Reduced-Order Subscales for POD models

Joan Baiges[*,1,2], Ramon Codina[2,1] and Sergio Idelsohn[3,1]

[1] Centre Internacional de Mètodes Numèrics a l'Enginyeria (CIMNE),
Edifici C1, Campus Nord UPC C/ Gran Capità S/N 08034 Barcelona, Spain
[2] Universitat Politècnica de Catalunya, Jordi Girona 1-3, Edifici C1, 08034 Barcelona, Spain
[3] Intitució Catalana de Recerca i Estudis Avançats (ICREA), Passeig Luís Companys 23, 08010 Barcelona, Spain
*jbaiges@cimne.upc.edu

March 2, 2015

In this work the Reduced-Order Subscales for Proper Orthogonal Decomposition models are presented. The basic idea consists in splitting the full-order solution into the part which can be captured by the reduced-order model and the part which cannot, the subscales, for which a model is required. The proposed model for the subscales is defined as a linear function of the solution of the reduced-order model. The coefficients of this linear function are obtained by comparing the solution of the full-order model with the solution of the reduced-order model for the same initial conditions, which, for convenience, are evaluated in the snapshots used to train the original reduced-order-model. The difference between both solutions are the subscales, for which a model can be built using a least-squares procedure. The subscales are then introduced as a correction in the reduced-order model, resulting in an important improvement in accuracy. The enhanced reduced-order model is tested in several numerical examples. These practical cases show that the use of the subscales leads to more accurate solutions, successfully corrects errors introduced by hyper-reduction, and allows to solve complex flow problems using a reduced number of degrees of freedom.

Contents

1 Introduction

Proper Orthogonal Decomposition (POD) reduced-order models consist in the projection of high-fidelity (full-order) representations of physical problems in Computational Mechanics onto low-dimensional spaces of solutions. These low-dimensional spaces are capable of capturing the most representative components of the solution, their main advantage being that the computations in the low-dimensional space can be done at a reduced computational cost. This has led researchers to apply POD based reduced-order models to a variety of physical and engineering problems: circuit design [34], multiscale modeling in solid mechanics [49], metal forming processes [36], and computational fluid dynamics, see e.g. [8, 7, 13, 24, 25, 35, 30, 47, 48]. Once applied to the physical problem of interest, these POD based reduced-order models can be used to solve engineering problems such as shape optimization [1, 11, 33, 39] and flow control [2, 5, 26, 23].

Despite all the advantages of reduced-order modeling strategies, POD models still have some unresolved issues which prevent them from being used in day-to-day engineering. One of these issues is related to the fact that the effect of the components of the full-order model not present in the low-order space is not taken into account in the reduced-order equations. This issue is well known in other Computational Mechanics contexts like finite elements, where stabilized formulations have been developed to deal with the instabilities of the Galerkin method. One of the most popular frameworks for developing stabilized formulations is the Variational Multiscale Method (VMS) [29]. In the Variational Multiscale Method, stabilized formulations are obtained by including, in the finite element equations, the effect of the part of the solution which cannot be captured by the finite element space. This part of the solution is denoted as the subscales. The contribution of the subscales turns out not only to be key for the stabilization of the finite element problem, but it also allows one to take into account important small-scale effects such as turbulence. Elaborate models for the subscales have been developed which allow one to improve the accuracy of VMS stabilized finite element methods. See for instance [21], where the time dependency of the subscales is taken into account, [17], where subscales orthogonal to the finite element space are considered, and [20, 19, 6] where the contribution of the subscales on the element boundaries is considered.

In this work we propose to apply the Variational Multiscale ideas to Proper Orthogonal Decomposition reduced-order models. Some applications of the VMS method to POD models can be found in the literature. In [42] discarded POD modes in a convection-diffusion equation are linked to solved modes through a master-slave relationship in order to improve the accuracy of the reduced-order model. The work presented in [43] makes use of a spectral viscosity in order to take into account the effect of discarded modes and a calibrated artificial viscosity is added to the POD model in [44]. In [10] a model for the error is developed by using a reduced-order basis for the modeling of the POD residuals, which are then included in the reduced-order equations. The number of degrees of freedom for the model is then equal to the addition of the number of reduced-order modes plus the number of basis functions for the modeling of the residual. In [48] a model for the effect of the high-frequency modes is obtained by decomposing the POD modes into coarse and fine modes and devising a model for the coupling between the coarse and the fine modes. In this case, the number of degrees of freedom of the reduced-order model is equal to the addition of the number of fine and coarse reduced-order modes. In [3] a calibrated master-slave relationship between slow and fast modes is used, which allows to reconstruct the solution of the fine-scale modes from the resolved coarse modes.

Contrary to the previous works on the Variational Multiscale Method applied to POD models, we will present a model for the reduced-order subscales which does not require of additional degrees of freedom. In order to do this, we will develop a strategy which aims to account for the intermediate component of the solution, that is, the component which can be represented by the finite element space, but cannot be captured by the reduced-order space. We call this component the reduced-order subscales. The full-order solution can then be split into the reduced-order solution and the subscales. The main idea is to consistently maintain this splitting in the derivation of the reduced-order equations.

Once the reduced-order equations accounting for the subscales have been derived, a model for them as a function of the reduced-order solution will be introduced. When the Variational Multiscale Method is applied to finite element problems, the model for the subscales is based on physical arguments. In the method presented in this work, on the contrary, we propose to base the model for the reduced-order subscales on the information contained in the snapshots used to build the reduced-order basis. The methodology is similar to calibrated POD methods [22, 31, 38], where the error of the POD model is used to modify the coefficients of the reduced-order system. In the method proposed in this work, we use the exact subscales of the POD model to develop a model only for the subscales, leaving the coefficients for the POD modes unmodified. This is done by computing the subscales of the reduced-order model for each snapshot, and then introducing a least-squares approximation for these subscales in terms of the reduced-order solution. The advantage of the approach we propose is that the reduced-order model is capable to adapt to the variations in the physical parameters which determine the flow.

This results in a modified reduced-order system. As it will be shown, when compared to the original reduced-order model, the modified system has several advantages: for the same number of degrees of freedom, more accurate solutions are obtained. Also, when using hyper-reduction techniques [37, 27, 40, 4, 45, 8], it is capable to adapt to errors introduced by hyper-reduction and correct them. Finally, when dealing with problems which are typically difficult to solve using reduced-order models, it is capable of providing accurate solutions by using only a reduced number of degrees of freedom.

The paper is organized as follows: in Section 2 the Proper Orthogonal Decomposition framework for reduced-order modeling is described and the properties of the Singular Value Decomposition strategy for computing the reduced-order basis are summarized. In Section 3 the formulation for the subscales for Proper Orthogonal Decomposition methods is developed, and the Least-Squares model for computing the subscales is presented, paying special attention to its properties. Finally, numerical examples illustrate the features of the method in Section 4: the capability to improve the performance of reduced-order models, the ability to correct the errors introduced by hyper-reduction, the capability to adapt to variable physical parameters and the use of the subscales in order to solve complex flow problems. Conclusions close the paper in Section 5.

2   POD reduced order approximation of variational problems

In this section we summarize the Proper Orthogonal Decomposition (POD) framework in which the methods to be developed in this work are set. We also outline the properties of the Singular Value Decomposition (SVD) which are relevant for the development of reduced-order subscales for POD models.

2.1   POD for the reduced-order modeling of general problems

Let us define the unknown vector field $U \in \mathbb{R}^M$ and the full-order space $\mathcal{V} = \mathbb{R}^M$. M represents the dimension of the unknown and is associated with the number of nodes of the computational mesh. Let

us consider the fully discretized in time and space general non-linear transient system:

$$\boldsymbol{A}(\boldsymbol{U}^{n+1})\boldsymbol{U}^{n+1} = \boldsymbol{R}(\boldsymbol{U}^n, \boldsymbol{U}^{n-1}, ...), \tag{1}$$

where n is the time step counter, $\boldsymbol{A} \in \mathbb{R}^{M \times M}$ is the system matrix which might depend on $\boldsymbol{U}^{n+1}$, and $\boldsymbol{R}$ is the right-hand-side of the system, which might depend on the values of $\boldsymbol{U}$ at the preceding time steps. For simplicity, the dependence of $\boldsymbol{A}$ and $\boldsymbol{R}$ on $\boldsymbol{U}$ will be omitted from the notation in the following.

Departing from the previous discretized system, POD models are built by projecting it onto a low-dimensional subspace $\mathcal{V}_\Phi \subset \mathbb{R}^M$. Vectors $\boldsymbol{U}$ can now be approximated as:

$$\boldsymbol{U} \approx \boldsymbol{\Phi}\boldsymbol{U}_\Phi, \tag{2}$$

where $\boldsymbol{\Phi} \in \mathbb{R}^{M \times m}$ is an orthonormal base for $\mathcal{V}_\Phi$, m being the dimension of the reduced-order subspace, $m \ll M$. $\boldsymbol{U}_\Phi \in \mathbb{R}^m$ are the components of $\boldsymbol{U}$ in $\mathcal{V}_\Phi$ expressed in the reference system defined by $\boldsymbol{\Phi}$. A usual approach in POD is to actually decompose the unknown into the addition of a mean value constant vector and the reduced order representation. Equation (2) is replaced by:

$$\boldsymbol{U} \approx \boldsymbol{\Phi}\boldsymbol{U}_\Phi + \overline{\boldsymbol{U}}. \tag{3}$$

Two convenient operators are the restriction operator $\mathcal{R}_\Phi : \mathbb{R}^M \to \mathbb{R}^m$, which allows one to obtain the projection of a vector in $\mathbb{R}^M$ onto $\mathcal{V}_\Phi$ expressed in the reference system defined by $\boldsymbol{\Phi}$:

$$\mathcal{R}_\Phi\left(\boldsymbol{U}\right) = \boldsymbol{\Phi}^T\left(\boldsymbol{U} - \overline{\boldsymbol{U}}\right) \in \mathbb{R}^m,$$

and the extension operator $\mathcal{E} : \mathbb{R}^m \to \mathbb{R}^M$, which allows one to go from the expression of a vector in $\mathcal{V}_\Phi$ expressed in the reference system defined by $\boldsymbol{\Phi}$ to its expression in $\mathbb{R}^M$:

$$\mathcal{E}\left(\boldsymbol{U}_\Phi\right) = \boldsymbol{\Phi}\boldsymbol{U}_\Phi + \overline{\boldsymbol{U}} \in \mathbb{R}^M.$$

We also define the projection operator onto $\mathcal{V}_\Phi$, $\Pi_\Phi : \mathbb{R}^M \to \mathbb{R}^M$:

$$\Pi_\Phi\left(\boldsymbol{U}\right) = \mathcal{E}\left(\mathcal{R}_\Phi\left(\boldsymbol{U}\right)\right).$$

Note that $\boldsymbol{U} \neq \Pi_\Phi\left(\boldsymbol{U}\right)$ because all the information not contained in the reduced-order subspace $\mathcal{V}_\Phi$ is lost in the restriction step. Obviously, $\Pi_\Phi^2 = \Pi_\Phi$, i.e., Π_Φ is a projection.

Various methods exist for computing the reduced order basis $\boldsymbol{\Phi}$, in this work we focus in the Proper Orthogonal Decomposition [15, 32, 28], whose properties are summarized in Section 2.2.

Introducing (3) in (1) leads to the $M \times m$ overdetermined system:

$$\boldsymbol{A}\boldsymbol{\Phi}\boldsymbol{U}_\Phi^{n+1} = \boldsymbol{R} - \boldsymbol{A}\overline{\boldsymbol{U}},$$

where the system matrix times the mean value of the unknown $\overline{\boldsymbol{U}}$ have been sent to the right-hand-side of the system. As explained in [12, 14], if matrix $\boldsymbol{A}$ is symmetric and positive definite, a least-squares strategy for approximating the linearized overdetermined system leads to:

$$\boldsymbol{\Phi}^T\boldsymbol{A}\boldsymbol{\Phi}\boldsymbol{U}_\Phi^{n+1} = \boldsymbol{\Phi}^T\boldsymbol{R} - \boldsymbol{\Phi}^T\boldsymbol{A}\overline{\boldsymbol{U}}. \tag{4}$$

Let us introduce the following notation:

$$\begin{aligned} \boldsymbol{A}_\Phi &:= \boldsymbol{\Phi}^T\boldsymbol{A}\boldsymbol{\Phi} & \in \mathbb{R}^{m \times m}, & \tag{5}\\ \boldsymbol{R}_\Phi &:= \boldsymbol{\Phi}^T\left(\boldsymbol{R} - \boldsymbol{A}\overline{\boldsymbol{U}}\right) & \in \mathbb{R}^m, & \tag{6} \end{aligned}$$

where $\boldsymbol{A}_\Phi$ is the reduced-order system matrix, $\boldsymbol{R}_\Phi$ is the reduced-order right-hand-side accounting also for the contribution of the mean value constant vector to the right-hand-side of the system. The reduced-order system is now:

$$\boldsymbol{A}_\Phi \boldsymbol{U}_\Phi^{n+1} = \boldsymbol{R}_\Phi. \tag{7}$$

This is the final reduced-order system which needs to be solved. The computational cost of solving this system is reduced since typically $m \ll M$. For non-linear problems, additional hyper-reduction approaches might be required to reconstruct system (7) at a reduced computational cost (see [37, 27, 9, 16, 40, 41, 4, 46, 45, 8]).

