Weierstraß-Institut
für Angewandte Analysis und Stochastik
Leibniz-Institut im Forschungsverbund Berlin e. V.

Preprint ISSN 0946 – 8633

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

Alfonso Caiazzo[1], Traian Iliescu [2], Volker John [1 3] , Swetlana Schyschlowa [1]

submitted: February 1, 2013

[1] Weierstrass Institute
Mohrenstr. 39
10117 Berlin
Germany
email: Alfonso.Caiazzo@wias-berlin.de
email: Volker.John@wias-berlin.de
email: Swetlana.Schyschlowa@wias-berlin.de

[2] Virginia Tech
Department of Mathematics
456 McBryde Hall
Blacksburg, VA 24061-0123
U.S.A.
email: iliescu@vt.edu

[3] Free University of Berlin
Department of Mathematics and Computer Science
Arnimallee 6
14195 Berlin
Germany

No. 1765
Berlin 2013

2010 *Mathematics Subject Classification.* 76D05.

Key words and phrases. Navier–Stokes equations, proper orthogonal decomposition, velocity-pressure reduced order models, snapshot accuracy.

Edited by
Weierstraß-Institut für Angewandte Analysis und Stochastik (WIAS)
Leibniz-Institut im Forschungsverbund Berlin e. V.
Mohrenstraße 39
10117 Berlin
Germany

Fax: +49 30 20372-303
E-Mail: preprint@wias-berlin.de
World Wide Web: http://www.wias-berlin.de/

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 ROMs accuracy. Numerical studies are performed on a two-dimensional laminar flow past a circular obstacle. It turns out that, both in terms of accuracy and efficiency, the two ROMs that utilize pressure modes are clearly superior to the ROM that uses only velocity modes. The numerical results also show a strong correlation of the accuracy of the snap shots with the accuracy of the ROMs.

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 $p : (0, T] \times \Omega \to \mathbb{R}$

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

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

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

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

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

Furthermore, let $(\cdot, \cdot)$ denote the standard inner product in $L^2(\Omega)$ and let $\mathbf{u}_\mathbf{g}(t, \cdot) \in H^1(\Omega)$ 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 $\mathbf{u} - \mathbf{u}_\mathbf{g} \in V$ for all t, and $p \ : \ (0, T] \to Q$ such that

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

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 (ROM) 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. [3,9,10,11,13,12,15,20,21,23,31,35,41]. Here, the snapshots will be obtained from detailed numerical simulations. It is worth noticing that generally the snapshots might even come from experimental data.

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 paper investigates the impact of the snapshot accuracy on the vp-ROM results. The motivation and background for these two numerical investigations are presented in the following.

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., the simulation of fluid-structure interaction problems and the computation of relevant quantities, such as drag and lift coefficients on solid bodies, and for ROM simulations of shear flows [33]. Other reasons for including the pressure are connected to the definition of ROMs. Using only the velocity ROM leads to a comparatively simple model that can be simulated very efficiently. The rationale behind the velocity ROM, 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. However, most 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. Another reason, already pointed out in [7], is that the availability of the pressure enables the computation of ROM residuals. 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. And finally, the Navier–Stokes equations are sometimes equipped with boundary conditions that include the pressure such that the pressure might be necessary for this reason.

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 pressure that only uses the velocity POD modes [16,33]. One 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) [33], or together with them (i.e., the coupled, monolithic approach) [7,42]. 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 PCM-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. Overall, three vp-ROMs will be considered in the numerical studies. VMB-ROM and PCM-ROM solve the same equation for the pressure but in different finite-dimensional spaces. PCM-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. To motivate this investigation, we recall that, in order

to ensure a high computational efficiency, ROMs generally employ simple, and therefore potentially inaccurate, numerical methods. Indeed, ROMs generally avoid the solution of nonlinear systems and use, where possible, explicit time integration schemes, see, e.g., [4] for a detailed discussion. 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 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 is the impact of the snapshot accuracy onto the accuracy of the ROM results?"*

In this report, we will perform a first step in numerically investigating the effect of the snapshot accuracy on the ROM accuracy. To the best of the authors' knowledge, this represents the first such numerical investigation. 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 [21,32,36,40].

Let us 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 $\{\alpha_r(t), \alpha_r : [0, T] \to \mathbb{R}\}_{r=1}^R$, $\{\phi_r(x), \phi_r : \Omega \to \mathbb{R}^d\}_{r=1}^R$ 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 $t_1, \ldots, t_M$, 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 only that the norm is induced by an inner product $(\cdot, \cdot)$. 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 and let $\{v_n(\mathbf{x})\}_{n=1}^N$ be the nodal basis with $v_n(\mathbf{x}_k) = \delta_{nk}$. Then, the representations have the form

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

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 $R \leq \operatorname{rank}(U)$. In practice, it is usually $M \ll N$, which will be assumed also here. The function $\phi_r(\mathbf{x})$ will be identified with its vector of coefficients $(\phi_r(\mathbf{x}_n))_{n=1}^N$.

Inserting the representations (4) into (3) and using that the norm is induced by an inner product, the approximation problem can be written as follows

$$\arg\min_{(\alpha_r,\phi_r)} \sum_{m=1}^{M} \tau\Big[\sum_{n=1}^{N}\sum_{l=1}^{N} \Big(u(t_m,\mathbf{x}_n) - \sum_{r=1}^{R} \alpha_r(t_m)\phi_r(\mathbf{x}_n)\Big) \\ \times\Big(u(t_m,\mathbf{x}_l) - \sum_{r=1}^{R} \alpha_r(t_m)\phi_r(\mathbf{x}_l)\Big)(v_n,v_l)\Big]. \tag{5}$$

The matrix S with $S_{ln} = (v_n, v_l)$ is symmetric and positive definite. For functions v, w that can be represented in the same form as (4), the inner product can be written as

$$(v, w) = v^T S w, \tag{6}$$

where on the right-hand side there are the vectors of the coefficients of v and w. It is clear that the result of (5) does not depend on τ, such that without loss of generality one can consider $\tau = 1$.

Problem (5) is an optimization problem in $\mathbb{R}^M \times \mathbb{R}^N$. Let $\{\phi_r\}_{r=1}^R$ be an orthogonal set of vectors from $\mathbb{R}^N$ with respect to the inner product (6) with $\|\phi_r\| = 1$. From Hilbert space theory it is known that then

$$\alpha_r(t_m) = (u(t_m,\cdot),\phi_r) = U_m^T S \phi_r, \tag{7}$$

where U_m is the m-th column of U. Inserting this expression into (5) and using the orthonormality of $\{\phi_r\}_{r=1}^R$, problem (5) can be reformulated as

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

Since the first term is a given constant, (5) becomes equivalent to maximizing the second term. The Lagrangian functional of this optimization problem has the form

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

Optimal values can be obtained only at the stationary points

$$0 = \partial_{\phi_r}\mathcal{L} = 2\sum_{m=1}^{M}(u(t_m,\cdot),\phi_r)(u(t_m,\cdot),\psi) - 2\lambda_r(\phi_r,\psi), \quad \forall\ \psi\in\mathbb{R}^N, \\ 0 = \partial_{\lambda_r}\mathcal{L} = (\phi_r,\phi_r) - 1,$$

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

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

for all $\psi \in \mathbb{R}^N$. This equality holds if and only if $\sum_{m=1}^{M}(u(t_m,\cdot),\phi_r)u(t_m,\cdot) = \lambda_r\phi_r$, which results in the eigenvalue problem in $\mathbb{R}^N$

$$UU^TS\phi_r = \lambda_r\phi_r. \tag{8}$$

Multiplying (8) from the left-hand side with $S^{1/2}$, 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 $R \leq \mathrm{rank}(U)$. 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. However, using (7) and multiplying (8) by U^TS from the left-hand side leads to an eigenvalue problem in R^M

$$U^TSU\alpha_r = \lambda_r\alpha_r, \quad U^TSU \in \mathbb{R}^{M\times M}, \tag{9}$$

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

$$\phi_r = \frac{U\alpha_r}{\|U\alpha_r\|} = \frac{U\alpha_r}{(\alpha_r^TU^TSU\alpha_r)^{1/2}} = \frac{U\alpha_r}{\lambda_r^{1/2}(\alpha_r^TU^TU\alpha_r)^{1/2}}, \quad r = 1,\ldots,R, \tag{10}$$

one obtains with (9)

$$UU^TS\phi_r = \frac{1}{\|U\alpha_r\|}U\left(U^TSU\alpha_r\right) = \frac{1}{\|U\alpha_r\|}U\lambda_r\alpha_r = \lambda_r\phi_r.$$

Thus, (λ_r,ϕ_r) with ϕ_r given by (10) solves (8). The approach of computing the eigenvalues λ_r by solving (9) and the eigenvectors or modes ϕ_r by (10) is called method of snapshots. It was first proposed in [36].

