STABILIZED REDUCED ORDER MODELS FOR LOW SPEED FLOWS

STABILIZED REDUCED ORDER MODELS FOR LOW SPEED FLOWS

RICARDO REYES SOTOMAYOR

On approaching real-time simulations for fluid flows

February 2020

Ricardo Reyes Sotomayor
Stabilized reduced order models for low speed flows
February 2020

PhD in Civil Engineering
Escola Tècnica Superior d’Enginyers de Camins, Canals i Ports de Barcelona
Universitat Politècnica de Catalunya
c/ Jordi Girona 1-3, 08034 Barcelona
https://www.upc.edu/

Advisors: Ramon Codina Rovira
Sergio Idelsohn Barg

To Magdalena

ABSTRACT

This thesis presents the a stabilized projection-based Reduced Order Model (ROM) formulation in low speed fluid flows using a Variational Multi-Scale (VMS) approach. To develop this formulation we use a Finite Element (FE) method for the Full Order Model (FOM) and a Proper Orthogonal Decomposition (POD) to construct the basis.

Additional to the ROM formulation, we introduce two techniques that became possible using this approach: a mesh-based hyper-reduction that uses an Adaptive Mesh Refinement (AMR) approach, and a domain decomposition scheme for ROMs.

To illustrate and test the proposed formulation we use five different models: a convection–diffusion–reaction, the incompressible Navier–Stokes, a Boussinesq approximation, a low Mach number model, and a three-field incompressible Navier–Stokes.

ACKNOWLEDGMENTS

This project has been possible thanks to the Colciencias scholarship for doctoral studies abroad (call 568 of 2012) from the Colombian government.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

AMR	Adaptive Mesh Refinement.
ASGS	Algebraic SubGrid Scale.
ASR	Adaptive Spectrum Refinement.
BDF	Backward Differentiation Formula.
DG	Discontinuous Galerkin.
FE	Finite Element.
FOM	Full Order Model.
OSGS	Orthogonal SubGrid Scale.
POD	Proper Orthogonal Decomposition.
RMS	Root-Mean-Square.
RMSD	Root-Mean-Square Deviation.
ROM	Reduced Order Model.
SE	Spectral Element.
SGS	SubGrid Scale.
SUPG	Streamline-Upwind Galerkin.
SVD	Singular Value Decomposition.
VMS	Variational Multi-Scale.

INTRODUCTION

One of the major drawbacks of accurate numerical simulation of complex problems is the simulation cost and time. Thus, based in the idea of obtaining more affordable solutions, model reduction has been developed using different techniques (see the reviews in [20, 32, 79]). In our case, since we want to retain the physical behavior throughout the whole simulation —opposed to only recovering a response of it, we choose a projection-based approach for the Reduced Order Model (ROM), where 'high fidelity' solutions are used as input data for the construction of a reduced basis in which the simulation is built upon.

Following the model reduction developments in [3, 80, 95] — specifically for fluid problems— we choose a Proper Orthogonal Decomposition (POD) model reduction approach. The POD aims to describe any phenomena that otherwise would be represented by a 'computationally expensive' numerical method with a surrogate lower dimensional model. This surrogate model is obtained by projecting a numerical approximation onto a previously computed reduced space. Thus, the model reduction is arranged in two stages: an offline part, where the solution obtained from a 'high fidelity' Full Order Model (FOM) is used to build the desired reduced order subspace; and an online part, where by projecting the original model onto the reduced subspace, the ROM is built and subsequently solved.

The traditional POD model reduction approach presents certain drawbacks when considering non-linear complex problems: inherent numerical instabilities present in most standard formulations and added computational cost from representing the non-linearities over a linearized computational model.

To overcome the instability issue, several stabilization techniques formulated for projection-based ROMs consist in adding a stabilization term to the Galerkin formulation, which include classical Streamline-Upwind Galerkin (SUPG) [21, 56, 83], Variational Multi-Scale (VMS)-like approaches [13, 21], term by term stabilization methods [8, 67, 68, 73, 90], and some empirical methods [1, 14, 16, 70, 105].

We depart from these formulations by analyzing the problem from a different perspective: instead of formulating the ROM as a projection of the linear system that we want to solve, we describe it as a variational problem on its own. Thus, we can consider the formulation as a projection of the spectral basis of the ROM over a discretized domain, which eases including the VMS stabilization technique in the ROM.

Additional to the instabilities or ill-posedness that might appear in the projection-based ROMs, some of the information obtained by the compression algorithm might be lost, due to the nature of model reduction. Therefore, we justify our choice of the VMS stabilization in the idea that the subscales are designed not only to solve the instabilities related to the physical model, but also to add their contribution to the resolved scales [46], improving the under representation of low energy modes.

In our implementation of the VMS, we follow closely the formulations in [37, 46] for Finite Elements (FEs) problems, where the use of Orthogonal SubGrid Scale (OSGS) and dynamic subscales is encourage, showing improvement of the solution when compared to other approaches. We also follow [25, 26], where the appropriate selection of the reduced subspace is encouraged for non-linear problems, using oblique projections.

To decrease the computational cost of assembling a non linear problem, a wide variety of methods inspired by the work of Everson and Sirovich [51] have been introduced in [15, 18, 26, 62, 85, 92], with the term 'hyper-ROM' coined by Ryckelink in [91]. This family of

methods consists in using a small sample of the geometrical domain. Alternatively, we propose a mesh-based hyper-ROM that consist in evaluating the nonlinear terms on meshes coarser than the FOM one, trying not to increase the error of the ROM.

This approach suggest the use techniques already available for mesh-based formulations to obtain a better suited mesh for the hyper-reduction, i.e. Adaptive Mesh Refinement (AMR). While this can be done once in the offline stage, as proposed in [48, 58, 101] for the FOM solution and in [59] as a pre-computed mesh, the method we propose is to carry out a genuine AMR in the ROM stage, and thus to compute the error of the mesh-based ROM solution and perform an AMR according to this error. To achieve this, besides the refinement algorithm, a definition of the error estimators that allow one to decide over which domain region mesh refinement is required. The use of residual-base error estimators in FE problems is vast in the literature (see [11, 60, 61] and references within), and since they are explicit and do not require solving additional equations, they can be useful in our ROM context.

We follow an approach similar to [19], where the error estimator is defined using dynamic OSGS and including the definition of boundary subscales presented in [45]. As a refinement algorithm we use solely the hierarchical h-AMR strategy presented in [10].

Additional to the VMS stabilization and the hyper-reduction, we include two techniques that tackle to minor issues: a Petrov–Galerkin projection and a domain decomposition strategy. We follow the Petrov–Galerkin projection presented in [26], as a conditioning technique to improve the ROM approximation, based in the idea that selecting an appropriate projection space to solve the resulting linear system is fundamental to obtain a meaningful solution (see [93]). For the ROM domain decomposition technique we pursue a similar strategy as the one presented in [42] for FEs. Here, we present a non-overlapping

iterative method for homogeneous and heterogeneous boundaries using global and local bases.

To develop the formulation we focus on FE-ROMs, with a basis obtained using a FE-FOM. To test the formulation we follow five different physical problems: a convection–diffusion–reaction, an incompressible Navier–Stokes, a Boussinesq approximation, the zero Mach number Navier–Stokes approximation presented in [89], and the three-field incompressible Navier–Stokes formulation used in [27].

OBJECTIVES AND OUTLINE

The main purpose of the work is to formulate a ROM that allows to approximate near real time simulation of complex low speed flows resembling complex phenomena. To this end, our main objective is to develop a stabilized ROM formulation while keeping a low computational cost.

The dissertation is organized as follows: In part i we review briefly the standard approach for projection-based ROMs based on a POD and a FE method. Here, the necessary tools for the construction of the stabilized formulation are explained. In part ii the main work of the thesis is presented, where the description of the stabilized formulation is carried on for both the FOM and the ROM. In part iii we present the additional considerations and techniques needed to solve the desired physical problems, including the hyper reduction, the domain decomposition schemes and the selection of the appropriate projection space. And lastly, in part iv we present the physical models and the results obtained with our proposed method.

Part I

REDUCED ORDER MODELS

1 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 $\{\boldsymbol{\phi}^k\}_{k=1}^r$, with each vector being an array of n components. We can compute the best L^2-approximation of $\mathbf{y}$ in $\mathcal{Y}_r$ at a given time $t \in]0, t_f[$ by $\sum_{k=1}^r (\mathbf{y}, \boldsymbol{\phi}^k)\boldsymbol{\phi}^k$.

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 high-fidelity 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. [31, 80, 94]), following some traditional works in model reduction [3, 74, 75, 95–97] and some more recent works in stabilized ROMs [14, 21, 67], we choose the POD to construct the ROM basis. Additionally, we chose a FE method for the FOM discretization.

Note that in our case, the FOM for a continuous problem yields an ordinary differential equation of the form

$$\mathbf{M}\dot{\mathbf{y}} + \mathbf{K}\mathbf{y} = \mathbf{f}, \tag{1.1}$$

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, we can gather the data needed to construct the basis from solutions of the discrete problem as a collection of 'snapshots'. Thus, 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 a-th vector component of the k-th basis vector, with $a = 1, \dots, N$ and $k = 1, \dots, r$, we may identify $\boldsymbol{\phi}^k$ with the function

$$\boldsymbol{\phi}^k(\mathbf{x}) = \sum_{a=1}^{N} \varphi^a(\mathbf{x}) \boldsymbol{\phi}^{k,a}, \qquad k = 1, \dots, r, \quad \forall \mathbf{x} \in \Omega, \tag{1.2}$$

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

2 CONSTRUCTION OF THE BASIS

To construct the reduced order basis, we start by organizing a set of data previously obtained from the FOM solution over the time grid $0 \leqslant t_1 < t_m \leqslant t_f$ as

$$\{\mathbf{s}^j\}_{j=1}^m = \{\mathbf{s}^1, \mathbf{s}^2, \ldots, \mathbf{s}^m\}, \tag{2.1}$$

where $\mathbf{s}^j = \mathbf{y}(\mathbf{x}, t_j) - \bar{\mathbf{y}}$, $\mathbf{y}_j = \mathbf{y}(\mathbf{x}, t_j) \subset \mathcal{Y}_h$ is the j-th time snapshot and $\bar{\mathbf{y}}$ is the mean value of the snapshots. We can denote as $\mathcal{Y}_R = \text{span}\{\mathbf{s}^j\}_{j=1}^m$ the space spanned by the ensemble of data (snapshots space), and given that in general the snapshots are not linearly independent, it holds $R \leqslant m$, with $R = \dim(\mathcal{Y}_R)$.

If $\{\boldsymbol{\phi}^k\}_{k=1}^R$ is an orthonormal basis of $\mathcal{Y}_R$, then any element $\mathbf{y}_j$ in the ensemble in equation (2.1) can be written as

$$\mathbf{y}_j = \bar{\mathbf{y}} + \sum_{k=1}^{R} (\mathbf{s}_j, \boldsymbol{\phi}^k)\boldsymbol{\phi}^k. \tag{2.2}$$

2.1 PROPER ORTHOGONAL DECOMPOSITION

Using the collected data, we follow the POD to construct a basis $\{\boldsymbol{\phi}^k\}_{k=1}^r$ that provides the best possible approximation of the given set of 'snapshots', with $r \leqslant R$. Here, we describe shortly the how we approach the POD for our particular case, that is, where the snapshots are solutions of a FE-FOM. For a more extensive and general description of the method see e.g. [52, 53, 55, 102].

The POD can be formulated as an optimization problem, where for every $k \in \{1, \ldots, r\}$ the mean square error between the elements $\mathbf{s}_j$,

$1 \leqslant j \leqslant M$, and the corresponding partial sum in equation (2.2) is minimized on average as

$$\min_{\{\boldsymbol{\phi}^k\}_{k=1}^r} \quad \frac{1}{m}\sum_{j=1}^{m}\left\|\mathbf{s}_j - \sum_{k=1}^{r}(\mathbf{s}_j,\boldsymbol{\phi}^k)\boldsymbol{\phi}^k\right\|^2,$$
$$\text{subject to } (\boldsymbol{\phi}^i,\boldsymbol{\phi}^j) = \delta_{ij}, \quad 1 \leqslant i,j \leqslant r. \tag{2.3}$$

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 $r \leqslant R \leqslant m \ll Nn$.

2.2 SINGULAR VALUE DECOMPOSITION

We can write the problem in equation (2.3) in a discrete FE setting as

$$\min_{\mathbf{M}^{1/2}\boldsymbol{\Phi}\in\mathbb{R}^{Nn\times r}} \left\|\mathbf{M}^{1/2}\mathbf{S} - \mathbf{M}^{1/2}\boldsymbol{\Phi}\boldsymbol{\Phi}^\top\mathbf{M}\mathbf{S}\right\|^2,$$
$$\text{subject to} \qquad \boldsymbol{\Phi}^\top\mathbf{M}\boldsymbol{\Phi} = \mathbf{I} \in \mathbb{R}^{r\times r}, \tag{2.4}$$

where $\mathbf{S}$ is the set of M organized discrete FE snapshots.

To solve the discrete problem in equation (2.4), we follow the SVD of the weighted snapshot collection $\widehat{\mathbf{S}} = \mathbf{M}^{1/2}\mathbf{S}$, yielding a factorization of the form

$$\widehat{\mathbf{S}} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{V} \tag{2.5}$$

where the resulting left singular-vectors $\mathbf{U} = \mathbf{M}^{1/2}\boldsymbol{\Phi}$ of size $Nn \times R$ represent the weighted basis, and the diagonal matrix $\boldsymbol{\Lambda}$ represents the R singular values of the problem in descending order.

In other similar approaches to constructing the reduced order basis, the left and right singular-vectors —$\mathbf{U}$ and $\mathbf{V}$— are used as representations of the space and time decompositions of the same set of data (see e.g. [3, 4, 33]).

Lastly, based on the Eckart-Young-Mirsky theorem we can reduce furthermore the basis by truncating the reduced basis $\boldsymbol{\Phi}$ at the r-th

column. As a criterion for the truncation, we use the retained energy η defined in [95] as

$$\eta = \frac{\sum_{k=1}^{r} \lambda^k}{\sum_{k=1}^{R} \lambda^k}, \tag{2.6}$$

where λ^k is the k-th component of $\boldsymbol{\Lambda}$ 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. [31, 94]). The stabilized formulation shown in the following sections should be valid for any basis regardless of the technique used to obtain it.

2.3 PROPERTIES OF THE BASIS

Although it is not the goal of this work to analyze any specific method to construct the reduced order basis, it is important to mention some properties of the one obtained using the POD over a FE ensemble of data:

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

$$\langle \boldsymbol{\phi}^k, \boldsymbol{\phi}^l \rangle_{\mathcal{M}} = \int_\Omega \boldsymbol{\phi}^{k\top} \boldsymbol{\mathcal{M}} \boldsymbol{\phi}^l = \delta^{kl}, \quad k, l = 1, \dots r, \tag{2.7a}$$

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

$$\boldsymbol{\phi}^{k\top} \mathbf{M} \boldsymbol{\phi}^l = \delta^{kl} \quad k, l = 1, \dots r, \tag{2.7b}$$

   where $\mathbf{M}$ is the matrix in equation (1.1).
2. It satisfies homogeneous Dirichlet boundary conditions when calculated using a mean centered-trajectory approximation (equation (2.2)) .
3. It has an optimal approximation when truncated (Eckart-Young-Mirsky theorem in equation (2.6)).

3 PROJECTION-BASED REDUCED ORDER MODELS

Using the basis construction and properties in chapter 2 we can define a standard Galerkin projection ROM for equation (1.1) as

$$\mathbf{M}_r \dot{\mathbf{y}}_r + \mathbf{K}_r \mathbf{y}_r = \mathbf{f}_r, \tag{3.1}$$

where $\mathbf{y}_r$ is the time dependent solution array of r components, $\mathbf{M}_r = \boldsymbol{\phi}^\top \mathbf{M} \boldsymbol{\phi} = \mathbf{I}$ is the ROM mass matrix, $\mathbf{I}$ is the identity matrix, $\mathbf{K}_r = \boldsymbol{\phi}^\top \mathbf{K} \boldsymbol{\phi}$ is the projected stiffness matrix and $\mathbf{f}_r = \boldsymbol{\phi}^\top \mathbf{f}$ is the projected force vector. Note that only the non linear parts of the problem would need to be recalculated as the problem evolves, so in the case of linear problems a simpler eigenvalue approximation would suffice.

HOMOGENEOUS DIRICHLET BOUNDARY CONDITIONS. To implement homogeneous boundary conditions, we follow the mean centered-trajectory approximation in equation (2.2), and accordingly write the ROM unknown as

$$y_r(x,t) := \sum_{a=1}^{N} \varphi^a(x) \left[\bar{y} + \sum_{k=1}^{r} \phi^{k,a} Y^k(t) \right], \tag{3.2}$$

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.

Part II

VARIATIONAL MULTI-SCALE

4 STABILIZATION VIA VARIATIONAL MULTI-SCALE

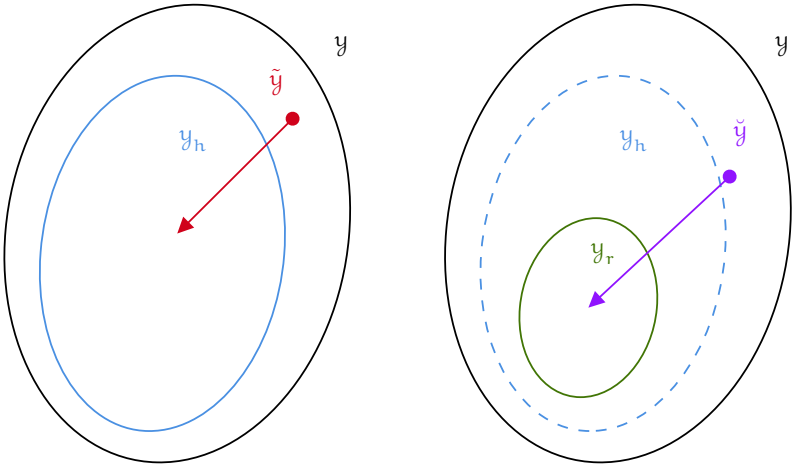
In this chapter we formulate a residual-based stabilized method for the projection-based Reduced Order Models (ROMs) described before by using the Variational Multi-Scale (VMS) idea introduced in [63, 64]. First, we decide to look at the problem at hand from a different scope; instead of formulating 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. Furthermore, we will see that using the Finite Element (FE) method as the Full Order Model (FOM) 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 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 [41] for FEs and the methods developed in [81, 104] 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 [46]. 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 [38]— by the orthonormal nature of the reduced space basis.



(a) FOM $\mathcal{Y}_h$ and FOM-OSGS $\tilde{\mathcal{Y}}$.

(b) ROM $\mathcal{Y}_r$ and ROM-OSGS $\breve{\mathcal{Y}}$.

Figure 4.1: Illustration of the space $\mathcal{Y}$ and the subspaces in FOM and ROM.

To illustrate the equivalence between the OSGS in both the ROM and the FOM, figure 4.1 shows a comparison between FOM and ROM resolved scale —$\mathcal{Y}_h$ and $\mathcal{Y}_r$— and subscale —$\tilde{\mathcal{Y}}$ and $\check{\mathcal{Y}}$— spaces. These subspaces are such that $\mathcal{Y} = \mathcal{Y}_h \oplus \tilde{\mathcal{Y}} = \mathcal{Y}_r \oplus \check{\mathcal{Y}}$, where $\mathcal{Y}$ is the space of functions where the problem is defined, assumed to be a Hilbert space. Let us remark that $\check{\mathcal{Y}}$ is the complementary in $\mathcal{Y}$ of $\mathcal{Y}_r$, not the complementary in $\mathcal{Y}_h$, as it was done in [14]. Therefore, subscales of the ROM space may not belong to the FOM space.

Essentially, stabilization techniques formulated for projection-based ROMs, including the VMS one, 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 OSGS as done in [35]. Furthermore, to define $\boldsymbol{\tau}$ we follow the same Fourier analysis of the subscales as the one described in [37] for the FE method.

4.1 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 $\mathbf{y}(\mathbf{x}, t)$ of n components such that

$$\boldsymbol{\mathcal{M}}(\mathbf{y})\partial_t \mathbf{y} + \boldsymbol{\mathcal{L}}(\mathbf{y}; \mathbf{y}) = \mathbf{f}, \quad \text{in } \Omega, \ t \in]0, t_f[, \tag{4.1}$$

where d is the number of space dimensions, $\boldsymbol{\mathcal{M}}(\mathbf{y})$ is the mass matrix, $\mathbf{f}(\mathbf{x}, t)$ is the forcing term, ∂_t is the derivative over time, and $\boldsymbol{\mathcal{L}}$ is a nonlinear differential operator in space of first or second order defined as

$$\mathcal{L}(y;z) := A_i^c(y)\partial_i z + A_i^f(y)\partial_i z - \partial_i\big(K_{ij}(y)\partial_j z\big) + S(y)z.$$

$A_i^c(y)$, $A_i^f(y)$, $K_{ij}(y)$ and $S(y)$ are $n \times n$ matrices and ∂_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, if $\Gamma = \Gamma_D \cup \Gamma_N$, the initial and boundary conditions are

$$\begin{aligned} y(x,0) &= y_0(x), && \text{in } \Omega, && t = 0,\\ \mathcal{D}y &= \mathcal{D}y_D && \text{on } \Gamma_D, && t \in]0,t_f[,\\ \mathcal{F}(y;y) &= t_N, && \text{on } \Gamma_N, && t \in]0,t_f[, \end{aligned}$$

where y_0 is the prescribed initial condition, y_D is the prescribed Dirichlet boundary condition, $\mathcal{D}$ is the Dirichlet operator associated to $\mathcal{L}$, $\mathcal{F}$ is a flux operator defined as

$$\mathcal{F}(y;z) := n_i K_{ij}(y)\partial_j z - n_i A_i^f(y)z,$$

n is the normal to Γ 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 Ω where y is sought for each time t, satisfying $\mathcal{D}y = \mathcal{D}y_D$ on Γ_D, and let $\mathcal{Y}_0$ be its corresponding space of time independent test functions that satisfy $\mathcal{D}v = 0$ 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 $y :]0,t_f[\longrightarrow \mathcal{Y}$ such that

$$\begin{aligned} (\mathcal{M}(y)\partial_t y, v) + \langle \mathcal{L}(y;y), v\rangle &= \langle f, v\rangle, && t \in]0,t_f[,\ \forall v \in \mathcal{Y}_0,\\ (y,v) &= (y_0,v), && t = 0,\ \forall v \in \mathcal{Y}_0. \end{aligned} \tag{4.2}$$

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

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

we can write an equivalent problem to equation (4.2): find a vector function $\mathbf{y} :]0, t_f[\to \mathcal{Y}$ such that

$$
(\boldsymbol{\mathcal{M}}(\mathbf{y}) \partial_t \mathbf{y}, \mathbf{v}) + B(\mathbf{y}; \mathbf{y}, \mathbf{v}) = L(\mathbf{v}), \quad \forall \mathbf{v} \in \mathcal{Y}_0, \ t \in]0, t_f[. \tag{4.3}
$$

5 FULL ORDER MODEL

5.1 VARIATIONAL FOM FORMULATION

To formulate the discrete FOM for the variational problem in equation (4.3) we denote by $\mathcal{T}_h = \{K\}$ a partition of the domain Ω, with $h = \max\{h_K = \operatorname{diam}(K) | K \in \mathcal{T}_h\}$ the diameter of the partition, and by $\mathcal{Y}_h \subset \mathcal{Y}$ 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 $\mathcal{Y}$, and $\mathcal{Y}_h$ 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

$$\partial_t \mathbf{y} \,|_{t^j} \approx \delta_t \mathbf{y}^j := \sum_{l=0}^{s} \frac{\alpha^l \mathbf{y}^{j-l}}{\delta t}, \quad j = 1, 2, \ldots, \tag{5.1}$$

where the operator $\delta_t \mathbf{y}^j$ indicates a discrete temporal derivatives over functions $\mathbf{y}$ 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, $t_f/\delta t$.

The discrete Galerkin FE version of the problem in equation (4.3) is: find $\{\mathbf{y}_h^j\} \subset \mathcal{Y}_h$, such that

$$(\boldsymbol{\mathcal{M}}(\hat{\mathbf{y}})\delta_t \mathbf{y}_h, \mathbf{v}_h) + B(\hat{\mathbf{y}}; \mathbf{y}_h, \mathbf{v}_h) = L(\mathbf{v}_h), \quad \forall \mathbf{v}_h \in \mathcal{Y}_h, \tag{5.2}$$

which we call the Galerkin form of the FOM.

Since the solution to the discrete problem in equation (5.2) may suffer from numerical instabilities that depend on the expression of the matrices that define the operator $\mathcal{L}$, arising from the (possible) lack of coercivity of the problem and the compatibility restrictions between the components of y, a stabilized formulation is needed. We can write such formulation for equation (5.2) in a general form as

$$\begin{aligned}(\mathcal{M}(\hat{y})\delta_t y_h, v_h) + B(\hat{y}; y_h, v_h) \\ + \langle \tau_h(\hat{y})\mathcal{R}_h(\hat{y}; y_h), \mathcal{P}_h(\hat{y}; v_h)\rangle = L(v_h), \quad \forall v_h \in \mathcal{Y}_h,\end{aligned} \tag{5.3}$$

where $\hat{y}$ is an approximation to y for which two possibilities are discussed in remark 3. To simplify the notation, hereafter we shall write y instead of $\hat{y}$.

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 [41] to formulate the stabilized FOM.

5.2 VMS IN THE FULL ORDER MODEL

The VMS method consists in decomposing the space of the unknown into the finite-dimensional space —resulting from the finite element discretization— $\mathcal{Y}_h$, and a continuous one $\tilde{\mathcal{Y}}$, so that $\mathcal{Y} = \mathcal{Y}_h \oplus \tilde{\mathcal{Y}}$. The unknown and the test functions are accordingly split as $y = y_h + \tilde{y}$ and $v = v_h + \tilde{v}$, respectively. Then, once discretized in time the continuous problem in equation (4.3) expands into two: find $\{y_h^j\} \subset \mathcal{Y}_h$ and $\{\tilde{y}^j\} \subset \tilde{\mathcal{Y}}$, such that

$$\begin{aligned}(\mathcal{M}(y)\delta_t y_h, v_h) + B(y; y_h, v_h) + (\mathcal{M}(y)\delta_t \tilde{y}, v_h) \\ + B(y; \tilde{y}, v_h) = L(v_h), \quad \forall v_h \in \mathcal{Y}_h,\end{aligned} \tag{5.4a}$$

$$(\boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \mathbf{y}_h, \tilde{\mathbf{v}}) + B(\mathbf{y}; \mathbf{y}_h, \tilde{\mathbf{v}}) + (\boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \tilde{\mathbf{y}}, \tilde{\mathbf{v}}) + B(\mathbf{y}; \tilde{\mathbf{y}}, \tilde{\mathbf{v}}) = L(\tilde{\mathbf{v}}), \quad \forall\, \tilde{\mathbf{v}} \in \tilde{\mathcal{Y}}. \tag{5.4b}$$

This corresponds to take $\hat{\mathbf{y}} = \mathbf{y}$ in equation (5.3). We refer to functions in $\mathcal{Y}_h$ as the FOM *resolved scales* and to functions in $\tilde{\mathcal{Y}}$ as the FOM *SubGrid Scales (SGS)* or simply *subscales*.

Since we want to avoid terms that involve applying the differential operator $\boldsymbol{\mathcal{L}}$ over the subscales $\tilde{\mathbf{y}}$, we re-write the form $B(\mathbf{y}; \tilde{\mathbf{y}}, \mathbf{v}_h)$ in equation (5.4a) by integrating by parts. That leads to the formal adjoint of the operator $\boldsymbol{\mathcal{L}}$, defined as

$$\boldsymbol{\mathcal{L}}^*(\mathbf{y}; \mathbf{v}_h) := \mathbf{S}(\mathbf{y})^\top \mathbf{v}_h - \partial_i \mathbf{A}_i^c(\mathbf{y})^\top \mathbf{v}_h - \partial_i \mathbf{A}_i^f(\mathbf{y})^\top \mathbf{v}_h - \partial_i \left(\mathbf{K}_{ji}^\top(\mathbf{y}) \partial_j \mathbf{v}_h \right),$$

and the adjoint of flux operator $\boldsymbol{\mathcal{F}}$, defined as

$$\boldsymbol{\mathcal{F}}^*(\mathbf{y}; \mathbf{v}_h) := \mathbf{n}_i \mathbf{K}_{ji}^\top(\mathbf{y}) \partial_j \mathbf{v}_h + \mathbf{n}_i \mathbf{A}_i^{c\top}(\mathbf{y}) \mathbf{v}_h.$$

Moreover, we define the residual of the resolved scales as $\mathbf{r}(\mathbf{y}; \mathbf{y}_h) := \mathbf{f} - \boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \mathbf{y}_h - \boldsymbol{\mathcal{L}}(\mathbf{y}; \mathbf{y}_h)$ at each time level t^j, and integrate by parts the forms $B(\mathbf{y}; \mathbf{y}_h, \tilde{\mathbf{v}})$ and $B(\mathbf{y}; \tilde{\mathbf{y}}, \tilde{\mathbf{v}})$ in equation (5.4b).

Thus, we can re-write equation (5.4) as

$$(\boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \mathbf{y}_h, \mathbf{v}_h) + B(\mathbf{y}; \mathbf{y}_h, \mathbf{v}_h) + (\boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \tilde{\mathbf{y}}, \mathbf{v}_h) + \sum_K \langle \tilde{\mathbf{y}}, \boldsymbol{\mathcal{L}}^*(\mathbf{y}; \mathbf{v}_h) \rangle_K + \sum_K \langle \tilde{\mathbf{y}}, \boldsymbol{\mathcal{F}}^*(\mathbf{y}; \mathbf{v}_h) \rangle_{\partial K} = L(\mathbf{v}_h), \quad \forall \mathbf{v}_h \in \mathcal{Y}_h, \tag{5.5a}$$

$$\sum_K \langle \boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \tilde{\mathbf{y}}, \tilde{\mathbf{v}} \rangle_K + \sum_K \langle \boldsymbol{\mathcal{L}}(\mathbf{y}, \tilde{\mathbf{y}}), \tilde{\mathbf{v}} \rangle_K + \sum_K \langle \boldsymbol{\mathcal{F}}(\mathbf{y}; \tilde{\mathbf{y}}), \tilde{\mathbf{v}} \rangle_{\partial K} = \sum_K \langle \mathbf{r}(\mathbf{y}; \mathbf{y}_h), \tilde{\mathbf{v}} \rangle_K, \quad \forall \tilde{\mathbf{v}} \in \tilde{\mathcal{Y}}, \tag{5.5b}$$

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.

5.3 FOM SUBSCALES

Considering that the subscale problem in equation (5.5b) 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 τ_K is the matrix of stabilization parameters that approximates the inverse of the differential operator $\mathcal{L}$ on each element K. The definition of τ_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 [37]. Furthermore, assuming that the normal fluxes of the total unknown are continuous across the inter-element boundaries, the boundary terms in equation (5.5) vanish and we arrive to

$$\mathcal{M}(y)\delta_t\tilde{y} + \tau_K^{-1}(y)\tilde{y} = r(y;y_h) + \tilde{v}^{\perp}, \quad \text{in } K \in \mathcal{T}_h, \tag{5.6}$$

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

$$\sum_K \langle \tilde{v}^{\perp}, \tilde{v} \rangle_K = 0, \quad \forall \tilde{v} \in \tilde{\mathcal{Y}}, \tag{5.7}$$

defined by the choice of the space of subscales $\tilde{\mathcal{Y}}$ and its orthogonal complement $\tilde{\mathcal{Y}}^{\perp}$. By defining $\tilde{\Pi}^{\perp}$ the L^2 projection onto $\tilde{\mathcal{Y}}^{\perp}$ we can impose the condition in equation (5.7). Projecting equation (5.6) onto $\tilde{\mathcal{Y}}^{\perp}$ and using equation (5.7), we get $\tilde{v}^{\perp} = -\tilde{\Pi}^{\perp}(r(y;y_h))$ and therefore

$$\mathcal{M}(y)\delta_t\tilde{y} + \tau_K^{-1}(y)\tilde{y} = \tilde{\Pi}(r(y;y_h)), \quad \text{in } K \in \mathcal{T}_h, \tag{5.8}$$

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

Remark 1. *Another way to impose this condition is by using the weighted inner product* $(\cdot,\cdot)_\tau = \sum_K \langle \tau_K \cdot, \cdot \rangle_K$ *and the associated projection* $\tilde{\Pi}_\tau^{\perp}$ *onto* $\tilde{\mathcal{Y}}^{\perp}$. *This way, projecting equation* (5.6) *onto* $\tilde{\mathcal{Y}}^{\perp}$ *and using equation* (5.7)*, leads to* $\tilde{v}^{\perp} = -\tilde{\Pi}_\tau^{\perp}(r(y;y_h))$. *Further details on this approximation can be found in [38].*

ALGEBRAIC AND ORTHOGONAL SUBSCALES. Depending on the choice of the projection $\tilde{\Pi}$ —which involves the choice of the space $\tilde{\mathcal{Y}}$, we get different approximation methods for the subscales. In [37] two alternatives for formulating the subscales are described: an Algebraic SubGrid Scale (ASGS) formulation that consists in taking $\tilde{\mathcal{V}}^{\perp} = 0$ and therefore $\tilde{\Pi} = \mathbf{I}$, giving $\mathcal{R}_h(\mathbf{y}; \mathbf{y}_h) = \mathbf{r}(\mathbf{y}; \mathbf{y}_h)$, and an OSGS formulation that consists in defining the space of subscales as orthogonal to the complement of $\mathcal{Y}_h$, having $\tilde{\mathcal{Y}} = \mathcal{Y}_h^{\perp}$. This implies satisfying the condition $\sum_K \langle \mathcal{V}_h, \tilde{\mathcal{V}} \rangle = 0$ and therefore $\tilde{\Pi} := \Pi_h^{\perp} = \mathbf{I} - \Pi_h$, where Π_h is the projection onto the FE space, giving $\mathcal{R}_h(\mathbf{y}; \mathbf{y}_h) = \Pi_h^{\perp}(\mathbf{f} - \mathcal{L}(\mathbf{y}; \mathbf{y}_h))$.

QUASI-STATIC AND DYNAMIC SUBSCALES. Following [46] 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. $\partial_t \tilde{\mathbf{y}} = 0$— and yielding the subscale approximation $\tau_K^{-1}(\mathbf{y})\tilde{\mathbf{y}} = \mathcal{R}_h(\mathbf{y}; \mathbf{y}_h)$, and dynamic subscales, obtained when such time derivative is considered, approximating the subscales as the solution of $\mathcal{M}\delta_t \tilde{\mathbf{y}} + \tau_K^{-1}(\mathbf{y})\tilde{\mathbf{y}} = \mathcal{R}_h(\mathbf{y}; \mathbf{y}_h)$. Here, an effective stabilization parameter can be defined as $\tau_{K,t}^{-1}(\mathbf{y}) := \alpha^0 \delta t^{-1} \mathcal{M} + \tau_K^{-1}(\mathbf{y})$.

Remark 2. *As it is discussed in [46], 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 $\mathcal{P}_h(\mathbf{y}, \mathcal{V}_h) = \mathcal{L}^*(\mathbf{y}, \mathcal{V}_h)$, and choosing between algebraic and orthogonal, and quasi-static and dynamic SGS, we can obtain a general stabilized approximation for $\mathbf{y}_h$ of the form of equation (5.3), in principle with $\hat{\mathbf{y}} = \mathbf{y}_h + \tilde{\mathbf{y}}$ (written as $\mathbf{y}$). Moreover, we can rewrite this equation as

$$(\mathcal{M}(\mathbf{y})\delta_t \mathbf{y}_h, \mathcal{V}_h) + B_h(\mathbf{y}; \mathbf{y}_h, \mathcal{V}_h) = L_h(\mathbf{y}; \mathcal{V}_h), \quad \forall \mathcal{V}_h \in \mathcal{Y}_h. \tag{5.9}$$

Thus, we define the forms $B_h(y; y_h, v_h) = B(y; y_h, v_h) + B_s(y; y_h, v_h)$ and $L_h(y; v_h) = L(v_h) + L_s(y; v_h)$, and the stabilization terms $B_s(y; y_h, v_h)$ and $L_s(y; v_h)$, using a first order BDF to approximate the time derivative of the SGS, for each proposed case in table 5.1 and table 5.2, respectively.

	ASGS
Dynamic	$-\sum_K \langle \mathcal{M}(y)\delta_t y_h + \mathcal{L}(y; y_h), \tau_{K,t}\mathcal{L}^*(y, v_h)\rangle_K$
	$-\sum_K \langle \mathcal{M}(y)\delta_t y_h + \mathcal{L}(y; y_h), [I - \tau_K^{-1}\tau_{K,t}]v_h\rangle_K$
Quasi-static	$-\sum_K \langle \mathcal{M}(y)\delta_t y_h + \mathcal{L}(y; y_h), \tau_K\mathcal{L}^*(y, v_h)\rangle_K$
	OSGS
Dynamic	$-\sum_K \langle \Pi_h^{\perp}(\mathcal{L}(y; y_h)), \tau_{K,t}\mathcal{L}^*(y, v_h)\rangle_K$
Quasi-static	$-\sum_K \langle \Pi_h^{\perp}(\mathcal{L}(y; y_h)), \tau_K\mathcal{L}^*(y, v_h)\rangle_K$

Table 5.1: FOM $B_s(y; y_h, v_h)$ defined for dynamic and quasi-static ASGS and OSGS.