In the case $\boldsymbol{A}$ is not symmetric and positive definite, instead of projecting the system onto $\mathcal{V}_\Phi$, a Petrov-Galerkin projection is required in order to ensure stability of the reduced-order system [12]. A suitable Petrov-Galerkin projector is $\boldsymbol{\Phi}^T \boldsymbol{A}^T$, which results in the reduced-order system:

$$\boldsymbol{A}_{\Phi A} \boldsymbol{U}_\Phi^{n+1} = \boldsymbol{R}_{\Phi A}, \tag{8}$$

where now:

$$\begin{aligned}
\boldsymbol{A}_{\Phi A} &:= \boldsymbol{\Phi}^T \boldsymbol{A}^T \boldsymbol{A} \boldsymbol{\Phi} &&\in \mathbb{R}^{m \times m}, &\tag{9} \\
\boldsymbol{R}_{\Phi A} &:= \boldsymbol{\Phi}^T \boldsymbol{A}^T \left(\boldsymbol{R} - \boldsymbol{A}\overline{\boldsymbol{U}} \right) &&\in \mathbb{R}^{m}, &\tag{10}
\end{aligned}$$

The methods explained in this work are valid both for the reduced-order system (7) and for the Petrov-Galerkin system (8). For simplicity we are going to do all the developments using system (7).

2.2 Construction of Φ through a Singular Value Decomposition

We have already presented the reduced-order system (7). However, we still have not explained how to obtain the reduced-order basis $\boldsymbol{\Phi}$. Since the process for obtaining this reduced-order basis is relevant for the development of the reduced-order subscales for the POD model, we summarize it here.

In Proper Orthogonal Decomposition models, the reduced-order basis functions are obtained by collecting a series of snapshots, applying a Singular Value Decomposition (SVD) to these snapshots and finally keeping only the first m resulting basis functions. Let us define the snapshots matrix:

$$\boldsymbol{\mathcal{U}} \in \mathbb{R}^{M \times N},$$

where N is the number of snapshots. Each column of $\boldsymbol{\mathcal{U}}$ corresponds to a snapshot, a solution to the full order system (1) for a certain configuration. We will denote the jth column (snapshot) of the snapshot collection as $\boldsymbol{\mathcal{U}}_{(:,j)}$. The collection of snapshots is expected to provide a rich enough set of solutions so that a representative basis $\boldsymbol{\Phi}$ can be extracted from them. These snapshots can be obtained both by using experimental and numerical methods, although in this work we focus in the case where the snapshots are obtained through the numerical solution of the associated full-order system in different configurations.

Once the set of snapshots has been collected, matrix $\boldsymbol{\mathcal{U}}$ is decomposed into the addition of a mean value matrix and the product of three matrices through the process known as the Singular Value Decomposition. We start by defining the general one matrix as a matrix full of ones:

$$\mathbf{1}^{k \times l} \in \mathbb{R}^{k \times l} \quad | \quad \mathbf{1}_{ij}^{k \times l} = 1 \quad \forall i, j.$$

This allows us to define the mean value of the snapshots as:

$$\overline{\boldsymbol{U}} = \frac{\boldsymbol{\mathcal{U}} \mathbf{1}^{N \times 1}}{N} \in \mathbb{R}^{M \times 1},$$

5

and the snapshots' mean matrix :

$$\overline{\mathcal{U}} = \overline{U}\mathbf{1}^{1 \times N} \in \mathbb{R}^{M \times N},$$

which finally leads to the decomposition:

$$\mathcal{U} = \overline{\mathcal{U}} + \mathbf{\Phi}_0 \mathbf{\Sigma}_0 \mathbf{\Psi}_0^T.$$

In the previous expression, $\mathbf{\Phi}_0 \in \mathbb{R}^{M \times M}$ is a basis for $\mathbb{R}^M$. Supposing that there are $N_0 \leq N$, $N_0 \leq M$ linearly independent snapshots, the first N_0 basis functions (columns) of $\mathbf{\Phi}_0$ are the reduced-order basis functions, while the remaining $(M - N_0)$ basis functions account for the subspace of $\mathbb{R}^M$ orthogonal to the snapshots. The last $M - N_0$ basis functions are not constructed from the snapshots. $\mathbf{\Sigma}_0 \in \mathbb{R}^{M \times N}$ is the eigenvalues matrix and $\mathbf{\Psi}_0 \in \mathbb{R}^{N \times N}$ is the matrix which accounts for the representation of the snapshots in the new basis. The decomposition of $\mathcal{U} - \overline{\mathcal{U}}$ as the product of these three matrices has very convenient properties:

- $\mathbf{\Phi}_0$ is and orthogonal matrix, that is:

$$\mathbf{\Phi}_0 \mathbf{\Phi}_0^T = \boldsymbol{I}^{M \times M},$$

  each column of $\mathbf{\Phi}_0$ containing one of the orthonormal vectors which compose the basis, with $\boldsymbol{I}^{M \times M}$ denoting the $M \times M$ identity matrix. The first N_0 columns of $\mathbf{\Phi}_0$ are capable of exactly representing all the snapshots contained in $\mathcal{U}$. The last $(M - N_0)$ columns contain the vectors of a basis of a subspace orthogonal to the snapshots.

- $\mathbf{\Sigma}_0$ is a non-negative matrix with all terms zero except along the diagonal, each term in the diagonal containing the eigenvalue associated to the corresponding basis function. Moreover, the eigenvalues in $\mathbf{\Sigma}_0$ (and the basis functions in $\mathbf{\Phi}_0$) are ordered in a decreasing order so that the largest eigenvalues appear first. These N eigenvalues are a precise measure of the relative importance of each of the first N basis functions in $\mathbf{\Phi}_0$ in the representation of the set of snapshots $\mathcal{U}$.

- $\mathbf{\Psi}_0$ is also an orthogonal matrix, each of its rows corresponding to one of the snapshots in $\mathcal{U}$, represented using the N first basis functions of $\mathbf{\Phi}_0$.

Taking into account that the eigenvalues in $\mathbf{\Sigma}_0$ and their corresponding basis functions are ordered in a decreasing order, and due to the fact that the eigenvalues usually decrease very quickly, it is possible to obtain a good approximation of $\mathcal{U}$ by keeping only the $m \ll M$ first basis functions of $\mathbf{\Phi}_0$. We define:

$$\begin{aligned}
\mathbf{\Phi} &:= \mathbf{\Phi}_{0(:,1:m)} \in \mathbb{R}^{M \times m}, \\
\mathbf{\Sigma} &:= \mathbf{\Sigma}_{0(1:m,1:m)} \in \mathbb{R}^{m \times m}, \\
\mathbf{\Psi} &:= \mathbf{\Psi}_{0(1:N,1:m)} \in \mathbb{R}^{N \times m}.
\end{aligned}$$

The key to the reduced computational cost of POD models is the low value of m when compared to M. The set of snapshots can now be approximated as:

$$\mathcal{U} - \overline{\mathcal{U}} \approx \mathbf{\Phi}\mathbf{\Sigma}\mathbf{\Psi}^T.$$

Note that $\mathbf{\Sigma}\mathbf{\Psi}^T$ is precisely the representation (projection) of the snapshots in the reduced-order basis $\mathbf{\Phi}$, which we will denote as $\mathcal{U}_\Phi$:

$$\mathcal{U}_\Phi := \mathbf{\Sigma}\mathbf{\Psi}^T \in \mathbb{R}^{m \times N}. \tag{11}$$

It is convenient to note now that $\mathcal{U}_\Phi$ can also be expressed as the restriction of $\mathcal{U}$:

$$\mathcal{U}_\Phi = \mathcal{R}_\Phi(\mathcal{U}) := \mathbf{\Phi}^T(\mathcal{U} - \overline{\mathcal{U}}).$$

3 Reduced-Order Subscales for POD models

In this section the general formulation of the reduced-order subscales for POD models is presented. We also pay special attention to the algorithmic process followed for the construction of the model for the subscales in practical cases, and we describe some adjustments required for an optimal performance of the subscales model.

3.1 Decomposition of the unknown into the reduced-order part and the Subscales

Let us start the section by defining the solution to the full-order problem (1) for a given time step as:

$$\boldsymbol{U}^{n+1} = \boldsymbol{A}^{-1}\boldsymbol{R},$$

and recalling that we consider the best possible approximation to the full-order solution $\boldsymbol{U}^{n+1}$ in the reduced-order subspace to be the projection $\Pi_\Phi\left(\boldsymbol{U}^{n+1}\right)$, its expression in the reduced-order basis $\boldsymbol{\Phi}$ being $\mathcal{R}_\Phi\left(\boldsymbol{U}^{n+1}\right)$.

Let us now consider the original unknown of the problem and, instead of introducing approximation (3), split it into the optimal reduced-order part (the projection of the unknown into the reduced-order subspace) and the part which cannot be captured by the reduced-order space, which we call the subscales $\tilde{\boldsymbol{U}}^{n+1}$:

$$\boldsymbol{U}^{n+1} = \Pi_\Phi\left(\boldsymbol{U}^{n+1}\right) + \tilde{\boldsymbol{U}}^{n+1}. \tag{12}$$

It is clear from this decomposition that the space for the subscales $\mathcal{V}_\Phi^\perp$ must be the orthogonal complementary space of $\mathcal{V}_\Phi$:

$$\mathcal{V} = \mathcal{V}_\Phi \oplus \mathcal{V}_\Phi^\perp,$$

$$\mathcal{V}_\Phi^\perp = \left\{\boldsymbol{X} \in \mathcal{V}: \quad \boldsymbol{X}^T\boldsymbol{Y} = 0 \quad \forall \boldsymbol{Y} \in \mathcal{V}_\Phi\right\}.$$

If we replace the original reduced-order approximation of the unknown (3) by the exact expression (12), we insert it in the full-order system (1) and we do the least-squares procedure as in (4), we obtain:

$$\boldsymbol{A}_\Phi\mathcal{R}_\Phi\left(\boldsymbol{U}^{n+1}\right) + \boldsymbol{\Phi}^T\boldsymbol{A}\tilde{\boldsymbol{U}}^{n+1} = \boldsymbol{R}_\Phi, \tag{13}$$

where we have included the additional terms due to the fact that we are considering the contribution from the subscales. Note that although $\mathcal{V}_\Phi^\perp$ is orthogonal to $\mathcal{V}_\Phi$, the product $\boldsymbol{A}\tilde{\boldsymbol{U}}^{n+1}$ does not necessarily belong to $\mathcal{V}_\Phi^\perp$, and then $\boldsymbol{\Phi}^T\boldsymbol{A}\tilde{\boldsymbol{U}}^{n+1} \neq \boldsymbol{0}$. Neglecting this contribution is precisely what prevents the reduced-order model from attaining the best possible performance. The objective of this work is to develop a model for the contribution of the subscales so that a close to the optimal solution for the reduced-order system is obtained.

The model we propose to take into account the effect of the subscales is to define a field $\boldsymbol{S}^{n+1} \in \mathcal{V}_\Phi$ and its expression in the reduced-order basis $\boldsymbol{\Phi}$, $\boldsymbol{S}_\Phi^{n+1} \in \mathbb{R}^m$, such that:

$$\boldsymbol{A}_\Phi\boldsymbol{S}_\Phi^{n+1} \approx \boldsymbol{\Phi}^T\boldsymbol{A}\tilde{\boldsymbol{U}}^{n+1},$$

and the reduced-order system now becomes:

$$\boldsymbol{A}_\Phi\left(\boldsymbol{U}_\Phi^{n+1} + \boldsymbol{S}_\Phi^{n+1}\right) = \boldsymbol{R}_\Phi, \tag{14}$$

$\boldsymbol{S}_\Phi^{n+1}$ will be modeled as a function of $\boldsymbol{U}_\Phi^{n+1}$. The simple approach we propose consists of building a linear model for $\boldsymbol{S}_\Phi^{n+1}$ by means of a $\boldsymbol{C}_S \in \mathbb{R}^{m \times m}$ coefficients matrix and a $\boldsymbol{D}_S \in \mathbb{R}^m$ independent term, such that:

$$\boldsymbol{S}_\Phi^{n+1} = \boldsymbol{C}_s\boldsymbol{U}_\Phi^{n+1} + \boldsymbol{D}_S. \tag{15}$$

This model can be justified by considering the full-order system and its splitting into the reduced-order and the orthogonal components. In order to do this, we introduce the orthogonal space basis:

$$\boldsymbol{\Phi}_\perp \in \mathbb{R}^{M \times (M-m)},$$

and the restriction onto this orthogonal space:

$$\mathcal{R}_\perp \left(\boldsymbol{U}^{n+1} \right) = \boldsymbol{\Phi}_\perp^T \left(\boldsymbol{U} - \overline{\boldsymbol{U}} \right) \in \mathbb{R}^{M-m}.$$