In practice, 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, obtaining 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 the values $u'(t_m,\mathbf{x})$ 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, e.g., see [2,5,37,41] for recent publications. In the case of the Navier–Stokes equations, the solution of the problem consists of two components, velocity and pressure $(\mathbf{u}, p)$. 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. 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 spatial basis obtained by applying the POD to 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 (2) yields the following ROM: Find $(\mathbf{u}_{\mathrm{ro}}, p_{\mathrm{ro}})$ with $\mathbf{u}_{\mathrm{ro}} - \overline{\mathbf{u}} \ : \ (0, T] \to \mathrm{span}\{\boldsymbol{\varphi}'_r\}_{r=1}^R$ and $p_{\mathrm{ro}} - \overline{p} \ : \ (0, T] \to \mathrm{span}\{\psi'_r\}_{r=1}^R$ such that for $r = 1, \ldots, R$,

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

and $\mathbf{u}_{\mathrm{ro}}(0, \mathbf{x})$ is an approximation of the initial condition with the POD modes.

3.1 Velocity ROM

In many, probably even most, published reports on ROMs for incompressible flows, only a ROM for the velocity is 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 $\overline{\mathbf{u}}(\mathbf{x})$, 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 in (12) and the continuity equation in (12) drop out. A pressure term is still needed if the considered problem possesses boundary conditions (near regions of interest) which include the pressure. If this is not the case, the ROM for the velocity based on the Galerkin projection has the following form: Find

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

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

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

Note that (13) requires only a POD for velocity snapshots. Furthermore, the mass matrix $(\boldsymbol{\varphi}'_s, \boldsymbol{\varphi}'_r)$, $s, r \geq 1$, 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 [43]. Most of the inf-sup stable pairs, in particular the most popular ones like the Taylor–Hood finite element, are only discretely divergence-free. The magnitude of the divergence of the finite element solution can be even large. Indeed, the standard finite element convergence theory shows that the $L^2(\Omega)$ norm of the divergence has the same order as the error in the $L^2(\Omega)$ norm of the velocity gradient.

The reduction from (12) to (13), however, can be achieved in certain situations by using the argument that the snapshots are discretely divergence-free. This situation holds if the finite element mass conservation 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 mass conservation equation is disturbed by additional terms, as in the case of finite element pairs that do not fulfill the 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 requirement for ROMs is the computational efficiency. For this reason, one usually avoids complicated and time-consuming numerical methods within the ROM framework, see [4] for a detailed discussion. 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 $\{t^k\}_{k=1}^K$, 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

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

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

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

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

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

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 if they use pressure POD modes or not. If pressure modes are utilized, 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 POD 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 is built considering only a POD basis for velocity, the pressure field must be reconstructed a posteriori. There are several proposals in the literature on how to utilize the velocity POD modes to compute a ROM for the pressure [16,33]. These approaches are based on the pressure Poisson equation

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

which is obtained by taking the divergence of the momentum equation of the Navier–Stokes equations (1). Equation (15) is equipped with Dirichlet boundary conditions on Γ_{out} and homogeneous Neumann boundary conditions on all other boundaries. The main idea used in [16,33] 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

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

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

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

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

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

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. Thus, the ROM pressure component $p_{\text{ro}}(\mathbf{x})$ can be efficiently computed at each time step by using (17). In [33], 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.

Pressure models that are linear in $\alpha_r'(t)$ were proposed also in [16,33]. In both approaches, minimization problems have to be solved for determining the coefficients in the ansatz. In our opinion, these methods are less clear than the approach leading to the VMB-ROM. For this reason, only the VMB-ROM will be considered in the numerical investigations in Section 4.

3.2.2 A velocity-pressure ROM with pressure-correction scheme (PCM-ROM)

A different approach for incorporating a pressure into a ROM for incompressible flows consists in considering a pressure-correction projection scheme [14,38]. This semi-implicit scheme consists of two sub-steps. First, a velocity field $\tilde{\mathbf{u}}_{\text{ro}}^{k+1}$ is computed by solving the convection-diffusion equation

$$\frac{1}{\tau}(\tilde{\mathbf{u}}_{\text{ro}}^{k+1} - \mathbf{u}_{\text{ro}}^{k}) - \nu\Delta\tilde{\mathbf{u}}_{\text{ro}}^{k+1} + (\tilde{\mathbf{u}}_{\text{ro}}^{k}\cdot\nabla)\tilde{\mathbf{u}}_{\text{ro}}^{k+1} = \mathbf{0} \quad \text{in } \Omega, \tag{19}$$

subject to the boundary conditions

$$\tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1} = \mathbf{g}(\mathbf{x}) \text{ at } \Gamma_{\mathrm{in}}, \quad \tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1} = \mathbf{0} \text{ at } \Gamma_0, \quad \nu \nabla \tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1} \mathbf{n} = \mathbf{0} \text{ at } \Gamma_{\mathrm{out}}.$$

The use of $\tilde{\mathbf{u}}_{\mathrm{ro}}^{k}$ as convection field is supported by the error analysis from [19], see also the discussion of this topic in [18]. In the second step, the velocity $\tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1}$ is projected onto a divergence-free space, obtaining an incompressible velocity field and a corresponding pressure, by solving

$$\frac{1}{\tau}(\mathbf{u}_{\mathrm{ro}}^{k+1} - \tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1}) + \nabla p_{\mathrm{ro}}^{k+1} = 0, \quad \nabla \cdot \mathbf{u}_{\mathrm{ro}}^{k+1} = 0 \quad \text{in } \Omega, \tag{20}$$

equipped with the boundary condition $\mathbf{u}_{\mathrm{ro}}^{k+1} \cdot \mathbf{n} = 0$ on $\partial\Omega$. By taking the divergence of the first equation of (20), the projection step (20) can be reformulated as a Poisson equation for the pressure

$$-\Delta p_{\mathrm{ro}}^{k+1} = -\frac{1}{\tau} \nabla \cdot \tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1} \quad \text{in } \Omega, \tag{21}$$

with a homogeneous Dirichlet boundary condition on Γ_{out} and a homogeneous Neumann boundary condition on $\partial\Omega \setminus \Gamma_{\mathrm{out}}$. The incompressible velocity $\mathbf{u}_{\mathrm{ro}}^{k+1}$ can be computed a posteriori, after having solved the pressure Poisson equation (21), via

$$\mathbf{u}_{\mathrm{ro}}^{k} = \tilde{\mathbf{u}}_{\mathrm{ro}}^{k} - \tau \nabla p_{\mathrm{ro}}^{k}. \tag{22}$$

In practice, substituting (22) into (19), the time iteration is formulated only in terms of $(\tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1}, p_{\mathrm{ro}}^{k+1})$.

The projection scheme (19)–(21) belongs to the family of the so-called pseudo-compressible methods. Note that $\tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1}$ is not discretely divergence-free and that the pressure equation depends on the compressibility of the velocity field. For this reason, equation (21) is generally not suited to compute a ROM pressure if the POD basis for velocity is computed from a set of (discretely) divergence-free snapshots.

The pressure correction method, however, suggests an alternative approach for deriving a pressure equation. Taking the divergence of (19), assuming $\mathbf{u}_{\mathrm{ro}}^{k}$ to be divergence-free and $-\nu \nabla \cdot \Delta \tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1}$ to be negligible, one obtains

$$\frac{1}{\tau} \nabla \cdot \tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1} = -\nabla \cdot \left((\tilde{\mathbf{u}}_{\mathrm{ro}}^{k} \cdot \nabla) \tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1} \right) \quad \text{in } \Omega.$$

Assuming that $\tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1}$ has been already computed, $\tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1}$ can be used as convection. Then, the pressure Poisson equation (21) can be written in the form

$$-\Delta p_{\mathrm{ro}}^{k+1} = \nabla \cdot \left((\tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1} \cdot \nabla) \tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1} \right) \quad \text{in } \Omega. \tag{23}$$

In [1], it was suggested to compute the ROM pressure by applying the Galerkin projection to (23) on the pressure POD modes $\{\psi_r'\}_{r=1}^{R}$. This suggestion leads

to the following method: Find

$$p_{\mathrm{ro}}^{k+1} = \overline{p} + \sum_{r=1}^{R} (\beta_r')^{k+1} \psi_r' \,, \tag{24}$$

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

$$\left(\nabla p_{\mathrm{ro}}^{k+1}, \nabla \psi_r'\right) = \left(\nabla \cdot \left((\tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1} \cdot \nabla)\tilde{\mathbf{u}}_{\mathrm{ro}}^{k+1}\right), \psi_r'\right) \tag{25}$$

with a homogeneous Dirichlet boundary condition on Γ_{out} and a homogeneous Neumann boundary condition on $\partial\Omega \backslash \Gamma_{\mathrm{out}}$. In the numerical studies, the ROM (14) together with (25), will be referred to as the PCM-ROM (pressure-correction-motivated ROM).