	ASGS
Dynamic	$-\sum_K \langle f, \tau_{K,t}\mathcal{L}^*(y, v_h) + [I - \tau_K^{-1}\tau_{K,t}]v_h\rangle_K$
	$-\sum_K \langle \delta t^{-1}\mathcal{M}(y)\tilde{y}^{j-1}, \tau_{K,t}\mathcal{L}^*(y, v_h)\rangle_K$
	$+\sum_K \langle \delta t^{-1}\mathcal{M}(y)\tilde{y}^{j-1}, \tau_K^{-1}\tau_{K,t}v_h\rangle_K$
Quasi-static	$-\sum_K \langle f, \tau_K\mathcal{L}^*(y, v_h)\rangle_K$
	OSGS
Dynamic	$-\sum_K \langle \Pi_h^{\perp}(f) + \delta t^{-1}\mathcal{M}(y)\tilde{y}^{j-1}, \tau_{K,t}\mathcal{L}^*(y, v_h)\rangle_K$
Quasi-static	$-\sum_K \langle \Pi_h^{\perp}(f), \tau_K\mathcal{L}^*(y, v_h)\rangle_K$

Table 5.2: FOM $L_s(y; v_h)$ defined for dynamic and quasi-static ASGS and OSGS.

Considering the results in [37] and in [9, 43, 46], where the subscale approximation is shown to influence convergence and stability, we choose to formulate the FE-FOM using dynamic OSGS. It is important to mention that the stabilization method chosen for the FOM has no impact in the development of the stabilized ROM formulation in chapter 6.

Remark 3. *The treatment of the term y in the operators $\mathcal{M}(y)$, $\mathcal{L}(y,\cdot)$, $\mathcal{L}^*(y,\cdot)$, $\mathcal{F}(y,\cdot)$ 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 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 equation (5.4a). See [37] for a detailed description of the nonlinear subscales.*

Remark 4. *The practical way to compute the projection $\Pi_h^\top$ is to compute the projection Π_h and then use $\Pi_h^\top = I - \Pi_h$. Therefore, by denoting as $z_h = \Pi_h(\mathcal{L}(y;y_h) - f)]$, the problem in equation (5.9) —for dynamic OSGS— becomes: find $[y_h, z_h]$ in $\mathcal{Y}_h \times \mathcal{Y}_h$ such that*

$$
\begin{aligned}
&(\mathcal{M}(y)\delta_t y_h, v_h) + B(y;y_h,v_h) \\
&\qquad - \sum_K \langle \mathcal{L}(y;y_h) - z_h, \tau_{K,t}(y)\mathcal{L}^*(y,v_h)\rangle_K \\
&\qquad = L(y;v_h) - \sum_K \langle f, \tau_{K,t}(y)\mathcal{L}^*(y,v_h)\rangle_K \\
&\qquad - \sum_K \langle \delta t^{-1}\mathcal{M}(y)\tilde{y}^{j-1}, \tau_{K,t}(y)\mathcal{L}^*(y,v_h)\rangle_K, \qquad (5.10a)
\end{aligned}
$$

$$
(z_h, \zeta_h) = \sum_K \langle \mathcal{L}(y;y_h) - f, \zeta_h\rangle_K, \qquad (5.10b)
$$

which must hold for all $[v_h, \zeta_h]$ in $\mathcal{Y}_{h,0} \times \mathcal{Y}_h$. To approximate equation (5.10b) we follow the block iteration algorithm in [38], having $[\Pi_h^\top(\mathcal{L}(y;y_h))]^{i+1} = [\mathcal{L}(y;y_h)]^{i+1} - [\Pi_h(\mathcal{L}(y;y_h))]^i$, where the superscript i denotes the iteration counter.

5.4 FOM SUBSCALES ON THE BOUNDARIES

If we do not consider the fluxes as continuous over the element boundaries, we need to add the term that includes the subscales in the boundaries in equation (5.5) $-\sum_K \langle \tilde{y}, \mathcal{F}^*(y;v_h)\rangle_{\partial K}$. To approximate it, we follow the formulation for the FE problem in [45].

We start by writing the weak continuity of the total fluxes on the element boundaries as

$$\begin{aligned} 0 &= \sum_K \langle \mathcal{F}(y;y), \tilde{v} \rangle_{\partial K} \\ &= \sum_K \langle \mathcal{F}(y;y_h), \tilde{v} \rangle_{\partial K} + \sum_K \langle \mathcal{F}(y;\tilde{y}), \tilde{v} \rangle_{\partial K}, \quad \forall \tilde{v} \in \tilde{\mathcal{Y}}, \end{aligned} \tag{5.11}$$

assuming for the moment that there is no Neumann boundary. This equation is a consequence of the fact that the flux operator $\mathcal{F}(y;y)$ has to be continuous across the inter-element boundaries.

Let E be an edge of the finite element partition (face, in 3D), shared by two element domains K_1 and K_2. In the case E is an edge on the boundary, it is understood that K_2 is void. Let $\mathcal{F}(y;y_r)|_{\partial K_i \cap E}$ the flux computed with the normal from K_i, $i = 1,2$, and define the jump of fluxes as

$$[\![\mathcal{F}(y;z)]\!]_E := \mathcal{F}(y;z)|_{\partial K_1 \cap E} + \mathcal{F}(y;z)|_{\partial K_2 \cap E}.$$

We could define this term as (twice) the average of fluxes, although we prefer to call it a jump, since the two terms are computed with normals of opposite sense.

The continuity of fluxes that implies equation (5.11) can be written as

$$0 = [\![\mathcal{F}(y;y)]\!]_E = [\![\mathcal{F}(y;y_r)]\!]_E + [\![\mathcal{F}(y;\breve{y})]\!]_E, \quad \text{in } E \in \mathcal{T}_h. \tag{5.12}$$

Now we make the key approximation

$$[\![\mathcal{F}(y;\breve{y})]\!]_E = \tau_E^{-1} \breve{y}_E, \tag{5.13}$$

where τ_E is a matrix of stabilization parameters to be determined and $\breve{y}_E$ is the single valued subscale on E. This yields the model we shall use:

$$\breve{y}_E = -\tau_E [\![\mathcal{F}(y;y_r)]\!]_E. \tag{5.14}$$

In the case of edges belonging to Neumann boundaries, $E_N = \partial K \cap \Gamma_N$, equation (5.12) needs to be replaced by

$$\mathbf{t}_N = \boldsymbol{\mathcal{F}}(\mathbf{y}; \mathbf{y}_r)|_{E_N} + \boldsymbol{\mathcal{F}}(\mathbf{y}; \check{\mathbf{y}})|_{E_N}, \tag{5.15}$$

and equation (5.14) needs to be replaced by

$$\check{\mathbf{y}}_{E_N} = -\boldsymbol{\tau}_E(\boldsymbol{\mathcal{F}}(\mathbf{y}; \mathbf{y}_r)|_{E_N} - \mathbf{t}_N). \tag{5.16}$$

Remark 5. *The formulation arising from equation (5.14) is equivalent to the one presented in [45] with the element interiors subscales evaluated at the edge E neglected.*

We can include the subscales in the boundaries in the stabilized formulation in equation (5.9), by re-defining the forms $B_h(\mathbf{y}; \mathbf{y}_h, \mathbf{v}_h) = B(\mathbf{y}; \mathbf{y}_h, \mathbf{v}_h) + B_s(\mathbf{y}; \mathbf{y}_h, \mathbf{v}_h) + B_b(\mathbf{y}; \mathbf{y}_h, \mathbf{v}_h)$ and $L_h(\mathbf{y}; \mathbf{v}_h) = L(\mathbf{y}; \mathbf{v}_h) + L_s(\mathbf{y}; \mathbf{v}_h) + L_b(\mathbf{y}; \mathbf{v}_h)$ with

$$\begin{aligned} B_b(\mathbf{y}; \mathbf{y}_h, \mathbf{v}_h) = &- \sum_E \langle [\![\boldsymbol{\mathcal{F}}(\mathbf{y}; \mathbf{y}_h)]\!], \boldsymbol{\tau}_E [\![\boldsymbol{\mathcal{F}}^*(\mathbf{y}; \mathbf{v}_h)]\!] \rangle_E \\ &- \sum_{E_N} \langle \boldsymbol{\mathcal{F}}(\mathbf{y}; \mathbf{y}_h), \boldsymbol{\tau}_E \boldsymbol{\mathcal{F}}^*(\mathbf{y}; \mathbf{v}_h) \rangle_{E_N}, \end{aligned} \tag{5.17a}$$

$$L_b(\mathbf{y}; \mathbf{v}_h) = - \sum_{E_N} \langle \mathbf{t}_N, \boldsymbol{\tau}_E \boldsymbol{\mathcal{F}}^*(\mathbf{y}; \mathbf{v}_h) \rangle_{E_N}. \tag{5.17b}$$

6 REDUCED ORDER MODEL

6.1 VARIATIONAL ROM FORMULATION

To formulate the variational projection-based ROM problem in an analogous way to the FOM problem in section 5.1, we denote the approximation space as $\mathcal{Y}_r \subset \mathcal{Y}$. This way, we can write the variational ROM of the problem in equation (4.3) as: find $\{\mathbf{y}_r^j\} \subset \mathcal{Y}_r$, such that

$$(\boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \mathbf{y}_r, \mathbf{v}_r) + B(\mathbf{y}; \mathbf{y}_r, \mathbf{v}_r) = L(\mathbf{v}_r), \quad \forall \mathbf{v}_r \in \mathcal{Y}_r, \tag{6.1}$$

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 5.1 are inherited. Hence, a stabilization for the ROM is also necessary. By implementing the same VMS method in equation (6.1) we can write a stabilized ROM in the form of equation (5.3) as

$$\begin{aligned}(\boldsymbol{\mathcal{M}}(\hat{\mathbf{y}})\delta_t \mathbf{y}_r, \mathbf{v}_r) + B(\hat{\mathbf{y}}; \mathbf{y}_r, \mathbf{v}_r) \\ + \langle \boldsymbol{\tau}_r(\hat{\mathbf{y}})\boldsymbol{\mathcal{R}}_r(\hat{\mathbf{y}}; \mathbf{y}_r), \boldsymbol{\mathcal{P}}_r(\hat{\mathbf{y}}; \mathbf{v}_r)\rangle = L(\mathbf{v}_r), \quad \forall \mathbf{v}_r \in \mathcal{Y}_r,\end{aligned} \tag{6.2}$$

where, similarly to the FOM case, $\hat{\mathbf{y}}$ is an approximation to $\mathbf{y}$ that may take into account the subscales or not. We will simply write $\mathbf{y}$ in what follows.

6.2 VMS IN THE REDUCED ORDER MODEL

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 $\breve{\mathcal{Y}}$, so that $\mathcal{Y} = \mathcal{Y}_r \oplus \breve{\mathcal{Y}}$, with the unknown and the test functions split as $\mathbf{y} = \mathbf{y}_r + \breve{\mathbf{y}}$ and $\mathbf{v} = \mathbf{v}_r + \breve{\mathbf{v}}$, respectively. Consequently, the ROM problem in equation (6.1) also expands into two, finding $\{\mathbf{y}_r^j\} \subset \mathcal{Y}_r$ and $\{\breve{\mathbf{y}}^j\} \subset \breve{\mathcal{Y}}$, such that

$$\begin{aligned}(\boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \mathbf{y}_r, \mathbf{v}_r) + B(\mathbf{y}; \mathbf{y}_r, \mathbf{v}_r) + (\boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \breve{\mathbf{y}}, \mathbf{v}_r) \\ + B(\mathbf{y}; \breve{\mathbf{y}}, \mathbf{v}_r) = L(\mathbf{v}_r), \quad \forall \mathbf{v}_r \in \mathcal{Y}_r, \end{aligned} \tag{6.3a}$$

$$\begin{aligned}(\boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \mathbf{y}_r, \breve{\mathbf{v}}) + B(\mathbf{y}; \mathbf{y}_r, \breve{\mathbf{v}}) + (\boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \breve{\mathbf{y}}, \breve{\mathbf{v}}) \\ + B(\mathbf{y}; \breve{\mathbf{y}}, \breve{\mathbf{v}}) = L(\breve{\mathbf{v}}), \quad \forall \breve{\mathbf{v}} \in \breve{\mathcal{Y}}. \end{aligned} \tag{6.3b}$$

To avoid terms that involve applying the differential operator $\boldsymbol{\mathcal{L}}$ over the subscales $\breve{\mathbf{y}}$, we follow the same approach as in section 5.2, this way we can re-write equation (6.3) as

$$\begin{aligned}(\boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \mathbf{y}_r, \mathbf{v}_r) + (\boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \breve{\mathbf{y}}, \mathbf{v}_r) + B(\mathbf{y}; \mathbf{y}_r, \mathbf{v}_r) + \sum_K \langle \breve{\mathbf{y}}, \boldsymbol{\mathcal{L}}^*(\mathbf{y}; \mathbf{v}_r)\rangle_K \\ + \sum_K \langle \breve{\mathbf{y}}, \boldsymbol{\mathcal{F}}_n^*(\mathbf{y}; \mathbf{v}_r)\rangle_{\partial K} = L(\mathbf{v}_r), \quad \forall \mathbf{v}_r \in \mathcal{Y}_r, \end{aligned} \tag{6.4a}$$

$$\begin{aligned}\sum_K \langle \boldsymbol{\mathcal{M}}(\mathbf{y})\delta_t \breve{\mathbf{y}}, \breve{\mathbf{v}}\rangle_K + \sum_K \langle \boldsymbol{\mathcal{L}}(\mathbf{y}, \breve{\mathbf{y}}), \breve{\mathbf{v}}\rangle_K + \sum_K \langle \boldsymbol{\mathcal{F}}_n(\mathbf{y}; \breve{\mathbf{y}}), \breve{\mathbf{v}}\rangle_{\partial K} \\ = \sum_K \langle \mathbf{r}(\mathbf{y}; \mathbf{y}_r), \breve{\mathbf{v}}\rangle_K, \quad \forall \breve{\mathbf{v}} \in \breve{\mathcal{Y}}. \end{aligned} \tag{6.4b}$$

These equations have the same structure as equations (5.5a) and (5.5b), 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 $\mathbf{y}_r$ 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.

6.3 ROM SUBSCALES

By following the same supposition of continuous fluxes and using the same approximation for the spatial operator $-\mathcal{L}(\mathbf{y}, \breve{\mathbf{y}}) \approx \boldsymbol{\tau}_K^{-1}(\mathbf{y})\breve{\mathbf{y}}$ in each element K— as in section 5.3, we can write an equation similar to equation (5.6) for the ROM-SGS,

$$\mathcal{M}(\mathbf{y})\delta_t\breve{\mathbf{y}} + \boldsymbol{\tau}_K^{-1}(\mathbf{y})\breve{\mathbf{y}} = \mathbf{r}(\mathbf{y};\mathbf{y}_r) + \breve{\mathbf{v}}^{\perp}, \quad \text{in } K \in \mathcal{T}_h. \tag{6.5}$$

Likewise, to complete the approximation of the subscales, the term $\breve{\mathbf{v}}^{\perp}$ satisfies a similar condition as in equation (5.7) $-\sum_K \langle \breve{\mathbf{v}}^{\perp}, \breve{\mathbf{v}} \rangle_K = 0$, $\forall \breve{\mathbf{v}} \in \breve{\mathcal{Y}}$, and therefore also defined by the choice of the space of ROM subscales $\breve{\mathcal{Y}}$ and its orthogonal complement $\breve{\mathcal{Y}}^{\perp}$. Thus, denoting the L^2 projection onto $\breve{\mathcal{Y}}^{\perp}$ as $\breve{\boldsymbol{\Pi}}^{\perp}$and defining $\breve{\mathbf{v}}^{\perp} = -\breve{\boldsymbol{\Pi}}^{\perp}(\mathbf{r}(\mathbf{y};\mathbf{y}_r))$, we can rewrite equation (6.5) as

$$\mathcal{M}(\mathbf{y})\delta_t\breve{\mathbf{y}} + \boldsymbol{\tau}_K^{-1}(\mathbf{y})\breve{\mathbf{y}} = \breve{\boldsymbol{\Pi}}(\mathbf{r}(\mathbf{y};\mathbf{y}_r)), \quad \text{in } K \in \mathcal{T}_h, \tag{6.6}$$

where similarly to equation (5.8) for the FOM-SGS, $\breve{\boldsymbol{\Pi}} = \mathbf{I} - \breve{\boldsymbol{\Pi}}^{\perp}$ is the projection onto the subscale space $\breve{\mathcal{Y}}$ and $\mathbf{I}$ is the identity.

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

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

By defining $\mathcal{P}_r(y, v_r) = \mathcal{L}^*(y, v_r)$, and choosing $\tau_r(y)$ as $\tau_K(y)$ or $\tau_{K,t}(y)$, and $\mathcal{R}_r(y; y_r)$ as $r(y; y_r)$ or $\Pi_r^\perp(f - \mathcal{L}(y; y_r))$, we can obtain a general stabilized approximation for y_r of the form of equation (6.2). Moreover, we can rewrite this equation as

$$(\mathcal{M}(y)\delta_t y_r, v_r) + B_r(y; y_r, v_r) = L_r(v_r), \quad \forall v_r \in \mathcal{Y}_r. \tag{6.7}$$

In the same way as for the FOM, we write 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, we define the stabilization terms $B_s(y; y_r, v_r)$ and $L_s(y; v_r)$ for each proposed case in table 6.1 and table 6.2, respectively.

	ASGS
Dynamic	$-\sum_K \langle \mathcal{M}(y)\delta_t y_r + \mathcal{L}(y; y_r), \tau_{K,t}\mathcal{L}^*(y, v_r)\rangle_K$
	$-\sum_K \langle \mathcal{M}(y)\delta_t y_r + \mathcal{L}(y; y_r), [I - \tau_K^{-1}\tau_{K,t}]v_r\rangle_K$
Quasi-static	$-\sum_K \langle \mathcal{M}(y)\delta_t y_r + \mathcal{L}(y; y_r), \tau_K\mathcal{L}^*(y, v_r)\rangle_K$
	OSGS
Dynamic	$-\sum_K \langle \Pi_r^\perp(\mathcal{L}(y; y_r)), \tau_{K,t}\mathcal{L}^*(y, v_r)\rangle_K$
Quasi-static	$-\sum_K \langle \Pi_r^\perp(\mathcal{L}(y; y_r)), \tau_K\mathcal{L}^*(y, v_r)\rangle_K$

Table 6.1: ROM $B_s(y; y_r, v_r)$ defined for dynamic and quasi-static ASGS and OSGS.

	ASGS
Dynamic	$-\sum_K \langle f, \tau_{K,t}\mathcal{L}^*(y, v_r) + [I - \tau_K^{-1}\tau_{K,t}]v_r\rangle_K$
	$-\sum_K \langle \delta t^{-1}\mathcal{M}(y)\breve{y}^{j-1}, \tau_{K,t}\mathcal{L}^*(y, v_r)\rangle_K$
	$+\sum_K \langle \delta t^{-1}\mathcal{M}(y)\breve{y}^{j-1}, \tau_K^{-1}\tau_{K,t}v_r\rangle_K$
Quasi-static	$-\sum_K \langle f, \tau_K\mathcal{L}^*(y, v_r)\rangle_K$
	OSGS
Dynamic	$-\sum_K \langle \Pi_r^\perp(f) + \delta t^{-1}\mathcal{M}(y)\breve{y}^{j-1}, \tau_{K,t}\mathcal{L}^*(y, v_r)\rangle_K$
Quasi-static	$-\sum_K \langle \Pi_r^\perp(f), \tau_K\mathcal{L}^*(y, v_r)\rangle_K$

Table 6.2: ROM $L_s(y; v_r)$ defined for dynamic and quasi-static ASGS and OSGS.

The term $\Pi_r^\top(\mathcal{L}(y;y_r)-f)$ can be computed in a way similar to the FOM term $\Pi_h^\top(\mathcal{L}(y;y_h)-f)$, as explained in remark 4. The problem has to be completed with initial conditions for the subscales. We assume that $\breve{y}^0=0$, for all $x\in\Omega$, which means that we assume for simplicity that the initial condition belongs to the space of resolvable scales.

It is important to note 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 6. *The non-linearity in the operators $\mathcal{M}(y)$, $\mathcal{L}(y,y_r)$, $\mathcal{L}^*(y,y_r)$, $\mathcal{F}(y,y_r)$ and $\mathcal{F}^*(y,y_r)$ is treated using $y=y_r$, which corresponds to the linear subscale choice of the FOM.*

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 $\{\boldsymbol{\phi}^k\}_{k=1}^{Nn}$, so that $\mathcal{Y}_r=\text{span}\{\boldsymbol{\phi}^k\}_{k=1}^{r}$. When OSGS are used, we have the following explicit representation of the ROM-SGS:*

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

6.4 ROM SUBSCALES ON THE BOUNDARIES

In the same way as for the FOM in section 5.4, we approximate the subscales in the boundaries by balancing the fluxes over the edges and defining them as single valued, yielding

$$0=[\![\mathcal{F}(y;y_r)]\!]_E+[\![\mathcal{F}(y;\breve{y})]\!]_E,\quad \text{in } E\in\mathcal{T}_h. \tag{6.8}$$

And using the same type of approximation for the subscales on the boundaries

$$\breve{y}_E=-\tau_E[\![\mathcal{F}(y;y_r)]\!]_E. \tag{6.9}$$

In the same way as for the FOM, we can include the subscales in the boundaries in the stabilized formulation in equation (6.7), by re-defining the forms $B_h(y; y_h, v_h) = B(y; y_h, v_h) + B_s(y; y_h, v_h) + B_b(y; y_h, v_h)$ and $L_h(y; v_h) = L(y; v_h) + L_s(y; v_h) + L_b(y; v_h)$ with

$$B_b(y; y_h, v_h) = -\sum_E \langle [\![\mathcal{F}(y; y_h)]\!], \tau_E [\![\mathcal{F}^*(y; v_h)]\!] \rangle_E - \sum_{E_N} \langle \mathcal{F}(y; y_h), \tau_E \mathcal{F}^*(y; v_h) \rangle_{E_N}, \tag{6.10a}$$

$$L_b(y; v_h) = -\sum_{E_N} \langle t_N, \tau_E \mathcal{F}^*(y; v_h) \rangle_{E_N}. \tag{6.10b}$$

6.5 STABILIZED FORMULATION

By using orthogonal subscales in the element interiors (using the OSGS formulation in tables 6.1 and 6.2) and the subscales on the boundaries given by equation (6.9), we can write the stabilized VMS-ROM approximation we propose as

$$\begin{aligned}
&(\mathcal{M}(y)\delta_t y_r, v_r) + B(y; y_r, v_r) \\
&\quad - \sum_K \langle \check{\Pi}(\mathcal{L}(y; y_r)), \tau_{K,t}(y) \mathcal{L}^*(y, v_r) \rangle_K \\
&\quad - \sum_E \langle [\![\mathcal{F}(y; y_r)]\!], \tau_E [\![\mathcal{F}^*(y; v_r)]\!] \rangle_E \\
&\quad - \sum_{E_N} \langle \mathcal{F}(y; y_r), \tau_E \mathcal{F}^*(y; v_r) \rangle_{E_N} = L(y; v_r) \\
&\quad - \sum_K \langle \check{\Pi}(f) + \delta t^{-1} \mathcal{M}(y) \check{y}^{j-1}, \tau_{K,t}(y) \mathcal{L}^*(y, v_r) \rangle_K \\
&\quad - \sum_{E_N} \langle t_N, \tau_E \mathcal{F}^*(y; v_r) \rangle_{E_N}, \quad \forall v_r \in \mathcal{Y}_r, \text{ at } t^j, \ j = 1, 2, \dots
\end{aligned} \tag{6.11}$$

6.6 BEHAVIOR OF THE STABILIZATION PARAMETERS

To define the stabilization parameter $\tau_K(y)$ in the stabilized ROM formulation, we follow a Fourier analysis of the subscale equation as the one done in [37] 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

$$-\nu\Delta y + a\cdot\nabla y + \sigma y = f \qquad \text{in } \Omega, \tag{6.12}$$

with a a constant convection velocity, and $\nu > 0$ and $\sigma \geqslant 0$ the diffusion and reaction coefficients, respectively.

Following equation (6.4b), we can state a subscale equation as

$$-\nu\Delta\breve{y} + a\cdot\nabla\breve{y} + \sigma\breve{y} = \mathcal{R}_r(y_r) \qquad \text{in } K \in \mathcal{T}_h, \tag{6.13}$$

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 [37], showing that they are also valid for the ROM problem because it is also based on the existence of the FE mesh (see chapter 1).

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

$$\widehat{g}(k) := \int_K e^{-ik\cdot x} g(x)\, d\Omega_x, \tag{6.14a}$$

$$\widehat{\frac{\partial g}{\partial x_j}}(k) = \int_{\partial K} n_j e^{-ik\cdot x} g(x)\, d\Gamma_x + ik_j\widehat{g}(k), \tag{6.14b}$$

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 $\breve{y}$ 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 equation (6.14b) 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

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

Finally, taking the Fourier transform of equation (6.13) and By defining the dimensionless wave number $\boldsymbol{\kappa} := h\mathbf{k}$, where h is a characteristic length that can be defined by the initial discretization given by the partition $\mathcal{T}_h$, we get

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

from which, using Plancherel's formula and the mean value theorem, we can identify the approximation τ_K in section 6.3 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:

$$\tau_K = \left[\left(c_1\frac{\nu}{h^2}\right)^2 + \left(c_2\frac{|\mathbf{a}|}{h}\right)^2 + \sigma^2\right]^{-1/2}, \tag{6.15}$$

where c_1 and c_2 are constants independent of ν, σ, $\mathbf{a}$ 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 6.1 shows the first 6 modes of the basis obtained from the convection-diffusion-reaction problem in chapter 11.

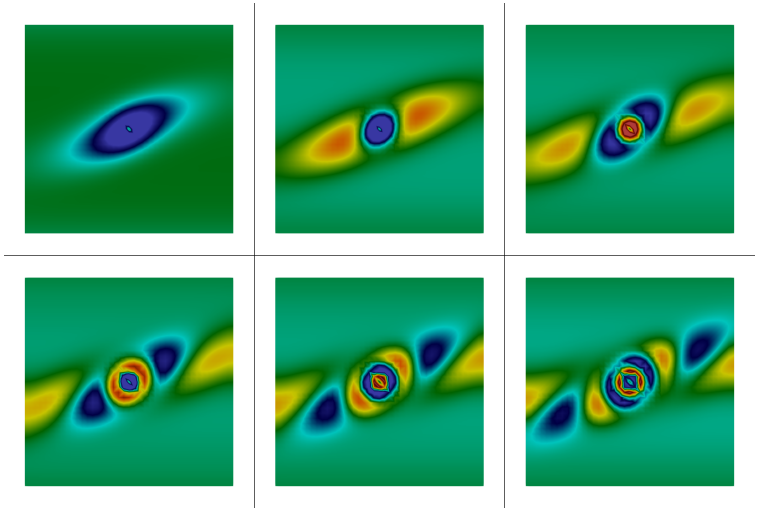


Figure 6.1: First 6 modes of the basis obtained from a convection-diffusion-reaction problem embedded in a Couette flow.

6.6.1 *Stabilization parameters for the boundary subscales*

To define the stabilization parameter $\tau_E(y)$ for the subscales on the boundaries we can use a similar argument as for the element interiors in section 6.6. Using the same convection-diffusion-reaction problem in equation (6.12) with the boundary flux defined as

$$\mathcal{F}(y; y_r) = n \cdot (\nu \nabla \breve{y}) \quad \text{in } \Gamma, \tag{6.16}$$

we can write the stabilization parameter for the boundary subscales as

$$\tau_E = \left(c_E \frac{\nu}{h} \right)^{-1} \tag{6.17}$$

where c_E is an algorithmic constant independent of ν and h.

Remark 9. *The parameter τ_E is similar to the one obtained in [45] where it is defined using a direct approximation of the fluxes over the edge. There the constant c_E is associated with the fraction of h in which the subscale on the edge is taken into account inwards the element.*

6.7 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 analyzed, which is beyond the scope of this work. A brief description of the main features of the numerical analysis of VMS-based FEs can be found in [41]. 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, $|y_r|$, $y_r \in \mathcal{Y}_r$. 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.,

$$|y - y_r| \leqslant C\, E_{I,r}(h), \quad E_{I,r}(h) := \inf_{z_r \in \mathcal{Y}_r} |y - z_r|, \tag{6.18}$$

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 (6.18), 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 $|y - y_r| \leqslant |y - z_r| + |z_r - y_r|$ for any $z_r \in Y_r$. Taking the infimum for $z_r \in Y_r$, 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

$$|y - y_h| \leqslant C\, E_{I,h}(h), \quad E_{I,h}(h) := \inf_{z_h \in Y_h} |y - z_h|. \tag{6.19}$$

The literature about the behavior of $E_{I,h}(h)$ in equation (6.19) is vast. However, the estimate in equation (6.18) is usually expressed in terms of the eigenvalues not taken into account in the Proper Orthogonal Decomposition (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 (6.18) is optimal, in the sense that the error of the method is bounded by the interpolation error, which is the best one can expect.

7 OTHER STABILIZATION METHODS

Using equation (5.7), the definitions in tables 5.1 and 5.2, and writing its stabilization part as $\langle \boldsymbol{\tau}_r(\mathbf{y})\boldsymbol{\mathcal{R}}_r(\mathbf{y};\mathbf{y}_r), \boldsymbol{\mathcal{P}}_r(\mathbf{y};\mathbf{v}_r)\rangle$ we can describe our proposed VMS method with time dependent OSGS as

- a full residual method, $\boldsymbol{\mathcal{R}}_r$ depends on the residual $\mathbf{r}$,
- with orthogonality between the subscale space $\breve{\mathcal{Y}}$ and the resolved scale space $\mathcal{Y}_r$, $\breve{\boldsymbol{\Pi}}$ is defined as the orthogonal projection over the resolved scales, $\boldsymbol{\Pi}_r^{\perp}$,
- with control over all the skew-symmetric terms of $\boldsymbol{\mathcal{L}}$, $\boldsymbol{\mathcal{P}}_r$ 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) [21, 56, 83], VMS approaches [13, 21, 98], term by term stabilization methods [8, 67, 68, 73, 90], and some empirical methods [1, 14, 16, 70, 105]. 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 $\boldsymbol{\mathcal{R}}_r(\mathbf{y};\mathbf{y}_r) = \mathbf{r}(\mathbf{y};\mathbf{y}_r)$, $\breve{\boldsymbol{\Pi}} = \mathbf{I}$ and $\boldsymbol{\mathcal{P}}_r(\mathbf{y};\mathbf{v}_r) = \mathbf{A}_i^c(\mathbf{y})^\top \partial_i \mathbf{v}_r$, 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 ASGS stabilization described in tables 5.1 and 5.2. And finally, by defining $\boldsymbol{\mathcal{R}}_r(\mathbf{y};\mathbf{y}_r) = \mathbf{A}_i^c(\mathbf{y})\partial_i \mathbf{y}_r$, $\breve{\boldsymbol{\Pi}} = \boldsymbol{\Pi}_r^{\perp}$ and $\boldsymbol{\mathcal{P}}_r(\mathbf{y};\mathbf{v}_r) = \mathbf{A}_i^c(\mathbf{y})^\top \partial_i \mathbf{v}_r$, we can describe the term-by-term stabilization methods 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 [54, 72] for FEs.

Along with the definitions of $\mathcal{R}_r$ and $\mathcal{P}_r$ we can classify the matrix of stabilization parameters τ_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 τ_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.

In table 7.1 we summarize some of the stabilization methods for ROMs using the aforementioned classifying criteria for $\mathcal{P}_r$, $\mathcal{R}_r$ and τ_r. We exclude the purely empirical methods.

$\mathcal{R}_r$	Residual	[13, 21, 56, 98]
	Term-by-term	[8, 67, 68, 90]
	Orthogonal	[8, 67, 68, 90]
	Non-Orthogonal	[13, 21, 56, 98]
$\mathcal{P}_r$	Complete $\mathcal{L}^*$	[13, 21, 98]
	Convective term	[8, 56, 67, 68, 90]
τ_r	τ_h	[13, 21, 56, 90, 98]
	Empirical	[8, 67, 68]

Table 7.1: Comparison of the main characteristics of stabilization methods for ROMs.

7.1 TERM-BY-TERM STABILIZATION

In this section, we describe the OSGS term-by-term stabilization proposed in [28, 38] for a FE case, which results in a generalized version of the methods proposed in [8, 67, 68, 90] to stabilize ROMs. In section 12.1 we present a comparison between this method and the OSGS we proposed in chapter 6.

The method consist in splitting the subscales as $\breve{y} = \sum_l \breve{y}_l$, which is done considering the stabilization terms as orthogonal between each other. That is, each term of the linear operator is stabilized with its adjoint counterpart, neglecting the cross terms.

By writing the operators $\mathcal{P}$ and $\mathcal{R}$ as

$$\mathcal{R}(y; y_r) = \begin{bmatrix} \Pi_r^{\perp}(A_i^c(y)\partial_i y_r) \\ \Pi_r^{\perp}(A_i^f(y)\partial_i y_r) \\ -\Pi_r^{\perp}(\partial_i(K_{ij}(y)\partial_j y_r)) \\ \Pi_r^{\perp}(S(y)y_r) \end{bmatrix},$$

$$\mathcal{P}(y; v_r) = \begin{bmatrix} -\partial_i A_i^c(y)^{\top} v_r \\ -\partial_i A_i^f(y)^{\top} v_r \\ -\partial_i\left(K_{ji}^{\top}(y)\partial_j v_r\right) \\ S(y)^{\top} v_r \end{bmatrix},$$

we can define the subscales as $\partial_t \breve{y}_l + \tau_r^{-1}\breve{y}_l = \mathcal{R}_r^l(y; y_r)$. Thus, getting an OSGS term-by-term formulation where the stabilization forms in equation (6.7) are defined as

$$B_s(y; y_r, v_r) = -\sum_l \langle \mathcal{R}_r^l(y; y_r), \tau_t \mathcal{P}_r^l(y; v_r))\rangle_K,$$

$$L_s(y; v_r) = -\sum_l \langle \delta t^{-1}\breve{y}_l, \tau_t \mathcal{P}_r^l(y; v_r)\rangle_K,$$

with the stabilization parameters τ_r defined in the same way as for the original OSGS formulation.

Part III

ADDITIONAL TECHNIQUES

8 PETROV-GALERKIN PROJECTION

For the Full Order Model (FOM) case, the formulation we propose is given by equation (5.5a) with the subscales $\tilde{\mathbf{y}}$ obtained from equation (5.8), (see section 5.3), leading to a matrix problem of the form

$$\mathbf{A}_h(\mathbf{y}_h)\mathbf{y}_h = \mathbf{b}_h, \tag{8.1}$$

for each time step, 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 Reduced Order Model (ROM) case is given by equation (6.4a) with the subscales $\check{\mathbf{y}}$ obtained from equation (6.6), with the options described in section 6.3. Which leads at each time step to the matrix form of the problem

$$\mathbf{A}_r(\mathbf{y}_r)\mathbf{y}_r = \mathbf{b}_r, \tag{8.2}$$

where $\mathbf{A}_r$ is a matrix of size $r \times r$, and $\mathbf{y}_r$ and $\mathbf{b}_r$ are arrays of r components, which include, as in equation (8.1), the approximation of the temporal derivative in the left hand side of the equation, and the forcing terms and the previous values of the ROM solution and the subscales in the right hand side.

Following chapter 3 and denoting $\mathbf{\Phi}$ the $Nn \times r$ matrix whose r columns are the nodal values of the basis functions of the ROM space, we can consider the FOM solution as approximated by $\mathbf{y}_h \approx \mathbf{\Phi}\mathbf{y}_r$.

Furthermore, if the FOM test function is taken in the ROM subspace, equation (8.1) yields

$$\boldsymbol{\Phi}^\top \mathbf{A}_\mathrm{h}(\boldsymbol{\Phi}\mathbf{y}_\mathrm{r})\boldsymbol{\Phi}\mathbf{y}_\mathrm{r} = \boldsymbol{\Phi}^\top \mathbf{b}_\mathrm{h}. \tag{8.3}$$

This equation can be considered the FOM one projected onto the ROM subspace. It coincides with equation (8.2) 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. Bear in mind that for nonlinear problems, differences may arise depending on the treatment of the non-linearities.

If we consider that the ROM is the projected FOM given by equation (8.3), we observe that it corresponds to the so called Galerkin projection in [93]. Following [93, Chapter 5], we can also consider the Petrov–Galerkin projection proposed in [26] as the appropriate selection of subspaces to solve the discrete problem using a projection method with best conditioning. Omitting the dependence of $\mathbf{A}_\mathrm{h}$ on $\boldsymbol{\Phi}\mathbf{y}_\mathrm{r}$, in our problem this Petrov–Galerkin projection would lead to the algebraic problem

$$\boldsymbol{\Phi}^\top \mathbf{A}_\mathrm{h}^\top \mathbf{A}_\mathrm{h} \boldsymbol{\Phi}\mathbf{y}_\mathrm{r} = \boldsymbol{\Phi}^\top \mathbf{A}_\mathrm{h}^\top \mathbf{b}_\mathrm{h}. \tag{8.4}$$

Let us comment on the need of using the Petrov–Galerkin projection in equation (8.4) instead of the Galerkin projection in equation (8.3). For linear problems we have experimentally found that they both yield virtually the same results, and therefore equation (8.3) —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 (8.4)) 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 Variational Multi-Scale (VMS) approach are crucial to get stable and accurate solutions.

9 DOMAIN DECOMPOSITION

In Finite Element (FE) problems, when a domain decomposition strategy is used to couple two (or more) subdomains, either for homogeneous or heterogeneous problems, one needs to carefully design the transmission conditions. First, if the interpolation is assumed to be discontinuous across subdomains one has to introduce terms to account for this discontinuity, in the spirit of the Discontinuous Galerkin (DG) method. Furthermore, if an iteration-by-subdomain is employed, the iterative scheme may be made more robust by the introduction of boundary terms that may be understood as subscales on the element boundaries in the VMS context (see [42]). We extend this well understood fact —in the FE context, to the FE-based ROM.