Similarly we define:

$$\begin{aligned}
\boldsymbol{A}_\perp &:= \boldsymbol{\Phi}_\perp^T \boldsymbol{A} \boldsymbol{\Phi}_\perp & &\in \mathbb{R}^{(M-m) \times (M-m)}, \\
\boldsymbol{A}_{\Phi\perp} &:= \boldsymbol{\Phi}^T \boldsymbol{A} \boldsymbol{\Phi}_\perp & &\in \mathbb{R}^{m \times (M-m)}, \\
\boldsymbol{A}_{\perp\Phi} &:= \boldsymbol{\Phi}_\perp^T \boldsymbol{A} \boldsymbol{\Phi} & &\in \mathbb{R}^{(M-m) \times m}, \\
\boldsymbol{R}_\perp &:= \boldsymbol{\Phi}_\perp^T \left(\boldsymbol{R} - \boldsymbol{A} \overline{\boldsymbol{U}} \right) & &\in \mathbb{R}^{m},
\end{aligned}$$

and the full-order system written in terms of the reduced-order and the orthogonal terms is:

$$\left[\begin{array}{cc} \boldsymbol{A}_\Phi & \boldsymbol{A}_{\Phi\perp} \\ \boldsymbol{A}_{\perp\Phi} & \boldsymbol{A}_\perp \end{array} \right] \left[\begin{array}{c} \mathcal{R}_\Phi \left(\boldsymbol{U}^{n+1} \right) \\ \mathcal{R}_\perp \left(\boldsymbol{U}^{n+1} \right) \end{array} \right] = \left[\begin{array}{c} \boldsymbol{R}_\Phi \\ \boldsymbol{R}_\perp \end{array} \right].$$

From the second equation we get:

$$\mathcal{R}_\perp \left(\boldsymbol{U}^{n+1} \right) = \boldsymbol{A}_\perp^{-1} \left(\boldsymbol{R}_\perp - \boldsymbol{A}_{\perp\Phi} \mathcal{R}_\Phi \left(\boldsymbol{U}^{n+1} \right) \right),$$

which we can replace in the first one, and then:

$$\left(\boldsymbol{A}_\Phi - \boldsymbol{A}_{\Phi\perp} \boldsymbol{A}_\perp^{-1} \boldsymbol{A}_{\perp\Phi} \right) \mathcal{R}_\Phi \left(\boldsymbol{U}^{n+1} \right) = \boldsymbol{R}_\Phi - \boldsymbol{A}_{\Phi\perp} \boldsymbol{A}_\perp^{-1} \boldsymbol{R}_\perp.$$

Obviously, it is unfeasible to compute $\boldsymbol{A}_\perp^{-1}$, $\boldsymbol{A}_{\Phi\perp}$, and $\boldsymbol{A}_{\perp\Phi}$ at a reduced cost . This is why we are approximating the additional terms due to the subscales as:

$$\begin{aligned}
-\boldsymbol{A}_{\Phi\perp} \boldsymbol{A}_\perp^{-1} \boldsymbol{A}_{\perp\Phi} &\approx \boldsymbol{A}_\Phi \boldsymbol{C}_S, \\
\boldsymbol{A}_{\Phi\perp} \boldsymbol{A}_\perp^{-1} \boldsymbol{R}_\perp &\approx \boldsymbol{A}_\Phi \boldsymbol{D}_S.
\end{aligned}$$

The final reduced-order system becomes:

$$\left(\boldsymbol{A}_\Phi + \boldsymbol{A}_\Phi \boldsymbol{C}_S \right) \boldsymbol{U}_\Phi = \boldsymbol{R}_\Phi - \boldsymbol{A}_\Phi \boldsymbol{D}_S. \tag{16}$$

More complex models involving non-linear dependencies and dependencies on the value of the subscales in previous time steps (time dependent subscales) could also be developed, although this possibility is not pursued in this work. Once the general expression for $\boldsymbol{S}_\Phi^{n+1}$ has been defined, it only remains to define a methodology to obtain $\boldsymbol{C}_S$ and $\boldsymbol{D}_S$. For simplicity, in the following sections we will refer to $\boldsymbol{S}_\Phi^{n+1}$ as the subscales, although we should refer to them as the field in $\mathcal{V}_\Phi$ which accounts for the effect of the subscales.

3.2 An a posteriori Least-Squares model for the Subscales

The strategy we propose for the modeling of the subscales consists in, once the reduced-order basis is built, using the snapshots from the solution of the full-order problem in order to train a least-squares model for the subscales.

Let us take the set of snapshots $\mathcal{U}$, and consider that they have been obtained by numerically solving the full-order system (1). If the problem we are facing is transient, we will have computed each of the snapshots in $\mathcal{U}$ by solving a time step of the scheme (1) departing from the solution at the immediately preceding time step. This is so because the right-hand-side of (1) depends at least on the values of U at the previous step. If the problem is not transient, then only the boundary conditions and physical parameters are required to solve (1).

From now on, and for the sake of conciseness, we focus in the case of a transient problem where a first order time derivative of the unknown is discretized using a single time step finite difference scheme. In this case we denote as $\mathcal{U}_0 \in \mathbb{R}^{M \times N}$ the collection of the initial conditions for the snapshots which, when solving a single time step of (1), lead to the collection of snapshots $\mathcal{U}$, that is, the collection of solutions that precede those of the snapshots. Note that there might be a nonempty intersection between $\mathcal{U}_0$ and $\mathcal{U}$. In fact, $\mathcal{U}_0$ is introduced for convenience, and not strictly necessary (see Remark 2). Physical parameters and boundary conditions are considered to be constant for all snapshots for simplicity.

For a given snapshot U^{n+1}, which we store in the jth column of the snapshot collection, $\mathcal{U}_{(:,j)}$, its initial conditions are U^n, which we store in the jth column of the initial conditions snapshots matrix, $\mathcal{U}_{0(:,j)}$. From the singular value decomposition of the snapshots we already have the expression of the snapshots expressed in the reduced-order basis, $\mathcal{R}_\Phi \left(U^{n+1} \right)$, stored in $\mathcal{U}_\Phi$ as defined in (11). Similarly, we define the matrix of initial conditions snapshots expressed in the reduced-order basis, $\mathcal{U}_{\Phi 0} \in \mathbb{R}^{m \times N}$, as:

$$\mathcal{U}_{\Phi 0} = \mathcal{R}_\Phi \left(\mathcal{U}_0 \right).$$

The idea now is to evaluate the error of the original reduced-order model (7) at each snapshot when compared to the solution of the full-order system (1), and use this information to find the matrix coefficients of C_S and D_S in (15).

If we know the initial conditions U^n (stored in $\mathcal{U}_0$) for a given time step to be solved, and its solution U^{n+1} (stored in $\mathcal{U}$) we can solve the time step with the reduced-order model and evaluate the reduced-order error. We start by building the right-hand side of the system:

$$\boldsymbol{R} \left(\Pi_\Phi \left(U^n \right) \right),$$

In the case we are dealing with a non-linear problem, we also build the system matrix:

$$\boldsymbol{A} \left(\Pi_\Phi \left(U^{n+1} \right) \right).$$

Remark 1 Note that we have *not* used the full order fields U^n and U^{n+1} in order to build $\boldsymbol{R}$ and $\boldsymbol{A}$, but instead we have used their projection onto $\mathcal{V}_\Phi$, $\Pi_\Phi \left(U^n \right)$, recovered through an extension from $\mathcal{U}_{\Phi 0}$, and $\Pi_\Phi \left(U^{n+1} \right)$, recovered through an extension from $\mathcal{U}_\Phi$. This needs to be done in this manner in order to obtain the most accurate modeling of the subscales, since in the real solution of the reduced-order model we always depart from initial conditions in the reduced-order space. $\triangle$

From $\boldsymbol{A} \left(\Pi_\Phi \left(U^{n+1} \right) \right)$ and $\boldsymbol{R} \left(\Pi_\Phi \left(U^n \right) \right)$ we build $\boldsymbol{A}_{\Phi\Pi}$ and $\boldsymbol{R}_{\Phi\Pi}$ following the definitions in equations (5)-(6):

$$\boldsymbol{A}_{\Phi\Pi} = \boldsymbol{\Phi}^T \boldsymbol{A} \left(\Pi_\Phi \left(U^{n+1} \right) \right) \boldsymbol{\Phi},$$
$$\boldsymbol{R}_{\Phi\Pi} = \boldsymbol{\Phi}^T \left(\boldsymbol{R} \left(\Pi_\Phi \left(U^n \right) \right) - \boldsymbol{A} \left(\Pi_\Phi \left(U^{n+1} \right) \right) \overline{U} \right),$$
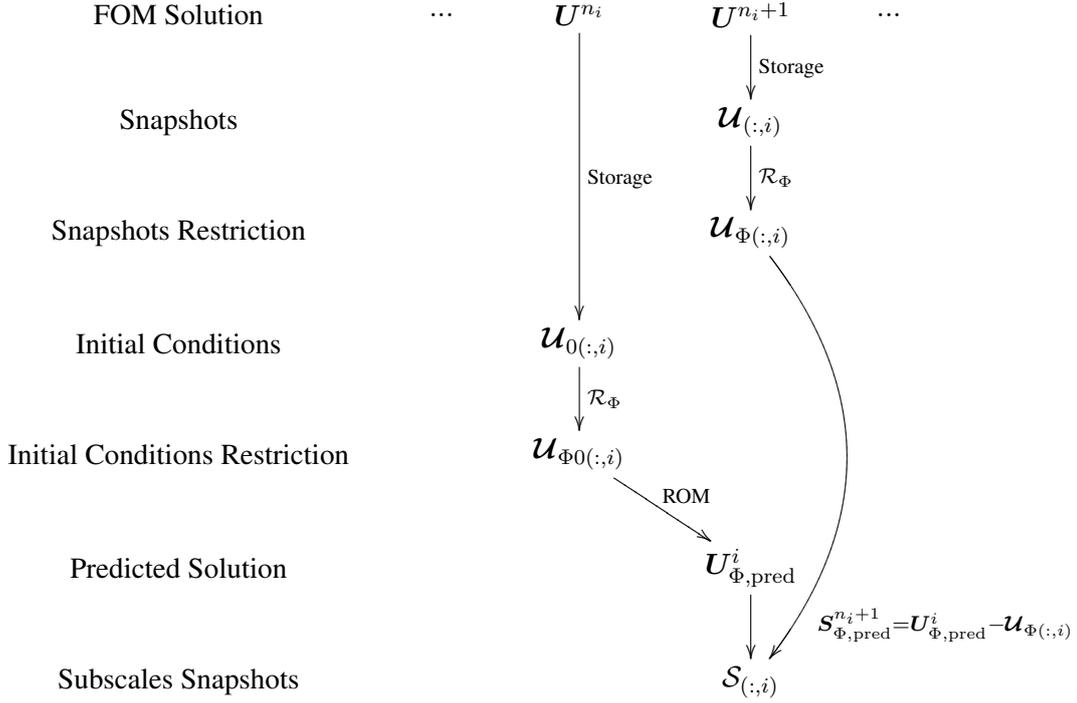
Figure 1: Summary of the process for computing the subscale associated to the ith snapshot, captured at time step $n_i + 1$.

and we solve the reduced-order system (7) in order to obtain the result predicted by the reduced-order model *without the subscales*, $\boldsymbol{U}_{\Phi,\text{pred}}^{n+1}$:

$$\boldsymbol{A}_{\Phi\Pi}\boldsymbol{U}_{\Phi,\text{pred}}^{n+1} = \boldsymbol{R}_{\Phi\Pi}. \tag{17}$$

Comparing the reduced-order system without the subscales (17) with the reduced-order system with the subscales (14), and imposing that the solution to the reduced-order system with the subscales is as close as possible to the restriction of the solution of the full-order model $\left(\boldsymbol{U}_{\Phi}^{n+1} \approx \mathcal{R}_{\Phi}\left(\boldsymbol{U}^{n+1}\right)\right)$, it is clear that the predicted subscales must be:

$$\boldsymbol{S}_{\Phi,\text{pred}}^{n+1} = \boldsymbol{U}_{\Phi,\text{pred}}^{n+1} - \mathcal{R}_{\Phi}\left(\boldsymbol{U}^{n+1}\right). \tag{18}$$

The value for the predicted susbscales field $\boldsymbol{S}_{\Phi,\text{pred}}^{n+1}$ can be evaluated in this manner for each snapshot in the collection of snapshots $\mathcal{U}$, and then we have a *collection of snapshots for the predicted subscales* $\boldsymbol{\mathcal{S}}_{\Phi} \in \mathbb{R}^{m \times N}$, each of its columns corresponding to a predicted subscale snapshot $\boldsymbol{S}_{\Phi,\text{pred}}^{n+1}$. Fig. 1 summarizes the process for computing the subscale associated to a specific snapshot.

