SISSA
Scuola
Internazionale
Superiore di
Studi Avanzati

Mathematics Area – PhD course in
Mathematical Analysis, Modelling, and Applications

On the application of the Reduced Basis Method to Fluid-Structure Interaction problems

Candidate:
Monica Nonino

Advisor:
Prof. Gianluigi Rozza
Co-advisor:
Dr. Francesco Ballarin

Academic Year 2019-2020

Declaration of Authorship

I, Monica Nonino, declare that this thesis titled, "On the application of the Reduced Basis Method to Fluid–Structure Interaction problems" and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

Acknowledgements

I would like to thank my supervisor, Professor Gianluigi Rozza, for the constant support he has given me in the past four years. My PhD experience has been beautiful, demaning, challenging, frustrating from times to times, but it was an extraordinary opportunity, that gave me the possibility to grow as a woman and as a researcher. All of this would not have been possible if Professor Rozza had not welcome me in his research group. Thank you, for I could never be more grateful to have had you as a guiding figure.
I thank also my co–advisor Dr. Francesco Ballarin: thank you for always finding the time to discuss things with me, to answer my questions, to help me whenever I needed it the most. I learnt so much from you, and last but not least, the patience it takes to be a wonderful teacher.
I would like to thank Professor Yvon Maday, who always gave me "food for thought" and who always started interesting and stimulating conversations. A huge thank you goes also to another collaborator, Dr. Efthymios Karatzas, who patiently guided me in the world of CutFEM, always answering all of my questions and my doubts: thank you for finding the time to collaborate with me!
Now I switch to italian, for more personal akcknowledgements...
Ringrazio per primi i miei compagni di dottorato: questo viaggio non sarebbe stato lo stesso, senza di voi. A Luca, Federico, Maria, Matteo e Saddam: grazie per aver alleggerito con sorrisi e battute anche le giornate più pesanti, grazie per avermi insegnato molto, soprattutto ad uscire dai miei schemi. Grazie per aver condiviso tutto con me: concerti, feste, pianti, risate, discorsi seri e meno seri. Mi avete arricchito la vita. Ringrazio Flavia, che con la sua dolcezza e la sua gentilezza, è sempre stata un porto sicuro per me. Ringrazio Alessandro, l'amico di una vita: hai portato con te l'atmosfera di casa che mi ha sempre donato serenità. Grazie per esserci stato, dopo tutti questi anni: spero di poter continuare a festeggiare i nostri grandi traguardi, sempre sorridenti e soprattutto, insieme! Infine, ringrazio le persone a me più care. Ringrazio la nonna Mafalda: sei un dono incredibile, con i tuoi sorrisi, la tua intelligenza e la tua forza di volontà! Spero di diventare come te un giorno. Grazie perchè non ti dimentichi mai di me, e perchè mi spingi sempre a dare il massimo. Sono una nipote fortunatissima. Ringrazio anche la nonna Renata, che è sempre presente con il cuore, che rappresenta la forza, la perseveranza, la saggezza accumulata in una vita. Ringrazio mia mamma Franca, per aver sempre creduto in me, per essere sempre stata pronta, con una chiamata, a ricordarmi che lei è lì e mi pensa. Grazie perchè mi insegni molto più di quanto tu te ne renda conto. Ringrazio mio papà Mario, perchè senza di lui tutto questo non sarebbe stato possibile: grazie per aver creduto in me, per avermi sempre spronata, per essere sempre lì pronto ad aiutarmi in qualsiasi modo; grazie per i caffè al mattino, e per i viaggi in treno per portarmi il mazzo di chiavi di casa! Ringrazio mio fratello Isacco, che è una costante fonte di ispirazione, di interesse, di confronto. Grazie perchè mi vuoi bene nonostante la mia vita piena di impegni, e perchè, nonostante siamo entrambi ben che adulti, mi permetti ancora di essere la tua sorellona. Ringrazio Bernard: le parole non bastano per dire quanto ti amo. Sei la mia roccia, il mio punto fisso (anche se giri parecchio!): averti conosciuto è stata una delle fortune più grandi della vita. Mi hai aperto la mente e il cuore. Crescere e maturare a fianco a te è un'esperienza unica, così come lo è stato condividere fino alla fine questo percorso: non sarebbe la stessa cosa, senza di te. La tua forza, la tua perseveranza, la tua determinazione sono un esempio per me. Spero di poter trascorrere il resto della mia vita a festeggiare piccoli e grandi traguardi insieme a te. Infine, ringrazio me stessa, per essermi data questa possibilità fantastica che è

stato il Dottorato: questo percorso mi ha arricchita così tanto, che se dovessi ritornare indietro nel tempo, farei la stessa scelta. Sempre.

Trieste, 24 Settembre 2020.

Contents

List of Figures

List of Tables

Dedico questa tesi alla mia splendida famiglia

Introduction

In the last decades there has been a growing interest for approximation techniques that exploit high performance computing with different fields of applications: industrial applications, naval engineering, aeronautics engineering, medical engineering. Some examples that may arise from these various fields are: heat transfer problem, electromagnetic problems, structural mechanics problems (linear/ nonlinear elasticity), fluid problems, acoustic problems. In all these examples, the models are described through a (system of) partial differential equation (PDE) that usually depends on a given number of parametrs; these parameters can describe the geometrical configuration of the physical domain over which the problem is formulated, they can describe some physical quantities (e.g. the Reynolds number for a fluid or the Lamè constants for a solid), or some boundary conditions. For all these models we usually focus on a particular quantity of interest, such as the maximum temperature of a system, a pressure drop, a channel flowrate. The idea is that, for each of these applications we have an input, which is represented by a given value of the parameter(s) entering in the equation, and we would like to compute, for each such value, the ouput of interest. Unfortunately to compute such output for each new value of the parameter is a difficult task, that is expensive both in terms of time computation and in terms of computer memory, even on modern HPC systems. It is exactly at this point that the Reduced Basis Method (RBM) comes into play: the idea at the core of the method is to simulate the behaviour of the solution of our system of interest for some chosen values of the parameters in the PDE. This is usually done with some well established discretization technique, such as Finite Element Method (FEM); another discretization method, used for example in the compressible framework in computational fluid dynamics is the Finite Volume Method (FV), and yet another possibility is the CutFEM method. Once we have computed these solutions, in an expensive offline phase, we can use them to build some other basis functions: with these new basis functions, in the inexpensive online phase, we approximate the solution of the system, for a new value of the parameter.
With this idea in mind, the goal of this thesis is to give an extensive overview on the application of the Reduced Basis Method to Fluid–Structure Interaction (FSI) problems. Depending on the approach adopted to adress a multiphysics problem, we will see how we can adapt, modify, improve the RBM in order to obtain a complete model order reduction procedure. In the following, we will introduce several different test cases: a toy problem that describes the behaviour of two leaflets under the influence of the jet of a fluid; a FSI problem whose solution exhibits a transport dominated behaviour, and also some computational fluid dynamics toy problems, that depend on a parameter (that can be physical or geometrical). For each one of these test cases, we first introduce the problem formulation, then we present the model order reduction procedure.

In Chapter 1 we give a short introduction to the Reduced Basis Method: we present the mathematical setting within which the method is formulated, we give some definitions and we set some notation.

In Chapter 2 we introduce briefly Fluid–Structure Interaction problems: we present some of the many everyday situations where multiphysics systems are used to simulate the solution of a particular problem of interest. We then give a quick overview of the two main approaches that are used to adress coupled problems, namely partitioned (segregated) approaches and monolithic approaches. Finally, we set the mathematical formalism that is necessary to introduce the Arbitrary Lagrangian Eulerian (ALE) formulation, widely used in the FSI community.

Chapter 3 is devoted to the implementation of a model order reduction technique for FSI problems that are solved through a partitioned procedure. We present the problem formulation and we design the model order reduction procedure; finally, we show some numerical results. In the second part of the Chapter we introduce a similar test case, characterized by a geometrical parametrization of the domain; we therefore modify slightly the reduction procedure, and at the end we present some numerical results.

Chapter 4 is devoted to the study of reduced order models for advection dominated problems: for these problems indeed, the corresponding solution manifold shows a slow decay of the Kolmogorov n-width, or, in practical calculations, its computational surrogate given by the magnitude of the eigenvalues returned by a Proper Orthogonal Decomposition on the solution manifold. The problems considered in this Chapter are two: the first test case is a time dependent computational fluid dynamics problem (CFD), where a fluid flows around a circular obstacle that rotates counterclockwise; this problem is studied both in the non parametrized and in the parametrized setting. The second test case is a FSI problem, which describes the interaction between a fluid flowing in a channel and the compliant walls of the channel. We enrich the standard reduction method with an additional preprocessing during the offline phase, in order to obtain smaller reduced basis spaces. Such preprocessing consists in the composition of the snapshots with a transportation map belonging to a family of smooth and invertible functions that map the physical domain of the problem into itself. A comparison between the results of the novel offline stage and the standard one is presented.

In Chapter 5 we focus on steady and unsteady Navier–Stokes problems, possibly with moving boundaries, that are discretized with an unfitted mesh Finite Element Method. The work presented in this Chapter extends the existing approaches in the Reduced Basis Method within a CutFEM setting to nonlinear CutFEM discretization as well as to evolutionary in time fluid flow problems. We construct and investigate a unified and geometry independent reduced basis which overcomes many barriers and complications of the past. These complications may occur whenever geometrical morphings are taking place: in this way, we may avoid remeshing and transformations to reference domains, and we may be able to handle complex geometries. The combination of a fixed background mesh in a fixed extended background geometry with reduced order techniques appears beneficial and advantageous in many industrial and engineering applications, which could not be resolved efficiently with standard discretization techniques and standard reduced order models.
The work presented in this Chapter is a preliminary work that lays the foundation for a more general approach, that we would like to develop in the future: this approach will be applied to FSI problems, in particular in those situations where the structure undergoes a large deformation. Indeed, in the presence of large deformations in the physical system, classical discretization methods, such as the FEM, would require very

expensive procedures, like remeshing; instead, with a CutFEM approach this barrier can be overcome.

In Chapter 6 we summarize all the work presented in this thesis, and we give some general conclusions and future perspectives for these lines of research.

At the end of the thesis, in the Appendix, we present more in detail the calculations that led to the formulation of a particular coupling condition, which is used in the algorithm presented in Chapter 3.

Chapter 1

The Reduced Basis Method: an introduction

1.1 Some historical background

The aim of this overview is to present some historical background for the developement of the RBM, and to provide some literature for the interested reader. The following quick overview does not aspire to be complete and exhaustive, as there are many new works that bring along advances in some of the aspects of the RBM.
The idea of the RBM, as we briefly mentioned in the Introduction, is to identify some reduced basis functions, and to use those functions to effectively approximate the solution of the problem of interest, for any new value of the parameter in the system. This idea is certainly not recent, and some early work focusing on the efficient evaluation in the many–query context can be found in [62]; some early work focusing instead on efficient parameter continuation techniques for nonlinear problems can be found in [3, 134, 135]. These early works were soon extended to a more large class of problems; however, at that stage, the developed methods were still missing a posteriori error estimators, and thus there were still a lot of open questions about the accuracy and the reliability of the reduction techniques. For this reason a great effort was put in the definition of error estimator procedures and rigorous a posteriori error bounds, see [140, 123, 122, 117, 140]. Thanks to the avaliability of some error estimators, it has been possible to design effective sampling strategies [130, 124, 154, 174, 28, 145], which are particularly useful in presence of many parameters in the problem, and that are based on Greedy algorithms. An alternative to these sampling strategies, that does not rely on the avaliability of error estimators, is given by the Proper Orthogonal Decomposition (POD) [81, 108, 149, 181]: in the next Section we will explain briefly this technique, since it is the one that has been used in all the works presented in this thesis. In general, a combination of the two techniques (POD and sampling with a Greedy algorithm) can be used, especially for those problems which are both unsteady (i.e. time dependent) and parametric: for a reference, see [29, 46, 82]. A posteriori error bounds for the steady Navier–Stokes problem can be found in [173], however, for unsteady problems, the certification of the reduction technique represents a great challenge, see [129, 110, 107]. This is also the reason why in this thesis we do not present any certification of the procedure that we have used: the formulation and the study of error estimators for unsteady Fluid–Structure Interaction problems is a great challenge still nowadays, and it goes beyond the scope of this work to present certified error bounds for the systems that we will consider. Concerning multiphysics problems, especially Fluid–Structure Interaction problems, some first model reduction approaches were proposed in [111, 109] for some simple setting.

At the early stages of its developement, the proposed Reduced Basis Method did not fully decouple the Finite Element discretization of the parametrized problem of interest, from the online projection to a reduced basis space; the method proposed therefore showed a rather modest gain in terms of computational savings. From this starting point, a lot of effort has been put in achieving a fully decoupling between the Finite Element discretization and the reduced order model, through an offline–online decoupling. In the case of affine parameter dependence, this decoupling is rather natural and has been proposed in several works, see for example [93, 140, 15]. On the contrary the decoupling is much more difficult in the case of non affine parameter dependence; this difficulty has led to the developement of the Empirical Interpolation Method (EIM), see [77, 116, 119, 121]. Thanks to the EIM, it has been possible to design Reduced Basis Method that can be effectively applied also to very complex may–query, real time applications, including nonlinear problems.

1.2 The Reduced Basis Method

In the following, we fix the mathematical setting in which the RBM operates, we give some definitions and we show how to carry on a model order reduction procedure for a given problem of interest; the reader interested in a detailed discussion is referred to [90, 146].
Let us consider a sufficiently regular physical domain $\Omega \subset \mathbb{R}^d$, with $d = 1, 2, 3$, and let us denote by Γ_D the portion of $\partial\Omega$ where we impose Dirichlet boundary conditions, and by Γ_N the portion of $\partial\Omega$ where we impose Neumann boundary conditions. Let $\mu \in \mathcal{P}$ be a parameter, with $\mathcal{P} \subset \mathbb{R}^k$ the parameter space and with $k \geq 1$. In order to ease the exposition we now consider only scalar–valued variables, but everything we will say can be applied also to vector–valued variables. We want to solve the following problem: for every $\mu \in \mathcal{P}$, find $u(\mu)\colon \Omega \mapsto \mathbb{R}$ such that:

$$\begin{cases} \mathcal{L}(u;\mu) = f(\mu) & \text{in } \Omega, \\ u(\mu) = g(\mu) & \text{on } \Gamma_D, \\ \partial_n u(\mu) = h(\mu) & \text{on } \Gamma_N, \end{cases}$$

where $\mathcal{L}$ is a linear operator, and g and h are some prescribed functions . The weak formulation of the previous problem can be written in the following form: for every $\mu \in \mathcal{P}$ find $u(\mu) \in H^1(\Omega)$ such that $u(\mu) = g(\mu)$ on Γ_D, and such that

$$a(u(\mu), v; \mu) = \ell(v; \mu), \tag{1.1}$$

for all $v \in H^1_D(\Omega)$, where $H^1_D(\Omega) := \{v \in H^1(\Omega) | v = 0 \quad \text{on } \Gamma_D\}$. Let us denote now by $||\cdot||_{H^1}$ the intrinsic norm of the Hilbert space $H^1(\Omega)$. Under the hypothesis that: for all $\mu \in \mathcal{P}$ the bilinear form $a(\cdot;\cdot,\mu)$ is symmetric, coercive and continuous with respect to the norm $||\cdot||_{H^1}$, and the linear form $\ell(\cdot;\mu)$ is continuous with respect to the norm $||\cdot||_{H^1}$, thanks to the Lax–Milgram theorem we can say that the problem (1.1) is well posed.
Once we have stated the weak formulation of the problem of interest, the first question that we have to adress is: how can we find the reduced basis that will allow us to efficiently approximate any solution of (1.1), for any $\mu \in \mathcal{P}$? The first thing to do, in this sense, is to carry out a discretization step. For this reason we introduce an approximation space $V_h \subset H^1(\Omega)$: V_h can be, for example, the space of continuous functions that are piecewise polynomials of degree k on the elements of a suitable

triangulation of the physical domain Ω. The discretized version of problem (1.1) is now: for any $\mu \in \mathcal{P}$, find $u_h(\mu) \in V_h$ such that $u_h(\mu) = g(\mu)$ on Γ_D and:

$$a(u_h(\mu), v_h; \mu) = \ell(v_h; \mu), \tag{1.2}$$

for all $v_h \in V_{h,0}$, where $V_{h,0} := \{v \in V_h | v = 0 \quad \text{on } \Gamma_D\}$.

Definition 1.2.1. *Problem* (1.2) *is called* truth problem *or* high fidelity problem. *The solution* $u_h(\mu)$ *is called the* snapshot.

Let us now introduce the solution manifold of the discretized version of the problem of interest:

Definition 1.2.2. *The* discretized solution manifold *of* (1.2) *is*

$$\mathcal{M}_h = \{u_h(\mu), \quad \mu \in \mathcal{P}\}.$$

A basic assumption for the RBM is that $\mathcal{M}_h$ has a low dimension, meaning that $\mathcal{M}_h$ (and so any discretized solution of (1.2)) can be approximated with a small error by the span of a small number of appropriately chosen basis functions. This assumption is fundamental in order to have an efficient model order reduction, and in Chapter 4 we will see what happens to the RBM when this assumption is no longer satisfied.
Let us assume that we have the basis functions $\Phi_1, \dots, \Phi_N$, and let $V_N := \text{span}\{\Phi_i\}_{i=1}^N$. We can now define

Definition 1.2.3. *The* reduced solution $u_N(\mu)$*, for any* $\mu \in \mathcal{P}$*, is defined as follows:*

$$u_N(\mu) = \sum_{j=1}^{N} \underline{u}_N^j(\mu)\Phi_j,$$

with $\underline{u}_N^j(\mu) \in \mathbb{R}\ \forall \mu \in \mathcal{P}$. The reduced solution is therefore a linear combination of the basis functions Φ_j.
The reduced order problem now reads: for every $\mu \in \mathcal{P}$, find $\underline{u}_N^1(\mu), \dots, \underline{u}_N^N(\mu)$ such that:

$$\sum_{j=1}^{N} \underline{u}_N^j(\mu) a(\Phi_j, \Phi_k; \mu) = \ell(\Phi_k; \mu), \quad \forall k = 1, \dots, N. \tag{1.3}$$

We now assume that the problem satisfies an affine parameter dependence, i.e. it is possible to write

$$a(\Phi_j, \Phi_k; \mu) = \sum_{q=1}^{M} \theta_a^q(\mu) a_q(\Phi_j, \Phi_k),$$
$$\ell(\Phi_k; \mu) = \sum_{q=1}^{M} \theta_\ell^q(\mu) \ell_q(\Phi_k),$$

then we can see that, in order to be able to find a reduced order solution, we can build and store *once and for all* the parameter independent quantities $(\boldsymbol{A}^q)_{jk} := a_q(\Phi_j, \Phi_k)$ and $(\boldsymbol{L}^q)_k := \ell_q(\Phi_k)$ during the offline stage. Then, during the online stage, we will compute the parameter dependent quantities $\theta_a^q(\mu)$ and $\theta_\ell^q(\mu)$, and we will build the

following:

$$\boldsymbol{A}_N^q := \boldsymbol{Z}^T \boldsymbol{A}^q \boldsymbol{Z},$$
$$\boldsymbol{A}_N(\mu) := \sum_{q=1}^{M} \theta_a^q(\mu) \boldsymbol{A}_N^q,$$
$$\boldsymbol{L}_N^q := \boldsymbol{Z}^T \boldsymbol{L}_N^q,$$
$$\boldsymbol{L}_N(\mu) := \sum_{q=1}^{M} \theta_\ell^q(\mu) \boldsymbol{L}^q.$$

Here $\boldsymbol{Z}$ is the *reduced basis matrix*: the j–th column of $\boldsymbol{Z}$ represents the j–th reduced basis Φ_j expressed in terms of the basis functions that generate the space V_h. At the end we will have to solve the following linear system:

$$\boldsymbol{A}_N(\mu)\underline{u}_N(\mu) = L_N(\mu).$$

We remark here that, if we introduce $\mathcal{N} = \dim V_h$, then $N \ll \mathcal{N}$: the dimension of the system to be solved in the online phase is much smaller than the dimension of the system to be solved in the offline phase (which is therefore the more expensive phase). Now that we have defined everything that we need, the next step is to understand how we can find a suitable set of reduced basis functions.

1.3 Reduced basis functions generation

As we previously mentioned, there are two possible ways of building the reduced basis functions, in the case of a parametrized problem: we can either use a sampling strategy based on a Greedy algorithm, or we can use a Proper Orthogonal Decomposition (POD). In the remaining of the thesis we will be working with the POD, nevertheless we briefly explain here also the Greedy procedure. Both strategies rely on a *discretized parameter space* $\mathcal{P}_{train} = \{\mu_1, \dots, \mu_{N_{train}}\} \subset \mathcal{P}$.

Greedy algorithm

The Greedy procedure aims at progressively constructing the reduced basis space V_N, and it relies on the avaliability of an error estimator, which we will call $\varepsilon(\cdot)$:

$$||u_h(\mu) - u_N(\mu)||_\mu := \sqrt{a(u_h(\mu) - u_N(\mu), u_h(\mu) - u_N(\mu); \mu)} \leq \varepsilon(\mu), \quad \forall \mu \in \mathcal{P}.$$

Given the training set $\mathcal{P}_{train}$, the Greedy algorithm works in the following way: at the n–th iteration of the algorithm, we have a reduced basis space $V_N^{(n)} := \text{span}\{\Phi_1, \dots, \Phi_n\}$. At iteration $n+1$ we consider the parameter μ_{n+1} that satisfies the property

$$\mu_{n+1} = \arg \max_{\mu \in \mathcal{P}_{train}} \varepsilon(\mu);$$

we then compute the corresponding snapshot $u_h(\mu_{n+1})$, and we add it to the reduced basis space. The procedure can be stopped when the maximum error estimated with $\varepsilon(\cdot)$ falls under a certain tolerance. We remark that this procedure allows us to construct hierarchical reduced basis spaces $V_N^{(1)} \subset \dots \subset V_N^{(N)}$.

Proper Orthogonal Decomposition

The Proper Orthogonal Decomposition (POD) is quite different with respect to the previous strategy: in fact, in the POD we directly compute the snapshots $u_h(\mu)$ corresponding to *all* the parameters in $\mathcal{P}_{train}$. Then, we somehow get rid of the unnecessary information and retain only the "most important" functions.

Definition 1.3.1. *The* POD space V_N *is the linear space of dimension* N *that minimizes the following quantity:*

$$\sqrt{\frac{1}{N_{train}} \sum_{\mu \in \mathcal{P}_{train}} \inf_{v \in E} ||u_h(\mu) - v||^2_{H^1}}$$

over all the linear subspaces $E \subset V_{\mathcal{M}} := span\{u_h(\mu), \mu \in \mathcal{P}_{train}\}$ *of dimension exactly* N*.*

To construct the space V_N, we start by computing the snapshots $u_h(\mu)$ for $\mu \in \mathcal{P}_{train}$. With these snapshots we build the so called *correlation matrix* $\boldsymbol{C} \in \mathbb{R}^{N_{train} \times N_{train}}$:

$$\boldsymbol{C}_{ij} = \frac{1}{N_{train}} (u_h(\mu_i), u_h(\mu_j))_{H^1}, \quad 1 \leq i, j \leq N_{train}. \tag{1.4}$$

Then we solve an eigenvalue–eigenvector problem:

$$\boldsymbol{C}\boldsymbol{v}_k = \lambda_k \boldsymbol{v}_k, \quad 1 \leq k \leq N,$$

where $\lambda_1 \geq \ldots \geq \lambda_N$ are ordered by decreasing order of magnitude. Finally the reduced basis functions Φ_k are defined as:

$$\Phi_k = \frac{1}{N_{train}} \sum_{j=1}^{N_{train}} (\boldsymbol{v}_k)_j u_h(\mu_j),$$

where $(\boldsymbol{v}_k)_j$ denotes the j–th component of the eigenvector $\boldsymbol{v}_k$.
As we can see from the definition of the Greedy algorithm and of the POD, the latter is a more expensive procedure, as it requires the solution of the truth model for all the parameters in the training set. Nevertheless the POD can be implemented also for those classes of problems where there is no error estimator avaliable, as it happens for the problems that we will focus on, namely unsteady nonlinear FSI problems. For this reason, in the rest of the thesis we will rely on the POD.

