic energy, which was the criterion used to compute the POD basis, and these quantities of interest.

All simulations were performed with the code MooNMD [26] on a grid obtained by three uniform red refinements of the coarse grid presented in Fig. 1, where the resolution of the cylinder was improved with each refinement. The Navier–Stokes equations were discretized in space using the inf-sup stable Taylor–Hood Q_2/Q_1 finite elements, resulting in 107 712 velocity degrees of freedom and 13 616 pressure degrees of freedom. For the time discretization, the Crank–Nicolson time integration scheme with the time step $\tau = 0.005$ was employed, which showed, among simple time stepping schemes, a good balance between numerical accuracy and computational efficiency [27,28].

4.2. Numerical methods for computing the snapshots

One of the goals of this report is to numerically investigate the effect of the snapshot accuracy on the results obtained with the vp-ROMs. Different numerical methods on the same grids in time and space were employed for computing snapshots of different accuracies.

The most expensive numerical method, denoted by SP-NONLIN, requires the solution of a nonlinear saddle point problem at each discrete time. The nonlinear problem is solved by a fixed point iteration (Picard iteration), as described, e.g., in [24]. The second numerical method, denoted by SP-LIN, uses the IMEX version of the Crank–Nicolson scheme, similarly to (14). Thus, the convective term is discretized explicitly in the convective component $((\mathbf{u}^k\cdot\nabla)\mathbf{u}^{k+1}, \mathbf{v})$ and all other terms are handled implicitly. SP-LIN yields one linear saddle point problem at each time iteration. Finally, the third numerical method, denoted by PC, removes even the saddle point character of the problem, combining the Crank–Nicolson IMEX scheme with the standard incremental pressure-correction scheme, which is the so-called van Kan scheme [18,43]. At each discrete time, PC requires only the solution of one linear equation for the velocity, where the equations for the velocity components are decoupled, and one linear equation for the pressure. PC provides two approximations for the velocity. Here, the non-incompressible velocity approximation that satisfies the boundary conditions is used.

Clearly, the three different numerical methods possess different numerical costs. In the simulations for computing the snapshots, SP-NONLIN took about 2.6 times longer than SP-LIN, and SP-LIN took about 2.2 longer than PC. But it can be also expected that the three methods exhibit differences in the accuracy. This expectation is met by the results presented in Fig. 3 and Table 2. One can observe that SP-NONLIN, the numerical method with the highest computational price, is also the most accurate one, as the results for all reference values are within the reference intervals given in Table 2. The accuracy deteriorates for SP-LIN and for PC, but one can see that the results of SP-LIN are still considerably more accurate than

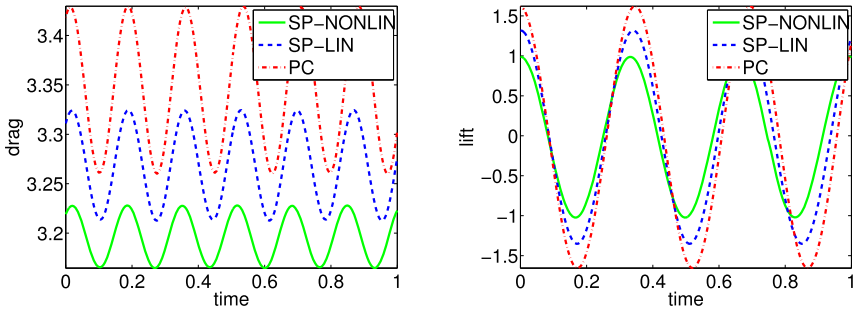

Fig. 3. Drag and lift coefficients for the finite element simulations.

Table 2
Maximal drag coefficient, maximal lift coefficient, and Strouhal number for the finite element simulations.

	$c_d^{\max}$	$c_l^{\max}$	St
SP-NONLIN	3.23	1.01	0.302
SP-LIN	3.32	1.35	0.294
PC	3.43	1.65	0.288
Reference results from [39]	[3.22, 3.24]	[0.99, 1.01]	[0.295, 0.305]

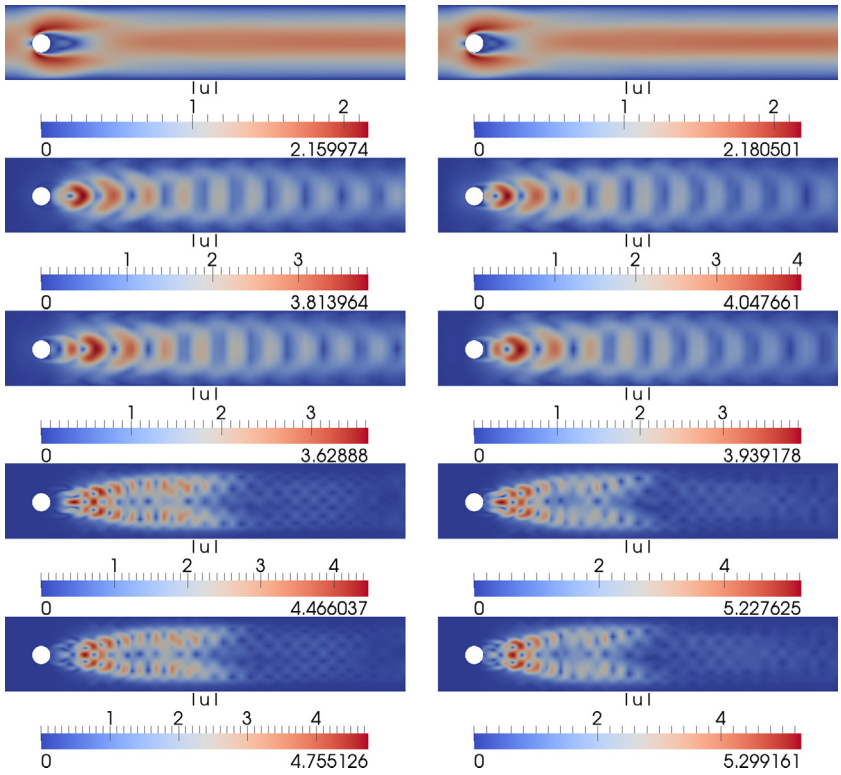

Fig. 4. Norm of the mean velocity (top) and the first POD modes of the velocity fluctuations: POD basis computed from SP-NONLIN (left) and PC (right).

the results computed with PC. Accordingly, we obtained three sets of snapshots: of the highest accuracy, of intermediate accuracy, and of the lowest accuracy.