Remark 2 It is also possible to, instead of computing the subscale taking as initial conditions the immediately previous time step, compute it taking as initial conditions the previous snapshot with an adjusted time step size, $\delta t_{\Phi} = \delta t \cdot n_s$, where n_s is the number of time steps between snapshots. This would allow us to avoid storing the initial conditions snapshots. However, we have not pursued this possibility in this work. $\triangle$

Once the snapshot coefficients for the solution, $\boldsymbol{\mathcal{U}}_{\Phi}$, and the subscales, $\boldsymbol{\mathcal{S}}_{\Phi}$, are available, the model for the subscales as a function of the optimal solution can be built. The model we propose is a linear relationship between the subscales and the solution of the reduced-order model. This is the simplest

possible model, and although more elaborated models could be built, these have not been pursued in this work. We obtain it by minimizing the distance between the subscales snapshots and the subscales resulting from applying the proposed model (15) to the solution snapshots via least-squares:

$$\boldsymbol{C}_S, \boldsymbol{D}_S = \arg \min_{\boldsymbol{a} \in \mathbb{R}^{m \times m}, \boldsymbol{b} \in \mathbb{R}^m} \sum_{i=1}^{N} ||\boldsymbol{\mathcal{S}}_{\Phi,(:,i)} - \boldsymbol{a}\boldsymbol{\mathcal{U}}_{\Phi(:,i)} - \boldsymbol{b}||^2 \tag{19}$$

Once matrix $\boldsymbol{C}_S$ and the independent term $\boldsymbol{D}_S$ are defined, the reduced-order model taking into account the subscales can be obtained by replacing $\boldsymbol{C}_S$ and $\boldsymbol{D}_S$ in (16):

$$\boxed{\boldsymbol{A}_\Phi \left(\boldsymbol{I}_\Phi + \boldsymbol{C}_S\right) \boldsymbol{U}_\Phi^{n+1} = \boldsymbol{R}_\Phi - \boldsymbol{A}_\Phi \boldsymbol{D}_S}, \tag{20}$$

where $\boldsymbol{I}_\Phi = \boldsymbol{I}^{m \times m}$ denotes the $m \times m$ identity matrix. Note that, if the approximation for the subscales is exact, $\mathcal{R}_\Phi \left(\boldsymbol{U}^{n+1}\right)$ is the solution of (20) since, rearranging the system we have:

$$\begin{aligned} \boldsymbol{A}_\Phi \left(\mathcal{R}_\Phi \left(\boldsymbol{U}^{n+1}\right) + \boldsymbol{C}_S \mathcal{R}_\Phi \left(\boldsymbol{U}^{n+1}\right) + \boldsymbol{D}_S\right) &= \boldsymbol{R}_\Phi \Rightarrow \\ \boldsymbol{A}_\Phi \left(\mathcal{R}_\Phi \left(\boldsymbol{U}^{n+1}\right) + \boldsymbol{S}_\Phi^{n+1}\right) &= \boldsymbol{R}_\Phi \Rightarrow \\ \boldsymbol{A}_\Phi \boldsymbol{U}_{\Phi,\mathrm{pred}}^{n+1} &= \boldsymbol{R}_\Phi, \end{aligned}$$

which is exact due to the definition of $\boldsymbol{U}_{\Phi,\mathrm{pred}}^{n+1}$ in (17).

3.3 Least Squares and the SVD

Let us study with some more detail the least-squares problem (19). This system is equivalent to saying that we want to model the subscales snapshots as:

$$\boldsymbol{\mathcal{S}}_\Phi^T \approx \boldsymbol{\mathcal{U}}_\Phi^T \boldsymbol{C}_S^T + \boldsymbol{1}^{N \times 1} \boldsymbol{D}_S^T. \tag{21}$$

This is an overdetermined system ($N > m$), so we solve it via least squares. The least squares problem is:

$$\begin{bmatrix} \boldsymbol{\mathcal{U}}_\Phi \boldsymbol{\mathcal{U}}_\Phi^T & \boldsymbol{\mathcal{U}}_\Phi \boldsymbol{1}^{N \times 1} \\ \boldsymbol{1}^{1 \times N} \boldsymbol{\mathcal{U}}_\Phi^T & \left(\boldsymbol{1}^{1 \times N} \boldsymbol{1}^{N \times 1}\right) \end{bmatrix} \begin{bmatrix} \boldsymbol{C}_S^T \\ \boldsymbol{D}_S^T \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mathcal{U}}_\Phi \boldsymbol{\mathcal{S}}_\Phi^T \\ \boldsymbol{1}^{1 \times N} \boldsymbol{\mathcal{S}}_\Phi^T \end{bmatrix}. \tag{22}$$

In the previous system of equations, the dimensions of $\boldsymbol{D}_S^T$ are $1 \times m$, while the dimensions of $\boldsymbol{C}_S^T$ are $m \times m$. As a consequence, equation (22) is equivalent to m square systems of dimension $m + 1$. Each of these systems of equations corresponds to the model for the subscale of a single degree of freedom.

Note that:

$$\boldsymbol{\mathcal{U}}_\Phi \boldsymbol{\mathcal{U}}_\Phi^T = \boldsymbol{\Sigma} \boldsymbol{\Psi}^T \boldsymbol{\Psi} \boldsymbol{\Sigma}^T = \boldsymbol{\Sigma} \boldsymbol{\Sigma},$$

since $\boldsymbol{\Psi}$ is orthogonal and $\boldsymbol{\Sigma}$ is diagonal, $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}^T$.

We now define:

$$\overline{\boldsymbol{\Sigma}^2} := \frac{\boldsymbol{\Sigma}\boldsymbol{\Sigma}}{N}.$$

$\overline{\boldsymbol{\Sigma}^2}$ is also diagonal, the values on the diagonal being the squared eigenvalues of the original SVD for the snapshots divided by the number of snapshots N. Moreover:

$$\left(\boldsymbol{1}^{1 \times N} \cdot \boldsymbol{1}^{N \times 1}\right) = N \in \mathbb{R}.$$

We define the mean of the snapshots expressed in the reduced order basis as:

$$\overline{\boldsymbol{\mathcal{U}}_\Phi} := \frac{\boldsymbol{\mathcal{U}}_\Phi \boldsymbol{1}^{N \times 1}}{N} \in \mathbb{R}^{m \times 1},$$

11

and the mean of the subscales expressed in the reduced order basis as:

$$\overline{\boldsymbol{\mathcal{S}}_\Phi^T} := \frac{\mathbf{1}^{1\times N}\boldsymbol{\mathcal{S}}_\Phi^T}{N} \in \mathbb{R}^{1\times m}.$$

Finally the mean of the product between each of the basis functions and the subscales is defined as:

$$\overline{\boldsymbol{\mathcal{U}}_\Phi \boldsymbol{\mathcal{S}}_\Phi^T} := \frac{\boldsymbol{\mathcal{U}}_\Phi \boldsymbol{\mathcal{S}}_\Phi^T}{N}.$$

The least squares system (22) can now be written as:

$$\begin{bmatrix} \overline{\boldsymbol{\Sigma}^2} & \overline{\boldsymbol{\mathcal{U}}_\Phi} \\ \overline{\boldsymbol{\mathcal{U}}_\Phi^T} & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{C}_S^T \\ \boldsymbol{D}_S^T \end{bmatrix} = \begin{bmatrix} \overline{\boldsymbol{\mathcal{U}}_\Phi \boldsymbol{\mathcal{S}}_\Phi^T} \\ \overline{\boldsymbol{\mathcal{S}}_\Phi^T} \end{bmatrix} \tag{23}$$

From the second group of equations of (23) we have:

$$\boxed{\boldsymbol{D}_S^T = \overline{\boldsymbol{\mathcal{S}}_\Phi^T} - \overline{\boldsymbol{\mathcal{U}}_\Phi^T}\boldsymbol{C}_S^T}$$

Note that $\boldsymbol{D}_S$ is null if both $\overline{\boldsymbol{\mathcal{S}}_\Phi^T}$ and $\overline{\boldsymbol{\mathcal{U}}_\Phi^T}$ are null.

Plugging this result in the first group of equations of (23) we have:

$$\left[\overline{\boldsymbol{\Sigma}^2} - \overline{\boldsymbol{\mathcal{U}}_\Phi}\,\overline{\boldsymbol{\mathcal{U}}_\Phi^T}\right] \boldsymbol{C}_S^T = \overline{\boldsymbol{\mathcal{U}}_\Phi \boldsymbol{\mathcal{S}}_\Phi^T} - \overline{\boldsymbol{\mathcal{U}}_\Phi}\,\overline{\boldsymbol{\mathcal{S}}_\Phi^T}. \tag{24}$$

The second term on the right hand side of (24) vanishes either if the mean of the subscales is null, or if the mean of the snapshots is null. The right hand side of (24) is precisely the covariance between the snapshots (expressed in the reduced order basis) and the subscales:

$$\text{cov}_{\mathcal{US}} = \overline{\boldsymbol{\mathcal{U}}_\Phi \boldsymbol{\mathcal{S}}_\Phi^T} - \overline{\boldsymbol{\mathcal{U}}_\Phi}\,\overline{\boldsymbol{\mathcal{S}}_\Phi^T} \in \mathbb{R}^{m\times m}.$$

The left hand side is the variance of the snapshots:

$$\text{var}_{\mathcal{U}} = \overline{\boldsymbol{\mathcal{U}}_\Phi \boldsymbol{\mathcal{U}}_\Phi^T} - \overline{\boldsymbol{\mathcal{U}}_\Phi}\,\overline{\boldsymbol{\mathcal{U}}_\Phi^T} = \overline{\boldsymbol{\Sigma}^2} - \overline{\boldsymbol{\mathcal{U}}_\Phi}\,\overline{\boldsymbol{\mathcal{U}}_\Phi^T} \in \mathbb{R}^{m\times m}.$$

The variance matrix is diagonal due to the orthogonality of $\boldsymbol{\Psi}$. Now:

$$\boxed{\boldsymbol{C}_S^T = \text{var}_{\mathcal{U}}^{-1}\text{cov}_{\mathcal{US}}}.$$

It is important to remark that since $\text{var}_{\mathcal{U}}$ is a diagonal matrix, the values of each of the coefficients in $\boldsymbol{C}_S$ do not depend on each other. As a consequence, the model for the subscales is composed of the addition of the independent contributions of each degree of freedom. This allows us to drop some of the entries in $\boldsymbol{C}_S$ and still obtain a meaningful model. This fact is the basis for the adjustments to the model proposed in the following section.

3.4 Final adjustments to the reduced-order model for the subscales

In this section we introduce two corrections to the model for the subscales which have an important impact on the results obtained from the final reduced-order model. The first correction aims at reducing the computational cost of the model for the subscales, the second one aims to improve its accuracy.

Dropping the components of the model with a negligible contribution

Pearson's correlation coefficient matrix is a measure of the correlation between two variables. It will serve us to decide whether to include the contribution of a component of the solution in the model for the subscales or not. Let us start by defining the *diagonal variance matrix* of the subscales:

$$\begin{aligned} \text{var}_{\mathcal{S}} &\in& \mathbb{R}^{m \times m} \\ \text{var}_{\mathcal{S},ij} &=& \delta_{ij} \left[\overline{\boldsymbol{\mathcal{S}}_{\Phi}^{T} \boldsymbol{\mathcal{S}}_{\Phi}} - \overline{\boldsymbol{\mathcal{S}}_{\Phi}^{T}} \overline{\boldsymbol{\mathcal{S}}_{\Phi}} \right]_{ij} \end{aligned}$$

Pearson's correlation coefficient matrix is defined as:

$$\boldsymbol{P}_{S}^{T} = \text{var}_{\mathcal{U}}^{-1/2} \text{cov}_{\mathcal{US}} \text{var}_{\mathcal{S}}^{-1/2},$$

where $\text{var}^{1/2}$ denotes the diagonal standard deviation matrix. Note that we are taking advantage of the fact that the components of the snapshots expressed in the reduced-order basis are uncorrelated (since they were obtained using a SVD procedure).

The values in $\boldsymbol{P}_S$ will range from -1 to 1. Absolute values of $P_{S,ij}$ close to 1 indicate a high correlation between component i of the solution and component j of the subscales. Absolute values of $P_{S,ij}$ close to 0 indicate a low correlation between both quantities. Coefficients with a low correlation value have a low impact on the accuracy of the model for the subscales and can be dropped from the model matrix $\boldsymbol{C}_S$.

In our implementation of the model for the subscales, we have included a tolerance $\text{tol}_{\text{Pearson}}$, for which if $P_{S,ij} < \text{tol}_{\text{Pearson}}$ then we artificially fix $C_{S,ij} = 0$. Although we have not implemented it in the code used for the numerical examples, this would allow us to implement the $\boldsymbol{C}_s$ matrix in a sparse format. In the case of non-linear problems with a relatively large number of degrees of freedom m, this would help to reduce the computational cost of computing $\boldsymbol{A}_{\Phi}\boldsymbol{C}_S$.

Discarding the components of the model with large error

The model as defined until now is clear from coefficients which are weakly correlated to the error associated to each of the degrees of freedom of the reduced-order model. However, it is still possible that the model has a dependence on degrees of freedom which have a very large associated error, that is, which cannot be properly captured by the reduced-order model. Even with the subscales correction, the reduced-order model will hardly be able to reproduce these degrees of freedom accurately. Basing the model for the subscales on these degrees of freedom will result in poor performance and unstable behavior.