1.4 Perspectives

In this Chapter we gave a quick introduction to the Reduced Basis Method: even though the presentation is by no means exhaustive, we discussed the main aspects of the reduction technique, such as the two procedures that can be used to generate a reduced basis space. The research in the RBM community is very active and fruitful, and it is moving in a number of different directions. For what concerns the application of the reduction techniques to computational fluid dynamics (CFD), a few directions of research nowadays are: the use of artificial intelligence in the recognition of a physical phenomenon, and its incorporation in reduction methods (e.g. by means of convolutional autoencoders); the integration of the reduction methods with discretization

techniques other than the Finite Element Method (e.g. CutFEM, XFEM, Shifted Boundary Method) especially in the case of geometrically parametrized systems; the adaptation of the reduction methods in order to be able to deal with problems that are dominated by an advection effect. Analyzing all of these lines of research, and many more others, is out of the scope of this thesis; nevertheless some aspects previously mentioned will be indeed considered, such as the integration of RBM with CutFEM and the modification of the RBM for advection dominated problems.

Chapter 2

An overview of Fluid–Structure Interaction problems

Fluid–Structure Interaction (FSI) problems are a wide spread topic in the applied mathematics community. Even though the research for computational methods for solving FSI problems has a long history, a comprehensive presentation from a mathematical point of view is missing still nowadays: one of the reasons for this is the fact that the two subproblems, namely the Navier–Stokes equation and the elastic solid equation, are two big mathematical challenges on their own, see for example [75].
Despite their instrinsic complicated nature (see [53, 73]), FSI problems are very common and frequently used in the simulation of a lot of situations: in naval engineering, they are used to study the interaction between the water and the hull of a ship, see for example [115]; in biomedical applications FSI problems describe the interaction between the blood flow and the deformable walls of a vessel (as an example of the implementation of FSI in the medical field see [177, 141, 11, 142, 147, 177, 118]); in aeronautical engineering, FSI describes the way the air interacts with a plane or with (parts of) a shuttle, see [148, 49, 54, 114]. These are just some of the many examples of applications for FSI, and these problems have different characteristics, that lead to different dynamics of the coupled system: the aerospace applications are in the setting of high Reynolds number, and turbulent regime; in addition, the fluid is usually compressible and has a behaviour that requires a three dimensional model. On the other hand, for biomedical applications we are usually in a laminar regime, the models can be either two dimensional or three dimensional, and the blood is usually modelled as an incompressible fluid. Another very important difference in all these problems is represented by the physical quantities describing the solid, and mainly the solid density ρ_s, and its ratio with respect to the density ρ_f of the fluid under consideration: indeed, as we will mention in the next Chapter, there are some algorithms for solving FSI problems that may work very well for a problem that couples the air with a very stiff material, but cease to work when the problem couples the blood flow with some biological tissue. Other difficulties are given by the different assumptions on the structure, and hence the different dynamics that one can obtain, see e.g [18]. As we can see, therefore, FSI problems give rise to a wide set of different situations, different dynamics, and they require different computational tools, depending on the setting in which we are working: the aforementioned difficulties are just some of the many peculiarities of multiphysics problems, that contribute to make it a very difficult topic of research.
In the rest of this Chapter, we will first of all give a quick introduction on the dynamics of FSI problems; then we will give an overview of the mathematical challenges

hidden in these problems, and finally we will present briefly two different classes of approaches used to adress FSI problems, namely partitioned approaches and monolithic approaches.

2.1 Dynamics of Fluid–Structure Interaction problems

The dynamics of the FSI problems is realized by the interplay of the fluid and the solid: this interplay is possible thanks to the coupling of the two different physics at the FSI interface, namely the part of the physical domain that is common to the fluid subdomain and the solid subdomain. This coupling is the consequence of three different principles, that we present hereafter.

- *Continuity of the displacement*: this condition imposes the continuity of the solid displacement and the fluid displacement at the FSI interface. This is a geometrical condition, which is the mathematical translation of the hypothesis that the fluid and the solid domain do not overlap.
- *Continuity of the velocity*: this is a kinematic condition, that represents the hypothesis that the fluid sticks to the moving FSI interface. This condition resembles in some ways the no–slip boundary condition, very common in viscous fluid dynamics. In aeroelasticity application, the continuity of the velocity at the FSI interface is relaxed and substituted with the non penetrating condition, that prescribes the motion just in the direction normal to the FSI interface.
- *Balance of the stresses*: this is a classical action–reaction principle, that imposes the balance between the fluid and the solid stresses at the FSI interface.

As we can see, the previous three coupling conditions are imposed at the FSI interface: one of the big challenges and difficulties of coupled problems is represented by the fact that this interface is not fixed, but it deforms in time, and this deformation is not known a priori: it is one of the unknowns in the system. Another difficulty is represented by the fact that the fluid domain in FSI applications is a moving domain; in solid mechanics it is common to deal with moving domains, and the deformation of the domain is usually the unknown of the problem; for fluid dynamics instead one usually considers fixed domains. This different point of view is an intrinsic characteristic of the FSI problems, and it gives rise to a formalism, very known and widely used in the community, which is called the *Arbitray Lagrangian Eulerian formulation*.

2.1.1 The Arbitrary Lagrangian Eulerian formulation

Let $\Omega(t) \subset \mathbb{R}^2$ be the physical domain over which the FSI problem is formulated: $\Omega(t) = \Omega_f(t) \cup \Omega_s(t)$, where $\Omega_f(t) \subset \mathbb{R}^2$ and $\Omega_s(t) \subset \mathbb{R}^2$ are the fluid and the solid domain at time t, respectively; we also assume that the two domains do not overlap, i.e. $\Omega_f(t) \cap \Omega_s(t) = \emptyset$. In order to formulate the FSI problem, we will adopt an Arbitrary Lagrangian Eulerian (ALE) formulation; the ALE formulation [151, 50, 91, 19] is widely used for the simulation of Fluid–Structure Interaction problems. The ALE formulation arises from the fact that to describe the behaviour of a solid and the behaviour of a fluid we usually use two different kind of approaches; in fact, to describe the behaviour of a solid it is common practice to use the so called *Lagrangian formalism*: all the quantities and the conservation laws are formulated on the reference configuration $\hat{\Omega}_s = \Omega_s(t = 0)$. On the contrary, when describing the behaviour of

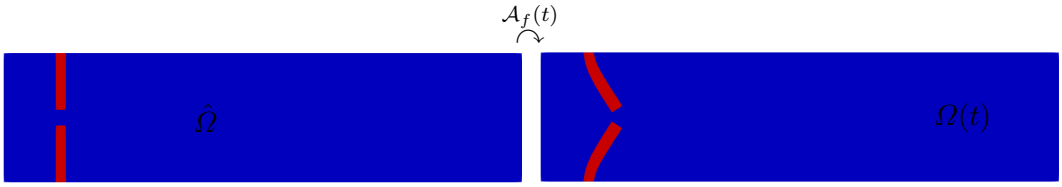


FIGURE 2.1: Example: domain reference configuration $\hat{\Omega}$ (left) and domain configuration at time t, $\Omega(t)$ (right). In blue we have the fluid domain, in red the solid domain.

a fluid, the definition of "domain reference configuration", "original position of a particle" is not clear, and therefore the *Eulerian formalism* is used instead: all the quantities and the conservation laws are formulated on the configuration $\Omega_f(t)$ at the current time t. In order to be able to describe both the fluid and the solid, a mixed formulation (the ALE formulation indeed) is used: the underlying idea is that of pulling back the fluid equations to an arbitrary time–independent configuration $\hat{\Omega}_f$: one possibel choice for $\hat{\Omega}_f$ is $\hat{\Omega}_f = \Omega_f(t=0)$, the domain at initial time. In Figure 2.1 we can see an example of a reference configuration and the configuration of the domain at the current time t. Let us see in the next paragraph how to introduce the ALE formalism; for a more detailed discussion about different approaches to describe coupled systems we refer to [83, 151]. Let $[0,T]$ be a time interval, and let $\hat{\Omega}_f$ be a *reference configuration* for the fluid.

Definition 2.1.1. *The ALE mapping $\mathcal{A}_f(t)$, for every $t \in [0,T]$ is defined as follows:*

$$\begin{aligned} \mathcal{A}_f(t)\colon \hat{\Omega}_f &\mapsto \Omega_f(t) \\ \hat{\boldsymbol{x}} &\mapsto \boldsymbol{x} = \hat{\boldsymbol{x}} + \hat{\boldsymbol{d}}_f(\hat{\boldsymbol{x}}, t), \end{aligned}$$

where $\hat{\boldsymbol{d}}_f(t)\colon \hat{\Omega}_f \mapsto \hat{\Omega}_f$ is the *mesh displacement*. The definition of $\hat{\boldsymbol{d}}_f$ usually changes depending on the kind of fluid problem we want to model; for a FSI problem, the mesh displacement is defined as an extension of the solid displacement to the whole fluid domain. Let $\hat{\Omega}_s \subset \mathbb{R}^2$ be the reference configuration for the solid, and let $\hat{\boldsymbol{d}}_s(t)\colon \hat{\Omega}_s \mapsto \mathbb{R}^2$ be the solid displacement: one possible way to define $\hat{\boldsymbol{d}}_f$ is through an *harmonic extension* of $\boldsymbol{d}_s$:

$$\begin{cases} -\Delta \hat{\boldsymbol{d}}_f = 0 & \text{in } \hat{\Omega}_f, \\ \hat{\boldsymbol{d}}_f = \hat{\boldsymbol{d}}_s & \text{on } \hat{\Gamma}_{FSI}, \end{cases}$$

where $\hat{\Gamma}_{FSI}$ is the fluid–structure interface in the reference configuration. For other alternatives on how to define $\boldsymbol{d}_f$ we refer to [151].

Remark 2.1.2. We underline that $\hat{\boldsymbol{d}}_f$ represents the displacement of the grid points, therefore it is not a quantity with a real physical meaning, but rather a geometrical quantity that describes the deformation of the mesh, according to the deformation of the physical domain. It is also important to underline that $\partial_t \hat{\boldsymbol{d}}_f \neq \hat{\boldsymbol{u}}_f$: in fact, while $\hat{\boldsymbol{u}}_f$ represents the velocity of the fluid, $\partial_t \hat{\boldsymbol{d}}_f$ is again a geometrical quantity, that can be interpreted as the velocity with which the mesh moves.

A great attention has to be paid to the definition of the mesh displacement, as different definitions for $\hat{\boldsymbol{d}}_f$ lead to different levels of regularity: if we loose regularity due to the mesh displacement, then we loose regularity at the FSI interface, which is exactly where the coupling between the two physics takes place. It is beyond the scope of

this thesis to discuss the regularity of different definitions of the mesh displacement; nonetheless we refer the interested reader to Chapter 5.3.5 of [151].

2.2 Approaches to Fluid–Structure Interaction problems

In general, to solve a FSI problem there are two different kind of approaches that we can adopt: the first approach consists of a partitioned, or segregated, procedure [14, 58, 56, 59, 57, 42], whereas the second approach consists of the so called monolithic procedure [13, 182]. In the following we briefly introduce partitioned and monolithic algorithms, as they will be employed in the next two Chapters of this thesis, respectively.

2.2.1 Partitioned approaches

Existing simulation tools for fluid dynamics and for structural dynamics are well developed and are used on a daily basis in industrial applications. It is therefore natural to try to combine these computational tools, to adress coupled problems: this is exactly the rationale behind a partitioned algorithm. Indeed, in a partitioned procedure, we solve *separately* the fluid and the solid problems, and then we couple the two physics with some iterative procedure, see for example [72]. Even though the advantage of a partitioned algorithm is the possibility of combining different discretization tools for the two physics (e.g. Finite Volumes for the fluid and Finite Elements for the structure), the drawback of these procedures is that, under some physical and geometrical conditions, they turn out to be unstable: this happens, for example, if the physical domain has a slender shape, or if the fluid density ρ_f is close to the solid density ρ_s, and this is usually the case in haemodynamics applications, where the density of the blood is quite close to the density of the walls of the vessel. The reason for this instability is the so called *added mass effect*: the fluid acts like an added mass to the solid, thus changing its natural behaviour; we refer to [41] for a detailed derivation of the added mass effect and relative consequences.
Partitioned algorithms differ from one another, according to the strategy used to impose the coupling conditions at the FSI interface; indeed, we can classify partitioned algorithms in the following three categories:

- *Explicit algorithms*: after having discretized in time the FSI problem, the conditions on the continuity of the displacement and on the continuity of the velocity at the interface are treated explicitly at every time–step. These algorithms, also known as *weakly* or *loosely* coupled algorithms [30], are succesfully applied in aerodynamics applications, see [55, 139], but some studies (see [41, 96, 112]) showed that they are unstable under some physical and geometrical conditions, due to the added mass effect, as we previously mentioned.
- *Implicit algorithms*: in these algorithms, also known as *strongly* coupled algorithms, the coupling conditions are treated implicitly at every time–step, see for example [175, 176]. This implicit coupling represents a way to circumvent the instability problems due to the added mass effect; nevertheless, an implicit treatment of the coupling conditions leads to algorithms that are more expensive in terms of computational time.

- *Semi–implicit algorithms*: in these algorithms (see [7, 8, 20]), the continuity of the displacement is treated explicitly, whereas the other coupling conditions are treated implicitly. This alternative represents a tradeoff between the computational cost of the algorithm and its stability in relation to the physical and geometrical properties of the problem. In Chapter 3 we will see a reduced order method that is based on this kind of partitioned approach.

2.2.2 Monolithic models

In a monolithic algorithm (see for example [13, 151, 65, 61, 9]), the fluid and the solid problem are solved *simultaneously*. This usually results in algorithms that are more stable, and this is highly desirable, especially if we wish to use large time–steps in our simulations. The main drawback is given by the fact that they deeply rely on the avaliability of an *ad hoc software* that can be used to solve the fluid problem and the solid problem: in this sense, monolithic algorithms are less flexible and more tailored to the particular problem at hand. Moreover, as we will see in Chapter 4, in this thesis, in order to pursue a Galerkin discretization of the original problem, we use two Lagrange multipliers to impose the continuity of the displacement and of the velocity at the interface: this results in the introduction of two new unknowns in the coupled problem.

2.3 Perspectives

In this Chapter we gave a general introduction to the world of Fluid–Structure Interaction problems, and we gave an overview of the two main approaches used in the community to adress such coupled systems. As we have seen, multiphysics problems give rise to a large number of complications, due to the fact that they couple two different physics. From a computational point of view, the research for discretization tools that are able to capture the behaviour of the two physics, while still being computationally tractable, is still very active, both in the framework of partitioned algorithms and of monolithic algorithms. It is not in the scope of this work to give an opinion on which of the two procedures is the best: indeed, the main goal of the author is to present different algorithms which can be used to pursue a model order reduction of some FSI problems of interest. Therefore, in the following two Chapters we will see a Reduced Basis Method that has been designed as a partitioned procedure, and which can thus potentially be combined with different computational tools in the offline phase, and then a Reduced Basis Method that is instead a monolithic procedure.

Chapter 3

Partitioned Reduced Basis Method for FSI problems

The main goal of this Chapter is to present an algorithm that combines the Reduced Basis Method with a partitioned approach to solve multiphysics problems. The first Fluid–Structure Interaction problem considered aims at describing the behaviour of a fluid in a two dimensional cavity with deformable leaflets. We then move on to a more complex Fluid–Structure Interaction problem, where we introduce a geometrical parameter that takes into account the length of the leaflets. The material in this Chapter is contained in a paper in preparation for journal publication, see [133].

3.1 Motivation

Partitioned algorithms, as we have mentioned in the previous Chapter, are particularly useful when we want to couple different discretization tools for the two physics, namely the fluid and the solid problem; even if we decide to use the same computational software, with a segregated procedure we can, in principle, use two different space discretization for the fluid and for the solid, see [74]: this turns out to be extremely useful in the case we want to simulate, for example, the behaviour of a system which has a stiff structure: if the solid does not deform too much, there is no need to use a very fine mesh, whereas it may be useful to use a finer mesh to describe the fluid behaviour, for example in the proximity of the FSI interface, or in the presence of boundary layers.
Additionaly, from the reduced order model point of view, adressing a coupled problem by means of a partitioned procedure is advantageous in terms of computational efficiency: indeed, in the online phase of the RBM, we have to solve, separately, smaller systems. Moreover, with some minor changes such as change of variables and appropriate choices for the couplings, it is possible to further reduce the dimension of the online systems, as we will see in the following.
The goal of the work presented in this Chapter is to combine a segregated procedure with a reduced order model, and to test and analyze the performance of the obtained algorithm.
This Chapter is structured as follows: in Section 3.2 we introduce the first Fluid–Structure Interaction problem of interest, namely the interaction of a fluid with a thick, two dimensional, structure; in Section 3.3 we introduce the partitioned procedure at the high order level. In Section 3.4 we derive the partitioned procedure at the reduced order level, and in Section 3.5 we present the numerical results. In

FIGURE 3.1: Physical reference configuration. Blue domain: the reference fluid configuration $\hat{\Omega}_f$. Red leaflets: the reference solid configuration $\hat{\Omega}_s$. The fluid–structure interface $\hat{\Gamma}_{FSI}$ is depicted in green. $\hat{\Gamma}_s^D$: the part of the leaflets that does not move.

Section 3.6 we introduce the geometrically parametrized version of the problem of interest: in Section 3.6.1 we present the ALE formalism in the presence of a geometrical parametrization of the domain; in Section 3.6.2 we give the strong formulation of the problem of interest, and in Section 3.7 we describe the algortihm at the high order level. In Section 3.8 we introduce the reduced order model, and then we present some numerical results in Section 3.9.

3.2 Problem formulation: time dependent FSI

We now present the formulation of the first FSI problem: a time–dependent, nonlinear, non–parametrized multiphysics test case. We want to simulate the behaviour of an incompressible fluid interacting with a deformable solid, in the time interval $[0, T]$; Figure 3.1 shows the physical domain in its reference configuration. The coupled FSI problem reads as follows: for every $t \in [0, T]$, find $\boldsymbol{u}_f(t)\colon \Omega_f(t) \mapsto \mathbb{R}^2$, $p_f(t)\colon \Omega_f(t) \mapsto \mathbb{R}$ and $\hat{\boldsymbol{d}}_s(t)\colon \hat{\Omega}_s \mapsto \mathbb{R}^2$ such that:

$$\begin{cases} \rho_f(\partial_t \boldsymbol{u}_f + (\boldsymbol{u}_f \cdot \nabla)\boldsymbol{u}_f) - \mathrm{div}\sigma_f(\boldsymbol{u}_f, p_f) = \boldsymbol{b}_f \quad \text{in } \Omega_f(t) \times (0, T], \\ \mathrm{div}\boldsymbol{u}_f = 0 \quad \text{in } \Omega_f(t) \times (0, T], \\ \rho_s \partial_{tt} \hat{\boldsymbol{d}}_s - \hat{\mathrm{div}}\hat{\boldsymbol{P}}(\hat{\boldsymbol{d}}_s) = \hat{\boldsymbol{b}}_s \quad \text{in } \hat{\Omega}_s \times (0, T], \end{cases} \tag{3.1}$$

with some suitable initial conditions and boundary conditions, and with some coupling conditions that we are going to introduce subsequently. In system (3.1), the $\hat{\mathrm{div}}$ denotes the fact that the divergence is computed with respect to $\hat{\boldsymbol{x}}$, the space variable in the reference configuration. ρ_f and ρ_s are the fluid and the solid density, respectively; σ_f is the fluid *Cauchy stress tensor*, and $\hat{\boldsymbol{P}}$ is the solid *second Piola–Kirchoff tensor*, defined respectively as:

$$\begin{aligned} \sigma_f(\boldsymbol{u}_f, p_f) &= \mu_f(\nabla \boldsymbol{u}_f + \nabla^T \boldsymbol{u}_f) - p_f \boldsymbol{I}, \\ \hat{\boldsymbol{P}}(\hat{\boldsymbol{d}}_s) &= \lambda_s \mathrm{tr}\varepsilon_s(\hat{\boldsymbol{d}}_s)\boldsymbol{I} + 2\mu_s \varepsilon_s(\hat{\boldsymbol{d}}_s), \\ \varepsilon_s(\hat{\boldsymbol{d}}s) &= \frac{1}{2}(\hat{\nabla}\hat{\boldsymbol{d}}_s + \hat{\nabla}^T \hat{\boldsymbol{d}}_s). \end{aligned}$$

μ_f is the fluid viscosity, while μ_s and λ_s are the Lamè coefficients of the solid. As we may see from system (3.1), the fluid problem is cast in the current fluid physical domain $\Omega_f(t)$, whereas the solid problem is already formulated in the reference solid configuration $\hat{\Omega}_s$. With the formalism introduced in the previous section 2.1.1, we

are nevertheless able to perform a pull–back of the fluid equations onto the fluid reference configuration $\hat{\Omega}_f$, thus obtaining a more homogeneous formulation of the whole coupled system. In order to perform a pull-back of the Navier–Stokes equation, we define:

$$\boldsymbol{F} := \hat{\nabla}\mathcal{A}_f, \qquad J := \det\boldsymbol{F},$$

the gradient of the ALE map and its determinant, respectively. With these quantities we are ready to present the strong form of the coupled problem in an ALE formulation: for every $t \in [0,T]$, find the fluid velocity $\hat{\boldsymbol{u}}_f(t)\colon \hat{\Omega}_f \mapsto \mathbb{R}^2$, the fluid pressure $\hat{p}_f(t)\colon \hat{\Omega}_f \mapsto \mathbb{R}$, the fluid displacement $\hat{\boldsymbol{d}}_f(t)\colon \hat{\Omega}_f \mapsto \mathbb{R}^2$ and the solid deformation $\hat{\boldsymbol{d}}_s(t)\colon \hat{\Omega}_s \mapsto \mathbb{R}^2$ such that:

$$\begin{cases} \rho_f J(\partial_t \hat{\boldsymbol{u}}_f + \hat{\nabla}\hat{\boldsymbol{u}}_f \boldsymbol{F}^{-1}(\hat{\boldsymbol{u}}_f - \partial_t \hat{\boldsymbol{d}}_f)) - \hat{\mathrm{div}}(J\hat{\sigma}_f(\hat{\boldsymbol{u}}_f, \hat{p}_f)\boldsymbol{F}^{-T}) = \boldsymbol{b}_f \quad \text{in } \hat{\Omega}_f \times (0,T], \\ \hat{\mathrm{div}}(J\boldsymbol{F}^{-1}\boldsymbol{u}_f) = 0 \quad \text{in } \hat{\Omega}_f \times (0,T], \\ -\hat{\Delta}\hat{\boldsymbol{d}}_f = 0 \quad \text{in } \hat{\Omega}_f \times (0,T], \\ \rho_s \partial_{tt}\hat{\boldsymbol{d}}_s - \hat{\mathrm{div}}\hat{\boldsymbol{P}}(\hat{\boldsymbol{d}}_s) = \boldsymbol{b}_s \quad \text{in } \hat{\Omega}_s \times (0,T], \end{cases} \tag{3.2}$$

The fluid tensor $\hat{\sigma}_f$ is the representation in the reference configuration of the Cauchy stress tensor:

$$\hat{\sigma}_f(\hat{\boldsymbol{u}}_f, \hat{p}_f) = \mu_f(\hat{\nabla}\hat{\boldsymbol{u}}_f \boldsymbol{F}^{-1} + \boldsymbol{F}^{-T}\hat{\nabla}^T\hat{\boldsymbol{u}}_f) - \hat{p}_f \boldsymbol{I}.$$

System (3.2) is completed by some suitable initial conditions, by the boundary conditions

$$\begin{cases} J\hat{\sigma}_f(\hat{\boldsymbol{u}}_f, \hat{p}_f)\boldsymbol{F}^{-T}\hat{\boldsymbol{n}} = -p_{in}(t)\hat{\boldsymbol{n}} \quad \text{on } \hat{\Gamma}_{in}, \\ J\hat{\sigma}_f(\hat{\boldsymbol{u}}_f, \hat{p}_f)\boldsymbol{F}^{-T}\hat{\boldsymbol{n}} = -p_{out}(t)\hat{\boldsymbol{n}} \quad \text{on } \hat{\Gamma}_{out} \\ \hat{\boldsymbol{d}}_s = 0 \quad \text{on } \hat{\Gamma}_s^D, \end{cases}$$

and by the following coupling conditions:

$$\begin{cases} \hat{\boldsymbol{d}}_f = \hat{\boldsymbol{d}}_s \quad \text{on } \hat{\Gamma}_{FSI} \\ \hat{\boldsymbol{u}}_f = \partial_t \boldsymbol{d}_s \quad \text{on } \hat{\Gamma}_{FSI}, \\ J\hat{\sigma}_f(\hat{\boldsymbol{u}}_f, \hat{p}_f)\boldsymbol{F}^{-T}\hat{\boldsymbol{n}}_f = -\hat{\boldsymbol{P}}(\hat{\boldsymbol{d}}_s)\hat{\boldsymbol{n}}_s \quad \text{on } \hat{\Gamma}_{FSI}. \end{cases} \tag{3.3}$$

In the previous equations the vector $\hat{\boldsymbol{n}}$ represents the normal vector to the inlet (or outlet) boundary in the reference configuration, whereas $\hat{\boldsymbol{n}}_f$ is the vector normal to the FSI interface $\hat{\Gamma}_{FSI}$, outgoing the fluid domain, and $\hat{\boldsymbol{n}}_s$ is the vector normal to the FSI interface $\hat{\Gamma}_{FSI}$, outgoing the solid domain.