Note that (23) is formally equivalent to (15). Although the derivations of VMB-ROM and PCM-ROM are based on the same equation, the ROM pressures of the two ROMs are computed using different discrete spaces. In the VMB-ROM, the ROM pressure is represented in terms of the velocity POD modes, see (18), whereas in the PCM-ROM the ROM pressure is represented in terms of the pressure POD modes, cf. (24).

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. In the latter, 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. 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. Based on this argument, we do not expect that an inf-sup condition is satisfied for ROMs if the dimension of both spaces is equal to R. But even for using different dimensions, to the authors' best knowledge, there are no results on how to establish an inf-sup condition. For this reason, a stabilization with respect to a possible violation of the inf-sup condition should probably be included into a coupled velocity-pressure ROM. Among the stabilizations for incompressible flow problems [8], the class of residual-based approaches seems to be promising

in our opinion, since these immediately allow in addition the stabilization of dominant convection. These approaches are also the basis of residual-based variational multiscale methods [6].

A popular residual-based stabilization is the SUPG/PSPG/grad-div method, see [8] 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

$$s_h(\mathbf{u},\mathbf{v},p,q) = \sum_{K\in\mathcal{T}^h} \Big(\partial_t\mathbf{u} - \nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p - \mathbf{f}, \delta_{K,\mathbf{u}}(\mathbf{u}\cdot\nabla)\mathbf{v} + \delta_{K,p}\nabla q\Big)_K , \tag{26}$$

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

$$\sum_{K\in\mathcal{T}^h} \Big(\nabla\cdot\mathbf{u}, \mu_K\nabla\cdot\mathbf{v}\Big)_K , \tag{27}$$

where μ_K denotes the stabilization parameter function. The SUPG term in (26) accounts for stabilizing dominating convection, the grad-div term (27) accounts for improving the discrete conservation of mass, and the PSPG term in (26) accounts for stabilizing a violated inf-sup condition.

Note that the SUPG/PSPG/grad-div method has already been used in [7,42] within a ROM framework. However, in [7,42] the ROM pressure was not computed by solving a separate pressure equation.

One of the main requirements for a ROM is computational efficiency. From this point of view, an explicit treatment of (26) and (27) is advantageous, see [4]. On the other hand, the stabilization of the inf-sup condition has to appear in the system matrix in order to become effective. Thus, the corresponding term has to be treated implicitly.

In the residual for the momentum balance (26), the viscous term is generally neglected, since it is of little importance in the interesting case of small viscosity. Denote by

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

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 stabiliza-

tion terms

$$-\sum_{K\in\mathcal{T}^h}\delta_{K,\mathbf{u}}\Big(\mathbf{res}_{\rm ro}^k,(\mathbf{u}_{\rm ro}^k\cdot\nabla)\boldsymbol{\varphi}_r'\Big)_K,-\sum_{K\in\mathcal{T}^h}\mu_K\Big(\nabla\cdot\mathbf{u}_{\rm ro}^k,\nabla\cdot\boldsymbol{\varphi}_r'\Big)_K,\ r=1\ldots,R, \tag{28}$$

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

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

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

$$-\sum_{K\in\mathcal{T}^h}\delta_{K,p}\Big(\mathbf{res}_{\rm ro}^{k+1}-\nabla p_{\rm ro}^{k+1},\nabla\psi_r'\Big)_K,\quad r=1,\ldots,R, \tag{29}$$

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

$$\begin{pmatrix}A_{\rm ro} & B_{\rm ro}^T\\ B_{\rm ro} & C_{\rm ro}\end{pmatrix}, \tag{30}$$

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

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

Consider now the ROM matrix (30) 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 $B_{\rm ro}$ vanishes. Hence, instead of solving a coupled system with matrix (30), one has to solve two decoupled equations. After having computed the velocity, the right hand side (29) of the continuity equation can be evaluated. If the stabilizations of dominating convection and of violating the mass conservation (28) can be neglected, as for the flow problem considered in Section 4, the velocity equation corresponding to (30) is the same as that in the velocity ROM (13). For the pressure equation corresponding to (30), the matrix is the discretization of a scaled Laplacian. If all stabilization parameters $\{\delta_{K,p}\}$ were the same, a scaling would lead to the same matrix as in (23).

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

$$\sum_{K\in\mathcal{T}^h} \delta_{K,p}\Big(\nabla p_{\rm ro}^{k+1}, \nabla\psi_r'\Big)_K = \\ = -\sum_{K\in\mathcal{T}^h} \delta_{K,p}\left(\frac{\mathbf{u}_{\rm ro}^{k+1}-\mathbf{u}_{\rm ro}^{k}}{\tau} + (\mathbf{u}_{\rm ro}^{k+1}\cdot\nabla)\mathbf{u}_{\rm ro}^{k+1} - \mathbf{f}_{\rm ro}^{k+1}, \nabla\psi_r'\right)_K, \quad r=1,\ldots,R. \tag{31}$$

Below, the ROM (14) together with (31), will be referred to as SM-ROM (stabilization-motivated ROM). The SM-ROM (14), (31) is, to the best of the authors' knowledge, new.

In (31), 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. Thus, by analogy with the finite element setting, the SM-ROM most probably does not satisfy the inf-sup condition and, thus, is prone to numerical instability. For this case, following the finite element theory, we used $\delta_{K,p} = C\,h_K$ in (31), where C is a generic constant and h_K is the diameter of the mesh cell K, [8]. Note that the value of the constant C has no effect on the SM-ROM, since it appears on both sides of (31). Thus, without loss of generality, we used $\delta_{K,p} = h_K$. It is also worth emphasizing that the derivation of the SM-ROM does not rely on the velocity snapshots being divergence-free.

4 Numerical studies

First, this section presents numerical results for the three vp-ROMs introduced in Section 3. Second, it investigates the impact of the snapshot accuracy on the vp-ROM accuracy. 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 [34]. This problem is given in

$$\Omega = \{(0,2.2)\times(0,0.41)\}\setminus\{\mathbf{x}\ :\ (\mathbf{x}-(0.15,0.15))^2 \le 0.05^2\},$$

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.

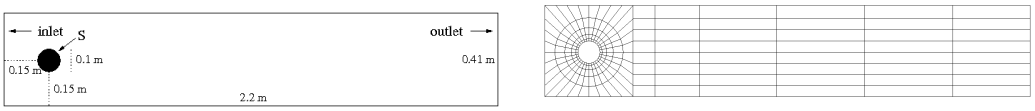

Fig. 1. The flow domain (left) and the coarse grid (right).

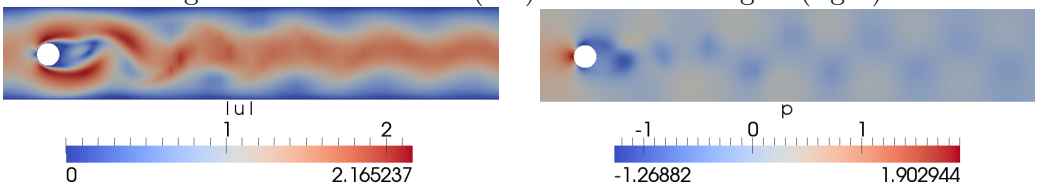

Fig. 2. Snapshots of the finite element solution.

Reference values of the drag and lift coefficients at the cylinder and corresponding reference intervals were defined in [34], see Table 1. These functionals were computed as described in [24,25]. In the periodic regime, another important characterization is the Strouhal number St, which is correlated to the frequency of the vortex shedding. 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 vp-ROM accuracy. As already mentioned in the introduction, 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 [18,39]. 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 which satisfies the boundary conditions is used.

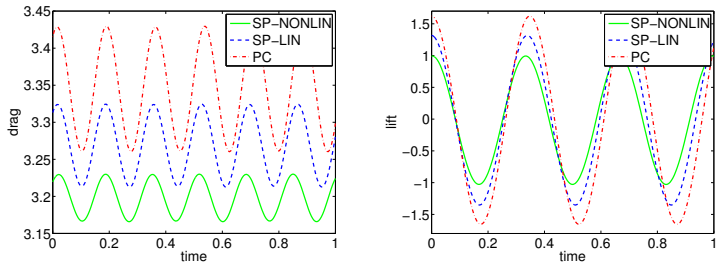

Fig. 3. Drag and lift coefficients for the finite element simulations.

Clearly, the three different numerical methods possess different numerical costs. In the simulations for computing the snapshots, SP-NONLIN took about 1.8 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 1. 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 1. 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 the results computed with PC. Accordingly, we obtained three sets of snapshots: very accurate ones, moderately accurate ones, and inaccurate ones.