In order to avoid this issue, we will also drop from the coefficient matrix all the terms in the columns corresponding to degrees of freedom for which the relative error of the reduced-order model is larger than a tolerance, $\text{tol}_{\text{error}}$. We define the relative error associated to the ith degree of freedom as:

$$e_i = \frac{||\boldsymbol{\mathcal{S}}_{\Phi(i,:)}||}{||\boldsymbol{\mathcal{U}}_{\Phi(i,:)}||}.$$

If $e_j > \text{tol}_{\text{error}}$, then we drop (set to zero) all the coefficients in column j of matrix $\boldsymbol{C}_S$:

$$\boldsymbol{C}_{S(:,j)} = \boldsymbol{0}^{m \times 1} \quad \text{if } e_j > \text{tol}_{\text{error}}.$$

This means that coefficients with large error are not used to model the subscales, although we do try to model the subscales for degree of freedom j, that is, $\boldsymbol{C}_{S(j,:)} \neq \boldsymbol{0}^{1 \times m}$. Again this can be easily done because due to the orthogonality of $\boldsymbol{\Psi}$, the snapshots' variance matrix is diagonal.

3.5 Algorithm for the computation of the subscales model

In Alg. 1 we summarize the algorithm for computing the model for the subscales:

Algorithm 1 Algorithm for the calculation of the subscales model.

---
1. Collect the high-fidelity solution snapshots $\mathcal{U}, \mathcal{U}_0$.

2. Perform the singular value decomposition of $\mathcal{U}$, compute the restrictions $\mathcal{R}_\Phi(\mathcal{U})$, $\mathcal{R}_\Phi(\mathcal{U}_0)$.

3. For all the snapshots initial conditions $\mathcal{R}_\Phi(\mathcal{U}_0)$, solve the reduced-order model problem and obtain the subscales snapshots matrix $\mathcal{S}_\Phi$ by comparing the reduced-order solution with $\mathcal{R}_\Phi(\mathcal{U})$.

4. Compute the variance of the snapshots expressed in the reduced-order basis $\mathrm{var}_\mathcal{U}$.

5. Compute the variance of the subscales expressed in the reduced-order basis $\mathrm{var}_\mathcal{S}$.

6. Compute the covariance of the snapshots against the subscales $\mathrm{cov}_{\mathcal{US}}$.

7. Compute $\boldsymbol{C}_S^T = \mathrm{var}_\mathcal{U}^{-1}\mathrm{cov}_{\mathcal{US}}$.

8. Compute the Pearson's correlation coefficient matrix $\boldsymbol{P}_S$.

9. Modify $\boldsymbol{C}_S$ by dropping the coefficients $C_{S,ij}$ for which $P_{ij} < \mathrm{tol}_{\mathrm{Pearson}}$.

10. Compute the relative error associated to each degree of freedom.

11. Modify $\boldsymbol{C}_S$ by dropping the columns $C_{S(:,j)}$ for which $e_j > \mathrm{tol}_{\mathrm{error}}$.

12. Compute $\boldsymbol{D}_S^T = \overline{\mathcal{S}_\Phi^T} - \overline{\mathcal{U}_\Phi^T}\boldsymbol{C}_S^T$.

---

4 Numerical examples

In this section we present some numerical examples which illustrate the performance of the reduced-order subscales in practical cases. We first focus in problems with few degrees of freedom, for which we show the capability of the subscales model to improve the accuracy of the solution obtained with the reduced-order model. We also show that, since the subscales are trained specifically for each reduced-order model, they can help to correct the additional errors introduced by hyper-reduction. Then we move to problems with varying conditions, for which we show that the reduced-order model with the subscales is capable to adapt to variations in the physical properties of the continuum medium. Finally we focus in a problem for which, past a certain point, increasing the number of degrees of freedom of the reduced-order solution leads to a poor improvement of accuracy. We show that in these cases the reduced-order subscales are capable of achieving an accurate solution with a relatively low number of degrees of freedom.

4.1 Improving the performance of the reduced-order model

In the first numerical example we apply the developed method to a low Reynolds number incompressible flow. The incompressible Navier-Stokes equations consist in finding a velocity field $\boldsymbol{u}$ and a pres-
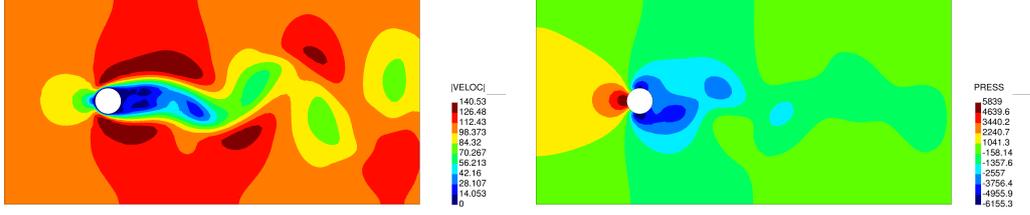
Figure 2: Velocity and pressure fields for the incompressible flow past a cylinder.

sure p such that:

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \nabla p = \boldsymbol{f},$$
$$\nabla \cdot \boldsymbol{u} = 0.$$

in the computational domain Ω and for time $t > 0$, where $\boldsymbol{f}$ is the vector of body forces and ν the kinematic viscosity. Appropriate initial and boundary conditions have to be appended to this problem. The full-order model consists of an equal interpolation finite element method stabilized using the Variational Multiscale Method [29]. Details and the motivation for the used formulation can be found in [18].

In this numerical example we study the incompressible flow around a cylinder at $\mathrm{Re} = 100$. The computational domain consists of a 16×8 rectangle with a unit-diameter cylinder centered at $(4, 4)$. The horizontal inflow velocity is set to 100 at $x = 0$. Slip boundary conditions which allow the flow to move in the direction parallel to the walls are set at $y = 0$ and $y = 8$, and velocity is set to 0 at the cylinder surface. The viscosity has been set to $\nu = 1$, which yields a Reynolds number $\mathrm{Re} = 100$ based on the diameter of the cylinder and the inflow velocity.

Regarding the numerical discretization, a backward Euler scheme has been used for the time integration with time step $\delta t = 0.001$. The advective non-linear terms in the left-hand-side matrix have been treated in an explicit way, considering an advective velocity $\boldsymbol{u}^{n+1*}$, an explicit approximation to $\boldsymbol{u}^{n+1}$. For Backward Euler Schemes we take:

$$\boldsymbol{u}^{n+1*} = \boldsymbol{u}^n.$$

In this example, a coarse 7294 linear element mesh has been used to solve the problem. Fig. 2 shows the velocity norm and pressure fields at a certain time step after a statistically stationary state has been achieved.

For the reduced-order model, the Petrov-Galerkin projection defined in (8) is used due to the fact that the matrix of the problem is not symmetric. In order to build the reduced-order basis, 50 snapshots of the solution are taken, chosen so that they are extracted equally spaced in time in the original full-order simulation. The solution to this problem is not complex, and it is sufficient to keep only the first 10 basis functions to obtain an accurate representation of the solution using the reduced-order model, see [8] where this problem is solved using an explicit reduced-order model. In order to illustrate the effect of the subscales, however, we are going to solve it using 2 reduced-order basis functions. If only 2 degrees of freedom are used, the reduced-order model is not capable of capturing the solution, and the resulting error is large. However, we will see how adding the subscales to this very poor model allows us to significantly improve its performance. The model for the subscales has been trained using $\mathrm{tol}_{\mathrm{Pearson}} = 0.01$ and $\mathrm{tol}_{\mathrm{error}} = 0.4$.
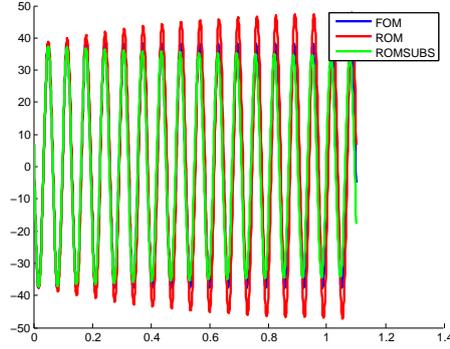
Figure 3: Time history of the vertical velocity at (6.5,4). Comparison of the full-order model, reduced-order model without subscales, and the reduced order model with subscales.
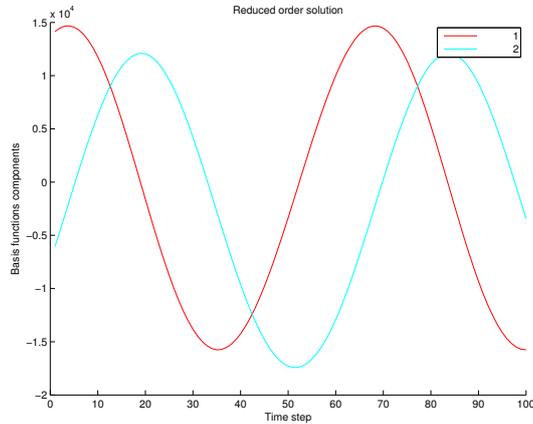

Figure 4: Reduced-order solution (values for each degree of freedom) in time. 2 degree-of-freedom model.

Fig. 3 shows a comparison of the time history of the vertical velocity at coordinates $(6.5, 4)$. In the full-order model solution, the amplitude of the vertical velocity oscillation is approximately 38. When the reduced-order model with 2 degrees of freedom is used, the obtained amplitude is 48. The error is large because the reduced-order model is not rich enough. However, when the model for the subscales is considered, the obtained amplitude is 35, much closer to the real amplitude, although still not perfect. The differences between the full-order and the reduced-order model with the subscales can be understood if we study the quality of the modeling for the subscales. Fig. 4 shows the time history of the reduced-order solution in time for the two first basis functions. We can see that the frequency of the time history for both degrees of freedom is relatively low. Since the modeled subscales have a linear dependency on the solution, the frequency of the modeled subscales will also be low. As a consequence, the modeled subscales will not be able to adapt to high-frequency oscillations of the real subscales. This is illustrated in Fig. 5, where the real subscales are compared to the linearly modeled subscales. It is clear from this figure that the modeled subscales are not capable to adapt to the higher frequency real subscales, which results in modeled subscales which are quite different from the real subscales. In spite of this, the subscales turn out in an important improvement in the solution.

16
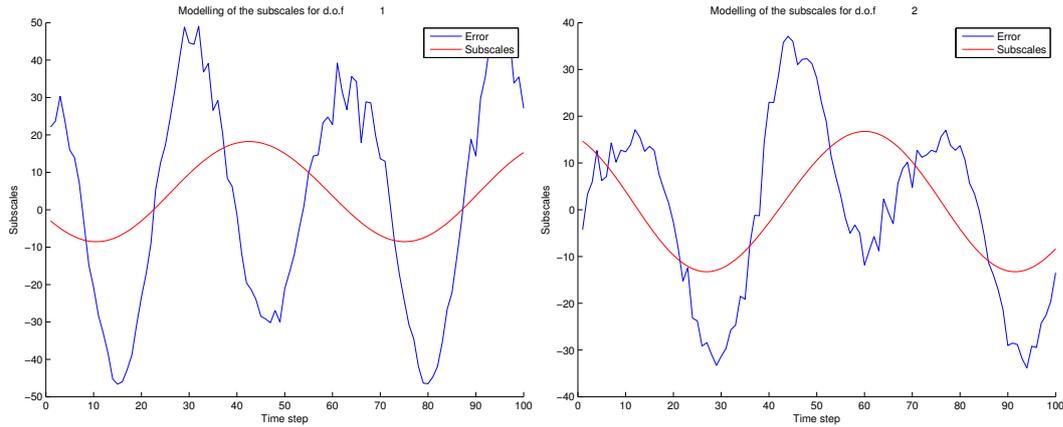
Figure 5: Real (Error) versus modeled (Subscales) subscales for the reduced-order model with 2 degrees of freedom.

Let us now do the same exercise for a reduced-order with 5 degrees of freedom. In this case the model without the subscales already does a decent job in capturing the full-order solution, so it is not worth comparing the reduced-order solution with and without the subscales. However, it is interesting to see how the quality of the modeled subscales increases as we increase the number of degrees of freedom of the model.

Fig. 6 shows the time history of the reduced-order solution when 5 degrees of freedom are used. The 2 first degrees of freedom are the same as in the previous case, but now we have 3 additional degrees of freedom. These degrees of freedom have larger oscillation frequencies. Since the subscales depend linearly on the solution, this means that the modeled subscales will have a better capability to adapt to high frequency oscillations. This is illustrated in Fig. 7, where the real subscales versus the modeled subscales are plot. We see that the low frequencies of the first and second components of the subscales are properly captured now by the linear subscale model. However, for the fifth degree of freedom, the oscillation frequency of the real subscales is larger than the highest oscillation frequency for all the degrees of freedom, and the subscales cannot be accurately represented by the linear model.

It is observed recurrently that the subscales for the last degrees of freedom of the reduced-order model cannot be represented accurately by the linear model. However, these degrees of freedom are the ones with a lower relative weight in the solution (this is ensured by the properties of the Singular Value Decomposition) and the effect of this poor modeling in the accuracy of the final solution is small. Nevertheless, this opens the door to introducing a non-linear model for the subscales, a possibility not explored in this work.