Domain decomposition schemes for ROM have been implemented in different areas, as part of hybrid ROM-FOM methods [12, 22, 47, 71, 78], as part of the physical model (e.g. fluid structure interaction cases [100] or coupled Stokes-Darcy flows [82]) and as part of local ROMs (using local bases computed separately for different regions of the domain [1, 65, 66, 86] or as a partitioned global basis [12]).

By reviewing these techniques, we can argue that at least two sources of instability might appear when formulating a domain decomposition ROM: the discontinuities that appear naturally from the transmission conditions in the subdomain interfaces and the discontinuity in the local bases. In the same way as the instabilities that arise from the problem and space reduction are tackled with the VMS simultaneously; we aim to solve both problems by adding the boundary SubGrid Scales (SGS), i.e., the components of the solution that cannot be captured by the finite element mesh and that need to be approximated somehow.

In this chapter, we implement the stabilized VMS-ROM formulation presented in chapter 6 to a non-overlapping domain decomposition problem as done for the FE counterpart in [42]. For simplicity we consider it without Neumann boundary conditions.

9.1 PROBLEM STATEMENT

Suppose that the domain Ω is split as $\bar{\Omega} = \bar{\Omega}_1 \cup \bar{\Omega}_2$, with $\Gamma = \bar{\Omega}_1 \cap \bar{\Omega}_2$. Considering the fluxes in Γ as unknowns and enforcing continuity weakly, we can write the problem as finding $\mathbf{y}_1$, $\mathbf{y}_2$ and $\mathbf{t}_\Gamma$ such that

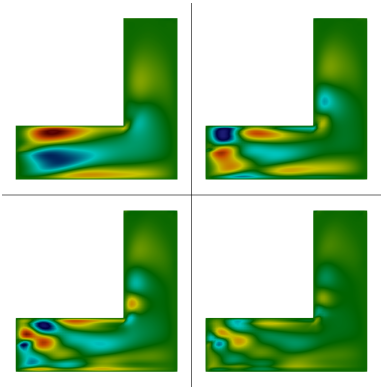
$$
\begin{aligned}
\mathcal{M}_1(\mathbf{y}_1)\partial_t \mathbf{y}_1 + \mathcal{L}_1(\mathbf{y}_1;\mathbf{y}_1) &= \mathbf{f}, && \text{in } \Omega_1, && t \in]0, t_f[, && \\
\mathcal{D}\mathbf{y}_1 &= \mathcal{D}\mathbf{y}_{D_1} && \text{on } \Gamma_{D_1}, && t \in]0, t_f[, && \\
\mathcal{M}_2(\mathbf{y}_2)\partial_t \mathbf{y}_2 + \mathcal{L}_2(\mathbf{y}_2;\mathbf{y}_2) &= \mathbf{f}, && \text{in } \Omega_2, && t \in]0, t_f[, && \\
\mathcal{D}\mathbf{y}_2 &= \mathcal{D}\mathbf{y}_{D_2} && \text{on } \Gamma_{D_2}, && t \in]0, t_f[, && \\
\mathcal{D}\mathbf{y}_1 &= \mathcal{D}\mathbf{y}_2 && \text{on } \Gamma, && t \in]0, t_f[, && \text{(9.1a)} \\
\mathcal{F}_1(\mathbf{y}_1;\mathbf{y}_1) &= \mathbf{t}_\Gamma && \text{on } \Gamma, && t \in]0, t_f[, && \text{(9.1b)} \\
\mathcal{F}_2(\mathbf{y}_2;\mathbf{y}_2) &= -\mathbf{t}_\Gamma && \text{on } \Gamma, && t \in]0, t_f[, && \text{(9.1c)}
\end{aligned}
$$

where Γ_{D_i} is the boundary of Ω_i excluding Γ with Dirichlet data $\mathbf{y}_{D_i}$, $i = 1, 2$. Variables and operators with subscript i correspond to their global counterpart restricted to Ω_i. Obviously, what follows can be easily extended to several interacting subdomains.

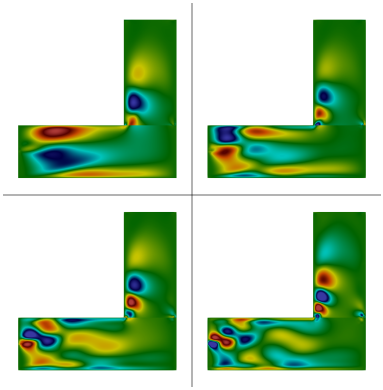
The transmission conditions (9.1a)-(9.1c) already suggest the possibility of using a standard Dirichlet-Neumann coupling. However, rather than this we will consider the problem posed in the whole domain Ω and allowing for a discontinuity in the interpolation of the unknowns over Γ. To account for this discontinuity, we will introduce the terms derived in [40] for discontinuous Galerkin methods, now applied to our ROM, as well as the boundary stabilization terms described in the previous section.

Let us remark that the ROM basis can be computed globally or locally, independently for each subdomain (see [12]). Although it is evident that the local approach yields discontinuous bases, the interpolation of the global approach may also be discontinuous if different degrees of freedom are taken at each side of the interface.

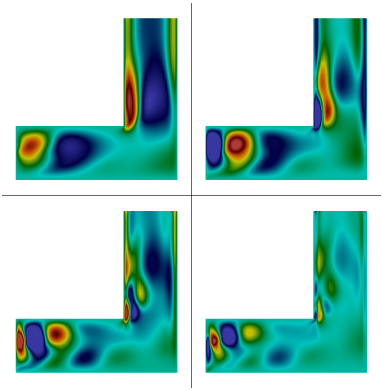
To illustrate the discontinuity in the basis when it is calculated in a local way —that is, one basis per subdomain— in figure 9.1 we show a comparison between the bases obtained using a global domain and local subdomains in an incompressible Navier-Stokes L-shaped flow problem (section 12.3).



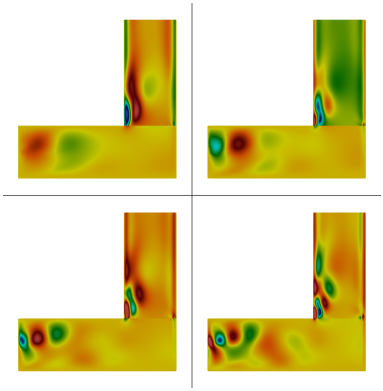
(a) x-Velocity global basis.



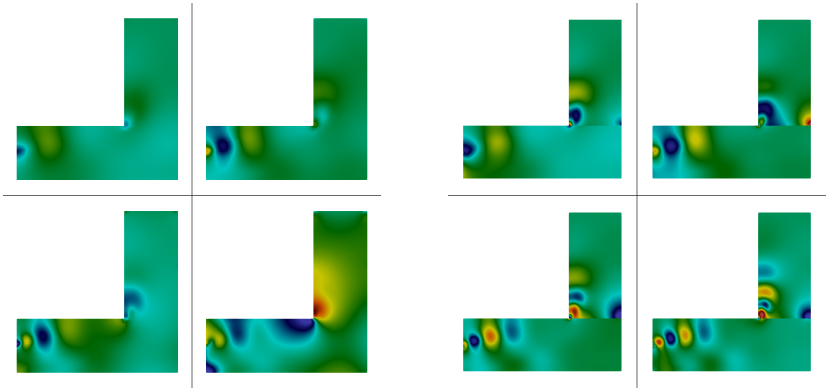
(b) x-Velocity local bases.



(c) y-Velocity global basis.



(d) y-Velocity local bases.



(e) Pressure global basis. (f) Pressure local bases.

Figure 9.1: Modes 2 to 5 of the global and local bases obtained from an incompressible Navier-Stokes flow in an L-shaped domain.

9.2 TREATMENT OF THE DISCONTINUITY ACROSS THE INTERFACE

Let us consider the continuous problem (4.3), but now considering that there is a discontinuity across the interface Γ. Imposing weakly the continuity condition (9.1a) *à la* Nitsche, problem (4.3) can be written as

$$\sum_{i=1}^{2} [(\mathcal{M}_i(y_i)\partial_t y_i, v_i) + B_i(y_i; y_i, v_i)] + \alpha\langle[\![\mathcal{D}y]\!], [\![\mathcal{D}v]\!]\rangle_\Gamma + \langle[\![\mathcal{D}y]\!], [\![\mathcal{F}^*(y; v)]\!]\rangle_\Gamma + \langle[\![\mathcal{F}(y; y)]\!], [\![\mathcal{D}v]\!]\rangle_\Gamma = \sum_{i=1}^{2} L_i(v_i), \quad (9.2)$$

which must hold for all test functions v_i, $i = 1, 2$, in the appropriate spaces. In this expression, α is a penalty-like parameter (that scales as $1/h$ in the discrete problem) and the jump of the Dirichlet operator $[\![\mathcal{D}y]\!]$ is understood to be computed with a fixed normal to Γ. The term $\langle[\![\mathcal{F}(y; y)]\!], [\![\mathcal{D}v]\!]\rangle_\Gamma$ is obtained from integration by parts in each subdomain and the (consistent) term $\langle[\![\mathcal{D}y]\!], [\![\mathcal{F}^*(y; v)]\!]\rangle_\Gamma$ is introduced to ensure adjoint consistency.

9.3 VMS-ROM APPROXIMATION

We are now in a position to combine the treatment of the discontinuous interface in equation (4.3) with the stabilized formulation given by equation (6.11). To simplify the writing, we shall not use the expanded form of the subscales in the element interiors and on the element boundaries, except for those on Γ. The problem to be solved consists in finding sequences $\{y_{i,r}^j\}$, $i = 1, 2$, in the appropriate ROM spaces, such that

$$\begin{aligned}
&\sum_{i=1}^{2}\Big[(\mathcal{M}_i(y_i)\delta_t y_{i,r}, v_{i,r}) + B_i(y_i; y_{i,r}, v_{i,r}) \\
&\quad + \sum_{K_i}\langle \check{y}_i, \mathcal{L}^*(y_i; v_{i,r})\rangle_{K_i} + \sum_{E_i}\langle \check{y}_{i,E}, [\![\mathcal{F}^*(y; v_r)]\!]\rangle_{E_i}\Big] \\
&\quad - \langle [\![\mathcal{F}(y; y_r)]\!], \tau_E [\![\mathcal{F}^*(y; v_r)]\!]\rangle_\Gamma + \alpha\langle [\![\mathcal{D} y_r]\!], [\![\mathcal{D} v_r]\!]\rangle_\Gamma \\
&\quad + \langle [\![\mathcal{D} y_r]\!], [\![\mathcal{F}^*(y; v_r)]\!]\rangle_\Gamma + \langle [\![\mathcal{F}(y; y_r)]\!], [\![\mathcal{D} v_r]\!]\rangle_\Gamma \\
&\quad = \sum_{i=1}^{2} L_i(v_{i,r}), \quad \forall v_{i,r}, i = 1, 2 \text{ att}^j,\ j = 1, 2, \ldots.
\end{aligned} \tag{9.3}$$

In this expression, $\{K_i\}$ denotes the elements of the underlying FE partition in Ω_i and $\{E_i\}$ the *interior* edges.

A word of explanation is needed concerning the calculation of integrals over Γ when the FE meshes in Ω_1 and in Ω_2 do not coincide at this interface. Let us consider for example the term

$$\begin{aligned}
&\langle [\![\mathcal{F}(y; y_r)]\!], \tau_E [\![\mathcal{F}^*(y; v_r)]\!]\rangle_\Gamma \\
&\quad = \langle \mathcal{F}_1(y_1; y_{1,r}), \tau_E \mathcal{F}_1^*(y_1; v_{1,r})\rangle_\Gamma \\
&\quad + \langle \mathcal{F}_2(y_2; y_{2,r}), \tau_E \mathcal{F}_2^*(y_2; v_{2,r})\rangle_\Gamma \\
&\quad + \langle \mathcal{F}_1(y_1; y_{1,r}), \tau_E \mathcal{F}_2^*(y_2; v_{2,r})\rangle_\Gamma \\
&\quad + \langle \mathcal{F}_2(y_2; y_{2,r}), \tau_E \mathcal{F}_1^*(y_1; v_{1,r})\rangle_\Gamma.
\end{aligned} \tag{9.4}$$

First of all, a characteristic element length h needs to be defined to compute τ_E; in the case of different element sizes from both sides of

Γ, we choose the largest h. The first two terms in the last expression offer no difficulty, since Γ can be discretized using either the mesh of Ω_1 or of Ω_2 and numerical integration can be performed in the usual manner. The difficulty arises in the 'cross' terms. Consider for example $\langle \mathcal{F}_1(y_1; y_{1,r}), \tau_E \mathcal{F}_2^*(y_2; v_{2,r})\rangle_\Gamma$. This term appears when $v_{2,r} \neq 0$, i.e., when obtaining the discrete equations for the unknowns in Ω_2 and therefore the mesh of this domain needs to be used. Since $\mathcal{F}_1(y_1; y_{1,r})$ can be computed in the mesh of Ω_1, some sort of interpolation of these boundary values to the edges of the mesh of Ω_2 on Γ will be required to evaluate the term we are considering. Analogous comments apply to the last term in equation (9.4).

9.4 ITERATION-BY-SUBDOMAIN STRATEGY

In this work we approach the domain decomposition problem using an iteration-by-subdomain strategy, where we compute the unknowns in each subdomain assuming the data from the other known, and proceeding iteratively until convergence. This general idea encompasses in particular the classical Dirichlet-Neumann coupling, but there are also other possibilities. Likewise, the iteration due to the non-linearity of the problem can be dealt with at the same time as the domain decomposition coupling or through a nested strategy.

The equation to be solved in Ω_1 is obtained by taking $v_{1,r} \neq 0$, $v_{2,r} = 0$ in equation (9.3). Assuming the non-linearity coupled with the iteration-by-subdomain and using a superscript in parenthesis to denote the iteration counter, k being the current iteration, this equation can be written as

$$
\begin{aligned}
&(\mathcal{M}_1(y_1^{(k-1)})\delta_t y_{1,r}^{(k)}, v_{1,r}) + B_1\left(y_1^{(k-1)}; y_{1,r}^{(k)}, v_{1,r}\right) \\
&\quad + \sum_{K_1} \langle \check{y}_1^{(k)}, \mathcal{L}^*(y_1^{(k-1)}; v_{1,r})\rangle_{K_1} + \sum_{E_1} \langle \check{y}_{1,E}^{(k)}, \mathcal{F}^*(y_1^{(k-1)}; v_{1,r})\rangle_{E_1} \\
&\quad - \langle \mathcal{F}(y_1^{(k-1)}; y_{1,r}^{(l)}), \tau_E \mathcal{F}^*(y_1^{(k-1)}; v_{1,r})\rangle_\Gamma
\end{aligned}
$$

$$
\begin{aligned}
&- \langle \boldsymbol{\mathcal{F}}(\mathfrak{y}_1^{(k-1)}; \mathfrak{y}_{1,r}^{(l)}), \tau_E \boldsymbol{\mathcal{F}}^*(\mathfrak{y}_1^{(k-1)}; \mathfrak{v}_{1,r}) \rangle_\Gamma \\
&+ \alpha \langle \boldsymbol{\mathcal{D}} \mathfrak{y}_{1,r}^{(m)} - \boldsymbol{\mathcal{D}} \mathfrak{y}_{2,r}^{(k-1)}, \boldsymbol{\mathcal{D}} \mathfrak{v}_{1,r} \rangle_\Gamma \\
&+ \langle \boldsymbol{\mathcal{F}}(\mathfrak{y}_1^{(k-1)}; \mathfrak{y}_{1,r}^{(l)}) + \boldsymbol{\mathcal{F}}(\mathfrak{y}_2^{(k-1)}; \mathfrak{y}_{2,r}^{(k-1)}), \boldsymbol{\mathcal{D}} \mathfrak{v}_{1,r} \rangle_\Gamma \\
&+ \langle \boldsymbol{\mathcal{D}} \mathfrak{y}_{1,r}^{(m)} - \boldsymbol{\mathcal{D}} \mathfrak{y}_{2,r}^{(k-1)}, \boldsymbol{\mathcal{F}}^*(\mathfrak{y}_1^{(k-1)}; \mathfrak{v}_{1,r}) \rangle_\Gamma = L_1(\mathfrak{v}_{1,r}),
\end{aligned}
\tag{9.5}
$$

where the indexes l and m determine the type of coupling to be used. For example, for $l = k$, $m = k-1$ subdomain 1 would use the Neumann datum from subdomain 2, whereas for $l = k-1$, $m = k$ it would use the Dirichlet condition from subdomain 2. Nevertheless, a combined condition, with $l = m = k$, could also be used.

The equation for Ω_2, obtained by taking $\mathfrak{v}_{2,r} \neq 0$, $\mathfrak{v}_{1,r} = 0$ in equation (9.3), is

$$
\begin{aligned}
&(\boldsymbol{\mathcal{M}}_2(\mathfrak{y}_2^{(k-1)}) \delta_t \mathfrak{y}_{2,r}^{(k)}, \mathfrak{v}_{2,r}) + B_2\left(\mathfrak{y}_2^{(k-1)}; \mathfrak{y}_{2,r}^{(k)}, \mathfrak{v}_{2,r}\right) \\
&\quad + \sum_{K_2} \langle \breve{\mathfrak{y}}_2^{(k)}, \boldsymbol{\mathcal{L}}^*(\mathfrak{y}_2^{(k-1)}; \mathfrak{v}_{2,r}) \rangle_{K_2} \\
&\quad + \sum_{E_2} \langle \breve{\mathfrak{y}}_{2,E}^{(k)}, \boldsymbol{\mathcal{F}}^*(\mathfrak{y}_2^{(k-1)}; \mathfrak{v}_{2,r}) \rangle_{E_2} \\
&\quad - \langle \boldsymbol{\mathcal{F}}(\mathfrak{y}_1^{(n)}; \mathfrak{y}_{1,r}^{(n)}), \tau_E \boldsymbol{\mathcal{F}}^*(\mathfrak{y}_2^{(k-1)}; \mathfrak{v}_{2,r}) \rangle_\Gamma \\
&\quad - \langle \boldsymbol{\mathcal{F}}(\mathfrak{y}_2^{(k-1)}; \mathfrak{y}_{2,r}^{(m)}), \tau_E \boldsymbol{\mathcal{F}}^*(\mathfrak{y}_2^{(k-1)}; \mathfrak{v}_{2,r}) \rangle_\Gamma \\
&\quad + \alpha \langle \boldsymbol{\mathcal{D}} \mathfrak{y}_{1,r}^{(n)} - \boldsymbol{\mathcal{D}} \mathfrak{y}_{2,r}^{(l)}, -\boldsymbol{\mathcal{D}} \mathfrak{v}_{2,r} \rangle_\Gamma \\
&\quad + \langle \boldsymbol{\mathcal{F}}(\mathfrak{y}_1^{(n)}; \mathfrak{y}_{1,r}^{(n)}) + \boldsymbol{\mathcal{F}}(\mathfrak{y}_2^{(k-1)}; \mathfrak{y}_{2,r}^{(m)}), -\boldsymbol{\mathcal{D}} \mathfrak{v}_{2,r} \rangle_\Gamma \\
&\quad + \langle \boldsymbol{\mathcal{D}} \mathfrak{y}_{1,r}^{(n)} - \boldsymbol{\mathcal{D}} \mathfrak{y}_{2,r}^{(l)}, \boldsymbol{\mathcal{F}}^*(\mathfrak{y}_2^{(k-1)}; \mathfrak{v}_{2,r}) \rangle_\Gamma = L_2(\mathfrak{v}_{2,r}).
\end{aligned}
\tag{9.6}
$$

Note that the appearance of indexes l and m in equation (9.6) has been swapped with respect to that of equation (9.5). Now, on top of the options $l, m = k-1, k$, there is also the possibility of using $n = k$ or $n = k-1$, i.e., to use the value just computed from equation (9.5) of the unknown in subdomain 1 or to use the value of the previous iteration. The first choice would be a Gauss-Seidel-type of coupling, whereas the second would be a Jacobi-type iterative scheme.

To be able to converge to any solution using this iterative strategy, we require an additional relaxation scheme for the transmitted quantities. Here, we have chosen an Δ^2 Aitken relaxation scheme as the one used in [100], which reads

$$\mathfrak{y}_i^{(k)} = \mathfrak{y}_i^{(k)} + \omega_i^{(k)}(\mathfrak{y}_i^{(k)} - \mathfrak{y}_i^{(k-1)}), \quad i = 1, 2, \tag{9.7}$$

where $\omega_i^{(k)}$ is the relaxation parameter.

10 HYPER REDUCTION

When constructing the ROM, if the nonlinear terms are evaluated using the FE mesh of the FOM, the cost of assembling the matrices of the resulting algebraic system for the ROM is of the same order as that of the FOM. Thus, some sort of approximation is thus required to evaluate these nonlinear contributions, which prevent from computing the system matrices in the off-line stage. The method resulting from these approximations is called hyper-ROM.

Most hyper-ROM strategies are based on some sort of sampling to evaluate nonlinear terms [13, 18, 26, 62, 85, 92]. Here we will follow a completely different approach, where the mesh-based hyper-ROM we propose consists in the solution of the described ROM problem using a coarser mesh than the FOM one.

The mesh-based hyper-reduction consists in the refinement or coarsening of areas over the domain Ω that are deemed of more or less importance, respectively. In general, we want to use this coarser mesh without increasing the order of the error introduced by the ROM. In this regard we could consider two options: a pre-computed refined mesh (i. e. based on an measure of error of the reduced basis as in [59] or on a beforehand knowledge of the problem as in [103]), or the method we propose here: to carry out a genuine Adaptive Mesh Refinement (AMR) in the ROM stage, and thus to compute the error of the (mesh-based) ROM solution and performing an AMR according to this error.

10.1 MESH-BASED HYPER-ROM

Let $\mathcal{T}_H$ a finite element partition of size $H > h$. Let us assume for both $\mathcal{T}_h$ and $\mathcal{T}_H$ a standard Lagrangian interpolation, with N_h nodes in $\mathcal{T}_h$ and N_H in $\mathcal{T}_H$ ($N_H < N_h$), $\{\varphi_h^a(\boldsymbol{x})\}$ being the basis for the FOM space ($a = 1, \dots, N_h$) and $\{\varphi_H^A(\boldsymbol{x})\}$ the basis for the FE space constructed from $\mathcal{T}_H$ ($A = 1, \dots, N_H$). Finally, let $\{\boldsymbol{\phi}_h^k(\boldsymbol{x})\}_{k=1}^r$ be the ROM basis obtained from the FOM solution. If $\boldsymbol{\phi}_h^{k,a}$, $a = 1, \dots, N_h$, are the nodal values of $\boldsymbol{\phi}_h^k(\boldsymbol{x})$ (those obtained from the Proper Orthogonal Decomposition (POD)), we may write the ROM solution at each time $t \in]0, t_f[$ as

$$\boldsymbol{y}_{r,h}(\boldsymbol{x}, t) = \sum_{k=1}^{r} \boldsymbol{\phi}_h^k(\boldsymbol{x}) Y^k(t) = \sum_{k=1}^{r} \left[\sum_{a=1}^{N_h} \varphi_h^a(\boldsymbol{x}) \boldsymbol{\phi}^{k,a} \right] Y^k(t), \qquad (10.1)$$

where subscript h in $\boldsymbol{y}_{r,h}(\boldsymbol{x}, t)$ has been introduced to emphasize that it is computed from the FOM mesh and $\{Y^k(t)\}_{k=1}^r$ are the ROM degrees of freedom.

If equation (10.1) is used to evaluate the nonlinear terms, the cost will be $\mathcal{O}(N_h)$. The idea of the mesh-based hyper-ROM we propose is to project the ROM basis obtained from the FOM mesh to the new coarser mesh. Although other possibilities could be used, in this work we use a simple interpolation as projection. Thus, the nodal values of the ROM basis at the nodes of $\mathcal{T}_H$ can be obtained as

$$\boldsymbol{\phi}_h^k(\boldsymbol{x}_A) = \sum_{a=1}^{N_h} \varphi_h^a(\boldsymbol{x}_A) \boldsymbol{\phi}^{k,a}, \quad A = 1, \dots, N_H, \qquad (10.2)$$

prior to any matrix evaluation, and then used to express the ROM solution as

$$\boldsymbol{y}_{r,H}(\boldsymbol{x}, t) = \sum_{k=1}^{r} \boldsymbol{\phi}_H^k(\boldsymbol{x}) Y^k(t) = \sum_{k=1}^{r} \left[\sum_{A=1}^{N_H} \varphi_H^A(\boldsymbol{x}) \boldsymbol{\phi}_h^k(\boldsymbol{x}_A) \right] Y^k(t). \qquad (10.3)$$

The cost of using equation (10.3) to evaluate the nonlinear terms will be $\mathcal{O}(N_H)$, rather than the cost $\mathcal{O}(N_h)$ of using equation (10.1).

10.2 AMR APPROACH TO THE HYPER-ROM

The success of the mesh-based hyper-ROM relies in the refinement or coarsening of areas that are deemed of more or less importance, respectively. In this regard we could consider two options: a pre-computed mesh based on some definition of the basis approximation error as in [59], or an AMR based on the real-time ROM error. In this work we propose a standard h-AMR approach using the algorithm described in [10]. Although the refinement and rebalance algorithms are easily included in the ROM, special attention should be given to the treatment of hanging nodes: since in this algorithm they are just an interpolation of the adjacent nodes and therefore do not provide any new information to the solution, we do not assemble them in the matrices or vectors.

In the AMR we propose, we know the FOM mesh $\mathcal{T}_h$ beforehand and we determine adaptively the coarse mesh $\mathcal{T}_H$ to be used in the hyper-ROM. These two meshes are in principle unrelated, but in the first iteration of the AMR procedure the latter can also be taken as a sub-mesh of the former, so that $\mathcal{T}_h$ is nested in $\mathcal{T}_H$.

The target mesh $\mathcal{T}_H$ of the AMR procedure should be such that the error of $y_{r,H}$ given by equation (10.3) should be of the same order as the error of $y_{r,h}$ given by equation (10.1) (after time discretization), that is to say,

$$\|y - y_{r,H}\| \approx \|y - y_{r,h}\|,$$

in a certain norm $\|\cdot\|$. However, as far as we are aware, *a priori* error estimates for the ROM are only available in terms of the eigenvalues discarded in the singular value decomposition associated to the POD, and not in terms of the mesh size h, even in the particular case of FE-based ROMs. Nevertheless, any *a posteriori* error estimator ξ, valid for a FE mesh, is in fact the only thing we need to drive the AMR. Indeed, if ξ_h is an error indicator for $\mathcal{T}_h$ and ξ_H the corresponding one

for $\mathcal{I}_H$ (for example, the maximum of the element-wise error values), the condition to be fulfilled is

$$\xi_H \leqslant C\xi_h,$$

C being a constant close to 1. In practice, ξ_h is never computed, since $\mathbf{y}_{r,h}$ is too expensive to obtain. The AMR is thus stopped when ξ_H is below a given target error, which could be linked for example to the error of the FOM solution from which the ROM basis has been constructed.

10.3 SUBSCALES AS AN ERROR MEASURE

Besides the refinement algorithm itself, the second key ingredient of AMR techniques is the definition of the error estimators that allow one to decide over which domain region mesh refinement is required. The use of residual-base error estimators in FE problems —including interior and boundary subscales— is vast in the literature (see for example [11, 60, 61] and references within). This type of error estimators can become useful in the ROM context, since they are explicit and therefore do not require solving additional equations.

In this work, we use the equations for subscales in the element interiors and in the boundaries to define an explicit error estimator that can be computed during the ROM solution. For that purpose we follow [11, 19], where a scaled L^2-norm of the subscales is defined as $|\breve{\mathbf{y}}|^2_{\boldsymbol{\tau}} := \breve{\mathbf{y}}^\top \boldsymbol{\tau}^{-1} \breve{\mathbf{y}}$, using the matrix of stabilization parameters $\boldsymbol{\tau}$ as scaling matrix. This way, we compute the error estimator at each element as

$$\xi_K^2 = \int_K \breve{\mathbf{y}}^\top \boldsymbol{\tau}_K^{-1} \breve{\mathbf{y}} + \sum_{E \subset \partial K} \int_E \breve{\mathbf{y}}_E^\top \boldsymbol{\tau}_E^{-1} \breve{\mathbf{y}}_E, \tag{10.4}$$

with $\breve{\mathbf{y}}$ obtained by solving equation (6.6) at each time level t^j and $\breve{\mathbf{y}}_E$ given by equation (6.9) (with the adequate modification for Neumann-type edges).

Since the subscales at the element interiors are part of the stabilization of equation (6.7), they are always computed —using equation (6.6)— at the integration points. However, when using these subscales as error estimation, they can be computed at any location in the interior of the element (e.g. at the center of the element or at the nodes).

Rather, the subscales on the element boundaries are defined by the inter-element jump operator, and therefore, the contribution of neighboring elements must be included. To allow computing the second term in equation (10.4) locally for each element, we follow the approach in [11, 19], where the approximation described in [23, 49] and the error estimator in [106] are used to express the term mentioned in terms of norms evaluated only in the element interior.

Part IV

PHYSICAL PHENOMENA

11 CONVECTION-DIFFUSION-REACTION

The aim of this example is to illustrate the behavior of the basis modes (figure 6.1). It consist in a dynamic two dimensional convection–diffusion–reaction problem stated as

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

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 11.1.

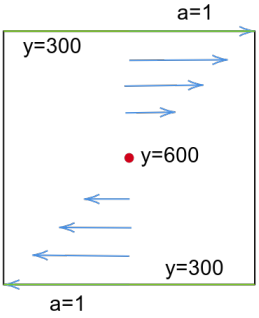
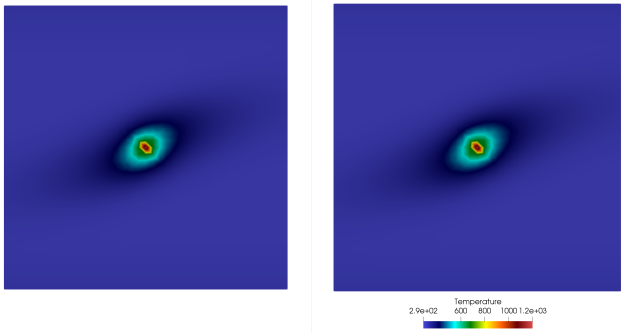


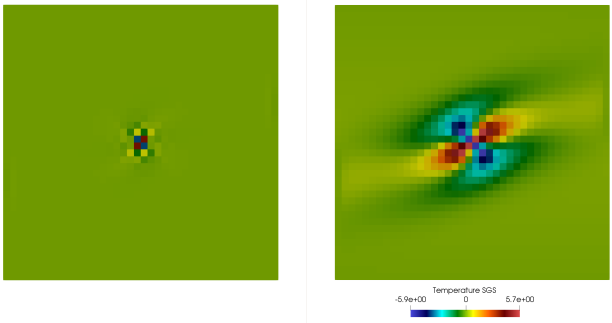
Figure 11.1: Convection–diffusion–reaction problem.

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 Full Order Model (FOM) and the Reduced Order Model (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 solved using $r = 6$ basis vectors and stabilized using dynamic Orthogonal SubGrid Scale (OSGS) and the stabilization parameter defined in equation (6.15), with constants $c_1 = 4$ and $c_2 = 2$. As mentioned in chapter 8, the Galerkin and the Petrov-Galerkin projections yield very similar results for this linear problem. Figure 11.2 shows 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.



(a) FOM and ROM resolved scales.



(b) FOM and ROM subscales.

Figure 11.2: Convection–diffusion–reaction problem: contour plots.

12 | INCOMPRESSIBLE NAVIER-STOKES

In this chapter we present a set of examples where we test the basic characteristics of the stabilized ROM formulation and the additional techniques described in parts ii and iii, using the incompressible Navier–Stokes equations. The problem consists of finding a velocity $\mathbf{u} : \Omega \times]0, t_f[\to \mathbb{R}^d$ and a pressure $p : \Omega \times]0, t_f[\to \mathbb{R}$ such that

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

where ν denotes the kinematic viscosity and $\mathbf{f}$ a vector of body forces. Initial and boundary conditions are set for Γ as

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

Let $H^1(\Omega)$ be the space of functions whose distributional derivatives belong to $L^2(\Omega)$, and let $H^1_0(\Omega)$ the space of vector functions in $H^1(\Omega)$, that vanish on Γ_D. Let also $\mathcal{V}_0 = (H^1_0(\Omega))^d$ and $\mathcal{Q}_0 = L^2(\Omega)/\mathbb{R}$, and define $\mathcal{V} = L^2(0, t_f; \mathcal{V}_0)$ and $\mathcal{Q} = L^1(0, t_f; \mathcal{Q}_0)$. This way, the variational form of the problem is defined in the spaces $\mathcal{Y} = \mathcal{V} \times \mathcal{Q}$ for the trial solutions $-\mathbf{y} = [\mathbf{u}, p]-$ and $\mathcal{Y}_0 = \mathcal{V}_0 \times \mathcal{Q}_0$ for the test functions $-\mathbf{v} = [\mathbf{v}, q]$.

We write the terms that correspond to the abstract ones introduced in chapter 6, using dynamic OSGS and assuming the force term to belong to the Finite Element (FE) space (see tables 6.1 and 6.2), as

$$\boldsymbol{\mathcal{M}}(\mathbf{y}) = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0}^\top & 0 \end{bmatrix}, \ \boldsymbol{\mathcal{L}}(\mathbf{y}; \mathbf{y}_r) = \begin{bmatrix} \mathbf{u} \cdot \nabla \mathbf{u}_r - \nu \Delta \mathbf{u}_r + \nabla p_r \\ \nabla \cdot \mathbf{u}_r \end{bmatrix},$$

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

$$\mathcal{F}(\mathbf{y};\mathbf{y}_r) = \begin{bmatrix} \mathbf{n}\cdot p_r\mathbf{I} - \mathbf{n}\cdot\nu\nabla\mathbf{u}_r \\ 0 \end{bmatrix},\ \mathcal{F}^*(\mathbf{y};\mathbf{y}_r) = \begin{bmatrix} \mathbf{n}\cdot q_r\mathbf{I} + \mathbf{n}\cdot\nu\nabla\boldsymbol{v}_r \\ 0 \end{bmatrix},$$

yielding the forms

$$B(\mathbf{y};\mathbf{y}_r,\mathbf{v}_r) = \langle\mathbf{u}\cdot\nabla\mathbf{u}_r,\boldsymbol{v}_r\rangle + (\nu\nabla\mathbf{u}_r - p_r\mathbf{I},\nabla\boldsymbol{v}_r) + (\nabla\cdot\mathbf{u}_r,q_r), \quad (12.1a)$$

$$\begin{aligned} B_s(\mathbf{y};\mathbf{y}_r,\mathbf{v}_r) = &-\sum_K\langle\breve{\boldsymbol{\Pi}}(\partial_t\mathbf{u}),\tau_{1,t}(\mathbf{u}\cdot\nabla\boldsymbol{v}_r + \nu\Delta\boldsymbol{v}_r + \nabla q_r)\rangle_K \\ &-\sum_K\langle\breve{\boldsymbol{\Pi}}(\mathbf{u}\cdot\nabla\mathbf{u}_r - \nu\Delta\mathbf{u}_r),\tau_{1,t}(\mathbf{u}\cdot\nabla\boldsymbol{v}_r + \nu\Delta\boldsymbol{v}_r + \nabla q_r)\rangle_K \\ &-\sum_K\langle\breve{\boldsymbol{\Pi}}(\nabla p_r),\tau_{1,t}(\mathbf{u}\cdot\nabla\boldsymbol{v}_r + \nu\Delta\boldsymbol{v}_r + \nabla q_r)\rangle_K \\ &-\sum_K\langle\breve{\boldsymbol{\Pi}}(\nabla\cdot\mathbf{u}_r),\tau_2\nabla\cdot\boldsymbol{v}_r\rangle_K, \end{aligned} \quad (12.1b)$$

$$\begin{aligned} B_b(\mathbf{y};\mathbf{y}_r,\mathbf{v}_r) = &-\sum_E\langle[\![\mathbf{n}\cdot p_r]\!][\![\boldsymbol{\tau}_E]\!],[\![\mathbf{n}\cdot q_r + \mathbf{n}\cdot\nu\nabla\boldsymbol{v}_r]\!]\rangle_E \\ &+\sum_E\langle[\![\mathbf{n}\cdot\nu\nabla\mathbf{u}_r]\!][\![\boldsymbol{\tau}_E]\!],[\![\mathbf{n}\cdot q_r + \mathbf{n}\cdot\nu\nabla\boldsymbol{v}_r]\!]\rangle_E, \end{aligned} \quad (12.1c)$$

$$L_s(\mathbf{y};\mathbf{v}_r) = -\sum_K\langle\breve{\boldsymbol{\Pi}}(f) + \delta t^{-1}\breve{\mathbf{u}},\tau_{1,t}(\mathbf{u}\cdot\nabla\boldsymbol{v}_r + \nu\Delta\boldsymbol{v}_r + \nabla q_r)\rangle_K, \quad (12.1d)$$

and likewise for the FOM formulation in chapter 5. Note that as described before, we use the subscript h for the FOM and the subscript r for the ROM spaces and functions. We use a Backward Differentiation Formula (BDF) time integration scheme of second order for the resolved scales and of first order for the subscales, with $\breve{\mathbf{u}}$ evaluated at t^j when solving for the unknowns evaluated at t^{j+1}.

