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

Ricardo Reyes\textsuperscript{a}, Ramon Codina\textsuperscript{a,b,}\textsuperscript{*}

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

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 subcales, non-linearity in the subscale 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 discretised 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.

\textsuperscript{*}Corresponding author

Email addresses: ricardo.reyes@upc.edu (Ricardo Reyes), ramon.codina@upc.edu (Ramon Codina)

Preprint submitted to Computer Methods in Applied Mechanics and Engineering July 9, 2019
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 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

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 Scale (OSGS); 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 OSGS in both the ROM and the FOM,
figure 1 shows a comparison between FOM and ROM resolved scale —\( Y_h \) and
\( Y_r \) — and subscale —\( \tilde{Y} \) and \( \tilde{\tilde{Y}} \) — spaces. These subspaces are such that
\( Y = Y_h \oplus \tilde{Y} = Y_r \oplus \tilde{\tilde{Y}} \), where \( Y \) is the space of functions where the problem is defined, assumed to be a Hilbert
space.

Let us remark that \( \tilde{\tilde{Y}} \) is the complementary in \( Y \) of \( Y_r \), not the complementary in
\( Y_h \), 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 —\( \mathcal{R} \),
an operator applied to the test function —\( \mathcal{P} \), and a matrix of numerical parameters —\( \tau \). To
construct \( \mathcal{R} \) and \( \mathcal{P} \) we follow a VMS approach that includes the necessary conditions
to define the OSGS as done in [8]. Furthermore, to define \( \tau \) 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 in FE-ROMs as in [10], where the spectral
basis is obtained using a FE-FOM and then projected onto a discretised 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
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 $\Gamma$ and in a time interval from zero to a final time $t_f$, that consists in finding a vector function $y(x, t)$ of $n$ components such that

$$\mathcal{M}(y) \partial_t y + \mathcal{L}(y; y) = f, \quad \text{in } \Omega, \quad t \in [0, t_f],$$

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

Additionally, the initial and boundary conditions are set as

$$y(x, 0) = y_0(x) \quad \text{in } \Omega, \quad t = 0,$$

$$y = y_D \quad \text{on } \Gamma_D, \quad t \in [0, t_f[,$$

$$\mathcal{F}(y; y) = t_N \quad \text{on } \Gamma_N, \quad t \in [0, t_f[,$$

VMS-FOM using FEs. In section 5 we formulate the VMS-ROM, the way to approximate the subcales and present an analysis of their behaviour to define the numerical parameters $\tau$. 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.
where \( y_0 \) is the prescribed initial condition, \( \Gamma_D \) and \( \Gamma_N \) are a partition of \( \Gamma \), \( y_D \) is the prescribed Dirichlet boundary condition, \( \mathcal{F}(y,z) := n_i K_i(y) \partial_z y - n_i A_i(y) z \) is a flux operator, \( n \) is the normal to \( \Gamma \) whose \( i \)-th component is \( n_i \), and \( t_N \) is the prescribed Neumann boundary condition.

Let us denote by \( \mathcal{Y} \) the space of functions in \( \Omega \) where \( y \) is sought for each time \( t \), incorporating Dirichlet boundary conditions, and let \( \mathcal{Y}_0 \) be its corresponding space of time independent test functions that vanish on \( \Gamma_D \), both assumed to be subspaces of \( L^2(\Omega)^n \). Let also \( \langle \cdot, \cdot \rangle \) be the integral over \( \Omega \) of the product of two functions, assumed to be well defined, and let \( \langle \cdot, \cdot \rangle \) be the \( L^2 \)-inner product in \( \Omega \). We can write the variational form of the problem as finding \( y : [0,t_1] \rightarrow \mathcal{Y} \) such that

\[
\begin{align*}
(\mathcal{M}(y) \partial_t y, v) + (\mathcal{L}(y, z), v) &= (f, v), \quad t \in [0,t_1], \quad \forall v \in \mathcal{Y}_0, \\
(y, v) &= (y_0, v), \quad t = 0, \quad \forall v \in \mathcal{Y}_0.
\end{align*}
\]

Now, denoting by \( \langle \cdot, \cdot \rangle_{\Gamma_N} \) the integral of the product of two functions defined on \( \Gamma_N \) and defining the forms

\[
\begin{align*}
B(y,z,v) &= \langle A_i(y) \partial_z z, v \rangle - \langle z, \partial_i (A_i(y)^T v) \rangle + \langle S(y) z, v \rangle \\
&\quad + \langle K_i(y) \partial_i z, v \rangle, \\
L(v) &= \langle f, v \rangle + \langle t_{\Lambda}(v), v \rangle_{\Gamma_N},
\end{align*}
\]

we can write an equivalent problem to equation (2): find a vector function \( y : [0,t_1] \rightarrow \mathcal{Y} \) such that

\[
(\mathcal{M}(y) \partial_t y, v) + B(y,z,v) = L(v), \quad \forall v \in \mathcal{Y}_0, \quad t \in [0,t_1].
\]

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 \( \mathcal{Y}_r \subset \mathcal{Y} \) of dimension \( r \) that has a complete orthonormal basis \( \{\phi^k\}_{k=1}^r \), each vector being an array of \( n \) components. We can compute the best \( L^2 \)-approximation of \( y \) in \( \mathcal{Y}_r \) at a given time \( t \in [0,t_1] \) by \( \sum_{k=1}^r (y, \phi^k) \phi^k \).

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 equation (3) yields an ordinary differential equation of the form

\[
M\dot{y} + Ky = f.
\]

\[4\]
where \( y \) is a time dependent array of \( Nn \) components and \( \dot{y} \) denotes the time differentiation, \( M \) and \( K \) are the resulting mass and stiffness matrices and \( f \) is the resulting force vector. After discretising 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 \( \phi^{k,a} \) is the \( a \)-th vector component of the \( k \)-th basis vector, with \( a = 1, \ldots, N \) and \( k = 1, \ldots, r \), we may identify \( \phi^k \) with the function

\[
\phi^k(x) = \sum_{a=1}^{N} \varphi^a(x) \phi^{k,a}, \quad k = 1, \ldots, r, \quad \forall x \in \Omega, \tag{5}
\]

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

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 \( \mathcal{M} \)-inner product given by

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

and likewise, if \( \phi^k \) are ordered arrays

\[
\phi^{k\top} \mathcal{M} \phi^l = \delta^{kl} \quad k, l = 1, \ldots, r, \tag{6b}
\]

where \( \mathcal{M} \) is the matrix in equation (4).