4.3 Impact of the snapshot accuracy on the POD modes

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

	$c_d^{\max}$	$c_l^{\max}$	St
SP-NONLIN	3.23	1.02	0.302
SP-LIN	3.32	1.35	0.294
PC	3.43	1.65	0.288
reference results from [34]	$[3.22, 3.24]$	$[0.98, 1.02]$	$[0.295, 0.305]$

Table 1
Maximal drag coefficient, maximal lift coefficient, and Strouhal number for the finite element simulations.

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 first POD modes of the velocity and pressure fluctuations, respectively. For clarity of presentation, only the most accurate (SP-NONLIN) and the most inaccurate (PC) numerical methods are considered. Both Fig. 4 and

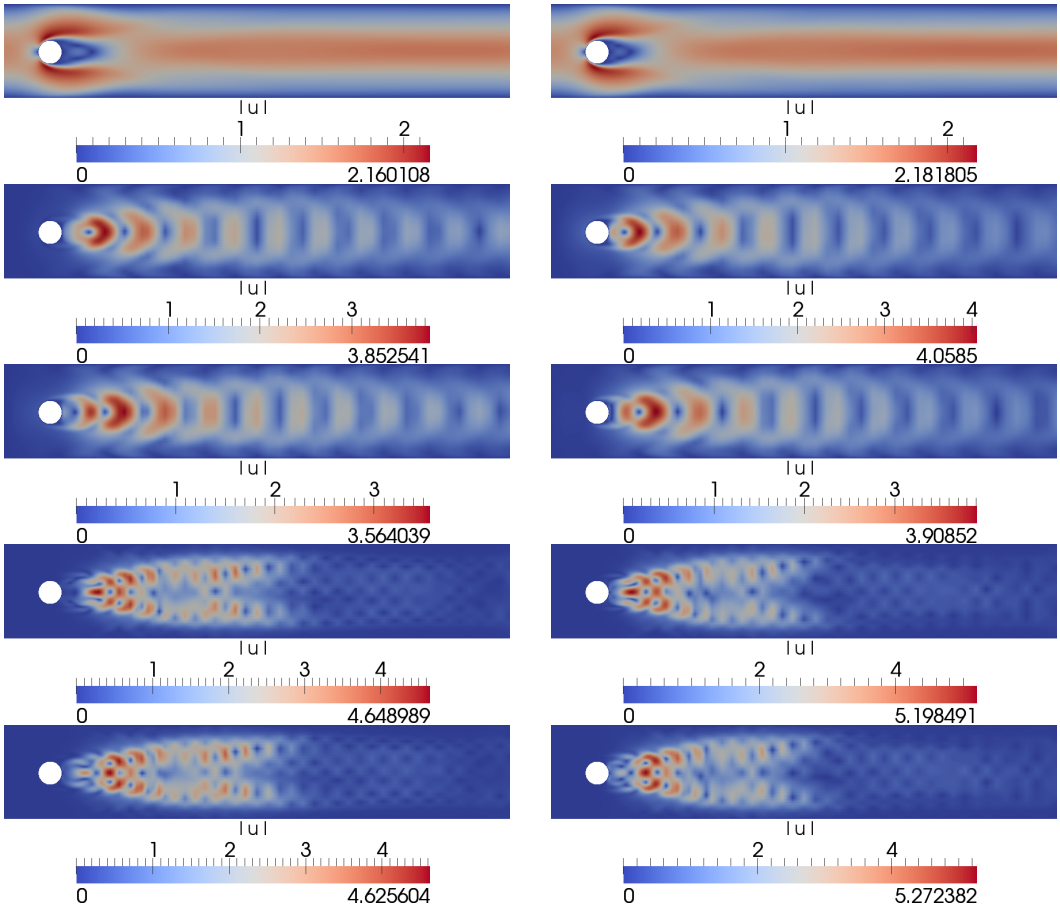

Fig. 4. Norm of the mean velocity (top) and the first POD modes of the velocity fluctuations: POD basis computed from SP-NONLIN (left) and PC (right).

Fig. 5 show that, although structurally similar, the maximum and minimum

values for the norms are quite different for the two numerical methods. One can observe that for the velocity these differences increase with increasing POD mode index.

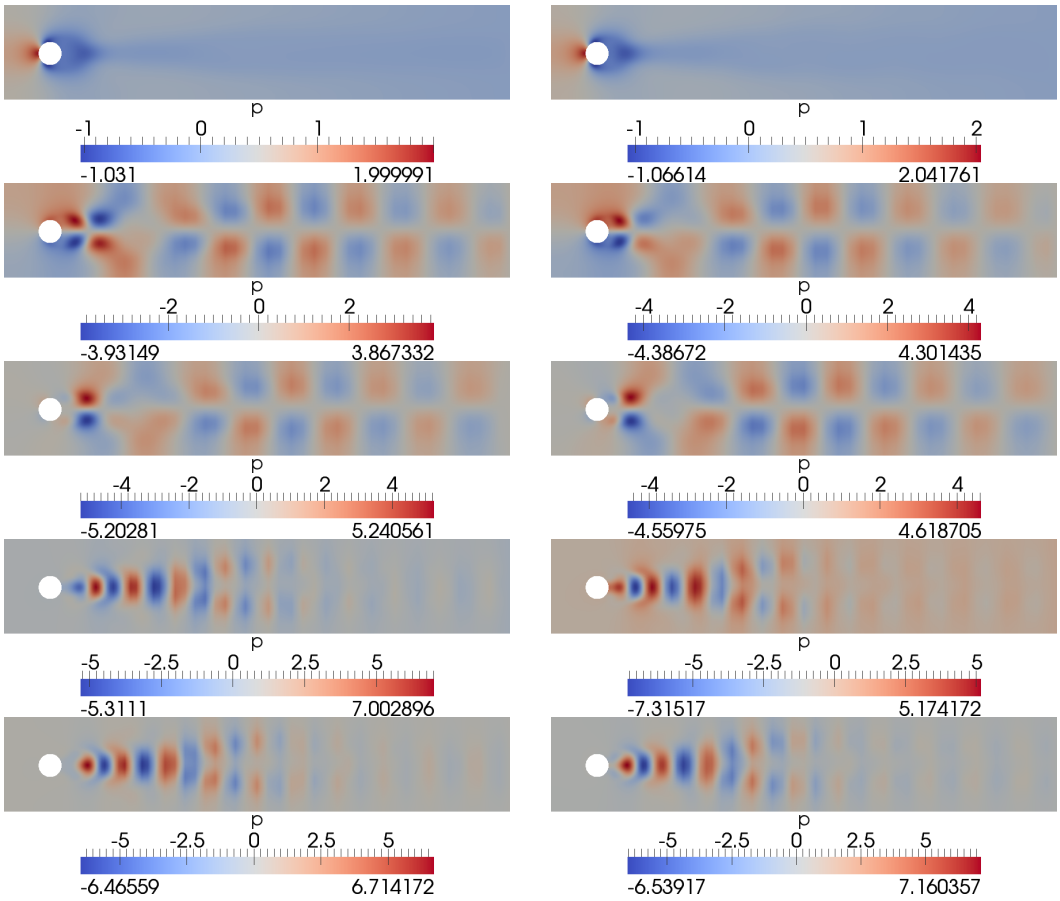

Fig. 5. Mean pressure (top) and the first POD modes of the pressure fluctuations: POD basis computed from SP-NONLIN (left) and PC (right).

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 (7)–(9), the MER is defined as follows

$$\begin{aligned}\mathrm{MER}_R &= \frac{\frac{1}{2}\sum_{m=1}^{M} \|\mathbf{u}'(t_m,\mathbf{x}) - \sum_{r=1}^{R} \alpha_r'(t_m)\boldsymbol{\varphi}_r'(\mathbf{x})\|_{L^2}^2}{\frac{1}{2}\sum_{m=1}^{M} \|\mathbf{u}'(t_m,\mathbf{x})\|_{L^2}^2} \\ &= \frac{\mathrm{trace}(U^T S U) - \sum_{r=1}^{R}\lambda_r}{\mathrm{trace}(U^T S U)} = 1 - \frac{\sum_{r=1}^{R}\lambda_r}{\sum_{r=1}^{M}\lambda_r}\,.\end{aligned}$$

Figure 6 shows $\{\lambda_r\}$ and MER_R for the velocity and pressure fluctuations for the three sets of snapshots. It can be observed that the number of non-zero POD eigenvalues is almost the same for SP-LIN and PC, but it is about twice as high for the snapshots obtained with SP-NONLIN. Moreover, although the

behaviors are similar for the lower modes, the POD basis computed with SP-NONLIN displays a slower decay both in eigenvalues and MER_R for $R > 17$. One possible explanation for this observation is that the high accuracy snapshots capture more details of the flow than the other two sets of snapshots.

