Available online at www.sciencedirect.com
ScienceDirect
Comput. Methods Appl. Mech. Engrg. 363 (2020) 112844
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

Projection-based reduced order models for flow problems: A variational multiscale approach

Ricardo Reyes[a,*], Ramon Codina[a,b]

[a] Universitat Politècnica de Catalunya (UPC), Jordi Girona 1-3, 08034 Barcelona, Spain
[b] International Centre for Numerical Methods in Engineering (CIMNE), C/ Gran Capità S/N, 08034 Barcelona, Spain

Received 10 July 2019; accepted 11 January 2020
Available online xxxx

Abstract

In this paper we present a Variational Multi-Scale stabilized formulation for a general projection-based Reduced Order Model. In the stabilized formulation we address techniques already analysed in Variational Multi-Scale-Finite Element methods: time-dependent subscales, non-linearity in the subscales approximation and orthogonality between the solution space and the subscale space. Additionally, we describe a mesh based hyper-Reduced Order Model technique and implement a Petrov–Galerkin projection technique. At the end of the article, we test the proposed Reduced Order Model formulation using the incompressible Navier–Stokes problem.

Keywords: Reduced Order Model (ROM); Finite element method; Variational Multi-Scale method (VMS); Hyper-reduction; Proper Orthogonal Decomposition (POD)

1. Introduction

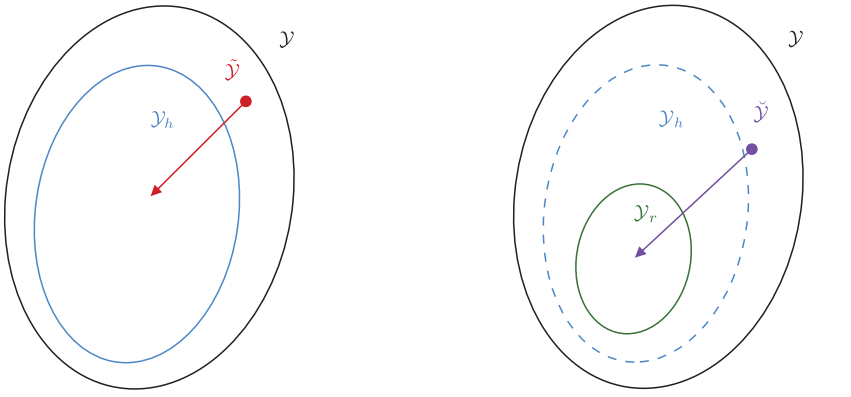
In this work we aim to formulate a residual-based stabilized method for projection-based Reduced Order Models (ROMs) by using the Variational Multi-Scale (VMS) idea introduced in [1]. First, we decide to look at the problem at hand from a different scope; instead of formulating the ROM as a projection of the high-fidelity problem that we wish to represent, we describe it as a variational problem on its own. Hence, we can consider the formulation as a projection of the spectral basis of the ROM over a discretized domain, thus making it somewhat a Spectral Element (SE) method. For linear problems both approaches coincide, but departing from the variational formulation provides more flexibility in the design of the approximation of nonlinear problems and simplifies its analysis.

We will assume throughout that our high fidelity model, to which we shall also refer as Full Order Model (FOM), is a Finite Element (FE) method. This, as we shall see, has significant implications in the design of the ROM.

Analogously to other Galerkin approximations (FE, SE, etc.), the standard Galerkin projection-based ROM might suffer from instabilities or ill-posedness depending on the formulation used. Likewise, due to the nature of model reduction which involves truncating the basis, some of the information obtained by the compression algorithm is lost. Therefore, a VMS stabilized formulation intents to work out two issues: instabilities related to the physical

* Corresponding author.
E-mail addresses: ricardo.reyes@upc.edu (R. Reyes), ramon.codina@upc.edu (R. Codina).

https://doi.org/10.1016/j.cma.2020.112844
0045-7825/



(a) FOM $\mathscr{Y}_h$ and FOM-OSGS $\tilde{\mathscr{Y}}$ spaces. (b) ROM $\mathscr{Y}_r$ and ROM-OSGS $\check{\mathscr{Y}}$ spaces.

Fig. 1. Illustration for $\mathscr{Y}$ and its subspaces in FOM and ROM.

model and under representation of low energy modes. Since in practice the projection-based ROM lies in between a SE method with equispaced nodes and a FE method, we can extrapolate some of the hypotheses derived to stabilized problems when using these two methods. For the specific case of VMS we follow the survey in [2] for FEs and the methods developed in [3,4] for SEs.

The key idea of VMS methods is to split the unknowns into their finite element component, i.e. the resolved scale, and the remainder, i.e. the subgrid scale or subscale. The choice of the VMS stabilization technique for the ROM is justified in the idea that the subscales are designed not only to stabilize the problem but also to represent their contribution and their interactions onto the resolved scales [5]. In that sense we have developed an equivalent model to the VMS-FE, which we believe allows us to add to the model not only the necessary stabilization but also a part that accounts for the high frequency modes left out from the reduced basis. Moreover, we add to the standard VMS formulation the choice of Orthogonal SubGrid Scales (OSGSs); this can be justified – besides the advantages in FE formulations described for example in [6] – by the orthonormal nature of the reduced space basis when using a method as the Proper Orthogonal Decomposition (POD) for the computation of such basis.

To illustrate the equivalence between the OSGSs in both the ROM and the FOM, Fig. 1 shows a comparison between FOM and ROM resolved scale —$\mathscr{Y}_h$ and $\mathscr{Y}_r$— and subscale —$\tilde{\mathscr{Y}}$ and $\check{\mathscr{Y}}$— spaces. These subspaces are such that $\mathscr{Y} = \mathscr{Y}_h \oplus \tilde{\mathscr{Y}} = \mathscr{Y}_r \oplus \check{\mathscr{Y}}$, where $\mathscr{Y}$ is the space of functions where the problem is defined, assumed to be a Hilbert space. Let us remark that $\check{\mathscr{Y}}$ is the complementary in $\mathscr{Y}$ of $\mathscr{Y}_r$, not the complementary in $\mathscr{Y}_h$, as it was done in [7]. Therefore, subscales of the ROM space may not belong to the FOM space.

Essentially, stabilization techniques formulated for projection-based ROMs – including the VMS-ROM formulation proposed here – consist in adding a stabilization term to the Galerkin formulation. Generally, this term can be written as the L^2 product within each element domain of an operator of the residual of the equation to be solved —$\boldsymbol{\mathcal{R}}$, an operator applied to the test function —$\boldsymbol{\mathcal{P}}$, and a matrix of numerical parameters —$\boldsymbol{\tau}$. To construct $\boldsymbol{\mathcal{R}}$ and $\boldsymbol{\mathcal{P}}$ we follow a VMS approach that includes the necessary conditions to define the OSGSs as done in [8]. Furthermore, to define $\boldsymbol{\tau}$ we follow the same Fourier analysis of the subscales as the one described in [9] for the FE method.

To develop the formulation we focus on FE-ROMs as in [10], where the spectral basis is obtained using a FE-FOM and then projected onto a discretized domain that is not necessarily of the same size as the FOM. This allows for the use of a mesh-based hyper-ROM —which relies in the use of coarser meshes to decrease the computational cost. We also include the Petrov–Galerkin projection as described in [11], which we interpret as a way to precondition the numerical system in order to overcome the singularity caused by the non-symmetric nature of the problem [12].

The paper is organized as follows. In Sections 2 and 3 we describe a general variational problem and a general model reduction method. In Section 4 we describe the stabilized VMS-FOM using FEs. In Section 5 we formulate the VMS-ROM, the way to approximate the subscales and present an analysis of their behavior to define the numerical parameters $\boldsymbol{\tau}$. In Section 6 we describe the discrete problem, including the Petrov–Galerkin projection

and the mesh based hyper-reduction. In Section 7 we compare the VMS-ROM proposed with other stabilization methods. In Section 8 we present numerical results that test the formulation, including a flow passed a cylinder and a backward facing step using the incompressible Navier–Stokes equation. And finally, in Section 9 we present some concluding remarks.

2. Variational problem

Let us start by describing a general convection–diffusion–reaction variational problem posed in a spatial domain $\Omega \subset \mathbb{R}^d$ with a boundary Γ and in a time interval from zero to a final time t_f, that consists in finding a vector function $\boldsymbol{y}(\boldsymbol{x}, t)$ of n components such that

$$\boldsymbol{\mathcal{M}}(\boldsymbol{y})\partial_t \boldsymbol{y} + \boldsymbol{\mathcal{L}}(\boldsymbol{y}; \boldsymbol{y}) = \boldsymbol{f}, \qquad \text{in } \Omega, \quad t \in \,]0, t_\mathrm{f}[, \tag{1}$$

where d is the number of space dimensions, $\boldsymbol{\mathcal{M}}(\boldsymbol{y})$ a mass matrix, $\boldsymbol{f}(\boldsymbol{x}, t)$ a forcing term, ∂_t the derivative over time, and $\boldsymbol{\mathcal{L}}$ a nonlinear differential operator in space of first or second order defined as $\boldsymbol{\mathcal{L}}(\boldsymbol{y}; \boldsymbol{z}) := \boldsymbol{A}_i^c(\boldsymbol{y})\partial_i \boldsymbol{z} + \boldsymbol{A}_i^f(\boldsymbol{y})\partial_i \boldsymbol{z} - \partial_i\big(\boldsymbol{K}_{ij}(\boldsymbol{y})\partial_j \boldsymbol{z}\big) + \boldsymbol{S}(\boldsymbol{y})\boldsymbol{z}$. $\boldsymbol{A}_i^c(\boldsymbol{y})$, $\boldsymbol{A}_i^f(\boldsymbol{y})$, $\boldsymbol{K}_{ij}(\boldsymbol{y})$ and $\boldsymbol{S}(\boldsymbol{y})$ are $n \times n$ matrices and ∂_i denotes differentiation with respect to the ith Cartesian coordinate x_i. Note that we take $\boldsymbol{\mathcal{L}}(\boldsymbol{y}; \boldsymbol{z})$ linear in $\boldsymbol{z}$.

Additionally, with $\Gamma = \Gamma_D \cup \Gamma_N$, the initial and boundary conditions are set as

$$\begin{aligned}
\boldsymbol{y}(\boldsymbol{x}, 0) &= \boldsymbol{y}_0(\boldsymbol{x}) && \text{in } \Omega, && t = 0,\\
\boldsymbol{\mathcal{D}}\boldsymbol{y} &= \boldsymbol{\mathcal{D}}\boldsymbol{y}_D && \text{on } \Gamma_D, && t \in \,]0, t_\mathrm{f}[,\\
\boldsymbol{\mathcal{F}}(\boldsymbol{y}; \boldsymbol{y}) &= \boldsymbol{t}_N && \text{on } \Gamma_N, && t \in \,]0, t_\mathrm{f}[,
\end{aligned}$$

where $\boldsymbol{y}_0$ is the prescribed initial condition, $\boldsymbol{y}_D$ is the prescribed Dirichlet boundary condition, $\boldsymbol{\mathcal{D}}$ is the Dirichlet operator associated to $\boldsymbol{\mathcal{L}}$, $\boldsymbol{\mathcal{F}}(\boldsymbol{y}; \boldsymbol{z}) := n_i \boldsymbol{K}_{ij}(\boldsymbol{y})\partial_j \boldsymbol{z} - n_i \boldsymbol{A}_i^f(\boldsymbol{y})\boldsymbol{z}$ is a flux operator, $\boldsymbol{n}$ is the normal to Γ whose ith component is n_i, and $\boldsymbol{t}_N$ is the prescribed Neumann boundary condition.

Let us denote by $\mathscr{Y}$ the space of functions in Ω where $\boldsymbol{y}$ is sought for each time t, incorporating the Dirichlet boundary conditions $\boldsymbol{\mathcal{D}}\boldsymbol{y} = \boldsymbol{\mathcal{D}}\boldsymbol{y}_D$, and let $\mathscr{Y}_0$ be its corresponding space of time independent test functions that satisfy $\boldsymbol{\mathcal{D}}\boldsymbol{v} = \boldsymbol{0}$ on Γ_D, both assumed to be subspaces of $L^2(\Omega)^n$. Let also $\langle \cdot, \cdot \rangle$ be the integral over Ω of the product of two functions, assumed to be well defined, and let $(\cdot, \cdot)$ be the L^2-inner product in Ω. We can write the variational form of the problem as finding $\boldsymbol{y} :]0, t_\mathrm{f}[\longrightarrow \mathscr{Y}$ such that

$$\begin{aligned}
(\boldsymbol{\mathcal{M}}(\boldsymbol{y})\partial_t \boldsymbol{y}, \boldsymbol{v}) + \langle \boldsymbol{\mathcal{L}}(\boldsymbol{y}; \boldsymbol{y}), \boldsymbol{v} \rangle &= \langle \boldsymbol{f}, \boldsymbol{v} \rangle, && t \in \,]0, t_\mathrm{f}[, && \forall \boldsymbol{v} \in \mathscr{Y}_0,\\
(\boldsymbol{y}, \boldsymbol{v}) &= (\boldsymbol{y}_0, \boldsymbol{v}), && t = 0, && \forall \boldsymbol{v} \in \mathscr{Y}_0.
\end{aligned} \tag{2}$$

Now, denoting by $\langle \cdot, \cdot \rangle_{\Gamma_N}$ the integral of the product of two functions defined over Γ_N and defining the forms

$$\begin{aligned}
B(\boldsymbol{y}; \boldsymbol{z}, \boldsymbol{v}) &= \langle \boldsymbol{A}_i^c(\boldsymbol{y})\partial_i \boldsymbol{z}, \boldsymbol{v} \rangle - \langle \boldsymbol{z}, \partial_i(\boldsymbol{A}_i^f(\boldsymbol{y})^\top \boldsymbol{v}) \rangle + \langle \boldsymbol{S}(\boldsymbol{y})\boldsymbol{z}, \boldsymbol{v} \rangle + \langle \boldsymbol{K}_{ij}(\boldsymbol{y})\partial_j \boldsymbol{z}, \partial_i \boldsymbol{v} \rangle,\\
L(\boldsymbol{v}) &= \langle \boldsymbol{f}, \boldsymbol{v} \rangle + \langle \boldsymbol{t}_N, \boldsymbol{v} \rangle_{\Gamma_N},
\end{aligned}$$

we can write an equivalent problem to Eq. (2): find a vector function $\boldsymbol{y} :]0, t_\mathrm{f}[\to \mathscr{Y}$ such that

$$(\boldsymbol{\mathcal{M}}(\boldsymbol{y})\partial_t \boldsymbol{y}, \boldsymbol{v}) + B(\boldsymbol{y}; \boldsymbol{y}, \boldsymbol{v}) = L(\boldsymbol{v}), \qquad \forall \boldsymbol{v} \in \mathscr{Y}_0, \quad t \in \,]0, t_\mathrm{f}[. \tag{3}$$

3. Model reduction

Projection-based ROMs rely on the existence of a reduced dimensional subspace that approximates the solution space. Accordingly, there exists a space $\mathscr{Y}_r \subset \mathscr{Y}$ of dimension r that has a complete orthonormal basis $\{\boldsymbol{\phi}^k\}_{k=1}^r$, each vector being an array of n components. We can compute the best L^2-approximation of $\boldsymbol{y}$ in $\mathscr{Y}_r$ at a given time $t \in \,]0, t_\mathrm{f}[$ by $\sum_{k=1}^r (\boldsymbol{y}, \boldsymbol{\phi}^k)\boldsymbol{\phi}^k$.

In practice, the construction of the reduced space does not come from a continuous setting but rather as finding the best low-dimensional approximation of an ensemble of data, where the data can be obtained from experimental measurements or numerical solutions. The traditional way to construct a ROM – regardless of the preferred method to construct the space basis – is based on an offline–online stage approach, where in the offline stage a FOM is solved to gather the data needed to build the reduced space basis, and in the online stage the ROM is solved. From the different techniques that exist to compute a reduced basis from an ensemble of data (see e.g. [13–15]), following

some traditional works in model reduction (see e.g. [16–21]) and some more recent works in stabilized ROMs (as those in [7,22,23]), we choose the POD to construct the ROM basis.

Let us start noting that in our case the FOM for the continuous problem in Eq. (3) yields an ordinary differential equation of the form

$$\mathbf{M}\dot{\mathbf{y}} + \mathbf{K}\mathbf{y} = \mathbf{f}, \tag{4}$$

where $\mathbf{y}$ is a time dependent array of Nn components and $\dot{\mathbf{y}}$ denotes the time differentiation, $\mathbf{M}$ and $\mathbf{K}$ are the resulting mass and stiffness matrices and $\mathbf{f}$ is the resulting force vector. After discretizing in time as explained in Section 4, a collection of snapshots can be collected by time stepping. These snapshots are arrays of Nn components, and therefore the vectors of the ROM basis are also arrays of Nn components. In our FE context, however, we can identify them as piece-wise polynomial functions. Indeed, if $\boldsymbol{\phi}^{k,a}$ is the ath vector component of the kth basis vector, with $a = 1, \dots, N$ and $k = 1, \dots, r$, we may identify $\boldsymbol{\phi}^k$ with the function

$$\boldsymbol{\phi}^k(\boldsymbol{x}) = \sum_{a=1}^{N} \varphi^a(\boldsymbol{x})\boldsymbol{\phi}^{k,a}, \qquad k = 1, \dots, r, \quad \forall \boldsymbol{x} \in \Omega, \tag{5}$$

where $\varphi^a(\boldsymbol{x})$ is the FE interpolation function of the ath node. We shall indistinctly use $\boldsymbol{\phi}^k$ to denote the kth component of the POD basis understood as an array of Nn components or as a vector function $\boldsymbol{\phi}^k : \Omega \longrightarrow \mathbb{R}^n$.

Although it is not the goal of this work to analyse any specific method, it is important to mention some properties of the basis obtained applying the POD to a FE ensemble of data:

1. It is orthogonal with respect to the $\boldsymbol{\mathcal{M}}$-inner product given by

$$\langle \boldsymbol{\phi}^k, \boldsymbol{\phi}^l \rangle_{\mathcal{M}} = \int_\Omega \boldsymbol{\phi}^{k\top} \boldsymbol{\mathcal{M}} \boldsymbol{\phi}^l = \delta^{kl}, \quad k, l = 1, \dots, r, \tag{6a}$$

   and likewise, if $\boldsymbol{\phi}^k$ are ordered arrays

$$\boldsymbol{\phi}^{k\top} \mathbf{M} \boldsymbol{\phi}^l = \delta^{kl} \quad k, l = 1, \dots, r, \tag{6b}$$

   where $\mathbf{M}$ is the matrix in Eq. (4).
2. It satisfies homogeneous Dirichlet boundary conditions when calculated using a mean centered-trajectory method (see Remark 1).
3. It has an optimal approximation when truncated (Eckart–Young–Mirsky theorem).

In Appendix we show a short description on how to construct the ROM basis using POD; for a more extensive description see e.g. [24–26].