The stabilization parameter matrix is defined following [36, 37] as

$$\boldsymbol{\tau}_K(\mathbf{y}) = \operatorname{diag}(\tau_1\mathbf{I},\tau_2) = \begin{bmatrix} (c_1\frac{\nu}{h^2} + c_2\frac{|\mathbf{u}|_K}{h})^{-1}\mathbf{I} & 0^\top \\ 0 & \nu + \frac{c_2}{c_1}|\mathbf{u}|h \end{bmatrix},$$

in $K\in\mathcal{T}_h$, with $c_1 = 4$ and $c_2 = 2$ for linear elements, $\tau_{1,t}^{-1} = \delta t^{-1} + \tau_1^{-1}$, $\tau_E = \left(c_E\frac{\nu}{h}\right)^{-1}$ and $c_E = 0.01$.

12.1 FLOW OVER A CYLINDER

The first numerical example for the incompressible Navier–Stokes problem is the two dimensional flow over a cylinder that generates a von Kármán vortex street. 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 (figure 12.1).

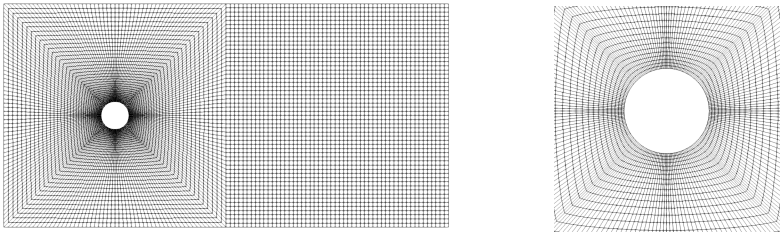
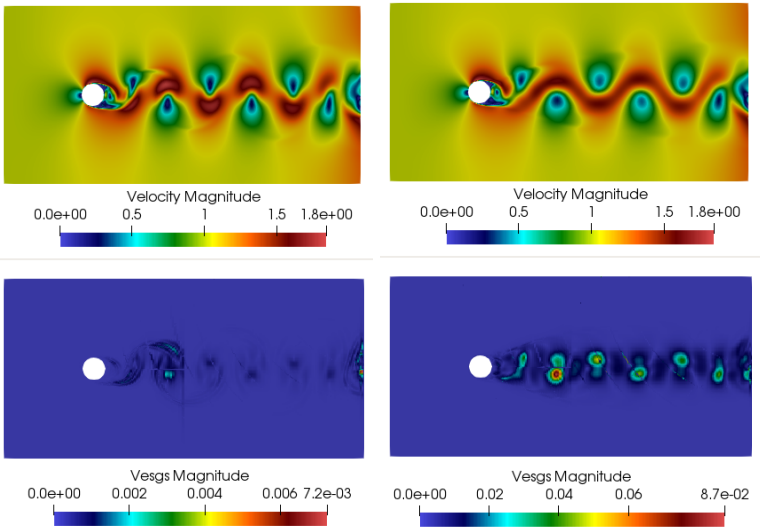


Figure 12.1: Flow over a cylinder: mesh with 92960 grid points and 23040 quadrilateral elements.

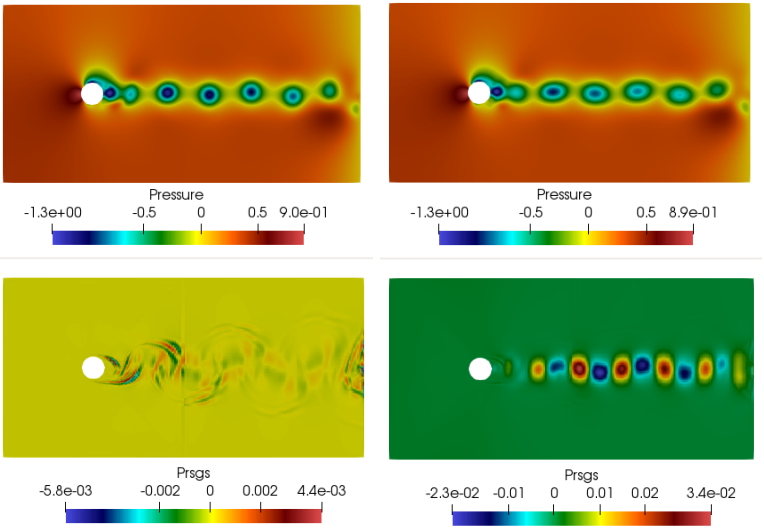
To obtain the fully developed vortex shedding characteristic of this problem, a preliminary simulation is performed until $t = 100$, reset as $t = 0$. From that time, solutions from the next 500 time steps of velocity and pressure are used to calculate the ROM basis. A time step of $\delta t = 0.05$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

Figures 12.2 and 12.3 show the 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.



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

Figure 12.2: Flow over a cylinder: velocity (top) and velocity subscales (bottom).



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

Figure 12.3: Flow over a cylinder: pressure (top) and pressure subscales (bottom).

To evaluate the accuracy of the ROM, we perform some numerical tests using 7 different sets of bases, varying the retained energy from $\eta_1 = 0.975$ to $\eta_7 = 0.6$ (see equation (2.6)), where we aim to compare some of the different formulations described in chapter 6.

The numerical tests consist in comparing:

- Approximation of the subscales:
  - OSGS vs. Algebraic SubGrid Scale (ASGS).
  - Dynamic vs. quasi-static.
- 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 [28].

Figure 12.4 depicts the number of basis vectors r as a function of the retained energy η for the 7 sets of bases.

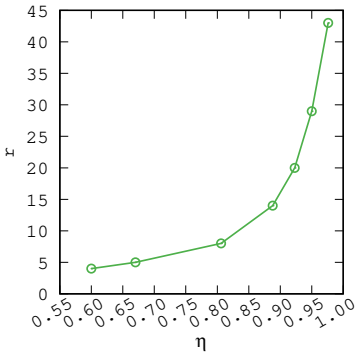


Figure 12.4: Flow over a cylinder: number of basis vectors r as a function of retained energy η.

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

$$F^{\circ}(t) = \left| \int_{\Gamma_{\circ}} \mathcal{F}_n(\mathbf{u}(\mathbf{x}, t))\, d\Gamma \right|, \tag{12.2}$$

where $\Gamma_{\circ}$ is the cylinder boundary.

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. Defining $F^{\circ}_{j,\mathrm{FOM}}$ as the total force obtained with the FOM at time t^j, $j = 1, \ldots, S$, and $F^{\circ}_{j,\mathrm{ROM}}$ as the one obtained with the ROM, we set

$$F^{\circ}_{\mathrm{RMSD}} = \sqrt{\frac{1}{S}\sum_{j=1}^{S}\left(F^{\circ}_{j,\mathrm{ROM}} - F^{\circ}_{j,\mathrm{FOM}}\right)^2}. \tag{12.3}$$

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.

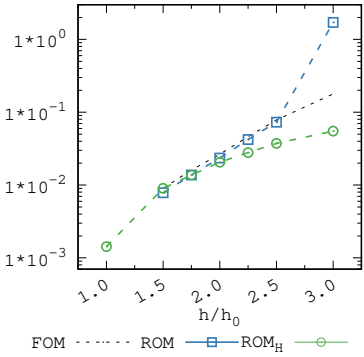


Figure 12.5: Flow over a cylinder: $F^{\circ}_{\mathrm{RMSD}}$ convergence for FOM, ROM and ROM_H with $\eta = 0.9$.

Figure 12.5 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.

ORTHOGONALITY The first test performed consists in comparing the two definitions of the projection $\tilde{\Pi}_\tau$. For this purpose we solve the ROM using the two types of subscales defined in section 6.3: OSGS and ASGS.

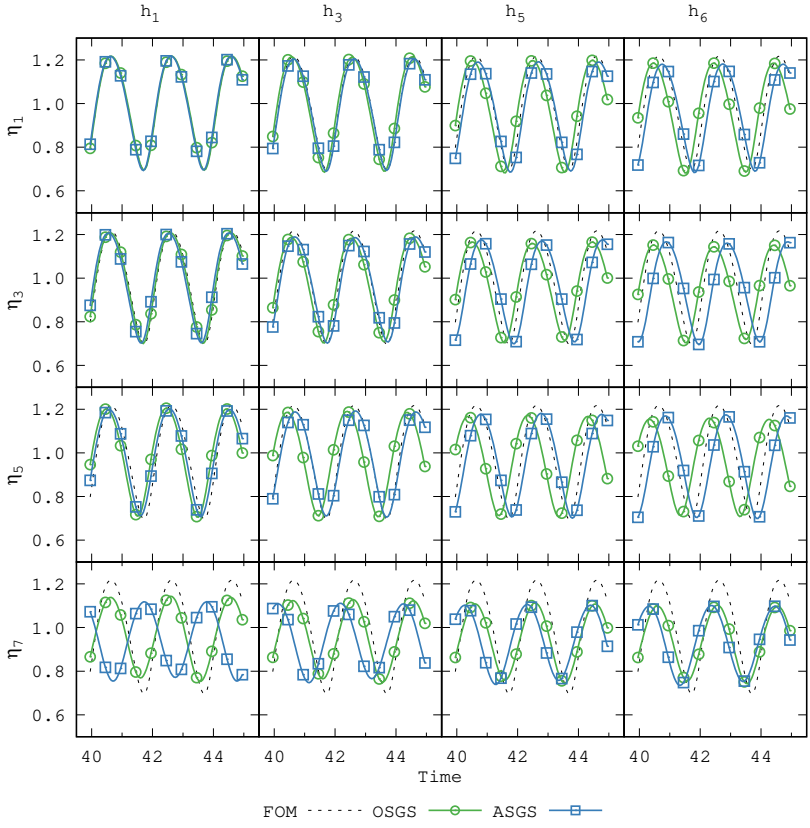
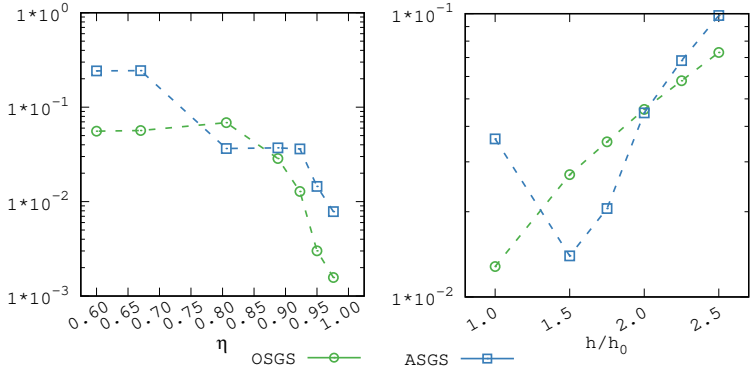


Figure 12.6: Flow over a cylinder: comparison of F° for OSGS and ASGS.

Figure 12.6 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 12.7 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 OSGS, with the error decreasing with the addition of basis vectors and a finer mesh. As in figure 12.6, the convergence for the ASGS is irregular.



(a) Retained energy. ROM. $\frac{h}{h_0} = 1.0$.

(b) Element size. Hyper-ROM. $\eta = 0.925$

Figure 12.7: Flow over a cylinder: F°_{RMSD} convergence for OSGS and ASGS.

Figure 12.8 shows a comparison of the spectra of the force for three combinations of η and h, with the same behavior as the previous results, a better approximation is achieved when using OSGS and a bigger amount of basis vectors r.

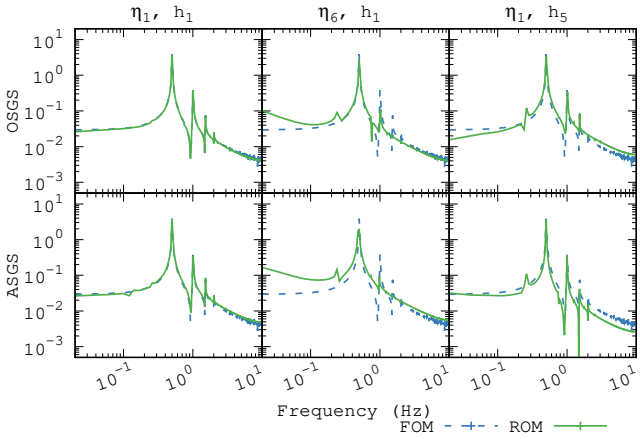
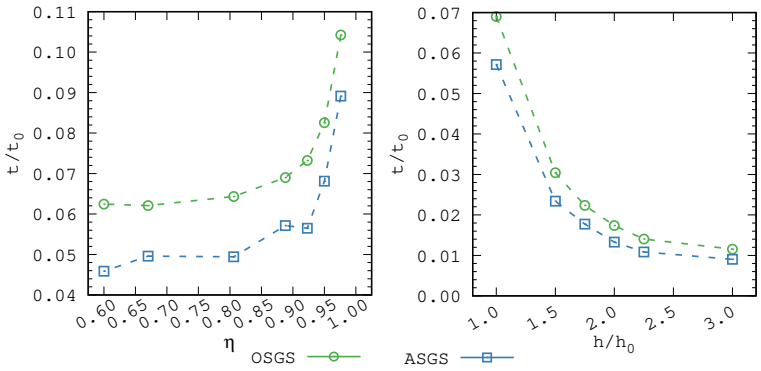


Figure 12.8: Flow over a cylinder: Fourier transform of F° for OSGS and ASGS.

Figure 12.9 shows the computational time ratio for ROM and hyper-ROM solutions. Computational savings are up to 99% when using smaller basis and coarser meshes. Despite the use of OSGS being more costly, the computational time difference between the two types of subscales is less than 2%.

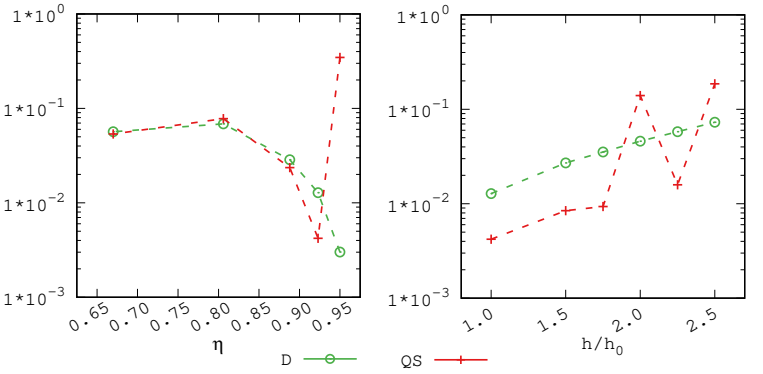


(a) Retained energy. ROM. $\frac{h}{h_0} = 1.0$

(b) Element size. Hyper-ROM. $\eta = 0.925$

Figure 12.9: Flow over a cylinder: computational time ratio for OSGS and ASGS.

DYNAMIC SUBSCALES In the second test we compare dynamic and quasi-static subscales. Figure 12.10 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 presented in [43, 46] for a Variational Multi-Scale (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. $\eta = 0.925$

Figure 12.10: Flow over a cylinder: F°_{RMSD} convergence for dynamic and quasi-static OSGS.

INTERPOLATION ORDER In the third test we analyze 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 (figure 12.1) 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).

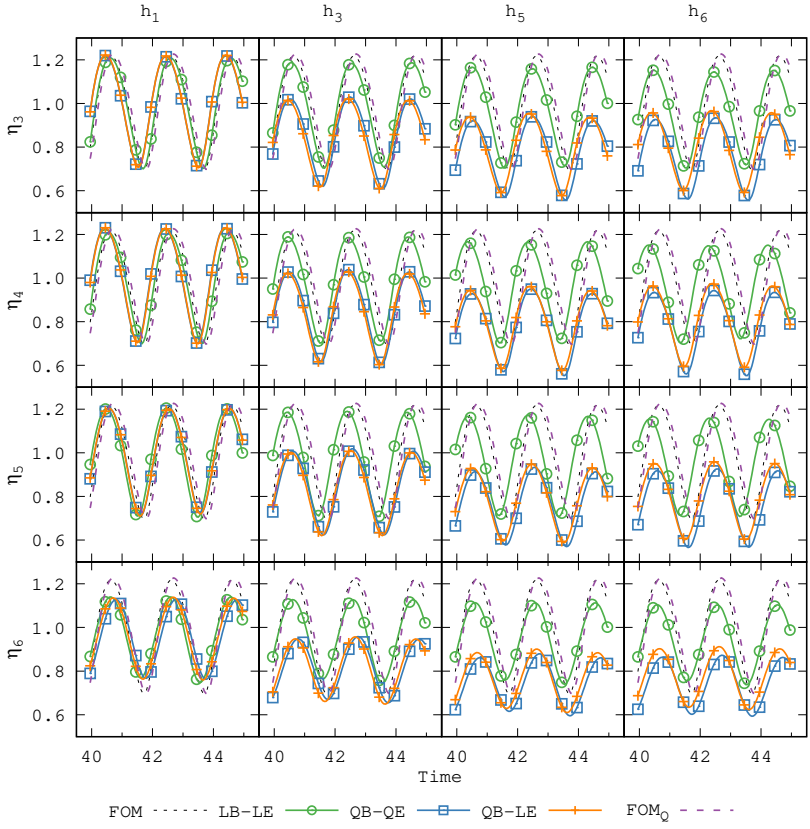


Figure 12.11: Flow over a cylinder: comparison of $\bar{F}^{\circ}$ for LB-LE, QB-QE and QB-LE.

Figure 12.11 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 12.12 depicts a comparison of the number of vectors in the basis as a function of the retained energy for bases calculated with bilinear and biquadratic elements.

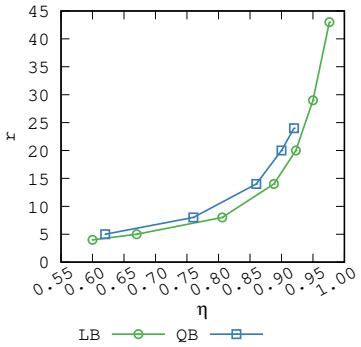
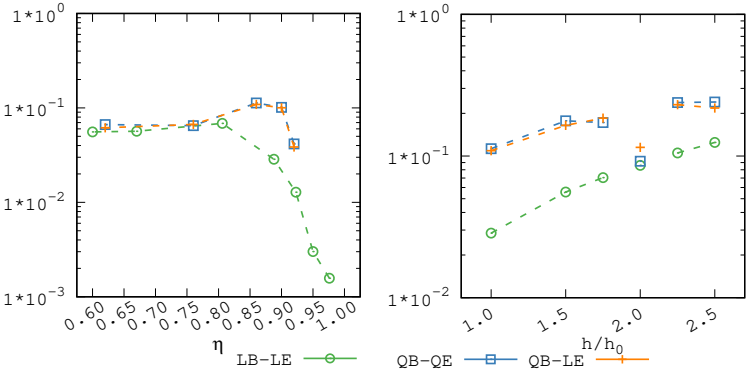


Figure 12.12: Flow over a cylinder: number of basis vectors r as a function of the retained energy η for basis computed with biquadratic and bilinear elements.



(a) Retained energy. ROM. $\frac{h}{h_0} = 1.0$

(b) Element size. Hyper-ROM. $\eta = 0.925$

Figure 12.13: Flow over a cylinder: F°_{RMSD} convergence for LB-LE, QB-QE and QB-LE.

Figure 12.13 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 $\eta = 0.925$. 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 12.14 shows a comparison of the spectra of the force for three combinations of η and h.

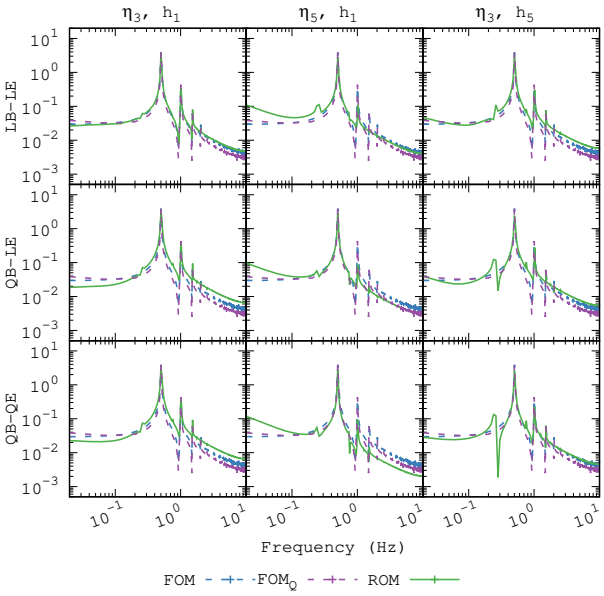
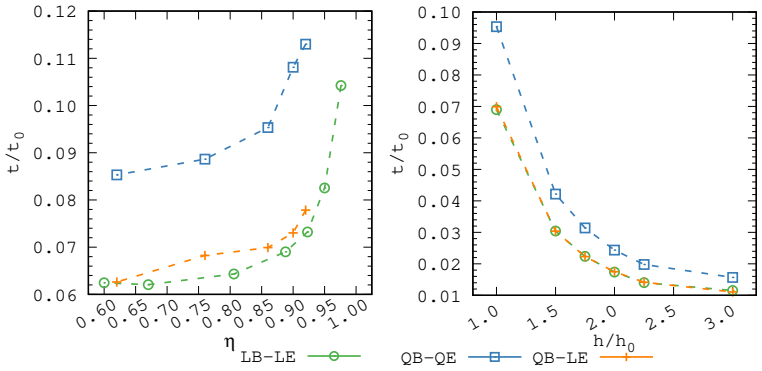


Figure 12.14: Flow over a cylinder: Fourier transform of F° for LB-LE, QB-QE and QB-LE.

Figure 12.15 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.
$\eta = 0.925$

Figure 12.15: Flow over a cylinder: computational time ratio t/t_0 of the ROM solution (t) with respect to the FOM solution (t_0) for LB-LE, QB-QE and QB-LE.

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 [8, 67, 68, 90] with the particularity that the stabilization term of the pressure is added in order to be able to use equal interpolations for $\mathbf{u}_r$ and p_r.

Following the formulation presented in [28] for FEs, we split the velocity subscales as $\breve{\mathbf{u}} = \breve{\mathbf{u}}_1 + \breve{\mathbf{u}}_2$. Furthermore, we assume that the convective and pressure terms in the solution of the subscales are orthogonal and neglect second order space derivatives, yielding

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

Using this definition of the velocity subscales we end with an OSGS formulation for the ROM with the stabilization terms are defined as

$$
\begin{aligned}
B_s(\mathbf{y};\mathbf{y}_r,\mathbf{v}_r) = &-\sum_K \langle \Pi_r^{\perp}(\mathbf{u}\cdot\nabla\mathbf{u}_r), \tau_{1,t}\mathbf{u}\cdot\nabla\mathbf{v}_r\rangle_K \\
&-\sum_K \langle \Pi_r^{\perp}(\nabla\cdot\mathbf{u}_r), \tau_2\nabla\cdot\mathbf{v}_r\rangle_K - \sum_K \langle \Pi_r^{\perp}(\nabla p_r), \tau_{1,t}\nabla q_r\rangle, \\
L_s(\mathbf{y};\mathbf{v}_r) = &-\sum_K \langle \delta t^{-1}\breve{\mathbf{u}}_1, \tau_{1,t}\mathbf{u}\cdot\nabla\mathbf{v}_r\rangle_K - \sum_K \langle \delta t^{-1}\breve{\mathbf{u}}_2, \tau_{1,t}\nabla q_r\rangle,
\end{aligned}
$$

instead of (12.1b)-(12.1c), with the stabilization parameters defined in the same way as for the original OSGS formulation.

In figure 12.16 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 [38] 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.

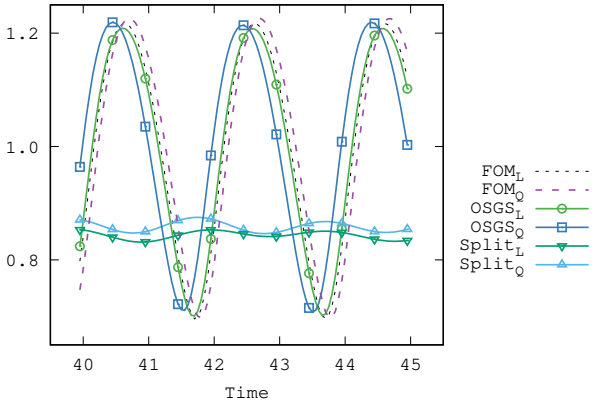


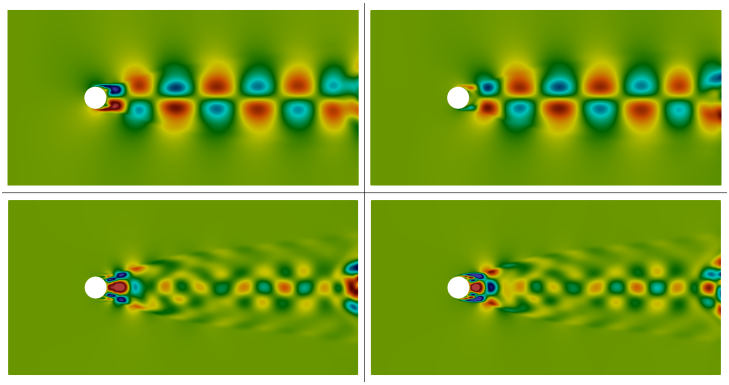
Figure 12.16: Flow over a cylinder: comparison of F° for OSGS and term-by-term (Split) stabilization methods using bilinear (L) and biquadratic (Q) elements.

12.1.1 *AMR hyper-ROM*

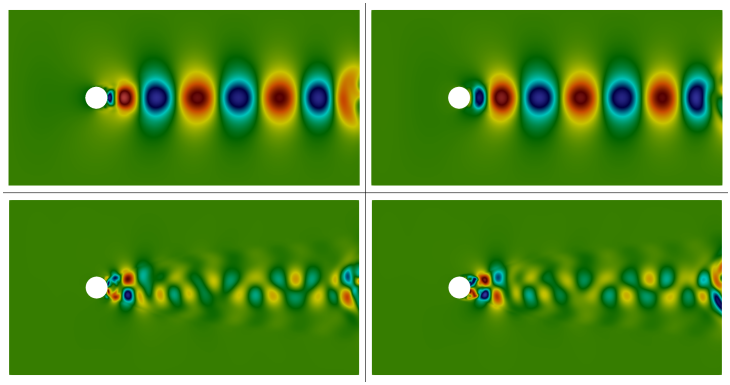
Now, we use the same two dimensional flow over a cylinder to test the proposed mesh-base hyper-ROM, for which we use a FOM and a ROM with the same mesh, and a hyper-ROM and an Adaptive Mesh Refinement (AMR) hyper-ROM with a substantially lower amount of elements.

For the FOM and the ROM the domain is discretized using a symmetric mesh of 92320 bilinear elements and a time step $\delta t = 0.05$. We gather 1000 snapshots after an initial transient of 100 steps of simulation to compute the ROM basis. The hyper-ROM domain is discretized using a similarly distributed mesh consisting of 18360 bilinear elements, whilst for the AMR hyper-ROM we start with a discretized domain of 1840 bilinear elements reaching on average 19392 elements using a three level refinement with the refinement tolerance set as $10^{-6} < \xi_K^2 < 10^{-5}$.

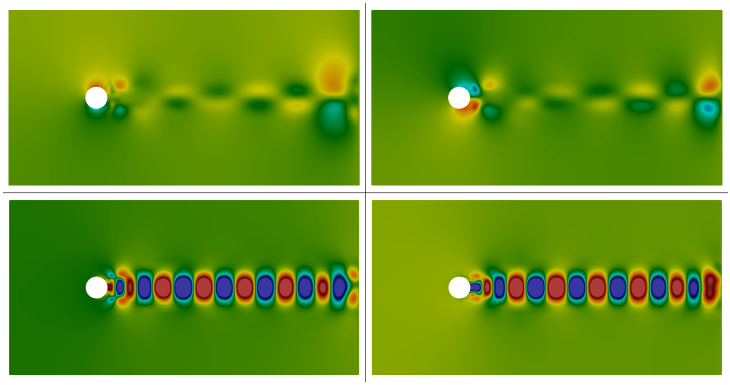
In this example we solve the ROM and both hyper-ROMs using three sets of basis of $r = [4, 5, 6]$ modes with a retained energy of $\eta = [0.6, 0.67, 0.75]$. Figure 12.17 shows the four first modes of the reduced basis for velocity and pressure and figure 12.18 shows the fine, coarse and refined meshes.



(a) x-velocity.



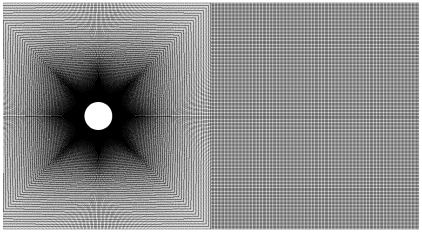
(b) y-velocity.



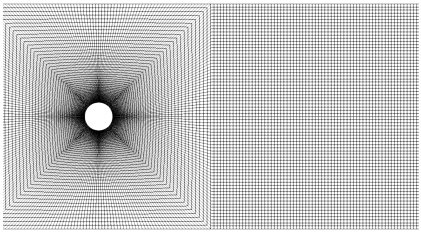
(c) Pressure.

Figure 12.17: Flow over a cylinder: first four modes of the reduced basis for velocity and pressure.

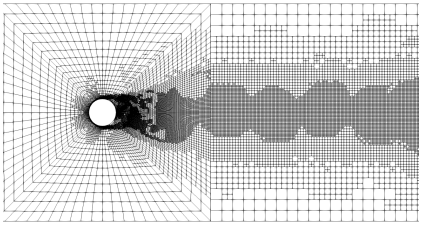
Figure 12.19 shows the calculated ξ_K^2 for the four cases at $t = 50$. In the same way as in the previous example, when comparing figures 12.17 and 12.19d we observe that the refinement occurs in the same areas where the basis contains more information. Figure 12.20 shows a comparison of contour plots of velocity magnitude and pressure for FOM and AMR hyper-ROM, where the quality of the approximation is evident, being the results obtained with both approaches.



(a) FOM mesh.

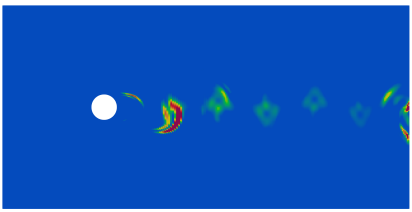


(b) Hyper-ROM mesh.

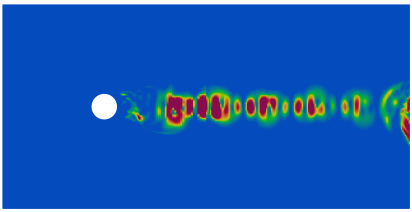


(c) AMR hyper-ROM mesh at $t = 50$.

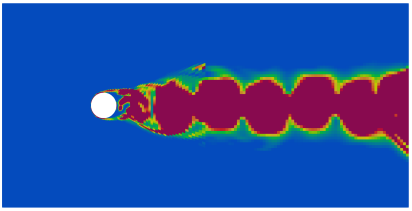
Figure 12.18: Flow over a cylinder: mesh for FOM, hyper-ROM and AMR hyper-ROM.



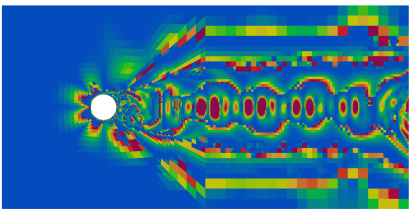
(a) FOM velocity magnitude.



(b) AMR hyper-ROM velocity magnitude.

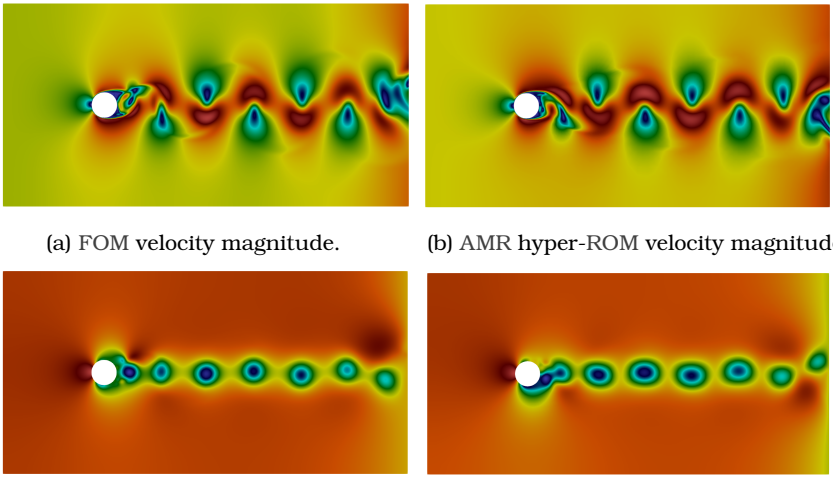


(c) FOM pressure.



(d) AMR hyper-ROM pressure.

Figure 12.19: Flow over a cylinder: contour plot of $\xi_K^2 \in [10^{-6}, 10^{-5}]$ for FOM, ROM, hyper-ROM and AMR hyper-ROM at $t = 50$.



(a) FOM velocity magnitude.

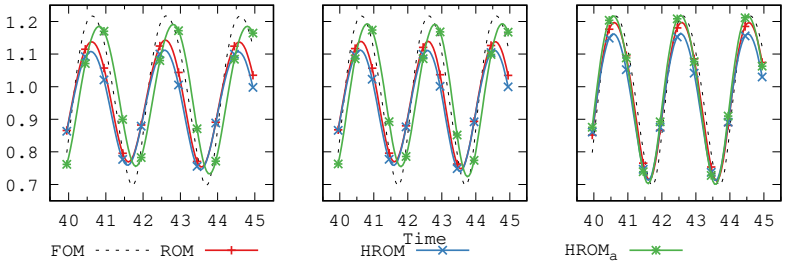
(b) AMR hyper-ROM velocity magnitude.

(c) FOM pressure.

(d) AMR hyper-ROM pressure.

Figure 12.20: Flow over a cylinder: contour plots for FOM and AMR hyper-ROM at $t = 50$ with $\eta = 0.75$, $r = 6$.

To compare the results we use again the norm of the total force exerted over the cylinder boundary $F^\circ(t)$ and its RMSD F°_{RMSD} defined in equations (12.2) and (12.3). Figure 12.21 shows the comparison of the total force in the time interval $[40, 45]$ for all the cases, using three different basis sizes.



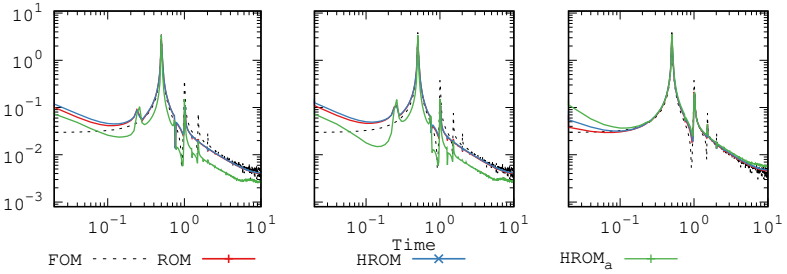
(a) $\eta = 0.6$, $r = 4$.

(b) $\eta = 0.67$, $r = 5$.

(c) $\eta = 0.75$, $r = 6$.

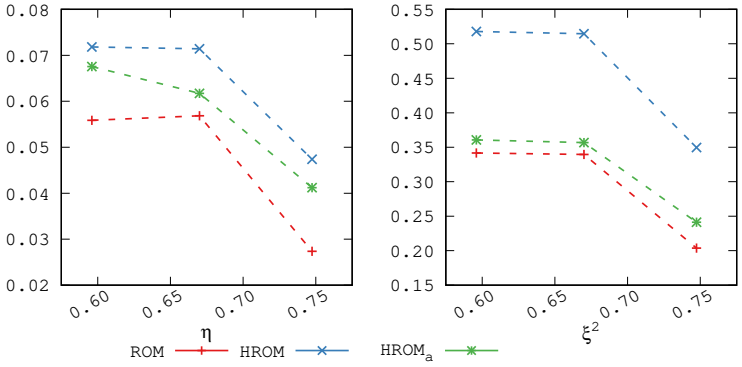
Figure 12.21: Flow over a cylinder: F° for FOM, ROM, hyper-ROM and AMR hyper-ROM.

Figure 12.22 shows a comparison of the spectra of the force for the same cases. Where unexpectedly both comparisons indicate that using the AMR approach the solution not only approaches better than the uniform hyper-ROM but also gives better results in amplitude and frequency than the ROM.



(a) $\eta = 0.6$, $r = 4$. (b) $\eta = 0.67$, $r = 5$. (c) $\eta = 0.75$, $r = 6$.

Figure 12.22: Flow over a cylinder: Fourier transform of F° for FOM, ROM, hyper-ROM and AMR hyper-ROM.



(a) F°_{RMSD}. (b) Time average of ξ^2.

Figure 12.23: Flow over a cylinder: F°_{RMSD} and ξ^2 for ROM, hyper-ROM and AMR hyper-ROM.

Figure 12.23 shows the RMSD of the total force F°_{RMSD} and the time average of the SubGrid Scale (SGS) norm $\xi^2 = \sum_K \xi^2_K$ for the

all the cases considered. Although the qualitative comparison of the total force F° in figure 12.20 and the spectrum in figure 12.21 displays a better performance of the AMR hyper-ROM over the ROM, the quantitative comparison using the RMSD reveals that, albeit being better than its uniform mesh counterpart, the hyper-ROM approaches the ROM solution quality.

12.2 FLOW OVER A BACKWARD-FACING STEP

The second numerical example for the incompressible Navier–Stokes problem consist in a two dimensional flow over a backward-facing step similar to the one presented in [69] and solved using ROMs in [13, 14, 30].

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. No turbulence model is used, letting the stabilization act as an Implicit Large Eddy Simulation model [41, 57], if required.

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.

In this example we use a basis size of $r = 21$ with a retained energy of $\eta = 0.95$. Figures 12.24 and 12.25 show velocity and pressure contours for FOM, ROM and hyper-ROM at $t = 50$. Note that despite

the complexity of the problem, the contour plots for the three cases present similar behavior, capturing successfully the periodicity and formation of the vortexes.