4.3. Impact of the snapshot accuracy on the POD modes

This section focuses on the influence of using numerical methods of different accuracy on the POD basis.

From the simulations with SP-NONLIN, SP-LIN, and PC, after having collected snapshots over the time interval $[0, 2]$ for each discrete time, three different POD bases were generated. Figs. 4 and 5 display the norm of the mean and of the

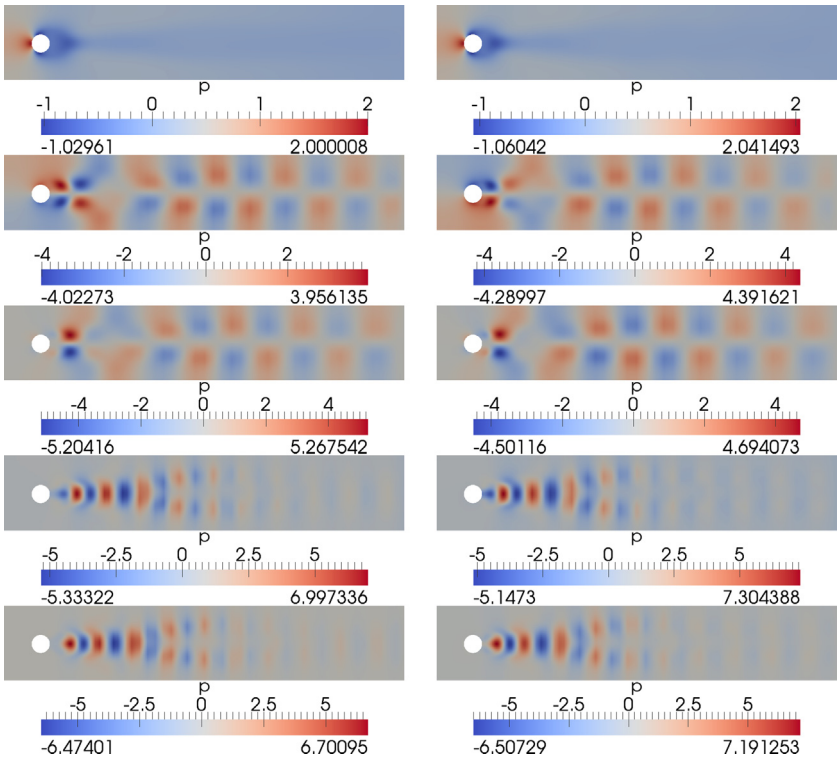

Fig. 5. Mean pressure (top) and the first POD modes of the pressure fluctuations: POD basis computed from SP-NONLIN (left) and PC (right).

first POD modes of the velocity and pressure fluctuations, respectively. For clarity of presentation, only the most accurate (SP-NONLIN) and the lowest accurate (PC) numerical methods are considered. Both Fig. 4 and Fig. 5 show that, although structurally similar, the maximum and minimum values are quite different for the two numerical methods. One can observe that for the velocity these differences increase with increasing POD mode index. Next, the POD bases are investigated in terms of the POD eigenvalues $\{\lambda_r\}$, defined in (8), and the missing energy ratio (MER) of the discarded POD modes of the fluctuations. For a POD basis of rank R, using (5), (8), and (9), the MER is defined as follows

$$\begin{aligned}\mathrm{MER}_R &= \frac{\frac{1}{2}\sum_{m=1}^{M}\|\mathbf{u}'(t_m,\mathbf{x})-\sum_{r=1}^{R}\alpha_r'(t_m)\boldsymbol{\varphi}_r'(\mathbf{x})\|_{L^2}^2}{\frac{1}{2}\sum_{m=1}^{M}\|\mathbf{u}'(t_m,\mathbf{x})\|_{L^2}^2}\\ &= \frac{\mathrm{trace}(U^TSU)-\sum_{r=1}^{R}\lambda_r}{\mathrm{trace}(U^TSU)} = 1-\frac{\sum_{r=1}^{R}\lambda_r}{\sum_{r=1}^{M}\lambda_r}.\end{aligned}$$

Fig. 6 shows $\{\lambda_r\}$ and MER_R for the velocity and pressure fluctuations for the three sets of snapshots. It can be observed that all sets of snapshots lead to a similar number of nonzero POD eigenvalues.

Fig. 6 also shows that there are steep decreases in the eigenvalues of the velocity POD modes, e.g., after the second and the sixth mode. Similar jumps can be seen in the eigenvalues of the pressure POD modes after the second, fourth, and eighth mode. Correspondingly, there are strong decreases in the missing energy ratio. It is interesting to note that the velocity and pressure jumps in the eigenvalues and the missing energy ratio seem not to be correlated. This observation supports the point of view that using a different number of velocity and pressure POD modes might be advantageous. The study of this issue, however, is outside the scope of this report and will not be further pursued herein.

4.4. Assessment of the vp-ROMs

This subsection presents an assessment of the effect of the snapshot accuracy on the three vp-ROMs introduced in Section 3, see Table 1.

Theoretical error estimates in [21], see also [29,33], show that the total error in the numerical discretization of velocity-type ROMs consists of three parts: the spatial error due to the finite element discretization, the temporal error due to the time-stepping scheme, and the POD error due to the POD truncation. In the present numerical investigations of vp-ROMs, however, the spatial and temporal error components are constant, since the mesh size and the time step are fixed. Thus, assuming that the ROM estimates in [21,29,33] can be extended to vp-ROMs, for increasing values of R, one expects the