Remark 1. To implement the boundary conditions, we use the mean centered-trajectory method, and accordingly write the ROM unknown as

$$\boldsymbol{y}_r(\boldsymbol{x}, t) := \sum_{a=1}^{N} \varphi^a(\boldsymbol{x}) \left[\bar{\boldsymbol{y}} + \sum_{k=1}^{r} \boldsymbol{\phi}^{k,a} Y^k(t) \right],$$

where $\bar{\boldsymbol{y}}$ is the mean value of the FOM solution. Thus, the ROM unknowns $\{Y^k\}_{k=1}^r$ are in fact increments with respect to this mean value.

4. Variational FOM formulation

To formulate the discrete FOM for the variational problem in Eq. (3) we denote by $\mathscr{T}_h = \{K\}$ a partition of the domain Ω, with $h = \max\{h_K = \mathrm{diam}(K) | K \in \mathscr{T}_h\}$ the diameter of the partition, and by $\mathscr{Y}_h \subset \mathscr{Y}$ the corresponding FE approximation space. To simplify the notation, we will consider here homogeneous Dirichlet conditions, so that the space for the test functions is also $\mathscr{Y}$, and $\mathscr{Y}_h$ the finite element subspace.

For the time discretization we consider a uniform partition of the time interval $]0, t_f[$, with δt the time step size and j the superscript indicating the current time step $t^j = j\delta t$. For conciseness, we consider that temporal derivatives are approximated using a Backward Differentiation Formula (BDF) scheme, given by

$$\partial_t \boldsymbol{y} \,|_{t^j} \approx \delta_t \boldsymbol{y}^j := \sum_{l=0}^{s} \frac{\alpha^l \boldsymbol{y}^{j+1-l}}{\delta t}, \quad j = 1, 2, \dots, \tag{7}$$

where the operator $\delta_t \boldsymbol{y}^j$ indicates a discrete temporal derivatives over functions $\boldsymbol{y}$ at time t^j and the coefficients α^l depend on the order of approximation of the scheme s. In the following, the superscript j will be omitted if there is no possibility of confusion. When considering sequences of functions, j will be assumed to run from 0 to the total number of time steps, $t_\mathrm{f}/\delta t$.

The discrete Galerkin FE version of the problem in Eq. (3) is: find $\{\boldsymbol{y}_h^j\} \subset \mathscr{Y}_h$, such that

$$(\boldsymbol{\mathcal{M}}(\boldsymbol{y}_h)\delta_t \boldsymbol{y}_h, \boldsymbol{v}_h) + B\left(\boldsymbol{y}_h; \boldsymbol{y}_h, \boldsymbol{v}_h\right) = L(\boldsymbol{v}_h), \qquad \forall \boldsymbol{v}_h \in \mathscr{Y}_h, \quad \text{at } t^j, \ j = 1, 2, \ldots, \tag{8}$$

which we call the Galerkin form of the FOM.

Since the solution to the discrete problem in Eq. (8) may suffer from numerical instabilities that depend on the expression of the matrices that define the operator $\boldsymbol{\mathcal{L}}$, arising from the (possible) lack of coercivity of the problem and the compatibility restrictions between the components of $\boldsymbol{y}$, a stabilized formulation is needed. We can write such formulation for Eq. (8) in a general form as

$$(\boldsymbol{\mathcal{M}}(\hat{\boldsymbol{y}})\delta_t \boldsymbol{y}_h, \boldsymbol{v}_h) + B\left(\hat{\boldsymbol{y}}; \boldsymbol{y}_h, \boldsymbol{v}_h\right) + \langle \boldsymbol{\tau}_h(\hat{\boldsymbol{y}})\boldsymbol{\mathcal{R}}_h(\hat{\boldsymbol{y}}; \boldsymbol{y}_h), \boldsymbol{\mathcal{P}}_h(\hat{\boldsymbol{y}}; \boldsymbol{v}_h)\rangle = L(\boldsymbol{v}_h), \tag{9}$$

for all $\boldsymbol{v}_h \in \mathscr{Y}_h$, at t^j, $j = 1, 2, \ldots$, where $\hat{\boldsymbol{y}}$ is an approximation to $\boldsymbol{y}$ for which two possibilities are discussed in Remark 4. To simplify the notation, hereafter we shall write $\boldsymbol{y}$ instead of $\hat{\boldsymbol{y}}$.

Albeit several stabilization methods exist in the literature that could be used to stabilize the FOM without affecting the VMS-ROM stabilization proposed in this work, we chose to work within the same framework. We follow the survey on VMS-FE methods applied to computational fluid dynamics in [2] to formulate the stabilized FOM.

4.1. VMS for FOM

The VMS method consists in decomposing the space of the unknown into the finite-dimensional space – resulting from the finite element discretization – $\mathscr{Y}_h$, and a continuous one $\tilde{\mathscr{Y}}$, so that $\mathscr{Y} = \mathscr{Y}_h \oplus \tilde{\mathscr{Y}}$. The unknown and the test functions are accordingly split as $\boldsymbol{y} = \boldsymbol{y}_h + \tilde{\boldsymbol{y}}$ and $\boldsymbol{v} = \boldsymbol{v}_h + \tilde{\boldsymbol{v}}$, respectively. Then, once discretized in time the continuous problem in Eq. (3) expands into two: find $\{\boldsymbol{y}_h^j\} \subset \mathscr{Y}_h$ and $\{\tilde{\boldsymbol{y}}^j\} \subset \tilde{\mathscr{Y}}$, such that

$$\begin{aligned}
&(\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \boldsymbol{y}_h, \boldsymbol{v}_h) + (\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \tilde{\boldsymbol{y}}, \boldsymbol{v}_h) + B\left(\boldsymbol{y}; \boldsymbol{y}_h, \boldsymbol{v}_h\right) \\
&\quad + B\left(\boldsymbol{y}; \tilde{\boldsymbol{y}}, \boldsymbol{v}_h\right) = L(\boldsymbol{v}_h), \qquad \forall \boldsymbol{v}_h \in \mathscr{Y}_h, \quad \text{at } t^j, \ j = 1, 2, \ldots,
\end{aligned} \tag{10a}$$

$$\begin{aligned}
&(\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \boldsymbol{y}_h, \tilde{\boldsymbol{v}}) + (\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{v}}) + B\left(\boldsymbol{y}; \boldsymbol{y}_h, \tilde{\boldsymbol{v}}\right) \\
&\quad + B\left(\boldsymbol{y}; \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{v}}\right) = L(\tilde{\boldsymbol{v}}), \qquad \forall\ \tilde{\boldsymbol{v}} \in \tilde{\mathscr{Y}}, \quad \text{at } t^j, \ j = 1, 2, \ldots.
\end{aligned} \tag{10b}$$

This corresponds to take $\hat{\boldsymbol{y}} = \boldsymbol{y}$ in Eq. (9). We refer to functions in $\mathscr{Y}_h$ as the FOM *resolved scales* and to functions in $\tilde{\mathscr{Y}}$ as the FOM *SubGrid Scales (SGSs)* or simply *subscales*.

Since we want to avoid terms that involve applying the differential operator $\boldsymbol{\mathcal{L}}$ over the subscales $\tilde{\boldsymbol{y}}$, we re-write the form $B\left(\boldsymbol{y}; \tilde{\boldsymbol{y}}, \boldsymbol{v}_h\right)$ in Eq. (10a) by integrating by parts. That leads to a formal adjoint of the operator $\boldsymbol{\mathcal{L}}(\boldsymbol{y}; \cdot)$, defined as $\boldsymbol{\mathcal{L}}^*(\boldsymbol{y}; \boldsymbol{v}_h) := -\partial_i\left(\boldsymbol{K}_{ji}^\top(\boldsymbol{y})\partial_j \boldsymbol{v}_h\right) - \partial_i \boldsymbol{A}_i^c(\boldsymbol{y})^\top \boldsymbol{v}_h - \partial_i \boldsymbol{A}_i^f(\boldsymbol{y})^\top \boldsymbol{v}_h + \boldsymbol{S}(\boldsymbol{y})^\top \boldsymbol{v}_h$ and a flux function on the boundary $\boldsymbol{\mathcal{F}}^*(\boldsymbol{y}; \boldsymbol{v}_h) := \boldsymbol{n}_i \boldsymbol{K}_{ji}^\top(\boldsymbol{y})\partial_j \boldsymbol{v}_h + n_i \boldsymbol{A}_i^{c\top}(\boldsymbol{y})\boldsymbol{v}_h$. Moreover, we define the residual of the resolved scales as $\boldsymbol{r}(\boldsymbol{y}; \boldsymbol{y}_h) := \boldsymbol{f} - \boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \boldsymbol{y}_h - \boldsymbol{\mathcal{L}}(\boldsymbol{y}; \boldsymbol{y}_h)$ at each time level t^j, and integrate by parts the forms $B\left(\boldsymbol{y}; \boldsymbol{y}_h, \tilde{\boldsymbol{v}}\right)$ and $B\left(\boldsymbol{y}; \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{v}}\right)$ in Eq. (10b). In this way, we can re-write Eq. (10) as

$$\begin{aligned}
&(\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \boldsymbol{y}_h, \boldsymbol{v}_h) + (\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \tilde{\boldsymbol{y}}, \boldsymbol{v}_h) + B\left(\boldsymbol{y}; \boldsymbol{y}_h, \boldsymbol{v}_h\right) + \sum_K \langle \tilde{\boldsymbol{y}}, \boldsymbol{\mathcal{L}}^*(\boldsymbol{y}; \boldsymbol{v}_h)\rangle_K \\
&\qquad + \sum_K \langle \tilde{\boldsymbol{y}}, \boldsymbol{\mathcal{F}}^*(\boldsymbol{y}; \boldsymbol{v}_h)\rangle_{\partial K} = L(\boldsymbol{v}_h), \qquad \forall \boldsymbol{v}_h \in \mathscr{Y}_h,
\end{aligned} \tag{11a}$$

$$\begin{aligned}
&\sum_K \langle \boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{v}}\rangle_K + \sum_K \langle \boldsymbol{\mathcal{L}}(\boldsymbol{y}, \tilde{\boldsymbol{y}}), \tilde{\boldsymbol{v}}\rangle_K + \sum_K \langle \boldsymbol{\mathcal{F}}(\boldsymbol{y}; \tilde{\boldsymbol{y}}), \tilde{\boldsymbol{v}}\rangle_{\partial K} \\
&+ \sum_K \langle \boldsymbol{\mathcal{F}}(\boldsymbol{y}; \boldsymbol{y}_h), \tilde{v}\rangle_{\partial K} = \sum_K \langle \boldsymbol{r}(\boldsymbol{y}; \boldsymbol{y}_h), \tilde{\boldsymbol{v}}\rangle_K, \qquad \forall \tilde{\boldsymbol{v}} \in \tilde{\mathscr{Y}},
\end{aligned} \tag{11b}$$

at t^j, $j = 1, 2, \ldots$, with $\langle \cdot, \cdot \rangle_K$ and $\langle \cdot, \cdot \rangle_{\partial K}$ the L^2 inner product over an element $K \in \mathscr{T}_h$ and its boundary, respectively.

4.2. FOM subscales

Considering that the subscale problem in Eq. (11b) is infinite dimensional, the approximation $\mathcal{L}(\boldsymbol{y};\tilde{\boldsymbol{y}}) = \boldsymbol{\tau}_K^{-1}(\boldsymbol{y})\tilde{\boldsymbol{y}}$ in each element K needs to be introduced to make the method computationally feasible, where $\boldsymbol{\tau}_K$ is the matrix of stabilization parameters that approximates the inverse of the differential operator $\mathcal{L}$ on each element K. The definition of $\boldsymbol{\tau}_K$ can be achieved following different methods which lead to slightly different sets of parameters. In this work we follow the Fourier analysis proposed in [9]. Furthermore, assuming that the normal fluxes of the total unknown are continuous across the inter-element boundaries, the boundary terms in Eq. (11) vanish and we arrive to

$$\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t\tilde{\boldsymbol{y}} + \boldsymbol{\tau}_K^{-1}(\boldsymbol{y})\tilde{\boldsymbol{y}} = \boldsymbol{r}(\boldsymbol{y};\boldsymbol{y}_h) + \tilde{\boldsymbol{v}}^{\perp}, \qquad \text{in } K \in \mathscr{T}_h, \quad \text{at } t^j, \ j = 1,2,\ldots, \tag{12}$$

where $\tilde{\boldsymbol{v}}^{\perp}$ is any function that satisfies the condition

$$\sum_K \langle \tilde{\boldsymbol{v}}^{\perp}, \tilde{\boldsymbol{v}}\rangle_K = 0, \quad \forall \tilde{\boldsymbol{v}} \in \tilde{\mathscr{V}}, \tag{13}$$

defined by the choice of the space of subscales $\tilde{\mathscr{V}}$ and its orthogonal complement $\tilde{\mathscr{V}}^{\perp}$. By defining $\tilde{\boldsymbol{\Pi}}^{\perp}$ the L^2 projection onto $\tilde{\mathscr{V}}^{\perp}$ we can impose the condition in Eq. (13). Projecting Eq. (12) onto $\tilde{\mathscr{V}}^{\perp}$ and using Eq. (13), we get $\tilde{\boldsymbol{v}}^{\perp} = -\tilde{\boldsymbol{\Pi}}^{\perp}(\boldsymbol{r}(\boldsymbol{y};\boldsymbol{y}_h))$ and therefore

$$\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t\tilde{\boldsymbol{y}} + \boldsymbol{\tau}_K^{-1}(\boldsymbol{y})\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{\Pi}}(\boldsymbol{r}(\boldsymbol{y};\boldsymbol{y}_h)) =: \boldsymbol{\mathcal{R}}_h(\boldsymbol{y};\boldsymbol{y}_h), \qquad \text{in } K \in \mathscr{T}_h, \quad \text{at } t^j, \ j = 1,2,\ldots, \tag{14}$$

where $\tilde{\boldsymbol{\Pi}} = \boldsymbol{I} - \tilde{\boldsymbol{\Pi}}^{\perp}$ is the projection onto the subscale space $\tilde{\mathscr{V}}$ and $\boldsymbol{I}$ is the identity.

Remark 2. Another way to impose this condition is by using the weighted inner product $(\cdot,\cdot)_\tau = \sum_K \langle \boldsymbol{\tau}_K \cdot, \cdot\rangle_K$ and the associated projection $\tilde{\boldsymbol{\Pi}}_\tau^{\perp}$ onto $\tilde{\mathscr{V}}^{\perp}$. This way, projecting Eq. (12) onto $\tilde{\mathscr{V}}^{\perp}$ and using Eq. (13), leads to $\tilde{\boldsymbol{v}}^{\perp} = -\tilde{\boldsymbol{\Pi}}_\tau^{\perp}(\boldsymbol{r}(\boldsymbol{y};\boldsymbol{y}_h))$. Further details on this approximation can be found in [6].

Algebraic and orthogonal subscales. Depending on the choice of the projection $\tilde{\boldsymbol{\Pi}}$ — which involves the choice of the space $\tilde{\mathscr{V}}$, we get different approximation methods for the subscales. In [9] two alternatives for formulating the subscales are described: an Algebraic SubGrid Scales (ASGSs) formulation that consists in taking $\tilde{\boldsymbol{v}}^{\perp} = \boldsymbol{0}$ and therefore $\tilde{\boldsymbol{\Pi}} = \boldsymbol{I}$, giving $\boldsymbol{\mathcal{R}}_h(\boldsymbol{y};\boldsymbol{y}_h) = \boldsymbol{r}(\boldsymbol{y};\boldsymbol{y}_h)$, and an OSGSs formulation that consists in defining the space of subscales as orthogonal to the complement of $\mathscr{V}_h$, having $\tilde{\mathscr{V}} = \mathscr{V}_h^{\perp}$. This implies satisfying the condition $\sum_K \langle \boldsymbol{v}_h, \tilde{\boldsymbol{v}}\rangle = 0$ and therefore $\tilde{\boldsymbol{\Pi}} := \boldsymbol{\Pi}_h^{\perp} = \boldsymbol{I} - \boldsymbol{\Pi}_h$, where $\boldsymbol{\Pi}_h$ is the projection onto the FE space, giving $\boldsymbol{\mathcal{R}}_h(\boldsymbol{y};\boldsymbol{y}_h) = \boldsymbol{\Pi}_h^{\perp}(\boldsymbol{f} - \boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h))$.

Quasi-static and dynamic subscales. Following [5] we add two other choices to the way we formulate the subscales: quasi-static subscales, obtained by neglecting the time derivative of the subscales – i.e. $\partial_t\tilde{\boldsymbol{y}} = \boldsymbol{0}$ – and yielding the subscale approximation $\boldsymbol{\tau}_K^{-1}(\boldsymbol{y})\tilde{\boldsymbol{y}} = \boldsymbol{\mathcal{R}}_h(\boldsymbol{y};\boldsymbol{y}_h)$, and dynamic subscales, obtained when such time derivative is considered, approximating the subscales as the solution of $\boldsymbol{\mathcal{M}}\delta_t\tilde{\boldsymbol{y}} + \boldsymbol{\tau}_K^{-1}(\boldsymbol{y})\tilde{\boldsymbol{y}} = \boldsymbol{\mathcal{R}}_h(\boldsymbol{y};\boldsymbol{y}_h)$. Here, an effective stabilization parameter can be defined as $\boldsymbol{\tau}_{K,t}^{-1}(\boldsymbol{y}) := \alpha^0\delta t^{-1}\boldsymbol{\mathcal{M}} + \boldsymbol{\tau}_K^{-1}(\boldsymbol{y})$.

Remark 3. As it is discussed in [5], the chosen time integration scheme for the subscales can be less accurate than for the FE equations without affecting the accuracy of the numerical scheme. Hence, we chose a first order BDF to approximate the time derivative in the SGSs equations when using a second order scheme for the FE equations.

By defining $\mathcal{P}_h(\boldsymbol{y},\boldsymbol{v}_h) = \mathcal{L}^*(\boldsymbol{y},\boldsymbol{v}_h)$, and choosing between algebraic and orthogonal, and quasi-static and dynamic SGSs, we can obtain a general stabilized approximation for $\boldsymbol{y}_h$ of the form of Eq. (9), in principle with $\hat{\boldsymbol{y}} = \boldsymbol{y}_h + \tilde{\boldsymbol{y}}$ (written as $\boldsymbol{y}$). Moreover, we can rewrite this equation as

$$(\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t\boldsymbol{y}_h, \boldsymbol{v}_h) + B_h\left(\boldsymbol{y};\boldsymbol{y}_h,\boldsymbol{v}_h\right) = L_h(\boldsymbol{y};\boldsymbol{v}_h), \quad \forall \boldsymbol{v}_h \in \mathscr{V}_h, \quad \text{at } t^j, \ j = 1,2,\ldots, \tag{15}$$

with the forms $B_h\left(\boldsymbol{y};\boldsymbol{y}_h,\boldsymbol{v}_h\right) = B\left(\boldsymbol{y};\boldsymbol{y}_h,\boldsymbol{v}_h\right) + B_s\left(\boldsymbol{y};\boldsymbol{y}_h,\boldsymbol{v}_h\right)$ and $L_h(\boldsymbol{y};\boldsymbol{v}_h) = L(\boldsymbol{v}_h) + L_s(\boldsymbol{y};\boldsymbol{v}_h)$, where the stabilization terms $B_s\left(\boldsymbol{y};\boldsymbol{y}_h,\boldsymbol{v}_h\right)$ and $L_s(\boldsymbol{y};\boldsymbol{v}_h)$ are defined for each proposed case in Tables 1 and 2, respectively. Considering the results in [9] and in [5,27], where the subscale approximation is shown to influence convergence and stability, we choose to formulate the FE-FOM using dynamic OSGSs.