4.2 Correcting the errors introduced by hyper-reduction

The system defined in equation (8) is the reduced order model associated to the original system (1). System (8) is of dimension m (which typically corresponds to few degrees of freedom). However, for non-linear problems, matrix $\boldsymbol{A}$ and vector $\boldsymbol{R}$ still need to be computed at each time step in order to build the reduced-order system matrix and right-hand-side, since they non-linearly depend on the unknowns. The cost of this operation is of order M, the dimension of the original system (1), which means that the expected orders of magnitude reduction of computational cost is not observed in practice.

Recently, a general approach for dealing with the non-linear terms of reduced order models has appeared [37, 27, 9, 16, 40, 41, 4, 46, 45, 8], giving rise to what are known as *hyper-reduced* order models. In these methods, the equations governing the problem of interest are only integrated at certain
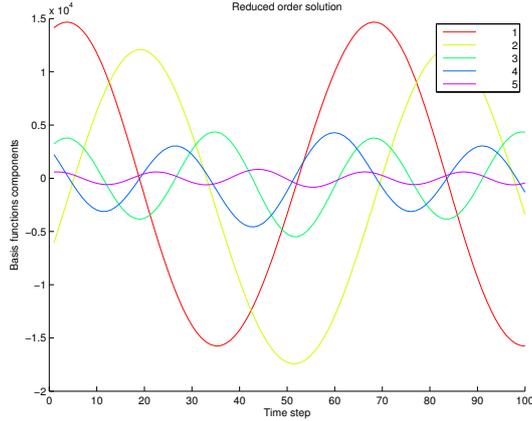
17

Figure 6: Reduced-order solution (values for each degree of freedom) in time. 5 degree-of-freedom model.

reduced-order integration points, instead of integrating them in the integration points of the full-order problem. The non-linear and parameter-dependent terms can then be recovered by means of a least-squares procedure from the values at the reduced-order integration points where the function to be approximated is computed. The use of hyper-reduction approaches for reconstructing $\boldsymbol{A}$ and $\boldsymbol{R}$ allows one to reduce the computational cost associated to the construction of the reduced-order system to $\mathcal{O}(m)$. However, using hyper-reduction also means that the reduced-order equations are integrated with less precision, resulting in sometimes less accurate models.

In this section we are going to show how the reduced-order subscales can help to reduce the error introduced by hyper-reduction. We depart from the case treated in the previous example and we apply to it a hyper-reduction scheme. Particularly, we are using a strategy similar to the explicit scheme described in [8], although with the following particularities:

- The equations are treated in an implicit manner.

- No least-squares extrapolation of the approximated unknowns is done.

- No special algorithm for selecting the reduced-order integration points is used, integration points are selected randomly amongst the nodes of the finite element mesh.

The objective of these simplifications is to on purpose have a hyper-reduced order model with an accuracy much worse than the model described in [8], and obviously, also much worse than the original reduced-order model without hyper-reduction.

After constructing the reduced-order model with the described properties, we train the reduced-order subscales using the process described in Section 3. Again $\text{tol}_{\text{Pearson}} = 0.01$ and $\text{tol}_{\text{error}} = 0.4$ have been used. Note that the subscales are trained *after* the reduced-order model is built, and *specifically* for it. The subscales for the hyper-reduced order model are going to be different from the subscales for the model without hyper-reduction. The subscales will be trained to correct the error of the model. In the case of hyper-reduction, this means that we will not only compensate for the error due to the low quality of the reduced-order subspace, but also for the error due to hyper-reduction.

This is illustrated in Fig. 8, where the 2 degrees of freedom hyper-reduced model is tested with and without subscales. As expected, the hyper-reduced model without the subscales performs much poorlier than the original reduced-order model. The oscillation amplitude in this case is 63, much larger than the correct oscillation amplitude of 38. This is so because, to the error due to the fact that only 2 degrees of freedom are used, we have to add the error due to hyper-reduction. We now add the subscales to the
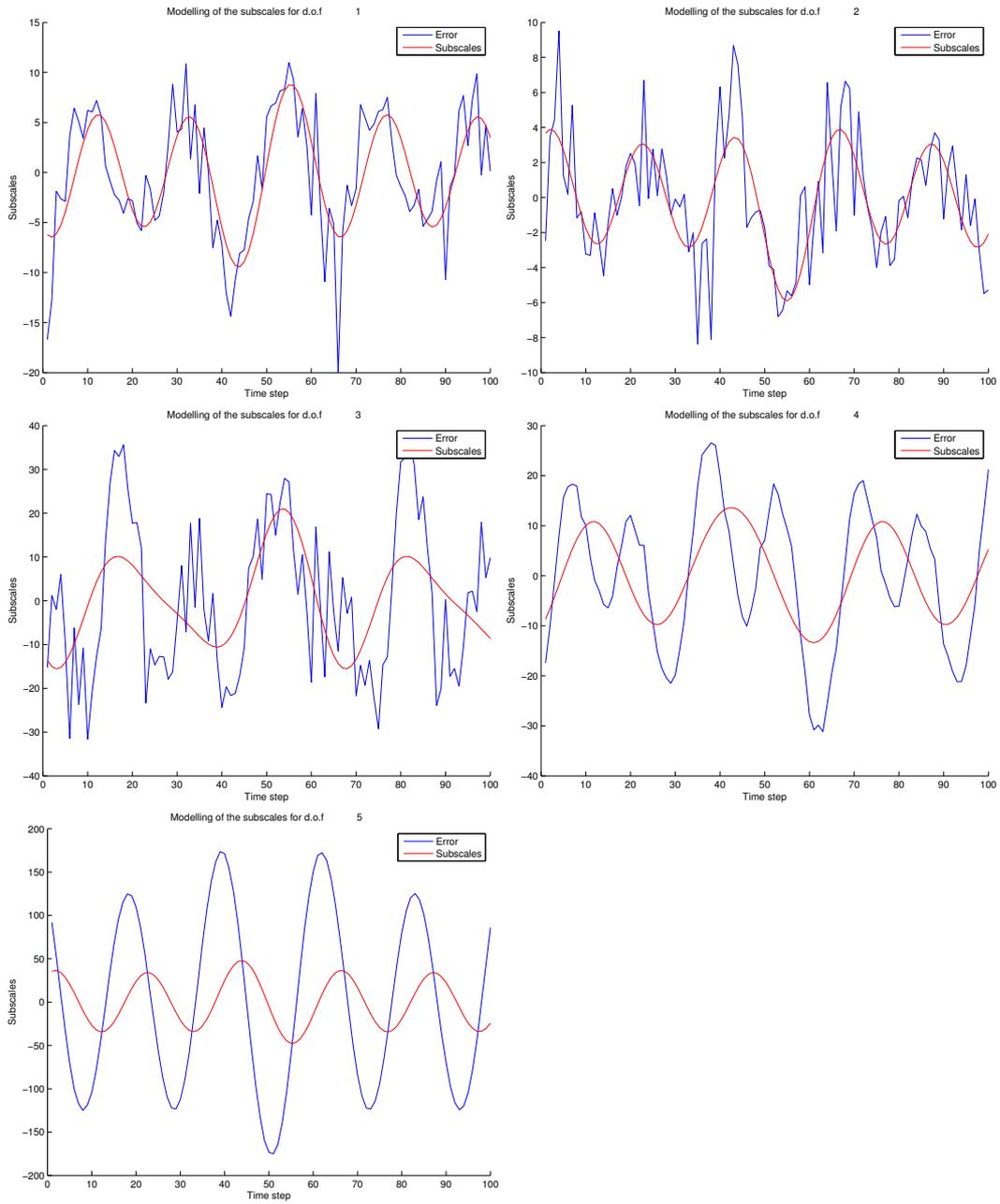
18

Figure 7: Real (Error) versus modeled (Subscales) subscales for the reduced-order model with 5 degrees of freedom.
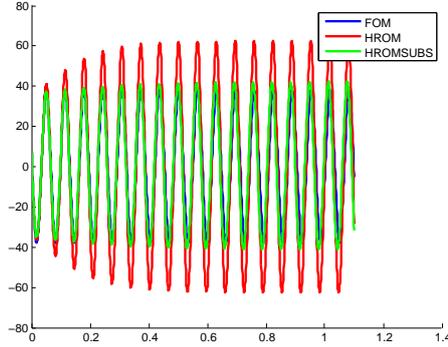
Figure 8: Time history of the vertical velocity at (6.5,4). Comparison of the full-order model, hyper-reduced model without subscales, and the hyper-reduced model with subscales.

model, which were trained specifically for the hyper-reduced model. The obtained amplitude is 42, the error of the hyper-reduced model with the subscales being almost equal to the error of the reduced-order model with the subscales (amplitude 35), and much better than the error of the reduced-order model without the subscales (amplitude 48).

From this example we conclude that the subscales can be successfully used to improve the performance of hyper-reduced models.

4.3 Adapting to variable physical parameters

The objective of this section is to show that, despite the fact the subscales are modeled a posteriori in a heuristic (non physics-based) manner, the model with the subscales is capable of adapting to changes in the physical properties of the continuum medium.

In this numerical example we are going to develop a reduced-order model which is capable of solving the flow past a cylinder at Reynolds numbers ranging from $\mathrm{Re} = 100$ to $\mathrm{Re} = 200$. The problem setting is the same as in the previous sections, except for the fact that now we are using a second order backward differences scheme for the time discretization, which means that the initial conditions snapshots involve the two preceding time steps. In this case the explicit advective velocity is taken as:

$$\boldsymbol{u}^{n+1*} = 2\boldsymbol{u}^n - \boldsymbol{u}^{n-1}.$$

In order to build the reduced-order model, we construct a reduced-order basis using 100 snapshots taken in a developed flow at $\mathrm{Re} = 100$, and 100 snapshots taken from a developed flow taken at $\mathrm{Re} = 200$. Again, the snapshots are chosen so that they are extracted equally spaced in time in the original full-order simulation. From these snapshots we keep the first 15 reduced-order basis functions from which we build the reduced-order model using the Petrov-Galerkin projection (8). In the case of the reduced-order model with the subscales, these are trained using the same snapshots as those used to form the reduced-order basis, and again we take $\mathrm{tol}_{\mathrm{Pearson}} = 0.01$ and $\mathrm{tol}_{\mathrm{error}} = 0.4$.

Fig. 9 shows the comparison of the full-order model, the reduced-order model without subscales, and the subscales enhanced reduced-order model for $\mathrm{Re} = 100$, $\mathrm{Re} = 150$, $\mathrm{Re} = 200$ and $\mathrm{Re} = 1000$. We can see how the reduced-order model without the subscales quickly looses amplitude in the $\mathrm{Re} = 100$ case, while it overestimates amplitude in the $\mathrm{Re} = 150$ and $\mathrm{Re} = 200$ cases. The model
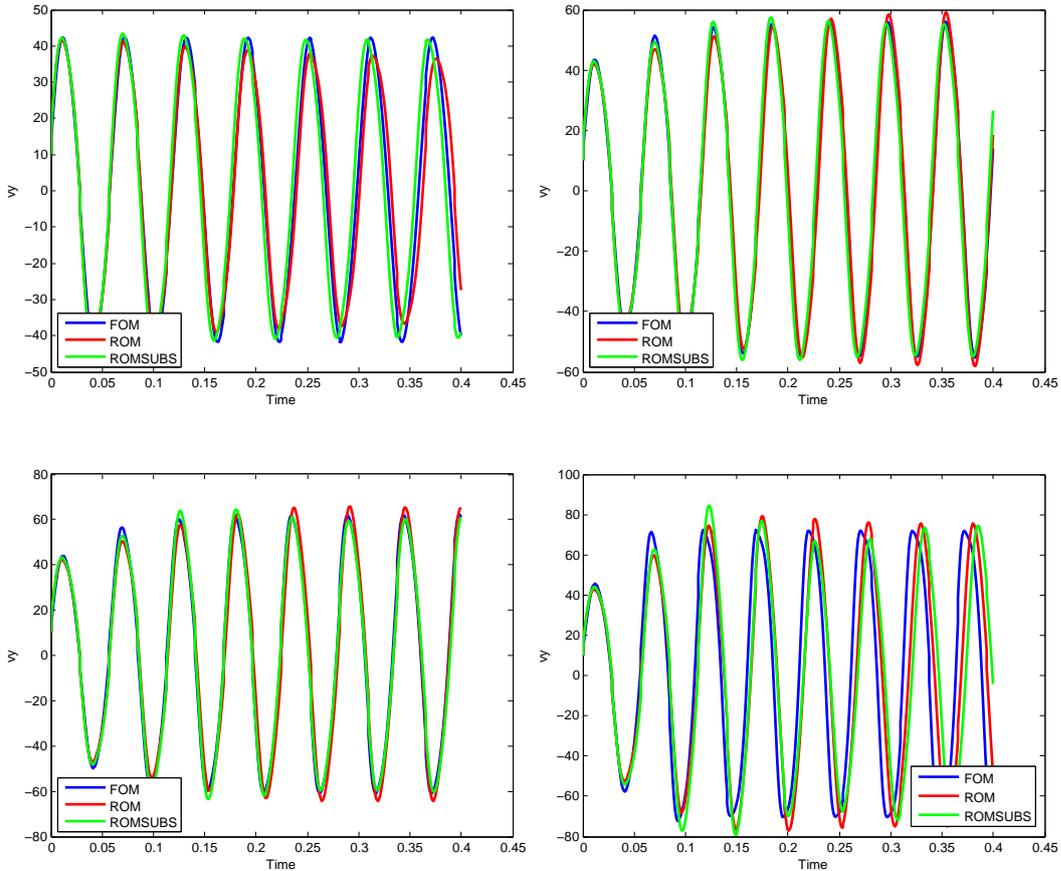
20

Figure 9: Comparison of the performance of the reduced-order model for several Reynolds numbers. Top Left: Reynolds 100. Top Right: Reynolds 150. Bottom Left: Reynolds 200. Bottom Right: Reynolds 1000

with the subscales, on the other hand, achieves a much more accurate representation, almost perfectly matching the full-order results in these cases. For the $\mathrm{Re} = 1000$ case, both the reduced-order model with and without the subscales show results which significantly differ from the full-order results. This is so because the solution of the $\mathrm{Re} = 1000$ case cannot be represented with the basis functions captured at $\mathrm{Re} = 100$ and $\mathrm{Re} = 200$.