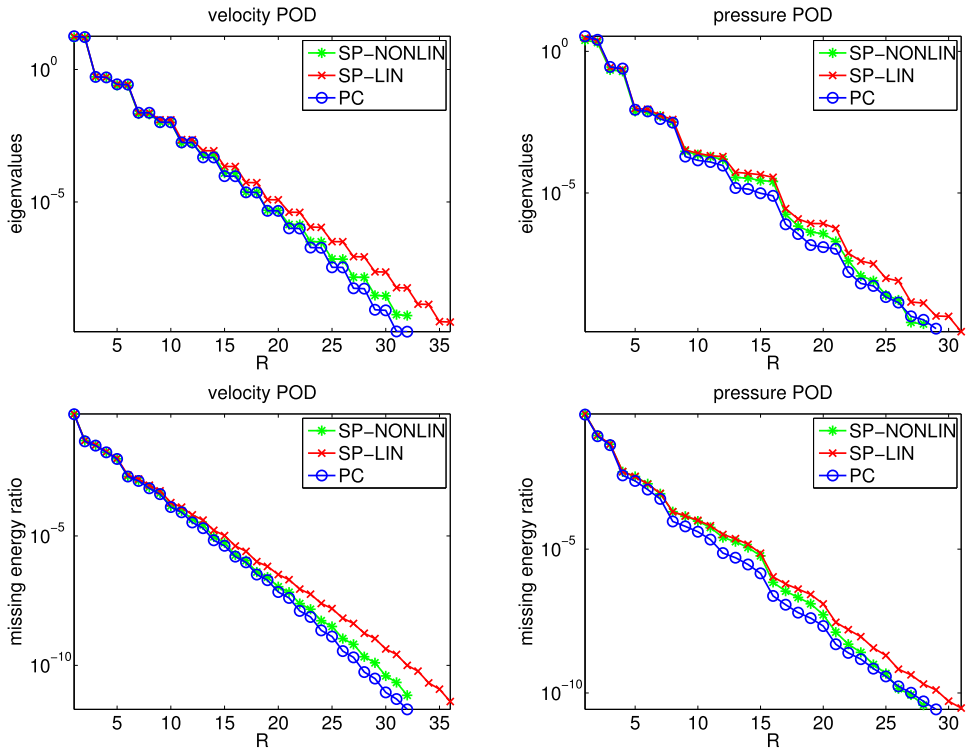

Fig. 6. Eigenvalues and missing energy ratio of the fluctuations (semilogarithmic plot).

POD error component of the vp-ROMs to initially decrease, but then to reach a plateau where the magnitude of the POD error component is the same as or lower than the magnitude of the spatial and temporal error components.

The momentum equation of all three vp-ROMs investigated in this section does not include the pressure term $-(p_{\mathrm{ro}}, \nabla \cdot \boldsymbol{\varphi}_r')$. When the POD modes were computed by solving a saddle point problem (SP-NONLIN, SP-LIN), the motivation was discussed in Section 3.1: since the snapshots are discretely divergence-free, the term $-(p_{\mathrm{ro}}, \nabla \cdot \boldsymbol{\varphi}_r')$ vanishes. In the case of PC, when the snapshots are obtained from a non-divergence-free velocity field, this argument does not hold. The impact of adding the pressure term to the vp-ROMs was numerically tested in this case and it was found that there was no qualitative change in the overall results. Thus, for the sake of brevity, only the results without a pressure term in the momentum equation will be presented.

To assess the behavior of the three vp-ROMs, the time evolution of the drag and lift coefficients, the error in the Strouhal number, the errors in the mean values of the drag and lift coefficients, and the errors in the root mean square (rms) of the drag and lift coefficients were monitored. Here, the errors are defined as the difference between the vp-ROM simulation and the simulation which was used for computing the snapshots. Let $c_{d,\mathrm{meth}}(t)$ denote the drag value computed with a certain numerical method (finite element method or vp-ROM). The rms value is defined by

$$c_{d,\mathrm{rms}} = \left[\frac{1}{N_\tau} \sum_{i=1}^{N_\tau} \left(\bar{c}_{d,\mathrm{meth}} - c_{d,\mathrm{meth}}(t_i) \right)^2 \right]^{1/2},$$

where N_τ is the number of time steps and $\bar{c}_{d,\mathrm{meth}}$ is the mean value of the drag coefficient for the considered method. For the lift coefficient, the rms value is defined analogously. The rms values provide information on the magnitude of the oscillations around the mean value. The frequency of the vortex shedding, needed for the computation of the Strouhal number, was computed using the inverse of the average period of the coefficients.

The IMEX Crank–Nicolson scheme for the velocity ROM (14) was always used with the time step $\tau = 0.005$. All simulations were performed in the time interval $[0, 2]$ and the reference values were computed over five periods for the lift coefficient.

4.4.1. vp-ROMs using snapshots of the highest accuracy

The numerical results for the three vp-ROMs using the snapshots from SP-NONLIN are presented in Figs. 7 and 8.

Fig. 7 displays the time evolution of the drag and lift coefficients. It can be observed that PMB-ROM and SM-ROM are able to reproduce the results of the underlying simulation for the snapshots very well with already $R = 6$ POD modes. In

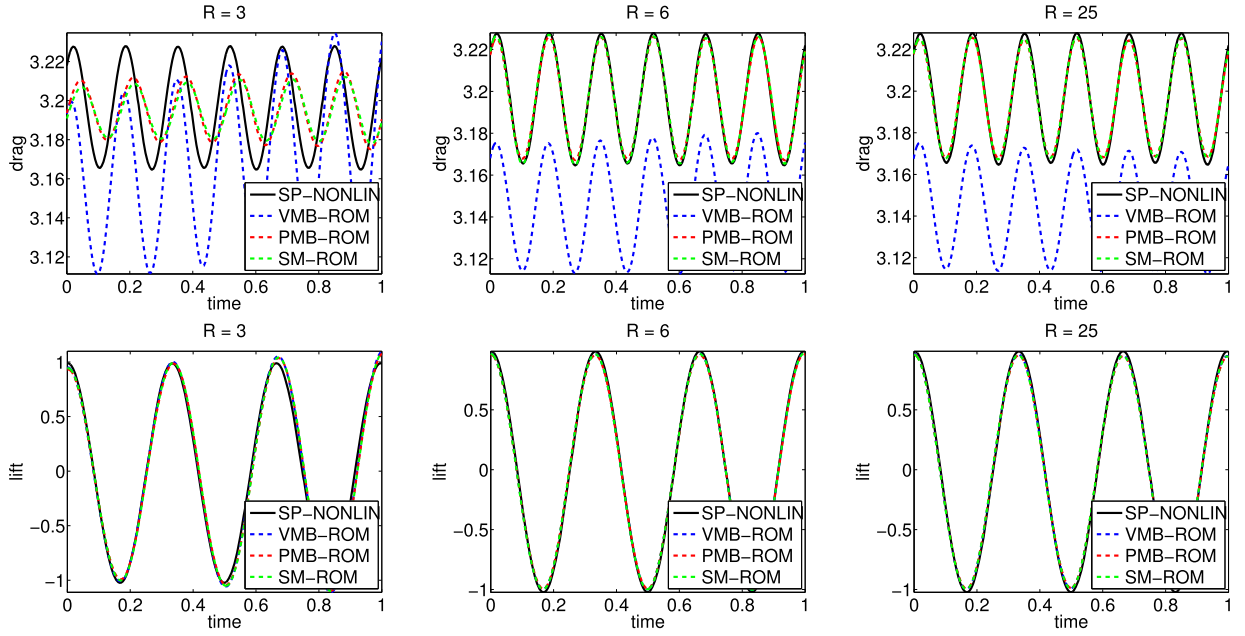

Fig. 7. Snapshots of the highest accuracy (SP-NONLIN): time evolution of the drag coefficient (top) and lift coefficient (bottom).