Table 1
FOM $B_s\left(\boldsymbol{y};\boldsymbol{y}_h,\boldsymbol{v}_h\right)$ defined for dynamic and quasi-static ASGSs and OSGSs.

	ASGS
Dynamic	$-\sum_K\langle\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t\boldsymbol{y}_h+\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h),\boldsymbol{\tau}_{K,t}\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_h)\rangle_K$
	$-\sum_K\langle\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t\boldsymbol{y}_h+\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h),[\boldsymbol{I}-\boldsymbol{\tau}_K^{-1}\boldsymbol{\tau}_{K,t}]\boldsymbol{v}_h\rangle_K$
Quasi-static	$-\sum_K\langle\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t\boldsymbol{y}_h+\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h),\boldsymbol{\tau}_K\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_h)\rangle_K$
	OSGS
Dynamic	$-\sum_K\langle\boldsymbol{\Pi}_h^{\perp}(\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h)),\boldsymbol{\tau}_{K,t}\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_h)\rangle_K$
Quasi-static	$-\sum_K\langle\boldsymbol{\Pi}_h^{\perp}(\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h)),\boldsymbol{\tau}_K\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_h)\rangle_K$

Table 2
FOM $L_s(\boldsymbol{y};\boldsymbol{v}_h)$ defined for dynamic and quasi-static ASGSs and OSGSs.

	ASGS
Dynamic	$-\sum_K\langle\boldsymbol{f},\boldsymbol{\tau}_{K,t}\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_h)+[\boldsymbol{I}-\boldsymbol{\tau}_K^{-1}\boldsymbol{\tau}_{K,t}]\boldsymbol{v}_h\rangle_K$
	$-\sum_K\langle\delta t^{-1}\boldsymbol{\mathcal{M}}(\boldsymbol{y})\tilde{\boldsymbol{y}}^j,\boldsymbol{\tau}_{K,t}\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_h)-\boldsymbol{\tau}_K^{-1}\boldsymbol{\tau}_{K,t}\boldsymbol{v}_h\rangle_K$
Quasi-static	$-\sum_K\langle\boldsymbol{f},\boldsymbol{\tau}_K\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_h)\rangle_K$
	OSGS
Dynamic	$-\sum_K\langle\boldsymbol{\Pi}_h^{\perp}(\boldsymbol{f})+\delta t^{-1}\boldsymbol{\mathcal{M}}(\boldsymbol{y})\tilde{\boldsymbol{y}}^j,\boldsymbol{\tau}_{K,t}\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_h)\rangle_K$
Quasi-static	$-\sum_K\langle\boldsymbol{\Pi}_h^{\perp}(\boldsymbol{f}),\boldsymbol{\tau}_K\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_h)\rangle_K$

Remark 4. The treatment of the term $\boldsymbol{y}$ in the operators $\boldsymbol{\mathcal{M}}(\boldsymbol{y})$, $\boldsymbol{\mathcal{L}}(\boldsymbol{y},\cdot)$, $\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\cdot)$, $\boldsymbol{\mathcal{F}}(\boldsymbol{y},\cdot)$ and $\boldsymbol{\mathcal{F}}^*(\boldsymbol{y},\cdot)$ prompts another alternative: linear subscales —in which $\boldsymbol{y}\approx\boldsymbol{y}_h$— and nonlinear subscales — in which $\boldsymbol{y}\approx\boldsymbol{y}_h+\tilde{\boldsymbol{y}}$. The terms 'linear' and 'nonlinear' refer to the appearance of the subscales only in the linear terms or also in the nonlinear ones. In this work we consider only linear subscales along with a Picard scheme for the FOM resolved scale non-linearities in Eq. (10a). See [9] for a detailed description of the nonlinear subscales.

Remark 5. The practical way to compute the projection $\boldsymbol{\Pi}_h^{\top}$ is to compute the projection $\boldsymbol{\Pi}_h$ and then use $\boldsymbol{\Pi}_h^{\top}=\boldsymbol{I}-\boldsymbol{\Pi}_h$. Therefore, by denoting as $z_h=\boldsymbol{\Pi}_h(\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h)-\boldsymbol{f})]$, the problem in Eq. (15) – for dynamic OSGSs – becomes: find $[\{\boldsymbol{y}_h^j\},\{z_h^j\}]$ in $\mathscr{Y}_h\times\mathscr{Y}_h$, at each time t^j, such that

$$\begin{aligned}(\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t\boldsymbol{y}_h,\boldsymbol{v}_h)+B\left(\boldsymbol{y};\boldsymbol{y}_h,\boldsymbol{v}_h\right)-\sum_K\langle\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h)-z_h,\boldsymbol{\tau}_{K,t}(\boldsymbol{y})\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_h)\rangle_K\\ =L(\boldsymbol{y};\boldsymbol{v}_h)-\sum_K\langle\boldsymbol{f}+\delta t^{-1}\boldsymbol{\mathcal{M}}(\boldsymbol{y})\tilde{\boldsymbol{y}}^n,\boldsymbol{\tau}_{K,t}(\boldsymbol{y})\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_h)\rangle_K,\end{aligned}\tag{16a}$$

$$(z_h,\boldsymbol{\zeta}_h)=\sum_K\langle\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h)-\boldsymbol{f},\boldsymbol{\zeta}_h\rangle_K,\tag{16b}$$

which must hold for all $[\boldsymbol{v}_h,\boldsymbol{\zeta}_h]$ in $\mathscr{Y}_{h,0}\times\mathscr{Y}_h$. To approximate Eq. (16b) we follow the block iteration algorithm in [6], having $[\boldsymbol{\Pi}_h^{\top}(\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h))]^{i+1}=[\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h)]^{i+1}-[\boldsymbol{\Pi}_h(\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h))]^i$, where the superscript i denotes the iteration counter.

5. Variational ROM formulation

To formulate the variational projection-based ROM problem —analogously to the FOM in Section 4— we denote the approximation space as $\mathscr{Y}_r\subset\mathscr{Y}$. We can write the variational ROM of the problem in Eq. (3) as: find $\{\boldsymbol{y}_r^j\}\subset\mathscr{Y}_r$, such that

$$(\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t\boldsymbol{y}_r,\boldsymbol{v}_r)+B\left(\boldsymbol{y};\boldsymbol{y}_r,\boldsymbol{v}_r\right)=L(\boldsymbol{v}_r),\qquad\forall\boldsymbol{v}_r\in\mathscr{Y}_r,\quad\text{at } t^j,\ j=1,2,\ldots,\tag{17}$$

which we describe as the Galerkin-ROM.

As the ROM approximation space is a subset of the FOM space —$\mathscr{Y}_r\subset\mathscr{Y}_h$— the instabilities described in Section 4 are inherited. Hence, a stabilization for the ROM is also necessary. By implementing the same VMS

method in Eq. (17) we can write a stabilized ROM in the form of Eq. (9) as

$$(\boldsymbol{\mathcal{M}}(\hat{\boldsymbol{y}})\delta_t \boldsymbol{y}_r, \boldsymbol{v}_r) + B\left(\hat{\boldsymbol{y}}; \boldsymbol{y}_r, \boldsymbol{v}_r\right) + \langle \boldsymbol{\tau}_r(\hat{\boldsymbol{y}})\boldsymbol{\mathcal{R}}_r(\hat{\boldsymbol{y}}; \boldsymbol{y}_r), \boldsymbol{\mathcal{P}}_r(\hat{\boldsymbol{y}}; \boldsymbol{v}_r)\rangle = L(\boldsymbol{v}_r), \tag{18}$$

for all $\boldsymbol{v}_r \in \mathscr{V}_r$, where, similarly to the FOM case, $\hat{\boldsymbol{y}}$ is an approximation to $\boldsymbol{y}$ that may take into account the subscales (see below) or not. We will simply write $\boldsymbol{y}$ in what follows.

5.1. VMS for ROM

To formulate the VMS for the ROM we follow the same procedure as for the FOM until the approximation of the subscales. Nevertheless, we describe the needed steps in somewhat a detailed way.

The unknown space is also decomposed in two: the approximation space of the ROM resolved scales $\mathscr{Y}_r$ and the space of the ROM-SGSs $\breve{\mathscr{Y}}$, so that $\mathscr{Y} = \mathscr{Y}_r \oplus \breve{\mathscr{Y}}$, with the unknown and the test functions split as $\boldsymbol{y} = \boldsymbol{y}_r + \breve{\boldsymbol{y}}$ and $\boldsymbol{v} = \boldsymbol{v}_r + \breve{\boldsymbol{v}}$, respectively. Consequently, the ROM problem in Eq. (17) also expands into two, finding $\{\boldsymbol{y}_r^j\} \subset \mathscr{Y}_r$ and $\{\breve{\boldsymbol{y}}^j\} \subset \breve{\mathscr{Y}}$, such that

$$\begin{aligned}
&(\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \boldsymbol{y}_r, \boldsymbol{v}_r) + (\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \breve{\boldsymbol{y}}, \boldsymbol{v}_r) + B\left(\boldsymbol{y}; \boldsymbol{y}_r, \boldsymbol{v}_r\right) \\
&\qquad + B\left(\boldsymbol{y}; \breve{\boldsymbol{y}}, \boldsymbol{v}_r\right) = L(\boldsymbol{v}_r), \qquad \forall \boldsymbol{v}_r \in \mathscr{V}_r, \quad \text{at } t^j, \ j = 1, 2, \ldots, \qquad (19a)\\
&(\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \boldsymbol{y}_r, \breve{\boldsymbol{v}}) + (\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \breve{\boldsymbol{y}}, \breve{\boldsymbol{v}}) + B\left(\boldsymbol{y}; \boldsymbol{y}_r, \breve{\boldsymbol{v}}\right) \\
&\qquad + B\left(\boldsymbol{y}; \breve{\boldsymbol{y}}, \breve{\boldsymbol{v}}\right) = L(\breve{\boldsymbol{v}}), \qquad \forall\, \breve{\boldsymbol{v}} \in \breve{\mathscr{Y}}, \quad \text{at } t^j, \ j = 1, 2, \ldots. \qquad (19b)
\end{aligned}$$

To avoid terms that involve applying the differential operator $\boldsymbol{\mathcal{L}}$ over the subscales $\breve{\boldsymbol{y}}$, we follow the same approach as in Section 4.1, yielding

$$\begin{aligned}
&(\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \boldsymbol{y}_r, \boldsymbol{v}_r) + (\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \breve{\boldsymbol{y}}, \boldsymbol{v}_r) + B\left(\boldsymbol{y}; \boldsymbol{y}_r, \boldsymbol{v}_r\right) + \sum_K \langle \breve{\boldsymbol{y}}, \boldsymbol{\mathcal{L}}^*(\boldsymbol{y}; \boldsymbol{v}_r)\rangle_K \\
&\qquad + \sum_K \langle \tilde{\boldsymbol{y}}, \boldsymbol{\mathcal{F}}^*(\boldsymbol{y}; \boldsymbol{v}_r)\rangle_{\partial K} = L(\boldsymbol{v}_r), \qquad \forall \boldsymbol{v}_r \in \mathscr{V}_r, \quad \text{at } t^j, \ j = 1, 2, \ldots, \qquad (20a)\\
&\sum_K \langle \boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \breve{\boldsymbol{y}}, \breve{\boldsymbol{v}}\rangle_K + \sum_K \langle \boldsymbol{\mathcal{L}}(\boldsymbol{y}, \breve{\boldsymbol{y}}), \breve{\boldsymbol{v}}\rangle_K + \sum_K \langle \boldsymbol{\mathcal{F}}(\boldsymbol{y}; \breve{\boldsymbol{y}}), \breve{\boldsymbol{v}}\rangle_{\partial K} \\
&\qquad + \sum_K \langle \boldsymbol{\mathcal{F}}(\boldsymbol{y}; \boldsymbol{y}_r), \breve{\upsilon}\rangle_{\partial K} = \sum_K \langle \boldsymbol{r}(\boldsymbol{y}; \boldsymbol{y}_r), \breve{\boldsymbol{v}}\rangle_K, \qquad \forall \breve{\boldsymbol{v}} \in \breve{\mathscr{Y}}, \qquad \text{at } t^j, \ j = 1, 2, \ldots. \qquad (20b)
\end{aligned}$$

These equations have the same structure as Eqs. (11a) and (11b), just replacing the FE unknown by the ROM one and the FE subscale by the ROM subscale. It is important to note that we consider $\boldsymbol{y}_r$ to belong to a subspace of the FE space, and therefore the summation over the elements and the evaluation in the elements above make full sense.

5.2. ROM subscales

By following the same supposition of continuous fluxes and using the same approximation for the spatial operator $-\boldsymbol{\mathcal{L}}(\boldsymbol{y}, \breve{\boldsymbol{y}}) \approx \boldsymbol{\tau}_K^{-1}(\boldsymbol{y})\breve{\boldsymbol{y}}$ in each element K – as in Section 4.2, we can write an equation similar to Eq. (12) for the ROM-SGSs:

$$\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \breve{\boldsymbol{y}} + \boldsymbol{\tau}_K^{-1}(\boldsymbol{y})\breve{\boldsymbol{y}} = \boldsymbol{r}(\boldsymbol{y}; \boldsymbol{y}_r) + \breve{\boldsymbol{v}}^{\perp}, \qquad \text{in } K \in \mathscr{T}_h, \quad \text{at } t^j, \ j = 1, 2, \ldots. \tag{21}$$

Likewise, to complete the approximation of the subscales, the term $\breve{\boldsymbol{v}}^{\perp}$ satisfies a similar condition as in Eq. (13) — $\sum_K \langle \breve{\boldsymbol{v}}^{\perp}, \breve{\boldsymbol{v}}\rangle_K = 0$, $\forall \breve{\boldsymbol{v}} \in \breve{\mathscr{Y}}$, and therefore also defined by the choice of the space of ROM subscales $\breve{\mathscr{Y}}$ and its orthogonal complement $\breve{\mathscr{Y}}^{\perp}$. Thus, denoting the L^2 projection onto $\breve{\mathscr{Y}}^{\perp}$ as $\breve{\boldsymbol{\Pi}}^{\perp}$ and defining $\breve{\boldsymbol{v}}^{\perp} = -\breve{\boldsymbol{\Pi}}^{\perp}(\boldsymbol{r}(\boldsymbol{y}; \boldsymbol{y}_r))$, we can rewrite Eq. (21) as

$$\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t \breve{\boldsymbol{y}} + \boldsymbol{\tau}_K^{-1}(\boldsymbol{y})\breve{\boldsymbol{y}} = \breve{\boldsymbol{\Pi}}(\boldsymbol{r}(\boldsymbol{y}; \boldsymbol{y}_r)) =: \boldsymbol{\mathcal{R}}_r(\boldsymbol{y}; \boldsymbol{y}_r), \qquad \text{in } K \in \mathscr{T}_h, \quad \text{at } t^j, \ j = 1, 2, \ldots, \tag{22}$$

where similarly to Eq. (14) for the FOM-SGSs, $\breve{\boldsymbol{\Pi}} = \boldsymbol{I} - \breve{\boldsymbol{\Pi}}^{\perp}$ is the projection onto the subscale space $\breve{\mathscr{Y}}$ and $\boldsymbol{I}$ is the identity.

Table 3
ROM $B_s\left(\boldsymbol{y};\boldsymbol{y}_r,\boldsymbol{v}_r\right)$ defined for dynamic and quasi-static ASGSs and OSGSs.

	ASGS
Dynamic	$-\sum_K\langle\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t\boldsymbol{y}_r+\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_r),\boldsymbol{\tau}_{K,t}\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_r)\rangle_K$ $-\sum_K\langle\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t\boldsymbol{y}_r+\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_r),[\boldsymbol{I}-\boldsymbol{\tau}_K^{-1}\boldsymbol{\tau}_{K,t}]\boldsymbol{v}_r\rangle_K$
Quasi-static	$-\sum_K\langle\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t\boldsymbol{y}_r+\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_r),\boldsymbol{\tau}_K\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_r)\rangle_K$
	OSGS
Dynamic	$-\sum_K\langle\boldsymbol{\Pi}_r^{\perp}(\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_r)),\boldsymbol{\tau}_{K,t}\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_r)\rangle_K$
Quasi-static	$-\sum_K\langle\boldsymbol{\Pi}_r^{\perp}(\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_r)),\boldsymbol{\tau}_K\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_r)\rangle_K$

Table 4
ROM $L_s(\boldsymbol{y};\boldsymbol{v}_r)$ defined for dynamic and quasi-static ASGSs and OSGSs.

	ASGS
Dynamic	$-\sum_K\langle\boldsymbol{f},\boldsymbol{\tau}_{K,t}\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_r)+[\boldsymbol{I}-\boldsymbol{\tau}_K^{-1}\boldsymbol{\tau}_{K,t}]\boldsymbol{v}_r\rangle_K$ $-\sum_K\langle\delta t^{-1}\boldsymbol{\mathcal{M}}(\boldsymbol{y})\breve{\boldsymbol{y}}^j,\boldsymbol{\tau}_{K,t}\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_r)-\boldsymbol{\tau}_K^{-1}\boldsymbol{\tau}_{K,t}\boldsymbol{v}_r\rangle_K$
Quasi-static	$-\sum_K\langle\boldsymbol{f},\boldsymbol{\tau}_K\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_r)\rangle_K$
	OSGS
Dynamic	$-\sum_K\langle\boldsymbol{\Pi}_r^{\perp}(\boldsymbol{f})+\delta t^{-1}\boldsymbol{\mathcal{M}}(\boldsymbol{y})\breve{\boldsymbol{y}}^j,\boldsymbol{\tau}_{K,t}\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_r)\rangle_K$
Quasi-static	$-\sum_K\langle\boldsymbol{\Pi}_r^{\perp}(\boldsymbol{f}),\boldsymbol{\tau}_K\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_r)\rangle_K$

Algebraic and orthogonal subscales. Analogously to the FOM subscales, for the choice of the space $\breve{\mathscr{Y}}$ and consequently for the definition of the projection $\breve{\boldsymbol{\Pi}}$, we get the same two different approximation methods for the subscales: ROM-ASGSs and ROM-OSGSs. For the ASGSs formulation $\breve{\boldsymbol{v}}^{\perp}=\boldsymbol{0}$ and $\breve{\boldsymbol{\Pi}}=\boldsymbol{I}$, yielding $\boldsymbol{\mathcal{R}}_r(\boldsymbol{y};\boldsymbol{y}_r)=\boldsymbol{r}(\boldsymbol{y};\boldsymbol{y}_r)$, while for the OSGSs formulation $\breve{\mathscr{Y}}=\mathscr{Y}_r^{\perp}$, $\sum_K\langle\boldsymbol{v}_r,\breve{\boldsymbol{v}}\rangle=\boldsymbol{0}$ and $\breve{\boldsymbol{\Pi}}:=\boldsymbol{\Pi}_r^{\perp}=\boldsymbol{I}-\boldsymbol{\Pi}_r$, yielding $\boldsymbol{\mathcal{R}}_r(\boldsymbol{y};\boldsymbol{y}_r)=\boldsymbol{\Pi}_r^{\perp}(\boldsymbol{f}-\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_r))$.