Remark 3.2.1. The gradient and the divergence in equation (3.2) are computed with respect to the spatial coordinates in the reference configuration, namely $\hat{\boldsymbol{x}}$. Nevertheless, from now on, since everything will be formulated and computed on the reference configuration, in order to ease the exposition, we will drop the $\hat{}$ notation.

3.3 Offline computational phase

In this Section we are going to describe the offline phase of the partitioned procedure that we use to solve the FSI problem described in the previous Section. The algorithm is based on a Chorin-Temam projection scheme for the Navier–Stokes equation [80, 78], and we choose to treat the coupling conditions (3.3) in a semi–implicit way (see also [14, 8, 57]). We first apply a time stepping procedure to design the algorithm, and then we show the space discretization of the whole procedure.

3.3.1 High fidelity semi–implicit scheme

We present the offline phase of the partitioned procedure: we use an operator splitting approach, based on a Chorin-Temam projection scheme with pressure Poisson formulation. The coupling between fluid and solid problem is imposed with a Robin coupling.

Semi–implicit coupling

Let ΔT be a time–step: we discretize the time interval $[0, T]$ with an equispaced sampling $\{t_0, \dots, t_{N_T}\}$, where $t_i = i\Delta T$, for $i = 0, \dots, N_T$ and $N_T = \frac{T}{\Delta T}$. We discretize the partial derivative of a function f with a first backward difference: $D_t f^{i+1} = \frac{f^{i+1} - f^i}{\Delta T}$, and $D_{tt} f^{i+1} = D_t(D_t f^{i+1})$, where $f^{i+1} = f(t^{i+1})$. Our partitioned scheme reads as follows: for $i = 0, \dots, N_T$:

Extrapolation of the mesh displacement $\boldsymbol{d}_f$:

find $\boldsymbol{d}_f^{i+1} \colon \Omega_f \mapsto \mathbb{R}^2$ such that:

$$\begin{cases} -\Delta \boldsymbol{d}_f^{i+1} = 0 \quad \text{in } \Omega_f, \\ \boldsymbol{d}_f^{i+1} = \boldsymbol{d}_s^i \quad \text{on } \Gamma_{FSI}. \end{cases} \tag{3.4}$$

Fluid explicit step:

find $\boldsymbol{u}_f^{i+1} \colon \Omega_f \mapsto \mathbb{R}^2$ such that:

$$\begin{cases} J\rho_f \Big(\dfrac{\boldsymbol{u}_f^{i+1} - \boldsymbol{u}_f^i}{\Delta T} + \nabla \boldsymbol{u}_f^{i+1} \boldsymbol{F}^{-1} (\boldsymbol{u}_f^{i+1} - D_t \boldsymbol{d}_f^{i+1}) \Big) - \\ \qquad\qquad - \mu_f \mathrm{div}(J\varepsilon(\boldsymbol{u}_f^{i+1}) \boldsymbol{F}^{-T}) + J\boldsymbol{F}^{-T} \nabla p_f^i = 0 \quad \text{in } \Omega_f, \\ \boldsymbol{u}_f^{i+1} = D_t \boldsymbol{d}_f^{i+1} \quad \text{on } \Gamma_{FSI}, \end{cases} \tag{3.5}$$

Implicit step:

1. **fluid projection substep (pressure Poisson formulation):** find $p_f^{i+1} \colon \Omega_f \mapsto \mathbb{R}^2$ such that:

$$\begin{cases} -\frac{\rho_f}{\Delta t} \mathrm{div}(J\boldsymbol{F}^{-1} \boldsymbol{u}_f^{i+1}) = -\mathrm{div}(J\boldsymbol{F}^{-1} \boldsymbol{F}^{-T} \nabla p_f^{i+1}) \quad \text{in } \Omega_f, \\ -\boldsymbol{F}^{-T} \nabla p_f^{i+1} \cdot J\boldsymbol{F}^{-T} \boldsymbol{n}_f = \rho_f D_{tt} \boldsymbol{d}_s^{i+1} \cdot J\boldsymbol{F}^{-T} \boldsymbol{n}_f \quad \text{on } \Gamma_{FSI}, \end{cases} \tag{3.6}$$

   subject to the boundary conditions:

$$\begin{cases} p_f^{i+1} = p^{in}(t^{i+1}) \quad \text{on } \Gamma_{in} \\ p_f^{i+1} = 0 \quad \text{on } \Gamma_{out}, \end{cases} \tag{3.7}$$

2. **structure projection substep:** find $\boldsymbol{d}_s^{i+1}\colon \Omega_s \mapsto \mathbb{R}^2$ such that:

$$\begin{cases} \rho_s D_{tt}\boldsymbol{d}_s^{i+1} - \mathrm{div}\boldsymbol{P}(\boldsymbol{d}_s^{i+1}) = 0 \quad \text{in } \Omega_s, \\ J\sigma_f(\boldsymbol{u}_f^{i+1}, p_f^{i+1})\boldsymbol{F}^{-T}\boldsymbol{n}_f = -\boldsymbol{P}(\boldsymbol{d}_s^{i+1})\boldsymbol{n}_s \quad \text{on} \Gamma_{FSI}. \end{cases} \tag{3.8}$$

subject to the boundary condition $\boldsymbol{d}_s^{i+1} = 0$ on Γ_s^D.
Remark 3.3.1. In the implicit step (3.6) we have chosen a pressure Poisson formulation; an alternative is to use a Darcy formulation, which is defined as follows: find p_f^{i+1} and $\tilde{\boldsymbol{u}}_f^{i+1}$ such that:

$$\begin{cases} \rho_f J\frac{\tilde{\boldsymbol{u}}_f^{i+1} - \boldsymbol{u}_f^{i+1}}{\Delta T} + J\boldsymbol{F}^{-T}\nabla p_f^{i+1} = 0 \quad \text{in } \Omega_f, \\ \mathrm{div}(J\boldsymbol{F}^{-1}\tilde{\boldsymbol{u}}_f^{i+1}) = 0 \quad \text{in } \Omega_f. \end{cases}$$

However, we choose to employ a Poisson formulation throughout this Chapter, for the sake of a more efficient reduced order model, since the Darcy formulation requires the introduction of an additional unknown $\tilde{\boldsymbol{u}}_f$, which translates in a larger system, comprised of both velocity and pressure, at the implicit step.

In order to enhance the stability of the projection scheme, we employ a Robin–Neumann coupling, as proposed in [5, 14]; for other references on this kind of coupling, we refer to [6, 59]. We thus replace condition $(3.6)_2$ with the following:

$$\alpha_{ROB}p^{i+1} + \boldsymbol{F}^{-T}\nabla p^{i+1}\cdot J\boldsymbol{F}^{-T}\boldsymbol{n}_f = \alpha_{ROB}p^{i+1,\star} - \rho_f D_{tt}\boldsymbol{d}_s^{i+1,\star}\cdot J\boldsymbol{F}^{-T}\boldsymbol{n}_f. \tag{3.9}$$

In equation (3.9), $p^{i+1,\star}$ and $\boldsymbol{d}_s^{i+1,\star}$ are suitable extrapolations of the fluid pressure and the solid displacement, respectively; we show in the next paragraph which kind of extrapolation we use. The constant α_{ROB} is defined as $\alpha_{ROB} = \frac{\rho_f}{z_p \Delta T}$ where z_p is called the *solid impedance*:

$$\begin{aligned} z_p &= \rho_s c_p, \\ c_p &= \sqrt{\frac{\lambda_s + 2\mu_s}{\rho_s}}. \end{aligned}$$

In Appendix A we show more in detail how to compute the coupling condition (3.9), starting from the idea proposed in [16].

Space discretization of the semi–implicit procedure

We now present the space discretized version of the algorithm introduced. We define the following function spaces for the fluid:

$$V(\Omega_f) := [H^1(\Omega_f)]^2, \quad E^f(\Omega_f) := [H^1(\Omega_f)]^2, \quad Q(\Omega_f) := L^2(\Omega_f),$$

endowed with the H^1 norm ($V(\Omega_f)$ and $E^f(\Omega_f)$) and the L^2 norm respectively, and the function space for the solid: $E^s(\Omega_s) = [H^1(\Omega_s)]^2$, endowed with the H^1 norm. We discretize in space the FSI problem, using second order Lagrange Finite Elements for the fluid velocity, the fluid displacement and the solid displacement, resulting in the discrete spaces $V_h \subset V$, $E_h^f \subset E^f$ and $E_h^s \subset E^s$, while the fluid pressure is discretized with first order Lagrange Finite Elements, resulting in the discrete space $Q_h \subset Q$. The non–homogeneous boundary condition $(3.7)_1$ can be easily treated without the use of

a Lagrange multiplier, by introducing a lifting function ℓ^{i+1} such that $\ell^{i+1} = p_{in}(t^{i+1})$ on Γ_{in} and $\ell^{i+1} = 0$ on Γ_{out}; we refer to [12] for more details. By introducing the homogenized pressure $p_f^{0,i+1} := p_f^{0,i+1} - \ell^{i+1}$, we can conclude now that $p_f^{0,i+1} \in Q_h^0$, where $Q_h^0 = \{q_h \in Q_h \colon q_h = 0 \quad \text{on } \Gamma_{in}\}$. The discretized version of the semi–implicit procedure reads as follows: for $i = 0, \dots, N_T$,

Extrapolation of the mesh displacement:

find $\boldsymbol{d}_{f,h}^{i+1} \in E_h^f$ such that $\forall \boldsymbol{e}_{f,h} \in E_h^f$:

$$\begin{cases} \int_{\Omega_f} \boldsymbol{d}_{f,h}^{i+1} \cdot \boldsymbol{e}_{f,h}\, dx = 0 \quad \text{in } \Omega_f, \\ \boldsymbol{d}_{f,h}^{i+1} = \boldsymbol{d}_{s,h}^{i} \quad \text{on } \Gamma_{FSI}. \end{cases} \tag{3.10}$$

Fluid explicit step:

find $\boldsymbol{u}_{f,h}^{i+1} \in V_h$ such that $\forall \boldsymbol{v}_h \in V_h$:

$$\begin{cases} \rho_f \displaystyle\int_{\Omega_f} J\Big(\frac{\boldsymbol{u}_{f,h}^{i+1} - \boldsymbol{u}_{f,h}^{i}}{\Delta T}\Big) \cdot \boldsymbol{v}_h\, dx + \rho_f \int_{\Omega_f} J(\nabla \boldsymbol{u}_{f,h}^{i+1} \boldsymbol{F}^{-1}(\boldsymbol{u}_{f,h}^{i+1} - D_t \boldsymbol{d}_{f,h}^{i+1})) \cdot \boldsymbol{v}_h\, dx \\ \quad + \mu_f \displaystyle\int_{\Omega_f} J\varepsilon(\boldsymbol{u}_{f,h}^{i+1}) \boldsymbol{F}^{-T} : \nabla \boldsymbol{v}_h\, dx + \int_{\Omega_f} J\boldsymbol{F}^{-T} \nabla p_{f,h}^{i} \cdot \boldsymbol{v}_h\, dx = 0 \quad \text{in } \Omega_f, \\ \boldsymbol{u}_{f,h}^{i+1} = D_t \boldsymbol{d}_{f,h}^{i+1} \quad \text{on } \Gamma_{FSI}, \end{cases} \tag{3.11}$$

Implicit step:

for any $j = 0, \dots$ until convergence:

1. **fluid projection substep (pressure Poisson formulation):** find $p_{f,h}^{0,i+1,j+1} \in Q_h^0$ such that $\forall q_h \in Q_h^0$:

$$\begin{aligned} &- \frac{\rho_f}{\Delta T} \int_{\Omega_f} \operatorname{div}(JF^{-1} \boldsymbol{u}_{f,h}^{i+1}) q_h\, dx - \rho_f \int_{\Gamma_{FSI}} (D_{tt} \boldsymbol{d}_{s,h}^{i+1,j}) \cdot J\boldsymbol{F}^{-T} \boldsymbol{n}_f q_h\, ds \\ &+ \alpha_{ROB} \int_{\Gamma_{FSI}} p_{f,h}^{i+1,j} q_h\, ds - \alpha_{ROB} \int_{\Gamma_{FSI}} \ell^{i+1} q_h\, ds \\ &- \int_{\Omega_f} J\boldsymbol{F}^{-T} \nabla \ell^{i+1} \cdot \boldsymbol{F}^{-T} \nabla q_h\, dx = \alpha_{ROB} \int_{\Gamma_{FSI}} p_{f,h}^{0,i+1,j+1} q_h\, ds + \\ &+ \int_{\Omega_f} J\boldsymbol{F}^{-T} \nabla p_{f,h}^{0,i+1,j+1} \cdot \boldsymbol{F}^{-T} \nabla q_h\, dx \end{aligned}$$

2. **structure projection substep:** find $\boldsymbol{d}_{s,h}^{i+1,j+1} \in E_h^s$ such that $\forall \boldsymbol{e}_{s,h} \in E_h^s$:

$$\begin{aligned} \rho_s \int_{\Omega_s} D_{tt} \boldsymbol{d}_{s,h}^{i+1,j+1} \cdot \boldsymbol{e}_{s,h}\, dx + \int_{\Omega_s} \boldsymbol{P}(\boldsymbol{d}_{s,h}^{i+1,j+1}) : \nabla \boldsymbol{e}_{s,h}\, dx = \\ = - \int_{\Gamma_{FSI}} J\sigma_f(\boldsymbol{u}_{f,h}^{i+1}, p_{f,h}^{i+1,j+1}) \boldsymbol{F}^{-T} \boldsymbol{n}_f \cdot \boldsymbol{e}_{s,h}\, dx \end{aligned}$$

subject to the boundary condition $\boldsymbol{d}_{s,h}^{i+1,j+1} = 0$ on Γ_D^s

We iterate between the two implicit substeps, until a convergence criteria is satisfied; we choose as stopping criteria a relative error on the increments of the pressure and the solid displacement, namely:

$$\max\Big(\frac{||p_{f,h}^{i+1,j+1} - p_{f,h}^{i+1,j}||_{Q_h}}{||p_{f,h}^{i+1,j+1}||_{Q_h}}; \frac{||\boldsymbol{d}_{s,h}^{i+1,j+1} - \boldsymbol{d}_{s,h}^{i+1,j}||_{E_h^s}}{||\boldsymbol{d}_{s,h}^{i+1,j+1}||_{E_h^s}}\Big) < \varepsilon,$$

where ε is a fixed tolerance.
In the pressure Poisson formulation, to impose the Robin coupling condition, we have chosen the pressure at the previous implicit iteration, namely $p_f^{i+1,j}$, as an extrapolation for the fluid pressure, and the same goes for the extrapolation of the structure displacement.

3.3.2 POD and reduced basis generation

For the generation of the reduced basis for the fluid velocity $\boldsymbol{u}_f$ and the fluid displacement $\boldsymbol{d}_f$ we pursue here the idea that was first proposed in [14]. For the homogenized fluid pressure p_f^0 and for the solid displacement $\boldsymbol{d}_s$ we employ a standard POD.

Change of variable for the fluid velocity

The main idea here is to introduce a change of variable in the fluid problem, in order to transform condition $(3.20)_2$ into a homogeneous boundary condition. The motivation of this choice is that, to impose condition $(3.20)_2$, we could use a Lagrange multiplier λ, but unfortunately introducing a new variable leads to an increased dimension of the system to be solved in the online phase. Therefore, in order to avoid this and in order to design a more efficient reduced method, we choose to transform the non–homogeneous coupling condition into a homogeneous one. In order to do this, we define a new variable $\boldsymbol{z}_{f,h}^{i+1}$:

$$\boldsymbol{z}_{f,h}^{i+1} := \boldsymbol{u}_{f,h}^{i+1} - D_t \boldsymbol{d}_{f,h}^{i+1}.$$

With this change of variable, equation $(3.20)_2$ is equivalent to the homogeneous boundary condition for the new variable:

$$\boldsymbol{z}_{f,h}^{i+1} = 0 \quad \text{on } \Gamma_{FSI},$$

for which no imposition by means of Lagrange multiplier is needed. Therefore, during the offline phase of the scheme, at every iteration $i+1$, after we have computed the velocity $\boldsymbol{u}_{f,h}^{i+1}$, we compute the change of variable $\boldsymbol{z}_{f,h}^{i+1}$. We then consider the following snapshots matrix:

$$\boldsymbol{\mathcal{S}}_z = [\underline{\boldsymbol{z}}_{f,h}^1, \dots, \underline{\boldsymbol{z}}_{f,h}^{N_T}] \in \mathbb{R}^{\mathcal{N}_u^h \times N_T},$$

where $\mathcal{N}_u^h = \dim V_h$ and we use the underline notation to denote the vector consisting of the FE degrees of freedom corresponding to each solution. We then apply a POD to the snapshots matrix $\boldsymbol{\mathcal{S}}_z$ and we retain the first $N_{\boldsymbol{z}}$ POD modes $\Phi_{\boldsymbol{z}}^1, \dots, \Phi_{\boldsymbol{z}}^{N_{\boldsymbol{z}}}$. We therefore have the reduced space:

$$V^N := \text{span}\{\Phi_{\boldsymbol{z}}^k\}_{k=1}^{N_{\boldsymbol{z}}},$$

and now it is clear that, since every $\Phi_{\boldsymbol{z}}^k$ satisfies the condition $\Phi_{\boldsymbol{z}}^k = 0$ on Γ_{FSI}, then also every element of V^N will satisfy the same condition.

Harmonic extension of the fluid displacement

In order to generate the reduced basis for the fluid displacement $\boldsymbol{d}_f$, we pursue again the idea presented in [14]. Therefore, we start by generating the snapshots matrix related to the solid displacement:

$$\boldsymbol{\mathcal{S}}_{d_s} = [\underline{\boldsymbol{d}}_{s,h}^1, \dots, \underline{\boldsymbol{d}}_{s,h}^{N_T}] \in \mathbb{R}^{\mathcal{N}_{\boldsymbol{d}_s}^h \times N_T},$$

where $\mathcal{N}_{\boldsymbol{d}_s}^h = \dim E_h^s$ and again the underline notation denotes the vector of the FE degrees of freedom corresponding to each solution of the solid displacement. We then apply a POD to the snapshots matrix and retain the first $N_{\boldsymbol{d}_s}$ POD modes $\Phi_{\boldsymbol{d}_s}^1, \dots, \Phi_{\boldsymbol{d}_s}^{N_{\boldsymbol{d}_s}}$, thus defining the reduced space for the solid problem:

$$E_N^s := \text{span}\{\Phi_{\boldsymbol{d}_s}^k\}_{k=1}^{N_{\boldsymbol{d}_s}}.$$

We then employ an harmonic extension of each one of the reduced basis $\Phi_{\boldsymbol{d}_s}^k$ to the fluid domain, thus obtaining the functions $\Phi_{\boldsymbol{d}_f}^k$ such that:

$$\begin{cases} -\Delta\Phi_{\boldsymbol{d}_f}^k = 0 \quad \text{in } \Omega_f, \\ \Phi_{\boldsymbol{d}_f}^k = \Phi_{\boldsymbol{d}_s}^k \quad \text{on } \Gamma_{FSI}. \end{cases}$$

We can then define the reduced space for the fluid displacement:

$$E_N^f := \text{span}\{\Phi_{\boldsymbol{d}_f}^k\}_{k=1}^{N_{\boldsymbol{d}_s}}.$$

The reason for defining the basis functions for $\boldsymbol{d}_f$ in such a way, instead of employing a standard POD on the set of snapshots for the fluid displacement computed in the offline phase relies in the fact that we want to avoid the introduction of another Lagrange multiplier to impose the non–homogeneous boundary condition $(3.19)_2$. With our method, we avoid to solve the reduced system related to (3.19): indeed, instead of solving an harmonic extension problem at every time–step in the online phase, we solve *once and for all* $N_{\boldsymbol{d}_s}$ harmonic extension problems in the expensive offline phase. Then, during the online phase, the reduced fluid displacement will be computed just as a linear combination of the basis $\Phi_{\boldsymbol{d}_f}^i$, with coefficients that are the coefficients of the reduced solid displacement at the previous time–step. We will see in the next Section the final formulation of the online phase of the algorithm.

3.4 Online computational phase

We are now ready to present the online formulation of the partitioned procedure. For every $i = 0, \dots, N_T$, we introduce the reduced functions $\boldsymbol{z}_{f,N}^{i+1}$, $p_{f,N}^{0,i+1}$, $\boldsymbol{d}_{s,N}^{i+1}$ of the form:

$$\boldsymbol{z}_{f,N}^{i+1} = \sum_{k=1}^{N_{\boldsymbol{z}_f}} \underline{\mathbf{z}}_k^{i+1} \boldsymbol{\Phi}_{\boldsymbol{z}_f}^k, \tag{3.12}$$

$$p_{f,N}^{0,i+1} = \sum_{k=1}^{N_p} \underline{p}_k^{0,i+1} \Phi_p^k, \tag{3.13}$$

$$\boldsymbol{d}_{s,N}^{i+1} = \sum_{k=1}^{N_{\boldsymbol{d}_s}} \underline{\mathbf{d}}_k^{i+1} \boldsymbol{\Phi}_{\boldsymbol{d}_s}^k. \tag{3.14}$$

Then the online phase of the partitioned procedure reads as follows:

Mesh displacement:

let $\boldsymbol{d}_{f,N}^{i+1}$ be defined by the reduced solid displacement at the previous time–step:

$$\boldsymbol{d}_{f,N}^{i+1} = \sum_{k=1}^{N_{\boldsymbol{d}_s}} \underline{\mathbf{d}}_k^{i} \boldsymbol{\Phi}_{\boldsymbol{d}_f}^k; \tag{3.15}$$

Fluid explicit step (with change of variable):

find $z_{f,N}^{i+1} \in V_N$ such that $\forall \boldsymbol{v}_N \in V_N$:

$$\begin{aligned}
&\rho_f \int_{\Omega_f} J\Big(\frac{\boldsymbol{z}_{f,N}^{i+1} - \boldsymbol{u}_{f,N}^{i}}{\Delta T}\Big) \cdot \boldsymbol{v}_N \, dx + \rho_f \int_{\Omega_f} J(\nabla(\boldsymbol{z}_{f,N}^{i+1} + D_t \boldsymbol{d}_{f,N}^{i+1}) \boldsymbol{F}^{-1} \boldsymbol{z}_{f,N}^{i+1}) \cdot \boldsymbol{v}_N \, dx \\
&+ \mu_f \int_{\Omega_f} J\varepsilon(\boldsymbol{z}_{f,N}^{i+1}) \boldsymbol{F}^{-T} : \nabla \boldsymbol{v}_N \, dx + \int_{\Omega_f} J \boldsymbol{F}^{-T} \nabla p_{f,N}^{i} \cdot \boldsymbol{v}_h \, dx = \\
&- \rho_f \int_{\Omega_f} J\Big(\frac{D_t \boldsymbol{d}_{f,N}^{i+1}}{\Delta T}\Big) \cdot \boldsymbol{v}_N \, dx - \mu_f \int_{\Omega_f} J\varepsilon(D_t \boldsymbol{d}_{f,N}^{i+1}) \boldsymbol{F}^{-T} : \nabla \boldsymbol{v}_N \, dx \quad \text{in } \Omega_f.
\end{aligned}$$

We then restore the reduced fluid velocity: $\boldsymbol{u}_{f,N}^{i+1} = \boldsymbol{z}_{f,N}^{i+1} + D_t \boldsymbol{d}_{f,N}^{i+1}$.