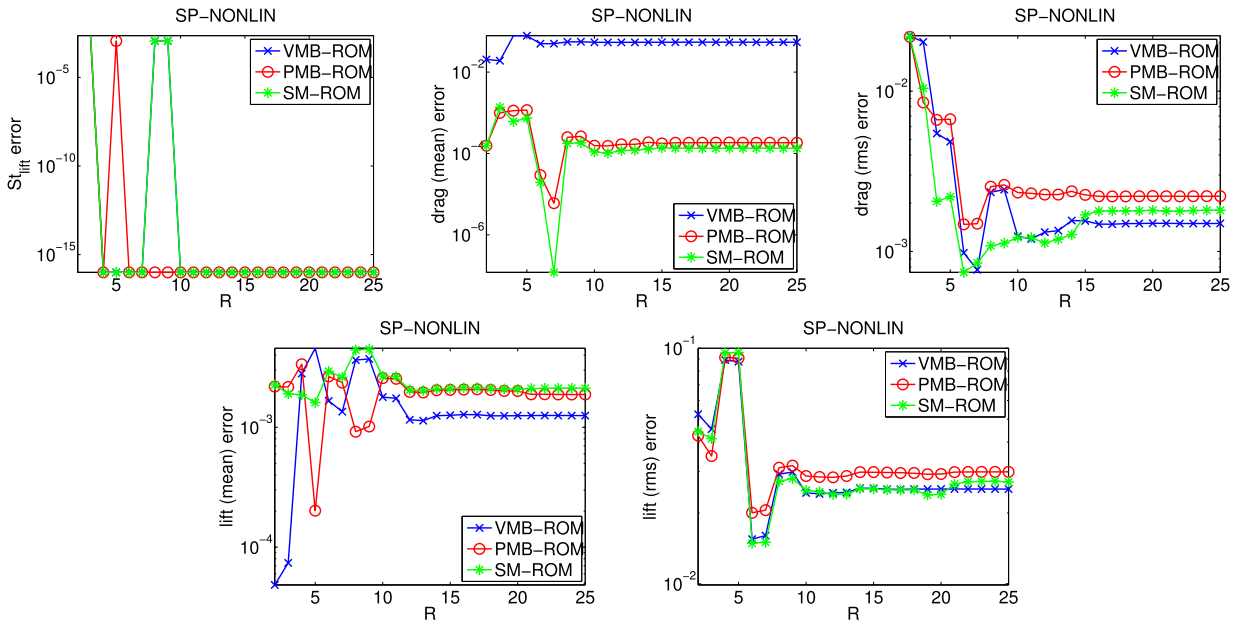

Fig. 8. Snapshots of the highest accuracy (SP-NONLIN): errors in the studied functionals.

contrast, the range of the drag coefficient computed with VMB-ROM is not correct, even for $R = 25$ POD modes. Since the mean drag is often very important in applications, this result reveals a considerable shortcoming of this method.

In more detail, for $R \leqslant 2$, the drag coefficient remained almost constant for all methods (results not shown). Clear improvements in the quality of the reproduction can be seen for PMB-ROM and SM-ROM when going from $R = 5$ to $R = 6$, which corresponds to a jump in the missing energy ratio of the velocity POD modes, see Fig. 6. For values $R \geqslant 6$ both PMB-ROM and SM-ROM yield drag coefficients within the reference intervals given in Table 2. For VMB-ROM, the size of the amplitude of the drag coefficient improved for $R = 6$ modes, but not the mean value of the drag. Even increasing the number of modes to $R = 25$, the mean value of the coefficient stays considerably below the reference. A closer look at the presentations of the drag in Fig. 7 reveals that also its time evolution is not fully periodic since the values of the peaks are slightly changing, which is another shortcoming of the method. Considering the lift coefficient, all three vp-ROMs perform

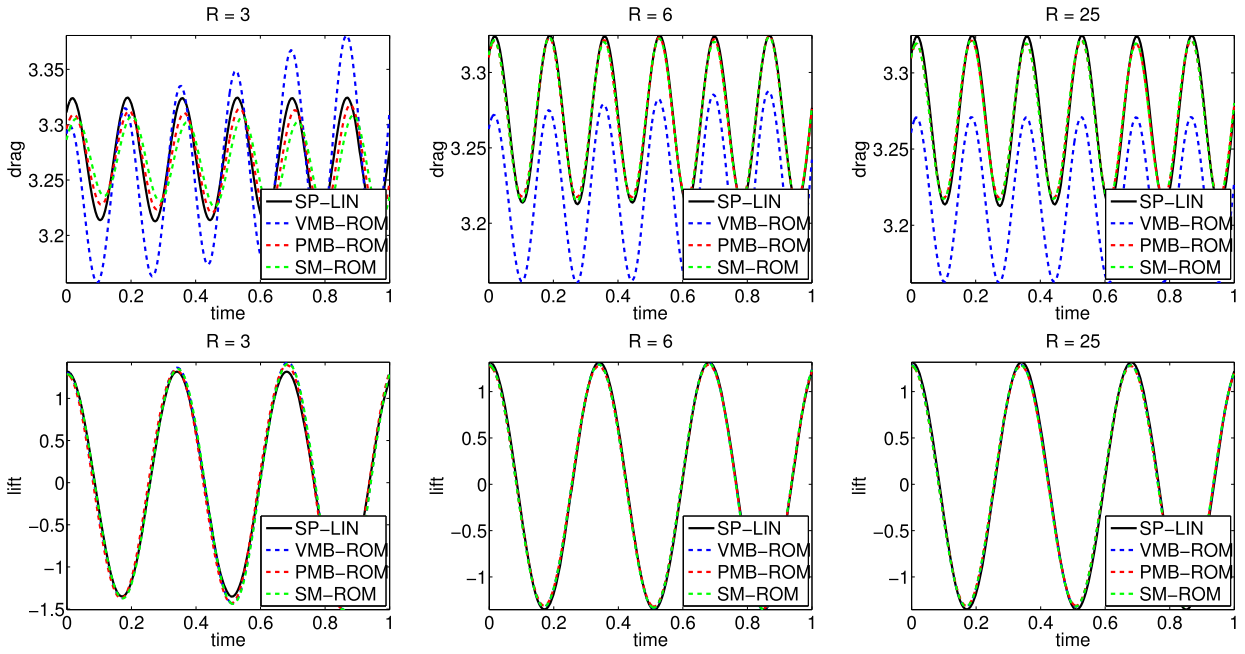

Fig. 9. Snapshots of the intermediate accuracy (SP-LIN): time evolution of the drag and lift coefficients.

well for $R \geqslant 6$. Altogether, despite the simple scheme that was used for the ROMs, very accurate results could be obtained, except for the drag coefficient using VMB-ROM.