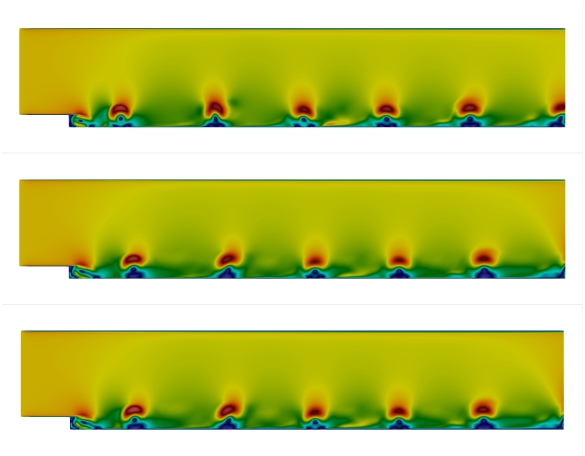


Figure 12.24: Flow over a backward-facing step: from top to bottom: velocity contours for FOM, ROM and hyper-ROM with $\eta = 0.95$ and $r = 21$.

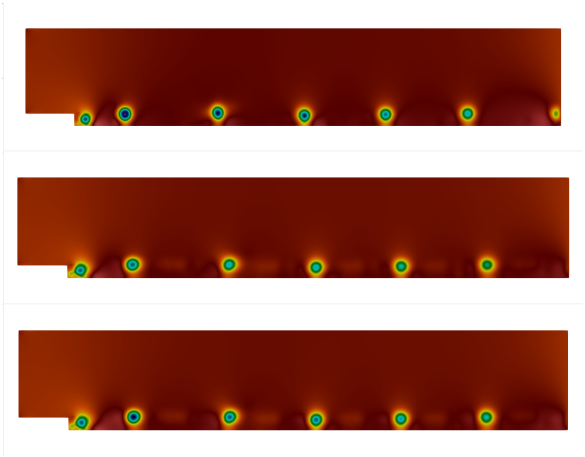
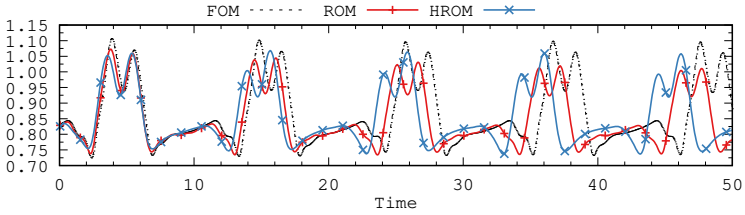
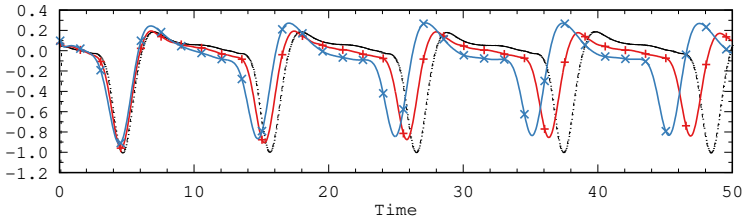


Figure 12.25: Flow over a backward-facing step: from top to bottom: pressure contours for FOM, ROM and hyper-ROM with $\eta = 0.95$ and $r = 21$.

Figure 12.26 shows a comparison of the velocity magnitude and the pressure at the control point $(22, 1)$ and figure 12.27 shows a comparison of the spectra of the velocity and the pressure at the same control point.



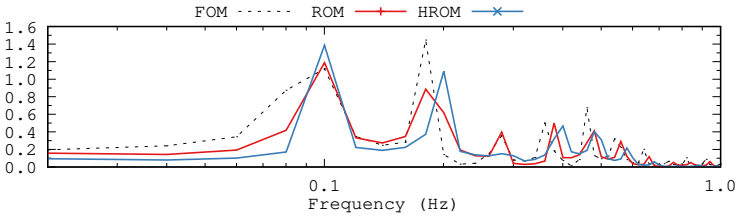
(a) Velocity magnitude



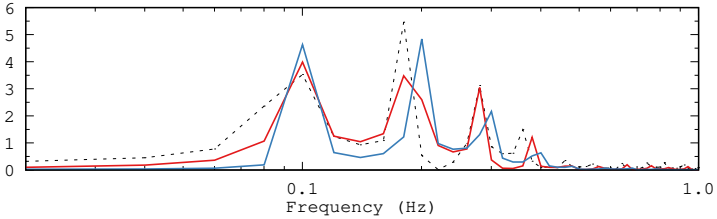
(b) Pressure

Figure 12.26: Flow over a backward-facing step: comparison of FOM, ROM and hyper-ROM with $\eta = 0.95$ and $r = 21$ at the control point $(22, 1)$.

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, and considering that this results occur near $\eta = 1$, we believe it can attributed to an *over-fitting* phenomenon. This idea is explored in [13, 56], where it is attributed to a lack of smoothness or noisiness in low energy basis vectors. 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.



(a) Velocity magnitude



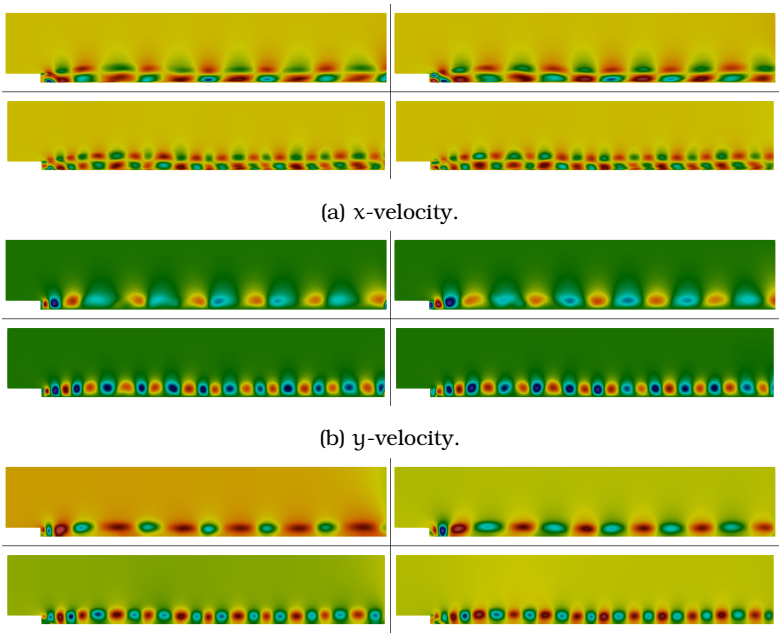
(b) Pressure

Figure 12.27: Flow over a backward-facing step: Fourier transform of FOM, ROM and hyper-ROM with $\eta = 0.95$ and $r = 21$ at the control point $(22, 1)$.

12.2.1 *AMR hyper-ROM*

We test the hyper-ROM formulation for the flow over a backward-facing step in a similar way as done for the previous example. We use for the FOM a discretized domain of 61520 quadrilateral bilinear elements and 62214 nodes and a time step of $\delta t = 0.05$ for the time discretization; we construct a ROM basis over 1000 time steps after an initial 100 steps of simulation. For the AMR hyper-ROM case, we start the solution from a mesh of 1497 quadrilateral bilinear elements and 1392 nodes, from which we follow a three level refinement with a tolerance set to $10^{-5} < \xi_K^2 < 10^{-4}$, with ξ_K given by equation (10.4). As in the previous example, to evaluate the AMR hyper-reduction, we also solve a hyper-ROM with a uniform mesh of 5568 quadrilateral bilinear elements.

In this example, we solve the ROM and hyper-ROMs using a basis of $r = 21$ modes with a retained energy of $\eta = 0.95$. Figure 12.28 shows the first four modes of the reduced basis for velocity and pressure.



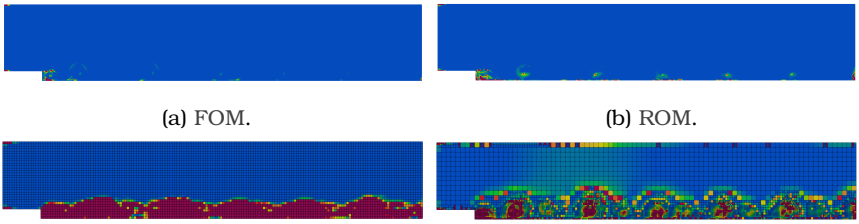
(a) x-velocity.

(b) y-velocity.

(c) Pressure.

Figure 12.28: Flow over a backward-facing step: first four modes of the reduced basis.

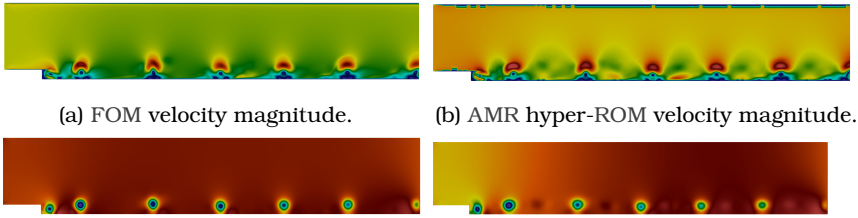
Figure 12.29 shows a comparison of the calculated ξ_K^2 at the time $t = 50$ for the four cases. Figures 12.29c and 12.29d show the meshes for the uniform and the AMR hyper-ROM, with 5568 and 4977 elements,respectively. Comparing figures 12.28 and 12.29d, we can identify the area with the finer mesh with the part where the basis contains more information, noting that in the AMR case the calculated ξ_K^2 is lower than the one for the uniform mesh hyper-ROM. In figure 12.30 we display contour plots of the velocity magnitude and the pressure for the AMR hyper-ROM for the last time step of the simulation, $t = 50$.



(a) FOM. (b) ROM.

(c) Hyper-ROM. (d) AMR hyper-ROM.

Figure 12.29: Flow over a backward-facing step: contour plot of $\xi_K^2 \in [10^{-5}, 10^{-4}]$ for FOM, ROM, hyper-ROM and AMR hyper-ROM cases at $t = 50$.



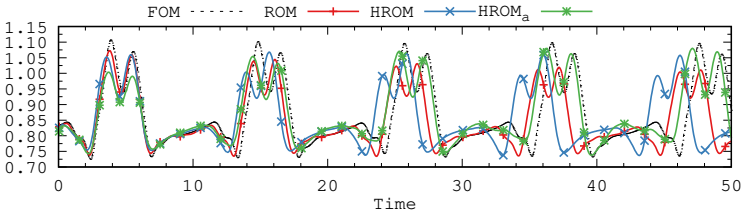
(a) FOM velocity magnitude. (b) AMR hyper-ROM velocity magnitude.

(c) FOM pressure. (d) AMR hyper-ROM pressure.

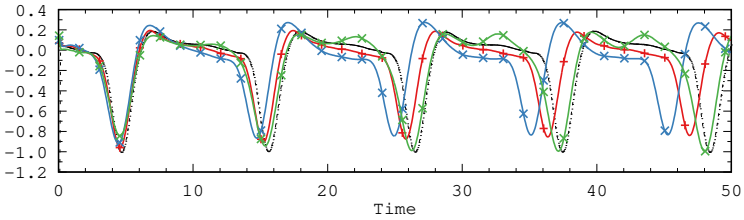
Figure 12.30: Flow over a backward-facing step: contour plots for FOM and AMR hyper-ROM at $t = 50$ with $\eta = 0.95$ and $r = 21$.

As done previously, to compare in a more quantitative way the solutions, we present in figure 12.31 the velocity and the pressure and in figure 12.32 a comparison of their spectra at the control point $(22, 1)$, for the FOM, ROM, hyper-ROM and AMR hyper-ROM (labeled HROM_a) with an average of 4723 elements.

In both figures the use of the AMR hyper-ROM improves the behavior of the solution compared to the uniform mesh hyper-ROM while maintaining a similar computational cost. Moreover, this method indicates improvement over the ROM, correcting the amplitude and frequency for velocity and pressure even further. As discussed in section 12.1, these better results for control points not necessarily mean an overall improvement for all points or for an integral measure.

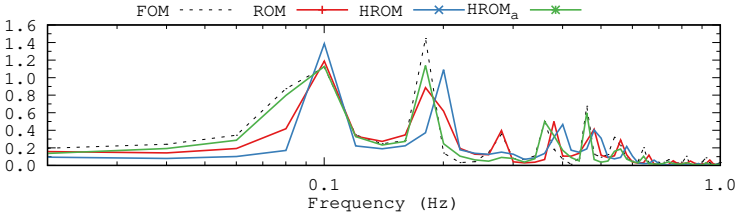


(a) Velocity magnitude.

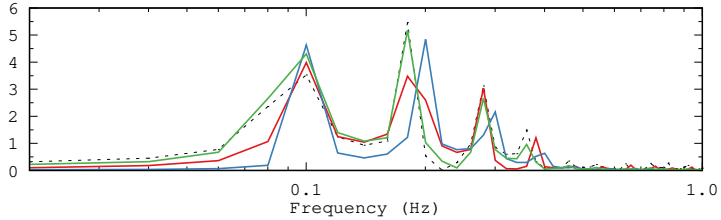


(b) Pressure.

Figure 12.31: Flow over a backward-facing step: comparison of FOM, ROM, hyper-ROM and AMR hyper-ROM at the control point $(22, 1)$.



(a) Velocity magnitude.



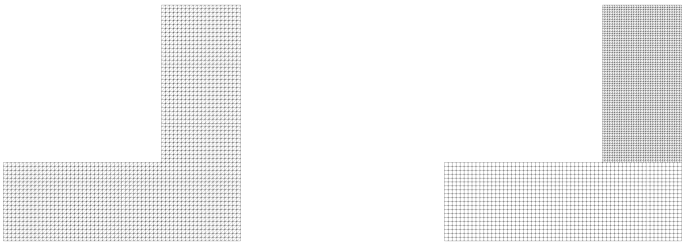
(b) Pressure.

Figure 12.32: Flow over a backward-facing step: Fourier transform of FOM, ROM, hyper-ROM and AMR hyper-ROM at the control point $(22, 1)$.

12.3 FLOW IN A L-SHAPED DOMAIN

In this example we solve the flow in the L-shaped domain presented in chapter 9. The computational domain consists in the union of two rectangles $[0,3] \times [0,1]$ and $[2,3] \times [2,3]$. The inlet is taken at $x = 0$, with a discontinuous inflow velocity $\mathbf{u} = (1,0)$ for $0 \leqslant y \leqslant \frac{1}{2}$ and $\mathbf{u} = (0,0)$ for $\frac{1}{2} < y \leqslant 1$, the outflow (where both the x and y velocity components are left free) is set at $y = 3$, whereas the rest of the boundaries have a prescribed null velocity. The viscosity is prescribed as $\nu = 0.005$.

To evaluate the performance of the domain decomposition algorithm, we solve the described problem using 2 approaches: an unified domain and a decomposed one —this one separated in the two rectangles described before. For the joined domain we use a mesh of 4000 triangular linear elements with an element size $h = 0.05$, and for the decomposed domain we use a mesh of of 1200 quadrilateral linear elements with an element size $h = 0.05$ for the lower rectangle and a mesh of 4356 triangular linear elements with an element size $h = 0.03$ for the upper one. Figure 12.33 shows a comparison of the meshes for the unified and decomposed domains.

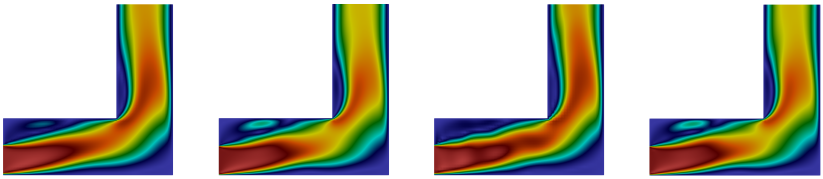


(a) Unified mesh. (b) Decomposed mesh.

Figure 12.33: Flow in an L-shaped domain: Comparison of meshes for unified and decomposed domains.

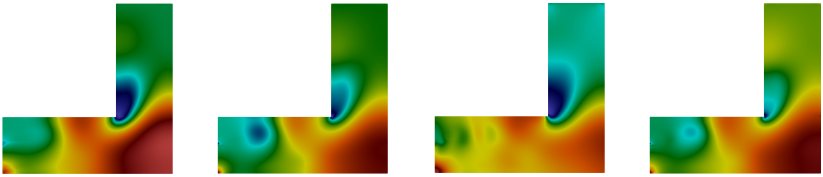
We solve both domain decomposition cases —FOM and ROM— using a Dirichlet condition in the first subdomain (from the second subdomain) and a Gauss-Seidel coupling, and setting $\alpha = 10.0/h$ (recall that it should scale as $1/h$ upon changes of mesh size) and $c_E = 0.001$. Moreover, we construct local bases for each subdomain, which means that the bases for both cases —unified or decomposed— have different relation between the retained energy η and the size of the basis r.

For all the cases we use a time step of $\delta t = 0.01$ and we construct the reduced basis over the 300 initial time steps of simulation. Likewise, the ROM simulations and the comparison with the FOM are done over the initial transient period $t = [0, 3]$. We choose to solve all the ROM cases using the same amount of retained energy $\eta = 0.95$ regardless the amount of basis vectors, yielding $r = 8$ for the unified mesh, $r = 5$ for the lower subdomain and $r = 6$ for the upper subdomain.



(a) FOM. (b) FOM-DD. (c) ROM (d) ROM-DD.

Figure 12.34: Flow in an L-shaped domain: velocity contours for FOM and ROM at $t = 3$, using unified and decomposed (DD) meshes.



(a) FOM. (b) FOM-DD. (c) ROM. (d) ROM-DD.

Figure 12.35: Flow in an L-shaped domain: pressure contours for FOM and ROM at $t = 3$, using unified and decomposed (DD) meshes.

Figures 12.34 and 12.35 display contour plots of velocity and pressure for FOM and ROM using both domain discretizations —unified and decomposed, at $t = 3$.

Figure 12.36 compares the velocity and pressure over the cut between subdomains for all the cases at the last time step of the simulation, $t = 3$. Although the velocity solutions behave in a similar way, we see that there is a noticeable improvement when using the ROM in the decomposed domain, approximating more precisely the FOMs.

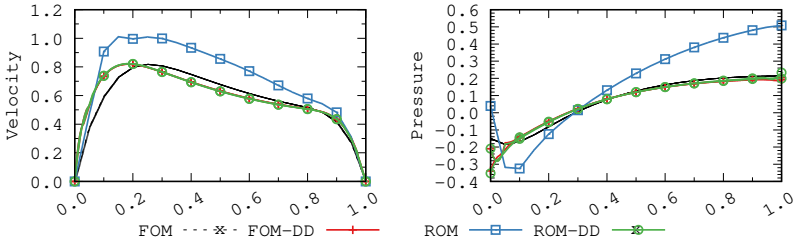


Figure 12.36: Flow in an L-shaped domain: velocity and pressure at $y = 1$ and $t = 3$.

In figure 12.37 we show the pressure at $x = 2$ and $x = 3$ for all the cases at $t = 3$, where we see how the pressure is in fact singular at the non-convex corner of the L. Interestingly, we also see how the ROM solution of the decomposed domain approximates more accurately the FOM solution than the one for the unified domain.

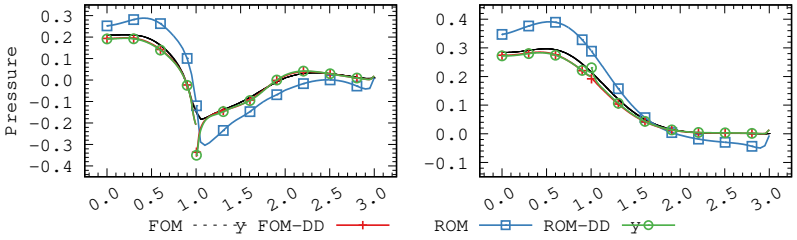


Figure 12.37: Flow in an L-shaped domain: pressure at $x = 2$ (left) and $x = 3$ (right) for $t = 3$.

13 BOUSSINESQ APPROXIMATION

In this chapter we extend the incompressible Navier-Stokes formulation in chapter 12 to thermally coupled flows using the Boussinesq approximation. The problem consists of finding a velocity $\mathbf{u} : \Omega\times]0, t_f[\to \mathbb{R}^d$, a pressure $p : \Omega\times]0, t_f[\to \mathbb{R}$ and a temperature $T : \Omega\times]0, t_f[\to \mathbb{R}$ such that

$$\begin{aligned}
\rho\partial_t\mathbf{u} + \rho\mathbf{u}\cdot\nabla\mathbf{u} - \mu\Delta\mathbf{u} + \nabla p = \rho\mathbf{f} - \rho\mathbf{g}\beta(T - T_r), \quad &\text{in } \Omega,\ t \in]0, t_f[,\\
\nabla\cdot\mathbf{u} = 0, \quad &\text{in } \Omega,\ t \in]0, t_f[,\\
\rho c_p\partial_t T + \rho c_p\mathbf{u}\cdot\nabla T - \nabla\cdot(k\nabla T) = Q, \quad &\text{in } \Omega,\ t \in]0, t_f[,
\end{aligned}$$

where, ρ denotes the density, μ the dynamic viscosity, $\mathbf{f}$ a vector of body forces, $\mathbf{g}$ the gravity acceleration vector, c_p the specific heat coefficient at constant pressure, k the thermal conductivity, β the thermal expansion coefficient, Q the heat source and T_r a reference temperature. Initial and boundary conditions are set for Γ, with $\Gamma_D^f \cup \Gamma_N^f = \Gamma$ and $\Gamma_D^f \cap \Gamma_N^f = \Gamma$ for $f = \mathbf{u}, T$, as

$$\begin{aligned}
\mathbf{u}(\mathbf{x}, 0) &= \mathbf{u}_0(\mathbf{x}) && \text{in } \Omega, \quad t \in]0, t_f[,\\
T(\mathbf{x}, 0) &= T_0(\mathbf{x}) && \text{in } \Omega, \quad t \in]0, t_f[,\\
\mathbf{u} &= \mathbf{u}_D && \text{on } \Gamma_D^{\mathbf{u}},\ t \in]0, t_f[,\\
T &= T_D && \text{on } \Gamma_D^{T},\ t \in]0, t_f[,\\
\boldsymbol{\mathcal{F}}(\mathbf{y}; [\mathbf{u}, p]) &= \mathbf{n}\cdot p\mathbf{I} - \mathbf{n}\cdot\mu\nabla\mathbf{u} = \mathbf{t}_N && \text{on } \Gamma_N^{\mathbf{u}},\ t \in]0, t_f[,\\
\boldsymbol{\mathcal{F}}(\mathbf{y}; T) &= \mathbf{n}\cdot k\nabla T = q_N && \text{on } \Gamma_N^{T},\ t \in]0, t_f[.
\end{aligned}$$

For the formulation of both FOM and ROM, we follow the stabilized formulation and the subscale approximation presented in [5, 44, 87] for FEs. Using the notation in chapter 12 we define the functional

spaces as follows. Let $\mathcal{V}_0 = (H_0^1(\Omega))^d$, $\mathcal{Q}_0 = L^2(\Omega)/\mathbb{R}$ and $\mathcal{W}_0 = H_0^1(\Omega)$, and define $\mathcal{V} = L^2(0, t_f; \mathcal{V}_0)$, $\mathcal{Q} = L^1(0, t_f; \mathcal{Q}_0)$ and $\mathcal{W} = L^2(0, t_f; \mathcal{W}_0)$. This way, the variational form of the Boussinesq problem is defined in the spaces $\mathcal{Y} = \mathcal{V} \times \mathcal{Q} \times \mathcal{W}$ for the trial solutions $-\mathbf{y} = [\mathbf{u}, p, T]-$ and $\mathcal{Y}_0 = \mathcal{V}_0 \times \mathcal{Q}_0 \times \mathcal{W}_0$ for the test functions $-\mathbf{v} = [\boldsymbol{v}, q, w]$.

We also define the terms that correspond to the abstract ROM ones in chapter 6 —and likewise for the FOM ones in chapter 5, using dynamic OSGS and assuming the force term to belong to the FE space (see tables 6.1 and 6.2), as

$$\boldsymbol{\mathcal{M}}(\mathbf{y}) = \begin{bmatrix} \rho\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0}^\top & 0 & 0 \\ \mathbf{0}^\top & 0 & \rho c_p \end{bmatrix}, \quad \boldsymbol{\mathcal{L}}(\mathbf{y}; \mathbf{y}_r) = \begin{bmatrix} \rho\mathbf{u} \cdot \nabla\mathbf{u}_r - \mu\Delta\mathbf{u}_r + \nabla p_r \\ \nabla \cdot \mathbf{u}_r \\ \rho c_p \mathbf{u} \cdot \nabla T_r - k\Delta T_r \end{bmatrix},$$

$$\boldsymbol{\mathcal{L}}^*(\mathbf{y}; \mathbf{v}_r) = \begin{bmatrix} -\rho\mathbf{u} \cdot \nabla\boldsymbol{v}_r - \mu\Delta\boldsymbol{v}_r - \nabla q_r \\ -\nabla \cdot \boldsymbol{v}_r \\ -\rho c_p \mathbf{u} \cdot \nabla w_r - k\Delta w_r \end{bmatrix},$$

$$\boldsymbol{\mathcal{F}}(\mathbf{y}; \mathbf{y}_r) = \begin{bmatrix} \mathbf{n} \cdot p_r\mathbf{I} - \mathbf{n} \cdot \nu\nabla\mathbf{u}_r \\ 0 \\ -k\mathbf{n} \cdot \nabla T_r \end{bmatrix}, \quad \boldsymbol{\mathcal{F}}^*(\mathbf{y}; \mathbf{y}_r) = \begin{bmatrix} \mathbf{n} \cdot q_r\mathbf{I} + \mathbf{n} \cdot \nu\nabla\boldsymbol{v}_r \\ 0 \\ k\mathbf{n} \cdot \nabla w_r \end{bmatrix},$$

yielding the forms

$$\begin{aligned} B(\mathbf{y}; \mathbf{y}_r, \mathbf{v}_r) &= \langle \rho\mathbf{u} \cdot \nabla\mathbf{u}_r, \boldsymbol{v}_r \rangle + (\mu\nabla\mathbf{u}_r - p_r\mathbf{I}, \nabla\boldsymbol{v}_r) \\ &\quad + (\nabla \cdot \mathbf{u}_r, q_r) + \langle \rho c_p \mathbf{u} \cdot \nabla T_r, w_r \rangle + (k\nabla T_r, \nabla w_r), \qquad (13.1a) \\ B_s(\mathbf{y}; \mathbf{y}_r, \mathbf{v}_r) &= -\sum_K \langle \breve{\boldsymbol{\Pi}}(\rho\partial_t\mathbf{u}_r), \tau_{1,t}(\rho\mathbf{u} \cdot \nabla\boldsymbol{v}_r + \mu\Delta\boldsymbol{v}_r + \nabla q_r) \rangle_K \\ &\quad - \sum_K \langle \breve{\boldsymbol{\Pi}}(\rho\mathbf{u} \cdot \nabla\mathbf{u}_r - \mu\Delta\mathbf{u}_r), \tau_{1,t}(\rho\mathbf{u} \cdot \nabla\boldsymbol{v}_r + \mu\Delta\boldsymbol{v}_r + \nabla q_r) \rangle_K \\ &\quad - \sum_K \langle \breve{\boldsymbol{\Pi}}(\nabla p_r), \tau_{1,t}(\rho\mathbf{u} \cdot \nabla\boldsymbol{v}_r + \mu\Delta\boldsymbol{v}_r + \nabla q_r) \rangle_K \\ &\quad - \sum_K \langle \breve{\boldsymbol{\Pi}}(\rho c_p \partial_t T_r, \tau_{3,t}(\rho c_p \mathbf{u} \cdot \nabla w_r + k\Delta w_r) \rangle_K \\ &\quad - \sum_K \langle \breve{\boldsymbol{\Pi}}(\rho c_p \mathbf{u} \cdot \nabla T_r - k\Delta T_r), \tau_{3,t}(\rho c_p \mathbf{u} \cdot \nabla w_r + k\Delta w_r) \rangle_K \end{aligned}$$

$$- \sum_K \langle \check{\boldsymbol{\Pi}}(\nabla \cdot \mathbf{u}_r), \tau_2 \nabla \cdot \boldsymbol{\nu}_r \rangle_K, \tag{13.1b}$$

$$\begin{aligned} B_b(\mathbf{y};\mathbf{y}_r,\boldsymbol{\upsilon}_r) = &- \sum_E \langle [\![\mathbf{n} \cdot p_r]\!][\![\boldsymbol{\tau}_{E,1}]\!], [\![\mathbf{n} \cdot q_r + \mathbf{n} \cdot \nu \nabla \boldsymbol{\nu}_r]\!] \rangle_E \\ &+ \sum_E \langle [\![\mathbf{n} \cdot \nu \nabla \mathbf{u}_r]\!][\![\boldsymbol{\tau}_{E,1}]\!], [\![\mathbf{n} \cdot q_r + \mathbf{n} \cdot \nu \nabla \boldsymbol{\nu}_r]\!] \rangle_E, \\ &- \sum_E \langle [\![\mathbf{n} \cdot k \nabla T_r]\!][\![\boldsymbol{\tau}_{E,2}]\!], [\![\mathbf{n} \cdot k \nabla w_r]\!] \rangle_E, \end{aligned} \tag{13.1c}$$

$$\begin{aligned} L_s(\mathbf{y};\boldsymbol{\upsilon}_r) = &- \sum_K \langle \check{\boldsymbol{\Pi}}(\mathbf{f}), \tau_{1,t}(\rho \mathbf{u} \cdot \nabla \boldsymbol{\nu}_r + \mu \Delta \boldsymbol{\nu}_r + \nabla q_r) \rangle_K \\ &- \sum_K \langle \delta t^{-1} \check{\mathbf{u}}, \tau_{1,t}(\rho \mathbf{u} \cdot \nabla \boldsymbol{\nu}_r + \mu \Delta \boldsymbol{\nu}_r + \nabla q_r) \rangle_K \\ &+ \delta t^{-1} \check{T}, \tau_{3,t}(\rho c_p \mathbf{u} \cdot \nabla w_r + k \Delta w_r) \rangle_K. \end{aligned} \tag{13.1d}$$

The stabilization parameter matrix is defined as

$$\begin{aligned} \boldsymbol{\tau}_K \mathbf{y}) &= \operatorname{diag}(\tau_1 \mathbf{I}, \tau_2, \tau_3) \\ &= \begin{bmatrix} (\frac{c_1 \mu}{h^2} + \frac{c_2 \rho |\mathbf{u}|_K}{h})^{-1} \mathbf{I} & 0 & 0 \\ 0^\top & \frac{\mu}{\rho} + \frac{c_2}{c_1} |\mathbf{u}| h & 0 \\ 0^\top & 0 & (\frac{c_1 k}{h^2} + \frac{c_2 \rho c_p |\mathbf{u}|_K}{h})^{-1} \end{bmatrix}, \end{aligned}$$

in $K \in \mathcal{T}_h$, with $c_1 = 4$ and $c_2 = 2$ for linear elements and $\tau_{i,t}^{-1} = \delta t^{-1} + \tau_i^{-1}$. And setting $c_E = 0.01$, we define the stabilization parameter matrix for the edges as

$$\boldsymbol{\tau}_E(\mathbf{y}) = \operatorname{diag}(\tau_{E,1} \mathbf{I}, 0, \tau_{E,2}) = \begin{bmatrix} (c_E \frac{\mu}{h})^{-1} \mathbf{I} & 0 & 0 \\ 0^\top & 0 & 0 \\ 0^\top & 0 & (c_E \frac{k}{h})^{-1} \end{bmatrix}.$$

We use a BDF time integration scheme of second order for the resolved scales and of first order for the subscales, with $\check{\mathbf{u}}$ and $\check{T}$ evaluated at t^j when solving for the unknowns evaluated at t^{j+1}. In our implementation, we have solved the problem in a standard iterative way, solving first for the velocity and the pressure and then for the temperature, and iterating until convergence.

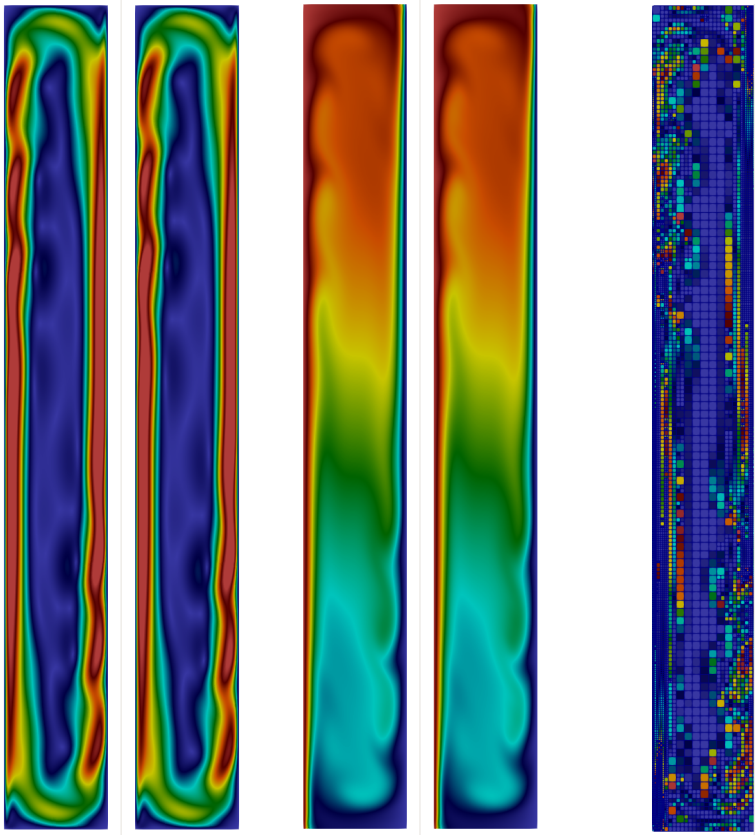
13.1 DIFFERENTIALLY HEATED CAVITY WITH ASPECT RATIO 8

For the first Boussinesq numerical example we model the flow in a differentially heated cavity with aspect ratio 8 as the one presented in [34, 44] with a Rayleigh number (defined below) $Ra = 5 \cdot 10^5$, where it is known that Hopf bifurcations occurs and the flow is oscillatory. A similar example using a ROM is presented in [17] for parameterized Rayleigh numbers and aspect ratios.

The computational domain is defined as $\Omega = [0, W] \times [0, H]$, with $H/W = 8$. The temperatures on the walls perpendicular to the x-coordinate (horizontal) are fixed to $T_h = 600$ (left) and $T_c = 300$ (right), while adiabatic boundary conditions are prescribed on the remaining walls. No slip conditions are set for the velocity over all walls. A gravity force $g = 6.5993$ is imposed in the negative y-coordinate (vertical). The initial temperature and density are set to $T_0 = 450$ and $\rho = 1$, respectively, and the dimensionless Prandtl and Rayleigh numbers are set to $Pr = \frac{c_p \mu}{\lambda} = 0.71$ and $Ra = \frac{\beta |g| \rho^2 c_p}{\lambda \mu}(T_h - T_c) = 5 \cdot 10^5$. The flow is considered an ideal gas with physical properties $R = 287.0$ (universal gas constant), $c_p = 1020$, $\mu = 0.0025$, $k = 3.59$ and $\beta = 0.00222$.

We solve the FOM and the ROM using a discretized domain of 80000 quadrilateral bilinear elements and a time step of $\delta t = 0.01$; we construct a ROM basis over 1000 time steps. We solve the uniform mesh hyper-ROM with a uniform mesh of 10476 elements. And lastly, we solve the AMR hyper-ROM, using a initial mesh of 1300 elements and a three-level refinement with a tolerance of $10^{-5} < \xi_K^2 < 10^{-4}$, yielding an average of 10220 elements.

Figure 13.1 shows a comparison of velocity and temperature contour plots between the FOM and the AMR hyper-ROM, and the refined mesh and the calculated ξ_K^2 at the time $t = 10$. We solve the three ROM cases using a basis with a retained energy $\eta = 0.9$ and $r = 13$ modes.



(a) Velocity contours. (b) Temperature contours. (c) AMR hyper-ROM ξ_K^2.

Figure 13.1: Differentially heated rectangular cavity: contour plots for FOM (left) and AMR hyper-ROM (right). AMR hyper-ROM ξ_K^2 and refined mesh. For $\eta = 0.9$, $r = 13$ and $t = 10$.

To compare the results and the performance of the AMR, we use the Nusselt number over the hot and cold walls, defined as

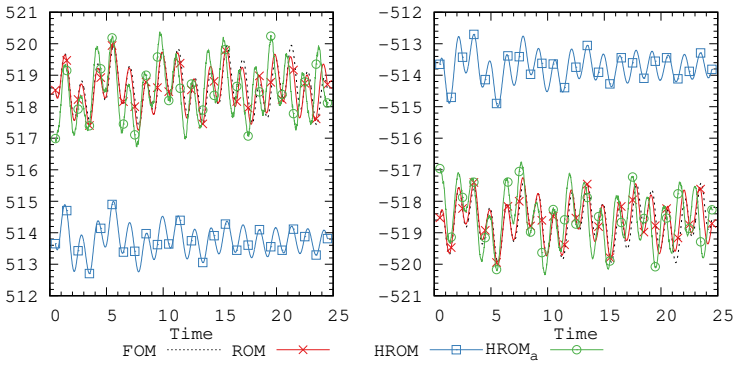
$$\mathrm{Nu}(t) = \left| \frac{1}{T_h - T_c} \int_0^H \frac{\partial T}{\partial x} \, dy \right|, \tag{13.2}$$

with $\frac{\partial T}{\partial x}$ evaluated at either $x = 0$ (hot wall) or $x = W$ (cold wall).