2. It satisfies homogeneous Dirichlet boundary conditions when calculated using a mean centered-trajectory method (see remark 1 below).

3. It has an optimal approximation when truncated (Eckart-Young-Mirsky theorem).

In appendix Appendix A 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

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

where \( \bar{y} \) 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 equation (3) we denote by \( \mathcal{T}_h = \{ K \} \) a partition of the domain \( \Omega \), with \( h = \max \{ h_K = \text{diam}(K) | K \in \mathcal{T}_h \} \) the
We chose to work within the same framework. We follow the survey on VMS-FE methods to stabilize the FOM without affecting the VMS-ROM stabilization proposed in this work, where the operator $\delta_t \mathbf{y}$ indicates a discrete temporal derivatives over functions $\mathbf{y}$ at time $t^j$ and the coefficients $\alpha^l$ 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, $t_f/\delta t$.

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

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

which we call the Galerkin form of the FOM.

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

$$\mathbf{M}(\mathbf{y}_h^j)\delta_t \mathbf{y}_h^j + B(\mathbf{y}_h^j, \mathbf{v}_h) + (\tau_h(\mathbf{y}_h)\mathcal{R}_h(\mathbf{y}_h, \mathbf{v}_h), \mathcal{P}_h(\mathbf{y}_h, \mathbf{v}_h)) = L(\mathbf{v}_h), \quad (9)$$

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

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 discretisation— $\mathcal{Y}_h$, and a continuous one $\mathcal{Y}$, so that $\mathcal{Y} = \mathcal{Y}_h \oplus \mathcal{Y}$. The unknown and the test functions are accordingly split as $\mathbf{y} = \mathbf{y}_h + \mathbf{y}$ and $\mathbf{v} = \mathbf{v}_h + \mathbf{v}$, respectively. Then, once discretised in time the continuous problem in equation (3) expands into two: find $\{\mathbf{y}_h^j\} \subset \mathcal{Y}_h$ and $\{\mathbf{y}_h^j\} \subset \mathcal{Y}$, such that

$$\mathbf{M}(\mathbf{y}_h^j)\delta_t \mathbf{y}_h^j + B(\mathbf{y}_h^j, \mathbf{v}_h) + B(\mathbf{y}; \mathbf{y}_h, \mathbf{v}_h)$$

$$+ B(\mathbf{y}; \mathbf{y}_h, \mathbf{v}_h) = L(\mathbf{v}_h), \quad \forall \mathbf{v}_h \in \mathcal{Y}_h, \quad \text{at } t^j, \; j = 1, 2, \ldots, \quad (10a)$$
\[ (\mathcal{M}(y)\delta_t y, \tilde{v}) + (\mathcal{M}(y)\delta_t \tilde{y}, \tilde{v}) + B(y; y_h, \tilde{v}) \]
\[ + B(y; \tilde{y}, \tilde{v}) = L(\tilde{v}), \quad \forall \tilde{v} \in \tilde{\mathcal{Y}}, \text{ at } t^j, \ j = 1, 2, \ldots \]  
(10b)

This corresponds to take \( \tilde{y} = y \) in equation (9). We refer to functions in \( \mathcal{Y}_h \) as the FOM resolved scales and to functions in \( \mathcal{Y} \) as the FOM SubGrid Scales (SGS) or simply subscales.

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

\[ (\mathcal{M}(y)\delta_t y_h, v_h) + (\mathcal{M}(y)\delta_t \tilde{y}, v_h) + B(y; y_h, v_h) + \sum_K \langle \tilde{y}, \mathcal{L}^*(y; v_h) \rangle_K \]
\[ + \sum_K \langle \tilde{y}, \mathcal{F}^*(y; v_h) \rangle_{\partial K} = L(v_h), \quad \forall v_h \in \mathcal{Y}_h, \]  
(11a)

\[ \sum_K \langle \mathcal{M}(y)\delta_t \tilde{y}, \tilde{v} \rangle_K + \sum_K \langle \mathcal{L}(y, \tilde{y}), \tilde{v} \rangle_K + \sum_K \langle \mathcal{F}(y; \tilde{y}), \tilde{v} \rangle_{\partial K} \]
\[ = \sum_K \langle r(y; y_h), \tilde{v} \rangle_K, \quad \forall \tilde{v} \in \tilde{\mathcal{Y}}, \]  
(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 \( K \in \mathcal{T}_h \) and its boundary, respectively.

### 4.2. FOM subscales

Considering that the subscale problem in equation (11b) is infinite dimensional, the approximation \( \mathcal{L}(y; \tilde{y}) = \tau_K^{-1}(y)\tilde{y} \) in each element \( K \) needs to be introduced to make the method computationally feasible, where \( \tau_K \) is the matrix of stabilization parameters that approximates the inverse of the differential operator \( \mathcal{L} \) on each element \( K \). The definition of \( \tau_K \) can be achieved following different methods which leads to slightly different sets of parameters. In this work we follow a 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 equation (11) vanish and we arrive to

\[ \mathcal{M}(y)\delta_t \tilde{y} + \tau_K^{-1}(y)\tilde{y} = r(y; y_h) + \tilde{v}^\perp, \quad \text{in } K \in \mathcal{T}_h, \text{ at } t^j, \ j = 1, 2, \ldots \]  
(12)

where \( \tilde{v}^\perp \) is any function that satisfies the condition

\[ \sum_K \langle \tilde{v}^\perp, \tilde{v} \rangle_K = 0, \quad \forall \tilde{v} \in \tilde{\mathcal{Y}}, \]  
(13)

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

$$
\mathcal{M}(y)\delta_t \bar{y} + \tau_K^{-1}(y)\bar{y} = \tilde{\Pi}(r(y; y_h)), \quad \text{in } K \in \mathcal{T}_h, \quad \text{at } t^j, \quad j = 1, 2, \ldots ,
$$

(14)

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

**Remark 2.** Another way to impose this condition is by using the weighted inner product $\langle \cdot, \cdot \rangle_r = \sum_K \langle \tau_K, \cdot \rangle_K$ and the associated projection $\tilde{\Pi}_r^\perp$ onto $\mathcal{Y}_{r^\perp}$. This way, projecting equation (12) onto $\mathcal{Y}_{r^\perp}$ and using equation (13), leads to $\tilde{\mathbf{v}}^\perp = -\tilde{\Pi}_r^\perp (r(y; y_h))$. Further details on this approximation can be found in [6].