Implicit step:

for any $j = 0, \dots$ until convergence:

TABLE 3.1: Values for the implementation of the offline phase.

Physical constants	**Value**
ρ_f	1 g/cm^3
μ_f	0.035 Poise
ρ_s	1.1 g/cm^3
μ_s	100000
λ_s	8000
Discretization details	**Value**
FE displacement order	2
FE velocity order	2
FE pressure order	1

1. **fluid projection substep:** find $p_{f,N}^{0,i+1,j+1} \in Q_N^0$ such that $\forall q_N \in Q_N^0$:

$$
\begin{aligned}
&-\frac{\rho_f}{\Delta T}\int_{\Omega_f} \mathrm{div}(JF^{-1}\boldsymbol{u}_{f,N}^{i+1}) q_N \, dx - \rho_f \int_{\Gamma_{FSI}} (D_{tt}\boldsymbol{d}_{s,N}^{i+1,j}) \cdot J\boldsymbol{F}^{-T}\boldsymbol{n}_f q_N \, ds \\
&+ \alpha_{ROB}\int_{\Gamma_{FSI}} p_{f,N}^{i+1,j} q_N \, ds - \alpha_{ROB}\int_{\Gamma_{FSI}} \ell^{i+1} q_N \, ds \\
&-\int_{\Omega_f} J\boldsymbol{F}^{-T}\nabla \ell^{i+1} \cdot \boldsymbol{F}^{-T}\nabla q_N \, dx = \alpha_{ROB}\int_{\Gamma_{FSI}} p_{f,N}^{i+1,j+1} q_N \, ds \\
&+\int_{\Omega_f} J\boldsymbol{F}^{-T}\nabla p_{f,N}^{i+1,j+1} \cdot \boldsymbol{F}^{-T}\nabla q_N \, dx;
\end{aligned}
$$

   we then recover the reduced fluid pressure $p_{f,N}^{i+1,j+1} = p_{f,N}^{0,i+1,j+1} + \ell^{i+1}$.

2. **structure projection substep:** find $\boldsymbol{d}_{s,N}^{i+1,j+1} \in E_N^s$ such that $\forall \boldsymbol{e}_s \in E_N^s$:

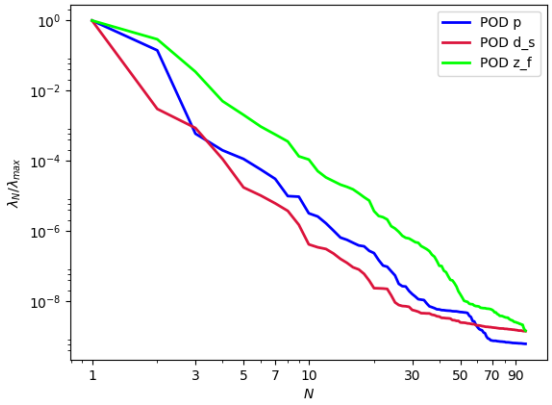
$$
\begin{aligned}
\rho_s \int_{\Omega_s} D_{tt}\boldsymbol{d}_{s,N}^{i+1,j+1} \cdot \boldsymbol{e}_N \, dx + \int_{\Omega_s} \boldsymbol{P}(\boldsymbol{d}_{s,N}^{i+1,j+1}) : \nabla \boldsymbol{e}_N \, dx = \\
= -\int_{\Omega_s} J\sigma_f(\boldsymbol{u}_{f,N}^{i+1}, p_{f,N}^{i+1,j+1})\boldsymbol{F}^{-T}\boldsymbol{n}_f \cdot \boldsymbol{e}_N \, dx
\end{aligned}
$$

3.5 Numerical results

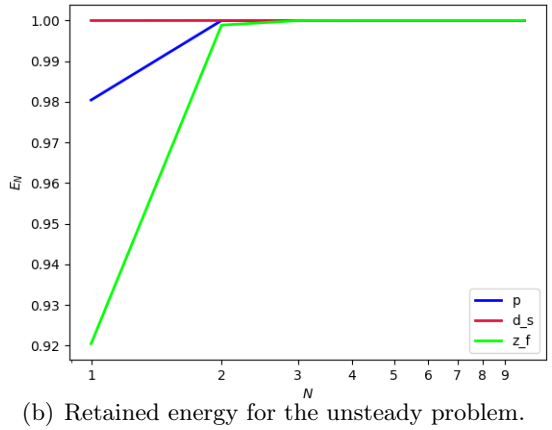
We now present some numerical results obtained with the semi–implicit scheme. For our simulation we used a time–step $\Delta T = 10^{-4}$, and a final time $T = 0.2$ s, for a total of $N_T = 2000$ iterations. The reference domain Ω is presented in Figure 3.1: the fluid domain is 2.5 cm height, and it is 10 cm long; the leaflets are situated 1 cm downstream the inlet boundary, they are 0.2 cm thick and 1.1 cm height.

The values of the physical constants used in the simulation are reported in Table 3.1. A pressure impulse $p_{in}(t)$ is applied at the inlet boundary, and after some time this impulse becomes constant:

$$
p_{in}(t) = \begin{cases} 5 - 5\cos\left(\frac{2\pi t}{T_{in}}\right) & \text{for } t \leq 0.1s, \\ 5 \qquad \text{for } t > 0.1s, \end{cases}
$$



(a) POD eigenvalues for the unsteady problem.



(b) Retained energy for the unsteady problem.

FIGURE 3.2: POD eigenvalues and retained energy for the unsteady, non parametric leaflets problem.

where $T_{in} = 0.4s$. We fix a tolerance of $\varepsilon = 10^{-6}$ as a stopping criterion for the subiterations between the pressure Poisson problem and the solid problem.

Since we do not consider the top and the bottom walls of the fluid domain to be deformable, we impose a homogeneous boundary condition for the fluid velocity on these walls.

Figure 3.2a shows the rate of decay of the first 100 eigenvalues associated with three unknowns of the problem, namely the change of variable for the fluid velocity $\boldsymbol{z}_f$, the pressure p_f and the solid displacement $\boldsymbol{d}_s$. It can be noticed that the rate of decay of the eigenvalues for the pressure and for the fluid change of variable is slower than the rate of decay of the eigenvalues of the solid displacement. Moreover, in Figure 3.2b we can notice that the first mode of the solid displacement retains 2% more energy compared to the first mode of the pressure, and 8% more energy with respect to the first mode of $\boldsymbol{z}_f$, which is the one that retains less energy. Figures 3.3 and 3.4 show two representative reduced order solutions: the fluid velocity and the solid displacement, respectively; as we can see from Figure 3.4, the reduced order model shows a good capability also in reproducing very small deformations. Figure 3.5 shows that, with $N = 20$ basis functions for each component of the solution, we have a good approximation error behaviour over time.

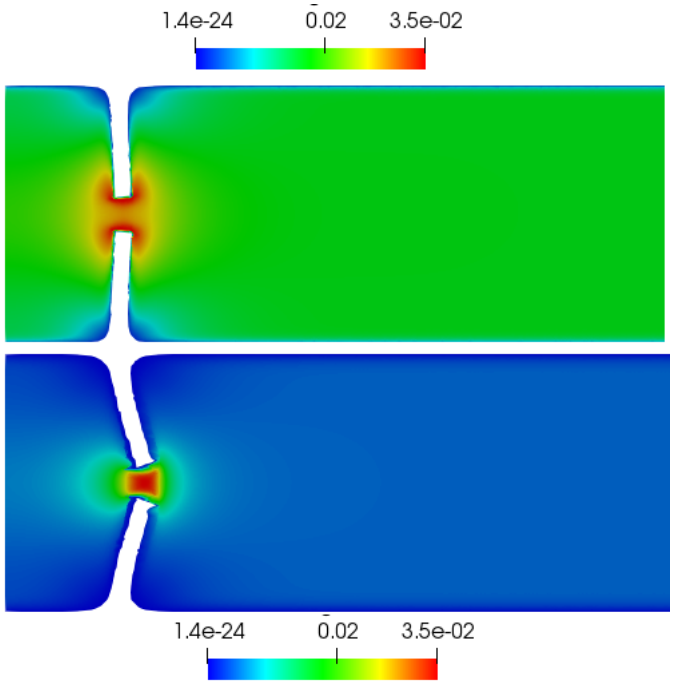


FIGURE 3.3: Reduced velocity $\boldsymbol{u}_{f,N}$ at time–step $t = 0.1s$ (top) and at time–step $t = 0.2s$ (bottom). The velocity has been obtained with $N_{z_f} = 20$ reduced basis.

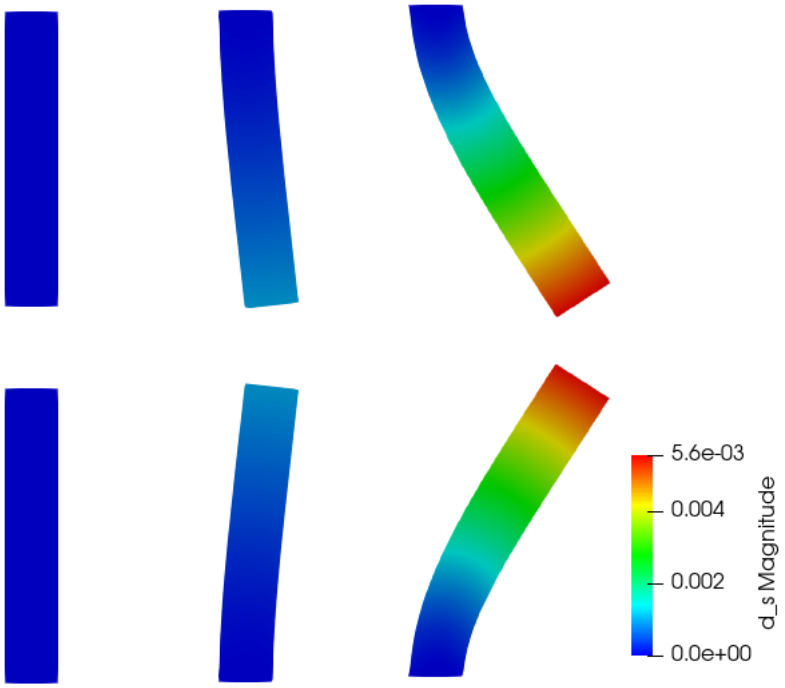


FIGURE 3.4: Reduced solid displacement $\boldsymbol{d}_{s,N}$ at time–step $t = 0.02s$ (left), $t = 0.1s$ (center) and at time–step $t = 0.2s$ (right). The displacement has been obtained with $N_d = 20$ reduced basis. The displacement has been magnified for visualization purposes.

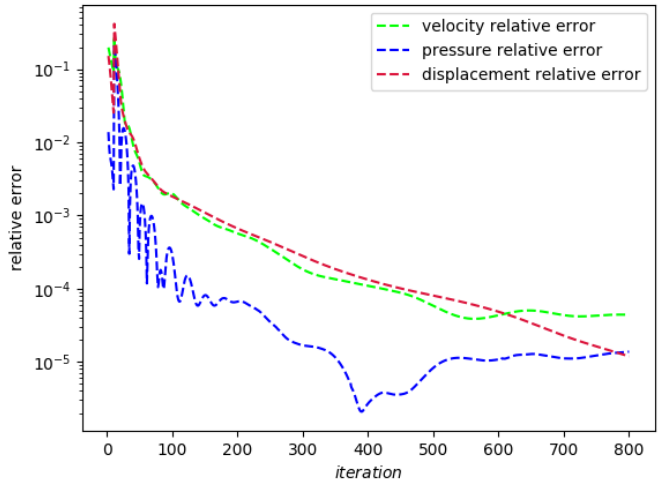


FIGURE 3.5: Error analysis: relative error behaviour, as a function of time.

3.6 Problem formulation: parameter–dependent geometry

In this Section we are going to present a test case that is similar to the one previously introduced, the difference being now the presence of a geometrical parameter μ, that represents the length of the leaflets.

3.6.1 ALE formulation in the presence of shape parametrization

Let us denote by $\Omega(t;\mu) := \Omega_f(t;\mu) \cup \Omega_s(t;\mu)$ the current physical domain: we now have a time dependence, and a parameter dependence. We introduce the time–independent *intermediate configuration* $\tilde{\Omega}(\mu) := \tilde{\Omega}_f(\mu) \cup \tilde{\Omega}_s(\mu)$, where we are considering the reference configuration of both physics, still taking into account the parameter dependence. Finally, we have the time–independent, parameter–independent *reference configuration* $\hat{\Omega} := \hat{\Omega}_f \cup \hat{\Omega}_s$.

We call T the shape parametrization map; for every $\mu \in \mathcal{P} \subset \mathbb{R}$, where $\mathcal{P} = [\mu_{min}, \mu_{max}]$ is the domain of the geometrical parameter, we have a map T_μ defined as follows:

$$\begin{aligned} T_\mu \colon \hat{\Omega} &\mapsto \tilde{\Omega}(\mu) \\ \hat{x} &\mapsto \tilde{x} = T_\mu(\hat{x}). \end{aligned}$$

We then have the ALE map $\mathcal{A}_f(t;\mu)$, already introduced in Section 2.1.1, which is now a map from the current parametrized fluid configuration $\Omega_f(t;\mu)$ and the intermediate fluid configuration $\tilde{\Omega}_f(\mu)$:

$$\begin{aligned} \mathcal{A}_f(t;\mu) \colon \tilde{\Omega}_f(\mu) &\mapsto \Omega_f(t;\mu) \\ \tilde{x} &\mapsto x = \tilde{x} + \tilde{\boldsymbol{d}}_f(\tilde{x};t,\mu), \end{aligned}$$

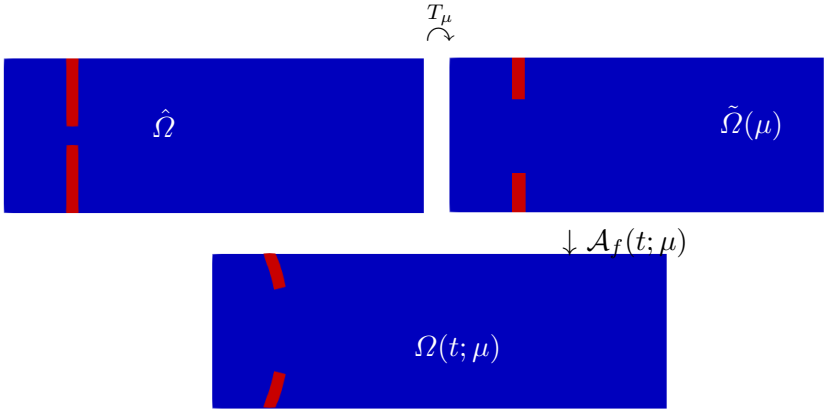


FIGURE 3.6: Domains: reference configuration $\hat{\Omega}$ (top left), parametrized reference configuration $\tilde{\Omega}(\mu)$ (top right), and original configuration $\Omega(t;\mu)$ (bottom).

where $\tilde{\boldsymbol{d}}_f$ is the mesh displacement already defined in Section 2.1.1.
Let us define the gradients and the determinants of the deformation maps:

$$\begin{aligned} \boldsymbol{G}(\hat{x};\mu) &= \hat{\nabla} T_\mu(\hat{x}), \qquad & K(\hat{x};\mu) &= \det \boldsymbol{G}(\hat{x};\mu), \\ \tilde{\boldsymbol{F}}(\tilde{x};\mu) &= \tilde{\mathrm{Id}} + \tilde{\nabla}\tilde{\boldsymbol{d}}_f, \qquad & \tilde{J}(\tilde{x};\mu) &= \det \tilde{\boldsymbol{F}}. \end{aligned} \tag{3.16}$$

We can pull-back the gradient $\tilde{\boldsymbol{F}}(\tilde{x};\mu)$ to the reference domain $\hat{\Omega}_f$, and we obtain $\boldsymbol{F}(\hat{x};\mu) = \mathrm{Id} + \hat{\nabla}\hat{\boldsymbol{d}}_f \boldsymbol{G}^{-1}(\hat{x},\mu)$. With this notation, we can conclude that the gradient of the deformation map from the reference configuration to the current configuration is given by $\boldsymbol{F}(\hat{x},\mu)\boldsymbol{G}(\hat{x},\mu)$; for the sake of the simplicity of the notation let us denote by $\boldsymbol{F}_\mu$ and $\boldsymbol{G}_\mu$ the gradients $\boldsymbol{F}(\hat{x},\mu)$ and $\boldsymbol{G}(\hat{x},\mu)$ respectively. We are now ready to state the strong form of the problem of interest.

3.6.2 Problem formulation

The strong form of the parametrized FSI problem reads as follows: find the fluid velocity $\boldsymbol{u}_f(t;\mu)\colon \Omega_f(t;\mu) \mapsto \mathbb{R}^2$, the fluid pressure $p_f(t;\mu)\colon \Omega_f(t;\mu) \mapsto \mathbb{R}$, the mesh displacement $\tilde{\boldsymbol{d}}_f(t;\mu)\colon \tilde{\Omega}_f(\mu) \mapsto \mathbb{R}^2$ and the solid displacement $\tilde{\boldsymbol{d}}_s(t;\mu)\colon \tilde{\Omega}_s(\mu) \mapsto \mathbb{R}^2$ such that:

$$\begin{cases} -\tilde{\Delta}\tilde{\boldsymbol{d}}_f = 0 \quad \text{in } \tilde{\Omega}_f(\mu) \times [0,T], \\ \tilde{\boldsymbol{d}}_f = \tilde{\boldsymbol{d}}_s \quad \text{on } \tilde{\Gamma}_{FSI} \times [0,T], \end{cases}$$

and

$$\begin{cases} \rho_f \partial_t \boldsymbol{u}_f|_{\tilde{x}} + \rho_f(\boldsymbol{u}_f - \partial_t \boldsymbol{d}_f|_{\tilde{x}}) \cdot \nabla \boldsymbol{u}_f - \mathrm{div}\sigma_f(\boldsymbol{u}_f, p_f) = \boldsymbol{b}_f \quad \text{in } \Omega_f(t;\mu) \times [0,T], \\ \mathrm{div}\boldsymbol{u}_f = 0 \quad \text{in } \Omega_f(t;\mu) \times [0,T], \\ \rho_s \partial_{tt} \tilde{\boldsymbol{d}}_s - \tilde{\mathrm{div}}\tilde{\boldsymbol{P}}(\tilde{\boldsymbol{d}}_s) = \tilde{\boldsymbol{b}}_s \quad \text{in } \tilde{\Omega}_s(\mu) \times [0,T]. \end{cases}$$

Here we notice that, again, the fluid problem is formulated in the current parametrized configuration $\Omega_f(t;\mu)$, whereas the solid problem is formulated in the parametrized reference configuration $\tilde{\Omega}_s(\mu)$. The quantity $\partial_t \boldsymbol{u}_f|_{\tilde{x}}$ represents the ALE time derivative: $\partial_t \boldsymbol{u}_f(\boldsymbol{x},t;\mu)|_{\tilde{x}} = \partial_t \tilde{\boldsymbol{u}}_f(\tilde{\boldsymbol{x}},t;\mu)$. Again, σ_f is the fluid Cauchy stress tensor, and $\tilde{\boldsymbol{P}}$ is the second Piola–Kirchoff stress tensor: their definition has been given in Section 3.2. The previous system is completed by some suitable initial conditions, by some

boundary conditions and by the following coupling conditions:

$$\begin{cases} \boldsymbol{d}_f = \boldsymbol{d}_s \quad \text{on } \Gamma_{FSI}(t;\mu) \\ \boldsymbol{u}_f = \partial_t \boldsymbol{d}_s \quad \text{on } \Gamma_{FSI}(t;\mu), \\ \tilde{J}\tilde{\sigma}_f(\tilde{\boldsymbol{u}}_f, \tilde{p}_f)\tilde{\boldsymbol{F}}^{-T}\tilde{\boldsymbol{n}}_f = -\tilde{\boldsymbol{P}}(\tilde{\boldsymbol{d}}_s)\tilde{\boldsymbol{n}}_s \quad \text{on } \tilde{\Gamma}_{FSI}(\mu), \end{cases}$$

being $\tilde{\sigma}_f$ the Cauchy stress tensor in the parametrized intermediate fluid domain $\tilde{\Omega}_f(\mu)$:

$$\tilde{\sigma}_f(\tilde{\boldsymbol{u}}_f, \tilde{p}_f) = \mu_f(\tilde{\nabla}\tilde{\boldsymbol{u}}_f\tilde{\boldsymbol{F}}^{-1}(\mu) + \tilde{\boldsymbol{F}}^{-T}(\mu)\tilde{\nabla}^T\tilde{\boldsymbol{u}}_f).$$

Thanks to the introduction of the pull–back maps, we can reformulate our problem in the reference configuration $\hat{\Omega}$: for every $t \in [0,T]$, find the fluid velocity $\hat{\boldsymbol{u}}_f(t) : \hat{\Omega}_f \mapsto \mathbb{R}^2$, the fluid pressure $\hat{p}_f(t) : \hat{\Omega}_f \mapsto \mathbb{R}$ and the solid deformation $\hat{\boldsymbol{d}}_s(t) : \hat{\Omega}_s \mapsto \mathbb{R}^2$ such that:

$$\begin{cases} \rho_f JK(\partial_t \hat{\boldsymbol{u}}_f + \hat{\nabla}\hat{\boldsymbol{u}}_f\boldsymbol{G}_\mu^{-1}\boldsymbol{F}_\mu^{-1}(\hat{\boldsymbol{u}}_f - \partial_t\hat{\boldsymbol{d}}_f)) \\ \qquad\qquad - \hat{\text{div}}(JK\hat{\sigma}_f(\hat{\boldsymbol{u}}_f, \hat{p}_f)\boldsymbol{F}_\mu^{-T}\boldsymbol{G}_\mu^{-T}) = \boldsymbol{b}_f \quad \text{in } \hat{\Omega}_f \times (0,T], \\ \hat{\text{div}}(JK\boldsymbol{G}_\mu^{-1}\boldsymbol{F}_\mu^{-1}\boldsymbol{u}_f) = 0 \quad \text{in } \hat{\Omega}_f \times (0,T], \\ \rho_s K\partial_{tt}\hat{\boldsymbol{d}}_s - \hat{\text{div}}(K\hat{\boldsymbol{P}}(\hat{\boldsymbol{d}}_s)\boldsymbol{G}_\mu^{-T}) = \boldsymbol{b}_s \quad \text{in } \hat{\Omega}_s \times (0,T], \end{cases} \tag{3.17}$$

where:

$$\begin{aligned} \hat{\sigma}_f(\hat{\boldsymbol{u}}_f, \hat{p}_f) &= \mu_f(\hat{\nabla}\hat{\boldsymbol{u}}_f\boldsymbol{G}_\mu^{-1}\boldsymbol{F}_\mu^{-1} + \boldsymbol{F}_\mu^{-T}\boldsymbol{G}_\mu^{-T}\hat{\nabla}^T\hat{\boldsymbol{u}}_f), \\ \hat{\boldsymbol{P}}(\hat{\boldsymbol{d}}_s) &= \lambda_s \text{tr}\varepsilon_s(\hat{\boldsymbol{d}}_s)\boldsymbol{I} + 2\mu_s\varepsilon_s(\hat{\boldsymbol{d}}_s), \\ \varepsilon_s(\hat{\boldsymbol{d}}s) &= \frac{1}{2}(\hat{\nabla}\hat{\boldsymbol{d}}_s\boldsymbol{G}_\mu^{-1} + \boldsymbol{G}_\mu^{-T}\hat{\nabla}^T\hat{\boldsymbol{d}}_s). \end{aligned} \tag{3.18}$$

We have the coupling conditions

$$\begin{cases} \hat{\boldsymbol{d}}_f = \hat{\boldsymbol{d}}_s \quad \text{on } \hat{\Gamma}_{FSI} \\ \hat{\boldsymbol{u}}_f = \partial_t\hat{\boldsymbol{d}}_s \quad \text{on } \hat{\Gamma}_{FSI}, \\ JK\hat{\sigma}_f(\hat{\boldsymbol{u}}_f, \hat{p}_f)\boldsymbol{F}_\mu^{-T}\boldsymbol{G}_\mu^{-T}\hat{\boldsymbol{n}}_f = -K\hat{\boldsymbol{P}}(\hat{\boldsymbol{d}}_s)\boldsymbol{G}_\mu^{-T}\hat{\boldsymbol{n}}_s \quad \text{on } \hat{\Gamma}_{FSI}, \end{cases}$$

and the following boundary conditions:

$$\begin{cases} \hat{\sigma}_f(\hat{\boldsymbol{u}}_f, \hat{p}_f)\hat{\boldsymbol{n}} = -p_{in}(t)\hat{\boldsymbol{n}} \quad \text{on } \hat{\Gamma}_{in}, \\ \hat{\sigma}_f(\hat{\boldsymbol{u}}_f, \hat{p}_f)\hat{\boldsymbol{n}} = -p_{out}(t)\hat{\boldsymbol{n}} \quad \text{on } \hat{\Gamma}_{out}, \\ \hat{\boldsymbol{d}}_s = 0 \quad \text{on } \hat{\Gamma}_s^D. \end{cases}$$

Again, $\hat{\boldsymbol{n}}$ represents the normal vector to the relative part of the boundary of the domain.