Quasi-static and dynamic subscales. As for the FOM, quasi-static —$\partial_t\breve{\boldsymbol{y}}=\boldsymbol{0}$— and dynamic subscales —$\partial_t\breve{\boldsymbol{y}}$ not negligible— can be considered in the ROM-SGSs, yielding the approximations $\boldsymbol{\tau}_K^{-1}(\boldsymbol{y})\tilde{\boldsymbol{y}}=\boldsymbol{\mathcal{R}}_r(\boldsymbol{y};\boldsymbol{y}_r)$ and $\boldsymbol{\mathcal{M}}\delta_t\breve{\boldsymbol{y}}+\boldsymbol{\tau}_K^{-1}(\boldsymbol{y})\tilde{\boldsymbol{y}}=\boldsymbol{\mathcal{R}}_r(\boldsymbol{y};\boldsymbol{y}_r)$, respectively.

By defining $\boldsymbol{\mathcal{P}}_r(\boldsymbol{y},\boldsymbol{v}_r)=\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{v}_r)$, and choosing $\boldsymbol{\tau}_r(\boldsymbol{y})$ as $\boldsymbol{\tau}_K(\boldsymbol{y})$ or $\boldsymbol{\tau}_{K,t}(\boldsymbol{y})$, and $\boldsymbol{\mathcal{R}}_r(\boldsymbol{y};\boldsymbol{y}_r)$ as $\boldsymbol{r}(\boldsymbol{y};\boldsymbol{y}_r)$ or $\boldsymbol{\Pi}_r^{\perp}(\boldsymbol{f}-\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_r))$, we can obtain a general stabilized approximation for $\boldsymbol{y}_r$ of the form of Eq. (18). Moreover, we can rewrite this equation as

$$(\boldsymbol{\mathcal{M}}(\boldsymbol{y})\delta_t\boldsymbol{y}_r,\boldsymbol{v}_r)+B_r\left(\boldsymbol{y};\boldsymbol{y}_r,\boldsymbol{v}_r\right)=L_r(\boldsymbol{v}_r),\qquad\forall\boldsymbol{v}_r\in\mathscr{Y}_r,\quad\text{at } t^j,\ j=1,2,\ldots. \tag{23}$$

with the forms $B_r\left(\boldsymbol{y};\boldsymbol{y}_r,\boldsymbol{v}_r\right)=B\left(\boldsymbol{y};\boldsymbol{y}_r,\boldsymbol{v}_r\right)+B_s\left(\boldsymbol{y};\boldsymbol{y}_r,\boldsymbol{v}_r\right)$ and $L_r(\boldsymbol{y};\boldsymbol{v}_r)=L(\boldsymbol{v}_r)+L_s(\boldsymbol{y};\boldsymbol{v}_r)$, where, using a first order BDF to approximate the time derivative of the SGSs, the stabilization terms $B_s\left(\boldsymbol{y};\boldsymbol{y}_r,\boldsymbol{v}_r\right)$ and $L_s(\boldsymbol{y};\boldsymbol{v}_r)$ are defined for each proposed case in Tables 3 and 4, respectively. The problem has to be completed with initial conditions for the subscales. We assume that $\breve{\boldsymbol{y}}^0=\boldsymbol{0}$, for all $\boldsymbol{x}\in\Omega$, which means that we assume for simplicity that the initial condition belongs to the space of resolvable scales.

We compute the term $\boldsymbol{\Pi}_r^{\top}(\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_r)-\boldsymbol{f})$ in a way similar to the FOM term $\boldsymbol{\Pi}_h^{\top}(\boldsymbol{\mathcal{L}}(\boldsymbol{y};\boldsymbol{y}_h)-\boldsymbol{f})$, as explained in Remark 5. It is important to notice that so far, the approximations for both FOM and ROM look exactly the same but for the space of variables for each of them —$\mathscr{Y}_h$ for the FOM and $\mathscr{Y}_r$ for the ROM— and consequently for the space of the subscales.

Remark 6. In this work, the non-linearity in the operators $\boldsymbol{\mathcal{M}}(\boldsymbol{y})$, $\boldsymbol{\mathcal{L}}(\boldsymbol{y},\boldsymbol{y}_r)$, $\boldsymbol{\mathcal{L}}^*(\boldsymbol{y},\boldsymbol{y}_r)$, $\boldsymbol{\mathcal{F}}(\boldsymbol{y},\boldsymbol{y}_r)$ and $\boldsymbol{\mathcal{F}}^*(\boldsymbol{y},\boldsymbol{y}_r)$ is treated using $\boldsymbol{y}=\boldsymbol{y}_r$, which corresponds to the linear subscale choice of the FOM case.

Remark 7 (*On the Importance of OSGSs*)**.** An interesting feature of *orthogonal* SGSs, that renders them particularly 'natural', is the following. Suppose that we have an orthonormal basis of $\mathscr{Y}_h$ given by $\{\boldsymbol{\phi}^k\}_{k=1}^{Nn}$, so that $\mathscr{Y}_r=\text{span}\{\boldsymbol{\phi}^k\}_{k=1}^{r}$. When OSGSs are used, we have the following explicit representation of the ROM-SGSs:

$$\breve{\mathscr{Y}}=\text{span}\{\boldsymbol{\phi}^k\}_{k=r+1}^{Nn}\oplus\tilde{\mathscr{Y}}.$$

5.3. Behavior of the stabilization parameters

The last component of the stabilized ROM formulation is the definition of the stabilization parameter $\boldsymbol{\tau}_K(\boldsymbol{y})$. We follow a Fourier analysis of the subscale equation as the one done in [9] for a FE problem. We have used the same expression for the ROM and for the FOM, and the reason to include the following development is to highlight that the same arguments can be applied in both cases. For simplicity we use in this analysis the scalar stationary convection–diffusion–reaction problem

$$-\nu\Delta y + \boldsymbol{a}\cdot\nabla y + \sigma y = f \qquad \text{in } \Omega,$$

with $\boldsymbol{a}$ a constant convection velocity, and $\nu > 0$ and $\sigma \geq 0$ the diffusion and reaction coefficients, respectively. Following Eq. (20b), we can state a subscale equation as

$$-\nu\Delta\check{y} + \boldsymbol{a}\cdot\nabla\check{y} + \sigma\check{y} = \mathcal{R}_r(y_r) \qquad \text{in } K \in \mathscr{T}_h, \tag{24}$$

with $\mathcal{R}_r(y; y_r)$ defined by using either the ASGSs or the OSGSs methods, and y_r the ROM approximation to y.

We summarize next the arguments introduced in [9], showing that they are also valid for the ROM problem because it is also based on the existence of the FE mesh (see Section 3).

Let us consider the Fourier transform of a generic function g and its derivative on $K \in \mathscr{T}_h$ as

$$\widehat{g}(\boldsymbol{k}) := \int_K e^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{x}} g(\boldsymbol{x}) d\Omega_{\boldsymbol{x}}, \tag{25a}$$

$$\frac{\widehat{\partial g}}{\partial x_j}(\boldsymbol{k}) = \int_{\partial K} n_j e^{-\mathrm{i}\boldsymbol{k}\cdot\boldsymbol{x}} g(\boldsymbol{x}) d\Gamma_{\boldsymbol{x}} + \mathrm{i}k_j\widehat{g}(\boldsymbol{k}), \tag{25b}$$

where $\mathrm{i} = \sqrt{-1}$, $\boldsymbol{k} = (k_1, \ldots, k_d)$ is the wave number, and n_j is now the jth component of the normal exterior to K. Since the subscales $\check{y}$ are the part of the continuous solution that cannot be represented by the ROM approximation, we can argue that their Fourier representation will be dominated by components with high wave numbers. Furthermore, this is also true for the basis $\boldsymbol{\phi}$, whose lower energy modes have a higher spatial frequency. Considering that in Eq. (25b) the second term on the right hand side of the expression dominates the first one for high wave numbers, we can approximate the derivative of g on K as

$$\frac{\widehat{\partial g}}{\partial x_j}(\boldsymbol{k}) \approx \mathrm{i}k_j\widehat{g}(\boldsymbol{k}).$$

Now, we define the dimensionless wave number $\boldsymbol{\kappa} := h\boldsymbol{k}$, where h is a characteristic length. This characteristic length can be defined by the initial discretization given by the partition $\mathscr{T}_h$. Finally, taking the Fourier transform of Eq. (24) we get

$$\widehat{\check{y}}(\boldsymbol{\kappa}) \approx \widehat{\mathcal{L}}(\boldsymbol{\kappa})\widehat{\mathcal{R}}_r(\boldsymbol{\kappa}), \quad \widehat{\mathcal{L}}(\boldsymbol{\kappa}) := \left(\nu\frac{|\boldsymbol{\kappa}|^2}{h^2} + \mathrm{i}\boldsymbol{a}\cdot\frac{\boldsymbol{\kappa}}{h} + \sigma\right)^{-1},$$

from which, using Plancherel's formula and the mean value theorem, we can identify the approximation τ_K in Section 5.2 as $\widehat{\mathcal{L}}(\boldsymbol{\kappa}_0)$, where $\boldsymbol{\kappa}_0$ is the wave number for which the mean value theorem holds. This leads to the expression for the stabilization parameter:

$$\tau_K = \left[\left(c_1\frac{\nu}{h^2}\right)^2 + \left(c_2\frac{|\boldsymbol{a}|}{h}\right)^2 + \sigma^2\right]^{-1/2}, \tag{26}$$

where c_1 and c_2 are constants independent of ν, σ, $\boldsymbol{a}$ and h.

Remark 8. The choice of the characteristic length h is based on the partition $\mathscr{T}_h$. Hence, when the projection-based ROM is based on a mesh based approximation – FE method, SE method, etc. – h is defined in the same way as in any of these cases. In any case, the characteristic length must account for the minimum amount of points needed to represent the FOM solution; this implies that h also depends on the interpolation order.

To illustrate the high spatial frequency in the lower energy modes, Fig. 2 shows the first 6 modes of the basis obtained from the convection–diffusion–reaction problem presented in Section 8.1.

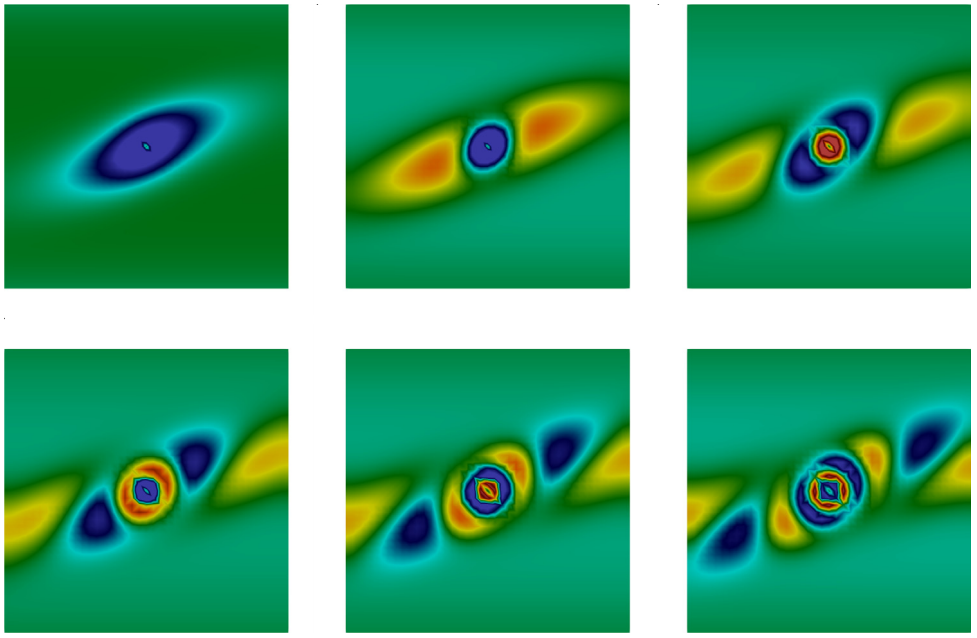


Fig. 2. First 6 modes of the basis obtained from a convection–diffusion–reaction problem embedded in a Couette flow.

5.4. Comments on stability and convergence

Since our proposed ROM formulation matches exactly the FE-FOM by just replacing the FOM space by the ROM one —which is a subspace of the former, the numerical analysis is also the same. A key point in this analysis is that the approximation functions of the ROM space are piece-wise polynomials, thus inheriting properties as the inverse estimates.

The analysis of the method depends on the particular problem being analysed, which is beyond the scope of this paper. A brief description of the main features of the numerical analysis of VMS-based FEs can be found in [2]. Here we only wish to stress the main point relevant to the ROM we have proposed.

The key point in the analysis of stabilized FE methods, and therefore of the ROM we consider, is the introduction of the appropriate working norm, $|||\boldsymbol{y}_r|||$, $\boldsymbol{y}_r \in \mathscr{Y}_r$. This norm includes the terms in the analysis of the Galerkin method – sometimes referred to as graph norm – plus some additional terms that the stabilization terms allow to control, such as pressure gradients in incompressible flow problems or the advective term in advection-dominated equations.

Once stability and well-posedness of the problem have been proved, the final goal of the numerical analysis is to show that the error of the method, measured as $|||\boldsymbol{y} - \boldsymbol{y}_r|||$, is bounded by the interpolation estimate, i.e.,

$$|||\boldsymbol{y} - \boldsymbol{y}_r||| \leq C\, E_{I,r}(h), \quad E_{I,r}(h) := \inf_{\boldsymbol{z}_r \in \mathscr{Y}_r} |||\boldsymbol{y} - \boldsymbol{z}_r|||, \tag{27}$$

with $\boldsymbol{y}$ being the solution of the continuous problem, $\boldsymbol{y}_r$ the ROM solution and C a generic positive constant. In general, the steps to arrive to the estimate in Eq. (27), considering for simplicity stationary and linear problems, are the following:

- Prove stability of the bilinear form of the problem, usually as an inf–sup condition.
- If there is a consistency error, prove that it is bounded by $E_{I,r}(h)$.
- Use the triangle inequality $|||\boldsymbol{y} - \boldsymbol{y}_r||| \leq |||\boldsymbol{y} - \boldsymbol{z}_r||| + |||\boldsymbol{z}_r - \boldsymbol{y}_r|||$ for any $\boldsymbol{z}_r \in \mathscr{Y}_r$. Taking the infimum for $\boldsymbol{z}_r \in \mathscr{Y}_r$, the first term is directly $E_{I,r}(h)$ and the second can be bounded by this error function using stability and the consistency error.

All these steps require the same techniques for both the FE-FOM and the FE-ROM. The final result in the former case is

$$|||\boldsymbol{y} - \boldsymbol{y}_h||| \leq C\, E_{I,h}(h), \quad E_{I,h}(h) := \inf_{z_h \in \mathscr{Y}_h} |||\boldsymbol{y} - \boldsymbol{z}_h|||. \tag{28}$$

The literature about the behavior of $E_{I,h}(h)$ in Eq. (28) is vast. However, the estimate in Eq. (27) is usually expressed in terms of the eigenvalues not taken into account in the POD construction we are employing. In our FE approach, this error could be related to the mesh size h, although we are not aware of such estimate. In any case, the estimate in Eq. (27) is optimal, in the sense that the error of the method is bounded by the interpolation error, which is the best one can expect.

6. Fully discrete approximation

6.1. Projected FOM Galerkin and Petrov–Galerkin projections

For the FOM case, the formulation we propose is given by Eq. (11a) with the subscales $\tilde{\boldsymbol{y}}$ obtained from Eq. (14), with the options described in Section 4.2. This leads at each time step to a matrix problem of the form

$$\mathbf{A}_h(\mathbf{y}_h)\mathbf{y}_h = \mathbf{b}_h, \tag{29}$$

with $\mathbf{A}_h$ a matrix of size $Nn \times Nn$ accounting for the approximation of the temporal derivative, $\mathbf{y}_h$ an array with the Nn unknowns and the right-hand-side array $\mathbf{b}_h$, of size Nn, incorporating the forcing terms and previous values of the FOM solution and the subscales. The dependence of $\mathbf{A}_h$ on $\mathbf{y}_h$ in the case of nonlinear problems has been explicitly displayed.

Similarly, the formulation we propose for the ROM case is given by Eq. (20a) with the subscales $\check{\boldsymbol{y}}$ obtained from Eq. (22), with the options described in Section 5.2. The matrix form of the problem is

$$\mathbf{A}_r(\mathbf{y}_r)\mathbf{y}_r = \mathbf{b}_r, \tag{30}$$

where now $\mathbf{A}_r$ is a matrix of size $r \times r$ and $\mathbf{y}_r$ and $\mathbf{b}_r$ are arrays of r components.

Let $\boldsymbol{\Phi}$ be the $Nn \times r$ matrix whose r columns are the nodal values of the basis functions of the ROM space. From Remark 1 we see that we may consider the FOM solution approximated by $\mathbf{y}_h \approx \boldsymbol{\Phi}\mathbf{y}_r$. Furthermore, if the FOM test function is taken in the ROM subspace, Eq. (29) yields

$$\boldsymbol{\Phi}^\top \mathbf{A}_h(\boldsymbol{\Phi}\mathbf{y}_r)\boldsymbol{\Phi}\mathbf{y}_r = \boldsymbol{\Phi}^\top \mathbf{b}_h. \tag{31}$$

This equation can be considered the FOM one projected onto the ROM subspace. It coincides with Eq. (30) for linear problems *and if the variational formulation employed for the FOM problem is the same as for the ROM one*, as in our case. For nonlinear problems, differences may arise depending on the treatment of the nonlinearities (see Section 6.2).

If we consider that our ROM is the projected FOM given by Eq. (31), we observe that it corresponds to the so called Galerkin projection [12]. Following [12, Chapter 5], we can also consider the Petrov–Galerkin projection in [11] as the appropriate selection of subspaces to solve the discrete problem using a projection method with best conditioning. Omitting the dependence of $\mathbf{A}_h$ on $\boldsymbol{\Phi}\mathbf{y}_r$, in our problem this Petrov–Galerkin projection would lead to the algebraic problem

$$\boldsymbol{\Phi}^\top \mathbf{A}_h^\top \mathbf{A}_h \boldsymbol{\Phi}\mathbf{y}_r = \boldsymbol{\Phi}^\top \mathbf{A}_h^\top \mathbf{b}_h. \tag{32}$$

Let us comment on the need of using the Petrov–Galerkin projection in Eq. (32) instead of the Galerkin projection in Eq. (31). For linear problems we have experimentally found that they both yield virtually the same results, and therefore Eq. (31) – which is the matrix version of the ROM formulation we have presented – suffices. However, in transient nonlinear problems we have found the Petrov–Galerkin projection (Eq. (32)) more robust, allowing one to obtain converged solutions in cases where the Galerkin projection fails to converge in the nonlinear iterative process. Note that, both in linear and nonlinear cases, the terms introduced by our VMS approach are crucial to get stable and accurate solutions.