To evaluate the behavior of the formulation in time, we also choose a RMSD of the ROM solution with respect to the FOM one. With $Nu_{j,FOM}$ defined as the Nusselt number obtained with the FOM at time t^j, $j = 1, \ldots, S$, and $Nu_{j,ROM}$ the one obtained with any of the ROMs, we define RMSD of the Nusselt number as

$$Nu_{RMSD} = \sqrt{\frac{1}{S}\sum_{j=1}^{S}\left(Nu_{j,ROM} - Nu_{j,FOM}\right)^2}. \tag{13.3}$$

Figure 13.2 compares the Nusselt number for FOM, ROM, hyper-ROM and AMR hyper-ROM, and figure 13.3 shows the Fourier transform of the Nusselt number for the same cases. As expected, the use of a more appropriate mesh —the AMR one— improves the behaviour of the hyper-ROM solution, but contrary to the previous examples, the use of AMR adds noise at the end of the spectrum.



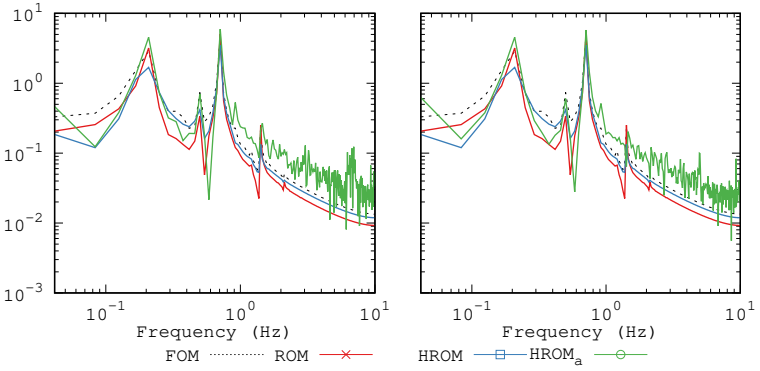
(a) Hot wall.

(b) Cold wall.

Figure 13.2: Differentially heated rectangular cavity: comparison of the Nusselt number between FOM, ROM, hyper-ROM and AMR hyper-ROM.

Table 13.1 shows the RMSD of the Nusselt number Nu_{RMSD}, the time average of the SGS norm $\overline{\xi^2}$ and the discrete time average of the Nusselt number $\overline{Nu}$ for ROM, hyper-ROM and AMR hyper-ROM. As expected, the RMSD results for the AMR hyper-ROM compared to the

uniform coarse mesh hyper-ROM are improved over integral quantities, with the average value of the Nusselt number approximating closely the ones for ROM.



(a) Hot wall.

(b) Cold wall.

Figure 13.3: Differentially heated rectangular cavity: Fourier transform of the Nusselt number for FOM, ROM, hyper-ROM and AMR hyper-ROM. Hot (left) and cold (right) walls.

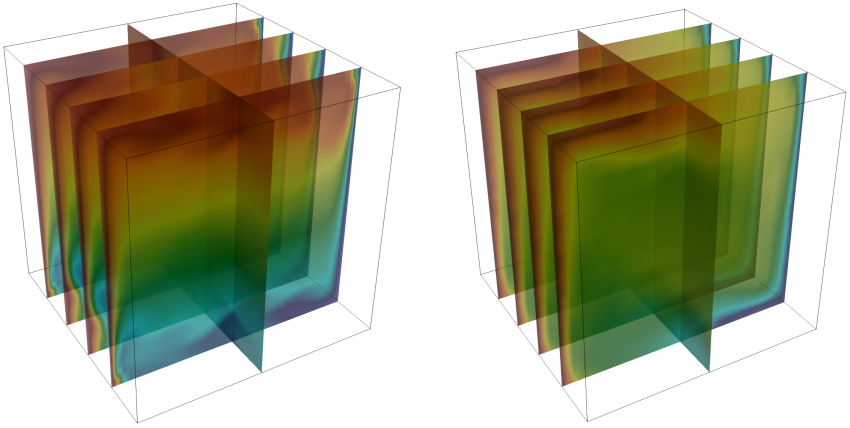
	ROM		Hyper-ROM	
	Hot	Cold	Hot	Cold
Nu_{RMSD}	0.2082	0.2082	4.8698	4.8698
$\overline{Nu}$	518.60	-518.60	513.77	-513.77
$\overline{\xi^2}$	0.3786		0.5956	

	AMR hyper-ROM	
	Hot	Cold
Nu_{RMSD}	0.8783	0.8586
$\overline{Nu}$	518.50	-518.52
$\overline{\xi^2}$	0.4895	

Table 13.1: Differentially heated rectangular cavity: Nu_{RMSD}, ξ^2 and $\overline{Nu}$ for ROM, hyper-ROM and AMR hyper-ROM

13.2 DIFFERENTIALLY HEATED CUBIC CAVITY

The second Boussinesq example consists in a 3D differentially heated cavity of aspect ratio 1 similar to the one solved in [7, 76]. The computational domain is defined as $\Omega = [0, L]^3$, with $L = 1$. The temperatures on the walls perpendicular to the x-coordinate (horizontal) are fixed to $T_h = 960$ and $T_c = 240$, while adiabatic boundary conditions are prescribed on the remaining walls. No slip conditions are set fot the velocities over all the walls. A gravity force $g = 12.035$ is imposed in the negative y-coordinate (vertical). The initial temperature is set to $T_0 = 600$, the density to $\rho_0 = 0.58841$, the dimensionless Prandtl number to $\mathrm{Pr} = \frac{c_p \mu}{\lambda} = 0.71$ and the Rayleigh number to $\mathrm{Ra} = \frac{\beta |g| \rho^2 c_p}{\lambda \mu}(T_h - T_c) = 3.55 \cdot 10^6$. As in the previous example the flow is considered an ideal gas with physical properties $R = 287.0$, $c_p = 1004.5$, $\mu = 0.0001$, $k = 1.4148$ and $\beta = 0.001666$.



(a) FOM. (b) AMR hyper-ROM.

Figure 13.4: Differentially heated cubic cavity: temperature contour plots for FOM (left) and AMR hyper-ROM (right) for $\eta = 0.95$, $r = 25$ and $t = 10$.

For the FOM and ROM, we use a uniform structured one composed of 64000 regular hexahedral elements of size $h = 0.025$. We collect 500 snapshots for velocity, pressure and temperature at every 2 time steps in a 10 seconds interval to construct the ROM basis. The three ROMs are solved using a basis with $\eta = 0.95$ and $r = 25$ modes.

We solve the AMR hyper-ROM starting from a mesh of 125 regular hexahedral elements, where following a three-level refinement with a refinement tolerance set as $5 \cdot 10^{-5} < \xi_K^2 < 5 \cdot 10^{-4}$ we reach an average of 16393 elements. And as in the previous examples, we compare it with a fixed and uniform mesh hyper-ROM of 15625 elements. Figure 13.4 shows a comparison of temperature contour plots between the FOM and the AMR hyper-ROM and figure 13.5 shows the refined mesh and the calculated ξ_K^2 at the time $t = 10$ for the AMR hyper-ROM.

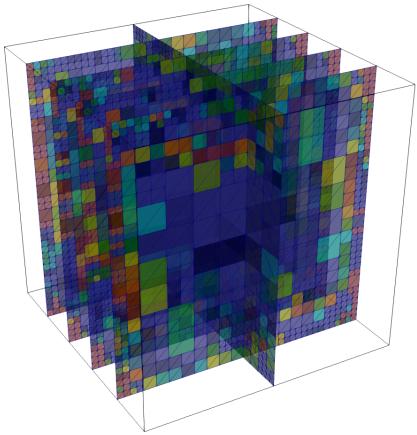
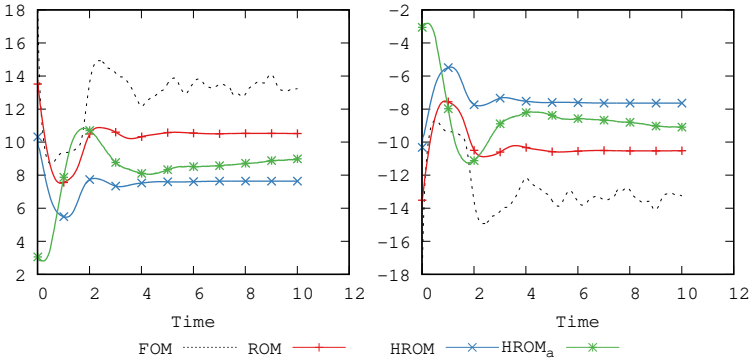


Figure 13.5: Differentially heated cubic cavity: AMR hyper-ROM ξ_K^2 with $5 \cdot 10^{-5} < \xi_K^2 < 5 \cdot 10^{-4}$. For $\eta = 0.95$, $r = 25$ and $t = 10$.

Figure 13.6 shows a comparison of the Nusselt number for FOM, ROM, hyper-ROM and AMR hyper-ROM, where the ROMs manage to capture the mean Nusselt number but not the whole behaviour. We believe that this could be attributed to the quality of the basis, which

depends on the FOM and the chosen physical model (see section 14.1).



(a) Hot wall. (b) Cold wall.

Figure 13.6: Differentially heated cubic cavity: Nusselt number evolution for FOM, ROM, hyper-ROM and AMR hyper-ROM.

Figure 13.7 shows a comparison of the Fourier transform of the Nusselt number for the hot and cold walls, where the AMR hyper-ROM shows a better approximation than the uniform mesh hyper-ROM.

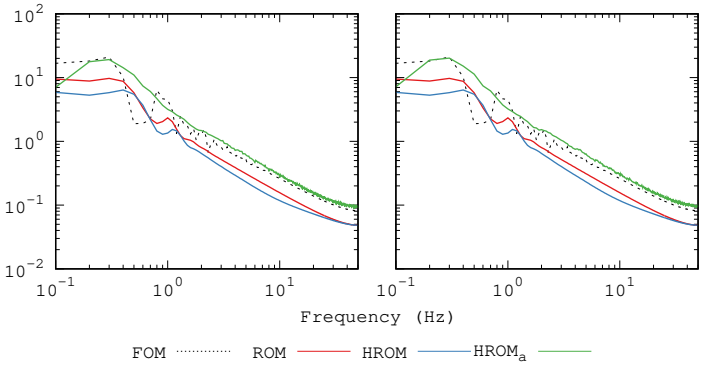


Figure 13.7: Differentially heated cubic cavity: Fourier transform of the Nusselt number for FOM, ROM, hyper-ROM and AMR hyper-ROM. Hot (left) and cold (right) walls.

14 LOW MACH NUMBER MODEL

In this chapter we solve a similar set of examples as the ones in chapter 13 using a more strongly coupled formulation: the zero Mach limit Navier-Stokes model presented in [84, 89].

This model is based on the splitting of the pressure into two relevant parts: mechanical p, which behaves as in the incompressible and Boussinesq approximations; and thermodynamic p^{th}, which is constant over the domain Ω but dependent on the energy added in such domain and therefore time dependent. Hence, we can define the low Mach number model as finding a velocity $\mathbf{u} : \Omega \to \mathbb{R}^d$, a pressure $p : \Omega \to \mathbb{R}$ and a temperature $T : \Omega \to \mathbb{R}$ such that

$$\begin{aligned}
\rho \partial_t \mathbf{u} + \rho \mathbf{u} \cdot \nabla \mathbf{u} - \nabla \cdot (2\mu \mathbf{s}(\mathbf{u})) + \nabla p &= \rho \mathbf{g}, \quad \text{in } \Omega, \ t \in]0, t_f[, \\
\partial_t \rho + \nabla \cdot (\rho \mathbf{u}) &= 0, \quad \text{in } \Omega, \ t \in]0, t_f[, \\
\rho c_p \partial_t T + \rho c_p \mathbf{u} \cdot \nabla T - \nabla \cdot (k \nabla T) - \beta T \frac{dp^{th}}{dt} &= Q, \quad \text{in } \Omega, \ t \in]0, t_f[,
\end{aligned}$$

where $\mathbf{s}(\mathbf{u}) = \nabla^s \mathbf{u} - \frac{1}{3} \nabla \cdot \mathbf{u}$, $\nabla^s \mathbf{u}$ denotes the symmetrical part of $\nabla \mathbf{u}$, ρ the density, μ the dynamic viscosity, $\mathbf{f}$ a vector of body forces, $\mathbf{g}$ the gravity acceleration vector, c_p the specific heat coefficient at constant pressure, γ the heat capacity ratio, k the thermal conductivity, β the thermal expansion coefficient, Q the heat source and T_0 a reference temperature.

Considering that a state equation is necessary to close the system, for simplicity, in this work we focus exclusively on ideal gases, with $p = \rho R T$ as the state equation, R the specific gas constant and $\beta = \frac{1}{T}$.

Initial and boundary conditions are set for Γ, with $\Gamma_D^f \cup \Gamma_N^f = \Gamma$ and $\Gamma_D^f \cap \Gamma_N^f = \Gamma$ for $f = \mathbf{u}, T$, as

$$\begin{aligned}
\mathbf{u}(\mathbf{x}, 0) &= \mathbf{u}_0(\mathbf{x}) && \text{in } \Omega, \quad t \in]0, t_f[, \\
T(\mathbf{x}, 0) &= T_0(\mathbf{x}) && \text{in } \Omega, \quad t \in]0, t_f[, \\
\mathbf{u} &= \mathbf{u}_D && \text{on } \Gamma_D^{\mathbf{u}}, \ t \in]0, t_f[, \\
T &= T_D && \text{on } \Gamma_D^{T}, \ t \in]0, t_f[, \\
\boldsymbol{\mathcal{F}}(\mathbf{y}; [\mathbf{u}, p]) &= \mathbf{n} \cdot p\mathbf{I} - \mathbf{n} \cdot (2\mu \mathbf{s}(\mathbf{u})) = \mathbf{t}_N && \text{on } \Gamma_N^{\mathbf{u}}, \ t \in]0, t_f[, \\
\mathcal{F}(\mathbf{y}; T) &= \mathbf{n} \cdot k\nabla T = q_N && \text{on } \Gamma_N^{T}, \ t \in]0, t_f[.
\end{aligned}$$

Finally, the formulation is complete with the thermodynamic pressure equations: for open flows ($\Gamma_N^{\mathbf{u}} \neq \emptyset$), where the thermodynamic pressure is determined through global conservation equations over Ω (equation (14.1a)); and for closed flows ($\Gamma_N^{\mathbf{u}} = \emptyset$), where the thermodynamic pressure is determined by the boundary conditions (equation (14.1b)):

$$p^{th} = p_0^{th} \frac{\int_\Omega \frac{1}{T_0} d\Omega}{\int_\Omega \frac{1}{T} d\Omega} \tag{14.1a}$$

$$\frac{|\Omega|}{(\gamma - 1)} \frac{dp^{th}}{dt} + \frac{\gamma p^{th}}{\gamma - 1} \int_{\partial\Omega} \mathbf{n} \cdot \mathbf{u} d\Gamma = \int_\Omega Q d\Omega + \int_{\partial\Omega} \mathbf{n} \cdot \lambda \nabla T d\Gamma \tag{14.1b}$$

Since for the zero Mach limit Navier-Stokes equations, the minimum regularity required is only known in very particular cases (see [5]). We denote the spaces for the trial solutions as $\mathcal{Y} = \mathcal{V} \times \mathcal{Q} \times \mathcal{W}$ —with $\mathbf{y} = [\mathbf{u}, p, T]$— and the corresponding ones for the test functions as $\mathcal{Y}_0 = \mathcal{V}_0 \times \mathcal{Q}_0 \times \mathcal{W}_0$ —with $\boldsymbol{\upsilon} = [\boldsymbol{v}, q, w]$.

Subscales approximation

To approximate the subscales in this problems we follow [6, 7, 87], where the temporal derivative of the subscales $\rho\partial_t \check{\mathbf{u}}$ and $\rho\partial_t \check{T}$ are substituted by their conservative counterparts momentum $\partial_t(\rho\check{\mathbf{u}})$ and

energy $\partial_t(\rho\check{T})$. To appropriately make these substitutions, we use in equations (6.6) and (6.7) the product rule of the derivative as

$$(\rho\partial_t\check{\mathbf{u}}, \mathbf{v}_r) + (\check{\mathbf{u}}\partial_t\rho, \mathbf{v}_r) = (\partial_t(\rho\check{\mathbf{u}}), \mathbf{v}_r)$$
$$(\rho\partial_t\check{T}, w_r) + (\check{T}\partial_t\rho, w_r) = (\partial_t(\rho\check{T}), w_r),$$

and the continuity equation to replace the temporal derivative of the density in the momentum and energy parts of equation (6.6) as

$$-\nabla\cdot(\rho(\mathbf{u}_r + \check{\mathbf{u}})) = \partial_t\rho.$$

This way, we can rewrite equation (6.6) as

$$\begin{aligned}
\partial_t(\rho\check{\mathbf{u}}_r) + \tau_1^{-1}\check{\mathbf{u}}_r &= \rho\mathbf{g} - \rho\partial_t\mathbf{u}_r - \rho(\mathbf{u}_r + \check{\mathbf{u}})\cdot\nabla\mathbf{u}_r + \nabla\cdot(2\mu\mathbf{s}(\mathbf{u}_r)) - \nabla p_r,\\
\tau_2^{-1}\check{p} &= -\partial_t\rho - \rho\nabla\cdot\mathbf{u}_r + \frac{\rho}{T_r + \check{T}}(\mathbf{u}_r + \check{\mathbf{u}})\cdot\nabla T_r,\\
c_p\partial_t(\rho\check{T}_r) + \tau_3^{-1}\check{T}_r &= Q + \frac{dp^{th}}{dt} - \rho c_p(\partial_t T_r + (\mathbf{u}_r + \check{\mathbf{u}})\cdot\nabla T_r) + \nabla\cdot(k\nabla T_r).
\end{aligned}$$

Using the modified SGS equations, choosing the subscales as dynamic OSGS and assuming the force terms belonging to the FE space (see tables 6.1 and 6.2), we can define the abstract terms for the ROM in chapter 6 —and likewise for the FOM in chapter 5— as

$$\boldsymbol{\mathcal{M}}(\mathbf{y}) = \begin{bmatrix} \rho\mathbf{I} & \mathbf{0} & \mathbf{0}\\ \mathbf{0}^\top & 0 & 0\\ \mathbf{0}^\top & 0 & \rho c_p \end{bmatrix},\ \boldsymbol{\mathcal{L}}(\mathbf{y};\mathbf{y}_r) = \begin{bmatrix} \rho\mathbf{u}\cdot\nabla\mathbf{u}_r - \nabla\cdot 2\mu\mathbf{s}(\mathbf{u}_r) + \nabla p_r\\ \nabla\cdot\rho\mathbf{u}_r\\ \rho c_p\mathbf{u}\cdot\nabla T_r - \nabla\cdot k\nabla T_r \end{bmatrix},$$

$$\boldsymbol{\mathcal{L}}^*(\mathbf{y};\mathbf{v}_r) = \begin{bmatrix} -\rho\mathbf{u}\cdot\nabla\mathbf{v}_r - \nabla\cdot 2\mu\mathbf{s}(\mathbf{v}_r) - \nabla q_r\\ -\nabla\cdot\rho\mathbf{v}_r\\ -\rho c_p\mathbf{u}\cdot\nabla w_r - \nabla\cdot k\nabla w_r \end{bmatrix},$$

$$\boldsymbol{\mathcal{F}}(\mathbf{y};\mathbf{y}_r) = \begin{bmatrix} \mathbf{n}\cdot p_r\mathbf{I} - \mathbf{n}\cdot 2\mu\mathbf{s}(\mathbf{u}_r)\\ \mathbf{0}\\ -\mathbf{n}\cdot k\nabla T_r \end{bmatrix},\ \boldsymbol{\mathcal{F}}^*(\mathbf{y};\mathbf{y}_r) = \begin{bmatrix} \mathbf{n}\cdot q_r\mathbf{I} + \mathbf{n}\cdot 2\mu\mathbf{s}(\mathbf{v}_r)\\ \mathbf{0}\\ \mathbf{n}\cdot k\nabla w_r \end{bmatrix},$$

yielding the forms

$$
\begin{aligned}
B(\mathbf{y};\mathbf{y}_r,\mathbf{v}_r) &= \langle \rho\mathbf{u}\cdot\nabla\mathbf{u}_r,\boldsymbol{\nu}_r\rangle + (2\mu\mathbf{s}(\mathbf{u}_r),\nabla^s\boldsymbol{\nu}_r) - (p_r\mathbf{I},\nabla\boldsymbol{\nu}_r) - (\rho\mathbf{u}_r,\nabla q_r)\\
&\quad + \langle \rho\mathbf{n}\cdot\mathbf{u}_r,q_r\rangle_{\partial K} + \langle \rho c_p\mathbf{u}\cdot\nabla T_r,w_r\rangle + (k\nabla T_r,\nabla w_r), \qquad (14.2a)\\
B_s(\mathbf{y};\mathbf{y}_r,\mathbf{v}_r) &= -\sum_K \langle \breve{\Pi}(\rho\partial_t\mathbf{u}_r),\tau_{1,t}(\rho\mathbf{u}\cdot\nabla\boldsymbol{\nu}_r + 2\mu\mathbf{s}(\boldsymbol{\nu}_r) + \rho\nabla q_r)\rangle_K\\
&\quad - \sum_K \langle \breve{\Pi}(\rho\mathbf{u}\cdot\nabla\mathbf{u}_r),\tau_{1,t}(\rho\mathbf{u}\cdot\nabla\boldsymbol{\nu}_r + 2\mu\mathbf{s}(\boldsymbol{\nu}_r) + \rho\nabla q_r)\rangle_K\\
&\quad + \sum_K \langle \breve{\Pi}(\nabla\cdot 2\mu\mathbf{s}(\mathbf{u}_r)),\tau_{1,t}(\rho\mathbf{u}\cdot\nabla\boldsymbol{\nu}_r + 2\mu\mathbf{s}(\boldsymbol{\nu}_r) + \rho\nabla q_r)\rangle_K\\
&\quad - \sum_K \langle \breve{\Pi}(\nabla p_r),\tau_{1,t}(\rho\mathbf{u}\cdot\nabla\boldsymbol{\nu}_r + 2\mu\mathbf{s}(\boldsymbol{\nu}_r) + \rho\nabla q_r)\rangle_K\\
&\quad - \sum_K \langle \breve{\Pi}(\rho c_p\partial_t T_r,\tau_{3,t}(\rho c_p\mathbf{u}\cdot\nabla w_r + \nabla\cdot k\nabla w_r)\rangle_K\\
&\quad - \sum_K \langle \breve{\Pi}(\rho c_p\mathbf{u}\cdot\nabla T_r),\tau_{3,t}(\rho c_p\mathbf{u}\cdot\nabla w_r + \nabla\cdot k\nabla w_r)\rangle_K\\
&\quad + \sum_K \langle \breve{\Pi}(\nabla\cdot k\nabla T_r),\tau_{3,t}(\rho c_p\mathbf{u}\cdot\nabla w_r + \nabla\cdot k\nabla w_r)\rangle_K\\
&\quad - \sum_K \langle \breve{\Pi}(\rho\nabla\cdot\mathbf{u}_r - \rho(\mathbf{u}_r+\breve{\mathbf{u}})\cdot\nabla T_r),\tau_2\nabla\cdot\boldsymbol{\nu}_r\rangle_K, \qquad (14.2b)\\
B_b(\mathbf{y};\mathbf{y}_r,\mathbf{v}_r) &= -\sum_E \langle [\![\mathbf{n}\cdot p_r]\!][\![\boldsymbol{\tau}_{E,1}]\!],[\![\mathbf{n}\cdot q_r + \mathbf{n}\cdot 2\mu\mathbf{s}(\boldsymbol{\nu}_r)]\!]\rangle_E\\
&\quad + \sum_E \langle [\![\mathbf{n}\cdot 2\mu\mathbf{s}(\mathbf{u}_r)]\!][\![\boldsymbol{\tau}_{E,1}]\!],[\![\mathbf{n}\cdot q_r + \mathbf{n}\cdot 2\mu\mathbf{s}(\boldsymbol{\nu}_r)]\!]\rangle_E,\\
&\quad - \sum_E \langle [\![\mathbf{n}\cdot k\nabla T_r]\!][\![\boldsymbol{\tau}_{E,2}]\!],[\![\mathbf{n}\cdot k\nabla w_r]\!]\rangle_E, \qquad (14.2c)\\
L_s(\mathbf{y};\mathbf{v}_r) &= -\sum_K \langle \breve{\Pi}(\mathbf{f}),\tau_{1,t}(\rho\mathbf{u}\cdot\nabla\boldsymbol{\nu}_r + 2\mu\mathbf{s}(\boldsymbol{\nu}_r) + \rho\nabla q_r)\rangle_K\\
&\quad - \sum_K \langle \delta t^{-1}\breve{\mathbf{u}},\tau_{1,t}(\rho\mathbf{u}\cdot\nabla\boldsymbol{\nu}_r + 2\mu\mathbf{s}(\boldsymbol{\nu}_r) + \rho\nabla q_r)\rangle_K\\
&\quad + \sum_K \langle \breve{\Pi}\left(Q + \frac{dp^{th}}{dt}\right),\tau_{3,t}(\rho c_p\mathbf{u}\cdot\nabla w_r + \nabla\cdot k\nabla w_r)\rangle_K\\
&\quad + \sum_K \langle \delta t^{-1}\breve{T},\tau_{3,t}(\rho c_p\mathbf{u}\cdot\nabla w_r + \nabla\cdot k\nabla w_r)\rangle_K\\
&\quad - \sum_K \langle \breve{\Pi}(\partial_t\rho),\tau_2\nabla\cdot\boldsymbol{\nu}_r\rangle_K. \qquad (14.2d)
\end{aligned}
$$

In this case, we use the definition of the stabilization parameter matrix defined in [37, 88] as

$$\boldsymbol{\tau}_K(\mathbf{y}) = \operatorname{diag}(\tau_1\mathbf{I}, \tau_2, \tau_3) = \begin{bmatrix} (\frac{c_1\mu}{h^2} + \frac{c_2\rho|\mathbf{u}|_K}{h})^{-1}\mathbf{I} & 0 & 0 \\ 0^\top & \frac{\mu}{\rho} + \frac{c_2}{c_1}|\mathbf{u}|h & 0 \\ 0^\top & 0 & (\frac{c_1 k}{h^2} + \frac{c_2\rho c_p|\mathbf{u}|_K}{h})^{-1} \end{bmatrix},$$

in $K \in \mathcal{T}_h$, with $c_1 = 4$ and $c_2 = 2$ for linear elements and $\tau_{i,t}^{-1} = \delta t^{-1} + \tau_i^{-1}$. And, setting $c_E = 0.01$, we define the stabilization parameter in the edges as

$$\boldsymbol{\tau}_E(\mathbf{y}) = \operatorname{diag}(\tau_{E,1}\mathbf{I}, 0, \tau_{E,2}) = \begin{bmatrix} (c_E\frac{\mu}{h})^{-1}\mathbf{I} & 0 & 0 \\ 0^\top & 0 & 0 \\ 0^\top & 0 & (c_E\frac{k}{h})^{-1} \end{bmatrix}.$$

As in the previous chapters, we use a BDF time integration scheme of second order for the resolved scales and of first order for the subscales, with $\check{\mathbf{u}}$ and $\check{T}$ evaluated at t^j when solving for the unknowns evaluated at t^{j+1}.

14.1 DIFFERENTIALLY HEATED CAVITY WITH ASPECT RATIO 1

The first numerical example for zero Mach limit Navier-Stokes problem consists of 2D and 3D versions of the initial transient part of a differentially heated cavity with aspect ratio 1 —similar to the one presented in [7, 76] and solved using the Boussinesq approximation in section 13.2. In both examples the flow is considered an ideal gas with physical properties $R = 287.0$, $c_p = 1004.5$, and $\mu = 0.001$.

The computational domain for both problems is defined as $\Omega = [0, L] \times [0, L]$ for the 2D problem, and $\Omega = [0, L] \times [0, L] \times [0, L]$ for the 3D problem, with $L = 1$. The temperatures on the walls perpendicular to the x-coordinate (horizontal) are fixed to $T_h = 960$ and

$T_c = 240$; while adiabatic boundary conditions are prescribed in the remaining walls. Additionally, no slip and impermeable conditions are set over all the walls, together with a homogeneous gravity force g prescribed in the negative y-coordinate (vertical). The initial thermodynamic pressure, temperature and density are set to $p_0^{th} = 101325$, $T_0 = 600$, and $\rho_0 = 0.58841$ respectively, and the dimensionless Prandtl and Rayleigh numbers are set to $Pr = \frac{c_p \mu}{\lambda} = 0.71$ and $Ra = \frac{2|g|\rho^2 c_p}{\lambda\mu} \frac{(T_h - T_c)}{(T_h + T_c)} = 3.55 \cdot 10^6$. For both cases, we use a constant time step size of $\delta t = 0.01$.

14.1.1 *Two dimensional case*

In the 2D problem, we use 2 uniform structured meshes composed of quadrilateral elements: one with 10000 elements and a mesh size $h = 0.01$, used for the solving the FOM and the ROM; and one with 2500 elements and a mesh size $h = 0.02$, for testing the hyper-ROM formulation. To construct the basis, we collect 500 snapshots for velocity, pressure and temperature at every 4 time steps in a 20 seconds interval.

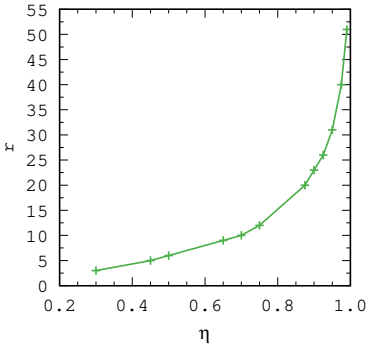
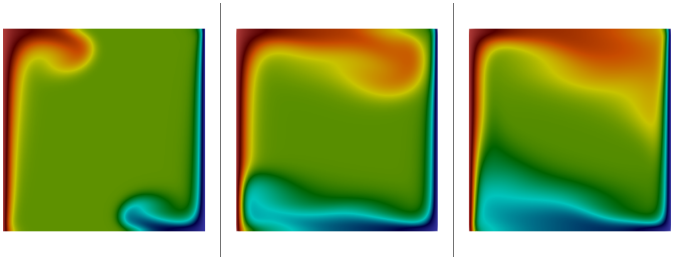
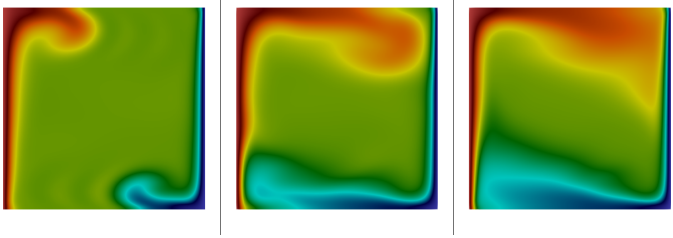


Figure 14.1: Differentially heated square cavity: number of basis vectors r as a function of retained energy η.

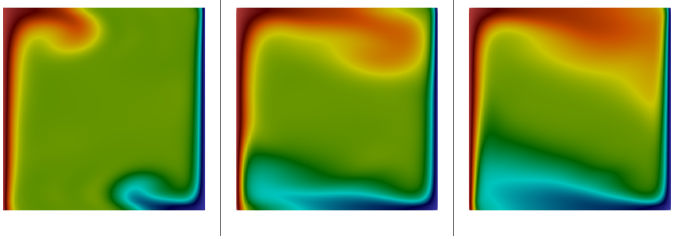
To evaluate the accuracy of the ROM and hyper-ROM formulations, we perform a series of numerical tests using 12 different sets of basis, varying the retained energy from $\eta_1 = 0.99$ to $\eta_{12} = 0.3$. Figure 14.1 depicts the number of vectors in the basis as a function of the retained energy. Figure 14.2 shows a comparison of temperature contours at different time steps for the FOM, ROM, and hyper-ROM, using a basis with a retained energy $\eta = 0.9$ and $r = 23$ basis vectors.



(a) FOM.



(b) ROM with $\eta = 0.9$ and $r = 23$.



(c) Hyper-ROM with $\eta = 0.9$ and $r = 23$.

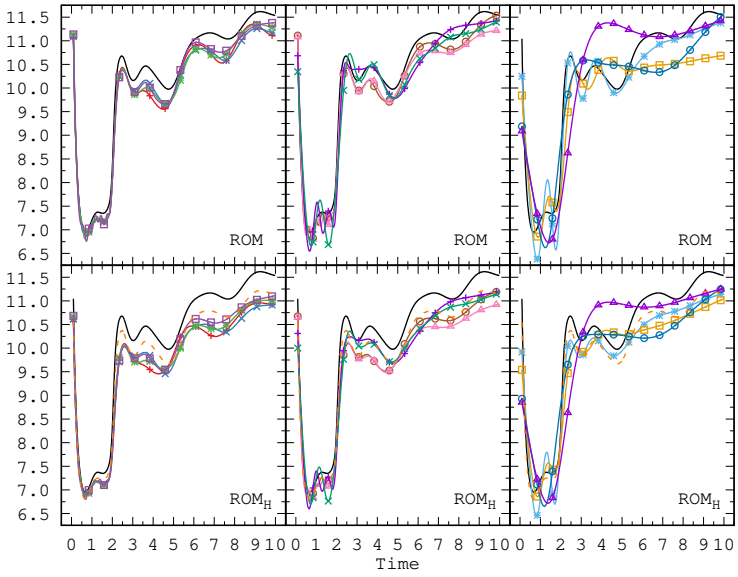
Figure 14.2: Differentially heated square cavity: temperature contour plots at $t = 1$, $t = 2$, and $t = 3$ for the 2D differentially heated cavity.

To compare the solutions we follow the same approach as in chapter 13, using the Nusselt number defined in equation (13.2). But since in this problem, we are computing an initial transient period of the physical phenomena we measure the quality of the solution using a root mean square of a discrete error instead of the RMSD described in equation (13.3). The discrete L^2-norm error of the Nusselt number —for each time step i— over all the nodes of coordinates x^a of a boundary Γ is defined as

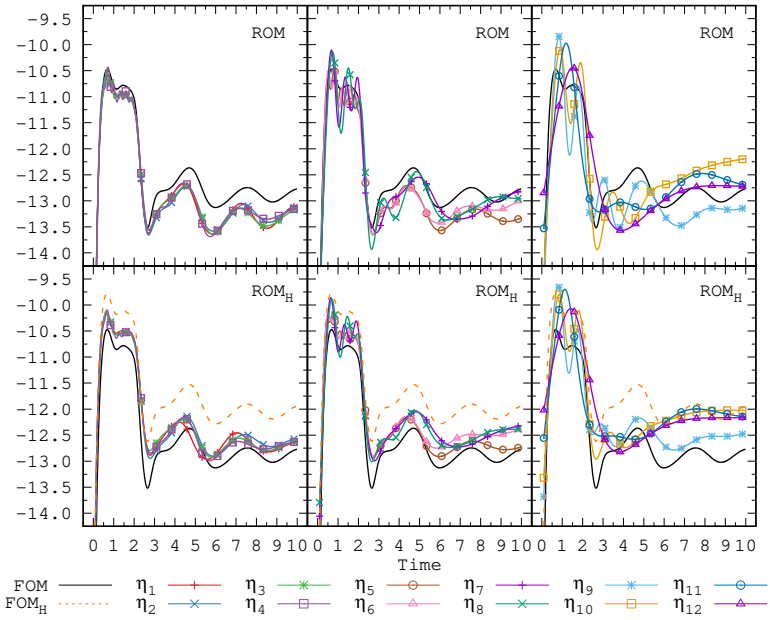
$$\epsilon_i = \frac{\sum_a \left((\mathrm{Nu}_{\mathrm{ROM}}(x^a) - \mathrm{Nu}_{\mathrm{FOM}}(x^a)) \right)^2}{\sum_a \left(\mathrm{Nu}_{\mathrm{FOM}}(x^a) \right)^2}, \tag{14.3}$$

and the root squared mean of ϵ_i —for N time steps— is defined as

$$\epsilon_{\mathrm{rms}} = \sqrt{\sum_{i=1}^{N} \left(\frac{\epsilon_i^2}{N} \right)}. \tag{14.4}$$



(a) Hot wall.



(b) Cold wall.

Figure 14.3: Differentially heated square cavity: Nusselt number evolution for FOM, ROM and hyper-ROM with a retained energy from $\eta_1 = 0.99$ to $\eta_{12} = 0.3$.

Figure 14.3 shows a comparison of the Nusselt number for FOM, ROM, and hyper-ROM. As expected, we observe a more diffusive behavior, in both the mean and the fluctuation of the Nusselt number, when fewer basis vectors are included. The hyper-ROM results appear to have the same behavior of the ROM with lower amplitude. Additionally, we include a FOM solution (labeled FOM_H) using the coarser mesh to evaluate how the hyper-ROM formulation relates to a FE mesh coarsening.

Figure 14.4 shows the convergence of ϵ_{RMS} over the hot and cold walls, and the average of them $\bar{\epsilon}_{\text{RMS}}$. We also include the convergence for the FOM solved using the coarse mesh. Although the convergence error does not have a clear slope, it behaves as expected, with the error

decreasing with the addition of basis vectors. It is important to notice that appears to be an optimal value for $\eta \neq 1$ —the maximum number of basis vectors, where the error reaches the minimum; considering that this results occur near $\eta = 1$, we attribute it to an *over-fitting* phenomenon as in section 12.2. The overall results for the hyper-ROM seem adequate, given that the error is the same order of the ROM and the coarse mesh FOM solutions.