To better assess the behavior of vp-ROMs, Fig. 8 displays their errors (semilogarithmic plots) with respect to the values of SP-NONLIN in the Strouhal number, in the mean drag, in the mean lift, in the drag rms, and in the lift rms. As discussed at the beginning of Section 4.4, one expects the POD error to decrease with increasing R, and to reach a level where it is dominated by the spatial and temporal errors. For small values of R, the plots in Fig. 8 follow this trend. Even if the errors are relatively large for very small R, they quickly stabilize around small values for $R \geqslant 6$. Because of the large error of VMB-ROM in the mean drag, we will restrict the assessment of the results from Fig. 8 to the other two methods. It can be seen that PMB-ROM and SM-ROM achieve qualitatively similar results. Often, SM-ROM reproduces the results of the simulations for computing the snapshots somewhat better, in particular for both rms values.

4.4.2. *vp-ROMs using snapshots of intermediate accuracy*

The numerical results for vp-ROMs using the intermediate accuracy snapshots from SP-LIN are presented in Figs. 9 and 10.

In Fig. 9, the good reproduction of the drag and lift computed with PMB-ROM and SM-ROM for $R \geqslant 6$ POD modes as well as the failure of VMB-ROM in the reproduction of the drag coefficient can be clearly seen.

The representation of the errors with respect to the values obtained with SP-LIN in Fig. 10 allows a more detailed assessment of the results. All errors decrease with increasing R until they reach a level where the POD error is dominated by the spatial and temporal errors. For instance, an improvement of the results can be seen when using $R = 6$ instead $R = 5$ POD modes. It can be observed that the reproductions of the mean lift and both rms values are somewhat better for SM-ROM compared with PMB-ROM. For all three vp-ROMs, one obtains for $R \geqslant 10$ the same Strouhal number as for the simulation for computing the snapshots. Again, PMB-ROM and SM-ROM, which use a simple numerical scheme, were able to reproduce the values of the underlying simulation for the snapshots very well.

4.4.3. *vp-ROMs using snapshots of the lowest accuracy*

Figs. 11 and 12 present the results for vp-ROMs using the lowest accuracy snapshots from PC.

The assessment of the results is similar to the other sets of snapshots. Again, PMB-ROM and SM-ROM were able to reproduce the results of the underlying snapshots quite well, whereas VMB-ROM failed for the drag coefficient. In Fig. 11, for $R = 25$, one can even see that the peaks of the drag coefficient computed with VMB-ROM increase notably, such that the computed flow is not periodic. For this set of snapshots, the results obtained with PMB-ROM for the lift coefficient and the rms values are somewhat closer to the values of the underlying snapshot simulation than the results of SM-ROM. Here, the vp-ROMs cannot reproduce the Strouhal number as well as for the other sets of snapshots. For $R \geqslant 10$, the lengths of the period differ by one time step τ.

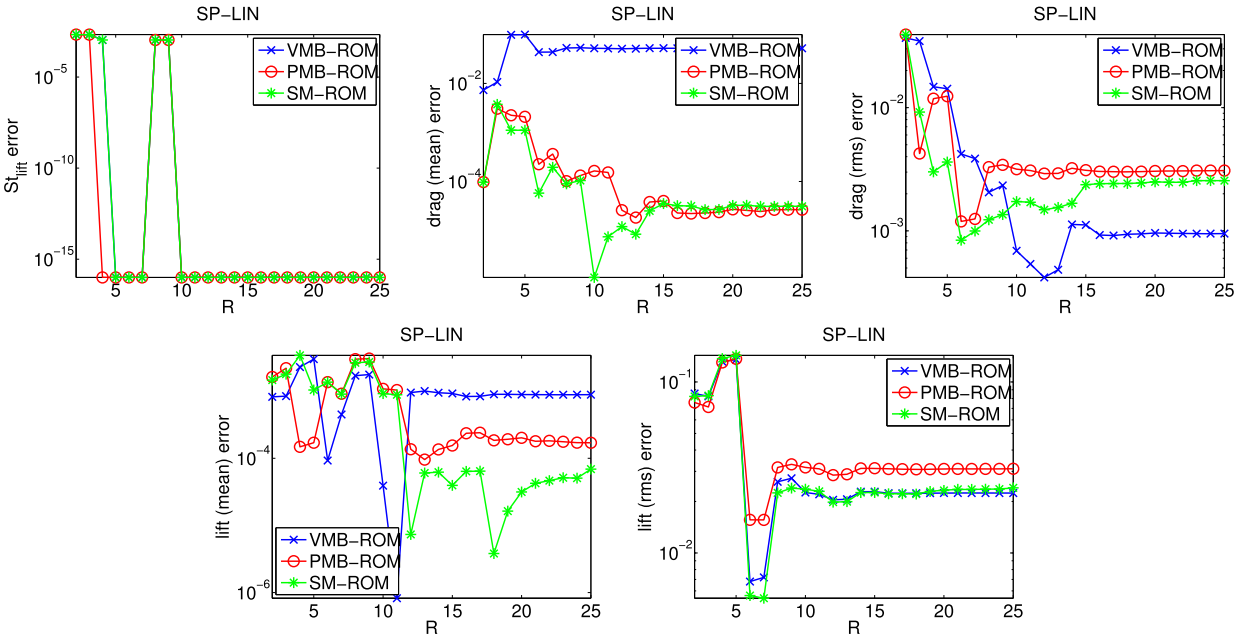

Fig. 10. Snapshots of intermediate accuracy (SP-LIN): errors in the studied functionals.