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

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. $\delta_t \bar{y} = 0$ — and yielding the subscale approximation $\tau_K^{-1}(y)\bar{y} = \mathcal{R}_h(y; y_h)$, and dynamic subscales, obtained when such time derivative is considered, approximating the subscales as the solution of $\mathcal{M}\delta_t \bar{y} + \tau_K^{-1}(y)\bar{y} = \mathcal{R}_h(y; y_h)$. Here, an effective stabilization parameter can be defined as $\tau_K^{-1}(y) := \alpha_0^\delta t^{-1} \mathcal{M} + \tau_K^{-1}(y)$.

**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 SGS equations when using a second order scheme for the FE equations.

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

$$
(\mathcal{M}(y)\delta_t y_h, v_h) + B_h (y; y_h, v_h) = L_h (y; v_h), \quad \forall v_h \in \mathcal{B}_h, \quad \text{at } t^j, \quad j = 1, 2, \ldots ,
$$

(15)

with the forms $B_h (y; y_h, v_h) = B (y; y_h, v_h) + B_s (y; y_h, v_h)$ and $L_h (y; v_h) = L (v_h) + L_s (y; v_h)$, where the stabilization terms $B_s (y; y_h, v_h)$ and $L_s (y; v_h)$ are defined for each proposed case in table 1 and table 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 OSGS.
Table 1: FOM $B_h(y; y_h, v_h)$ defined for dynamic and quasi-static ASGS and OSGS.

<table>
<thead>
<tr>
<th></th>
<th>Dynamic</th>
<th>Quasi-static</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASGS</td>
<td>$-\sum_K (M(y)\delta y_h + L(y; y_h), \tau_{K,i}^\top (y, v_h)) + [I - \tau_{K,i}^\top v_h]_K$</td>
<td>$-\sum_K (M(y)\delta y_h + L(y; y_h), \tau_{K,i}^\top (y, v_h))_K$</td>
</tr>
<tr>
<td>OSGS</td>
<td>$-\sum_K (\Pi_1^\top (L(y; y_h)), \tau_{K,i}^\top (y, v_h))_K$</td>
<td>$-\sum_K (\Pi_1^\top (L(y; y_h)), \tau_{K,i}^\top (y, v_h))_K$</td>
</tr>
</tbody>
</table>

Table 2: FOM $L_s(y; v_h)$ defined for dynamic and quasi-static ASGS and OSGS.

<table>
<thead>
<tr>
<th></th>
<th>Dynamic</th>
<th>Quasi-static</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASGS</td>
<td>$-\sum_K (f + \delta t^{-1} M(y)\tilde{y}, \tau_{K,i}^\top (y, v_h))_K$</td>
<td>$-\sum_K (f, \tau_{K,i}^\top (y, v_h))_K$</td>
</tr>
<tr>
<td>OSGS</td>
<td>$-\sum_K (\Pi_1^\top (f + \delta t^{-1} M(y)\tilde{y}), \tau_{K,i}^\top (y, v_h))_K$</td>
<td>$-\sum_K (\Pi_1^\top (f), \tau_{K,i}^\top (y, v_h))_K$</td>
</tr>
</tbody>
</table>

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

Remark 5. The practical way to compute the projection $\Pi_h^\perp$ is to compute the projection $\Pi_h$ and then use $\Pi_h = I - \Pi_h$. Therefore, by denoting as $z_h = \Pi_h (L(y; y_h) - f)$, the problem in equation (15) for dynamic OSGS— becomes: find $[y_h, z_h]$ in $\mathcal{Y}_h \times \mathcal{Y}_h$ such that

\[
(y_h, z_h) = -\sum_K (L(y; v_h) - \sum_K (\mathcal{L}(y; y_h) - z_h, \tau_{K,i} (y) \mathcal{L}^*(y, v_h)))_K,
\]

and

\[
(z_h, \zeta_h) = -\sum_K (\mathcal{L}(y; y_h) - f, \zeta_h)_K, \tag{16b}
\]

which must hold for all $[v_h, \zeta_h]$ in $\mathcal{Y}_h \times \mathcal{Y}_h$. To approximate equation (16b) we follow the block iteration algorithm in [6], having $[\Pi_h (L(y; y_h))]^{i+1} = [L(y; y_h)]^{i+1} - [\Pi_h (\mathcal{L}(y; y_h))]^i$, 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 $\mathcal{Y}_r \subset \mathcal{Y}$. We can write the variational ROM of problem (3) as: find $\{y^j_r\} \subset \mathcal{Y}_r$, such that

\[
(M(y)\delta y_r, v_r) + B(y; y_r, v_r) = L(v_r), \quad \forall v_r \in \mathcal{Y}_r, \quad \text{at} \ t^j, \ j = 1, 2, \ldots, \tag{17}
\]

which we describe as the Galerkin-ROM.
As the ROM approximation space is a subset of the FOM space — $\mathcal{Y}_r \subset \mathcal{Y}_h$ — the instabilities described in section 4 are inherited. Hence, a stabilization for the ROM is also necessary. By implementing the same VMS method in equation (17) we can write a stabilized ROM in the form of equation (9) as

$$\mathbf{E} \frac{\partial \mathbf{y}}{\partial t} + \mathbf{B} \mathbf{y} = \mathbf{F},$$