Remark 3.6.1. In this section we stressed the difference between entities on the current configuration, the parametrized intermediate configuration and the reference configuration, by using the superscripts $\tilde{}$ and $\hat{}$. However, since from now on everything will be cast in the reference configuration, and in order to make the notation as light as possible, we will drop all the superscripts.

3.7 Offline computational phase

In this Section we present the offline phase of the partitioned procedure in the presence of a parameter μ. We employ again a Chorin-Temam projection scheme for the Navier–Stokes equation. Since the whole procedure is very similar to the one already presented for the non–parametric case, and the only thing that changes is the weak formulation in the algortihm, we present directly the final algortihm. In order to do so, we define a time-stepping procedure by sampling the time interval $[0, T]$ with an equispaced sampling $\{t_0, \dots, t_N\}$, where $t_i = i\Delta T$, for $i = 0, \dots, N_T$ and $N_T = \frac{T}{\Delta T}$. We discretize the time derivative of a function f with a first backward difference: $D_t f^{i+1} = \frac{f^{i+1} - f^i}{\Delta T}$, and $D_{tt} f^{i+1} = D_t(D_t f^{i+1})$, where $f^{i+1} = f(t^{i+1})$.
In the following, we use the same function spaces that we have introduced in Section 3.3.1:

$$
\begin{aligned}
V(\Omega_f) &:= [H^1(\Omega_f)]^2, \\
E^f(\Omega_f) &:= [H^1(\Omega_f)]^2, \\
Q(\Omega_f) &:= L^2(\Omega_f), \\
E^s(\Omega_s) &= [H^1(\Omega_s)]^2,
\end{aligned}
$$

endowed with the H^1 norm ($V(\Omega_f)$, $E^f(\Omega_f)$ and $E^f(\Omega_s)$) and the L^2 norm respectively. We remark that in the previous definitions, the domains Ω_f and Ω_s are the *parameter independent reference configurations*. Again we discretize in space the FSI problem, using second order Lagrange Finite Elements for fluid velocity, the fluid displacement and the solid displacement, resulting in the discrete spaces $V_h \subset V$, $E_h^f \subset E^f$ and $E_h^s \subset E^s$, while the fluid pressure is discretized with first order Lagrange Finite Elements, resulting in the discrete space $Q_h \subset Q$. The space discretized version of the partitioned procedure now reads as follows: for $i = 0, \dots, N_T$,

Extrapolation of the mesh displacement:

find $\boldsymbol{d}_{f,h}^{i+1} \in E_h^f$ such that $\forall \boldsymbol{e}_{f,h} \in E_h^f$:

$$
\begin{cases}
\int_{\Omega_f} K \nabla \boldsymbol{d}_{f,h}^{i+1} \boldsymbol{G}_\mu^{-1} \cdot \nabla \boldsymbol{e}_{f,h} \boldsymbol{G}_\mu^{-1} \, dx = 0 \quad \text{in } \Omega_f, \\
\boldsymbol{d}_{f,h}^{i+1} = \boldsymbol{d}_{s,h}^i \quad \text{on } \Gamma_{FSI}.
\end{cases}
\tag{3.19}
$$

Fluid explicit step:

find $\boldsymbol{u}_{f,h}^{i+1} \in V_h$ such that $\forall \boldsymbol{v}_h \in V_h$:

$$
\begin{cases}
\rho_f \displaystyle\int_{\Omega_f} JK\Big(\frac{\boldsymbol{u}_{f,h}^{i+1} - \boldsymbol{u}_{f,h}^i}{\Delta T}\Big) \cdot \boldsymbol{v}_h \, dx + \rho_f \int_{\Omega_f} JK[\nabla \boldsymbol{u}_{f,h}^{i+1} \boldsymbol{G}_\mu^{-1} \boldsymbol{F}_\mu^{-1}](\boldsymbol{u}_{f,h}^{i+1} - D_t \boldsymbol{d}_{f,h}^{i+1}) \cdot \boldsymbol{v}_h \, dx \\
\quad + \mu_f \displaystyle\int_{\Omega_f} JK\varepsilon(\boldsymbol{u}_{f,h}^{i+1}) \boldsymbol{F}_\mu^{-T} \boldsymbol{G}_\mu^{-T} : \nabla \boldsymbol{v}_h \, dx + \int_{\Omega_f} JK \boldsymbol{F}_\mu^{-T} \boldsymbol{G}_\mu^{-T} \nabla p_{f,h}^i \cdot \boldsymbol{v}_h \, dx = 0 \quad \text{in } \Omega_f, \\
\boldsymbol{u}_{f,h}^{i+1} = D_t \boldsymbol{d}_{f,h}^{i+1} \quad \text{on } \Gamma_{FSI},
\end{cases}
\tag{3.20}
$$

Implicit step:

for any $j = 0, \dots$ until convergence:

1. **fluid projection substep (pressure Poisson formulation):** find $p_{f,h}^{i+1,j+1} \in Q_h$ such that $\forall q_h \in Q_h$:

$$
\begin{aligned}
&-\frac{\rho_f}{\Delta T}\int_{\Omega_f} \operatorname{div}(JK\boldsymbol{G}_\mu^{-1}\boldsymbol{F}_\mu^{-1}\boldsymbol{u}_{f,h}^{i+1})q_h\,dx + \alpha_{ROB}\int_{\Gamma_{FSI}} p_{f,h}^{i+1,j}q_h\,ds \\
&-\rho_f\int_{\Gamma_{FSI}}(D_{tt}\boldsymbol{d}_{s,h}^{i+1,j})\cdot JK\boldsymbol{F}_\mu^{-T}\boldsymbol{G}_\mu^{-T}\boldsymbol{n}_f q_h\,ds = \alpha_{ROB}\int_{\Gamma_{FSI}} p_{f,h}^{i+1,j+1}q_h\,ds + \\
&+\int_{\Omega_f} JK\boldsymbol{G}_\mu^{-T}\boldsymbol{F}_\mu^{-T}\nabla p_{f,h}^{i+1,j+1}\cdot\boldsymbol{G}_\mu^{-T}\boldsymbol{F}_\mu^{-T}\nabla q_h\,dx,
\end{aligned}
$$

   subject to the boundary conditions (3.7);

2. **structure projection substep:** find $\boldsymbol{d}_{s,h}^{i+1,j+1} \in E_h^s$ such that $\forall \boldsymbol{e}_{s,h} \in E_h^s$:

$$
\begin{aligned}
\rho_s\int_{\Omega_s} KD_{tt}\boldsymbol{d}_{s,h}^{i+1,j+1}\cdot\boldsymbol{e}_{s,h}\,dx + \int_{\Omega_s} K\boldsymbol{P}(\boldsymbol{d}_{s,h}^{i+1,j+1})\boldsymbol{G}_\mu^{-T} : \nabla\boldsymbol{e}_{s,h}\,dx = \\
= -\int_{\Gamma_{FSI}} JK\sigma_f(\boldsymbol{u}_{f,h}^{i+1,j+1}, p_{f,h}^{i+1,j+1})\boldsymbol{G}_\mu^{-T}\boldsymbol{F}_\mu^{-T}\boldsymbol{n}_f\cdot\boldsymbol{e}_{s,h}\,dx
\end{aligned}
$$

   subject to the boundary condition $\boldsymbol{d}_s = 0$ on Γ_D^s

In the fluid projection step, in order to enhance the stability of the method we have employed again a Robin boundary condition, which in the case of shape parametrization reads as follows:

$$
\begin{aligned}
\alpha_{ROB}p^{i+1} + \boldsymbol{G}_\mu^{-T}\boldsymbol{F}_\mu^{-T}\nabla p^{i+1}\cdot JK\boldsymbol{G}_\mu^{-T}\boldsymbol{F}_\mu^{-T}\boldsymbol{n}_f = \\
= \alpha_{ROB}p^{i+1,\star} - \rho_f D_{tt}\boldsymbol{d}_s^{i+1,\star}\cdot JK\boldsymbol{G}_\mu^{-T}\boldsymbol{F}_\mu^{-T}\boldsymbol{n}_f.
\end{aligned}
$$

3.7.1 POD-Greedy procedure

In order to find the reduced basis, we rely on a POD–Greedy procedure [29, 46]: this means that we explore the parameter space $\mathcal{P}$ with a Greedy procedure, and we explore in time with a POD. To explore the parameter space we actually use a *pseudo*–Greedy procedure, and this is because we do not have at hand an error estimator for a Fluid–Structure Interaction problem; we sample the parameter space $\mathcal{P}$ with M equispaced samples and we obtain the training set $\mathcal{P}_{train} = \{\mu_1, \dots, \mu_M\}$. We therefore have the following snapshots matrices:

$$
\begin{aligned}
\boldsymbol{\mathcal{S}}_z &= [\underline{\boldsymbol{z}}_h(t^1;\mu_1), \dots, \underline{\boldsymbol{z}}_h(t^{N_T};\mu_1), \dots, \underline{\boldsymbol{z}}_h(t^1;\mu_M), \dots, \underline{\boldsymbol{z}}_h(t^{N_T};\mu_M)] \in \mathbb{R}^{N_z^h\times\widehat{M}}, \\
\boldsymbol{\mathcal{S}}_{d_s} &= [\underline{\boldsymbol{d}}_{s,h}(t^1,\mu_1), \dots, \underline{\boldsymbol{d}}_{s,h}(t^{N_T},\mu_1), \dots, \underline{\boldsymbol{d}}_{s,h}(t^1,\mu_M), \dots, \underline{\boldsymbol{d}}_{s,h}(t^{N_T},\mu_M)] \in \mathbb{R}^{N_{d_s}^h\times\widehat{M}}, \\
\boldsymbol{\mathcal{S}}_p &= [\underline{p}_{f,h}(t^1,\mu_1), \dots, \underline{p}_{f,h}(t^{N_T},\mu_1), \dots, \underline{p}_{f,h}(t^1,\mu_M), \dots, \underline{p}_{f,h}(t^{N_T},\mu_M)] \in \mathbb{R}^{N_p^h\times\widehat{M}},
\end{aligned}
$$

where $\widehat{M} = N_T \cdot M$. We perform a POD on each snapshots matrix, obtaining the following modes: $\{\Phi_z^k\}_{k=1}^{N_z}$, $\{\Phi_{d_s}^k\}_{k=1}^{N_{d_s}}$, $\{\Phi_p^k\}_{k=1}^{N_p}$, with N_z, N_{d_s}, N_p chosen according to the rate of decay of the eigenvalues returned by the POD on $\boldsymbol{\mathcal{S}}_z$, $\boldsymbol{\mathcal{S}}_{d_s}$ and $\boldsymbol{\mathcal{S}}_p$, respectively. We denote by V_N, E_N^s, Q_N the reduced spaces spanned by these basis

functions respectively. The reduced solutions $\boldsymbol{z}_{f,N}^{i+1}(\mu)$, $p_{f,N}^{i+1}(\mu)$ and $\boldsymbol{d}_{s,N}^{i+1}(\mu)$ at time–step $i+1$ and for $\mu \in \mathcal{P}$ are defined as follows:

$$\boldsymbol{z}_{f,N}^{i+1}(\mu) = \sum_{k=1}^{N_{\boldsymbol{z}_f}} \underline{\mathbf{z}}_k^{i+1}(\mu)\Phi_{\boldsymbol{z}_f}^k, \tag{3.21}$$

$$p_{f,N}^{i+1}(\mu) = \sum_{k=1}^{N_p} \underline{p}_k^{i+1}(\mu)\Phi_p^k, \tag{3.22}$$

$$\boldsymbol{d}_{s,N}^{i+1}(\mu) = \sum_{k=1}^{N_{\boldsymbol{d}_s}} \underline{\mathbf{d}}_k^{i+1}(\mu)\Phi_{\boldsymbol{d}_s}^k, \tag{3.23}$$

Remark 3.7.1. We remark that also in the case of parameter dependence, we implement an harmonic extension of the reduced basis for the solid displacement. Once we have the mesh displacement reduced basis $\{\Phi_{d_f}^k\}_{k=1}^{N_{d_s}}$ and the relative reduced space E_N^f, we define the reduced mesh displacement as in (3.15).
Remark 3.7.2. Also in this case, if we have a non–homogeneous boundary condition for the fluid pressure, we take care of it with a lifting function ℓ. We will not write it explicitly in the formulation, in order to ease the notation.

3.8 Online computational phase

The online phase in the case of shape parametrization is similar to the online phase described in Section 3.4; indeed, for the reduced basis generation we adopted the same strategy: we relied on a change of variable for the fluid velocity, and on an harmonic extension of the solid displacement reduced basis to obtain the reduced basis for the mesh displacement. The reduced problem now reads: for every $i = 0, \dots, N_T$ and for $\mu \in \mathcal{P}$:

Mesh displacement:

let $\boldsymbol{d}_{f,N}^{i+1}(\mu)$ be defined by the reduced solid displacement at the previous time–step:

$$\boldsymbol{d}_{f,N}^{i+1}(\mu) = \sum_{k=1}^{N_{\boldsymbol{d}_s}} \underline{\mathbf{d}}_k^{i}(\mu)\Phi_{\boldsymbol{d}_f}^k; \tag{3.24}$$

Fluid explicit step (with change of variable):

find $z_{f,N}^{i+1}(\mu) \in V_N$ such that $\forall \boldsymbol{v}_N \in V_N$:

$$\rho_f \int_{\Omega_f} JK\Big(\frac{\boldsymbol{z}_{f,N}^{i+1}(\mu) - \boldsymbol{u}_{f,N}^{i}(\mu)}{\Delta T}\Big) \cdot \boldsymbol{v}_N \, dx + \mu_f \int_{\Omega_f} JK\varepsilon(\boldsymbol{z}_{f,N}^{i+1}(\mu))\boldsymbol{F}_\mu^{-T}\boldsymbol{G}_\mu^{-T} : \nabla \boldsymbol{v}_N \, dx$$
$$+ \rho_f \int_{\Omega_f} JK\nabla \boldsymbol{z}_{f,N}^{i+1}(\mu)\boldsymbol{G}_\mu^{-1}\boldsymbol{F}_\mu^{-1}\boldsymbol{z}_{f,N}^{i+1}(\mu) \cdot \boldsymbol{v}_N \, dx +$$
$$+ \rho_f \int_{\Omega_f} JK\nabla D_t\boldsymbol{d}_{f,N}^{i+1}(\mu)\boldsymbol{G}_\mu^{-1}\boldsymbol{F}_\mu^{-1}\boldsymbol{z}_{f,N}^{i+1}(\mu) \cdot \boldsymbol{v}_N \, dx +$$
$$+ \int_{\Omega_f} JK\boldsymbol{F}_\mu^{-T}\boldsymbol{G}_\mu^{-T}\nabla p_{f,N}^{i}(\mu) \cdot \boldsymbol{v}_h \, dx = -\rho_f \int_{\Omega_f} JK\Big(\frac{D_t\boldsymbol{d}_{f,N}^{i+1}(\mu)}{\Delta T}\Big) \cdot \boldsymbol{v}_N \, dx$$
$$- \mu_f \int_{\Omega_f} JK\varepsilon(D_t\boldsymbol{d}_{f,N}^{i+1}(\mu))\boldsymbol{F}_\mu^{-T}\boldsymbol{G}_\mu^{-T} : \nabla \boldsymbol{v}_N \, dx \quad \text{in } \Omega_f.$$

We then restore the reduced fluid velocity: $\boldsymbol{u}_{f,N}^{i+1}(\mu) = \boldsymbol{z}_{f,N}^{i+1}(\mu) + D_t\boldsymbol{d}_{f,N}^{i+1}(\mu)$.

Implicit step:

for any $j = 0, \dots$ until convergence:

1. **fluid projection substep:** find $p_{f,N}^{i+1,j+1}(\mu) \in Q_N$ such that $\forall q_N \in Q_N$:

$$-\frac{\rho_f}{\Delta T}\int_{\Omega_f} \mathrm{div}(JK\boldsymbol{G}_\mu^{-1}\boldsymbol{F}_\mu^{-1}\boldsymbol{u}_{f,N}^{i+1}(\mu))q_N \, dx + \alpha_{ROB}\int_{\Gamma_{FSI}} p_{f,N}^{i+1,j}(\mu)q_N \, ds$$
$$-\rho_f \int_{\Gamma_{FSI}} D_{tt}\boldsymbol{d}_{s,N}^{i+1,j}(\mu) \cdot JK\boldsymbol{F}_\mu^{-T}\boldsymbol{G}_\mu^{-T}\boldsymbol{n}_f q_N \, ds = \alpha_{ROB}\int_{\Gamma_{FSI}} p_{f,N}^{i+1,j+1}(\mu)q_N \, ds$$
$$+\int_{\Omega_f} JK\boldsymbol{F}_\mu^{-T}\boldsymbol{G}_\mu^{-T}\nabla p_{f,N}^{i+1,j+1}(\mu) \cdot \boldsymbol{F}_\mu^{-T}\boldsymbol{G}_\mu^{-T}\nabla q_N \, dx;$$

   subject to some suitable boundary conditions for the pressure.

2. **structure projection substep:** find $\boldsymbol{d}_{s,N}^{i+1,j+1}(\mu) \in E_N^s$ such that $\forall \boldsymbol{e}_s \in E_N^s$:

$$\rho_s \int_{\Omega_s} KD_{tt}\boldsymbol{d}_{s,N}^{i+1,j+1}(\mu) \cdot \boldsymbol{e}_N \, dx + \int_{\Omega_s} K\boldsymbol{P}(\boldsymbol{d}_{s,N}^{i+1,j+1}(\mu))\boldsymbol{G}_\mu^{-T} : \nabla \boldsymbol{e}_N \, dx =$$
$$= -\int_{\Omega_s} JK\sigma_f(\boldsymbol{u}_{f,N}^{i+1}(\mu), p_{f,N}^{i+1,j+1}(\mu))\boldsymbol{F}_\mu^{-T}\boldsymbol{G}_\mu^{-T}\boldsymbol{n}_f \cdot \boldsymbol{e}_N \, dx$$

3.9 Numerical results

We now present some numerical results concerning the parametrized version of the two dimensional FSI test case presented in Section 3.2. The original domain is shown in Figure 3.6, together with the reference configuration, and the parametrized reference configuration. The fluid domain is represented in blue, while the solid (the leaflets) is depicted in red. The height of the channel is 2.5 cm, its length is 10 cm; the leaflets are 0.2 cm thick, and they are situated 1 cm downstream the fluid inlet, which corresponds to the vertical boundary on the left of the blue domain. One geometrical parameter is

TABLE 3.2: Physical and geometrical constants and parameters, for the geometrically parametrized leaflets test case.

Physical parameter	**Value**
ρ_s	1.1 g/cm^3
μ_s	100000
λ_s	800000
ρ_f	1 g/cm^3
μ_f	0.035 Poise
Geometrical parameter	**Value**
μ	$[0.5, 1.0]$
FE displacement order	2
FE velocity order	2
FE pressure order	1
N_{train}	10
N	100

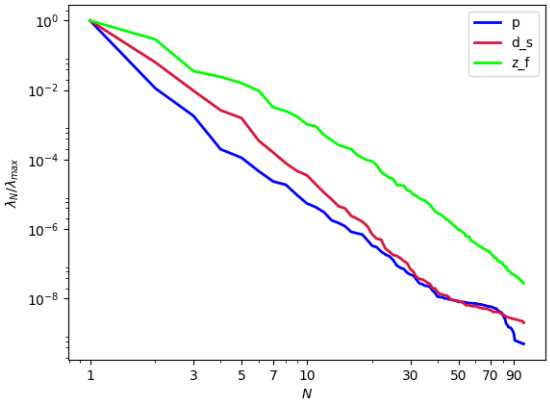
considered: the length μ of the leaflets, where we have chosen $\mu \in \mathcal{P} = [0.5, 1.0]$. An affine mapping T is chosen to deform the reference domain $\hat{\Omega}$, obtained for $\mu = 1.0$ cm, to the parametrized configuration $\tilde{\Omega}(\mu)$. Top and bottom walls of the blue domain are rigid, thus both the displacement $\boldsymbol{d}_f$ and the fluid velocity $\boldsymbol{u}_f$ are set to zero. Homogeneous Neumann condition is imposed on $\boldsymbol{u}_f$ on the outlet; a pressure profile $p_{in}(t)$ is described at the inlet, where:

$$p_{in}(t) = \begin{cases} 5 - 5\cos\left(\frac{2\pi t}{T_{in}}\right) & \text{for } t \leq 0.1s \\ 5 \qquad \text{for } t > 0.1s, \end{cases}$$

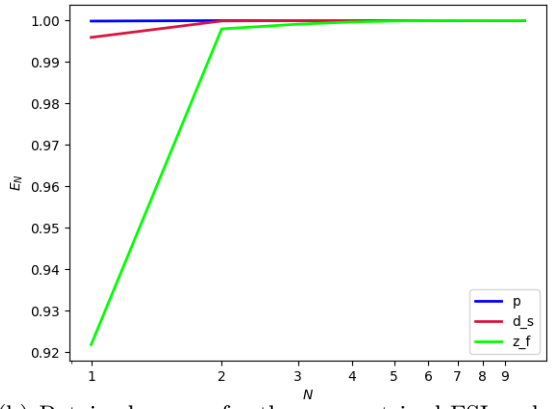
and $T_{in} = 0.4$ s. Also in this case we set a tolerance of $\varepsilon = 10^{-6}$ as a stopping criterion for the subiterations between the pressure problem and the solid problem.