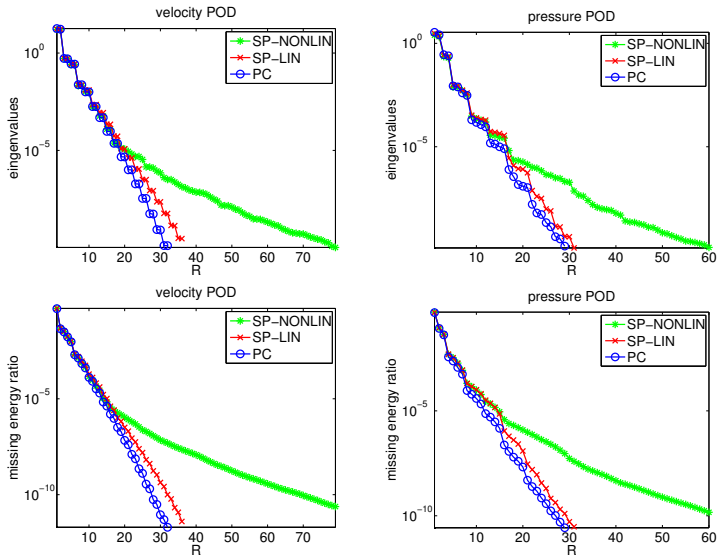

Fig. 6. Eigenvalues and missing energy ratio of the fluctuations.

Figure 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 (VMB-ROM, PCM-ROM, SM-ROM).

Theoretical error estimates in [22], see also [29,30], show that the total error in the numerical discretization of 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, however, the spatial and temporal error

components are constant, since the mesh size and the time step are fixed. Thus, for increasing values of R, one expects the POD error component of the vp-ROMs to initially decrease, but then to reach a plateau where the POD error component has the same or a lower magnitude than the spatial and temporal error components.

The momentum equation of all three vp-ROMs investigated in this section did 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 accuracy of the three vp-ROMs, the time evolution of the drag and lift coefficient, the error in the Strouhal number, the errors in the mean values of the drag and lift coefficient, and the error in the root mean square (rms) of the drag and lift coefficient were monitored. Let $c_{d,\mathrm{meth}}(t)$ denote the drag value computed with a certain numerical method (finite element method or ROM). The rms value is defined by

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

where N_τ is the number of time steps and $\overline{c}_{d,\mathrm{meth}}$ 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 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.

4.4.1 vp-ROMs using highly accurate snapshots

The numerical results for the three vp-ROMs using the high accuracy snapshots from SP-NONLIN are presented in Figs. 7 – 11.

Figure 7 displays the time evolution of the drag and lift coefficients. It can be observed that SM-ROM and PCM-ROM perform well, their results are close to those of the underlying simulation for the snapshots, whereas VMB-ROM

performs very poorly.

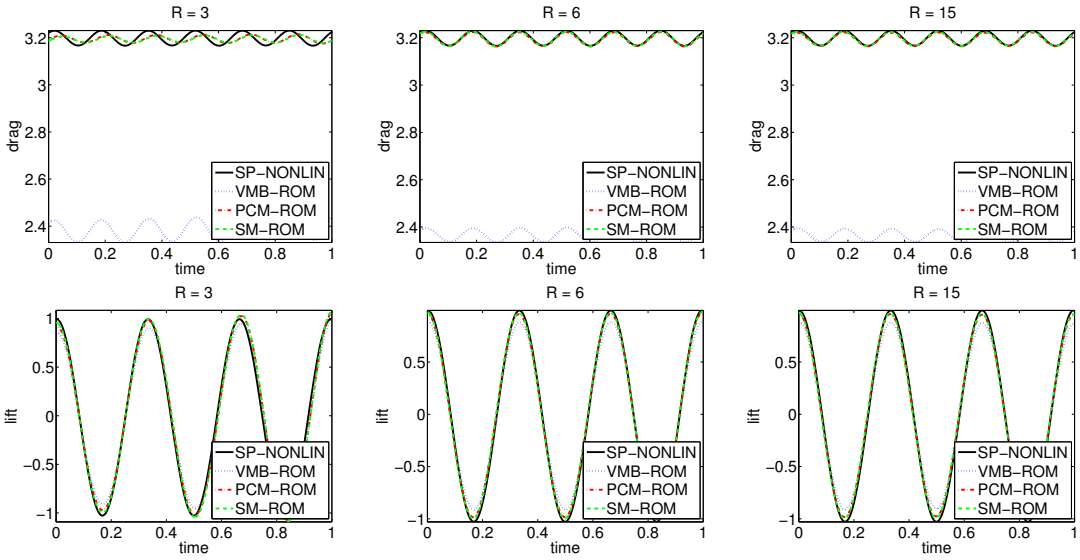

Fig. 7. High accuracy snapshots (SP-NONLIN): time evolution of the drag coefficient (top) and lift coefficient (bottom).

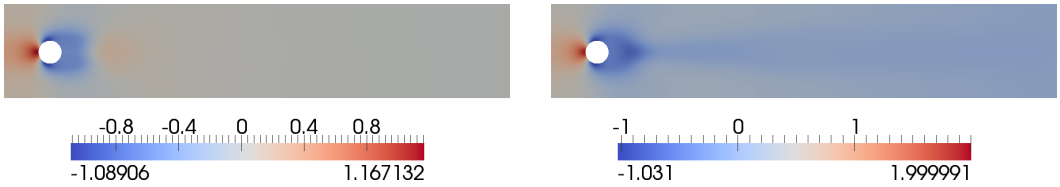

Fig. 8. High accuracy snapshots (SP-NONLIN): pressure coefficient $p_{00}(\mathbf{x})$ computed with VMB-ROM (left); pressure coefficient $\overline{p}(\mathbf{x})$ for PCM-ROM and SM-ROM (right).

To explain the inaccurate VMB-ROM results, note that the drag coefficient depends mainly on the pressure at the cylinder. In the simulations with VMB-ROM, the main contribution to the ROM pressure is $p_{00}(\mathbf{x})$, see Fig. 8, whereas the main part of the ROM pressure for PCM-ROM and SM-ROM is $\overline{p}(\mathbf{x})$, presented once more for convenience in Fig. 8. A comparison reveals that the pressure difference between the back and the front of the cylinder is much smaller for $p_{00}(\mathbf{x})$ than for $\overline{p}(\mathbf{x})$, which results in inaccurate drag forces. With respect to the lift coefficient, the results computed with VMB-ROM are better, but still worse than the results of PCM-ROM and SM-ROM. Since VMB-ROM displayed similar inaccuracies in the studies with moderate and low accuracy snapshots, we conclude that VMB-ROM is not competitive with PCM-ROM and SM-ROM. Thus, for clarity of presentation, the further evaluation of the numerical results will be restricted to PCM-ROM and SM-ROM.

A detailed presentation of the time evolution of the drag coefficient for PCM-ROM and SM-ROM can be found in Figure 9. For $R \leq 2$, the drag coefficient

is almost constant (results not shown). Clear improvements in the accuracy can be seen when going from $R = 2$ to $R = 3$ and from $R = 5$ to $R = 6$, which correspond to jumps in the eigenvalue distribution of the velocity POD modes, see Fig. 6. Overall, for moderate R values, i.e., $3 \leq R \leq 18$, both the PCM-ROM and the SM-ROM yield accurate drag coefficients, which are within the reference intervals given in Table 1. Thus, despite of the simple schemes that were used for the ROMs, very accurate results are computed with PCM-ROM and SM-ROM. For higher R values, however, both vp-ROMs display numerical instabilities. A similar phenomenon was observed in [4]. From the theoretical point of view, this behavior is unexpected, since, as already mentioned above, the POD error should not increase as R approaches the rank of the snapshot matrix. We do not currently have an explanation of this effect. Its investigation is outside the scope of this report and it will be a topic for further research. We also note that, while displaying numerical instability for high R values, SM-ROM and PCM-ROM are still more accurate when they employ high accuracy snapshots then when they employ moderate and low accuracy snapshots, see Sections 4.4.2 and 4.4.3 for comparisons.