From this example we conclude that the subscales enhanced reduced-order model is able to adapt to the changes in the physical parameters of the continuum medium, as long as these changes are sufficiently small so that the solution at the new physical parameters can be accurately enough represented by the reduced-order basis.

4.4 Using the subscales to solve complex flow problems

This section covers the perhaps most interesting numerical example amongst the ones presented, and the one that motivated this work. The example consists in the two-dimensional turbulent backward facing step at $\mathrm{Re} = 37000$, which we want to be able to simulate using reduced-order models.

The computational domain consists of a 44×9 rectangle. The 1 unit height step is placed at (4,0). A horizontal inflow velocity profile with mean velocity 1 is set at $x = 0$. A wall law boundary condition

is set at the top and bottom boundaries, with the wall distance characterizing the wall model being $\delta = 0.001$. The viscosity has been set to $\nu = 2.5 \cdot 10^{-5}$, which yields a Reynolds number Re = 37000 based on the size of the step and the inflow velocity. For the time integration, a second order backward differences scheme has been used with time step $\delta t = 0.2$. Again, an explicit approximation to the advective velocity has been used:

$$\boldsymbol{u}^{n+1*} = 2\boldsymbol{u}^n - \boldsymbol{u}^{n-1}.$$

A 34389 linear element mesh has been used to solve the problem.

A snapshot of the velocity norm and pressure full-order fields after the flow has been completely developed is shown in Fig. 10. The most critical zone is the step, where the vortexes are generated: the quality of the numerical solution in this area determines the quality of the simulation in the downstream region where the vortexes are propagated. The simulation of the vortex generation phenomena has turned out to be challenging for POD based reduced-order models, because the contribution of high-frequency modes is critical, and these are precisely the modes which the reduced-order model omits.

Let us first study the performance of the reduced-order model without the subscales. Again, a Petrov-Galerkin projection (8) is used for the discretization of the reduced-order model. In order to build the reduced-order basis, 200 equally-spaced-in-time snapshots of the solution are taken and two reduced-order models are considered: the first one with 35 degrees of freedom, and the second one with 80.

Fig. 11 compares the vertical velocity time history at a control point behind the step $(5, 1)$ for the full-order model, the reduced-order model with 35 degrees of freedom, and the reduced-order model with 80 degrees of freedom. It can be observed that the reduced-order models suffer from large errors. Moreover, increasing the number of degrees of freedom from 35 to 80 has little effect. This is so because in turbulent problems properly modeling the local dissipation is critical for accurately capturing the solution of the problem. This modeling of the local dissipation is done through turbulence models (the Smagorinsky turbulence model in this case), but the model is not taking into account the effect of the neglected reduced-order modes, the reduced-order subscales. This is the reason why a very large number of degrees of freedom is required for accurately capturing the solution to turbulent problems, even if a low number of degrees of freedom would suffice to accurately represent the solution.

However, the reduced-order subscales prove to be a suitable numerical tool to deal with this problem. The 35 degrees of freedom model is considered, and the subscales are trained using $\text{tol}_{\text{Pearson}} = 0.01$ and $\text{tol}_{\text{error}} = 0.3$. Fig. 12 compares the time history and vertical velocity spectrum for the backward facing step case using the reduced-order model with and without the modeling of the subscales. It can be observed that the reduced-order model using the subscales almost perfectly matches the results from the full-order model. With respect to the vertical velocity spectrum, Fig. 12 shows how the kinetic energy associated to high-frequency modes is over-estimated if no modeling for the subscales is used, while a good agreement is obtained when using the modeled subscales.

Finally, we test the behavior of the reduced-order model when the Reynolds number is modified. Fig. 13 shows the performance of the reduced-order model trained at $\text{Re} = 37000$, for several Reynolds numbers. At $\text{Re} = 32000$ the reduced-order model performs very well, and is capable of adapting to the diminution on the amplitude of the oscillations (from 0.41 to 0.39). Increasing the Reynolds number to $\text{Re} = 74000$ also results in good results: the reduced-order model is capable of capturing the increase in the oscillations amplitude (from 0.41 to 0.43). However, it is not capable of capturing the diminution on the oscillations frequency, and a phase difference is observed. This can be explained by observing Fig. 10, where it is clear that the reduced-order basis functions will only be able to reproduce oscillations in the wave-length of the original full-order model. If the frequency changes (and as a consequence so does the wave-length of the vortexes, the original basis functions cannot represent the new wave. Finally, for $\text{Re} = 18500$, the physics of the problem are too far away from the trained model, the subscales add
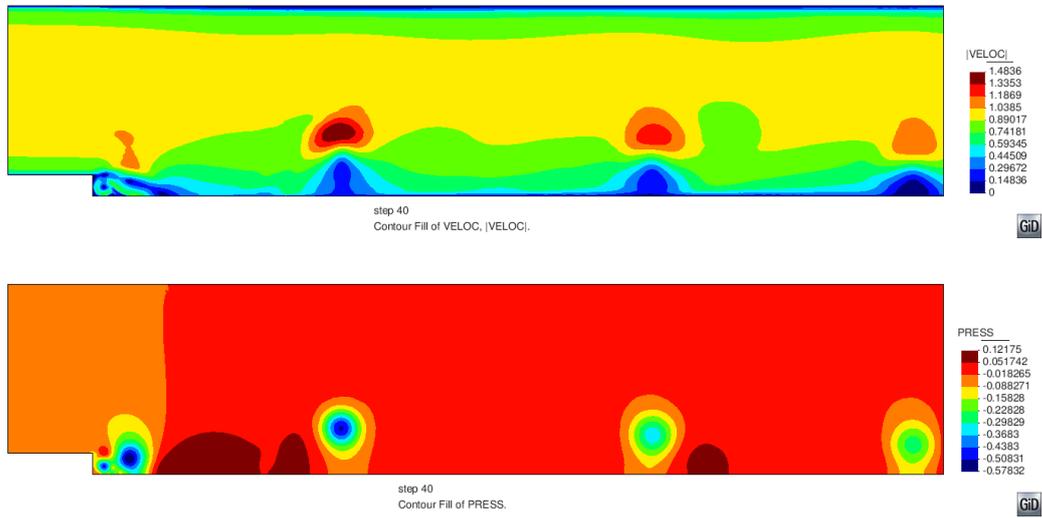
Figure 10: Velocity and pressure fields for the Backward Facing Step numerical example.
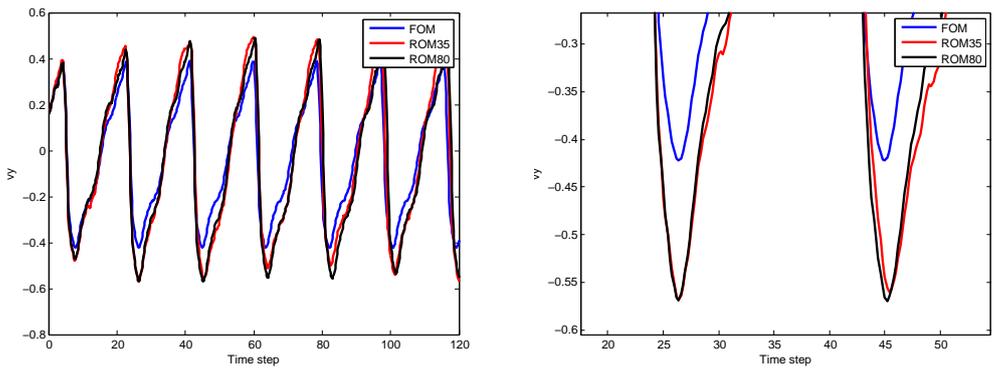

Figure 11: Comparison of results for the backward facing step case. Full-order model, reduced-order model with 35 degrees of freedom, reduced-order model with 80 degrees of freedom.
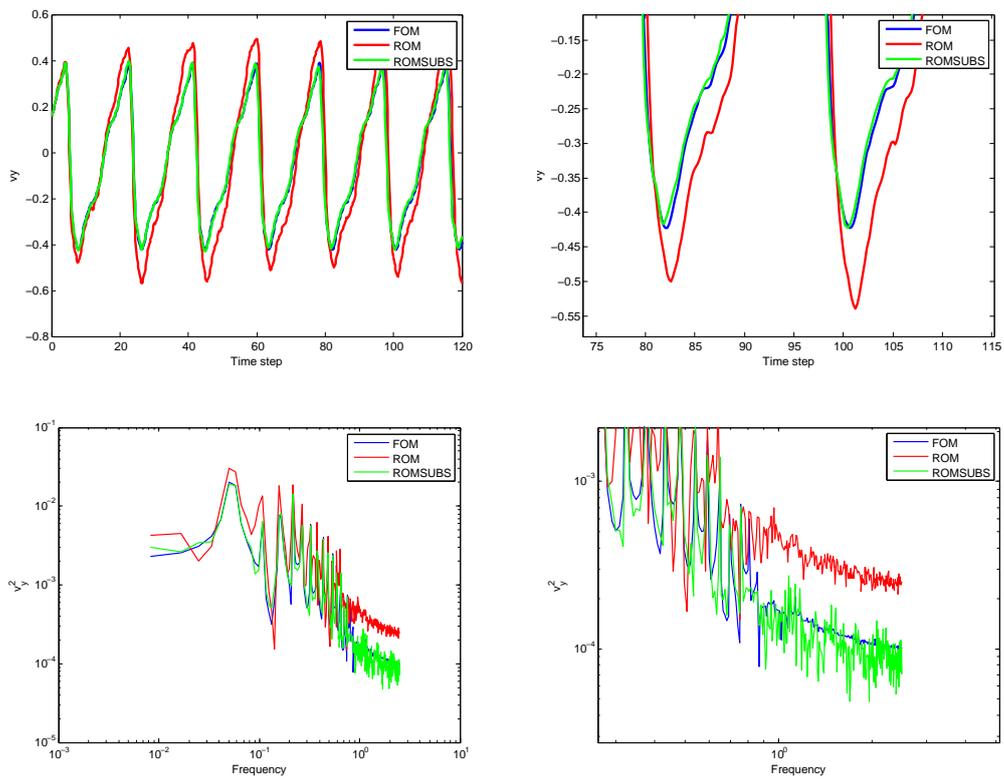
Figure 12: Comparison of results for the backward facing step case full-order model, reduced-order model without subscales (35 degrees of freedom), reduced-order model with subscales (35 degrees of freedom). Top: Vertical velocity time history. Bottom: Vertical velocity spectrum.
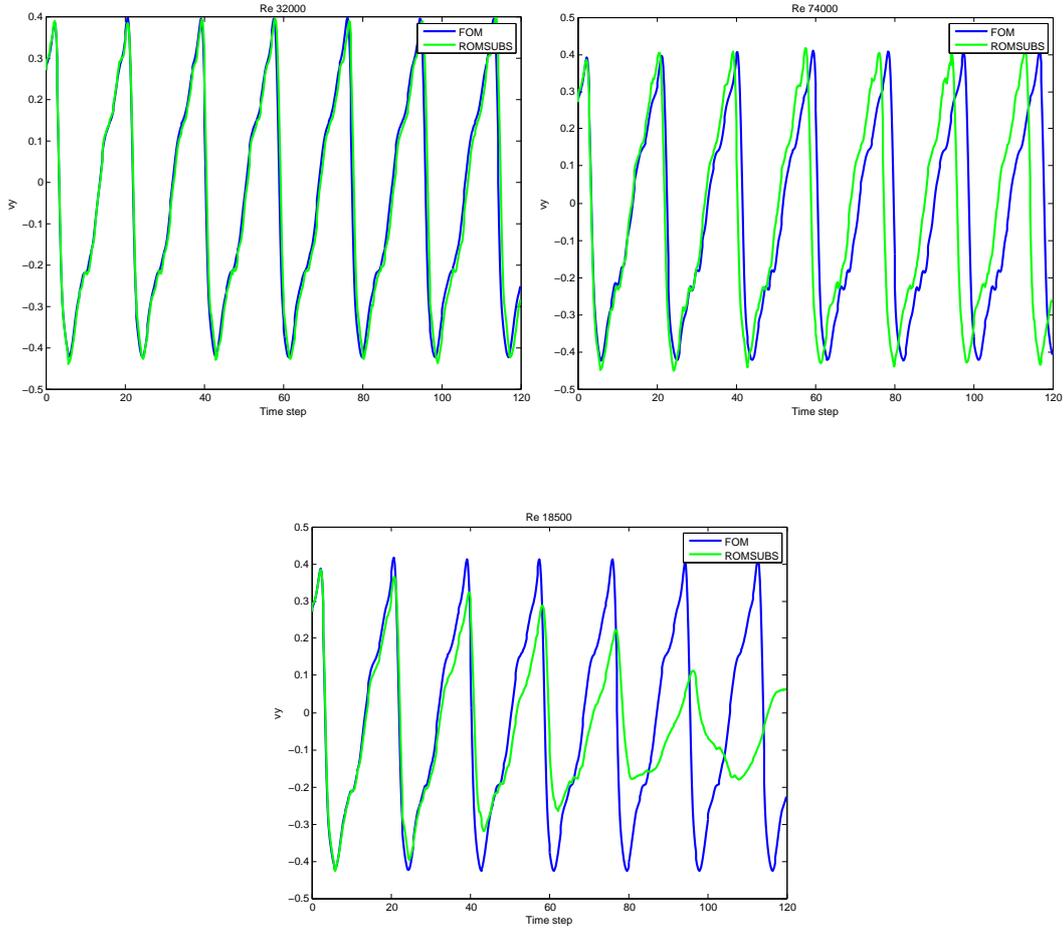
Figure 13: Behavior of the reduced-order model for different Reynolds numbers.

too much dissipation and the oscillations are progressively damped down. A new reduced-order model would be required to model the physics at this Reynolds number.