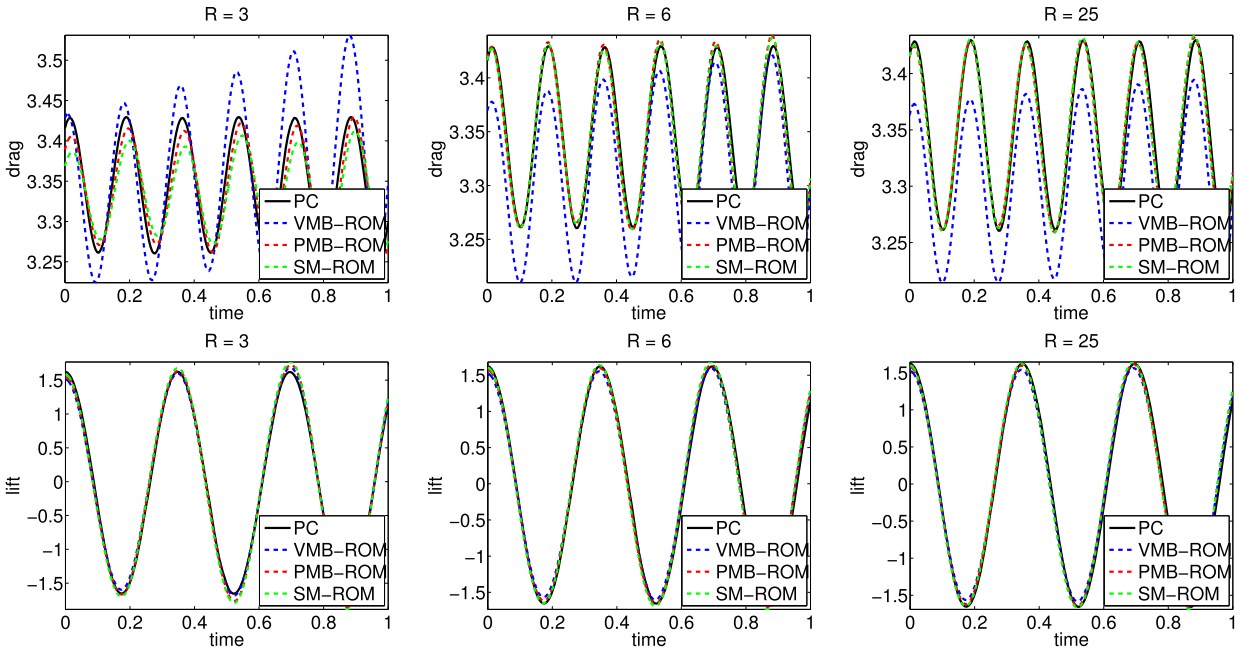

Fig. 11. Snapshots of the lowest accuracy (PC): time evolution of the drag and lift coefficients.

4.5. Computational efficiency

For completeness, this section will discuss the computational efficiency of the vp-ROMs. The computational time of a ROM can be divided into offline and online stages. The offline stage includes the computations that have to be performed only once, before the time iteration loop. The online stage consists of computations that have to be repeated at each time iteration inside the loop. For all three vp-ROMs that were investigated in this report, the ROM velocity was computed the same way. The ROM pressure, however, was computed differently.

The offline stage comprises the computation of the velocity modes and the precomputation of the ROM matrices and right-hand sides, so that the ROM online stage can be performed very fast. The ROM pressure for PMB-ROM and SM-ROM

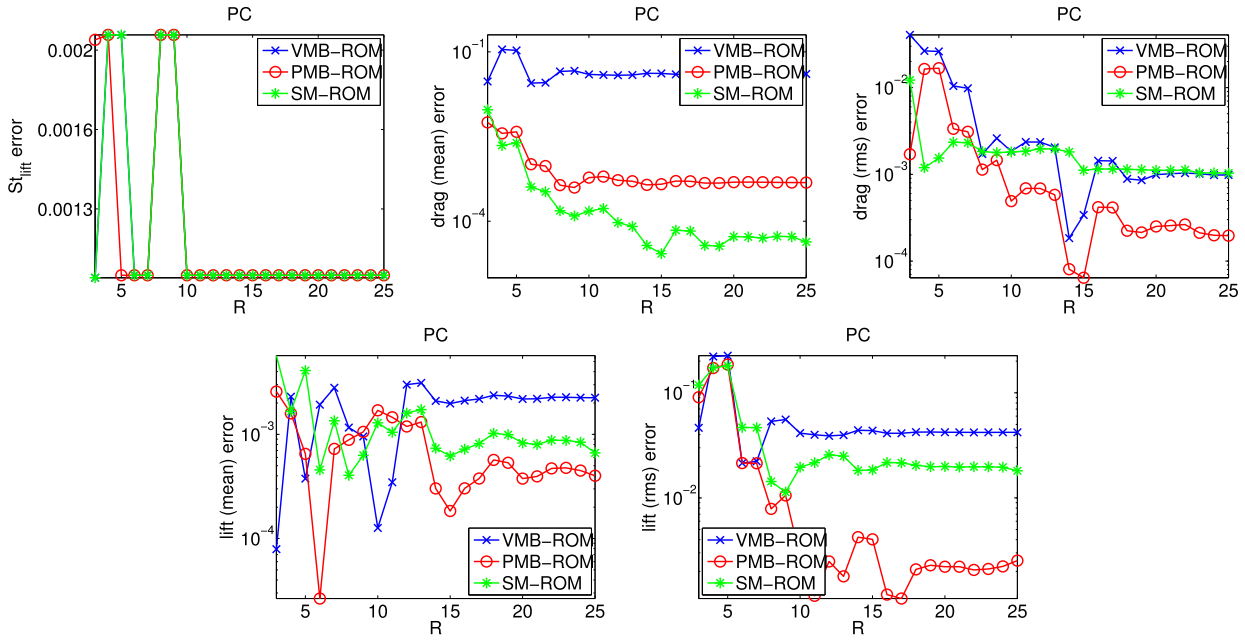

Fig. 12. Snapshots of the lowest accuracy (PC): errors in the studied functionals.

requires the computation of the pressure modes, which represents the most time consuming part of their offline stage, and the assembling and factorization of the matrices in (21) and (27). These procedures are not necessary for VMB-ROM. However, the pressure coefficients $p_{rs}(\mathbf{x})$ in (17) have to be precomputed by solving (18). In our numerical experiments, for $R = 25$, the computation of the $(R+1)R/2$ coefficients $p_{rs}(\mathbf{x})$ took about twice as long as the computation of the R pressure modes. Thus, in the offline stage, the computational costs of PMB-ROM and SM-ROM are lower than those of VMB-ROM.