For the simulation, we use a time–step $\Delta t = 10^{-4}$, for a maximum number of time–steps $N_T = 800$, thus $T = 0.08$s. Table 3.2 summarizes the details of the offline stage and of the FE discretization.

Figure 3.7a shows the rate of decay of the eigenvalues returned by a POD on $\boldsymbol{z}_f$, p_f and $\boldsymbol{d}_s$, respectively. As we can see, now the eigenvalues of the pressure decay faster than the ones for the displacement and for the fluid velocity. This is confirmed also at the level of retained energy, as we can see from Figure 3.7b: the first mode of the pressure is indeed the most energetic one, while the first mode of the velocity is the least energetic one. In Figure 3.8 we see a representative solution for the fluid velocity, for two different values of the geometrical parameter μ. As we can notice, for the same inlet pressure profile, when the leaflets are longer, at the final time of the simulation ($T = 0.08s$) we are close to the formation of a jet inbetween the two leaflets. Figure 3.9 shows the reduced displacement, for the three different values of the geometrical parameter: $\mu = 0.61$, $\mu = 0.78$ and $\mu = 1.0$; the influence of μ is clear: the longer the leaflets, the bigger their deformation is going to be, under the same physical parameters.



(a) POD eigenvalues for the parametrized FSI problem.



(b) Retained energy for the parametrized FSI problem.

FIGURE 3.7: POD eigenvalues and retained energy for the parametric leaflets problem.

3.10 Conclusions

In this Chapter we presented a partitioned reduced order model for adressing Fluid–Structure Interaction problems, in the time dependent case, and possibly with a geometrical parametrization. We rely on a Proper Orthogonal Decomposition for the generation of the reduced basis, and we introduce a change of variable in the problem formulation, in order to avoid the use of Lagrange multipliers to impose non–homogeneous boundary conditions. The procedure that we have proposed extends the work presented in [14], to the case of the coupling between an incompressible fluid and a thick, two dimensional, structure, also in the presence of geometrical parametrization. The results that we have obtained confirm the following points:

1. introducing a change of variable for the fluid velocity avoids the introduction of a further unknown in the system, namely a Lagrange multiplier;
2. avoiding to perform a POD directly on the snapshots of the mesh displacement $\boldsymbol{d}_f$ allows, again, to avoid the introduction of another Lagrange multiplier to impose the continuity of the displacement at the interface; also, the choice to extend harmonically the basis functions $\{\Phi\}_{i=1}^{N_{d_s}}$ accounts for just a small additional offline cost, namely the solution of N_{d_s} harmonic problems;

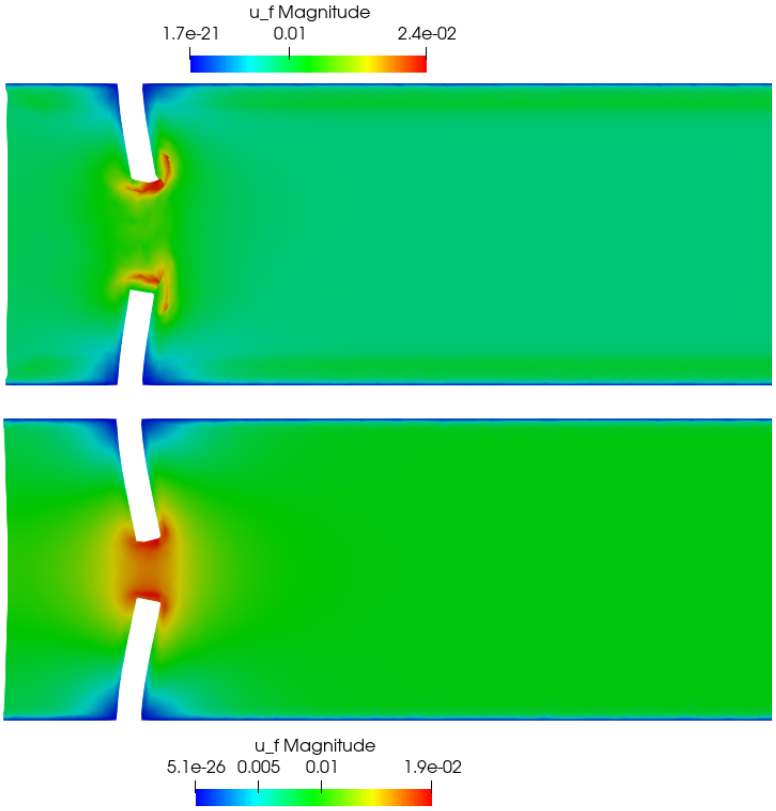


FIGURE 3.8: Representative solutions for the fluid velocity $\boldsymbol{u}_f$, obtained with the reduced order model proposed ($N = 100$ basis for the fluid velocity), for different values of the leaflet length μ: $\mu = 0.77$ (top), and $\mu = 1.0$ (bottom).

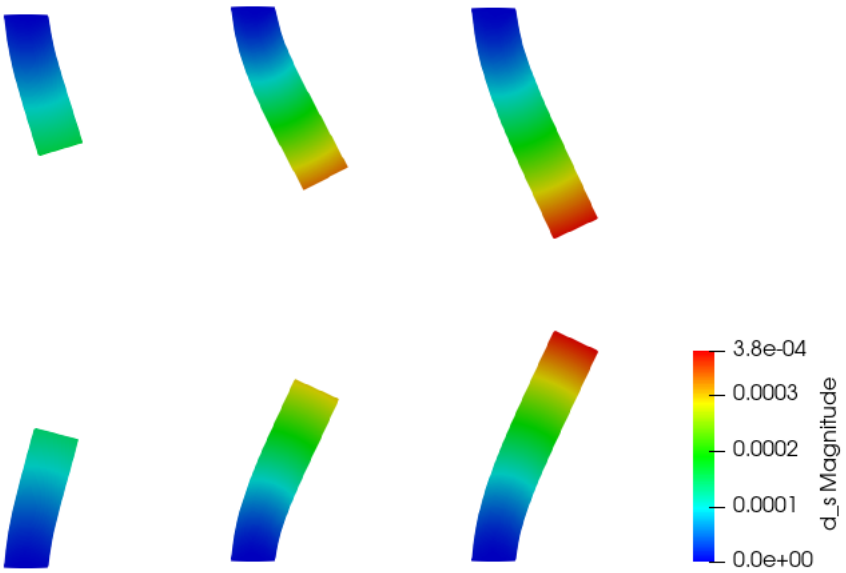


FIGURE 3.9: Representative solutions for the displacement of the leaflets, $\boldsymbol{d}_s$, obtained with the reduced order model proposed ($N = 100$ basis for the displacement), for different values of the leaflet length μ: $\mu = 0.61$ (left), $\mu = 0.78$ (center), and $\mu = 1.0$ (right). The displacement has been magnified by a factor 10^3 for visualization purposes.

3. during the online phase, computing the reduced mesh displacement through the step (3.24) translates in no additional online cost: indeed the computation of the reduced mesh displacement does not require the solution of an harmonic problem, but requires instead a linear combination of the basis functions $\{\Phi_{d_f}^k\}_{i=1}^{N_{d_s}}$, once the online solid displacement is known.

We also remark two important aspects of the procedure presented in this Chapter: first of all, we did not employ a supremizer enrichment of the fluid velocity space. This choice is motivated by the fact that, even at the FE level, the Chorin–Temam projection scheme with the pressure Poisson formulation can be applied succesfully also to velocity–pressure FE spaces that do not satisfy the inf–sup condition, see [79]. The second remarks is related to the fact that one can recover an efficient online–offline splitting thanks to the Empirical Interpolation Method (EIM), see for example [17, 120, 13]. The EIM has not been used at the moment in the nonlinear parametrized problem; we have focused in this Chapter on the developement and test of a reduced order segregated procedure for FSI problems which involve the coupling of an incompressible fluid with an elastic structure; in addition to this, another future step in the design of a partitioned reduced order model is the developement of an a posteriori error estimator.

Chapter 4

Reducing the Kolmogorov n-width by transportation maps

This Chapter is devoted to the formulation of a model order reduction procedure for those problems whose solution manifold exhibits a slow decay of the Kolmogorov n–width. Thanks to the introduction of an additional preprocessing on the set of snapshots during the offline phase of the RBM, we will see the improvements that we obtain in the rate of decay of the eigenvalues returned by a POD on the solution manifold. The advantage of adopting a reduced order method that is enriched with a preprocessing procedure during the offline phase, is represented by the fact that we are able to construct a reduced solution using a significantly smaller number of basis functions, with a consequent reduction of the dimension of the system to be solved in the online phase.
This Chapter is structured as follows: in Section 4.2 we introduce the preprocessing procedure. We then test the preprocessing procedure on three test cases: a fluid flow around a rotating cylinder in Section 4.3, a fluid in a channel with deformable walls in Section 4.4, and finally a fluid dynamics problem with physical parametrization in Section 4.5.
The material contained in this Chapter has been submitted for journal publication, see [132].

4.1 Motivation

The reduced basis method [90, 84, 156, 26, 110, 85, 129] is a powerful tool in the framework of fast simulations of parametrized partial differential equations (PDEs): its efficiency relies on the possibility to construct an approximation of the solution for any value of the parameter in the span of a few basis functions, which are computed in the (expensive) offline phase. Despite its capability has been acclaimed in a large variety of situations, model reduction of advection dominated problems is still a challenging task [137, 76, 52]. It has therefore become clear that a modification of the way the reduced basis method works is necessary, especially in order to be able to obtain a small basis set also in these more challenging situations.

Assume that Ω is the physical domain of the problem of interest. Let $\mathcal{P} \subset \mathbb{R}^p$, $p \in \mathbb{N}$, be a compact set, and denote by $\mu \in \mathcal{P}$ the parameter. Furthermore, let $t \in [0, T]$ be the time, for some $T > 0$. In the following the symbol η will either stand for μ in parametrized stationary problems, t in non-parametric unsteady problems, or the pair (t, μ) in parametrized unsteady problems. For any value of $\eta \in \mathcal{E}$, we seek the

solution $\mathbf{z}(\eta)\colon \Omega \to \mathbb{R}^Z$, $Z \in \mathbb{N}$, to the following PDE:

$$N(\mathbf{z}(\eta); \eta) = 0 \tag{4.1}$$

where N is a nonlinear operator working on functions defined over Ω, and $\mathcal{E} := \mathcal{P}$, $\mathcal{E} := [0, T]$ or $\mathcal{E} := [0, T] \times \mathcal{P}$, respectively, in the three aforementioned cases. Specific examples of problems of interests in computational fluid dynamics and Fluid–Structure Interaction will be introduced throughout the Chapter.

Let $z(\eta)$ be a component of $\mathbf{z}(\eta)$, and $\mathcal{M}_z$ be the solution manifold, embedded in some normed linear space $(X_z, ||\cdot||_{X_z})$, defined as:

$$\mathcal{M}_z = \{z(\eta), \quad \eta \in \mathcal{E}\}.$$

The choice of a component-wise solution manifold (rather than just one solution manifold associated to the vector $\mathbf{z}(\eta)$) will be motivated in Section 4.4. One fundamental assumption of the reduced basis method is that $\mathcal{M}_z$ can be approximated in an accurate way by a sequence of finite dimensional spaces: any element of $\mathcal{M}_z$ can be recovered using a linear combination of solutions of (4.1), which are computed only once and for all. The mathematical entity that incorporates this concept is the Kolmogorov n-width of $\mathcal{M}_z$.

Definition 4.1.1. *The* Kolmogorov n–width *of the solution manifold $\mathcal{M}_z$ is:*

$$D_n(\mathcal{M}_z, X_z) = \inf_{E_n \subset X_z} \sup_{f \in \mathcal{M}_z} \inf_{g \in E_n} ||f - g||_{X_z}, \tag{4.2}$$

where E_n is any linear subspace of X_z of dimension n.

The Kolmogorov n-width D_n tells us the entity of the error that we commit by approximating any element f of $\mathcal{M}_z$ with an element g of a linear space E_n. The faster D_n decays as we let n grow, the greater possibility we have to build a good linear approximation space of $\mathcal{M}_z$ of low dimension. In the majority of the problems there is no explicit analytic formula for D_n, yet there are some situations where we can compute good bounds on the Kolmogorov n-width [126, 47]. In general, since it is very difficult to provide such bounds on D_n, we can only hope that the n-width of the solution manifold is small. A heuristic way to check that this hypothesis is satisfied by the problem of interest is to run a POD on a set of snapshots, and check the rate of decay of the eigenvalues $\{\lambda_i\}_i$ returned by the POD: if the $\{\lambda_i\}_i$ decay fast, then we can expect to be able to build a good low dimensional linear approximation space for $\mathcal{M}_z$. This assumption often fails in transport-dominated problems, which show a very slow decay of the eigenvalues of the POD, and thus the inability of the reduced basis method to reconstruct any element of $\mathcal{M}_z$ by using a small number of basis functions.

A growing number of works which focus on constructing alternative (nonlinear) model order reduction techniques, that can be effectively applied to transport-dominated problems, has appeared in recent years. In [1] the authors show an L^1 norm minimization technique, to be used for the approximation of nonlinear hyperbolic equations, without anyway curing the problem of high dimensional solution manifolds. Approximated Lax pairs have been employed for model reduction of nonlinear problems arising in cardiac electrophysiology [68, 69]. In [68] the authors construct a reduced basis space at each time step: the reduced basis are the modes of a Schroedinger operator where the potential is the solution at the previous time step. The drawback of this procedure is the increase of the number of basis functions as the approximation accuracy requirement is increased. Nonlinear model reduction techniques in metric

spaces are proposed in [52], relying either on a tangent principal component analysis or barycentric greedy algorithm. Adaptivity is also employed to overcome difficulties arising in model reduction of transport dominated flows, see [39] and [4] for the use of localized reduced bases or [138] for the procedure to update the basis set. Several other different techniques rely on a composition with a suitably defined transport map (on which we will focus on in the rest of the Chapter). Such map may be obtained as the solution of a Monge-Kantorovich optimal mass transport problem [95, 24], or provided analytically in simple cases as the one-dimensional problem considered in [37]. More complicated configurations, possibly including shocks, can still be handled relying on transport maps based on more advanced shape parametrization maps [36]. Extension to multiple transport phenomena are also possible, most notably by means of the shifted POD [150], as well as (possibly interacting) shocks by recent extensions of the transported snapshots interpolation [179, 178, 180] and transported snapshots model reduction methods [128]. An effort towards the design of a reduction procedure that is not strictly related to the physical phenomenon under investigation, e.g. transportation of a quantity or translation of a wave, has been made in [113]. The authors propose a methodology which focuses on optimal projection of general dynamical systems onto arbitrary nonlinear trial manifolds, in the online stage, where this nonlinear trial manifold is computed from snapshot data alone, in the offline stage, thanks to the employment of convolutional autoencoders. The reader interested in the use of machine learning techniques in the model order reduction framework is referred to [63, 87]. While in simple cases the underlying transport maps are provided by the user, registration techniques can be employed to automate their selection in more complex geometrical configurations [169]. Application of the use of transport maps for model reduction based on an embedded high fidelity method is shown by in [102]. An alternative is the freezing method [136, 153, 27], in which the key tool is the identification of a Lie group acting on a frozen solution component. However, to properly develop and analyze the proposed methodology for complex problems in fluid dynamics, very involved mathematical tools and settings are needed which as of now hinders its applicability in a broader setting. This consideration, together with previous observations on other techniques, makes it clear that there is the need for a lighter, simpler and more natural framework. A methodology that satisfies these requirements and that is based on the definition of some transport maps has been introduced and applied to some toy problems by Cagniart [35] and Cagniart et al. [37]. The goal of this Chapter is to present an application of model reduction based on transported snapshots for problems in fluid dynamics and Fluid–Structure Interaction, focusing in particular on the comparison between results of the standard offline phase and the novel one which relies on transport. As (especially in fluid dynamics problems) further challenges are present during the online stage, such as keeping into account stabilization [172, 125] or turbulence modelling [43, 159], we limit our exposition to the offline stage; future research work will extend the results presented in this Chapter to the online stage.

4.2 Nonlinear model reduction by transport maps

In this section we summarize the nonlinear approach that we will apply in the forthcoming sections to fluid dynamics and Fluid–Structure Interaction problems. We will closely follow the presentation and the notation introduced by Cagniart et al. [37]

for a parametrized viscous Burgers equation. The idea proposed therein is to "pre-process" the solution manifold $\mathcal{M}_z$ by a composition with a map belonging to a family of smooth and invertible mappings:

$$\mathcal{F}_z = \{F_\eta : \Omega \to \Omega,\ F_\eta \text{ is smooth and invertible},\ \eta \in \mathcal{E}\}.$$

Maps in the family $\mathcal{F}_z$ depend on the same η appearing in (4.1), and are essentially problem-specific in the sense that different maps (translation, dilatation) should be employed in different settings: translation maps will be enough in some simple periodic setting, while dilatation maps are more suited in the case of non periodic boundary conditions. In any case it is important to stress that, independently on the particular expression of the preprocessing map at hand, the main goal of these maps is to align some feature of the solution of the problem under consideration (e.g. a shock or a peak) to a fixed point. In [37], a family of translations $F_\eta(x) = x - \gamma(\eta)$, $x \in \Omega \subset \mathbb{R}$, $\gamma(\eta) \in \Omega$, is employed for the viscous Burgers equation, parametrized by the parameter η: there, $\gamma(\eta)$ is chosen in such a way that the steepest points of the solutions coincide. Specific choices of $\mathcal{F}_z$ will be discussed alongside the problem of interest. We remark here that one may choose different families $\mathcal{F}_z$ for different components $z(\eta)$ of the solution $\mathbf{z}(\eta)$; however, as we shall see more in detail in Section 4.4, there may be no need of doing this: indeed, for the FSI problem that we are going to consider, we will see that all the components of the solution exhibit the same behaviour, i.e. the one of a travelling wave. This will turn out to be very useful and cost effective, both in the offline phase, where we need to learn the position of just one out of the n components of the solution, with $n = 6$ in our case, and in the online phase, where we will be able to employ a single family of *one*-parameter mappings, instead of n families of mappings.
We introduce the following:

Definition 4.2.1. *The preprocessed solution manifold $\mathcal{M}_{\mathcal{F}_z}$ is:*

$$\mathcal{M}_{\mathcal{F}_z} = \{z(\eta) \circ F_\eta^{-1}, \quad \eta \in \mathcal{E}\}.$$

Assuming that $\mathcal{F}_z$ is carefully chosen, $\mathcal{M}_{\mathcal{F}_z}$ has a smaller Kolmogorov n-width, compared to the one of $\mathcal{M}_z$. The practical realization of this preprocessing procedure is incorporated in the offline phase. Given a discrete training set $\mathcal{E}_{train}$, we compute each solution component $z(\eta_{train})$ associated to any $\eta_{train} \in \mathcal{E}_{train}$. The discrete approximations of the corresponding standard and preprocessed solution manifolds

$$\mathcal{M}_z^{tr} = \{z(\eta_{train}), \quad \eta_{train} \in \mathcal{E}_{train}\},$$
$$\mathcal{M}_{\mathcal{F}_z}^{tr} = \{z(\eta_{train}) \circ F_{\eta_{train}}^{-1}, \quad \eta_{train} \in \mathcal{E}_{train}\},$$

provide snapshots for a compression by a POD. The compression is here applied to both $\mathcal{M}_z^{tr}$ and $\mathcal{M}_{\mathcal{F}_z}^{tr}$ to provide a comparison between the standard offline phase and one with preprocessing, but in practical computations one would neglect the compression of $\mathcal{M}_z^{tr}$, as it is understood that $\mathcal{M}_{\mathcal{F}_z}^{tr}$ would result in a POD basis set $\{\Phi_i\}_{i=1}^N$ of lower dimension. In this Chapter we are going to focus mainly on the offline part of the reduced order method; results concerning the online phase are presented for the FSI problem in Section 4.4, where we aim at highlighting the good performance of the whole algorithm. Assume now that an accurate approximation $z_N(t^n)$ of the solution component $z(t^n)$ at time t^n is known, as a linear combination of our basis

functions Φ_i:

$$z_N(t^n) = \sum_{i=1}^{N} \alpha_i^n \Phi_i \circ F_{\eta_{train}^n}.$$

In order to recover $z_N(t^{n+1})$ as an expansion:

$$z_N(t^{n+1}) = \sum_{i=1}^{N} \alpha_i^{n+1} \Phi_i \circ F_{\eta_{train}^{n+1}},$$

the idea proposed in [37] is to iterate between the search for the reduced coordinates α_i^{n+1} and the search for a suitable parameter η_{train}^{n+1}. This procedure is carried out by means of a minimization problem of the L^2 norm of the residual evaluated at $\sum_{i=1}^{N} \alpha_i^{n+1} \Phi_i \circ F_{\eta_{train}^{n+1}}$. In this Chapter, in the presentation of the online phase for the FSI problem, we do not employ an L^2 norm minimization, as it was first suggested in Cagniart et al. [37]: we employ instead a polynomial interpolation method to learn the best suited parameter η_{train}^{n+1} to be used in the online phase.
To summarize, here is an outline of the general reduced basis method with preprocessing of the snapshots:

- compute the snapshots $z(\eta_{train})$ for $\eta_{train} \in \mathcal{E}_{train}$;
- build the family of preprocessing maps $\mathcal{F}_z = \{F_\eta,\ \eta \in \mathcal{E}\}$;
- preprocess the snapshots and obtain the preprocessed solution manifold $\mathcal{M}_{\mathcal{F}_z}^{tr}$;
- run a POD on $\mathcal{M}_{\mathcal{F}_z}^{tr}$ and obtain the basis functions $\{\Phi_i\}_{i=1}^N$;
- during the online phase, at timestep t^{n+1}, find $z_N(t^{n+1}) = \sum_{i=1}^{N} \alpha_i^{n+1} \Phi_i \circ F_{\eta_{train}^{n+1}}$, where η_{train}^{n+1} is a suitable parameter.

In the next two sections we will see two applications of this preprocessing procedure: the first problem we are going to study is a computational fluid dynamics (CFD) problem, and the second problem is a Fluid–Structure Interaction problem. These two applications are quite different for what concerns the physics behind them (the latter is a coupled multiphysics problem, the former is not), but they both feature a slow decay of the eigenvalues returned by running a POD on $\mathcal{M}_z^{tr}$. The first test case is characterized by the change of the direction of propagation of a vortex close to a rotating cylinder, a feature which is difficult to reproduce with a small number of basis functions. The second problem is a transport dominated problem, where the solution behaves like a wave travelling in the domain. Also in this case the travelling wave would be hard to be reconstructed with a standard model order reduction technique.

4.3 A CFD test case

In this Section we are going to show the results that we obtain with the preprocessing procedure on a CFD test case, namely a fluid past a rotating cylinder. The problem described in the following is inspired by the one presented in Cagniart's thesis [35], although our formulation is slightly different: in the reference, the direction of the fluid velocity at the inlet boundary is changing in time, and the cylinder is kept fixed, whereas on the contrary in our problem the inlet direction is constant, but the cylinder is rotating.