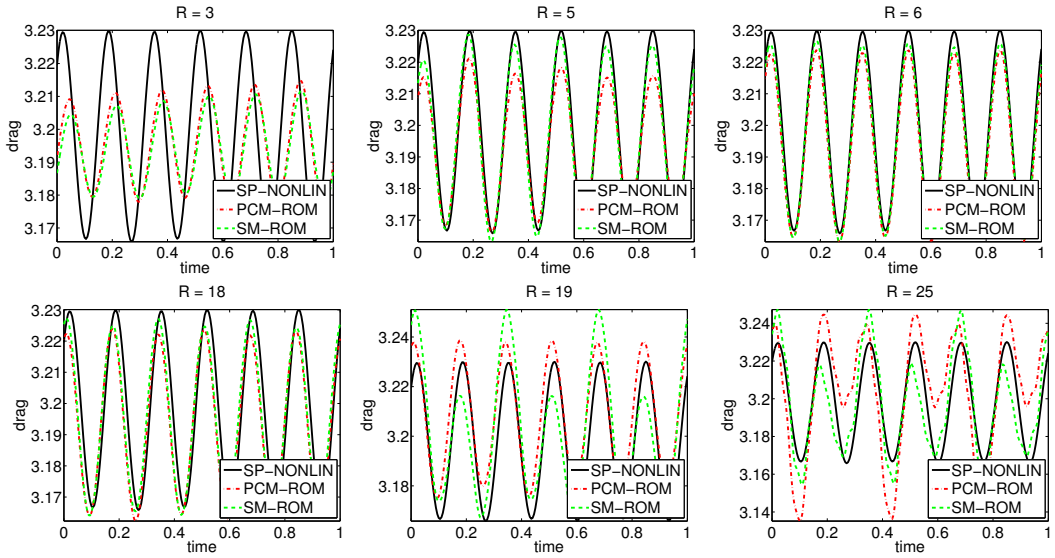

Fig. 9. High accuracy snapshots (SP-NONLIN): time evolution of the drag coefficient.

Figure 10 presents the time evolution of the lift coefficient for several values of R. The general behavior is similar to that of the drag coefficient in Fig. 9. There are, however, two minor differences. First, one needs at least $R = 6$ modes to get stable maximal values for the lift coefficients. Second, although some numerical inaccuracies can be noticed for higher R values, i.e., $R = 25$, one cannot see a pronounced numerical instability as that observed for the drag coefficient and $R = 25$.

To better assess the accuracy of PCM-ROM and SM-ROM, Fig. 11 displays the vp-ROMs' errors in the Strouhal number, in the mean drag, in the mean

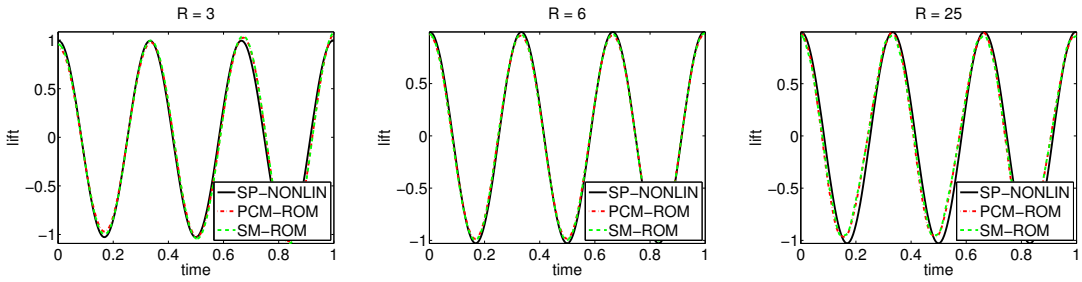

Fig. 10. High accuracy snapshots (SP-NONLIN): time evolution of the lift coefficient.

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 the it is dominated by the spatial and temporal errors. For small R values, the plots in Fig. 11 follow this trend. Indeed, even if the errors are relatively large for very small R, they quickly stabilize around small constant values for $5 \leq R \leq 17$. As already noticed in Fig. 9, for large R, the error degrades, although it remains within reasonable limits. Comparing the PCM-ROM with the SM-ROM results, one observes that SM-ROM yields somewhat more accurate results. An important conclusion is that both SM-ROM and PCM-ROM achieve, regardless of their simple numerical methods, the same order of accuracy as the underlying simulation for the snapshots, even when using a relatively low number of modes.

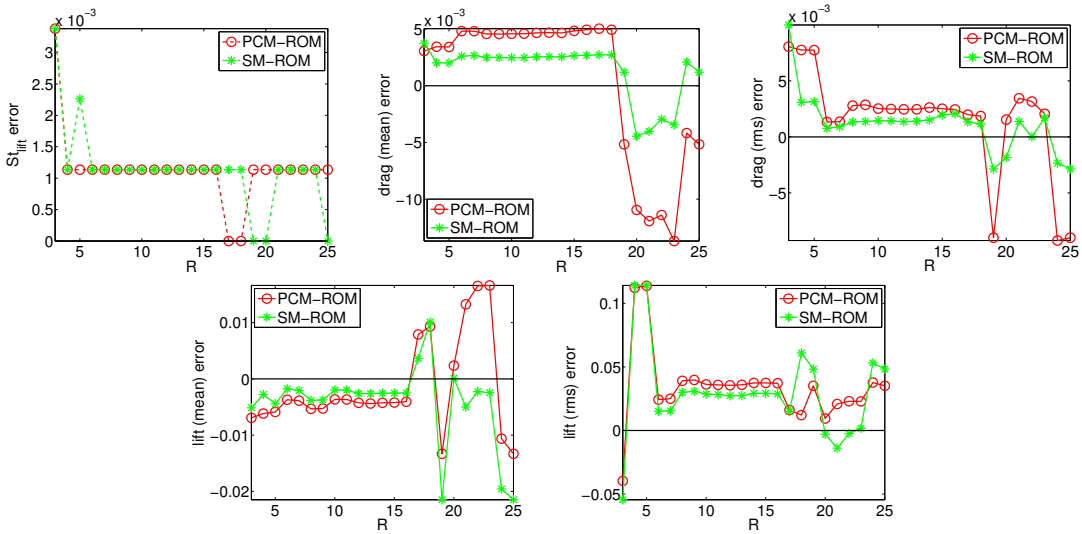

Fig. 11. High accuracy snapshots (SP-NONLIN): errors in the studied functionals.

4.4.2 vp-ROMs using moderately accurate snapshots

The numerical results for PCM-ROM and SM-ROM using the moderate accuracy snapshots from SP-LIN are presented in Figs. 12 and 13.

For the drag and lift coefficients, clear improvements can be seen when going from $R = 2$ to $R = 3$ and from $R = 5$ to $R = 6$, see Fig. 12. Like for the snapshots from SP-NONLIN, these values correspond to jumps in the eigenvalue distribution of the velocity modes, see Fig. 6. Overall, both PCM-ROM and SM-ROM yield drag and lift coefficients that are as accurate as the results of the underlying simulation for obtaining the snapshots. As opposed to the snapshots from SP-NONLIN, neither PCM-ROM nor SM-ROM displays numerical instabilities for higher values of R.

Figure 13 displays the PCM-ROM and the SM-ROM's error for all studied functionals. All errors decrease with increasing R until they reach a level where the POD error is dominated by the spatial and temporal errors. Comparing PCM-ROM with SM-ROM, one observes that the PCM-ROM results are somewhat more accurate for the mean lift coefficient and the SM-ROM results are generally more accurate for the other quantities of interest. For all functionals, both SM-ROM and PCM-ROM achieve the same order of accuracy as SP-LIN already for a small number of modes.

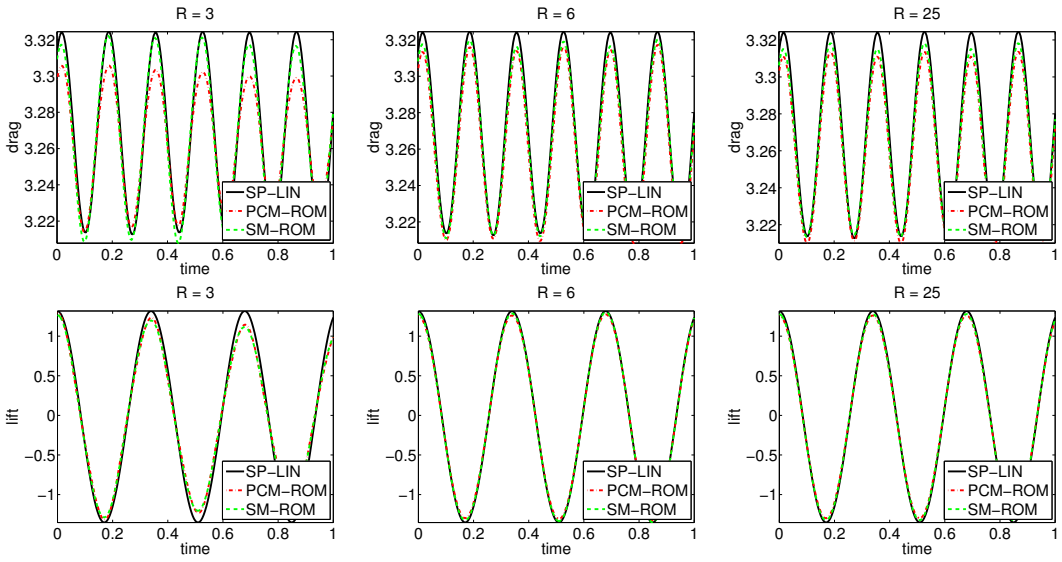

Fig. 12. Moderately accurate snapshots (SP-LIN): time evolution of the drag and lift coefficients.