for all $\mathbf{y} \in \mathcal{Y}_r$, where, similarly to the FOM case, $\mathbf{y}$ is an approximation to $\mathbf{y}$ that may take into account the subscales (see below) or not. We will simply write $\mathbf{y}$ 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 $\mathcal{Y}_r$ and the space of the ROM-SGS $\mathcal{Y}_h$, so that $\mathcal{Y} = \mathcal{Y}_r \oplus \mathcal{Y}_h$, with the unknown and the test functions split as $\mathbf{y} = \mathbf{y}_r + \hat{\mathbf{y}}$ and $\mathbf{v} = \mathbf{v}_r + \hat{\mathbf{v}}$, respectively. Consequently, the ROM problem in equation (17) also expands into two, finding $\{\mathbf{y}_r\} \subset \mathcal{Y}_r$ and $\{\hat{\mathbf{y}}\} \subset \mathcal{Y}_h$, such that

$$\begin{align*}
(\mathbf{M}(\mathbf{y})\delta_1 \mathbf{y}_r, \mathbf{v}_r) + \mathbf{B} (\mathbf{y}_r, \mathbf{v}_r) &= \mathbf{L}(\mathbf{v}_r), \quad \forall \mathbf{v}_r \in \mathcal{Y}_r, \text{ at } t^j, \quad j = 1, 2, \ldots, \\
(\mathbf{M}(\mathbf{y})\delta_1 \hat{\mathbf{y}}, \hat{\mathbf{v}}) + \mathbf{B} (\mathbf{y}_r, \hat{\mathbf{v}}) &= \mathbf{L}(\hat{\mathbf{v}}), \quad \forall \hat{\mathbf{v}} \in \mathcal{Y}_h, \text{ at } t^j, \quad j = 1, 2, \ldots.
\end{align*}$$

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

$$\begin{align*}
(\mathbf{M}(\mathbf{y})\delta_1 \mathbf{y}_r, \mathbf{v}_r) + (\mathbf{M}(\mathbf{y})\delta_1 \hat{\mathbf{y}}, \mathbf{v}_r) + \mathbf{B} (\mathbf{y}_r, \mathbf{v}_r) + \sum_K \langle \hat{\mathbf{y}}, \mathcal{L}_r^{-1}(\mathbf{y}; \mathbf{v}_r) \rangle_K \\
+ \sum_K \langle \hat{\mathbf{y}}, \mathcal{L}^{-1}_r(\mathbf{y}; \mathbf{v}_r) \rangle_K \\
= \mathbf{L}(\mathbf{v}_r), \quad \forall \mathbf{v}_r \in \mathcal{Y}_r, \text{ at } t^j, \quad j = 1, 2, \ldots.
\end{align*}$$

5.2. ROM subscales

By following the same supposition of continuous fluxes and using the same approximation for the spatial operator $-\mathcal{L}(\mathbf{y}, \hat{\mathbf{y}}) \approx \mathcal{L}_K^{-1}(\mathbf{y})\hat{\mathbf{y}}$ in each element $K$ — as in section 4.2, we can write an equation similar to equation (12) for the ROM-SGS:

$$\mathbf{M}(\mathbf{y})\delta_1 \hat{\mathbf{y}} + \mathcal{L}_K^{-1}(\mathbf{y})\hat{\mathbf{y}} = \mathbf{r}(\mathbf{y}; \mathbf{y}_r) + \bar{\mathbf{v}}^\perp \quad \text{in } K \in \mathcal{Y}_h, \quad \text{at } t^j, \quad j = 1, 2, \ldots.$$
Likewise, to complete the approximation of the subscales, the term $\tilde{\nu}^\perp$ satisfies a similar condition as in equation (13) $-\sum_K(\tilde{\nu}^\perp, \tilde{\nu})_K = 0, \forall \tilde{\nu} \in \tilde{\mathcal{Y}}$, and therefore also defined by the choice of the space of ROM subscales $\mathcal{Y}$ and its orthogonal complement $\mathcal{Y}^\perp$. Thus, denoting the $L^2$ projection onto $\mathcal{Y}^\perp$ as $\tilde{\Pi}^\perp$ and defining $\tilde{\nu}^\perp = -\tilde{\Pi}^\perp(r(y; y_r))$, we can rewrite equation (21) as

$$\mathcal{M}(y)\delta_{t}\tilde{y} + \tau_K^{-1}(y)\tilde{y} = \tilde{\Pi}(r(y; y_r)), \quad \text{in } K \in \mathcal{P}_h, \quad \text{at } t^j, \; j = 1, 2, \ldots, \quad (22)$$

where similarly to equation (14) for the FOM-SGS, $\tilde{\Pi} = I - \tilde{\Pi}^\perp$ is the projection onto the subscale space $\mathcal{Y}$ and $I$ is the identity.

**Algebraic and orthogonal subscales.** Analogously to the FOM subscales, for the choice of the space $\mathcal{Y}$ and consequently for the definition of the projection $\tilde{\Pi}$, we get the same two different approximation methods for the subscales: ROM-ASGS and ROM-OSGS. For the ASGS formulation $\tilde{\nu}^\perp = 0$ and $\tilde{\Pi} = I$, yielding $\mathcal{R}_r(y; y_r) = r(y; y_r)$, while for the OSGS formulation $\mathcal{Y} = \mathcal{Y}_r \perp$, $\sum_K(v_r, \tilde{\nu}) = 0$ and $\tilde{\Pi} = \Pi^\perp = I - \Pi_r$, yielding $\mathcal{R}_r(y; y_r) = \Pi^\perp(f - \mathcal{L}(y; y_r))$.

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

By defining $\mathcal{P}_r(y, v_r) = \mathcal{L}^\perp(y, v_r)$, and choosing $\tau_r(y)$ as $\tau_K(y)$ or $\tau_K^r(y)$, and $\mathcal{R}_r(y; y_r)$ as $r(y; y_r)$ or $\Pi^\perp(f - \mathcal{L}(y; y_r))$, we can obtain a general stabilized approximation for $y_r$ of the form of equation (18). Moreover, we can rewrite this equation as

$$(\mathcal{M}(y)\delta_t y_r, v_r) + B_r(y; y_r, v_r) = L_r(v_r), \quad \forall v_r \in \mathcal{Y}_r, \quad \text{at } t^j, \; j = 1, 2, \ldots, \quad (23)$$

with the forms $B_r(y; y_r, v_r) = B(y; y_r, v_r) + B_s(y; y_r, v_r)$ and $L_r(y; v_r) = L(v_r) + L_s(y; v_r)$, where, using a first order BDF to approximate the time derivative of the SGS, the stabilization terms $B_s(y; y_r, v_r)$ and $L_s(y; v_r)$ are defined for each proposed case in table 3 and table 4, respectively.