6.2. Mesh based hyper-ROM

Lastly, in order to reduce the computational cost of evaluating nonlinear terms, we use a mesh-based hyper-ROM. It is similar to the one presented in [28], and has to be considered as an alternative to the sampling-based domain reduction algorithms [11,29–33].

The mesh-based hyper-ROM consists in the solution of the described ROM problem using a coarser mesh than the one of the FOM. The implementation of this technique is done straightforwardly by replacing the expansion in Remark 1 – with $\bar{\boldsymbol{y}} = \boldsymbol{0}$, for simplicity – by

$$\boldsymbol{y}_r^H(\boldsymbol{x},t) := \sum_{a=1}^{N^H} \varphi^a(\boldsymbol{x}) \sum_{k=1}^{r} \boldsymbol{\phi}^{k,a} Y^k(t), \quad \forall \boldsymbol{x} \in \Omega, \quad t \in]0, t_{\mathrm{f}}[, \tag{33}$$

and taking as test functions

$$\boldsymbol{v}_r^H := \sum_{a=1}^{N^H} \varphi^a(\boldsymbol{x}) \sum_{k=1}^{r} \boldsymbol{\phi}^{k,a} \upsilon^k, \quad \forall \boldsymbol{x} \in \Omega,$$

where the superscript H indicates number of nodes of the coarser mesh, with $N^H < N$. The basis functions of the new mesh have still been denoted by $\{\varphi^a(\boldsymbol{x})\}$, now with $a = 1, \ldots, N^H$.

Ideally, the coarsening should be performed as a function of the 'less important' areas of the geometry, which can be achieved using already existing mesh refinement algorithms. The objective would be to coarsen the mesh so that a certain norm of $\boldsymbol{y}_r - \boldsymbol{y}_r^H$ is (asymptotically) smaller than this norm applied to $\boldsymbol{y} - \boldsymbol{y}_r$. However, we have not elaborated this point in this work and in the subsequent examples we have simply tested this technique using a uniform coarsening of the mesh.

Remark 9. When the basis is obtained from a mesh-based solution —a FE in our case, the coarsening of the mesh may imply an interpolation of the nodal values of the basis of the ROM space.

7. Comparison with other stabilization methods

Using Eq. (23), the definitions in Tables 3 and 4, and writing its stabilization part as $\langle \boldsymbol{\tau}_r(\boldsymbol{y}) \boldsymbol{\mathcal{R}}_r(\boldsymbol{y};\boldsymbol{y}_r), \boldsymbol{\mathcal{P}}_r(\boldsymbol{y};\boldsymbol{v}_r) \rangle$ we can describe our proposed VMS method with time dependent OSGSs as

- a full residual method, since $\boldsymbol{\mathcal{R}}_r$ depends on the residual $\boldsymbol{r}$,
- with orthogonality between the subscale space $\breve{\mathscr{Y}}$ and the resolved scale space $\mathscr{Y}_r$, since $\breve{\boldsymbol{\Pi}}$ is defined as the orthogonal projection over the resolved scales, $\boldsymbol{\Pi}_r^{\perp}$,
- with control over all the skew-symmetric terms of $\boldsymbol{\mathcal{L}}$, since $\boldsymbol{\mathcal{P}}_r$ is defined as the adjoint operator $\boldsymbol{\mathcal{L}}^*$,
- and with the numerical parameters $\boldsymbol{\tau}_r$ computed as in the FOM.

Previously proposed methods include classical Streamline-Upwind Galerkin (SUPG) [22,34,35], VMS approaches [22,32], term by term stabilizations [23,36–39], and some empirical methods [7,40–43]. Thus, by identifying these 4 main characteristics in each method we can compare the VMS-ROM with other stabilization techniques found in the literature.

First, by defining $\boldsymbol{\mathcal{R}}_r(\boldsymbol{y};\boldsymbol{y}_r) = \boldsymbol{r}(\boldsymbol{y};\boldsymbol{y}_r)$, $\breve{\boldsymbol{\Pi}} = \boldsymbol{I}$ and $\boldsymbol{\mathcal{P}}_r(\boldsymbol{y};\boldsymbol{v}_r) = \boldsymbol{A}_i^c(\boldsymbol{y})^{\top} \partial_i \boldsymbol{v}_r$, we can describe the SUPG stabilization as a full residual stabilization method, that lacks the orthogonality between spaces and with control only over the convective term of $\boldsymbol{\mathcal{L}}$. Next, we can identify the VMS stabilization approaches with the quasi-static ASGSs stabilization described in Tables 3 and 4. And finally, by defining $\boldsymbol{\mathcal{R}}_r(\boldsymbol{y};\boldsymbol{y}_r) = \boldsymbol{A}_i^c(\boldsymbol{y}) \partial_i \boldsymbol{y}_r$, $\breve{\boldsymbol{\Pi}} = \boldsymbol{\Pi}_r^{\perp}$ and $\boldsymbol{\mathcal{P}}_r(\boldsymbol{y};\boldsymbol{v}_r) = \boldsymbol{A}_i^c(\boldsymbol{y})^{\top} \partial_i \boldsymbol{v}_r$, we can describe the term by term stabilizations as a non-residual, orthogonal and with control only over the skew-symmetric part of $\boldsymbol{\mathcal{L}}$; this family of methods resembles the one described in [44,45] for FEs.

Along with the definitions of $\boldsymbol{\mathcal{R}}_r$ and $\boldsymbol{\mathcal{P}}_r$ we can classify the matrix of stabilization parameters $\boldsymbol{\tau}_r$ in two groups: the one defined based on the FOM counterpart and that in which these parameters are empirical. Although the use of an equivalent $\boldsymbol{\tau}_r$ between FOM and ROM is not an indispensable condition of the method, it is more convenient and avoids the use of parameters that might be specific of the numerical experiment at hand.

Table 5
Comparison of the main characteristics of stabilization methods for ROMs.

$\mathcal{R}_r$	Residual	[22,32,34]
	Term by term	[23,37–39]
	Orthogonal	[23,37–39]
	Non-Orthogonal	[22,32,34]
$\mathcal{P}_r$	Complete $\mathcal{L}^*$	[22,32]
	Convective term	[23,34,37–39]
$\boldsymbol{\tau}_r$	$\boldsymbol{\tau}_h$	[22,32,34,39]
	Empirical	[23,37,38]

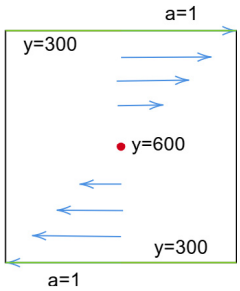


Fig. 3. Convection–diffusion–reaction problem.

Table 5 shows a comparison using the described criteria for the methods aforementioned, excluding the purely empirical ones. In the numerical examples in Section 8 we include a comparison between the OSGSs, ASGSs and a term by term OSGSs stabilization similar to the one proposed for FEs in [6,46], which results in an extended version of the one proposed in [38,39].

8. Numerical examples

In this section we present three numerical examples to test the behavior of the proposed stabilization method. The first example is a convection–diffusion–reaction problem used to illustrate the behavior of the basis in Section 5.3, whereas the second and the third examples consist in incompressible Navier–Stokes problems: a two dimensional flow around a cylinder and a two dimensional backward facing step.

In the three examples the FOM is solved using a stabilized FE method with orthogonal subscales as described in [9], the basis is obtained following the Singular Value Decomposition (SVD) described in the Appendix, the domain discretization for the ROM and hyper-ROM is done using FEs, the time integration is carried out by an implicit BDF, of second order for the resolved scales and of first order for the subscales, and the non-linearity is resolved using Picard's linearization scheme.

8.1. Convection–diffusion–reaction

The aim of this example is to illustrate the behavior of the basis modes (Fig. 2). It consist in a dynamic two dimensional convection–diffusion–reaction problem stated as:

$$\partial_t y - \nu \Delta y + \boldsymbol{a} \cdot \nabla y + \sigma y = 0,$$

with $\nu = 0.01$, $\sigma = 0.01$ and $f = 0$, immersed in a Couette-like flow with a velocity in the horizontal direction illustrated in Fig. 3.

The problem domain is $\Omega =]-1, 1[\times]-1, 1[$ with Dirichlet boundary conditions $y = 300$ in the upper and lower boundaries, adiabatic boundary conditions in the other two walls, initial conditions $y = 300$ except at the central node, which is set to $y = 600$. For both the FOM and the ROM we use a structured quadrilateral mesh with 1600 elements and mesh size $h = 0.05$. The basis is obtained from 100 snapshots and a time step of $\delta t = 0.05$ is used in both the FOM and the ROM.

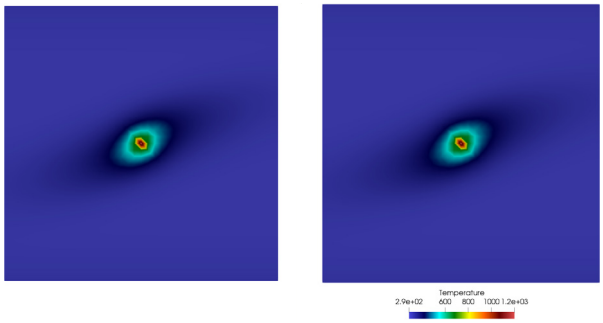


Fig. 4. Convection–diffusion–reaction resolved scales for FOM and ROM with $r = 6$.

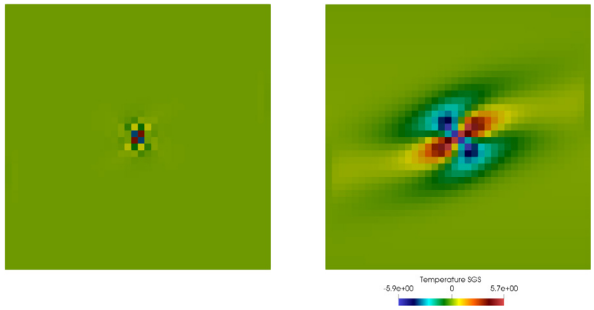


Fig. 5. Convection–diffusion–reaction subscales for FOM and ROM with $r = 6$.

The ROM is stabilized using dynamic OSGSs and the stabilization parameter defined in Eq. (26), with constants $c_1 = 4$ and $c_2 = 2$. As mentioned in Section 6.1, the Galerkin and the Petrov–Galerkin projections yield very similar results for this linear problem. Figs. 4 and 5 show the resolved scales and the subscales for the FOM and ROM at $t = 5$, with a more prominent contribution of the subscales in the ROM.

8.2. Incompressible Navier–Stokes

The velocity $\boldsymbol{u}$ and the pressure p of an incompressible flow are solution of the incompressible Navier–Stokes equations

$$\begin{aligned} \partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \nabla p = \boldsymbol{f} \qquad & \text{in } \Omega, \quad t \in \,]0, t_f[, \\ \nabla \cdot \boldsymbol{u} = 0 \qquad & \text{in } \Omega, \quad t \in \,]0, t_f[, \end{aligned}$$

where ν is the kinematic viscosity and $\boldsymbol{f}$ a vector of body forces. Initial and boundary conditions are set as

$$\begin{aligned} \boldsymbol{u}(\boldsymbol{x}, 0) = \boldsymbol{u}_0(\boldsymbol{x}) \qquad & \text{in } \Omega, \quad t \in \,]0, t_f[, \\ \boldsymbol{u} = \boldsymbol{u}_D \qquad & \text{on } \Gamma_D, \quad t \in \,]0, t_f[, \\ \boldsymbol{\mathcal{F}}(\boldsymbol{y}; \boldsymbol{y}) = \boldsymbol{n} \cdot p\boldsymbol{I} - \boldsymbol{n} \cdot \nu \nabla \boldsymbol{u} = \boldsymbol{t}_N \qquad & \text{on } \Gamma_N, \quad t \in \,]0, t_f[, \end{aligned}$$

where Γ_D and Γ_N are a partition of the boundary $\partial\Omega$.

Let $H^1(\Omega)$ be the space of functions whose distributional derivatives belong to $L^2(\Omega)$. Let $\mathcal{V}$ be the space of vector functions with components in $H^1(\Omega)$ that are equal to $\boldsymbol{u}_D$ on Γ_D, and $\mathcal{V}_0$ the space of vector functions, also with components in $H^1(\Omega)$, that vanish on Γ_D. Set $\mathcal{Q} = L^2(\Omega)$. The variational form of the problem we consider is defined in the spaces $\mathcal{Y} = \mathcal{V} \times \mathcal{Q}$ for the trial solutions (for each time t) and $\mathcal{Y}_0 = \mathcal{V}_0 \times \mathcal{Q}$ for the test functions, which we shall write as $\boldsymbol{y} = [\boldsymbol{u}, p]$ and $\boldsymbol{v} = [\boldsymbol{v}, q]$, respectively. As done along the text, a subscript h will be used for the FE-FOM spaces and functions and a subscript r for the ROM spaces and functions.

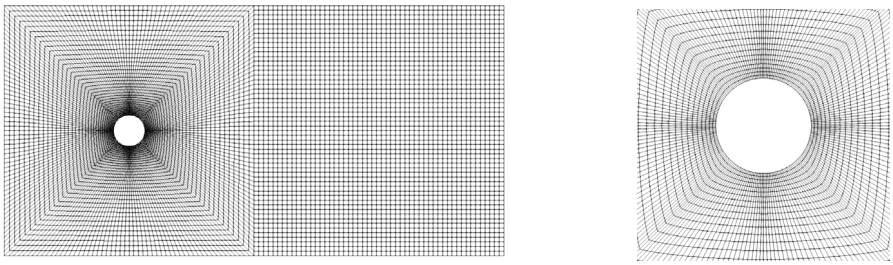


Fig. 6. Mesh with 92960 grid points and 23040 quadrilateral elements. Darker areas indicate a more refined mesh.

In the problem we consider now, the following terms correspond to the abstract ones introduced along the paper using dynamic OSGSs and assuming the force term to belong to the finite element space (see Tables 3 and 4):

$$
\begin{aligned}
\mathcal{M}(\boldsymbol{y}) &= \begin{bmatrix} \boldsymbol{I} & \boldsymbol{0}^\top \\ \boldsymbol{0} & 0 \end{bmatrix}, \\
\mathcal{L}(\boldsymbol{y};\boldsymbol{y}_r) &= \begin{bmatrix} \boldsymbol{u}\cdot\nabla\boldsymbol{u}_r - \nu\Delta\boldsymbol{u}_r + \nabla p_r \\ \nabla\cdot\boldsymbol{u}_r \end{bmatrix}, \quad \mathcal{L}^*(\boldsymbol{y};\boldsymbol{v}_r) = \begin{bmatrix} -\boldsymbol{u}\cdot\nabla\boldsymbol{v}_r - \nu\Delta\boldsymbol{v}_r - \nabla q_r \\ -\nabla\cdot\boldsymbol{v}_r \end{bmatrix}, \\
B(\boldsymbol{y};\boldsymbol{y}_r,\boldsymbol{v}_r) &= \langle \boldsymbol{u}\cdot\nabla\boldsymbol{u}_r, \boldsymbol{v}_r\rangle + (\nu\nabla\boldsymbol{u}_r - p_r\boldsymbol{I}, \nabla\boldsymbol{v}_r) + (\nabla\cdot\boldsymbol{u}_r, q_r), \\
B_s(\boldsymbol{y};\boldsymbol{y}_r,\boldsymbol{v}_r) &= -\sum_K \langle \boldsymbol{\Pi}_r^\perp(\boldsymbol{u}\cdot\nabla\boldsymbol{u}_r - \nu\Delta\boldsymbol{u}_r + \nabla p_r), \tau_{1,t}(\boldsymbol{u}\cdot\nabla\boldsymbol{v}_r + \nu\Delta\boldsymbol{v}_r + \nabla q_r)\rangle_K \\
&\quad - \sum_K \langle \boldsymbol{\Pi}_r^\perp(\nabla\cdot\boldsymbol{u}_r), \tau_2\nabla\cdot\boldsymbol{v}_r\rangle_K, \qquad (35a) \\
L_s(\boldsymbol{y};\boldsymbol{v}_r) &= -\sum_K \langle \boldsymbol{\Pi}_r^\perp(\boldsymbol{f} + \delta t^{-1}\breve{\boldsymbol{u}}, \tau_{1,t}(\boldsymbol{u}\cdot\nabla\boldsymbol{v}_r + \nu\Delta\boldsymbol{v}_r + \nabla q_r))\rangle_K, \qquad (35b)
\end{aligned}
$$

and likewise for $\boldsymbol{y}_h = [\boldsymbol{u}_h, p_h]$ in the FOM formulation. The stabilization parameter matrix is defined following [9, 47] as

$$
\begin{aligned}
\boldsymbol{\tau}_K^{-1}(\boldsymbol{y}) &= \mathrm{diag}(\tau_1\boldsymbol{I}, \tau_2) = \begin{bmatrix} (c_1\frac{\nu}{h^2} + c_2\frac{|\boldsymbol{u}|_K}{h})\boldsymbol{I} & \boldsymbol{0}^\top \\ \boldsymbol{0} & \nu + \frac{c_2}{c_1}|\boldsymbol{u}|h \end{bmatrix}, \qquad \text{in } K \in \mathcal{T}_h, \\
\tau_{1,t}^{-1} &= \delta t^{-1} + \tau_1^{-1},
\end{aligned}
$$

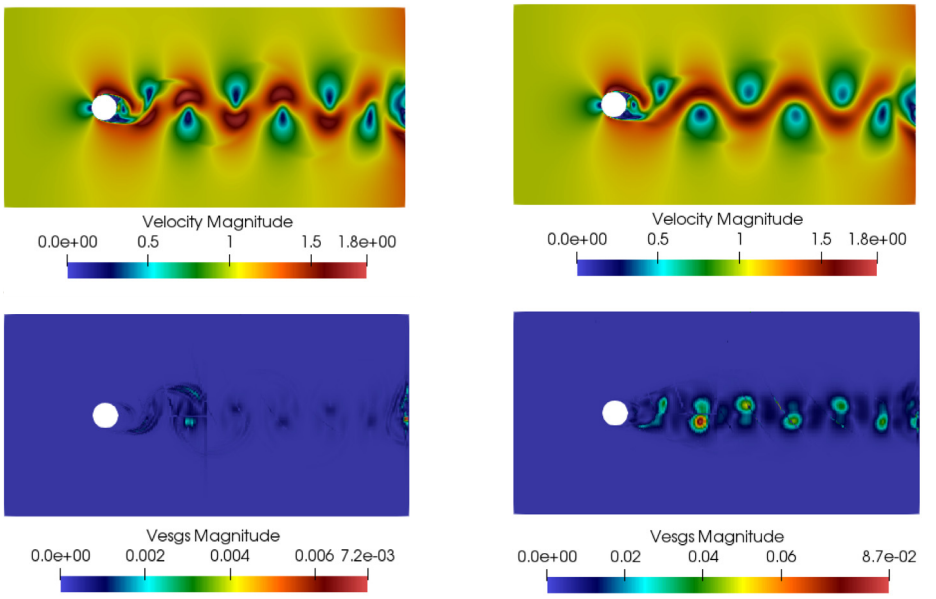
with $c_1 = 4$ and $c_2 = 2$ for linear elements. In Eq. (35b), a BDF1 time integration scheme has been assumed for the SGS, with $\breve{\boldsymbol{u}}$ evaluated at t^n when solving for the unknowns evaluated at t^{n+1}.