4.4.3 vp-ROMs using low accuracy snapshots

Figures 14 – 15 present the results for PCM-ROM and SM-ROM using the low accuracy snapshots from PC.

Concerning the temporal evolution of the drag and lift coefficients, Fig. 14, the jumps in the eigenvalue distribution of the velocity modes lead again to clear improvements when going from $R = 2$ to $R = 3$ and from $R = 5$ to $R = 6$. To achieve a stable maximum value for the lift coefficient, 8 POD

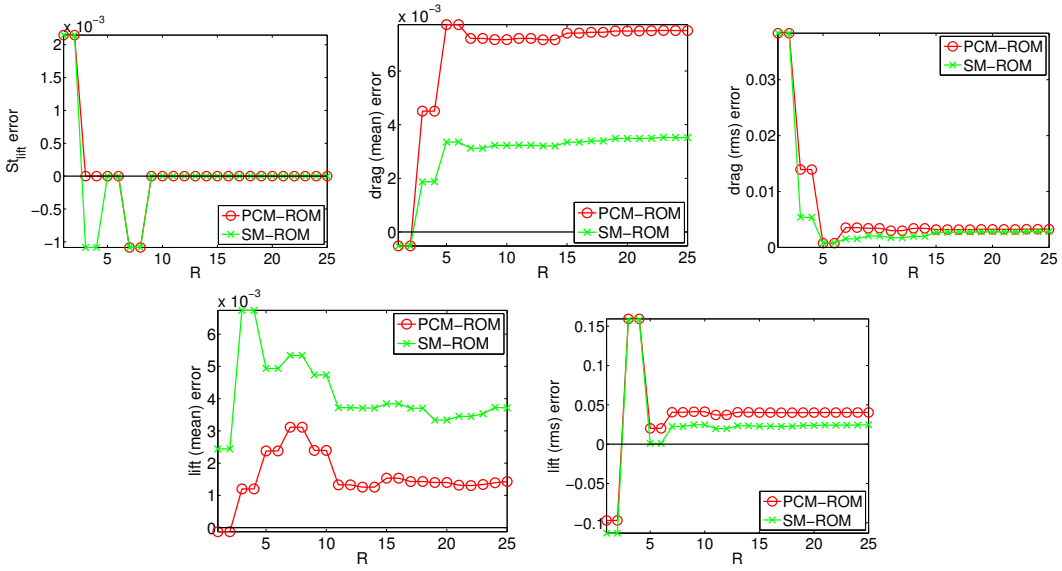

Fig. 13. Moderately accurate snapshots (SP-LIN): errors in the studied functionals.

modes are needed instead of 6 as for SP-LIN. Like for SP-LIN, there are no numerical instabilities for large values of R. As for both other sets of snapshots, the accuracy of the ROM results correlates strongly with the accuracy of the underlying simulation for computing the snapshots.

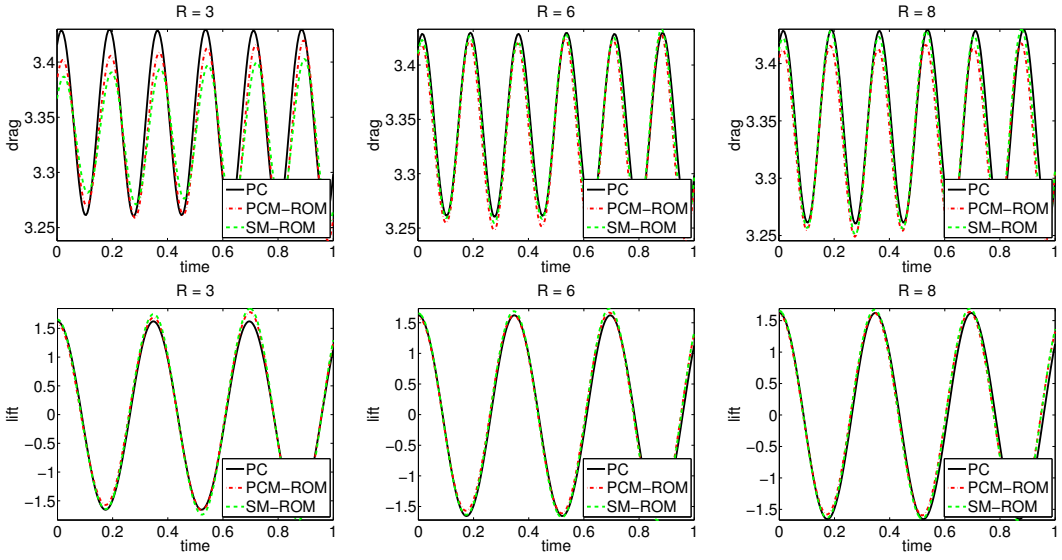

Fig. 14. Low accuracy snapshots (PC): time evolution of the drag and lift coefficients.

With respect to the errors in the studied functionals, displayed in Fig. 15, most of the conclusions drawn for the snapshots from SP-LIN can be transferred to the snapshots from PC. Only the error of the mean lift coefficient is considerably larger than for SP-LIN and SP-NONLIN. PCM-ROM and SM-ROM yield again similar results. Whereas the mean drag is computed more accurately with SM-ROM, the rms values were obtained more accurately with PCM-ROM.

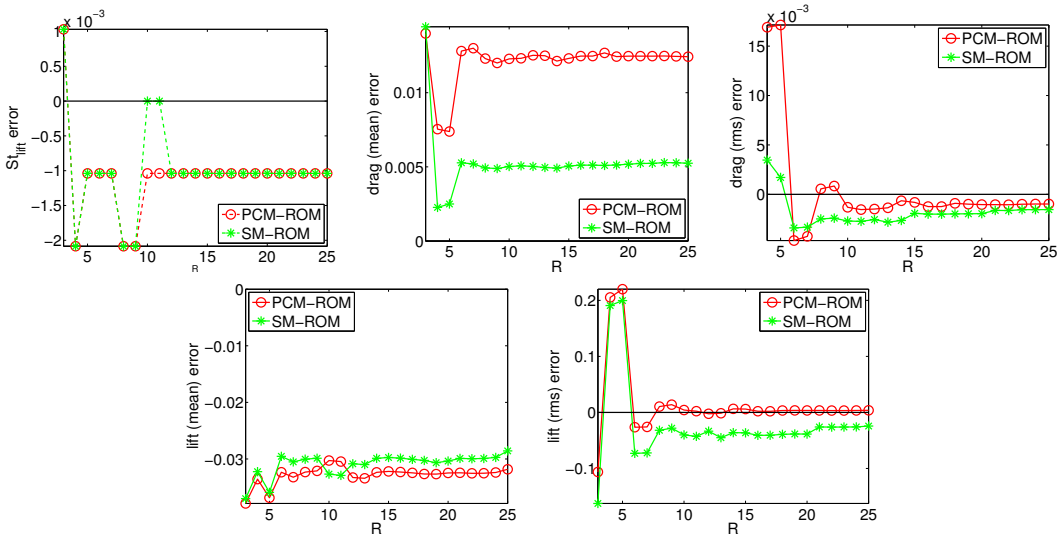

Fig. 15. Low accuracy snapshots (PC): errors in the studied functionals.

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 PCM-ROM and SM-ROM 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 (25) and (31). 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 PCM-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 coefficients $p_{rs}(\mathbf{x})$, 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 PCM-ROM and SM-ROM requires only $\mathcal{O}(R^2)$ operations, yielding relatively low computational times. On the other hand, the length of the vector of pressure coefficients $p_{rs}(\mathbf{x})$ used in VMB-ROM is the dimension of the pressure finite element space, which is much larger than R. In our numerical experiments, in the online stage, the computational times of PCM-ROM and SM-ROM were about the same for moderate values of R ($R < 15$), representing between 0.02% and 0.09% of the computing time of SP-NONLIN. For the same range of values of R, VMB-ROM was computationally more expensive, taking between 0.15% and 1.65% 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 PCM-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 is the impact of the snapshot accuracy onto the accuracy of the ROM results?"*

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 (PCM-ROM and SM-ROM) were clearly superior, both in terms of accuracy and efficiency, to the ROM that uses only velocity POD modes (VMB-ROM).

For studying the impact of the snapshot accuracy, three sets of snapshots were used: of high accuracy, moderate accuracy, and low accuracy. The numerical investigations showed that for all three velocity-pressure ROMs the accuracy of the numerical results was strongly correlated with the accuracy of the snapshots. Thus, this study clearly supports the approach of performing accurate (and probably time-consuming) simulations for computing the snapshots.