<table>
<thead>
<tr>
<th>ASGS</th>
<th>Dynamic</th>
<th>$-\sum_K(\mathcal{M}(y)\delta_{t}y_r, \mathcal{L}(y; y_r), \tau_K \mathcal{L}^\perp(y, v_r)) + [I - \tau_K^r \mathcal{L}^\perp(y, v_r)]_K$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASGS</td>
<td>Quasi-static</td>
<td>$-\sum_K(\mathcal{M}(y)\delta_{t}y_r, \mathcal{L}(y; y_r), \tau_K \mathcal{L}^\perp(y, v_r))_K$</td>
</tr>
<tr>
<td>OSGS</td>
<td>Dynamic</td>
<td>$-\sum_K(\Pi^\perp_r(\mathcal{L}(y; y_r)), \tau_K \mathcal{L}^\perp(y, v_r))_K$</td>
</tr>
<tr>
<td>OSGS</td>
<td>Quasi-static</td>
<td>$-\sum_K(\Pi^\perp_r(\mathcal{L}(y; y_r)), \tau_K \mathcal{L}^\perp(y, v_r))_K$</td>
</tr>
</tbody>
</table>

Table 3: ROM $B_s(y; y_r, v_r)$ defined for dynamic and quasi-static ASGS and OSGS.

<table>
<thead>
<tr>
<th>ASGS</th>
<th>Dynamic</th>
<th>$-\sum_K(\mathcal{M}(y)\delta_{t}y_r, \mathcal{L}(y; y_r), \tau_K \mathcal{L}^\perp(y, v_r)) + [I - \tau_K^r \mathcal{L}^\perp(y, v_r)]_K$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASGS</td>
<td>Quasi-static</td>
<td>$-\sum_K(\mathcal{M}(y)\delta_{t}y_r, \mathcal{L}(y; y_r), \tau_K \mathcal{L}^\perp(y, v_r))_K$</td>
</tr>
<tr>
<td>OSGS</td>
<td>Dynamic</td>
<td>$-\sum_K(\Pi^\perp_r(f), \tau_K \mathcal{L}^\perp(y, v_r))_K$</td>
</tr>
<tr>
<td>OSGS</td>
<td>Quasi-static</td>
<td>$-\sum_K(\Pi^\perp_r(f), \tau_K \mathcal{L}^\perp(y, v_r))_K$</td>
</tr>
</tbody>
</table>

Table 4: ROM $L_s(y; v_r)$ defined for dynamic and quasi-static ASGS and OSGS.
The problem has to be completed with initial conditions for the subscales. We assume that \( \hat{y}^0 = 0 \), for all \( x \in \Omega \), which means that we assume for simplicity that the initial condition belongs to the space of resolvable scales.

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

The term \( \Pi^\top_r (\mathcal{L}(y; y_r) - f) \) can be computed in a way similar to the FOM term \( \Pi^\top_h (\mathcal{L}(y; y_h) - f) \), 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 — \( \mathcal{Y}_h \) for the FOM and \( \mathcal{Y}_r \) for the ROM — and consequently for the space of the subscales.

**Remark 7 (On the importance of OSGS).** An interesting feature of orthogonal SGS, that renders them particularly ‘natural’, is the following. Suppose that we have an orthonormal basis of \( \mathcal{Y}_h \) given by \( \{ \phi^k \}_{k=1}^N \), so that \( \mathcal{Y}_r = \text{span}\{ \phi^k \}_{k=1}^r \). When OSGS are used, we have the following explicit representation of the ROM-SGS:

\[
\mathcal{Y} = \text{span}\{ \phi^k \}_{k=r+1}^N \oplus \mathcal{Y}.
\]

5.3. Behaviour of the stabilization parameters

The last component of the stabilized ROM formulation is the definition of the stabilization parameter \( \tau_K(y) \). 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

\[-\nu \Delta y + a \cdot \nabla y + \sigma y = f \quad \text{in} \ \Omega,\]

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

\[-\nu \Delta \hat{y} + a \cdot \nabla \hat{y} + \sigma \hat{y} = \mathcal{R}_r(y_r) \quad \text{in} \ K \in \mathcal{T}_h, \quad (24)\]

with \( \mathcal{R}_r(y; y_r) \) defined by using either the ASGS or the OSGS 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 \( K \in \mathcal{T}_h \) as

\[
\hat{g}(k) := \int_K e^{-ik \cdot x} g(x) dx, \quad (25a)
\]

\[
\frac{\partial \hat{g}}{\partial x_j} (k) = \int_{\partial K} n_j e^{-ik \cdot x} g(x) d\Gamma_x + ik_j \hat{g}(k), \quad (25b)
\]
where $i = \sqrt{-1}$, $k = (k_1, \ldots, k_d)$ is the wave number, and $n_j$ is now the $j$-th component of the normal exterior to $K$. Since the subscales $\tilde{y}$ 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 $\phi$, whose lower energy modes have a higher spatial frequency. Considering that in equation (25b) the second term in 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

$$\frac{\partial \tilde{g}}{\partial x_j}(k) \approx ik_j \tilde{g}(k).$$

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

$$\tilde{y}(\kappa) \approx \tilde{L}(\kappa) \tilde{R}_{\tau}(\kappa), \quad \tilde{L}(\kappa) := \left( \nu \frac{\kappa^2}{h^2} + i a \cdot \frac{\kappa}{h} + \sigma \right)^{-1},$$

from which, using Plancherel’s formula and the mean value theorem, we can identify the approximation $\tau_K$ in section 5.2 as $\tilde{L}(\kappa_0)$, where $\kappa_0$ is the wave number for which the mean value theorem holds. This leads to the expression for the stabilization parameter:

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

(26)

where $c_1$ and $c_2$ are constants independent of $\nu$, $\sigma$, $a$ and $h$.