8.2.1. Flow over a cylinder

The second numerical example is the two dimensional flow over a cylinder. The computational domain is $\Omega =]0, 16[\times]0, 8[$, with the cylinder D of diameter 1 and centered at $(4, 4)$. The velocity at $x = 0$ is prescribed to $(1, 0)$, whereas at $y = 0$ and $y = 8$ the y-velocity component is prescribed to 0 and the x-velocity component is left free. The outflow (where both the x and y velocity components are left free) is $x = 16$. The viscosity is prescribed to $\nu = 0.001$, resulting in a Reynolds number of 1000. The domain is discretized using a symmetric mesh of 92320 bilinear elements (Fig. 6).

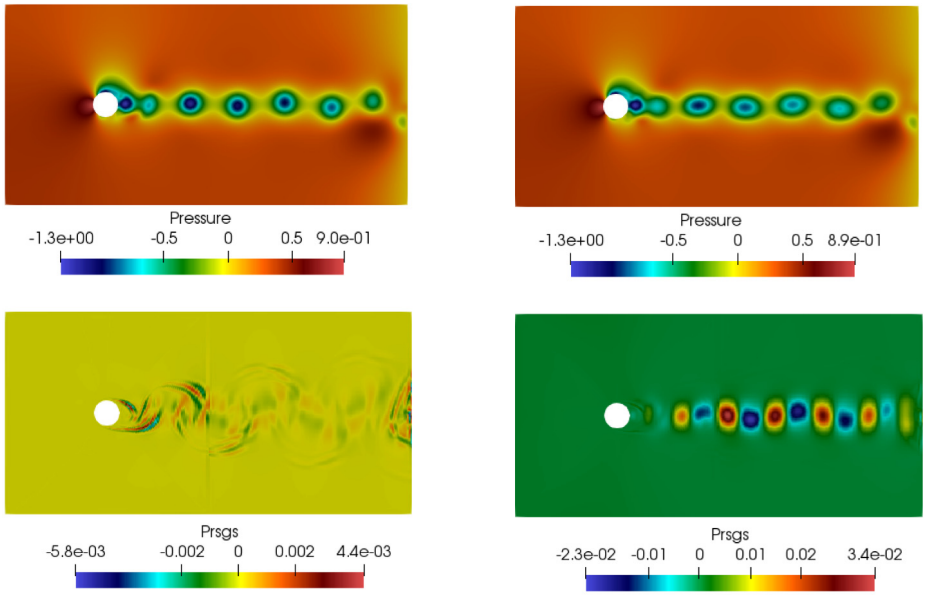
To obtain the fully developed vortex shedding characteristic of this problem, a preliminary simulation is performed until $t = 100$, reset as $t = 0$. From that time, solutions from the next 500 time steps of velocity and pressure are used to calculate the ROM basis. A time step of $\delta t = 0.05$ is used in both the FOM and the ROM.

Figs. 7 and 8 show contour plots of the velocity and pressure resolved scales and subscales for both the FOM and the ROM at $t = 50$. Note that for this example the contribution of the subscales in the ROM is bigger in almost one order of magnitude compared to the FOM.



(a) FOM (b) ROM with $\eta = 0.8$

Fig. 7. Velocity (top) and velocity subscales (bottom) at $t = 50$.



(a) FOM (b) ROM with $\eta = 0.8$

Fig. 8. Pressure (top) and pressure subscales (bottom) at $t = 50$.

To evaluate the accuracy of the ROM, we perform a series of numerical tests using 7 different sets of bases, varying the retained energy from $\eta_1 = 0.975$ to $\eta_7 = 0.6$ (see Eq. (A.5)). Fig. 9 depicts the number of vectors in the basis as a function of the retained energy.

These numerical tests consist in comparing:

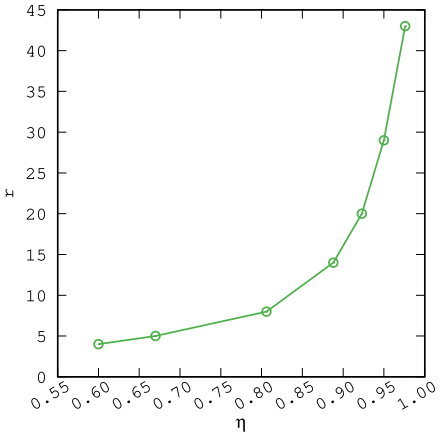


Fig. 9. Number of basis vectors r as a function of the retained energy η.

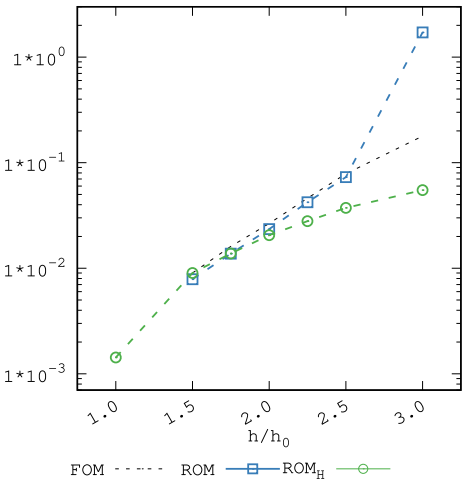


Fig. 10. $F^{\circ}_{\mathrm{RMSD}}$ convergence for FOM, ROM and ROM$_H$ with $\eta = 0.9$.

- Results for different types of subscales:
  - OSGSs vs. ASGSs.
  - Dynamic vs. quasi-static subscales.
- The interpolation order of the mesh, namely, linear and quadratic.
- The use of the full residual based stabilization or the simplified term-by-term method proposed in [46].

The comparison for all cases is done using the norm of the total force exerted over the cylinder, with the force F° on the cylinder boundary $\Gamma_{\circ}$ defined as

$$F^{\circ}(t) = \left| \int_{\Gamma_{\circ}} \boldsymbol{\mathcal{F}}_n(\boldsymbol{u}(\boldsymbol{x}, t)) \, \mathrm{d}\Gamma \right|.$$

Additionally, we choose a Root-Mean-Square Deviation (RMSD) of the ROM solution with respect to the FOM one as a way to measure the overall error. If $F^{\circ}_{j,\mathrm{FOM}}$ is the total force obtained with the FOM at time t^j, $j = 1, \ldots, S$, and $F^{\circ}_{j,\mathrm{ROM}}$ the one obtained with the ROM, we set

$$F^{\circ}_{\mathrm{RMSD}} = \sqrt{\frac{1}{S} \sum_{j=1}^{S} \left(F^{\circ}_{j,\mathrm{ROM}} - F^{\circ}_{j,\mathrm{FOM}} \right)^2}.$$

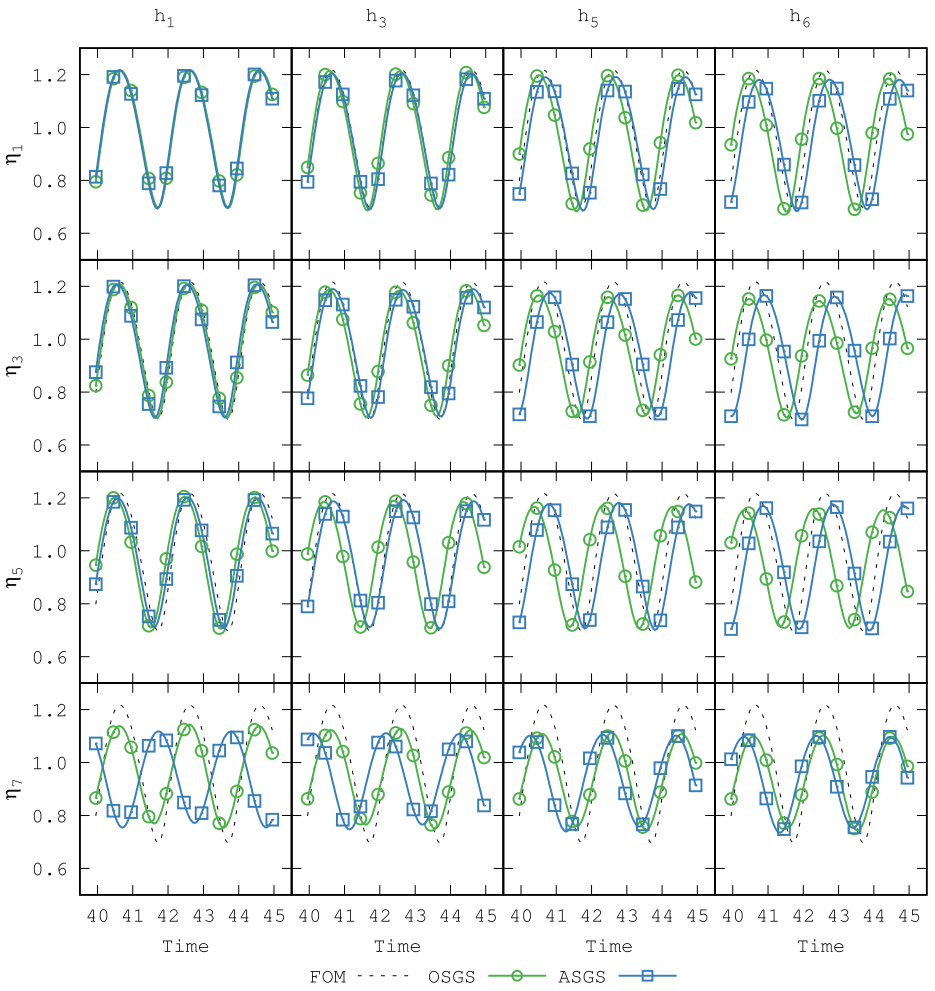


Fig. 11. Comparison of F° for OSGSs and ASGSs.

Along with the error measurement, to depict the behavior of the temporal evolution we perform a discrete Fourier transform of the force. And lastly, we assess the performance using the computational time ratio between the ROM and FOM solutions.

To test the behavior of the mesh based hyper-ROM, we perform the same numerical tests using 7 different meshes, varying from $^{h_1}/_{h_0} = 1$ to $^{h_7}/_{h_0} = 3.0$ with h_0 the FOM mesh element size for a fixed η. We call ROM_H the ROM solution obtained with the different hyper-ROM meshes. Thus, the models we consider are the standard FOM and the standard ROM for varying h and the ROM_H based on the finest mesh h_0 and different hyper-ROM meshes.

Fig. 10 shows a convergence of the RMSD for the FOM, the ROM using the basis obtained with the original fine mesh, and the ROM_H. Note that for coarser meshes the amount of information provided by the FOM solution is not sufficient to generate an adequate basis for the ROM.

Orthogonality. The first test performed consists in comparing the two definitions of the projection $\check{\boldsymbol{\Pi}}_\tau$. For this purpose we solve the ROM using the two types of subscales defined in Section 5: OSGSs and ASGSs. Fig. 11 depicts the total force in the time interval $[40, 45]$ for several combinations of mesh and basis sizes, using the two types of subscales. The OSGSs behaves as expected, it deteriorates when less amount of basis vectors and a coarser mesh are used, contrary to the ASGSs solution, which is more erratic. Fig. 12 shows a comparison of the spectra of the force for three combinations of η and h.

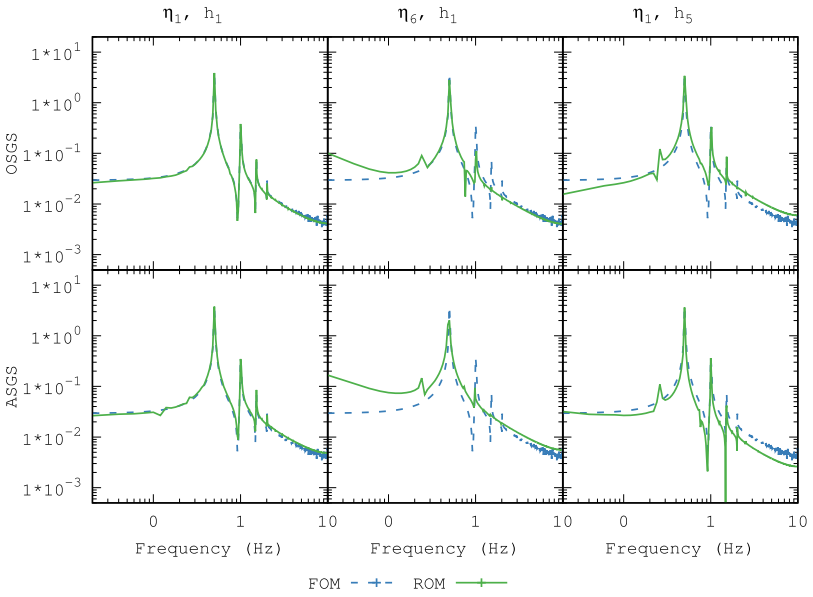
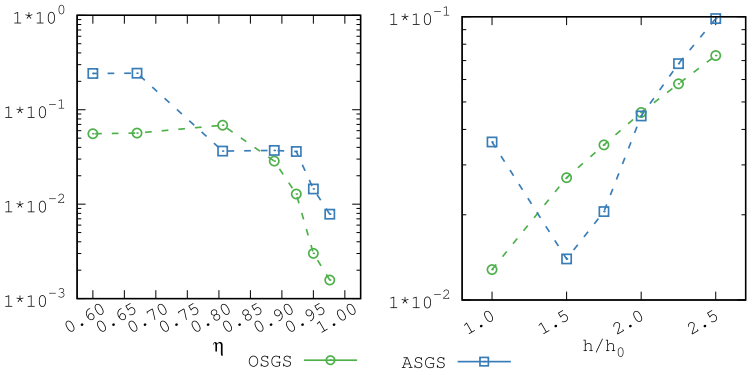


Fig. 12. Fourier transform of F° for OSGSs and ASGSs.



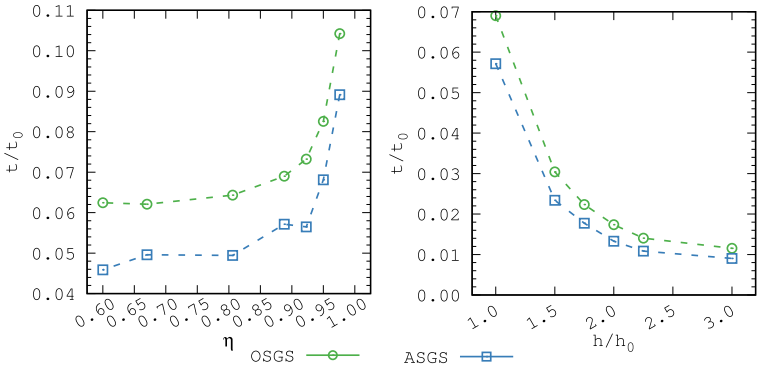
(a) Retained energy. ROM, $\frac{h}{h_0} = 1.0$ (b) Element size. Hyper-ROM. $\eta = 0.925$

Fig. 13. F°_{RMSD} convergence for OSGSs and ASGSs.

Fig. 13 depicts the force RMSD convergence for ROM – varying the retained energy η with a fixed $\frac{h}{h_0} = 1.0$ – and for hyper-ROM — varying the mesh element size h with a fixed $\eta = 0.925$. Although the convergence error does not have a clear slope, it behaves as expected for OSGSs, with the error decreasing with the addition of basis vectors and a finer mesh. Contrary to the ASGSs, where the convergence behavior is irregular.

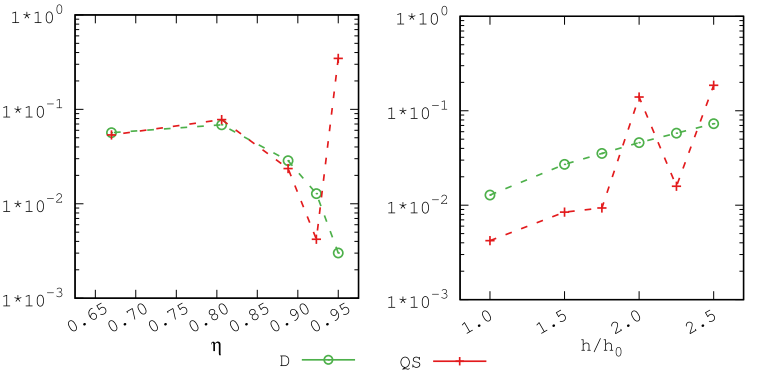
Fig. 14 shows the computational time ratio for ROM and hyper-ROM solutions, with the expected behavior. Computational savings up to 99% for smaller basis size and coarser meshes are observed. Despite the use of OSGSs being more costly, the computational time difference between the two types of subscales is less than 2%.

Dynamic subscales. In the second test we compare dynamic and quasi-static subscales. Fig. 15 depicts the RMSD of the force for ROM varying the retained energy η, and for hyper-ROM varying the mesh element size h. Here, the behavior of the ROM subscales mirror the results shown in [5,48] for a VMS-FE method: the stability of the formulation is improved when the dynamic subscales are used. The difference in computational cost for both cases is smaller than 0.05% in the ROM and negligible in the hyper-ROM case.



(a) Retained energy. ROM, $\frac{h}{h_0} = 1.0$ (b) Element size. Hyper-ROM. $\eta = 0.925$

Fig. 14. Computational time ratio for OSGSs and ASGSs.



(a) Retained energy. ROM, $\frac{h}{h_0} = 1.0$ (b) Element size. Hyper-ROM. $\eta = 0.925$

Fig. 15. F°_{RMSD} convergence for dynamic and quasi-static OSGSs.

Interpolation order. In the third test we analyse the behavior of the ROM depending on the interpolation order of the mesh used to calculate the basis and the one used in solving the problem. We calculate two different bases, with bilinear elements (Fig. 6) and with 9-nodes biquadratic elements maintaining a similar amount of nodes, 92320 for the bilinear mesh and 92960 for the biquadratic mesh. We solve three sets of examples:

1. using the bilinear element basis to solve a ROM with bilinear elements (LB–LE),
2. using the biquadratic element basis to solve a ROM with biquadratic elements (QB–QE),
3. and using the biquadratic element basis to solve a ROM with bilinear elements (QB–LE).

Fig. 16 depicts the force in the time interval $t = [40, 45]$ for several combinations of mesh and basis sizes. For the ROM case, results behave as expected with no noticeable difference between the different bases and meshes. Conversely for hyper-ROM, the use of biquadratic bases (QB) presents worse results for meshes of size different from h_0. Fig. 17 depicts a comparison of the number of vectors in the basis in function of the retained energy for bases calculated with bilinear and biquadratic elements.

Fig. 18 shows a comparison of the spectra of the force for three combinations of η and h. Fig. 19 depicts the force RMSD convergence for ROM varying the retained energy η with a fixed $\frac{h}{h_0} = 1.0$, and for hyper-ROM varying the mesh element size h with a fixed $\eta = 0.925$. Notice the better convergence in hyper-ROM for $\frac{h}{h_0} = 2$ when using the biquadratic basis, this point corresponds to a mesh where the location of the nodes coincides with the nodes in h_0.