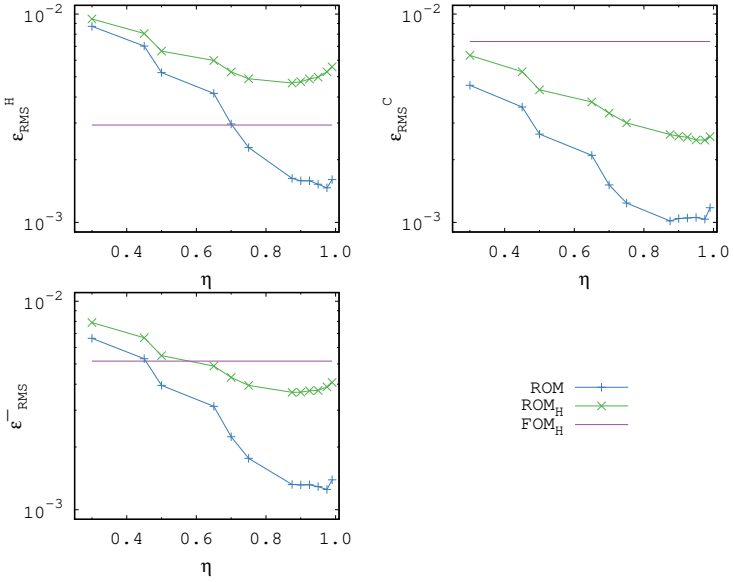
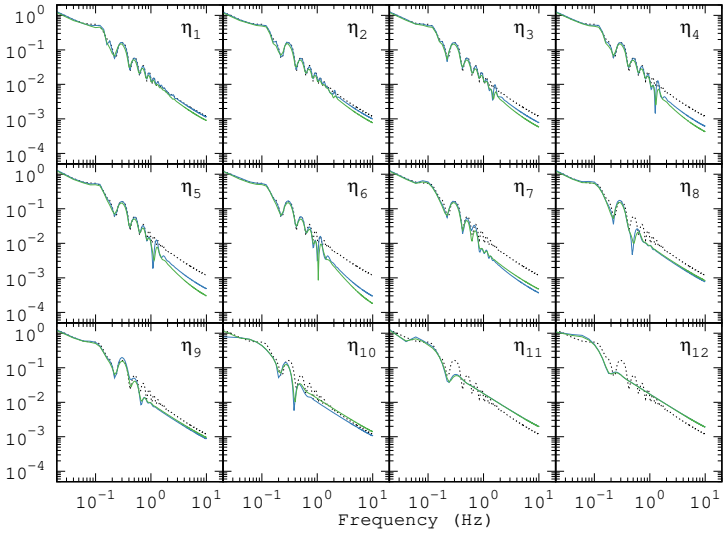
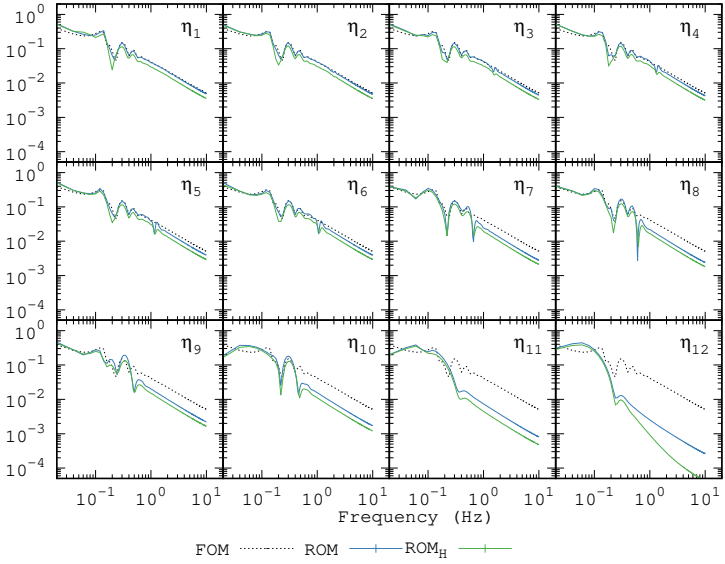


Figure 14.4: Differentially heated square cavity: ϵ_{RMS} convergence. Hot wall ϵ^{H}_{RMS}, cold wall ϵ^{C}_{RMS}, and mean $\bar{\epsilon}_{RMS}$.

To evaluate the temporal evolution we perform a discrete Fourier transform of the Nusselt number. In figure 14.5 we compare the Nusselt number spectra for the hot and cold walls respectively. Again we see how reducing the amount of basis vectors leads to a more diffusive solution. But in contrast to figure 14.4, in Nusselt number spectra we observe that the ROM and hyper-ROM spectra tend to the FOM spectra as we approach $\eta = 1$.



(a) Hot wall.



(b) Cold wall.

Figure 14.5: Differentially heated square cavity: Fourier transform of the Nusselt number for FOM, ROM and hyper-ROM with a retained energy from $\eta_1 = 0.99$ to $\eta_{12} = 0.3$.

Figure 14.6 shows the computational time ratio for all the cases under the same initial conditions, time step, and computer configuration. As expected the computational time is reduced when the coarser mesh is used. It is important to note that the computational time for the ROM and hyper-ROM is decreased not only by the solution of a smaller linear system but as well for having fewer non-linear iterations.

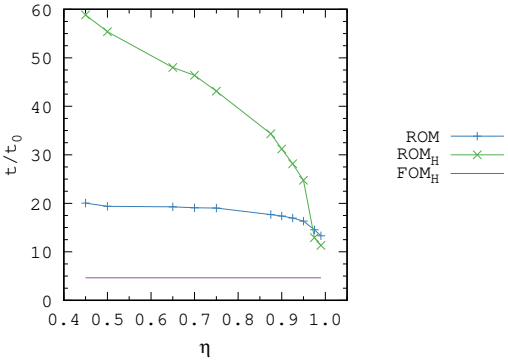


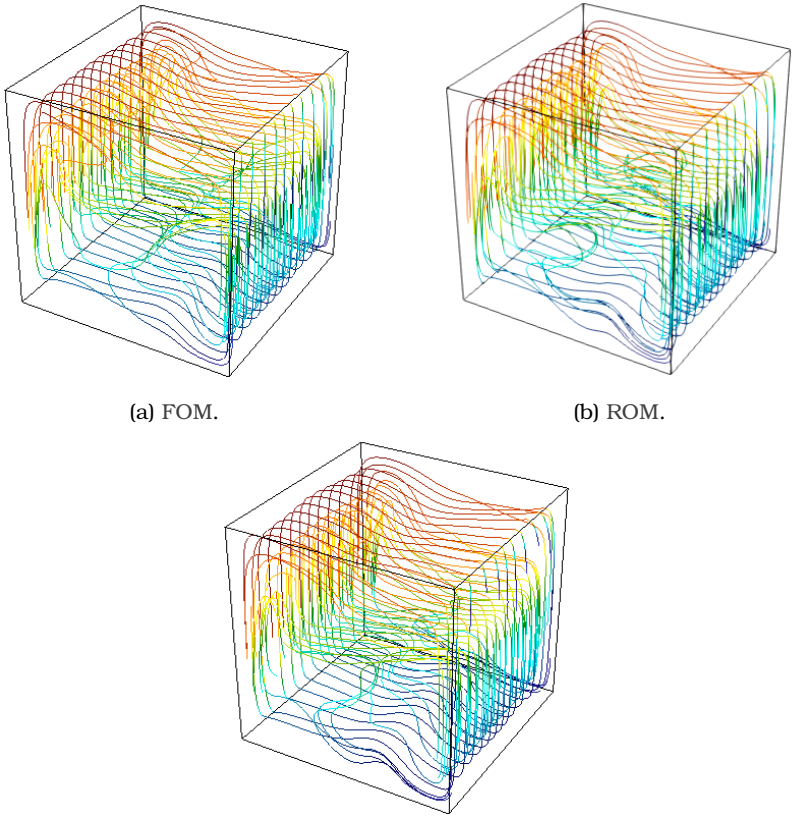
Figure 14.6: Differentially heated square cavity: computational time ratio t/t_0 of the ROM and hyper-ROM solutions (t) relative to the FOM solution (t_0).

14.1.2 *Three dimensional case*

For the 3D problem, we use 2 uniform structured meshes composed of regular hexahedral elements: one with 64000 elements and a mesh size $h = 0.025$, used for the solving the FOM and the ROM; and one with 35937 elements and a mesh size $h = 0.03$, for solving the hyper-ROM. To construct the basis, we collect 500 snapshots for velocity, pressure and temperature at every 2 time steps in a 10 seconds interval.

To test the 3D problem we perform similar tests as the ones in the 2D problem using 6 different sets of bases, varying the retained energy from $\eta_1 = 0.99$ to $\eta_6 = 0.5$. As in the 2D problem, we compute the Nusselt number, its RMSD and its discrete Fourier transform for

the hot and cold walls, getting similar results. Figure 14.7 shows a comparison of streamlines of the FOM, ROM and hyper-ROM solutions at the last time step of the simulation with $\eta = 0.8$ and $r = 13$.



(a) FOM.

(b) ROM.

(c) Hyper-ROM.

Figure 14.7: Differentially heated cubic cavity: streamlines at $t = 10$ for FOM, ROM and hyper-ROM with $\eta = 0.8$ and $r = 13$.

As in the previous example, we test the 3 cases using 6 different sets of basis, varying the retained energy η from 0.99 to 0.56. Figure 14.8 depicts the number of vectors in the basis as a function of the retained energy.

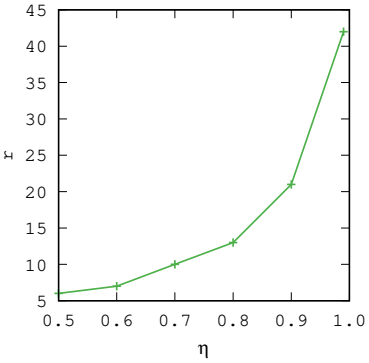


Figure 14.8: Differentially heated cubic cavity: number of basis vectors r as a function of retained energy η.

To compare the solutions we follow a similar approach as for the 2D case, we use the Nusselt number and the Root-Mean-Square (RMS) of the Nusselt number error defined in equations (14.3) and (14.4). Figure 14.9 shows the shows the convergence of ϵ_{RMS} for the ROM and the hyper-ROM over the hot and cold walls. The error convergence shows similar results as in the 2D case, including the *over-fitting* phenomena when the retained energy η approaches 1.

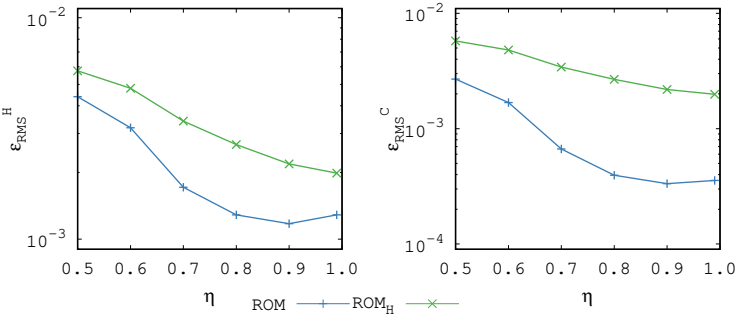
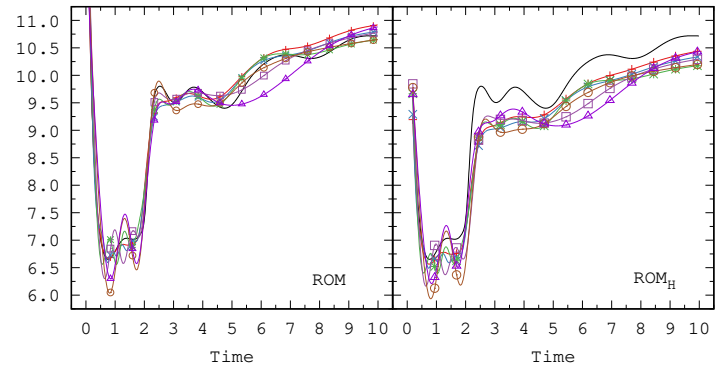
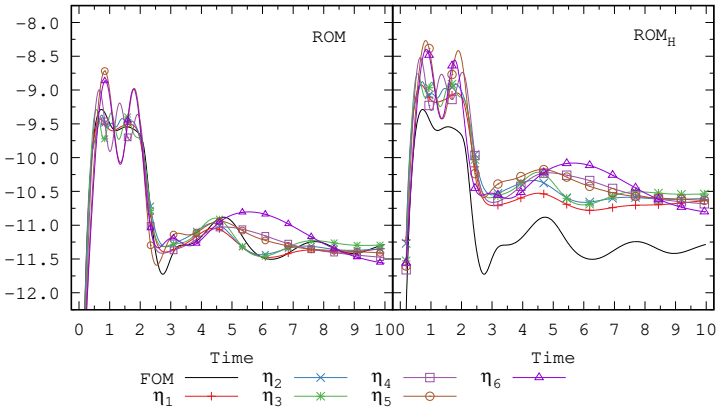


Figure 14.9: Differentially heated cubic cavity: ϵ_{RMS} convergence. Hot wall $\epsilon^{\text{H}}_{\text{RMS}}$ and cold wall $\epsilon^{\text{C}}_{\text{RMS}}$.

Figure 14.10 shows a comparison of the Nusselt number for FOM, ROM, and hyper-ROM. When compared to the Boussinesq approximation in section 13.2, we see an improvement in the quality of the approximation of the ROMs, specially considering that the physical properties and the mesh used are the same for both examples.



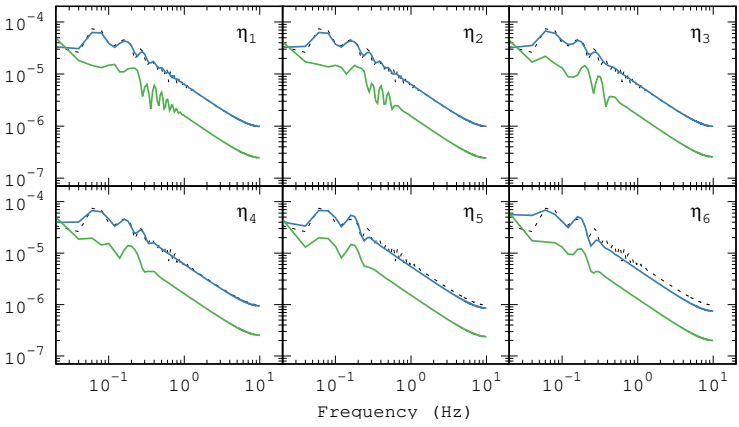
(a) Hot wall.



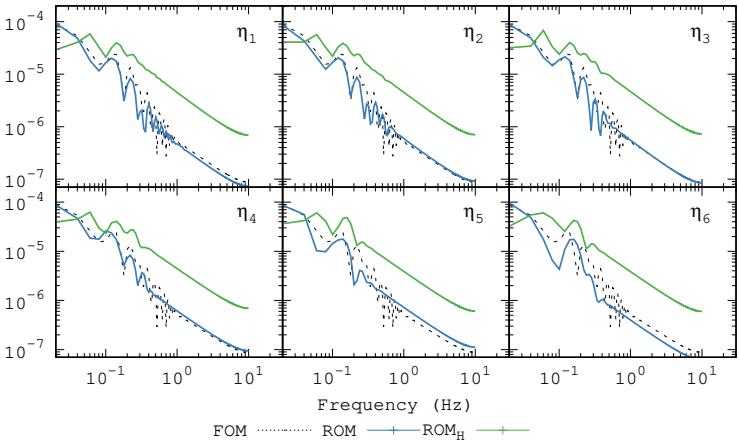
(b) Cold wall.

Figure 14.10: Differentially heated cubic cavity: Nusselt number evolution for FOM, ROM and hyper-ROM with a retained energy from $\eta_1 = 0.99$ to $\eta_6 = 0.56$.

In figure 14.11 we compare the Nusselt number spectra for the hot and cold walls. Although the results behave in a similar way as in the 2D case, the ROM and hyper-ROM seem to be more dissipative despite having a smaller mesh ratio. We believe this could be attributed to the quality of the basis, which directly depends on the quality of the FOM.



(a) Hot wall.



(b) Cold wall.

Figure 14.11: Differentially heated cubic cavity: Fourier transform of the Nusselt number.

Figure 14.12 shows the computational time ratio for ROM and hyper-ROM solutions. The reduction in the computational time for the 3D problem is larger than the one achieved for the 2D problem, using a similar amount of basis vectors.

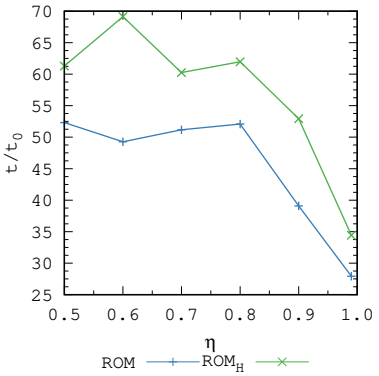


Figure 14.12: Differentially heated cubic cavity: computational time ratio t/t_0 of the ROM and hyper-ROM solutions (t) relative to the FOM solution (t_0).

15 APPLICATION PROBLEM: FLOW OVER A TWISTED RING

The formulation of this problem began looking for a colorful and characteristic 3D example which we could solve using our stabilized ROM formulation. Which means, having a fluctuating but not completely chaotic fluid flow. Thus, inspired by the von Kármán vortex street in section 12.1 and works in the particular properties of a Möbius strip —specifically a free fall [77]— we devise a numerical test consisting in a flow over a twisted ring.

The twisted ring is constructed from a square right cylinder, with the square length $l = 0.001$ and the cylinder twisted 180° over its axis. The ring has a mean radius of 0.25. Figure 15.1 illustrates the twisted ring.

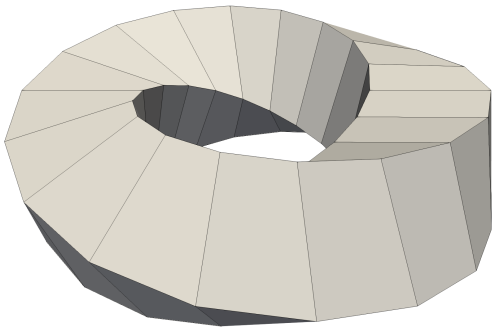


Figure 15.1: Flow over a twisted ring: twisted ring geometry.

The computational domain consists in a box $\Omega = [-2, 2] \times [0, 7.5] \times [-2, 2]$, with the twisted ring centered at $(0, 1, 0)$. The velocity at $y = 0$

is prescribed to $(0, 1, 0)$, the walls of the box $x = 2$, $x = -2$, $z = 2$ and $z = -2$ are set as symmetry boundary conditions, and the outflow at $y = 7.5$ is left free. he viscosity is prescribed to $\nu = 0.001$, resulting in a Reynolds number of 500. The domain is discretized using a mesh of 743810 tetrahedral elements.

To obtain the fully developed flow, we perform a preliminary simulation until $t = 100$, reset as $t = 0$. From that time, 400 equispaced solutions of velocity and pressure are used to calculate the ROM basis, with a time step of $\delta t = 0.05$.

To solve this example we use two different formulations: the incompressible Navier-Stokes equations described in chapter 12 and a three-field (velocity, pressure and stress) Navier-Stokes formulation similar to the one presented in [29] for FEs and used in [99] for solving a fluid structure interaction ROM.

15.1 THREE-FIELD INCOMPRESSIBLE NAVIER-STOKES

The problem consists of finding a velocity $\mathbf{u} : \Omega \times]0, t_f[\to \mathbb{R}^d$, a pressure $p : \Omega \times]0, t_f[\to \mathbb{R}$ and the deviatoric component of a stress $\mathbf{s} : \Omega \times]0, t_f[\to \mathbb{R}^{d \times d}$ such that

$$\begin{aligned} \rho \partial_t \mathbf{u} + \rho \mathbf{u} \cdot \nabla \mathbf{u} - \nabla \cdot \mathbf{s} + \nabla p = \mathbf{f} \qquad & \text{in } \Omega, \quad t \in]0, t_f[, \\ \nabla \cdot \mathbf{u} = 0 \qquad & \text{in } \Omega, \quad t \in]0, t_f[, \\ \frac{1}{2\mu} \mathbf{s} - \nabla^s \mathbf{u} = 0 \qquad & \text{in } \Omega, \quad t \in]0, t_f[, \end{aligned}$$

where μ denotes the kinematic viscosity, ρ the density, $\mathbf{f}$ a vector of body forces and $\nabla^s \mathbf{u}$ the symmetrical part of $\nabla \mathbf{u}$. Initial and boundary conditions are set for Γ as

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

We construct the functional spaces for this model using the notation in chapter 12 and following [39] —where a three-field formulation is proposed for a Stokes problem— and [29] —where the three-field formulation is used in a viscoelastic incompressible Navier-Stokes problem. Let $\mathcal{V}_0 = (H_0^1(\Omega))^d$, $\mathcal{Q}_0 = L^2(\Omega)/\mathbb{R}$ and $\mathcal{S}_0 = (H^1(\Omega))^{d\times d}_{sym}$, and define $\mathcal{V} = L^2(0,t_f;\mathcal{V}_0)$, $\mathcal{Q} = L^1(0,t_f;\mathcal{Q}_0)$ and $\mathcal{S} = L^1(0,t_f;\mathcal{S}_0)$. This way, the variational form of the problem is defined in the spaces $\mathcal{Y} = \mathcal{V}\times\mathcal{Q}\times\mathcal{S}$ for the trial solutions $-\mathbf{y} = [\mathbf{u},p,\mathbf{s}]-$ and $\mathcal{Y}_0 = \mathcal{V}_0\times\mathcal{Q}_0\times\mathcal{S}_0$ for the test functions $-\mathbf{v} = [\mathbf{v},q,\boldsymbol{\sigma}]$.

As for the standard incompressible Navier-Stokes (chapter 12), we also write the terms that correspond to the abstract ones using dynamic OSGS and assuming the force term to belong to the FE space, as

$$\boldsymbol{\mathcal{M}}(\mathbf{y}) = \begin{bmatrix} \rho\mathbf{I}^d & \mathbf{0} & \mathbf{0}^d\otimes\mathbf{0}^{d\times d}_{sym} \\ (\mathbf{0}^d)^\top & 0 & (\mathbf{0}^{d\times d}_{sym})^\top \\ \mathbf{0}^{d\times d}_{sym}\otimes\mathbf{0}^d & \mathbf{0} & \mathbf{0}^{d\times d}_{sym}\otimes\mathbf{0}^{d\times d}_{sym} \end{bmatrix},$$

$$\boldsymbol{\mathcal{L}}(\mathbf{y};\mathbf{y}_r) = \begin{bmatrix} \rho\mathbf{u}\cdot\nabla\mathbf{u}_r + \nabla p_r - \nabla\mathbf{s}_r \\ \nabla\cdot\mathbf{u}_r \\ \frac{1}{2\mu}\mathbf{s}_r - \nabla^s\mathbf{u}_r \end{bmatrix},$$

$$\boldsymbol{\mathcal{L}}^*(\mathbf{y};\mathbf{v}_r) = \begin{bmatrix} -\rho\mathbf{u}\cdot\nabla\mathbf{v}_r - \nabla q_r + \nabla\boldsymbol{\sigma}_r \\ -\nabla\cdot\mathbf{v}_r \\ \frac{1}{2\mu}\boldsymbol{\sigma}_r + \nabla^s\mathbf{v}_r \end{bmatrix},$$

$$\boldsymbol{\mathcal{F}}(\mathbf{y};\mathbf{y}_r) = \begin{bmatrix} \mathbf{n}\cdot p_r\mathbf{I} - \mathbf{n}\cdot\mu\nabla^s\mathbf{u}_r \\ \mathbf{0} \\ \mathbf{0}^{d\times d}_{sym} \end{bmatrix}, \quad \boldsymbol{\mathcal{F}}^*(\mathbf{y};\mathbf{y}_r) = \begin{bmatrix} \mathbf{n}\cdot q_r\mathbf{I} + \mathbf{n}\cdot\nu\nabla^s\mathbf{v}_r \\ \mathbf{0} \\ \mathbf{0}^{d\times d}_{sym} \end{bmatrix},$$

with $\mathbf{I}^d$, $\mathbf{0}^d$ and $\mathbf{0}^{d\times d}_{sym}$ of appropriate dimensions. With these term defined, we can write the forms

$$\begin{aligned} B(\mathbf{y};\mathbf{y}_r,\mathbf{v}_r) = {} & \langle\rho\mathbf{u}\cdot\nabla\mathbf{u}_r,\mathbf{v}_r\rangle + (\mathbf{s}_r,\nabla^s\mathbf{v}_r) - (p_r\mathbf{I},\nabla\mathbf{v}_r) \\ & + (\nabla\cdot\mathbf{u}_r,q_r) + (\frac{1}{2\mu}\mathbf{s},\boldsymbol{\sigma}_r) - (\nabla^s\mathbf{u},\boldsymbol{\sigma}_r), \end{aligned} \tag{15.1a}$$

$$
\begin{aligned}
B_s(\mathbf{y};\mathbf{y}_r,\mathbf{v}_r) = &-\sum_K \langle \check{\boldsymbol{\Pi}}(\rho\partial_t \mathbf{u}), \tau_{1,t}(\rho\mathbf{u}\cdot\nabla\boldsymbol{\nu}_r + \nabla q_r - \nabla\cdot\boldsymbol{\sigma}_r)\rangle_K \\
&- \sum_K \langle \check{\boldsymbol{\Pi}}(\rho\mathbf{u}\cdot\nabla\mathbf{u}_r), \tau_{1,t}(\rho\mathbf{u}\cdot\nabla\boldsymbol{\nu}_r + \nabla q_r - \nabla\cdot\boldsymbol{\sigma}_r)\rangle_K \\
&- \sum_K \langle \check{\boldsymbol{\Pi}}(\nabla p_r - \nabla\cdot\mathbf{s}), \tau_{1,t}(\rho\mathbf{u}\cdot\nabla\boldsymbol{\nu}_r + \nabla q_r - \nabla\cdot\boldsymbol{\sigma}_r)\rangle_K \\
&- \sum_K \langle \check{\boldsymbol{\Pi}}(\nabla^s\mathbf{u}_r - \frac{1}{2\mu}\mathbf{s}_r), \tau_3(\frac{1}{2\mu}\boldsymbol{\sigma}_r + \nabla^s\boldsymbol{\nu}_r)\rangle_K \\
&- \sum_K \langle \check{\boldsymbol{\Pi}}(\nabla\cdot\mathbf{u}_r), \tau_2\nabla\cdot\boldsymbol{\nu}_r\rangle_K, \qquad (15.1b)
\end{aligned}
$$

$$
L_s(\mathbf{y};\mathbf{v}_r) = -\sum_K \langle \check{\boldsymbol{\Pi}}(\mathbf{f}) + \delta t^{-1}\check{\mathbf{u}}, \tau_{1,t}(\rho\mathbf{u}\cdot\nabla\boldsymbol{\nu}_r - \nabla\cdot\boldsymbol{\sigma}_r + \nabla q_r)\rangle_K. \quad (15.1c)
$$

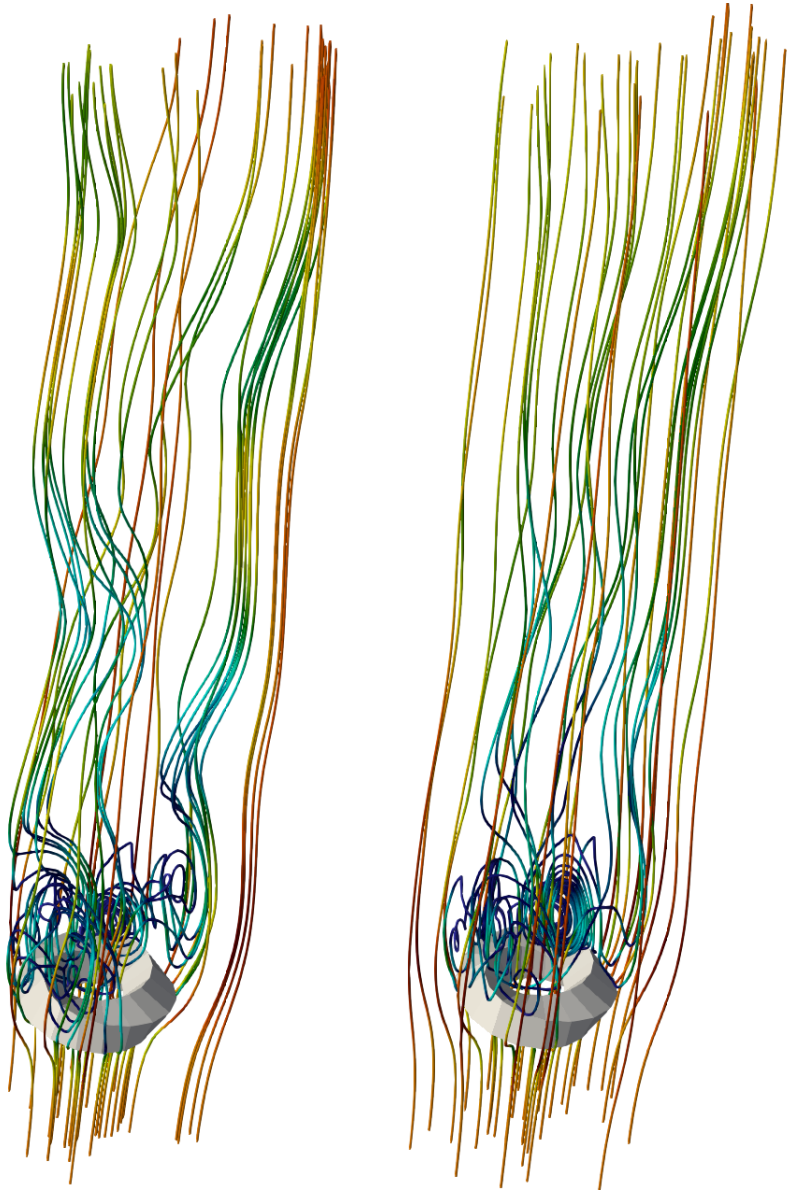
The stabilization parameter matrix is defined following [] as

$$
\boldsymbol{\tau}_K(\mathbf{y}) = \mathrm{diag}(\tau_1\mathbf{I},\tau_2) = \begin{bmatrix} (c_1\frac{\mu}{h^2} + c_2\frac{\rho|\mathbf{u}|_K}{h})^{-1}\mathbf{I}^d & \mathbf{0}^d & \mathbf{0}^d\otimes\mathbf{0}^{d\times d}_{sym} \\ (\mathbf{0}^d)^\top & c_3 2\mu & (\mathbf{0}^{d\times d}_{sym})^\top \\ \mathbf{0}^{d\times d}_{sym}\otimes\mathbf{0}^d & 0 & c_4 2\mu\mathbf{I}^{d\times d}_{sym} \end{bmatrix},
$$

in $K\in\mathcal{T}_h$, with $c_1 = 12$, $c_2 = 2$, $c_3 = c_4 = 0.25$ for linear elements and $\tau_{1,t}^{-1} = \delta t^{-1} + \tau_1^{-1}$. In the same way as in the standard incompressible Navier-Stokes equations, we use a BDF time integration scheme of second order for the resolved scales and of first order for the subscales, with $\check{\mathbf{u}}$ evaluated at t^j when solving for the unknowns evaluated at t^{j+1}.

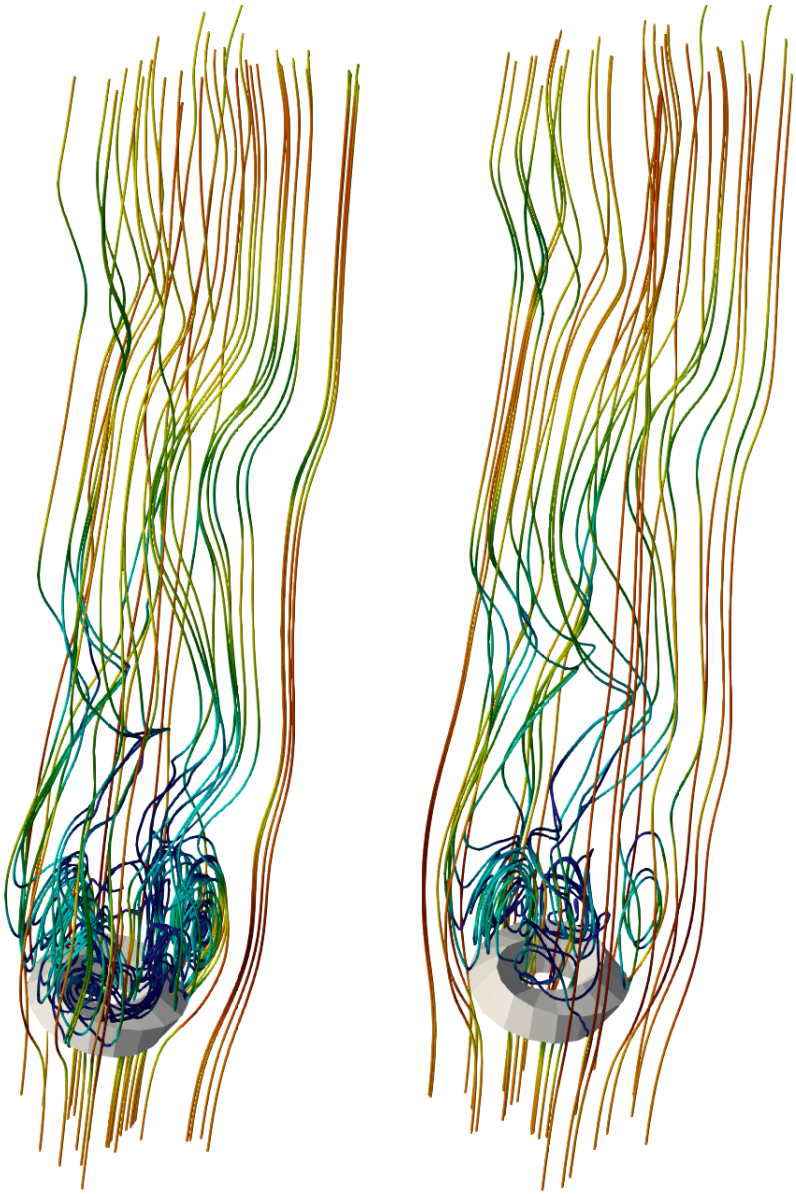
15.2 NUMERICAL RESULTS

To solve the ROM we use bases with the same retained energy $\eta = 0.95$ for both formulations, standard and three-field, albeit the difference in the amount of vectors for each, $r = 11$ for the standard and $r = 55$ for the three-field. We found that these sizes of the basis provide the best approximations for the ROMs. In figures 15.2 and 15.3 we show FOM and ROM streamlines at the last time step $t = 20$, for both the standard and the three-field formulations, respectively.



(a) FOM. (b) ROM.

Figure 15.2: Flow over a twisted ring: streamlines at $t = 20$ using the standard incompressible Navier-Stokes formulation.



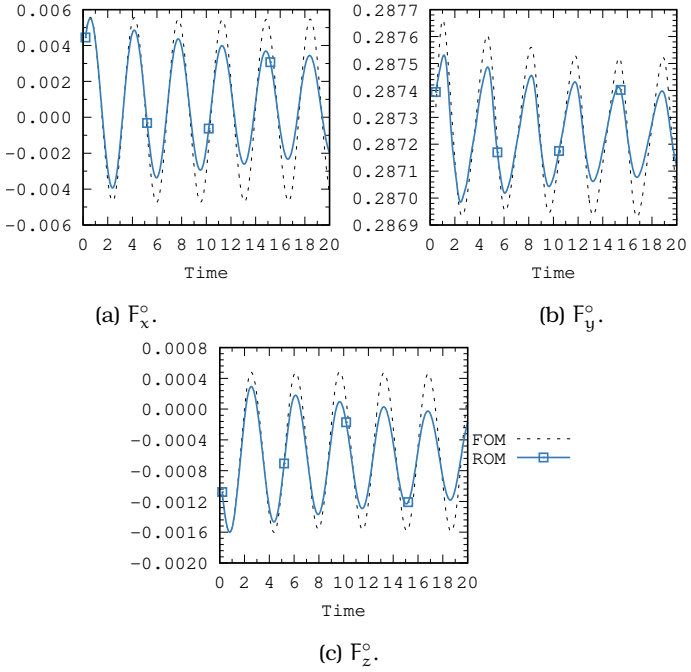
(a) FOM. (b) ROM.

Figure 15.3: Flow over a twisted ring: streamlines at $t = 20$ using the three-field incompressible Navier-Stokes formulation.

Similar to section 12.1, we compare in figures 15.4 and 15.5 the FOM and ROM forces over the ring for each direction ($\mathbf{F}^\circ = [F_x^\circ, F_y^\circ, F_z^\circ]$), for both the standard and the three-field formulations. With the force over the ring defined as

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

The ROMs for both cases are able to approximate adequately the behavior of the problem with the standard formulation being more diffusive over time. As in previous 3D examples, we believe that this behavior might be caused for lack of a good enough basis —one with enough information. Thus, better approximations of the FOM might be possible using more refine meshes for the FOM.

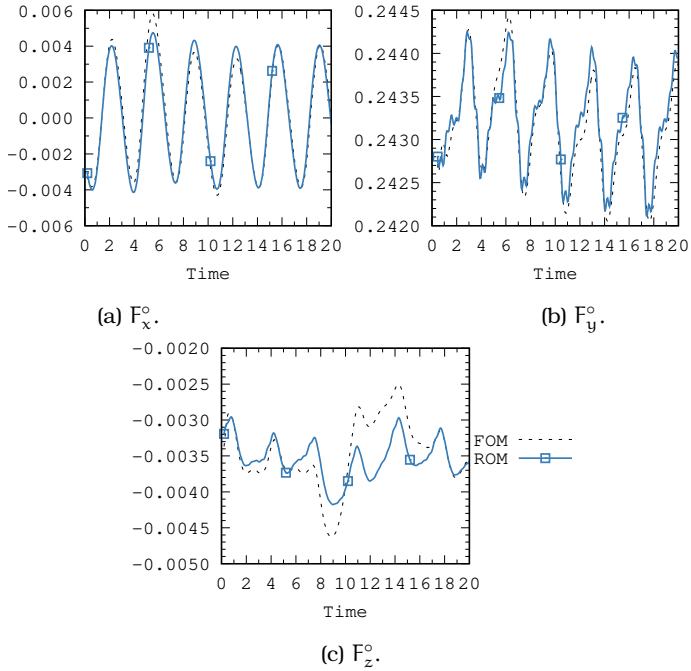


(a) F_x°.