**Remark 8.** The choice of the characteristic length $h$ is based on the partition $\mathcal{T}_h$. 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, figure 2 shows the first 6 modes of the basis obtained from the convection-diffusion-reaction problem presented in section 8.1.

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.
Figure 2: First 6 modes of the basis obtained from a convection-diffusion-reaction problem embedded in a Couette flow

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, $\| y_r \|$, $y_r \in \mathcal{Y}_r$. 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 $\| y - y_r \|$, is bounded by the interpolation estimate, i.e.,

$$\| y - y_r \| \leq C E_{I,r}(h), \quad E_{I,r}(h) := \inf_{z_r \in \mathcal{Y}_r} \| y - z_r \|, \quad (27)$$

with $y$ being the solution of the continuous problem, $y_r$ the ROM solution and $C$ a generic positive constant. In general, the steps to arrive to the estimate in equation (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 $\| y - y_r \| \leq \| y - z_r \| + \| z_r - y_r \|$ for any $z_r \in \mathcal{Y}_r$. Taking the infimum for $z_r \in \mathcal{Y}_r$, 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

$$||y - y_h|| \leq CE_{I,h}(h), \quad E_{I,h}(h) := \inf_{z_h \in Y_h} ||y - z_h||.$$  \hfill (28)

The literature about the behaviour of $E_{I,h}(h)$ in equation (28) is vast. However, the estimate in equation (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 equation (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 equation (11a) with the subscales $\tilde{y}$ obtained from equation (14), with the options described in section 4.2. This leads at each time step to a matrix problem of the form

$$A_h(y_h)y_h = b_h,$$  \hfill (29)

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

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

$$A_r(y_r)y_r = b_r,$$  \hfill (30)

where now $A_r$ a matrix of size $r \times r$ and $y_r$ and $b_r$ are arrays of $r$ components.

Let $\Phi$ 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 $y_h \approx \Phi y_r$. Furthermore, if the FOM test function is taken in the ROM subspace, equation (29) yields

$$\Phi^T A_h(\Phi y_r)\Phi y_r = \Phi^T b_h.$$  \hfill (31)

This equation can be considered the FOM one projected onto the ROM subspace. It coincides with equation (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 below).

If we consider that our ROM is the projected FOM given by equation (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 $\Phi y_r$, in our problem this Petrov-Galerkin projection would lead to the algebraic problem

$$\Phi^\top \mathbf{A}_h^\top \mathbf{A}_h \Phi y_r = \Phi^\top \mathbf{A}_h^\top \mathbf{b}_h.$$  \hfill (32)

Let us comment on the need of using the Petrov-Galerkin projection in equation (32) instead of the Galerkin projection in equation (31). For linear problems we have experimentally found that they both yield virtually the same results, and therefore equation (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 (equation (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 $\bar{y} = 0$, for simplicity—by

$$y_r^{H}(x, t) := \sum_{a=1}^{N^{H}} \varphi^a(x) \sum_{k=1}^{r} \phi^{k,a} y^k(t), \quad \forall x \in \Omega, \quad t \in [0, t_f], \quad (33)$$

and taking as test functions

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

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 $\{\varphi^a(x)\}$, 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 $y_r - y_r^{H}$ is (asymptotically) smaller than this norm applied to $y - y_r$. 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 equation (23), the definitions in tables 3 and 4, and writing its stabilization part as $\langle \tau_r(y) R_r(y; y_r), P_r(y; v_r) \rangle$ we can describe our proposed VMS method with time dependent OSGS as

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

Previously proposed methods include classical 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 $R_r(y; y_r) = r(y; y_r)$, $\Pi = I$ and $P_r(y; v_r) = A_c^i(y) \partial_i v_r$, 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 $\mathcal{L}$. Next, we can identify the VMS stabilization approaches with the quasi-static ASGS stabilization described in tables 3 and 4. And finally, by defining $R_r(y; y_r) = A_c^i(y) \partial_i y_r$, $\Pi = \Pi^r_r$ and $P_r(y; v_r) = A_c^i(y) \partial_i v_r$, we can describe the term by term stabilizations as a non-residual, orthogonal and with control only over the skew-symmetric part of $\mathcal{L}$; this family of methods resembles the one described in [44; 45] for FEs.

Along with the definitions of $R_r$ and $P_r$ we can classify the matrix of stabilization parameters $\tau_r$ 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 $\tau_r$ 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 shows a comparison using the described criteria for the methods aforementioned, excluding the purely empirical ones.

<table>
<thead>
<tr>
<th>$R_r$</th>
<th>Residual</th>
<th>[22; 32; 34]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Term by term</td>
<td>[23; 37–39]</td>
<td></td>
</tr>
<tr>
<td>Orthogonal</td>
<td>[23; 37–39]</td>
<td></td>
</tr>
<tr>
<td>Non-Orthogonal</td>
<td>[22; 32; 34]</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$P_r$</th>
<th>Complete $\mathcal{L}^*$</th>
<th>[22; 32]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Convective term</td>
<td>[23; 34; 37–39]</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$\tau_r$</th>
<th>$\tau_h$</th>
<th>[22; 32; 34; 39]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Empirical</td>
<td>[23; 37; 38]</td>
<td></td>
</tr>
</tbody>
</table>

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

In the numerical examples in section 8 we include a comparison between the OSGS, ASGS and a term by term OSGS 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 behaviour of the proposed stabilization method. The first example is a convection-diffusion-reaction problem used to illustrate the behaviour 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 SVD described in Appendix A, the domain discretisation 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 behaviour of the basis modes (figure 2). It consist in a dynamic two dimensional convection-diffusion-reaction problem stated as:

\[
\partial_t y - \nu \Delta y + a \cdot \nabla y + \sigma y = 0,
\]

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 figure 3.

![Figure 3: Convection-diffusion-reaction problem](image)

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.