It can be concluded from this example that the reduced-order subscales are capable of modeling the effect of the high-frequency modes discarded in the POD model. Moreover, the computational cost of the reduced-order model is not significantly increased by adding the reduced-order subscales: the additional number of operations required to evaluate the subscales matrices is $\mathcal{O}\left(m^3\right)$, which is the computational complexity of solving the reduced-order system. The reduced-order model performs well in a range of Reynolds numbers close to the Reynolds number at which it was trained, but, as expected, the accuracy becomes poorer for physical parameters far away from the training ones.

5 Conclusions

In this work we have presented a model for the Reduced-Order Subscales of POD models. The basic idea consists in splitting the full-order solution into the part which can be captured by the reduced-order model and the part which cannot, the subscales. When this splitting is introduced in the equations of the reduced-order model, it becomes apparent that the terms accounting for the subscales are missing,

which causes POD models to produce inaccurate solutions if no remedy is devised. Since the reduced-order subspace is not capable of capturing the subscales, a model for them is required.

The proposed model for the subscales is defined as a linear function of the solution of the reduced-order model. The coefficients of this linear function are obtained by comparing the solution of the full-order model and the solution of the reduced-order model for some trial solutions, departing from known initial conditions. For convenience, these trial solutions are taken to be the snapshots used to construct the original POD basis. The difference between both solutions (full- and reduced-order) are the subscales, for which a model can be built using a least-squares procedure. The subscales are then introduced as a correction in the reduced-order model, resulting in an important improvement in accuracy.

The specific form of the least squares problem to be solved for computing the subscales model has been studied. It is shown that, thanks to the properties of the Singular Value Decomposition used to build the reduced-order model basis, the coefficients of the subscales model matrices can be computed independently. This allows us to drop some of the terms in order to improve the accuracy of the model and reduce the final computational cost.

Finally, the enhanced reduced-order model is tested in several numerical examples. These practical cases show that the use of the subscales leads to more accurate solutions, successfully corrects errors introduced by hyper-reduction, and allows to solve complex flow problems using a reduced number of degrees of freedom.

Some questions remain open. For instance, the analysis of the error of the reduced-order model with the subscales, the possibility of introducing more elaborate models which take into account a non-linear relation between the reduced-order solution and the subscales, or the possibility of taking into account the time dependence of the subscales. These will be the focus of future research on the topic of reduced-order subscales.

References

[1] I. Akhtar, J. Borggaard, and A. Hay. Shape sensitivity analysis in flow models using a Finite-Difference approach. *Mathematical Problems in Engineering*, 2010:1–23, 2010.

[2] E. Arian, M. Fahl, and E. W. Sachs. Trust-Region proper orthogonal decomposition for flow control. *Institute for computers*, pages 2000–2101, 2000.

[3] G. Artana, A. Cammilleri, J. Carlier, and E. MÃ©min. Strong and weak constraint variational assimilations for reduced order fluid flow modeling. *Journal of Computational Physics*, 231(8):3264 – 3288, 2012.

[4] P. Astrid. *Reduction of Process Simulation Models: a proper orthogonal decomposition approach*. PhD thesis, Department of Electrical Engineering, Eindhoven University of Technology, 2004.

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

[6] J. Baiges and R. Codina. A variational multiscale method with subscales on the element boundaries for the Helmholtz equation. *International Journal for Numerical Methods in Engineering*, 93(6):664–684, 2013.

[7] J. Baiges, R. Codina, and S. Idelsohn. A Domain Decomposition strategy for Reduced Order Models. Application to the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 267:23–42, 2013.

[8] J. Baiges, R. Codina, and S. Idelsohn. Explicit Reduced Order Models for the stabilized finite element approximation of the incompressible Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 72:1219–1243, 2013.

[9] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

[10] M Bergmann, C-H Bruneau, and A Iollo. Enablers for robust pod models. *Journal of Computational Physics*, 228(2):516–538, 2009.

[11] M. Bergmann, L. Cordier, and J.P. Brancher. Drag minimization of the cylinder wake by Trust-Region proper orthogonal decomposition. *Notes on Numerical Fluid Mechanics and Multidisciplinary Design*, 95:19, 2007.

[12] T. Bui-Thanh, K. Willcox, and O. Ghattas. Model reduction for Large-Scale systems with High-Dimensional parametric input space. *SIAM Journal on Scientific Computing*, 30(6):3270, 2008.

[13] J. Burkardt, M. Gunzburger, and H. Lee. POD and CVT-based reduced-order modeling of Navier-Stokes flows. *Computer Methods in Applied Mechanics and Engineering*, 196(1-3):337–355, 2006.

[14] K. Carlberg, C. Bou-Mosleh, and C. Farhat. Efficient non-linear model reduction via a least-squares Petrov-Galerkin projection and compressive tensor approximations. *International Journal for Numerical Methods in Engineering*, 86(2):155–181, 2011.

[15] A. Chatterjee. An introduction to the proper orthogonal decomposition. *Current Science*, 78(7):808–817, 2000.

[16] S. Chaturantabut and D C Sorensen. Discrete empirical interpolation for nonlinear model reduction. *Technical Report TR09-05, Rice University, Houston Texas*, 2009.

[17] R. Codina. Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 190:1579–1599, 2000.

[18] R. Codina. A stabilized finite element method for generalized stationary incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 190:2681–2706, 2001.

[19] R. Codina and J. Baiges. Finite element approximation of transmission conditions in fluids and solids introducing boundary subgrid scales. *International Journal for Numerical Methods in Engineering*, 87:386–411, 2011.

[20] R. Codina, J. Principe, and J. Baiges. Subscales on the element boundaries in the variational two-scale finite element method. *Computer Methods in Applied Mechanics and Engineering*, 198:838–852, 2009.

[21] R. Codina, J. Principe, O. Guasch, and S. Badia. Time dependent subscales in the stabilized finite element approximation of incompressible flow problems. *Computer Methods in Applied Mechanics and Engineering*, 196:2413–2430, 2007.

[22] M Couplet, C Basdevant, and P Sagaut. Calibrated reduced-order pod-galerkin system for fluid flow modelling. *Journal of Computational Physics*, 207(1):192–220, 2005.

[23] AE Deane, IG Kevrekidis, G Em Karniadakis, and SA Orszag. Low-dimensional models for complex geometry flows: Application to grooved channels and circular cylinders. *Physics of Fluids A: Fluid Dynamics (1989-1993)*, 3(10):2337–2354, 1991.

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

[25] B. Glaz, L. Liu, and P. P. Friedmann. Reduced-Order nonlinear unsteady aerodynamic modeling using a Surrogate-Based recurrence framework. *AIAA Journal*, 48(10):2418–2429, 2010.

[26] W. R Graham, J. Peraire, and K. Y Tang. Optimal control of vortex shedding using low-order models. Part I-Open-loop model development. *International Journal for Numerical Methods in Engineering*, 44(7):945–972, 1999.

[27] M.A. Grepl, Y. Maday, N.C. Nguyen, and A.T. Patera. Efficient Reduced-Basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(03):575–605, 2007.

[28] P. Holmes, J. L. Lumley, and G. Berkooz. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press, 1998.

[29] T.J.R. Hughes, G.R. Feijóo, L. Mazzei, and J.B. Quincy. The variational multiscale method–a paradigm for computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 166:3–24, 1998.

[30] I. Kalashnikova and M. F. Barone. Stable and Efficient Galerkin Reduced Order Models for Non-Linear Fluid Flow. In *AIAA-2011-3110, 6th AIAA Theoretical Fluid Mechanics Conference*, Honolulu, 2011.

[31] V L Kalb and A E Deane. An intrinsic stabilization scheme for proper orthogonal decomposition based low-dimensional models. *Physics of Fluids (1994-present)*, 19(5):054106, 2007.

[32] D.D. Kosambi. Statistics in function space. *J. Indian Math. Soc*, 7:76–88, 1943.

[33] T. Lassila and G. Rozza. Parametric free-form shape design with PDE models and reduced basis method. *Computer Methods in Applied Mechanics and Engineering*, 199(23-24):1583–1592, 2010.

[34] F. Lihong. Review of model order reduction methods for numerical simulation of nonlinear circuits. *Applied Mathematics and Computation*, 167(1):576–591, 2005.

[35] D. J. Lucia and P. S. Beran. Projection methods for reduced order models of compressible flows. *Journal of Computational Physics*, 188(1):252–280, 2003.

[36] C. Maier, A. Epureanu, V. Marinescu, and F. M. Bogdan. A new concept of the reduced order modeling in metal forming. In *Proceedings of the 6th WSEAS international conference on Dynamical systems and control*, CONTROL'10, pages 133–136, Stevens Point, Wisconsin, USA, 2010. World Scientific and Engineering Academy and Society (WSEAS).

[37] N.C. Nguyen and J. Peraire. An efficient reduced-order modeling approach for non-linear parametrized partial differential equations. *International Journal for Numerical Methods in Engineering*, 76(1):27–55, 2008.

[38] L. Perret, E. Collin, and J. Delville. Polynomial identification of pod based low-order dynamical system. *Journal of Turbulence*, (7), 2006.

[39] G. Rozza, T. Lassila, and A. Manzoni. Reduced basis approximation for shape optimization in thermal flows with a parametrized polynomial geometric map. In J. S. Hesthaven and E. M. Ranquist, editors, *Spectral and High Order Methods for Partial Differential Equations*, volume 76, pages 307–315. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[40] D. Ryckelynck. A priori hyperreduction method: an adaptive approach. *Journal of Computational Physics*, 202(1):346 – 366, 2005.

[41] D. Ryckelynck. Hyper-reduction of mechanical models involving internal variables. *International Journal for Numerical Methods in Engineering*, 77(1):75–89, 2009.

[42] S. Y. Shvartsman and I. G. Kevrekidis. Nonlinear model reduction for control of distributed systems: A computer-assisted study. *AIChE Journal*, 44(7):1579–1595, 1998.

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

[44] T. R. Smith, J. Moehlis, and P. Holmes. Low-dimensional modelling of turbulence using the proper orthogonal decomposition: a tutorial. *Nonlinear Dynamics*, 41(1-3):275–307, 2005.

[45] A. Verhoeven, J. Maten, M. Striebel, and R. Mattheij. Model order reduction for nonlinear ic models. In Adam Korytowski, Kazimierz Malanowski, Wojciech Mitkowski, and Maciej Szymkat, editors, *System Modeling and Optimization*, volume 312 of *IFIP Advances in Information and Communication Technology*, pages 476–491. Springer Berlin Heidelberg, 2009.

[46] A. Verhoeven, T. Voss, P. Astrid, E.J.W. ter Maten, and T. Bechtold. Model order reduction for nonlinear problems in circuit simulation. *PAMM*, 7(1):1021603–1021604, 2007.

[47] K. Veroy and A. T Patera. Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: rigorous reduced-basis a posteriori error bounds. *International Journal for Numerical Methods in Fluids*, 47(8-9):773–788, 2005.

[48] Zhu Wang, Imran Akhtar, Jeff Borggaard, and Traian Iliescu. Proper orthogonal decomposition closure models for turbulent flows: a numerical comparison. *Computer Methods in Applied Mechanics and Engineering*, 237:10–26, 2012.

[49] J. Yvonnet and Q. He. The reduced model multiscale method (R3M) for the non-linear homogenization of hyperelastic media at finite strains. *J. Comput. Phys.*, 223(1):341–368, 2007.