Several research directions will be pursued in future. First, we will investigate the cause and possible remedies of the numerical oscillations observed in the ROM results when high accuracy snapshots and a large number of POD modes were used. Second, we will study whether the conclusions of this report carry over to the case of structure-dominated turbulent flows. Finally, the rigorous numerical analysis for discretizations of the new velocity-pressure ROM (SM-ROM) will be a topic of future research.

References

[1] Imran Akhtar, Ali H. Nayfeh, and Calvin J. Ribbens. On the stability and extension of reduced-order Galerkin models in incompressible flows. *Theor. Comput. Fluid Dyn.*, 23:213–237, 2009.

[2] David Amsallem and Charbel Farhat. Stabilization of projection-based reduced-order models. *Internat. J. Numer. Methods Engrg.*, 91(4):358–377, 2012.

[3] Jeanne A. Atwell and Belinda B. King. Reduced order controllers for spatially distributed systems via proper orthogonal decomposition. *SIAM J. Sci. Comput.*, 26(1):128–151 (electronic), 2004.

[4] Joan Baiges, Ramon Codina, and Sergio Idelsohn. Explicit reduced order models for the stabilized finite element approximation of the incompressible Navier–Stokes equations. technical report, Universitat Politécnica de Catalunya, 2012.

[5] M. Balajewicz and E. H. Dowell. Stabilization of projection-based reduced order models of the Navier–Stokes. *Nonlinear Dynamics*, 70:1619–1632, 2012.

[6] Y. Bazilevs, V. M. Calo, J. A. Cottrell, T. J. R. Hughes, A. Reali, and G. Scovazzi. Variational multiscale residual-based turbulence modeling for large eddy simulation of incompressible flows. *Comput. Methods Appl. Mech. Engrg.*, 197(1-4):173–201, 2007.

[7] M. Bergmann, C.-H. Bruneau, and A. Iollo. Enablers for robust POD models. *J. Comput. Phys.*, 228(2):516–538, 2009.

[8] M. Braack, E. Burman, V. John, and G. Lube. Stabilized finite element methods for the generalized Oseen problem. *Comput. Methods Appl. Mech. Engrg.*, 196(4-6):853–866, 2007.

[9] M. Buffoni, S. Camarri, A. Iollo, and M. V. Salvetti. Low-dimensional modelling of a confined three-dimensional wake flow. *J. Fluid Mech.*, 569:141–150, 2006.

[10] John Burkardt, Max Gunzburger, and Hyung-Chun Lee. POD and CVT-based reduced-order modeling of Navier-Stokes flows. *Comput. Methods Appl. Mech. Engrg.*, 196(1-3):337–355, 2006.

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

[12] Saifon Chaturantabut and Danny C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.*, 32(5):2737–2764, 2010.

[13] Saifon Chaturantabut and Danny C. Sorensen. A state space error estimate for POD-DEIM nonlinear model reduction. *SIAM J. Numer. Anal.*, 50(1):46–63, 2012.

[14] Alexandre Joel Chorin. Numerical solution of the Navier-Stokes equations. *Math. Comp.*, 22:745–762, 1968.

[15] D. Galbally, K. Fidkowski, K. Willcox, and O. Ghattas. Non-linear model reduction for uncertainty quantification in large-scale inverse problems. *Internat. J. Numer. Methods Engrg.*, 81(12):1581–1608, 2010.

[16] B. Galletti, C. H. Bruneau, L. Zannetti, and A. Iollo. Low-order modelling of laminar flow regimes past a confined square cylinder. *J. Fluid Mech.*, 503:161–170, 2004.

[17] Vivette Girault and Pierre-Arnaud Raviart. *Finite element methods for Navier-Stokes equations*, volume 5 of *Springer Series in Computational Mathematics*. Springer-Verlag, Berlin, 1986. Theory and algorithms.

[18] J. L. Guermond, P. Minev, and Jie Shen. An overview of projection methods for incompressible flows. *Comput. Methods Appl. Mech. Engrg.*, 195(44-47):6011–6045, 2006.

[19] J.-L. Guermond and L. Quartapelle. On the approximation of the unsteady Navier-Stokes equations by finite element projection methods. *Numer. Math.*, 80(2):207–238, 1998.

[20] Max D. Gunzburger, Janet S. Peterson, and John N. Shadid. Reduced-order modeling of time-dependent PDEs with multiple parameters in the boundary data. *Comput. Methods Appl. Mech. Engrg.*, 196(4-6):1030–1047, 2007.

[21] Philip Holmes, John L. Lumley, and Gal Berkooz. *Turbulence, coherent structures, dynamical systems and symmetry.* Cambridge Monographs on Mechanics. Cambridge University Press, Cambridge, 1996.

[22] T. Iliescu and Z. Wang. Variational multiscale proper orthogonal decomposition: Convection-dominated convection-diffusion-reaction equations. *Math. Comput.*, 2013. to appear.

[23] Angelo Iollo, Alain Dervieux, Jean-Antoine Désidéri, and Stéphane Lanteri. Two stable POD-based approximations to the Navier-Stokes equations. *Comput. Vis. Sci.*, 3(1-2):61–66, 2000.

[24] Volker John. On the efficiency of linearization schemes and coupled multigrid methods in the simulation of a 3D flow around a cylinder. *Internat. J. Numer. Methods Fluids*, 50(7):845–862, 2006.

[25] Volker John and Gunar Matthies. Higher-order finite element discretizations in a benchmark problem for incompressible flows. *Int. J. Numer. Methods Fluids*, 37(8):885–903, 2001.

[26] Volker John and Gunar Matthies. MooNMD—a program package based on mapped finite element methods. *Comput. Vis. Sci.*, 6(2-3):163–169, 2004.

[27] Volker John, Gunar Matthies, and Joachim Rang. A comparison of time-discretization/linearization approaches for the incompressible Navier-Stokes equations. *Comput. Methods Appl. Mech. Engrg.*, 195(44-47):5995–6010, 2006.

[28] Volker John and Joachim Rang. Adaptive time step control for the incompressible Navier-Stokes equations. *Comput. Methods Appl. Mech. Engrg.*, 199(9-12):514–524, 2010.

[29] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for parabolic problems. *Numer. Math.*, 90(1):117–148, 2001.

[30] Zhendong Luo, Jing Chen, I. M. Navon, and 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):1–19, 2008/09.

[31] H.V. Ly and H.T. Tran. Modeling and control of physical processes using proper orthogonal decomposition. *Math. Comput. Modelling*, 33(1):223–236, 2001.

[32] B. R. Noack, M. Morzynski, and G. Tadmor. *Reduced-Order Modelling for Flow Control*, volume 528. Springer Verlag, 2011.

[33] Bernd R. Noack, Paul Papas, and Peter A. Monkewitz. The need for a pressure-term representation in empirical Galerkin models of incompressible shear flows. *J. Fluid Mech.*, 523:339–365, 2005.

[34] M. Schäfer and S. Turek. The benchmark problem "Flow around a cylinder". In E.H. Hirschel, editor, *Flow Simulation with High-Performance Computers II*, volume 52 of *Notes on Numerical Fluid Mechanics*, pages 547–566. Vieweg, 1996.

[35] S. Sirisup and G. E. Karniadakis. A spectral viscosity method for correcting the long-term behavior of POD models. *J. Comput. Phys.*, 194(1):92–116, 2004.

[36] Lawrence Sirovich. Turbulence and the dynamics of coherent structures. I. Coherent structures. *Quart. Appl. Math.*, 45(3):561–571, 1987.

[37] R. Ştefănescu and I. M. Navon. POD/DEIM nonlinear model order reduction of an ADI implicit shallow water equations model. *J. Comput. Phys.*, 2012.

[38] R. Témam. Sur l'approximation de la solution des équations de Navier-Stokes par la méthode des pas fractionnaires. II. *Arch. Rational Mech. Anal.*, 33:377–385, 1969.

[39] J. van Kan. A second-order accurate pressure-correction scheme for viscous incompressible flow. *SIAM J. Sci. Statist. Comput.*, 7(3):870–891, 1986.

[40] S. Volkwein. Model reduction using proper orthogonal decomposition. *Lecture Notes, Faculty of Mathematics and Statistics, University of Konstanz*, 2011.

[41] Zhu Wang, Imran Akhtar, Jeff Borggaard, and Traian Iliescu. Proper orthogonal decomposition closure models for turbulent flows: a numerical comparison. *Comput. Methods Appl. Mech. Engrg.*, 237/240:10–26, 2012.

[42] J. Weller, E. Lombardi, M. Bergmann, and A. Iollo. Numerical methods for low-order modeling of fluid flows based on POD. *Internat. J. Numer. Methods Fluids*, 63(2):249–268, 2010.

[43] Shangyou Zhang. A new family of stable mixed finite elements for the 3D Stokes equations. *Math. Comp.*, 74(250):543–554, 2005.