The ROM is stabilized using dynamic OSGS and the stabilization parameter defined in equation (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. Figures 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 \( \mathbf{u} \) and the pressure \( p \) of an incompressible flow are solution of the incompressible Navier-Stokes equations

\[
\begin{align*}
\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} - \nu \Delta \mathbf{u} + \nabla p &= \mathbf{f} \quad \text{in } \Omega, \quad t \in [0, t_f], \\
\nabla \cdot \mathbf{u} &= 0 \quad \text{in } \Omega, \quad t \in [0, t_f],
\end{align*}
\]

where \( \nu \) is the kinematic viscosity and \( \mathbf{f} \) a vector of body forces. Initial and boundary conditions are set as

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

where \( \Gamma_D \) and \( \Gamma_N \) 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 $\mathcal{Y}$ be the space of vector functions with components in $H^1(\Omega)$ that are equal to $u_D$ on $\Gamma_D$, and $\mathcal{Y}_0$ the space of vector functions, also with components in $H^1(\Omega)$, that vanish on $\Gamma_D$. Set $\mathcal{Z} = L^2(\Omega)$. The variational form of the problem we consider is defined in the spaces $\mathcal{Y} = \mathcal{Y} \times \mathcal{Z}$ for the test solutions (for each time $t$) and $\mathcal{Y}_0 = \mathcal{Y}_0 \times \mathcal{Z}$ for the trial functions, which we shall write as $y = [u, p]$ and $v = [v, q]$, 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.

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

$$\mathcal{M}(y) = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix},$$

$$B(y; y_r, v_r) = (u \cdot \nabla u_r, v_r) + (\nu \nabla u_r - p_r I, \nabla v_r) + (\nabla \cdot u_r, q_r),$$

$$\mathcal{L}(y; y_r) = \begin{bmatrix} u \cdot \nabla u_r - \nu \Delta u_r + \nabla p_r \\ \nabla \cdot u_r \end{bmatrix},$$

$$\mathcal{L}^*(y; v_r) = \begin{bmatrix} -u \cdot \nabla v_r - \nu \Delta v_r - \nabla q_r \\ -\nabla \cdot v_r \end{bmatrix},$$

$$B_s(y; y_r, v_r) = \sum_K \langle \Pi^+_y(u \cdot \nabla u_r - \nu \Delta u_r + \nabla p_r), \tau_1(I u \cdot \nabla v_r + \nu \Delta v_r + \nabla q_r) \rangle_K,$$

$$L_s(y; v_r) = \sum_K \langle \delta t^{-1} \dot{u}, \tau_1(I u \cdot \nabla v_r + \nu \Delta v_r + \nabla q_r) \rangle_K,$$

and likewise for $y_h = [u_h, p_h]$ in the FOM formulation. The stabilization parameter matrix is defined following [9; 47] as

$$\tau^{-1}_K(y) = \text{diag}(\tau_1, \tau_2) = \begin{bmatrix} (c_1 \nu + c_2 |u|^2 h) I + c_2 \frac{|u|^2 h}{c_1} |u| I \end{bmatrix}, \quad \text{in } K \in \mathcal{T}_h,$$

$$\tau^{-1}_1 = \delta t^{-1} + \tau^{-1}_1,$$

with $c_1 = 4$ and $c_2 = 2$ for linear elements. In equation (35b), a BDF1 time integration scheme has been assumed for the SGS, with $\dot{u}$ 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 as $\nu = 0.001$, resulting in a Reynolds number of 1000. The domain is discretised using a symmetric mesh of 92320 bilinear elements (figure 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. Figures 7 and 8 show velocity and pressure contours of the resolved scales and the 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.

Figure 7: Velocity (top) and velocity subscales (bottom) at $t = 50$. (a) FOM (b) ROM with $\eta = 0.8$
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 equation (Appendix A.5)). Figure 9 depicts the number of vectors in the basis as a function of the retained energy.

The tests consist in comparing:

- Results for different types of subscales:
  - OSQS vs. ASGS.
  - 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] (see below).

The comparison for all cases is done using the norm of the total force exerted over the cylinder:

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

where $\Gamma_\circ$ is the cylinder boundary.
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 $F_{j,FOM}^\circ$ is the total force obtained with the FOM at time $t_j$, $j = 1, \ldots, S$, and $F_{j,ROM}^\circ$ the one obtained with the ROM model, we set

$$F_{RMSD}^\circ = \sqrt{\frac{1}{S} \sum_{j=1}^{S} (F_{j,ROM}^\circ - F_{j,FOM}^\circ)^2}.$$ 

Along with the error measurement, to depict the behaviour 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 behaviour 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 $\eta$. 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.

Figure 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 $\Pi_\tau$. For this purpose we solve the ROM using the two types of subscales defined in section 5: OSGS and ASGS. Figure 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 OSGS behaves as expected, it deteriorates when less amount of basis vectors and a coarser mesh are used, contrary to the ASGS solution, which is more erratic.
Figure 10: $F_{\text{RMSD}}^2$ convergence for FOM, ROM and ROM with $\eta = 0.9$.

Figure 12 depicts the force RMSD convergence for ROM —varying the retained energy $\eta$ with a fixed $\frac{\Delta E}{E} = 1.0$— 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 OSGS, with the error decreasing with the addition of basis vectors and a finer mesh. As in figure 11, the convergence for the ASGS is irregular.

Figure 13 shows a comparison of the spectra of the force for three combinations of $\eta$ and $h$.

Figure 14 shows the computational time ratio for ROM and hyper-ROM solutions, with the expected behaviour. Computational savings up to 99% for smaller basis size and coarser meshes are observed. Despite the use of OSGS 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. Figure 15 depicts the RMSD of the force for ROM varying the retained energy $\eta$, and for hyper-ROM varying the mesh element size $h$. Here, the behaviour of the ROM subscales mirror the results shown in [5; 48] for a VMS-FE method: the stability of the formulation is improved when 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.

*Interpolation order.* In the third test we analyse the behaviour 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 (figure 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).

Figure 16 depicts the force in the time interval $[40, 45]$ 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 $h_0$.

Figure 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.

Figure 18 depicts the force RMSD convergence for ROM varying the retained energy $\eta$ with a fixed $\frac{h_0}{h} = 1.0$, 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 $h_0$.

Figure 19 shows a comparison of the spectra of the force for three combinations of $\eta$ and $h$.

Figure 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.