(b) F_y°.

(c) F_z°.

Figure 15.4: Flow over a twisted ring: force over the ring ($\mathbf{F}^\circ$) for FOM and ROM using the standard incompressible Navier-Stokes formulation.

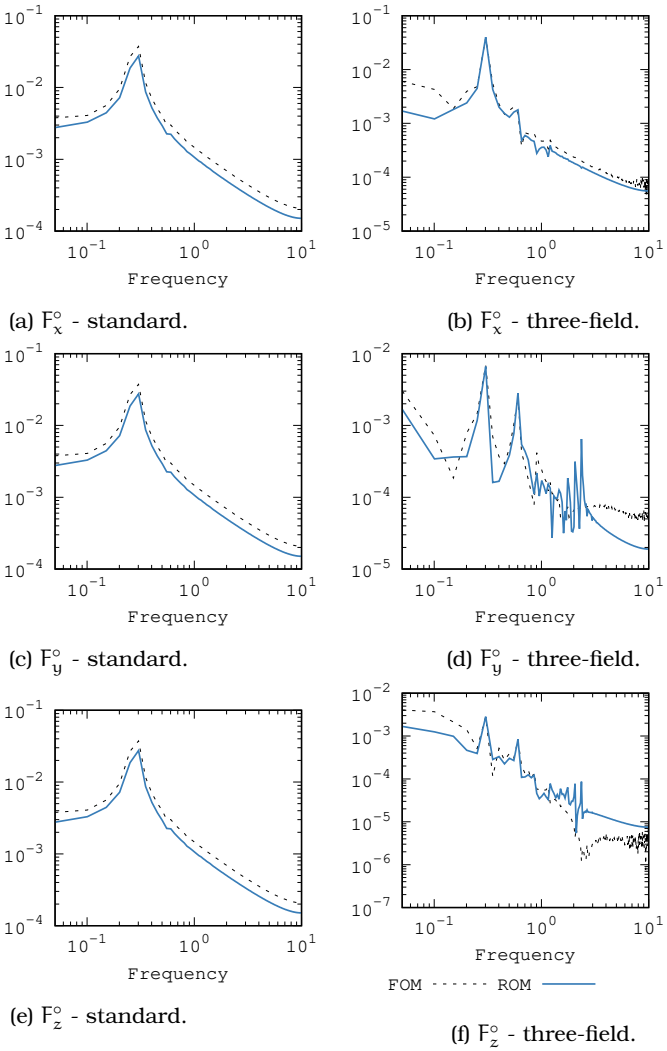


(a) F_x°. (b) F_y°.

(c) F_z°.

Figure 15.5: Flow over a twisted ring: force over the ring ($\mathbf{F}^\circ$) for FOM and ROM using the three-field incompressible Navier-Stokes formulation.

Figure 15.6 shows the comparison of the Fourier transform of each of the force components for the FOM and ROM solutions of the standard and the three-field formulations. We can see how in both cases, the ROM captures successfully the main modes of the solution. Furthermore, we infer that the less diffusive behavior in the three-field ROM might be related to the additional information that is present in the solution, specifically in y-component of the force (figures 15.6c and 15.6d), which could be provided by including the stress s in the solution.



(a) F_x° - standard.

(b) F_x° - three-field.

(c) F_y° - standard.

(d) F_y° - three-field.

(e) F_z° - standard.

(f) F_z° - three-field.

Figure 15.6: Flow over a twisted ring: Fourier transform of the force over the ring ($\mathbf{F}^\circ$) for FOM and ROMusing the standard (left) and the three-field (right) incompressible Navier-Stokes formulations.

SUMMARY

In this dissertation I have presented a stabilized projection-based Reduced Order Model (ROM) using a Variational Multi-Scale (VMS) approach, which we have tested solving fluid flow problems, in particular those that present a transitory unsteady but somewhat periodical behavior.

The first approach to develop an stabilized ROM formulation was based on what we perceived could be the contribution of the VMS to a projected problem onto a reduced space: the added stability from which is known in Finite Elements (FEs) and other mesh based formulations, and the improvement of the approximation achieved by adding the SubGrid Scale (SGS).

At this point we consider the problem from a different perspective: not as a projected system of equations but instead as a construction of the solution subspace on its own. From this point of view, although we keep using the reduced basis constructed with the standard Proper Orthogonal Decomposition (POD), we can relegate the basis construction and treat the problem as any mesh-based problem —as it were a Spectral Element (SE) method.

After implementing the fundamental characteristics of previously developed VMS-FE formulations onto our problem, we end with a robust and consistent ROM formulation that presents three main distinctive features: the orthogonality between resolved scales and the SGS, the time dependency of the SGS, and the use of the same stabilization parameters as for the Full Order Model (FOM) formulation. It is also important to note that an additional ingredient is needed when solving non-linear problems: the projection needs to be oblique (Petrov-Galerkin) to search for the solution in the appropriate subspace.

Additionally, by using our stabilized ROM formulation, we implement some techniques that are often used in FEs and other mesh-based formulations, resulting in:

- A mesh-based hyper-ROM using an online h-Adaptive Mesh Refinement (AMR), that modifies the mesh where the system is assemble. Here we use a error measure based on the SGS, for which we use not only the measure of the subscales at the interior of the elements but also on the boundaries. This technique improves the computational cost while retaining a similar quality of the solution compared to a ROM. The overall efficiency of this hyper-reduction method is heavily dependent of the efficiency of the refinement and interpolation algorithms.
- Domain decomposition techniques, where we use local bases, a Discontinuous Galerkin (DG) approach and an iterative-by-subdomain scheme. For this cases, the stabilized ROM behaves adequately regardless the use of a unified or a decomposed domain. Moreover, we found that when a singularity is present the use of the domain decomposition ROM approximates better than the one with a unified domain.

To test our formulation we use five different physical models:

1. A convection-diffusion-reaction problem, which we use as an illustrative example.
2. An incompressible Navier-Stokes problem, where we analyze exhaustively the consistency and robustness of the proposed formulation, including the mesh-based hyper-reduction and the domain decomposition methods.
3. A Boussinesq approximation of the Navier-Stokes equations, where we extend the formulation to thermally coupled flows, integrating the energy equation into the problem.

4. A zero Mach number limit of the compressible Navier-Stokes problem, where we test further some thermally coupled flows, solving similar problems as in the Boussinesq approximation.
5. A three-field incompressible Navier-Stokes approach with which we solve a more complex 3D flow.

We can summarize the most important features of the stabilization method and the additional techniques in:

- The use of orthogonal SGS shows an overall more consistent behavior over variations of η and h than the non-orthogonal SGS.
- 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.
- The use of the mesh based hyper-ROM show a convergence over the mesh size h that resembles the one for FE, which shows the consistency of the proposed VMS-ROM formulation.
- The use of AMR as an improved mesh-based hyper reduction technique, succeeds in decreasing the computational cost while maintaining the accuracy of the original ROM.
- The implementation of domain decomposition techniques used in other mesh based methods (e. g. FE) can be done straightforwardly, improving the solution around singularities.
- It outperforms other stabilization methods: the Algebraic Sub-Grid Scale (ASGS) is less stable over variations of η and h, and the term-by-term stabilization performs poorly, being over-diffusive at best.

OUTLOOK

The development of the VMS-ROM formulation and the additional techniques brought us some ideas that can be addressed in the future.

- Find an error bound for the VMS-ROM formulation proposed, based on the element size h, in a similar way as is done for FEs.
- Set a more relevant error tolerance target for the AMR hyper-ROM technique, based for example on an error bound for ROMs or in a measure defined for a particular physical phenomena.
- Implement the subscales error measure as an error indicator in an Adaptive Spectrum Refinement (ASR). (e. g. [24, 50] where a h-adaptive technique is formulated for the basis).
- Couple the proposed AMR hyper-reduction with an ASR technique, and extend them to develop a hybrid FOM-ROM online formulation to improve results in difficult areas.
- Use the domain decomposition techniques to implement further applications in different physical problems (e. g. fluid structure interaction [99, 100], fracture processes [47]) or different algorithmic schemes (e. g. optimization problems [2], FOM-ROM hybrid problems [22, 71, 78], parametric geometries [65, 66]).

PUBLICATIONS

As a result of the research done in this dissertation we submitted three articles to peer-reviewed scientific journals, in each of them a major part of the thesis is contained: the stabilized formulation described in part ii, the hyper-reduction and domain decomposition techniques described in chapters 9 and 10 and the application for the low Mach thermally coupled flows in chapter 14.

1. Reyes, R. & Codina, R. *Element boundary terms in reduced order models for flow problems: domain decomposition and adaptive coarse mesh hyper-reduction* 2019. Submitted.
2. Reyes, R. & Codina, R. *Projection-based reduced order models for flow problems: A variational multiscale approach* 2019. Accepted.
3. Reyes, R., Codina, R., Baiges, J. & Idelsohn, S. Reduced order models for thermally coupled low Mach flows. *Advanced Modeling and Simulation in Engineering Sciences* **5,** 28. ISSN: 2213-7467. doi:10.1186/s40323-018-0122-7 (2018).

BIBLIOGRAPHY

1. Amsallem, D., Zahr, M. J. & Farhat, C. Nonlinear model order reduction based on local reduced-order bases. *International Journal for Numerical Methods in Engineering* **92,** 891–916. ISSN: 0029-5981. doi:10.1002/nme.4371 (2012).

2. Antil, H., Heinkenschloss, M., Hoppe, R. H. & Sorensen, D. C. Domain decomposition and model reduction for the numerical solution of PDE constrained optimization problems with localized optimization variables. *Computing and Visualization in Science* **13,** 249–264. ISSN: 1432-9360. doi:10.1007/s00791-010-0142-4 (2010).

3. Aubry, N. On the hidden beauty of the proper orthogonal decomposition. *Theoretical and Computational Fluid Dynamics* **2,** 339–352. ISSN: 0935-4964. doi:10.1007/BF00271473 (1991).

4. Aubry, N., Guyonnet, R. & Lima, R. Spatio-temporal symmetries and bifurcations via bi-orthogonal decompositions. *Journal of Nonlinear Science* **2,** 183–215. ISSN: 0938-8974. doi:10.1007/BF02429855 (1992).

5. Avila, M. *Nonlinear subgrid finite element models for low Mach number flows coupled with radiative heat transfer*. PhD thesis. Universitat Politècnica de Catalunya, 2012. Available at: http://hdl.handle.net/2117/95614.

6. Avila, M., Codina, R. & Principe, J. Large eddy simulation of low Mach number flows using dynamic and orthogonal subgrid scales. *Computers and Fluids* **99,** 44–66. ISSN: 0045-7930. doi:10.1016/j.compfluid.2014.04.003 (2014).

7. Avila, M., Principe, J. & Codina, R. A finite element dynamical nonlinear subscale approximation for the low Mach number flow equations. *Journal of Computational Physics* **230,** 7988–8009. ISSN: 0021-9991. doi:10.1016/j.jcp.2011.06.032 (2011).

8. Azaïez, M., Chacón Rebollo, T. & Rubino, S. *Streamline derivative projection-based POD-ROM for convection-dominated flows. Part I : Numerical Analysis* arXiv: 1711.09780. 2017.

9. Badia, S. & Codina, R. On a multiscale approach to the transient Stokes problem: Dynamic subscales and anisotropic space–time discretization. *Applied Mathematics and Computation* **207,** 415–433. ISSN: 0096-3003. doi:10.1016/j.amc.2008.10.059 (2009).

10. Baiges, J. & Bayona, C. Refficientlib: An Efficient Load-Rebalanced Adaptive Mesh Refinement Algorithm for High-Performance Computational Physics Meshes. *SIAM Journal on Scientific Computing* **39,** C65–C95. ISSN: 1095-7197. doi:10.1137/15M105330X (2017).

11. Baiges, J. & Codina, R. Variational Multiscale error estimators for solid mechanics adaptive simulations: An Orthogonal Subgrid Scale approach. *Computer Methods in Applied Mechanics and Engineering* **325,** 37–55. ISSN: 0045-7825. doi:10.1016/j.cma.2017.07.008 (2017).

12. Baiges, J., Codina, R. & Idelsohn, S. A domain decomposition strategy for reduced order models. Application to the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering* **267,** 23–42. ISSN: 0045-7825. doi:10.1016/j.cma.2013.08.001 (2013).

13. Baiges, J., Codina, R. & Idelsohn, S. Explicit reduced-order models for the stabilized finite element approximation of the incompressible Navier-Stokes equations. *International Journal*

for Numerical Methods in Fluids **72,** 1219–1243. ISSN: 0271-2091. doi:10.1002/fld.3777 (2013).

14. Baiges, J., Codina, R. & Idelsohn, S. Reduced-order subscales for POD models. *Computer Methods in Applied Mechanics and Engineering* **291,** 173–196. ISSN: 0045-7825. doi:10.1016/j.cma.2015.03.020 (2015).

15. Baiges, J., Codina, R. & Idelsohn, S. R. *Reduced-Order Modelling Strategies for the Finite Element Approximation of the Incompressible Navier-Stokes Equations* in *Numerical Simulations of Coupled Problems in Engineering, Computational Methods in Applied Sciences* (ed Idelsohn, S. R.) 189–216 (Springer, Cham, 2014). doi:10.1007/978-3-319-06136-8_9.

16. Balajewicz, M. & Dowell, E. H. Stabilization of projection-based reduced order models of the Navier–Stokes. *Nonlinear Dynamics* **70,** 1619–1632. ISSN: 0924-090X. doi:10.1007/s11071-012-0561-5 (2012).

17. Ballarin, F. *et al. Certified Reduced Basis VMS-Smagorinsky model for natural convection flow in a cavity with variable height* arXiv: 1902.05729. 2019.

18. Barrault, M., Maday, Y., Nguyen, N. C. & Patera, A. T. An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique* **339,** 667–672. ISSN: 1631-073X. doi:10.1016/j.crma.2004.08.006 (2004).

19. Bayona, C., Codina, R. & Baiges, J. Variational multiscale error estimators for the adaptive mesh refinement of compressible flow simulations. *Computer Methods in Applied Mechanics and Engineering* **337,** 501–526. ISSN: 0045-7825. doi:10.1016/j.cma.2018.03.010 (2018).

20. Benner, P., Gugercin, S. & Willcox, K. A Survey of Projection-Based Model Reduction Methods for Parametric Dynamical Systems. *SIAM Review* **57,** 483–531. ISSN: 0036-1445. doi:10.1137/130932715 (2015).

21. Bergmann, M., Bruneau, C. H. & Iollo, A. Enablers for robust POD models. *Journal of Computational Physics* **228,** 516–538. ISSN: 0021-9991. doi:10.1016/j.jcp.2008.09.024 (2009).

22. Buffoni, M., Telib, H. & Iollo, A. Iterative methods for model reduction by domain decomposition. *Computers and Fluids* **38,** 1160–1167. ISSN: 0045-7930. doi:10.1016/j.compfluid.2008.11.008 (2009).

23. Burman, E., Fernández, M. A. & Hansbo, P. Continuous Interior Penalty Finite Element Method for Oseen's Equations. *SIAM Journal on Numerical Analysis* **44,** 1248–1274. ISSN: 0036-1429. doi:10.1137/040617686 (2006).

24. Carlberg, K. Adaptive h -refinement for reduced-order models. *International Journal for Numerical Methods in Engineering* **102,** 1192–1210. ISSN: 0029-5981. doi:10.1002/nme.4800 (2015).

25. Carlberg, K., Barone, M. & Antil, H. Galerkin v. least-squares Petrov–Galerkin projection in nonlinear model reduction. *Journal of Computational Physics* **330,** 693–734. ISSN: 0021-9991. doi:10.1016/j.jcp.2016.10.033 (2017).

26. Carlberg, K., Bou-Mosleh, C. & Farhat, C. Efficient non-linear model reduction via a least-squares Petrov-Galerkin projection and compressive tensor approximations. *International Journal for Numerical Methods in Engineering* **86,** 155–181. ISSN: 0029-5981. doi:10.1002/nme.3050 (2011).

27. Castillo, E. *Stabilized finite element formulations for the three-Field viscoelastic fluid flow problem* PhD thesis. Universitat

Politècnica de Catalunya, 2016. Available at: http://hdl.handle.net/2117/96306.

28. Castillo, E. & Codina, R. Dynamic term-by-term stabilized finite element formulation using orthogonal subgrid-scales for the incompressible Navier–Stokes problem. *Computer Methods in Applied Mechanics and Engineering* **349,** 701–721. ISSN: 0045-7825. doi:10.1016/j.cma.2019.02.041 (2019).

29. Castillo, E. & Codina, R. Variational multi-scale stabilized formulations for the stationary three-field incompressible viscoelastic flow problem. *Computer Methods in Applied Mechanics and Engineering* **279,** 579–605. ISSN: 0045-7825. doi:10.1016/j.cma.2014.07.006 (2014).

30. Chacón Rebollo, T. *et al.* On a certified smagorinsky reduced basis turbulence model. *SIAM Journal on Numerical Analysis* **55,** 3047–3067. ISSN: 0036-1429. doi:10.1137/17M1118233 (2017).

31. Chinesta, F., Ammar, A., Leygue, A. & Keunings, R. An overview of the proper generalized decomposition with applications in computational rheology. *Journal of Non-Newtonian Fluid Mechanics* **166,** 578–592. ISSN: 0377-0257. doi:10.1016/j.jnnfm.2010.12.012 (2011).

32. Chinesta, F., Huerta, A., Rozza, G. & Willcox, K. *Model Reduction Methods* in *Encyclopedia of Computational Mechanics Second Edition* (eds E. Stein, R. B. & Hughes, T.) 1–36 (John Wiley & Sons, Ltd, Chichester, UK, 2017). ISBN: 9781119003793. doi:10.1002/9781119176817.ecm2110.

33. Choi, Y. & Carlberg, K. Space–Time Least-Squares Petrov–Galerkin Projection for Nonlinear Model Reduction. *SIAM Journal on Scientific Computing* **41,** A26–A58. ISSN: 1064-8275. doi:10.1137/17M1120531 (2019).

34. Christon, M. a., Gresho, P. M. & Sutton, S. B. Computational predictability of time-dependent natural convection flows in enclosures (including a benchmark solution). *International Journal for Numerical Methods in Fluids* **40,** 953–980. ISSN: 0271-2091. doi:10.1002/fld.395 (2002).

35. Codina, R. Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods. *Computer Methods in Applied Mechanics and Engineering* **190,** 1579–1599. ISSN: 0045-7825. doi:10.1016/S0045-7825(00)00254-1 (2000).

36. Codina, R. A stabilized finite element method for generalized stationary incompressible flows. *Computer Methods in Applied Mechanics and Engineering* **190,** 2681–2706. ISSN: 0045-7825. doi:10.1016/S0045-7825(00)00260-7 (2001).

37. Codina, R. Stabilized finite element approximation of transient incompressible flows using orthogonal subscales. *Computer Methods in Applied Mechanics and Engineering* **191,** 4295–4321. ISSN: 0045-7825. doi:10.1016/S0045-7825(02)00337-7 (2002).

38. Codina, R. Analysis of a stabilized finite element approximation of the Oseen equations using orthogonal subscales. *Applied Numerical Mathematics* **58,** 264–283. ISSN: 0168-9274. doi:10.1016/j.apnum.2006.11.011 (2008).

39. Codina, R. Finite Element Approximation of the Three-Field Formulation of the Stokes Problem Using Arbitrary Interpolations. *SIAM Journal on Numerical Analysis* **47,** 699–718. ISSN: 0036-1429. doi:10.1137/080712726 (2009).

40. Codina, R. & Badia, S. On the design of discontinuous Galerkin methods for elliptic problems based on hybrid formulations. *Computer Methods in Applied Mechanics and Engineering* **263,**

158–168. ISSN: 0045-7825. doi:10.1016/j.cma.2013.05.004 (2013).

41. Codina, R., Badia, S., Baiges, J. & Principe, J. *Variational Multiscale Methods in Computational Fluid Dynamics* in *Encyclopedia of Computational Mechanics Second Edition* (eds E. Stein, R. B. & Hughes, T.) 1–28 (John Wiley & Sons, Ltd, Chichester, UK, 2018). ISBN: 9781119176817. doi:10.1002/9781119176817.ecm2117.

42. Codina, R. & Baiges, J. Finite element approximation of transmission conditions in fluids and solids introducing boundary subgrid scales. *International Journal for Numerical Methods in Engineering* **87,** 386–411. ISSN: 0029-5981. doi:10.1002/nme.3111 (2011).

43. Codina, R. & Principe, J. Dynamic subscales in the finite element approximation of thermally coupled incompressible flows. *International Journal for Numerical Methods in Fluids* **54,** 707–730. ISSN: 0271-2091. doi:10.1002/fld.1481 (2007).

44. Codina, R., Principe, J. & Avila, M. Finite element approximation of turbulent thermally coupled incompressible flows with numerical sub-grid scale modelling. *International Journal of Numerical Methods for Heat & Fluid Flow* **20,** 492–516. ISSN: 0961-5539. doi:10.1108/09615531011048213 (2010).

45. Codina, R., Principe, J. & Baiges, J. Subscales on the element boundaries in the variational two-scale finite element method. *Computer Methods in Applied Mechanics and Engineering* **198,** 838–852. ISSN: 0045-7825. doi:10.1016/j.cma.2008.10.020 (2009).

46. Codina, R., Principe, J., Guasch, O. & Badia, S. Time dependent subscales in the stabilized finite element approximation of incompressible flow problems. *Computer Methods in Applied*

Mechanics and Engineering **196,** 2413–2430. ISSN: 0045-7825. doi:10.1016/j.cma.2007.01.002 (2007).

47. Corigliano, A., Confalonieri, F., Dossi, M. & Mariani, S. Coupled domain decomposition–proper orthogonal decomposition methods for the simulation of quasi-brittle fracture processes. *Advanced Modeling and Simulation in Engineering Sciences* **3.** ISSN: 2213-7467. doi:10.1186/s40323-016-0081-9 (2016).

48. Eftang, J. L., Patera, A. T. & Rønquist, E. M. An hp Certified Reduced Basis Method for Parametrized Elliptic Partial Differential Equations. *SIAM Journal on Scientific Computing* **32,** 3170–3200. ISSN: 1064-8275. doi:10.1137/090780122 (2010).

49. El Alaoui, L. & Ern, A. Residual and hierarchical a posteriori error estimates for nonconforming mixed finite element methods. *ESAIM: Mathematical Modelling and Numerical Analysis* **38,** 903–929. ISSN: 0764-583X. doi:10.1051/m2an:2004044 (2004).

50. Etter, P. A. & Carlberg, K. T. *Online adaptive basis refinement and compression for reduced-order models via vector-space sieving* arXiv: 1902.10659. 2019.

51. Everson, R. & Sirovich, L. Karhunen–Loève procedure for gappy data. *Journal of the Optical Society of America A* **12,** 1657. ISSN: 1084-7529. doi:10.1364/JOSAA.12.001657 (1995).

52. Fahl, M. Computation of POD basis functions for fluid flows with lanczos methods. *Mathematical and Computer Modelling* **34,** 91–107. ISSN: 0895-7177. doi:10.1016/S0895-7177(01)00051-6 (2001).

53. Fahl, M. *Trust-region Methods for Flow Control based on Reduced Order Modelling* PhD thesis. Universitat Trier, 2001. Available at: https://ubt.opus.hbz-nrw.de/frontdoor/index/index/year/2004/docId/30.

54. Ganesan, S. & Tobiska, L. Stabilization by Local Projection for Convection–Diffusion and Incompressible Flow Problems. *Journal of Scientific Computing* **43,** 326–342. ISSN: 0885-7474. doi:10.1007/s10915-008-9259-8 (2010).

55. Giere, S. *Numerical and Analytical Aspects of POD-Based Reduced-Order Modeling in Computational Fluid Dynamics* PhD thesis. Freien Universität Berlin, 2016. Available at: https://refubium.fu-berlin.de/handle/fub188/13554.

56. Giere, S., Iliescu, T., John, V. & Wells, D. SUPG reduced order models for convection-dominated convection-diffusion-reaction equations. *Computer Methods in Applied Mechanics and Engineering* **289,** 454–474. ISSN: 0045-7825. doi:10.1016/j.cma.2015.01.020 (2015).

57. Guasch, O. & Codina, R. Statistical behavior of the orthogonal subgrid scale stabilization terms in the finite element large eddy simulation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering* **261-262,** 154–166. doi:10.1016/j.cma.2013.04.006 (2013).

58. Haasdonk, B., Dihlmann, M. & Ohlberger, M. A training set and multiple bases generation approach for parameterized model reduction based on adaptive grids in parameter space. *Mathematical and Computer Modelling of Dynamical Systems* **17,** 423–442. ISSN: 1387-3954. doi:10.1080/13873954.2011.547674 (2011).

59. Hale, J. S. *et al. A hyper-reduction method using adaptivity to cut the assembly costs of reduced order models* 2019http://hdl.handle.net/10993/36557.

60. Hauke, G., Doweidar, M. H. & Miana, M. The multiscale approach to error estimation and adaptivity. *Computer Methods*

in Applied Mechanics and Engineering **195,** 1573–1593. ISSN: 0045-7825. doi:10.1016/j.cma.2005.05.029 (2006).

61. Hauke, G., Fuster, D. & Lizarraga, F. Variational multiscale a posteriori error estimation for systems: The Euler and Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering* **283,** 1493–1524. ISSN: 0045-7825. doi:10.1016/j.cma.2014.10.032 (2015).

62. Hernández, J. A., Caicedo, M. A. & Ferrer, A. Dimensional hyper-reduction of nonlinear finite element models via empirical cubature. *Computer Methods in Applied Mechanics and Engineering* **313,** 687–722. ISSN: 0045-7825. doi:10.1016/j.cma.2016.10.022 (2017).

63. Hughes, T. J. Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods. *Computer Methods in Applied Mechanics and Engineering* **127,** 387–401. ISSN: 0045-7825. doi:10.1016/0045-7825(95)00844-9 (1995).

64. Hughes, T. J., Feijóo, G. R., Mazzei, L. & Quincy, J.-B. The variational multiscale method—a paradigm for computational mechanics. *Computer Methods in Applied Mechanics and Engineering* **166,** 3–24. ISSN: 0045-7825. doi:10.1016/S0045-7825(98)00079-6 (1998).

65. Iapichino, L., Quarteroni, A. & Rozza, G. A reduced basis hybrid method for the coupling of parametrized domains represented by fluidic networks. *Computer Methods in Applied Mechanics and Engineering* **221-222,** 63–82. ISSN: 0045-7825. doi:10.1016/j.cma.2012.02.005 (2012).

66. Iapichino, L., Quarteroni, A. & Rozza, G. Reduced basis method and domain decomposition for elliptic problems in networks and complex parametrized geometries. *Computers and Mathematics*

with Applications **71,** 408–430. ISSN: 0898-1221. doi:10.1016/j.camwa.2015.12.001 (2016).

67. Iliescu, T. & Wang, Z. Variational multiscale proper orthogonal decomposition: Convection-dominated convection-diffusion-reaction equations. *Mathematics of Computation* **82,** 1357–1378. ISSN: 0025-5718. doi:10.1090/S0025-5718-2013-02683-X (2013).

68. Iliescu, T. & Wang, Z. Variational multiscale proper orthogonal decomposition: Navier-stokes equations. *Numerical Methods for Partial Differential Equations* **30,** 641–663. ISSN: 0749-159X. doi:10.1002/num.21835 (2014).

69. Jovic, S. *An Experimental Study of a Separated/Reattached Flow Behind a Backward-Facing Step. Re=37000* tech. rep. April. National Aeronautics and Space Administration, Ames Research Center, 1996. https://ntrs.nasa.gov/search.jsp?R=19960047497.

70. Kalashnikova, I., van Bloemen Waanders, B., Arunajatesan, S. & Barone, M. F. Stabilization of projection-based reduced order models for linear time-invariant systems via optimization-based eigenvalue reassignment. *Computer Methods in Applied Mechanics and Engineering* **272,** 251–270. ISSN: 0045-7825. doi:10.1016/j.cma.2014.01.011 (2014).

71. Kerfriden, P., Goury, O., Rabczuk, T. & Bordas, S. P. A partitioned model order reduction approach to rationalise computational expenses in nonlinear fracture mechanics. *Computer Methods in Applied Mechanics and Engineering* **256,** 169–188. ISSN: 0045-7825. doi:10.1016/j.cma.2012.12.004 (2013).

72. Knobloch, P. & Lube, G. Local projection stabilization for advection-diffusion-reaction problems: One-level vs. two-level

approach. *Applied Numerical Mathematics* **59,** 2891–2907. ISSN: 0168-9274. doi:10.1016/j.apnum.2009.06.004 (2009).

73. Kragel, B. *Streamline Diffusion POD Models in Optimization*. PhD thesis. University of Trier, 2005. Available at: https://ubt.opus.hbz-nrw.de/frontdoor/index/index/docId/173.

74. Kunisch, K. & Volkwein, S. Galerkin Proper Orthogonal Decomposition Methods for a General Equation in Fluid Dynamics. *SIAM Journal on Numerical Analysis* **40,** 492–515. ISSN: 0036-1429. doi:10.1137/S0036142900382612 (2002).

75. Kunisch, K. & Volkwein, S. Galerkin proper orthogonal decomposition methods for parabolic problems. *Numer. Math.* **90,** 117–148. ISSN: 0029-599X. doi:10.1007/s002110100282 (2001).

76. Le Quéré, P. *et al.* Modelling of Natural Convection Flows with Large Temperature Differences: A Benchmark Problem for Low Mach Number Solvers. Part 1. Reference Solutions. *ESAIM: Mathematical Modelling and Numerical Analysis* **39,** 609–616. ISSN: 0764-583X. doi:10.1051/m2an:2005027 (2005).

77. Leweke, T., Thompson, M. C. & Hourigan, K. Motion of a Möbius band in free fall. *Journal of Fluids and Structures* **25,** 687–696. ISSN: 1095-8622. doi:10.1016/j.jfluidstructs.2009.04.007 (2009).

78. Lucia, D. J. & Beran, P. S. Projection methods for reduced order models of compressible flows. *Journal of Computational Physics* **188,** 252–280. ISSN: 0021-9991. doi:10.1016/S0021-9991(03)00166-9 (2003).

79. Lucia, D. J., Beran, P. S. & Silva, W. A. Reduced-order modeling: new approaches for computational physics. *Progress in Aerospace Sciences* **40,** 51–117. ISSN: 0376-0421. doi:10.1016/j.paerosci.2003.12.001 (2004).

80. Lumley, J. L. *Coherent Structures in Turbulence* in *Transition and Turbulence* (ed Meyer, R. E.) (Academic Press, 1981), 215–242. ISBN: 978-0-12-493240-1. doi:10.1016/B978-0-12-493240-1.50017-X.

81. Marras, S., Kelly, J., Giraldo, F. & Vázquez, M. Variational multiscale stabilization of high-order spectral elements for the advection–diffusion equation. *Journal of Computational Physics* **231,** 7187–7213. ISSN: 0021-9991. doi:10.1016/j.jcp.2012.06.028 (2012).

82. Martini, I., Rozza, G. & Haasdonk, B. Reduced basis approximation and a-posteriori error estimation for the coupled Stokes-Darcy system. *Advances in Computational Mathematics* **41,** 1131–1157. ISSN: 1572-9044. doi:10.1007/s10444-014-9396-6 (2015).

83. McLaughlin, B., Peterson, J. & Ye, M. Stabilized reduced order models for the advection–diffusion–reaction equation using operator splitting. *Computers & Mathematics with Applications* **71,** 2407–2420. ISSN: 0898-1221. doi:10.1016/j.camwa.2016.01.032 (2016).

84. Müller, B. Low Mach number asymptotics of the Navier-Stokes equations and numerical implications. *Journal of Engineering Mathematics* **34,** 97–109. doi:10.1023/A:1004349817404 (1999).

85. Nguyen, N. C., Patera, A. T. & Peraire, J. A 'best points' interpolation method for efficient approximation of parametrized functions. *International Journal for Numerical Methods in Engineering* **73,** 521–543. ISSN: 0029-5981. doi:10.1002/nme.2086 (2008).

86. Ohlberger, M. & Schindler, F. Error Control for the Localized Reduced Basis Multiscale Method with Adaptive On-Line Enrich-

ment. *SIAM Journal on Scientific Computing* **37,** A2865–A2895. ISSN: 1064-8275. doi:10.1137/151003660 (2015).

87. Principe, J. *Subgrid scale stabilized finite elements for low speed flows*. PhD thesis. Universitat Politècnica de Catalunya, 2008. Available at: http://hdl.handle.net/2117/94181.

88. Principe, J. & Codina, R. A stabilized finite element approximation of low speed thermally coupled flows. *International Journal of Numerical Methods for Heat & Fluid Flow* **18,** 835–867. ISSN: 0961-5539. doi:10.1108/09615530810898980 (2008).

89. Principe, J. & Codina, R. Mathematical models for thermally coupled low speed flows. *Advances in Theoretical and Applied Mechanics* **2,** 93–112. http://www.m-hikari.com/atam/atam2009/atam1-4-2009/principeATAM1-4-2009.pdf (2009).

90. Rubino, S. A streamline derivative POD-ROM for advection-diffusion-reaction equations. *ESAIM: Proceedings and Surveys* **64** (eds Carassus, L. *et al.*) 121–136. ISSN: 2267-3059. doi:10.1051/proc/201864121 (2018).

91. Ryckelynck, D. A priori hyperreduction method: An adaptive approach. *Journal of Computational Physics* **202,** 346–366. ISSN: 0021-9991. doi:10.1016/j.jcp.2004.07.015 (2005).

92. Ryckelynck, D., Vincent, F. & Cantournet, S. Multidimensional a priori hyper-reduction of mechanical models involving internal variables. *Computer Methods in Applied Mechanics and Engineering* **225-228,** 28–43. ISSN: 0045-7825. doi:10.1016/j.cma.2012.03.005 (2012).

93. Saad, Y. *Iterative Methods for Sparse Linear Systems* 421–426. ISBN: 978-0-89871-534-7. doi:10.1137/1.9780898718003 (Society for Industrial and Applied Mathematics, 2003).

94. Schmid, P. J. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics* **656,** 5–28. ISSN: 0022-1120. doi:10.1017/S0022112010001217 (2010).

95. Sirovich, L. Turbulence and the dynamics of coherent structures. I. Coherent structures. *Quarterly of Applied Mathematics* **45,** 561–571. ISSN: 0033-569X. doi:10.1090/qam/910462 (1987).

96. Sirovich, L. Turbulence and the dynamics of coherent structures. II. Symmetries and transformations. *Quarterly of Applied Mathematics* **45,** 573–582. ISSN: 0033-569X. doi:10.1090/qam/910463 (1987).

97. Sirovich, L. Turbulence and the dynamics of coherent structures. III. Dynamics and scaling. *Quarterly of Applied Mathematics* **45,** 583–590. ISSN: 0033-569X. doi:10.1090/qam/910464 (1987).

98. Stabile, G., Ballarin, F., Zuccarino, G. & Rozza, G. A reduced order variational multiscale approach for turbulent flows. *Advances in Computational Mathematics,* 1–15. ISSN: 1019-7168. doi:10.1007/s10444-019-09712-x (2019).

99. Tello, A. & Codina, R. *Three Field Fluid Structure Interaction by means of the Variational Multiscale method.* 2020. Submitted.

100. Tello, A., Codina, R. & Baiges, J. Fluid Structure Interaction by means of Variational Multiscale Reduced Order Models. *International Journal for Numerical Methods in Engineering.* ISSN: 0029-5981. doi:10.1002/nme.6321 (2020).

101. Ullmann, S., Rotkvic, M. & Lang, J. POD-Galerkin reduced-order modeling with adaptive finite element snapshots. *Journal of Computational Physics* **325,** 244–258. ISSN: 1090-2716. doi:10.1016/j.jcp.2016.08.018 (2016).

102. Volkwein, S. *Proper Orthogonal Decomposition: Theory and Reduced-Order Modelling* https://www.math.uni-konstanz.de/numerik/personen/volkwein/teaching/POD-Book.pdf.

103. Wang, Z., Akhtar, I., Borggaard, J. & Iliescu, T. Two-level discretizations of nonlinear closure models for proper orthogonal decomposition. *Journal of Computational Physics* **230,** 126–146. ISSN: 0021-9991. doi:10.1016/j.jcp.2010.09.015 (2011).

104. Wasberg, C. E., Gjesdal, T., Reif, B. A. P. & Andreassen, O. Variational multiscale turbulence modelling in a high order spectral element method. *Journal of Computational Physics* **228,** 7333–7356. ISSN: 0021-9991. doi:10.1016/j.jcp.2009.06.029 (2009).

105. Wells, D., Wang, Z., Xie, X. & Iliescu, T. An evolve-then-filter regularized reduced order model for convection-dominated flows. *International Journal for Numerical Methods in Fluids* **84,** 598–615. ISSN: 0271-2091. doi:10.1002/fld.4363 (2017).

106. Zienkiewicz, O. C. & Zhu, J. Z. The superconvergent patch recovery and a posteriori error estimates. Part 2: Error estimates and adaptivity. *International Journal for Numerical Methods in Engineering* **33,** 1365–1382. ISSN: 1097-0207. doi:10.1002/nme.1620330703 (1992).