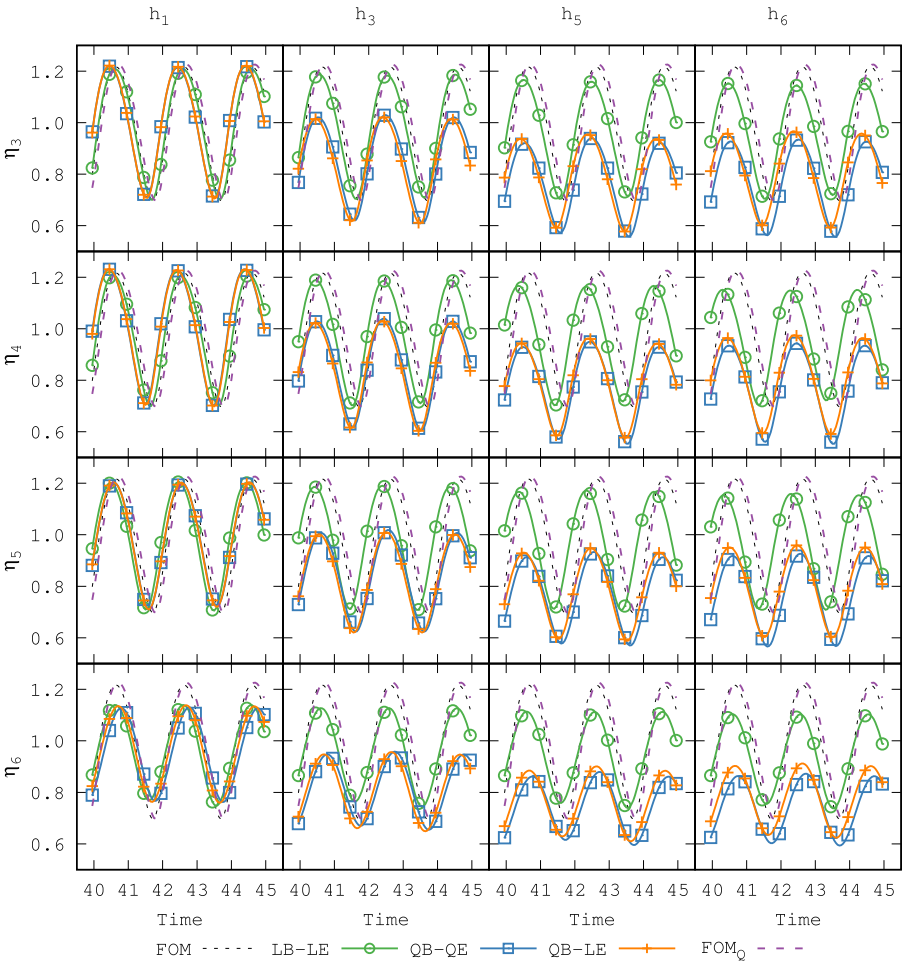


Fig. 16. Comparison of F° for bases obtained with biquadratic (QB) and bilinear (LB) elements projected over biquadratic (QE) and bilinear (LE) elements.

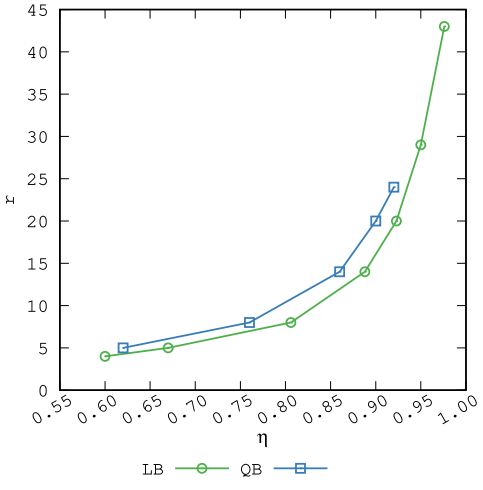


Fig. 17. Number of basis vectors r as a function of the retained energy η for bases calculated using biquadratic (QB) and bilinear (LB) elements.

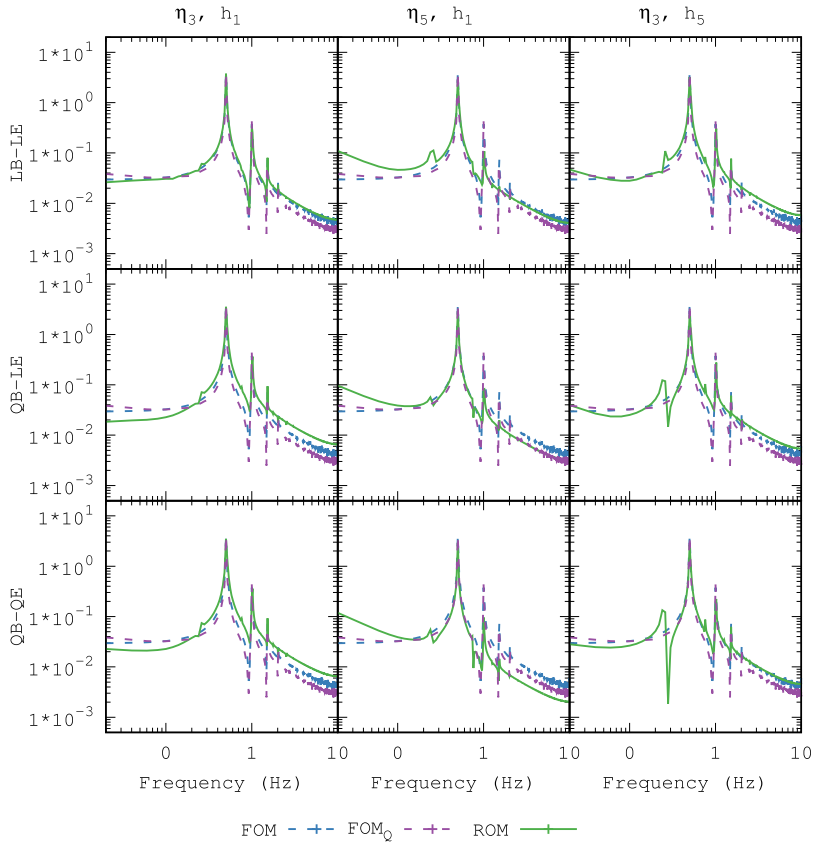
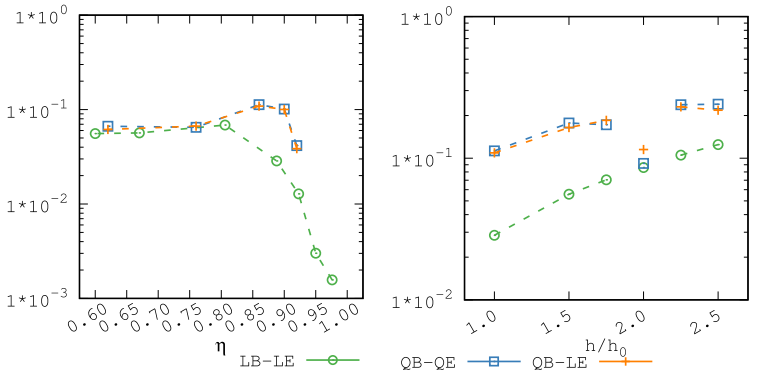


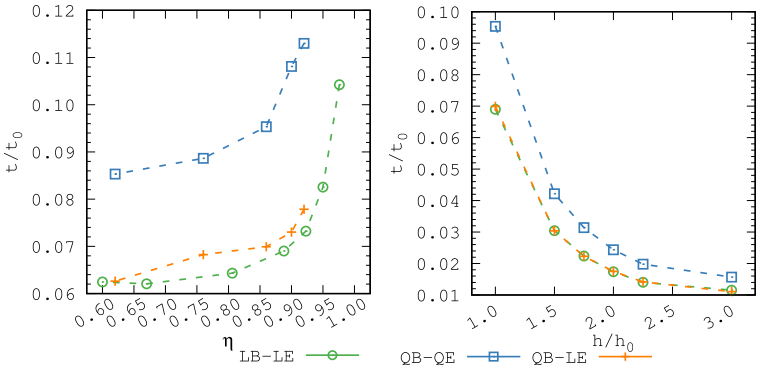
Fig. 18. Fourier transform of F° for bases obtained with biquadratic (QB) and bilinear (LB) elements projected over biquadratic (QE) and bilinear (LE) elements.



(a) Retained energy. ROM, $\frac{h}{h_0} = 1.0$ (b) Element size. Hyper-ROM. $\eta = 0.925$

Fig. 19. F°_{RMSD} convergence for bases obtained with biquadratic (QB) and bilinear (LB) elements projected over biquadratic (QE) and bilinear (LE) elements.

Fig. 20 shows the computational time ratio for ROM and hyper-ROM solutions. As expected the basis used presents almost no influence in the computational cost. When comparing the computational time ratio between bilinear and biquadratic elements, whereas for the FOM this ratio is close to 3, the same ratio for the ROM is lower than 1.5, showing that the cost associated with mesh resolution also decreases.



(a) Retained energy. ROM, $\frac{h}{h_0} = 1.0$ (b) Element size. Hyper-ROM, $\eta = 0.925$

Fig. 20. Computational time ratio t/t_0 of the ROM solution (t) with respect to the FOM solution (t_0) for bases obtained with biquadratic (QB) and bilinear (LB) elements projected over biquadratic (QE) and bilinear (LE) elements.

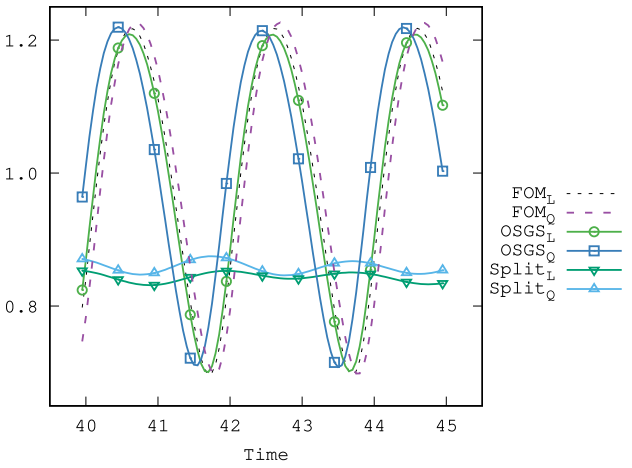


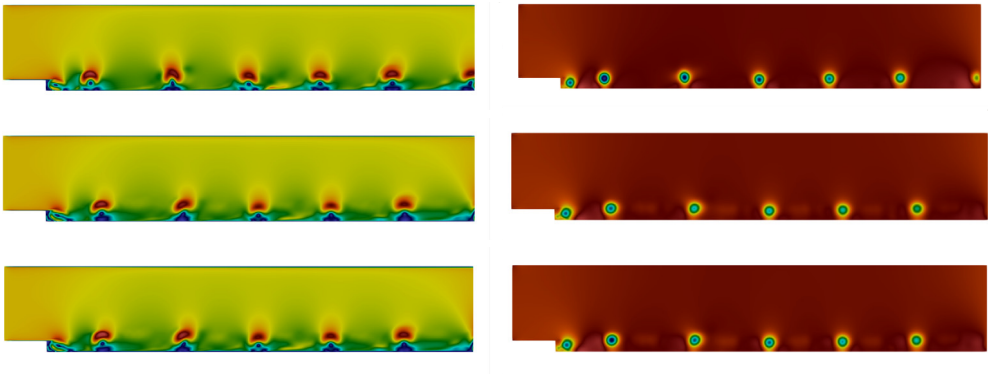
Fig. 21. Comparison of F° for OSGSs and term-by-term (Split) stabilizations using bilinear (subscript L) and biquadratic (subscript Q) elements.

Residual-based stabilization. Lastly, we compare the VMS stabilized formulation proposed against an OSGSs term-by-term stabilization. This formulation results in method similar to the ones described in [23,37–39] with the particularity that the stabilization term of the pressure is added in order to be able to use equal interpolations for $\boldsymbol{u}_r$ and p_r. Following the formulation presented in [46] for FEs, we split the velocity subscales as $\breve{\boldsymbol{u}} = \breve{\boldsymbol{u}}_1 + \breve{\boldsymbol{u}}_2$. Furthermore, we assume that the convective and pressure terms in the solution of the subscales are orthogonal and neglect second order space derivatives, yielding

$$\partial_t \breve{\boldsymbol{u}}_1 + \tau_1^{-1} \breve{\boldsymbol{u}}_1 = -\boldsymbol{\Pi}_r^{\perp}(\boldsymbol{u} \cdot \nabla \boldsymbol{u}_r),$$
$$\partial_t \breve{\boldsymbol{u}}_2 + \tau_1^{-1} \breve{\boldsymbol{u}}_2 = -\boldsymbol{\Pi}_r^{\perp}(\nabla p_r).$$

Using this definition of the velocity subscales and using the same stabilization parameters $\boldsymbol{\tau}$ as for the original OSGSs formulation, we redefine the stabilization terms (Eqs. (35a) and (35b)) as

$$B_s(\boldsymbol{y}; \boldsymbol{y}_r, \boldsymbol{v}_r) = -\sum_K \langle \boldsymbol{\Pi}_r^{\perp}(\boldsymbol{u} \cdot \nabla \boldsymbol{u}_r), \tau_{1,t} \boldsymbol{u} \cdot \nabla \boldsymbol{v}_r \rangle_K - \sum_K \langle \boldsymbol{\Pi}_r^{\perp}(\nabla \cdot \boldsymbol{u}_r), \tau_2 \nabla \cdot \boldsymbol{v}_r \rangle_K - \sum_K \langle \boldsymbol{\Pi}_r^{\perp}(\nabla p_r), \tau_{1,t} \nabla q_r \rangle,$$
$$L_s(\boldsymbol{y}; \boldsymbol{v}_r) = -\sum_K \langle \delta t^{-1} \breve{\boldsymbol{u}}_1, \tau_{1,t} \boldsymbol{u} \cdot \nabla \boldsymbol{v}_r \rangle_K - \sum_K \langle \delta t^{-1} \breve{\boldsymbol{u}}_2, \tau_{1,t} \nabla q_r \rangle.$$



(a) Velocity contour plots. (b) Pressure contours plots.

Fig. 22. From top to bottom: FOM, ROM and hyper-ROM with $\eta = 0.95$ and $r = 21$.

In Fig. 21 we compare the total force at the time interval [40, 45] for both dynamic OSGSs formulations, namely, the residual one and the term-by-term one, using bilinear and biquadratic elements. We also tested a quasi-static term-by-term stabilization, equivalent to the one presented in [6] for FEs, obtaining similar results. As it is observed, in this particular case the term-by-term method yields overdiffusive results. Let us stress again that this formulation is similar to the one presented in the references quoted above in what concerns the stabilization of the convective term.

8.2.2. Backward-facing step

The third numerical example consists in a two dimensional flow over a backward facing step similar to the one presented in [49] and solved using model reduction in [7,32]. The computational domain is the rectangle $]0, 44[\times]0, 9[$, with a unit length step placed at (4, 0). The inflow velocity at $x = 0$ is prescribed to (1.0, 0), whereas at the lower and upper boundaries a wall law boundary condition is set with the wall distance characterizing the wall model $\delta = 0.001$. The outflow (where both the x and y velocity components are left free) is set at $x = 44$. The viscosity is prescribed to $\nu = 0.00005$, resulting in a Reynolds number of 20000. The domain is discretized using a symmetric mesh of 61520 quadrilateral bilinear elements and 62214 nodes for the FOM and the ROM, and 34800 quadrilateral bilinear elements and 35321 nodes for the hyper-ROM.

As done for the previous example, a preliminary simulation is performed until $t = 100$ and the solution of the next 1000 time steps for velocity and pressure is gathered to calculate the basis. A time step of $\delta t = 0.05$ is used in both FOM, ROM and hyper-ROM cases. Fig. 22 shows velocity and pressure contour plots for FOM, ROM and hyper-ROM at $t = 50$.

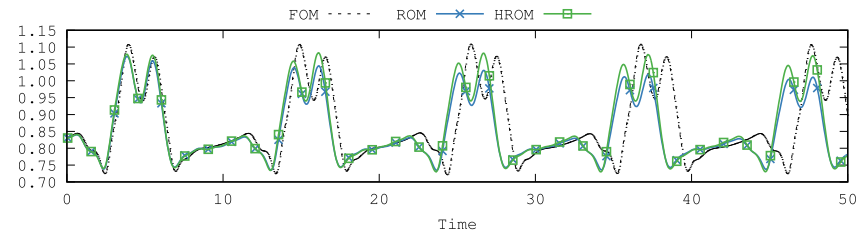
Fig. 23 shows a comparison of the velocity magnitude and the pressure at the control point (22, 1) for the FOM, ROM and hyper-ROM cases. Fig. 24 shows a comparison of the spectra of the velocity and the pressure at the same control point.

Despite the higher complexity of the solution in comparison to the previous examples, the ROM and hyper-ROM capture successfully the behavior of the flow, maintaining the amplitudes of velocity and pressure with a small shift in frequency at the end of the simulation. Further numerical examples were tested using a higher amount of basis vectors, resulting in the same overfitting phenomena reported in [10].

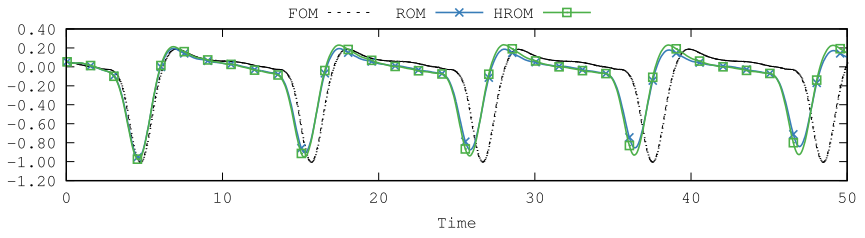
Regarding the performance, the ROM and hyper-ROM computational time ratios with respect to the FOM time are $t_{\text{ROM}}/t_{\text{FOM}} = 0.14$ and $t_{\text{HROM}}/t_{\text{FOM}} = 0.08$, respectively.

9. Conclusions

In this paper we have presented a VMS formulation for projection-based ROMs, with two main distinctive features: the orthogonality between resolved scales and SGSs and the time dependency of the latter. It has also been justified why the stabilization parameters can be the same for the ROM and for the original FOM. The most important features of this method are the following:

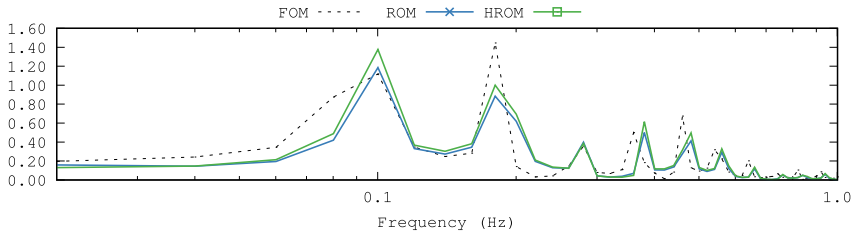


(a) Velocity magnitude

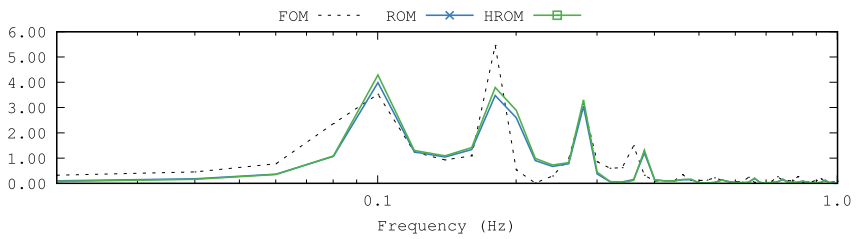


(b) Pressure

Fig. 23. Comparison at the control point (22, 1) for FOM, ROM and hyper-ROM with $\eta = 0.95$ and $r = 21$.