Residual-based stabilization. Lastly, we compare the VMS stabilized formulation proposed against an OSGS 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 $u_r$ and $p_r$. Following the formulation presented in [46] for FEs, we split the velocity subscales as $\tilde{u} = \tilde{u}_1 + \tilde{u}_2$. Furthermore, we assume that the convective and pressure terms in the solution of the subscales are orthogonal and neglect second order space derivatives, yielding

$$\partial_t \tilde{u}_1 + \tau_1^{-1} \tilde{u}_1 = -\Pi_r^\perp (u \cdot \nabla u_r),$$

$$\partial_t \tilde{u}_2 + \tau_1^{-1} \tilde{u}_2 = -\Pi_r^\perp (\nabla p_r).$$

Figure 14: Computational time ratio for OSGS and ASGS.

Figure 15: $F_{\text{RMSD}}$ convergence for dynamic and quasi-static OSGS.
Using this definition of the velocity subscales we end with an OSGS formulation for the ROM where the stabilization terms are defined as

\[ B_y(y; y_r, \nu_r) = -\sum_K \langle \Pi_1^+ (\nabla \cdot \nu_1), \tau_1, t \cdot \nabla \nu_r \rangle_K - \sum_K \langle \Pi_1^+ (\nabla \cdot \nu_r), \tau_2, \nabla \cdot \nu_1 \rangle_K \]

\[ -\sum_K \langle \Pi_1^+ (\nabla \cdot \nu_r), \tau_1, t \cdot \nabla q_r \rangle_K, \]

\[ L_y(y; \nu_r) = -\sum_K \langle \delta t^{-1} \bar{u}_1, \tau_1, t \cdot \nabla \nu_r \rangle_K - \sum_K \langle \delta t^{-1} \bar{u}_2, \tau_1, t \cdot \nabla q_r \rangle_K, \]

instead of (35a)-(35b), with the stabilization parameters defined in the same way as for the original OSGS formulation.

In figure 21 we compare the total force at the time interval \([40, 45]\) for both dynamic OSGS 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
8.2.2. Backward-facing step

The third numerical example consist 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 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.
Figure 19: Fourier transform of $F^\circ$ for bases obtained with biquadratic (QB) and bilinear (LB) elements projected over biquadratic (QE) and bilinear (LE) elements.

Figure 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.
Figure 21: Comparison of $F^\circ$ for OSGS and term-by-term (Split) stabilizations using bilinear (subscript L) and biquadratic (subscript Q) elements.

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 discretised 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. Figures 22 and 23 show velocity and pressure contours for FOM, ROM and hyper-ROM at $t = 50$.

Figure 24 shows a comparison of the velocity magnitude and the pressure at the control point $(22, 1)$ for the FOM, ROM and hyper-ROM cases. Figure 25 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 behaviour 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 $t_{ROM}/t_{FOM} = 0.14$ and $t_{HROM}/t_{FOM} = 0.08$, 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 SGS 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:
Figure 22: From top to bottom: velocity contours for FOM, ROM and hyper-ROM with $\eta = 0.95$ and $r = 21$.

Figure 23: From top to bottom: pressure contours for FOM, ROM and hyper-ROM with $\eta = 0.95$ and $r = 21$.

- The use of orthogonal SGSs shows an overall more consistent behaviour over variations of $\eta$ and $h$ than the non-orthogonal SGSs.
- The use of higher order interpolations in the domain discretisation 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.

The use of the mesh based hyper-ROM show 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.

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

- The use of ASGS shows a less stable behaviour over variations of $\eta$ 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.

### Appendix A. 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 $0 \leq t_1 < t_M \leq t_f$ as

$$\{s^j\}_{j=1}^M = \{s^1, s^2, \ldots, s^M\}, \quad \text{(Appendix A.1)}$$

where $s^j = y(x, t_j) - \bar{y}$, $y_j = y(x, t_j)$ is a $j$-th time snapshot and $\bar{y}$ the mean value of the snapshots. We can denote as $\mathcal{Y}_R = \text{span}\{s^j\}_{j=1}^M$ 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 $R = \text{dim}(\mathcal{Y}_R)$.

Now, if $\{\phi^k\}_{k=1}^R$ is an orthonormal basis of $\mathcal{Y}_R$, then any element $y_j$ in the ensemble in equation (Appendix A.1) can be written as

$$y_j = \bar{y} + \sum_{k=1}^R (s_j, \phi^k) \phi^k. \quad \text{(Appendix A.2)}$$

The POD consists in finding the orthonormal basis $\{\phi^k\}_{k=1}^R$ 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 $s_j$, $1 \leq j \leq M$, and the corresponding partial sum in equation (Appendix A.2) is minimized on average as

$$\min_{\{\phi^k\}_{k=1}^R} \frac{1}{M} \sum_{j=1}^M \left\| s_j - \sum_{k=1}^r (s_j, \phi^k) \phi^k \right\|^2,$$

subject to $(\phi^i, \phi^j) = \delta_{ij}$, $1 \leq i, j \leq r$. \quad \text{(Appendix A.3)}

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

34
In our case, using properties (6) we can write the problem in equation (Appendix A.3) in a discrete form as

$$\min_{\Phi \in \mathbb{R}^{Nn \times r}} \left\| M^{1/2} S - M^{1/2} \Phi \Phi^\top M S \right\|^2,$$

subject to

$$\Phi^\top M \Phi = I \in \mathbb{R}^{r \times r},$$

(Appendix A.4)

where $S$ the set of organized discrete snapshots.

To solve the problem in equation (Appendix A.4) we follow a SVD of the weighted snapshot collection $\tilde{S} = M^{1/2} S = U \Lambda V^\top$, where the resulting left singular-vectors of the decomposition represent the weighted basis $U = M^{1/2} \Phi$, of size $R$. Lastly, based on the Eckart-Young-Mirsky theorem we can reduce furthermore the basis by truncating the reduced basis $\Phi$ at the $r$-th column. As a criterion for the truncation, we use the retained energy $\eta$ defined in [16] as

$$\eta = \sum_{k=1}^{r} \lambda_k \sum_{k=1}^{R} \lambda_k,$$

(Appendix 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 $\Phi$, 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.

**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.

**References**