4.3.1 Problem formulation

In this problem we have a flow past a rotating cylinder; this situation, with or without the rotation feature, is quite interesting and has indeed been studied in a large number of papers, both in the incompressible case [21, 164, 99, 165] and in the compressible regime [170]. If the Reynolds number Re is greater than 47, then a vortex shedding phenomenon occurs [99]; when, in addition, the cylinder is rotating, it might happen that we see a change in the direction of propagation of this vortex. This phenomenon is quite complicated and is strictly related to a variety of different physical quantities; in fact, it is related not only to Re, but also to the cylinder rotation rate α, which is defined as:

$$\alpha = \frac{D\omega}{2U_\infty},$$

where D is the diameter of the cylinder, ω is the angular velocity of a point on the surface of the cylinder and U_∞ is the oncoming free stream velocity. For $\alpha < \alpha_L$ vortex shedding occurs, where α_L is a critical value that is a function of Re. We refer to [164] for different values of α related to different values of the Reynolds number. To model our problem we use the Navier-Stokes equation, with incompressibility constraint. Figure 4.1 shows the physical domain Ω for our problem. We do not take into account any physical parameter. Our problem reads as follows: for any time $t \in [0, T]$ find $\boldsymbol{u}_f(\cdot; t)\colon \Omega \mapsto \mathbb{R}^2$ and $p_f(\cdot; t)\colon \Omega \mapsto \mathbb{R}$ such that:

$$\begin{cases} \partial_t \boldsymbol{u}_f + (\boldsymbol{u}_f \cdot \nabla)\boldsymbol{u}_f - \frac{1}{Re}\Delta \boldsymbol{u}_f + \nabla p_f = \boldsymbol{b}_f & \text{in } \Omega \times [0,T], \\ -\text{div}\boldsymbol{u}_f = 0 & \text{in } \Omega \times [0,T], \\ \boldsymbol{u}_f = \boldsymbol{u}_{in} & \text{in } \Gamma_{in} \times [0,T], \\ \boldsymbol{u}_f = \boldsymbol{u}_{tan} & \text{in } \Gamma_{cyl} \times [0,T], \\ p_f \boldsymbol{n} - \frac{1}{Re}\nabla \boldsymbol{u}_f \cdot \boldsymbol{n} = \boldsymbol{0} & \text{in } (\Gamma_{top} \cup \Gamma_{bottom} \cup \Gamma_{out}) \times [0,T]. \end{cases}$$

We impose homogeneous Neumann conditions on top and bottom walls Γ_{top} and Γ_{bottom}, and also on the right boundary Γ_{out}, as all these boundaries are considered as outlets due to the fact that the vortex sheet rotates alongside with the cylinder. We impose a Dirichlet condition $\boldsymbol{u}_f = \boldsymbol{u}_{in}$ on Γ_{in}, where $\boldsymbol{u}_{in}$ is a fixed horizontal inflow, see Table 4.1. Furthermore, $\boldsymbol{u}_{tan}$ is the tangential velocity at the surface of the rotating cylinder. At the beginning of the simulation the cylinder is not moving, and it stays still until a fully developed vortex shedding phenomenon is reached; after that the cylinder starts to rotate counterclockwise, first with a constant acceleration β, then it keeps on rotating with a constant angular velocity. Thus, denoting by ω be the angular velocity of the cylinder, we set

$$\omega = \begin{cases} \omega_0 = 0 & \text{for } t \in [0, t_1], \\ \omega_t = \omega_0 + \beta(t - t_1) & \text{for } t \in [t_1, t_2], \\ \omega_f = \omega_{t_2} & \text{for } t \in [t_2, T]. \end{cases} \tag{4.3}$$

Once we have the angular velocity, we can compute $\boldsymbol{u}_{tan}$ thanks to the relation:

$$||\boldsymbol{u}_{tan}(t)|| = \omega_t r,$$

where r is the radius of the cylinder, and assuming $\boldsymbol{u}_{tan}(t)$ to be tangent to Γ_{cyl}.

As far as the problem formulation concerns, let us mention the fact that, in preparation of a forthcoming online stage, we enrich the space of the fluid velocity snapshots

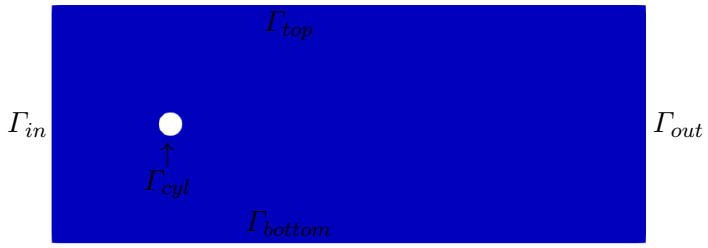


FIGURE 4.1: Physical domain of the CFD test case. The cylinder is depicted in white, the fluid domain in blue.

Physical constants	Value
r	2 cm
$\boldsymbol{u}_{in}$	$(1, 0)$ m/s
Re	100
$\boldsymbol{b}_f$	0
β	0.025 rad/s^2
Time constants	**Value**
Δt	0.25 s
t_1	75 s
t_2	95 s
T	145 s

TABLE 4.1: Constants values for the problem of a flow past a rotating cylinder.

$\boldsymbol{u}_f$ with some supremizers snapshots $\boldsymbol{s}$: the supremizer snapshots $\boldsymbol{s}$ are needed at the reduced order level in order to have a more stable approximation of the fluid pressure. For further details on the formulation of the supremizer in the POD framework of parametrized fluid flows we refer to [12]. In Table 4.1 we can find the problem data that we used in our simulation. After we obtain a fully developed Karman vortex, and after the cylinder starts to rotate, we have a noticeable change in the direction of propagation of the vortex. This change of direction may hinder the representation of the solution by a small number of basis functions, and this expectation is confirmed by running a POD on the fluid velocity snapshots collection, as we can see from Figure 4.2.

4.3.2 Preprocessing step

We will focus on the preprocessing of the fluid velocity $\boldsymbol{u}_f$, since it is more straightforward to visualize the direction of propagation of the Karman vortex and hence understand the idea beneath the deformation map. First of all, let us notice from Figure 4.3 that, when building the mesh, we have defined a fictitious subdomain Ω_{int} (in red). It is very important to choose the radius r of the circular subdomain Ω_{int} in such a way that Ω_{int} is entirely contained in the physical domain Ω and yet Ω_{int} is able to capture all the complex behaviour of the solution that is strictly related to the rotation of the cylinder. In fact the blue subdomain Ω_{ext} in Figure 4.3 deals with the small vortexes that originate in the wake of the cylinder before it starts to rotate: these vortexes propagate in the domain and their behaviour is not heavily affected by the rotation of the cylinder, and therefore we should be able to reproduce their behaviour with a coarse set of basis functions. Therefore as the solution features

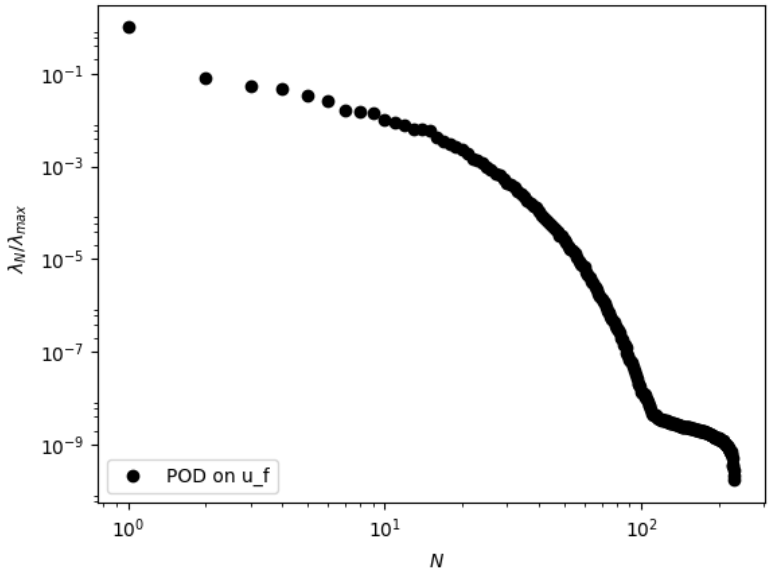


FIGURE 4.2: Decay of the eigenvalues for the fluid velocity past a rotating cylinder.

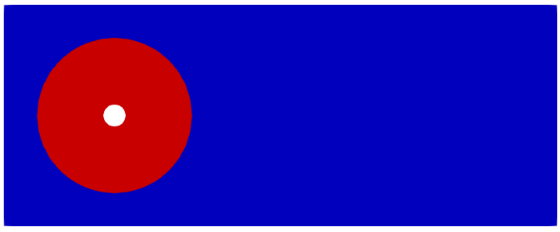


FIGURE 4.3: Subdivision of the physical domain Ω into Ω_{int} (red) and Ω_{ext} (blue).

the most interesting phenomena in a neighborhood of the cylinder Γ_{cyl}, we are not going to preprocess the entire snapshot, but we are going to focus instead just on its restriction on Ω_{int}. This procedure is more adapted, in the sense that we are not considering the solution manifold as a global entity, but we think of it as made of two manifolds, $\mathcal{M}_{ext}$ and $\mathcal{M}_{int}$: this division is based on the division of the physical domain of interest Ω into two subdomains Ω_{ext} and Ω_{int}. Therefore:

$$\mathcal{M}_{ext} = \{\boldsymbol{u}_f(\cdot, t)|_{\Omega_{ext}}, \quad t \in [0, T]\}, \tag{4.4}$$

$$\mathcal{M}_{int} = \{\boldsymbol{u}_f(\cdot, t)|_{\Omega_{int}}, \quad t \in [0, T]\}. \tag{4.5}$$

Being Ω_{ext} far from the rotating cylinder, and therefore far from the rotating phenomenon, we expect $\mathcal{M}_{ext}$ to be a better behaved solution manifold with respect to $\mathcal{M}_{int}$. This means that $\mathcal{M}_{ext}$ has a small Kolmogorov n-width, and does not need any preprocessing. On the contrary we will focus on $\mathcal{M}_{int}$, which will have a slowly decaying Kolmogorov n-width. Before going any further, let us remark that the subdivision of $\mathcal{M}_{\mathbf{u}_f}$ in $\mathcal{M}_{int}$ and $\mathcal{M}_{ext}$ is strictly dependent on the subdivision of Ω, therefore the fictitious cylinder has to be chosen wisely, in such a way that Ω_{int} is able to capture most of the rotating phenomenon. In particular, in our simulations, we have chosen Ω_{int} to be a cylinder of radius r, 7 times larger than the radius of the physical cylinder.
For the preprocessing step therefore we first restrict the snapshots to the subdomain Ω_{int}. Then, we want to build a (one-parameter) family of smooth and invertible maps

$$\mathcal{F}_{\boldsymbol{u}_f} = \{F_t \colon \Omega_{int} \to \Omega_{int}, \ t \in [0, T]\},$$

where:

$$F_t(x, y) = \begin{pmatrix} \cos\theta(t) & \sin\theta(t) \\ -\sin\theta(t) & \cos\theta(t) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{4.6}$$

and where $\theta(t)$ is the parameter identifying each map. In our case it is natural to choose, at each time t, $\theta(t)$ to be the angle spanned by the direction of propagation of the vortex (obtained through a postprocessing of the solution $\boldsymbol{u}_f(t)$) and the horizontal axis. The idea behind (4.6) is pretty straightforward: at each timestep t of our simulation, we compute how much the vortex has changed its direction of propagation, and we therefore find $\theta(t)$. After that, in the preprocessing step we take all the snapshots $\boldsymbol{u}_f(t)$, we restrict them to the subdomain Ω_{int}, and then we rotate them back to a horizontal direction of propagation of the vortex with $\boldsymbol{u}_f(t) \circ F_t^{-1}$.

Figure 4.4 shows some results of the preprocessing procedure. On the left column we find the original snapshots: as the time increases, the direction of propagation of the vortex changes. On the column to the right instead we see the corresponding preprocessed snapshots: the direction of propagation of the vortex has been rotated back to a horizontal direction. In Figure 4.5 we see the improvements (in terms of POD eigenvalues decay) that we get by applying the rotation to the snapshots: as we can see, after the preprocessing we obtain an improvement e.g. of almost 2 orders of magnitude comparing the 10-th eigenvalue of standard and preprocessed manifolds.

4.4 A multiphysics problem

We are now interested in applying the preprocessing procedure to a Fluid–Structure Interaction problem, whose solution exhibits a transport dominated behaviour. The

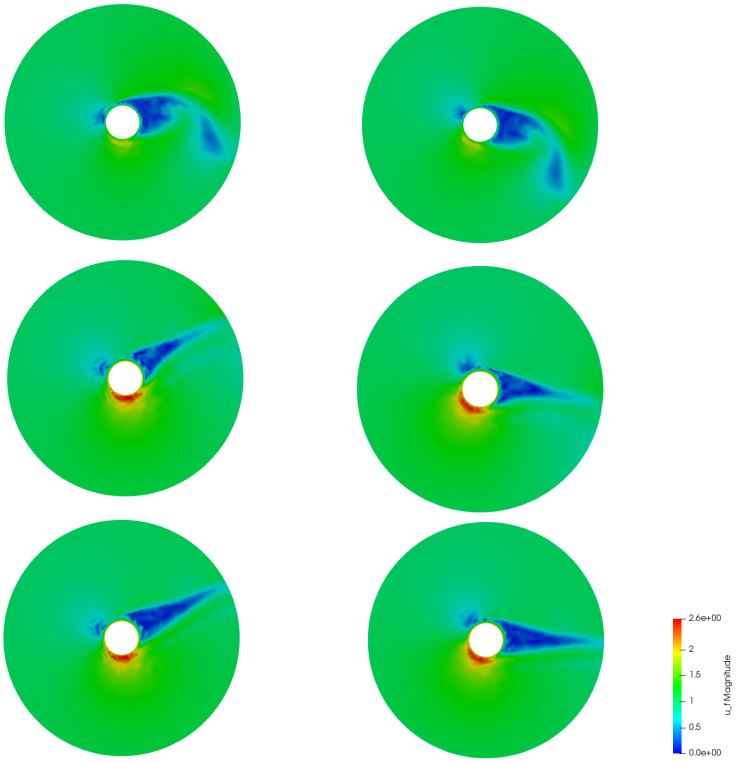


FIGURE 4.4: Fluid velocity snapshots at different iterations $i = 350, 444, 580$ (top to bottom) before the preprocessing (left column) and after the preprocessing (right column).

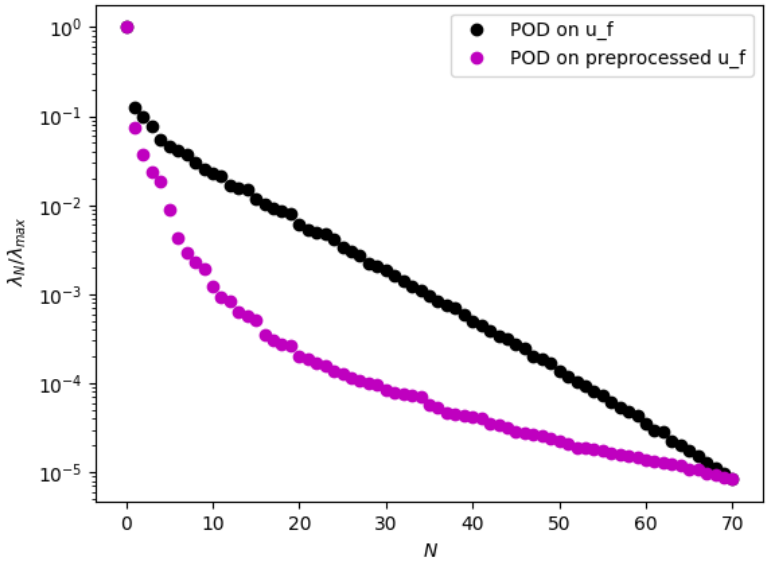


FIGURE 4.5: POD comparison before (black) and after (magenta) the preprocessing of $\mathbf{u}_f$.

FIGURE 4.6: Physical domain (reference configuration): fluid subdomain (blue) and structure subdomain (red). The fluid-structure interface coincides with the structure in our case; the structure has been magnified for visualization purposes.

problem formulation features an Arbitrary Lagrangian Eulerian (ALE) formulation, which has been introduced in Section 2.1.1. In this case we decide to adopt a different approach to solve the problem: instead of using a partitioned procedure, which has already been exploited in the previous Chapter, we employ here a monolithic approach, see for example [13]. For further details on reduced order models and applications of FSI problems we refer to [26, 111, 48].

4.4.1 Problem formulation

A two dimensional rectangle of height h_f and length L is filled with a Newtonian fluid. The top and the bottom walls of the rectangle represent the structure, which is considered to be deformable, and its thickness is negligible with respect to the height of the rectangle. Since the structure is thin, it is described by a one dimensional model. We further assume that the displacement of the walls in the horizontal direction is negligible, and hence the structure presents only vertical motion: the behaviour of the compliant walls is therefore described by the *generalized string model* [142, 147]. We want to describe the behaviour of the solution (and the domain itself) in the time interval $[0, T]$. Let $\Sigma(t)$ and $\Omega_f(t)$ be, respectively, the structure and the fluid domain at time t. Let $\hat{\Sigma}$ be the solid reference configuration (undeformed walls), and let $\hat{\Omega}_f$ be the fluid reference domain: for convenience we take $\hat{\Omega}_f = \Omega_f = \Omega_f(t = 0)$ (the blue fluid domain in Figure 4.6). Thanks to the introduction of the ALE map $\mathcal{A}_f(t)$ defined in Section 2.1.1, we can map the fluid equations back to the reference domain $\hat{\Omega}_f$. Let $\boldsymbol{F}$ be the gradient of $\mathcal{A}_f(t)$, and let J be the Jacobian. The Fluid–Structure Interaction problem, formulated on the reference configuration, reads: find fluid velocity $\hat{\boldsymbol{u}}_f(t)\colon \hat{\Omega}_f \to \mathbb{R}^2$, fluid pressure $\hat{p}_f(t)\colon \hat{\Omega}_f \to \mathbb{R}$, and structure displacement $\hat{d}_s(t)\colon \hat{\Sigma} \to \mathbb{R}$ such that:

$$\begin{cases} J\rho_f(\partial_t \hat{\boldsymbol{u}}_f + \boldsymbol{F}^{-1}((\hat{\boldsymbol{u}}_f - \partial_t \hat{\boldsymbol{d}}_f)\cdot\nabla)\hat{\boldsymbol{u}}_f) - \operatorname{div}(J\hat{\sigma}^f \boldsymbol{F}^{-T}) = \boldsymbol{b}_f & \text{in } \hat{\Omega}_f \times [0,T], \\ \operatorname{div}(J\boldsymbol{F}^{-1}\hat{\boldsymbol{u}}_f) = 0 & \text{in } \hat{\Omega}_f \times [0,T], \\ \rho_s h_s \partial_{tt}\hat{d}_s - c_0\partial_{xx}\hat{d}_s + c_1\hat{d}_s = -\hat{\sigma}^f(\hat{\boldsymbol{u}}_f, \hat{p}_f)\boldsymbol{n}\cdot\boldsymbol{n} & \text{in } \hat{\Sigma}. \end{cases} \tag{4.7}$$

Here ρ_f is the fluid viscosity, ρ_s is the structure viscosity, h_s is the structure height (or thickness), c_0 and c_1 are the structure constitutive parameters. $\hat{\sigma}^f$ is the Cauchy stress tensor for the fluid, in the reference configuration, and is defined as:

$$\hat{\sigma}_f(\hat{\boldsymbol{u}}_f, \hat{p}_f) = -\hat{p}_f I + \rho_f \nu_f \left(\nabla\hat{\boldsymbol{u}}_f \boldsymbol{F}^{-1} + \boldsymbol{F}^{-T}\nabla^T\hat{\boldsymbol{u}}_f\right).$$

ν_f being the kinematic viscosity of the fluid and I being the 2×2 identity matrix.

Also in this Chapter, in order to ease the notation, we drop the $\hat{}$, since everything will be formulated and treated in the reference configuration. Since it is a Fluid–Structure Interaction problem, we need some coupling conditions. Let us denote by

Physical constants	Value
ρ_f	1 g/cm^3
E_s	0.75×10^6 dyn/cm^2
ν_f	0.035 Poise
b_f	0
$p_{in}(t)$	$10^3 \times [1 - \cos(\frac{2\pi t}{T_{in}})]\chi_{[0,0.0025]}$
$p_{out}(t)$	0
ν_s	0.5
ρ_s	1.1 g/cm^3
c_0	$\frac{h_s E_s}{2(1+\nu_s)}$
c_1	$\frac{h_s E_s}{h_f^2(1-\nu_s^2)}$
h_s	0.1 cm
Time constants	**Value**
T_{in}	2.5×10^{-3} s
ΔT	10^{-4} s
N_T	150 s

TABLE 4.2: Problem data for the test case of a flow in a channel with deformable walls.

$\boldsymbol{n}$ the outward unit normal to Σ; we have:

$$\begin{cases} \boldsymbol{d}_f = d_s \boldsymbol{n} & \text{in } \Sigma, \quad \text{continuity of the displacement,} \\ \boldsymbol{u}_f = \partial_t d_s \boldsymbol{n} & \text{in } \Sigma, \quad \text{continuity of the velocity.} \end{cases} \tag{4.8}$$

Together with the coupling conditions we also have to give some boundary and some initial conditions. For the latter we assume that at time $t = 0$ the system is at rest; the boundary conditions can be summarized in the following system:

$$\begin{cases} \sigma_f(\boldsymbol{u}_f, p_f)\boldsymbol{n} = -p_{in}(t)\boldsymbol{n} & \text{in } \Gamma_{in} \times (0,T], \\ \sigma_f(\boldsymbol{u}_f, p_f)\boldsymbol{n} = -p_{out}(t)\boldsymbol{n} & \text{in } \Gamma_{out} \times (0,T], \\ d_s = 0 & \text{in } \partial\Sigma \times [0,T], \end{cases} \tag{4.9}$$

where third condition says that the structure is fixed at its extremities.
As far as the problem formulation concerns, let us remark that we adopted a supremizer enrichment technique also in this multiphysics test case, always to obtain, in the POD framework, a set of basis functions that allows for a stable approximation of the fluid pressure.

4.4.2 Transport dominated FSI problem

Problem data used for the simulation of our test case can be found in Table 4.2: corresponding values are taken from the numerical results presented in [168, 127].

The behaviour of the fluid pressure p_f and of the extended displacement $\boldsymbol{d}_f$ is shown in Figure 4.7 and Figure 4.8 respectively. Our problem is transport dominated: if we look at Figure 4.7 for example, the change in time of the position of the peak of the pressure wave will be a difficult feature to capture at the reduced order level with just a few modes. This expectation is finally confirmed at the numerical level, as we

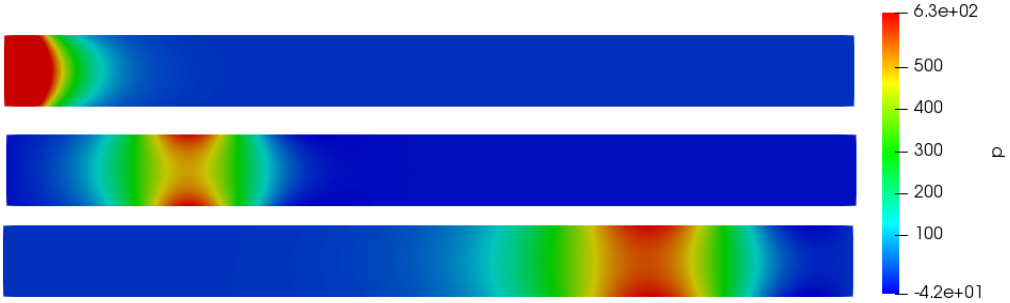


FIGURE 4.7: Fluid pressure behaviour: the solution is pictured here at time $t = 0.001$, $t = 0.005$ and $t = 0.015$. The peak of the wave is propagating into the domain, creating a transport phenomena.

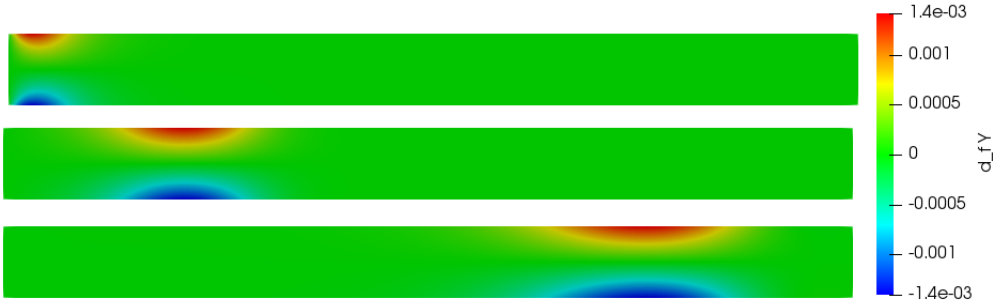


FIGURE 4.8: Vertical component of fluid displacement behaviour: again the solution is pictured at time $t = 0.001$, $t = 0.005$ and $t = 0.015$. The peak of the wave is still very small at the beginning, it grows for some time and then it starts to propagate.