In the online stage, the main difference is that VMB-ROM does not require the solution of a linear system for the pressure at each iteration, as the pressure is recovered as a linear combination of precomputed functions $p_{rs}(\mathbf{x})$, see (17). Thus, it would seem that the computational cost of VMB-ROM is lower than the computational cost of the two other vp-ROMs. We could observe, however, that this is not the case. In fact, the solution of the $R \times R$ linear system in PMB-ROM and SM-ROM requires only $\mathcal{O}(R^2)$ operations, yielding relatively low computational times. In addition, the cost of recovering the finite element approximation is $\mathcal{O}(RN_p)$, where N_p denotes the dimension of the pressure finite element space. On the other hand, for a given R, the computational complexity for VMB-ROM is $\mathcal{O}(R^2N_p)$, as the approximation of the finite element pressure solution is computed as a linear combination of the functions $p_{rs}(\mathbf{x})$, see (18), which are represented by N_p coefficients. Since $R \ll N_p$, the computational costs for VMB-ROM in the online stage are higher than those for the other two vp-ROMs. In our numerical experiments, in the online stage, the computational times of PMB-ROM and SM-ROM were about the same for moderate values of R ($R < 15$), representing between 0.01% and 0.06% of the computing time of SP-NONLIN. For the same range of values of R, VMB-ROM was computationally more expensive, taking between 0.10% and 1.15% of the time of SP-NONLIN.

5. Summary and outlook

The first goal of this report was to discuss and compare three different velocity–pressure ROMs. VMB-ROM uses only velocity POD modes, whereas PMB-ROM and SM-ROM use pressure POD modes as well. SM-ROM is, to our best knowledge, a novel model. The second goal was to perform the first step in answering the following question: "*How strong are the impacts of the snapshot accuracy, on the one hand, and of the* (*simple*) *numerical method used in the ROMs, on the other hand, onto the ROM results?*"

For studying this question, three sets of snapshots with different accuracy were used. The numerical investigations showed that the snapshots had, irrespectively of the way they were computed and of their accuracy, a much stronger impact than the numerical methods used in the vp-ROMs. Generally, the results of the simulation for computing the snapshots were reproduced quite well with the velocity–pressure ROMs. Therefore, this study clearly supports the approach of performing accurate (and probably time-consuming) simulations for computing the snapshots in order to obtain also accurate results in the ROM simulations.

Concerning the comparisons of the velocity–pressure ROMs, the main conclusion drawn from the numerical investigation is that the two ROMs that utilize pressure modes (PMB-ROM and SM-ROM) were superior, both in terms of reproducing the results of the simulations for computing the underlying snapshots and of efficiency, to the ROM that uses only velocity POD

modes (VMB-ROM). The results obtained with VMB-ROM for an important quantity of interest, the mean drag coefficient, were not satisfactory. SM-ROM could reproduce the results of the simulations for obtaining the snapshots with the highest and intermediate accuracy (SP-NONLIN and SP-LIN) somewhat better than PMB-ROM.

Several research directions will be pursued in future. For instance, we will study whether the conclusions of this report carry over to the case of structure-dominated turbulent flows. In addition, the rigorous numerical analysis for discretizations of the new velocity–pressure ROM (SM-ROM) will be a topic of future research.

Acknowledgements

We would like to acknowledge the anonymous referee whose valuable suggestions helped us to considerably improve this report.

References

[1] Imran Akhtar, Ali H. Nayfeh, Calvin J. Ribbens, On the stability and extension of reduced-order Galerkin models in incompressible flows, Theor. Comput. Fluid Dyn. 23 (2009) 213–237.
[2] David Amsallem, Charbel Farhat, Stabilization of projection-based reduced-order models, Int. J. Numer. Methods Eng. 91 (4) (2012) 358–377.
[3] Jeanne A. Atwell, Belinda B. King, Reduced order controllers for spatially distributed systems via proper orthogonal decomposition, SIAM J. Sci. Comput. 26 (1) (2004) 128–151 (electronic).
[4] Nadine Aubry, Philip Holmes, John L. Lumley, Emily Stone, The dynamics of coherent structures in the wall region of a turbulent boundary layer, J. Fluid Mech. 192 (1) (1988) 115–173.
[5] Joan Baiges, Ramon Codina, Sergio Idelsohn, Explicit reduced-order models for the stabilized finite element approximation of the incompressible Navier–Stokes equations, Int. J. Numer. Methods Fluids 72 (12) (2013) 1219–1243.
[6] M. Balajewicz, E.H. Dowell, Stabilization of projection-based reduced order models of the Navier–Stokes, Nonlinear Dyn. 70 (2012) 1619–1632.
[7] Y. Bazilevs, V.M. Calo, J.A. Cottrell, T.J.R. Hughes, A. Reali, G. Scovazzi, Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows, Comput. Methods Appl. Mech. Eng. 197 (1–4) (2007) 173–201.
[8] M. Bergmann, C.-H. Bruneau, A. Iollo, Enablers for robust POD models, J. Comput. Phys. 228 (2) (2009) 516–538.
[9] M. Braack, E. Burman, V. John, G. Lube, Stabilized finite element methods for the generalized Oseen problem, Comput. Methods Appl. Mech. Eng. 196 (4–6) (2007) 853–866.
[10] M. Buffoni, S. Camarri, A. Iollo, M.V. Salvetti, Low-dimensional modelling of a confined three-dimensional wake flow, J. Fluid Mech. 569 (2006) 141–150.
[11] John Burkardt, Max Gunzburger, Hyung-Chun Lee, POD and CVT-based reduced-order modeling of Navier–Stokes flows, Comput. Methods Appl. Mech. Eng. 196 (1–3) (2006) 337–355.
[12] W. Cazemier, R.W.C.P. Verstappen, A.E.P. Veldman, Proper orthogonal decomposition and low-dimensional models for driven cavity flows, Phys. Fluids 10 (7) (1998) 1685–1699.
[13] Saifon Chaturantabut, Danny C. Sorensen, Nonlinear model reduction via discrete empirical interpolation, SIAM J. Sci. Comput. 32 (5) (2010) 2737–2764.
[14] Saifon Chaturantabut, Danny C. Sorensen, A state space error estimate for POD-DEIM nonlinear model reduction, SIAM J. Numer. Anal. 50 (1) (2012) 46–63.
[15] D. Galbally, K. Fidkowski, K. Willcox, O. Ghattas, Non-linear model reduction for uncertainty quantification in large-scale inverse problems, Int. J. Numer. Methods Eng. 81 (12) (2010) 1581–1608.
[16] B. Galletti, C.H. Bruneau, L. Zannetti, A. Iollo, Low-order modelling of laminar flow regimes past a confined square cylinder, J. Fluid Mech. 503 (2004) 161–170.
[17] Vivette Girault, Pierre-Arnaud Raviart, Finite Element Methods for Navier–Stokes Equations, Springer Series in Computational Mathematics, vol. 5, Springer-Verlag, Berlin, 1986, Theory and algorithms.
[18] J.L. Guermond, P. Minev, Jie Shen, An overview of projection methods for incompressible flows, Comput. Methods Appl. Mech. Eng. 195 (44–47) (2006) 6011–6045.
[19] Max D. Gunzburger, Janet S. Peterson, John N. Shadid, Reduced-order modeling of time-dependent PDEs with multiple parameters in the boundary data, Comput. Methods Appl. Mech. Eng. 196 (4–6) (2007) 1030–1047.
[20] Philip Holmes, John L. Lumley, Gal Berkooz, Turbulence, Coherent Structures, Dynamical Systems and Symmetry, Cambridge Monographs on Mechanics, Cambridge University Press, Cambridge, 1996.
[21] T. Iliescu, Z. Wang, Variational multiscale proper orthogonal decomposition: Convection-dominated convection–diffusion–reaction equations, Math. Comput. 82 (2013).
[22] Angelo Iollo, Alain Dervieux, Jean-Antoine Désidéri, Stéphane Lanteri, Two stable POD-based approximations to the Navier–Stokes equations, Comput. Vis. Sci. 3 (1–2) (2000) 61–66.
[23] E. Jenkins, V. John, A. Linke, Leo G. Rebholz, On the parameter choice in grad-div stabilization for incompressible flow problems, Adv. Comput. Math. (2013), http://dx.doi.org/10.1007/s10444-013-9316-1, in press.
[24] Volker John, On the efficiency of linearization schemes and coupled multigrid methods in the simulation of a 3D flow around a cylinder, Int. J. Numer. Methods Fluids 50 (7) (2006) 845–862.
[25] Volker John, Gunar Matthies, Higher-order finite element discretizations in a benchmark problem for incompressible flows, Int. J. Numer. Methods Fluids 37 (8) (2001) 885–903.
[26] Volker John, Gunar Matthies, MooNMD—a program package based on mapped finite element methods, Comput. Vis. Sci. 6 (2–3) (2004) 163–169.
[27] Volker John, Gunar Matthies, Joachim Rang, A comparison of time-discretization/linearization approaches for the incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Eng. 195 (44–47) (2006) 5995–6010.
[28] Volker John, Joachim Rang, Adaptive time step control for the incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Eng. 199 (9–12) (2010) 514–524.
[29] K. Kunisch, S. Volkwein, Galerkin proper orthogonal decomposition methods for parabolic problems, Numer. Math. 90 (1) (2001) 117–148.
[30] T. Lassila, A. Manzoni, A. Quarteroni, G. Rozza, Model order reduction in fluid dynamics: challenges and perspectives, Technical report 29/2013, Politecnico di Milano, 2013.
[31] C. Leblond, C. Allery, C. Inard, An optimal projection method for the reduced-order modeling of incompressible flows, Comput. Methods Appl. Mech. Eng. 200 (33–36) (2011) 2507–2527.
[32] Alexander Linke, Collision in a cross-shaped domain—a steady 2d Navier–Stokes example demonstrating the importance of mass conservation in CFD, Comput. Methods Appl. Mech. Eng. 198 (41–44) (2009) 3278–3286.