(a) Velocity magnitude



(b) Pressure

Fig. 24. Fourier transform at the control point (22, 1) for FOM, ROM and hyper-ROM with $\eta = 0.95$ and $r = 21$.

- The use of orthogonal SGSs shows an overall more consistent behavior over variations of η and h than the non-orthogonal SGSs.
- The use of the mesh based hyper-ROM shows a convergence over the mesh size h that resembles the one for FE. This shows not only the consistency of the VMS-ROM, but also that adaptive mesh refinement techniques used in other mesh based methods (FE, for example) might be adequate for a mesh based hyper-ROM.
- The use of higher order interpolations in the domain discretization can be done straight-forwardly, and the characteristics of the FOM solution achieved are retained in the ROM even when lower order interpolations are used.

- The use of dynamic subscales seems to be relevant for the stabilization and consistency of the model, as in the FE counterpart.

Additionally, other stabilization methods were tested with less satisfactory results:

- The use of ASGSs shows a less stable behavior over variations of η and h.
- The use of term by term stabilizations, analyzed for FEs in [6], does not perform appropriately in projection-based ROMs, being over-diffusive at best.

Acknowledgements

R. Reyes acknowledges the scholarship received from COLCIENCIAS, from the Colombian Government. R. Codina acknowledges the support received from the ICREA Acadèmia Research Program, from the Catalan Government.

Appendix. Construction of the basis using POD

The objective of the POD method is finding a basis for a collection of high-fidelity data (snapshots) to use it as the basis of the desired reduced subspace. We start by organizing a set of data previously obtained from a FOM solution over the time grid $0 \leq t_1 < t_{\mathrm{M}} \leq t_{\mathrm{f}}$ as

$$\{\boldsymbol{s}^j\}_{j=1}^{M} = \{\boldsymbol{s}^1, \boldsymbol{s}^2, \ldots, \boldsymbol{s}^M\}, \tag{A.1}$$

where $\boldsymbol{s}^j = \boldsymbol{y}(\boldsymbol{x}, t_j) - \bar{\boldsymbol{y}}$, $\boldsymbol{y}_j = \boldsymbol{y}(\boldsymbol{x}, t_j) \subset \mathscr{Y}_h$ a jth time snapshot and $\bar{\boldsymbol{y}}$ the mean value of the snapshots. We can denote as $\mathscr{Y}_R = \mathrm{span}\{\boldsymbol{s}^j\}_{j=1}^{M}$ the space spanned by the ensemble of data (snapshot space), and given that in general the snapshots are not linearly independent, it holds $R \leq M$, with $R = \dim(\mathscr{Y}_R)$. Now, if $\{\boldsymbol{\phi}^k\}_{k=1}^{R}$ is an orthonormal basis of $\mathscr{Y}_R$, then any element $\boldsymbol{y}_j$ in the ensemble in Eq. (A.1) can be written as

$$\boldsymbol{y}_j = \bar{\boldsymbol{y}} + \sum_{k=1}^{R} (\boldsymbol{s}_j, \boldsymbol{\phi}^k)\boldsymbol{\phi}^k. \tag{A.2}$$

The POD consists in finding the orthonormal basis $\{\boldsymbol{\phi}^k\}_{k=1}^{r}$ that provides the best possible approximation of the given snapshots, with $r \leq R$. It can be formulated as an optimization problem, where for every $k \in \{1, \ldots, r\}$ the mean square error between the elements $\boldsymbol{s}_j$, $1 \leq j \leq M$, and the corresponding partial sum in Eq. (A.2) is minimized on average as

$$\begin{aligned} &\min_{\{\boldsymbol{\phi}^k\}_{k=1}^{r}} \quad \frac{1}{M}\sum_{j=1}^{M} \left\| \boldsymbol{s}_j - \sum_{k=1}^{r} (\boldsymbol{s}_j, \boldsymbol{\phi}^k)\boldsymbol{\phi}^k \right\|^2, \\ &\text{subject to } (\boldsymbol{\phi}^i, \boldsymbol{\phi}^j) = \delta_{ij}, \quad 1 \leq i, j \leq r. \end{aligned} \tag{A.3}$$

As a result, the POD space $\mathscr{Y}_r$ is a subspace of $\mathscr{Y}_h$. Altogether, the inclusions $\mathscr{Y}_r \subseteq \mathscr{Y}_R \subset \mathscr{Y}_h \subset \mathscr{Y}$ hold, with the dimensional relation $r \leq R \leq M \ll Nn$.

In our case, using properties (6) we can write the problem in Eq. (A.3) in a discrete form as

$$\begin{aligned} &\min_{\mathbf{M}^{1/2}\boldsymbol{\Phi} \in \mathbb{R}^{Nn \times r}} \left\| \mathbf{M}^{1/2}\mathbf{S} - \mathbf{M}^{1/2}\boldsymbol{\Phi}\boldsymbol{\Phi}^{\top}\mathbf{M}\mathbf{S} \right\|^2, \\ &\text{subject to} \quad \boldsymbol{\Phi}^{\top}\mathbf{M}\boldsymbol{\Phi} = \mathbf{I} \in \mathbb{R}^{r \times r}, \end{aligned} \tag{A.4}$$

where $\mathbf{S}$ is the set of organized discrete snapshots. To solve the problem in Eq. (A.4) we follow a SVD of the weighted snapshot collection $\widehat{\mathbf{S}} = \mathbf{M}^{1/2}\mathbf{S} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{V}$, where the resulting left singular-vectors of the decomposition represent the weighted basis $\mathbf{U} = \mathbf{M}^{1/2}\boldsymbol{\Phi}$, of size R.

Lastly, based on the Eckart–Young–Mirsky theorem we can reduce furthermore the basis by truncating the reduced basis $\boldsymbol{\Phi}$ at the rth column. As a criterion for the truncation, we use the retained energy η defined in [16] as

$$\eta = \frac{\sum_{k=1}^{r} \lambda^k}{\sum_{k=1}^{R} \lambda^k}, \tag{A.5}$$

where $\{\lambda^k\}_{k=1}^R$ are the SVD non-zero singular values and the truncation error is defined as $\epsilon = \sum_{k=r+1}^R (\lambda^k)^2$.

Note that this is the way we construct $\boldsymbol{\Phi}$, but there are several other ways to find a basis for the reduced order subspace (i.e. [14,15]). The VMS-ROM formulation should be valid for any basis regardless of the technique used to obtain it.

References

[1] T.J. Hughes, G.R. Feijóo, L. Mazzei, J.-B. Quincy, The variational multiscale method—a paradigm for computational mechanics, Comput. Methods Appl. Mech. Engrg. 166 (1–2) (1998) 3–24, http://dx.doi.org/10.1016/S0045-7825(98)00079-6.

[2] R. Codina, S. Badia, J. Baiges, J. Principe, Variational multiscale methods in computational fluid dynamics, in: Encyclopedia of Computational Mechanics, second ed., John Wiley & Sons, Ltd, Chichester, UK, 2018, pp. 1–28, http://dx.doi.org/10.1002/9781119176817.ecm2117.

[3] C.E. Wasberg, T. Gjesdal, B.A.P. Reif, O. Andreassen, Variational multiscale turbulence modelling in a high order spectral element method, J. Comput. Phys. 228 (19) (2009) 7333–7356, http://dx.doi.org/10.1016/j.jcp.2009.06.029.

[4] S. Marras, J. Kelly, F. Giraldo, M. Vázquez, Variational multiscale stabilization of high-order spectral elements for the advection–diffusion equation, J. Comput. Phys. 231 (21) (2012) 7187–7213, http://dx.doi.org/10.1016/j.jcp.2012.06.028.

[5] R. Codina, J. Principe, O. Guasch, S. Badia, Time dependent subscales in the stabilized finite element approximation of incompressible flow problems, Comput. Methods Appl. Mech. Engrg. 196 (21–24) (2007) 2413–2430, http://dx.doi.org/10.1016/j.cma.2007.01.002.

[6] R. Codina, Analysis of a stabilized finite element approximation of the Oseen equations using orthogonal subscales, Appl. Numer. Math. 58 (3) (2008) 264–283, http://dx.doi.org/10.1016/j.apnum.2006.11.011.

[7] J. Baiges, R. Codina, S. Idelsohn, Reduced-order subscales for POD models, Comput. Methods Appl. Mech. Engrg. 291 (2015) 173–196, http://dx.doi.org/10.1016/j.cma.2015.03.020.

[8] R. Codina, Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods, Comput. Methods Appl. Mech. Engrg. 190 (2000) 1579–1599, http://dx.doi.org/10.1016/S0045-7825(00)00254-1.

[9] R. Codina, Stabilized finite element approximation of transient incompressible flows using orthogonal subscales, Comput. Methods Appl. Mech. Engrg. 191 (39–40) (2002) 4295–4321, http://dx.doi.org/10.1016/S0045-7825(02)00337-7.

[10] R. Reyes, R. Codina, J. Baiges, S. Idelsohn, Reduced order models for thermally coupled low mach flows, Adv. Model. Simul. Eng. Sci. 5 (1) (2018) 28, http://dx.doi.org/10.1186/s40323-018-0122-7.

[11] K. Carlberg, C. Bou-Mosleh, C. Farhat, Efficient non-linear model reduction via a least-squares Petrov-Galerkin projection and compressive tensor approximations, Internat. J. Numer. Methods Engrg. 86 (2) (2011) 155–181, http://dx.doi.org/10.1002/nme.3050.

[12] Y. Saad, Iterative Methods for Sparse Linear Systems, Society for Industrial and Applied Mathematics, SIAM, Philadelphia, Pennsylvania, 2003, pp. 421–426, http://dx.doi.org/10.1137/1.9780898718003.

[13] J.L. Lumley, Coherent structures in turbulence, in: Transition and Turbulence, Academic Press, INC., 1981, pp. 215–242, http://dx.doi.org/10.1016/B978-0-12-493240-1.50017-X.

[14] P.J. Schmid, Dynamic mode decomposition of numerical and experimental data, J. Fluid Mech. 656 (693) (2010) 5–28, http://dx.doi.org/10.1017/S0022112010001217.

[15] F. Chinesta, A. Ammar, A. Leygue, R. Keunings, An overview of the proper generalized decomposition with applications in computational rheology, J. Non-Newton. Fluid Mech. 166 (11) (2011) 578–592, http://dx.doi.org/10.1016/j.jnnfm.2010.12.012.

[16] L. Sirovich, Turbulence and the dynamics of coherent structures. I. Coherent structures, Quart. Appl. Math. 45 (3) (1987) 561–571, http://dx.doi.org/10.1090/qam/910462.

[17] L. Sirovich, Turbulence and the dynamics of coherent structures. II. Symmetries and transformations, Quart. Appl. Math. 45 (3) (1987) 573–582, http://dx.doi.org/10.1090/qam/910463.

[18] L. Sirovich, Turbulence and the dynamics of coherent structures. III. Dynamics and scaling, Quart. Appl. Math. 45 (3) (1987) 583–590, http://dx.doi.org/10.1090/qam/910464.

[19] N. Aubry, On the hidden beauty of the proper orthogonal decomposition, Theor. Comput. Fluid Dyn. 2 (5–6) (1991) 339–352, http://dx.doi.org/10.1007/BF00271473.

[20] K. Kunisch, S. Volkwein, Galerkin proper orthogonal decomposition methods for parabolic problems, Numer. Math. 90 (2001) 117–148, http://dx.doi.org/10.1007/s002110100282.

[21] K. Kunisch, S. Volkwein, Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics, SIAM J. Numer. Anal. 40 (2) (2002) 492–515, http://dx.doi.org/10.1137/S0036142900382612.

[22] M. Bergmann, C.H. Bruneau, A. Iollo, Enablers for robust POD models, J. Comput. Phys. 228 (2) (2009) 516–538, http://dx.doi.org/10.1016/j.jcp.2008.09.024.

[23] T. Iliescu, Z. Wang, Variational multiscale proper orthogonal decomposition: convection-dominated convection-diffusion-reaction equations, Math. Comp. 82 (283) (2013) 1357–1378, http://dx.doi.org/10.1090/S0025-5718-2013-02683-X.

[24] M. Fahl, Computation of POD basis functions for fluid flows with lanczos methods, Math. Comput. Modelling 34 (1–2) (2001) 91–107, http://dx.doi.org/10.1016/S0895-7177(01)00051-6.

[25] S. Volkwein, Proper orthogonal decomposition: theory and reduced-order modelling, in: Lecture Notes, 2013.

[26] S. Giere, Numerical and Analytical Aspects of POD-Based Reduced-Order Modeling in Computational Fluid Dynamics (Ph.D. thesis), Freien Universität Berlin, 2016.

[27] S. Badia, R. Codina, On a multiscale approach to the transient Stokes problem: Dynamic subscales and anisotropic space–time discretization, Appl. Math. Comput. 207 (2) (2009) 415–433, http://dx.doi.org/10.1016/j.amc.2008.10.059.

[28] Z. Wang, I. Akhtar, J. Borggaard, T. Iliescu, Two-level discretizations of nonlinear closure models for proper orthogonal decomposition, J. Comput. Phys. 230 (1) (2011) 126–146, http://dx.doi.org/10.1016/j.jcp.2010.09.015.
[29] M. Barrault, Y. Maday, N.C. Nguyen, A.T. Patera, An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations, C. R. Acad. Sci., Paris 339 (9) (2004) 667–672, http://dx.doi.org/10.1016/j.crma.2004.08.006.
[30] N.C. Nguyen, A.T. Patera, J. Peraire, A 'best points' interpolation method for efficient approximation of parametrized functions, Internat. J. Numer. Methods Engrg. 73 (4) (2008) 521–543, http://dx.doi.org/10.1002/nme.2086.
[31] D. Ryckelynck, F. Vincent, S. Cantournet, Multidimensional a priori hyper-reduction of mechanical models involving internal variables, Comput. Methods Appl. Mech. Engrg. 225–228 (2012) 28–43, http://dx.doi.org/10.1016/j.cma.2012.03.005.
[32] J. Baiges, R. Codina, S. Idelsohn, Explicit reduced-order models for the stabilized finite element approximation of the incompressible Navier-Stokes equations, Internat. J. Numer. Methods Fluids 72 (12) (2013) 1219–1243, http://dx.doi.org/10.1002/fld.3777.
[33] J.A. Hernández, M.A. Caicedo, A. Ferrer, Dimensional hyper-reduction of nonlinear finite element models via empirical cubature, Comput. Methods Appl. Mech. Engrg. 313 (2017) 687–722, http://dx.doi.org/10.1016/j.cma.2016.10.022.
[34] S. Giere, T. Iliescu, V. John, D. Wells, SUPG reduced order models for convection-dominated convection-diffusion-reaction equations, Comput. Methods Appl. Mech. Engrg. 289 (2015) 454–474, http://dx.doi.org/10.1016/j.cma.2015.01.020.
[35] B. McLaughlin, J. Peterson, M. Ye, Stabilized reduced order models for the advection–diffusion–reaction equation using operator splitting, Comput. Math. Appl. 71 (11) (2016) 2407–2420, http://dx.doi.org/10.1016/j.camwa.2016.01.032.
[36] B. Kragel, Streamline Diffusion POD Models in Optimization (Ph.D. thesis), University of Trier, 2005.
[37] T. Iliescu, Z. Wang, Variational multiscale proper orthogonal decomposition: Navier-Stokes equations, Numer. Methods Partial Differential Equations 30 (2) (2014) 641–663, http://dx.doi.org/10.1002/num.21835.
[38] M. Azaïez, T. Chacón Rebollo, S. Rubino, Streamline derivative projection-based POD-ROM for convection-dominated flows. Part I : Numerical analysis, 2017, arXiv preprint, http://arxiv.org/abs/1711.09780.
[39] S. Rubino, A streamline derivative POD-ROM for advection-diffusion-reaction equations, ESAIM Proc. Surv. 64 (June) (2018) 121–136, http://dx.doi.org/10.1051/proc/201864121.
[40] D. Amsallem, M.J. Zahr, C. Farhat, Nonlinear model order reduction based on local reduced-order bases, Internat. J. Numer. Methods Engrg. 92 (10) (2012) 891–916, http://dx.doi.org/10.1002/nme.4371.
[41] M. Balajewicz, E.H. Dowell, Stabilization of projection-based reduced order models of the Navier–Stokes, Nonlinear Dynam. 70 (2) (2012) 1619–1632, http://dx.doi.org/10.1007/s11071-012-0561-5.
[42] I. Kalashnikova, B. van Bloemen Waanders, S. Arunajatesan, M.F. Barone, Stabilization of projection-based reduced order models for linear time-invariant systems via optimization-based eigenvalue reassignment, Comput. Methods Appl. Mech. Engrg. 272 (2014) 251–270, http://dx.doi.org/10.1016/j.cma.2014.01.011.
[43] D. Wells, Z. Wang, X. Xie, T. Iliescu, An evolve-then-filter regularized reduced order model for convection-dominated flows, Internat. J. Numer. Methods Fluids 84 (10) (2017) 598–615, http://dx.doi.org/10.1002/fld.4363.
[44] S. Ganesan, L. Tobiska, Stabilization by local projection for convection–diffusion and incompressible flow problems, J. Sci. Comput. 43 (3) (2010) 326–342, http://dx.doi.org/10.1007/s10915-008-9259-8.
[45] P. Knobloch, G. Lube, Local projection stabilization for advection-diffusion-reaction problems: One-level vs. two-level approach, Appl. Numer. Math. 59 (12) (2009) 2891–2907, http://dx.doi.org/10.1016/j.apnum.2009.06.004.
[46] E. Castillo, R. Codina, Dynamic term-by-term stabilized finite element formulation using orthogonal subgrid-scales for the incompressible Navier–Stokes problem, Comput. Methods Appl. Mech. Engrg. 349 (July) (2019) 701–721, http://dx.doi.org/10.1016/j.cma.2019.02.041.
[47] R. Codina, A stabilized finite element method for generalized stationary incompressible flows, Comput. Methods Appl. Mech. Engrg. 190 (20–21) (2001) 2681–2706, http://dx.doi.org/10.1016/S0045-7825(00)00260-7.
[48] R. Codina, J. Principe, Dynamic subscales in the finite element approximation of thermally coupled incompressible flows, Internat. J. Numer. Methods Fluids 54 (6–8) (2007) 707–730, http://dx.doi.org/10.1002/fld.1481.
[49] S. Jovic, An Experimental Study of a Separated/Reattached Flow Behind a Backward-Facing Step. Re = 37000, Nasa Technical Memorandum, (April) National Aeronautics and Space Administration, Ames Research Center, 1996.