[33] Zhendong Luo, Jing Chen, I.M. Navon, Xiaozhong Yang, Mixed finite element formulation and error estimates based on proper orthogonal decomposition for the nonstationary Navier–Stokes equations, SIAM J. Numer. Anal. 47 (1) (2008/09) 1–19.
[34] H.V. Ly, H.T. Tran, Modeling and control of physical processes using proper orthogonal decomposition, Math. Comput. Model. 33 (1) (2001) 223–236.
[35] B.R. Noack, M. Morzynski, G. Tadmor, Reduced-Order Modelling for Flow Control, vol. 528, Springer-Verlag, 2011.
[36] Bernd, R. Noack, Paul Papas, Peter A. Monkewitz, The need for a pressure-term representation in empirical Galerkin models of incompressible shear flows, J. Fluid Mech. 523 (2005) 339–365.
[37] Alfio Quarteroni, Gianluigi Rozza, Numerical solution of parametrized Navier–Stokes equations by reduced basis methods, Numer. Methods Partial Differ. Equ. 23 (4) (2007) 923–948.
[38] Gianluigi Rozza, Karen Veroy, On the stability of the reduced basis method for Stokes equations in parametrized domains, Comput. Methods Appl. Mech. Eng. 196 (7) (2007) 1244–1260.
[39] M. Schäfer, S. Turek, The benchmark problem “Flow around a cylinder”, in: E.H. Hirschel (Ed.), Flow Simulation with High-Performance Computers II, in: Notes on Numerical Fluid Mechanics, vol. 52, Vieweg, 1996, pp. 547–566.
[40] S. Sirisup, G.E. Karniadakis, A spectral viscosity method for correcting the long-term behavior of POD models, J. Comput. Phys. 194 (1) (2004) 92–116.
[41] Lawrence Sirovich, Turbulence and the dynamics of coherent structures. I. Coherent structures, Q. Appl. Math. 45 (3) (1987) 561–571.
[42] R. Ştefănescu, I.M. Navon, POD/DEIM nonlinear model order reduction of an ADI implicit shallow water equations model, J. Comput. Phys. 237 (2013) 95–114.
[43] J. van Kan, A second-order accurate pressure-correction scheme for viscous incompressible flow, SIAM J. Sci. Stat. Comput. 7 (3) (1986) 870–891.
[44] S. Volkwein, Model Reduction Using Proper Orthogonal Decomposition, Lecture Notes, Faculty of Mathematics and Statistics, University of Konstanz, 2011.
[45] Z. Wang, I. Akhtar, J. Borggaard, T. Iliescu, Two-level discretizations of nonlinear closure models for proper orthogonal decomposition, J. Comput. Phys. 230 (1) (2011) 126–146.
[46] Zhu Wang, Imran Akhtar, Jeff Borggaard, Traian Iliescu, Proper orthogonal decomposition closure models for turbulent flows: a numerical comparison, Comput. Methods Appl. Mech. Eng. 237 (240) (2012) 10–26.
[47] J. Weller, E. Lombardi, M. Bergmann, A. Iollo, Numerical methods for low-order modeling of fluid flows based on POD, Int. J. Numer. Methods Fluids 63 (2) (2010) 249–268.
[48] Shangyou Zhang, A new family of stable mixed finite elements for the 3D Stokes equations, Math. Comput. 74 (250) (2005) 543–554.