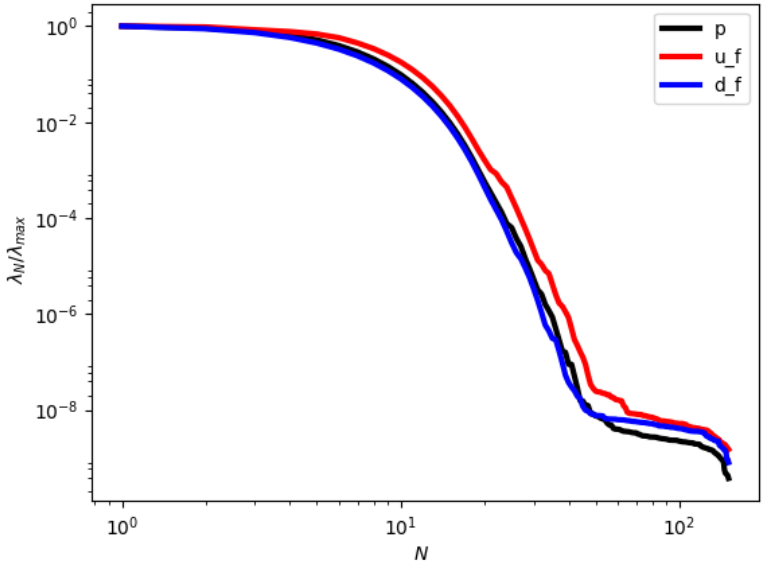


FIGURE 4.9: Decay of the eigenvalues for the POD on the fluid pressure (black line), fluid displacement (blue line) and fluid velocity (red line).

can see in Figure 4.9, by running a POD on p_f, $\boldsymbol{d}_f$, and also $\boldsymbol{u}_f$. Therefore we rely on a transformation on the set of solutions, in order to compensate this transport phenomenon.

Remark 4.4.1. In order to make the following exposition more clear and easy to read, from now on we focus only on a particular component of the solution of our problem, namely the fluid pressure. It is anyway important to keep in mind that, based on our simulation, all the components of the solution to the FSI problem are subject to a transport phenomenon, and hence every consideration that we are going to make on p_f could be easily applied to any of the other components of the solution.

4.4.3 Preprocessing step

Let us see more in detail how to apply this preprocessing procedure to the fluid pressure solution manifold $\mathcal{M}_{p_f}$. Figure 4.7 shows how the peak of the pressure is transported in the domain. We would like to align the peaks at all time steps in a reference configuration; in this way, we obtain a set of snapshots where the pressure wave is not moving at all. In this case therefore a low number of modes will be sufficient to give a good representation of the situation.

Starting from this observation, we build a one parameter family of mappings:

$$\mathcal{F}_{p_f} = \{F_t : \Omega_f \to \Omega_f, \quad t \in [0, T]\}$$

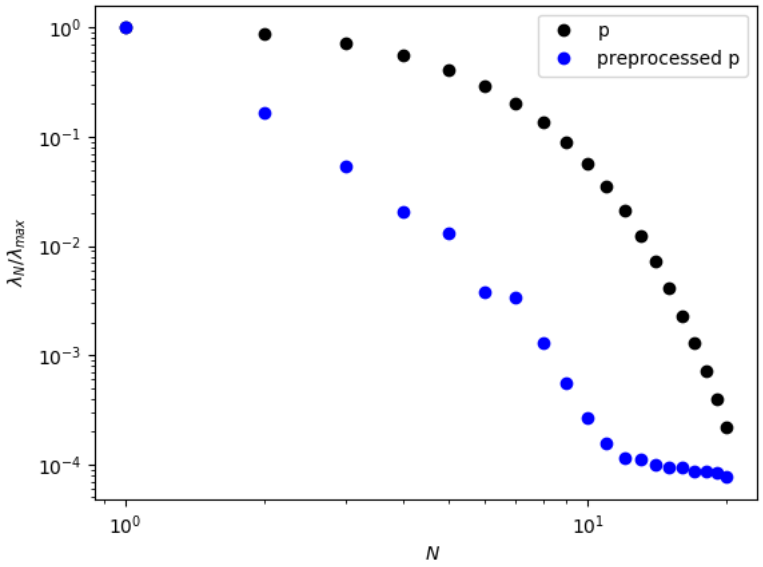
such that for every t in $[0, T]$, the peak of $p_f(F_t^{-1}(\cdot), t)$ is located not at its original position, but is instead moved to the middle of the domain. In this way the new snapshots $p_f(F_t^{-1}(\cdot), t)$ will all have the peak located at the exact same position, meaning the middle of the domain. When building the map F_t^{-1}, another aspect to which we should pay great attention is the boundary conditions. Since we are not working in a periodic setting, we want to make sure that the preprocessed snapshots satisfy the same boundary conditions as the original snapshots. An easy way to make sure that these requirement is satisfied is to keep the points in Γ_{in} and the points in Γ_{out} fixed.

A suitable map to be used in this problem (non periodic setting) is a dilatation map F_t^{-1}:

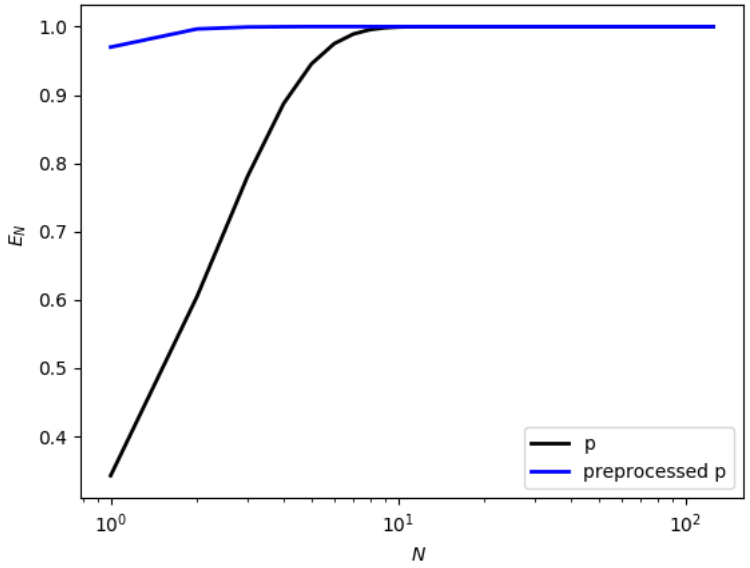
$$F_t^{-1}(x) = \frac{3x\gamma}{x(\gamma - 3) + 3(6 - \gamma)},$$

where $\gamma = \gamma(t)$ is the abscissa of the position of the peak of the wave at time t. We assume that the abscissa of the points on the inlet boundary Γ_{in} is $x = 0$ and the length of the domain Ω_f is $L = 6$; in addition, the position in which we are moving the peak of the wave at every time t is exactly in the center of the domain. Let us remark that the map F_t^{-1} is just a stretching in the horizontal direction: this is due to the fact that we do not have any transport phenomena in the vertical direction, and hence there is no need for a transformation in the $y-$axis. So, with an abuse of notation, we can think of $F_t^{-1}(x, y)$ as $F_t^{-1}(x, y) = (F_t^{-1}(x), y)$.

Our family of mappings $\mathcal{F}_{p_f}$ is a one-parameter family, therefore to identify the stretching map F_t^{-1} we have to identify the parameter $\gamma(t)$ at time t. One simple way of doing this is to use polynomial interpolation: we approximate the physical law governing the position of the peak of the wave $\gamma(t)$ with a polynomial $q_k(t)$ of degree k. Of course, when the physical phenomenon under investigation becomes particularly complex, also in presence of other parameters with respect to time, other techniques

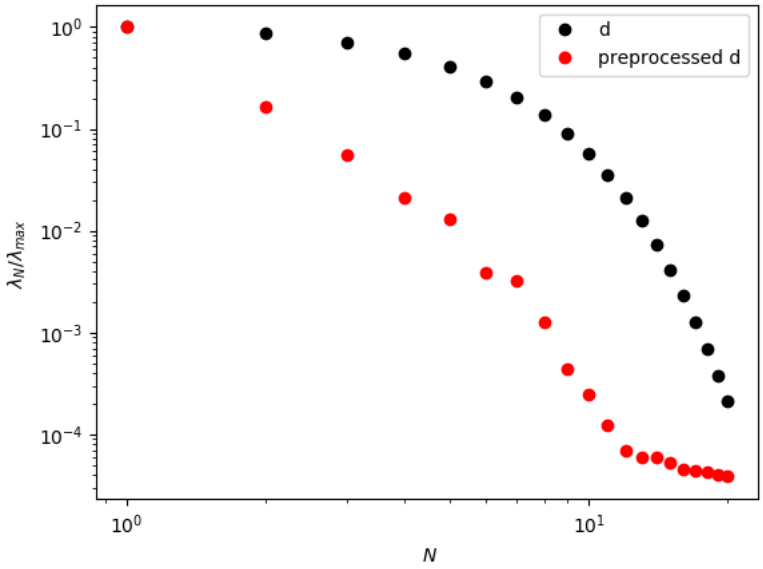


(a) Decay of the eigenvalues for the fluid pressure.

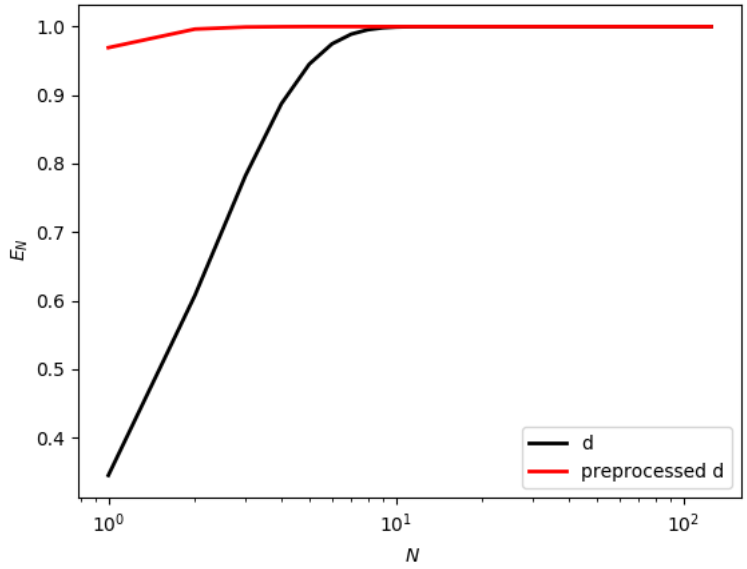


(b) Retained energy for the fluid pressure.

FIGURE 4.10: Top: comparison between the rate of decay of the eigenvalues for the pressure with and without preprocessing. Bottom: retained energy as a function of the number N of POD modes



(a) Decay of the eigenvalues for the displacement.



(b) Retained energy for the displacement.

FIGURE 4.11: Top: comparison between the rate of decay of the eigenvalues for the displacement with and without preprocessing. Bottom: retained energy as a function of the number N of POD modes

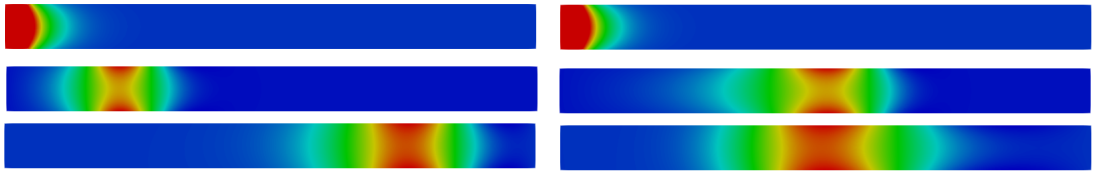


FIGURE 4.12: Original snapshots for p_f at time $t = 0.001$, $t = 0.005$ and $t = 0.015$ (left column), and corresponding preprocessed snapshots (right column).

may become more useful and performing, such as, for example, the polynomial regression or the artificial neural network presented in [171]. In our case, however, we see that a polynomial of degree $k = 1$ is sufficient to obtain very good results in the online phase. First of all we compute all the snapshots $p_1, \dots, p_{N_T}$, where $N_T = \frac{T}{\Delta t}$, $p_i = p_f(t_i)$ and $t_i = i\Delta t$. Once we have the snapshots, we compute the peak $\gamma_{T_{in}}$ and γ_{N_T}. The reason for choosing $\gamma_{T_{in}}$, instead of a more straightforward γ_0, is that usually in CFD applications, complex phenomena such as vortex formation, propagation of a quantity, require some time to develop, unlike the wave propagation problem presented in [37], or [171] for example. For our problem, T_{in} is a reasonable starting time for the preprocessing procedure: recall the expression of p_{in} from Table 4.2; as we can see, p_{in} represents a given pulse at the inlet boundary, which is nonzero up to time $t = T_{in} = 0.0025$. So, up to time $t = 0.0025$ the pulse makes the wave grow; when the pulse is null, the wave starts to propagate into the domain. At the reduced level this translates into the fact that, up to $t = T_{in}$ we adopt a standard reduction technique, using a standard reduced basis obtained with a standard POD on the first T_{in} snapshots. From the snapshot T_{in} to the last one, on the other hand, we first perform the preprocessing step, and then perform a POD.
In Figure 4.10 (a) we can compare the decay of the eigenvalues for the POD on the pressure with and without this preprocessing procedure. As we can see, with the preprocessing technique we do actually get an improvement in the decay of the eigenvalues, and in fact with less than 15 modes we reach a level of 10^{-3}, which is one order of magnitude less than the one we get with less than 15 modes in the standard case. We get the same results for the extended displacement $\boldsymbol{d}_f$, see Figure 4.10 (a). As we can see from Figure 4.10 (b) and 4.11 (b), with the preprocessing procedure, the first modes are able to capture almost 70% more energy with respect to the modes without preprocessing.

4.4.4 Online phase

In the previous paragraph we have briefly explained the preprocessing procedure adopted for the fluid pressure, which is nevertheless just one out of the many components of the solution of our coupled problem. We therefore stress the fact that interpolating with the polynomial $q_1(t)$ the position of the peak $\gamma(t)$ of the pressure wave $p(t)$ is enough for the whole preprocessing procedure to be applied to our multiphysics problem. Indeed, since the problem is coupled, the behaviour of each component of the solution is influenced and influences the others components. Thus, a peak in the pressure translates in a peak in the displacement of the solid and in a peak in the velocity of the fluid. Therefore with $q_1(t)$ we are able to perform a preprocessing of all the other components.
Once we have performed a standard POD on the preprocessed solution manifold $\mathcal{M}^z_{\mathcal{F}}$, where z denotes any of the components of the solution, we have at hand a set of preprocessed reduced spaces $V^z_N = \text{span}\{\Phi^z_j\}_{j=1}^N$. Then the reduced approximation z_N of the component z is defined as:

$$z_N(t_i) = \sum_{j=1}^{N} \alpha^i_j \Phi^z_j \circ \tilde{F}_{t^i}$$

where $\tilde{F}_{t^i}$ is the inverse map of the dilatation map previously introduced, with the important difference that now we are replacing the exact position of the peak $\gamma_i = \gamma(t^i)$

FIGURE 4.13: Pressure snapshots (left column) at time $t = 0.005$ (top) and final time $t = 0.015$ (bottom). Reduced order pressure simulation (right column) at time $t = 0.005$ and time $t = 0.015$. The reduced simulation has been obtained with $N = 4$ basis functions, for each component of the solution of the FSI problem.

FIGURE 4.14: Displacement snapshots (left column) at time $t = 0.005$ (top) and final time $t = 0.015$ (bottom). Reduced order displacement simulation (right column) at time $t = 0.005$ and time $t = 0.015$. The reduced simulation has been obtained with $N = 4$ basis functions, for each component of the solution of the FSI problem.

with $\tilde{\gamma}_i$, a *prediction* of the position of the peak of the wave at time t_i, obtained by simply evaluating $q_1(t_i)$.
We remark that, in order to have a more stable approximation of the pressure at the reduced order level, we enrich the reduced space associated to the fluid velocity with some supremizer modes, therefore $V_N^{u_f} = \text{span}\{\Phi_1^u, \dots, \Phi_{N_u}^u, \Phi_1^s, \dots, \Phi_{N_s}^s\}$.

4.4.5 Numerical results

As we can see from Figure 4.13 and Figure 4.14, the online step of the method performs very well. These results are even more striking, given the fact that we were able to obtain them by using just $N = 4$ modes for each component of the solution. Since in total we have 6 unknowns for our FSI problem (remember also the two Lagrange multipliers used to impose the coupling conditions), we use $N_{tot} = 24$ basis functions. To obtain similar results, but without the preprocessing procedure, we need to use at least 10 basis functions for each component, meaning at least 60 modes in total. Thus during the online phase we are decreasing the dimension of the system to be solved of at least 40. These considerations are finally confirmed by an error behaviour analysis, as shown in Figures 4.15 and 4.16. In these figures we can see that the overall behaviour of the relative error is improved by the preprocessing procedure. In Figure 4.15 the oscillating behaviour of the relative error without the preprocessing shows how the set of basis functions struggles to reproduce the exact location of the peak of each component of the solution. On the other end Figure 4.16 shows that the mean relative error is very low with just 4 basis functions for each component of the solutions.

4.5 CFD test case with a physical parameter

So far we have considered test cases where the only parameter was time, nevertheless we are interested in investigating what happens if we add another physical parameter

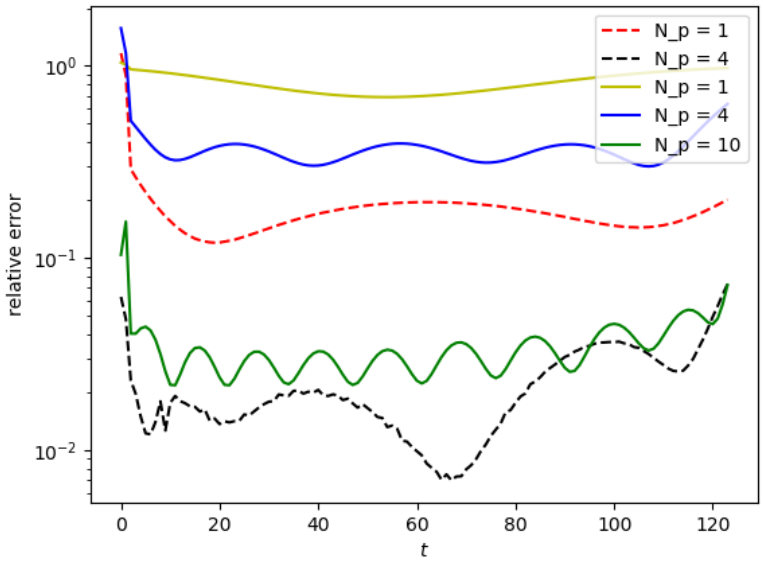


FIGURE 4.15: Analysis of the behaviour of the relative error for the fluid pressure approximation. Dashed lines were obtained employing the preprocessing procedure, continuous lines were obtained using a standar MOR.

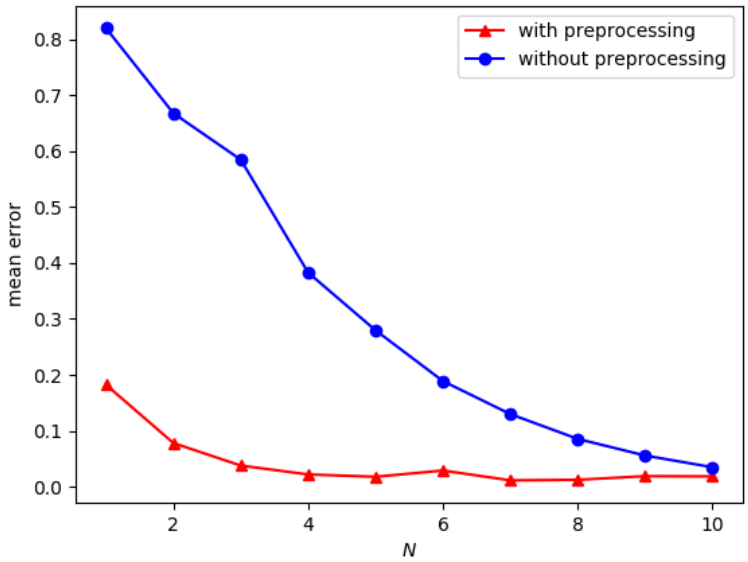


FIGURE 4.16: Behaviour of the mean relative error for the fluid pressure, depending on the number N of basis functions employed, with preprocessing (red) and without preprocessing (blue).

to the problem. We then go back to the CFD test case presented in Section 4.3, but now the Reynolds number Re will be considered as a parameter. The fluid velocity solution manifold will be defined as:

$$\mathcal{M}_{\boldsymbol{u}_f} = \{\boldsymbol{u}_f(t,\mu), \quad t \in [0,T], \quad \mu \in [Re_{\min}, Re_{\max}]\},$$

where now $\mu = Re$. Of course in this case we have to pay attention not only to the Reynolds number, but also to the rotation rate α of the cylinder, since it is strictly related to the development of a vortex shedding phenomenon. For our particular CFD test case we choose the following parameter range:

$$[Re_{\min}, Re_{\max}] = [47, 150],$$

and we choose $\alpha = 1.0$. We discretize the parameter space, choosing a set of parameter samplings $\{\mu_1, \ldots, \mu_N\}$. After we have obtained a set of snapshots for each parameter in the parameter sampling set, we observe that the change in the Reynolds number leads to changes in the behaviour of the fluid velocity, with the vortex shedding that may occur earlier or later, but in all the situations we see that after a while, due to the rotation of the cylinder, the direction of propagation of the vortex changes.

4.5.1 POD-Greedy

With the addition of a physical parameter, the exploration strategy will be carried out in a different way with respect to a standard POD on the set of snapshots; we are going to explore the parameter space with a pseudo-Greedy algorithm, and we are going to explore in time with a POD on the set of snapshots corresponding to each parameter selected by the pseudo-Greedy strategy. First of all we discretize the parameter space in order to obtain a sampling set of cardinality N of our choice; in order to do so, we choose a Lagrange distribution sampling, i.e:

$$\mu_i = Re_{min} \exp\left(\left(\frac{i-1}{N-1}\right) \log\left(\frac{Re_{max}}{Re_{min}}\right)\right).$$

Once we have the parameter sampling set $\{\mu_1, \ldots, \mu_N\}$, we compute the truth solution for each one of these parameters. Afterwards, the POD is applied in the following way:

1. for μ_1, we run a standard POD on the corresponding snapshots;
2. we now have at hand a set of reduced basis $\{\Phi_1, \ldots, \Phi_{M_1}\}$;
3. for μ_i, $i \geq 2$, we orthogonalize each snapshot $\boldsymbol{u}_f^j(\mu_i)$ with respect to the linear space $\text{span}(\Phi_1, \ldots, \Phi_{M_{i-1}})$;
4. we then run a POD on the set of orthogonalized snapshots, and add the resulting basis functions to the already existing set of basis functions.

We remark that we are not actually using a proper Greedy algorithm, because we are choosing a priori the parameter sampling set; the reason for this "pseudo"-Greedy is that it is beyond the scope of this thesis to introduce and to focus on an error estimator to be used for the Greedy algorithm, and also a simple Lagrange distribution will be sufficient to have an insight on what is going on before and after the preprocessing step. We also remark that different possibilities for the POD are possible: the one we are using here is based on an orthogonalization step (see [85]), where we are getting

rid of the superfluous information before pursuing a POD. Another possibility, which we have used in the previous Chapter, would be to run first a standard POD on the snapshots, for every parameter, and then, at the end, run another standard POD on the set of obtained reduced basis, again to get rid of the superfluous information (see [129]).

4.5.2 Preprocessing step

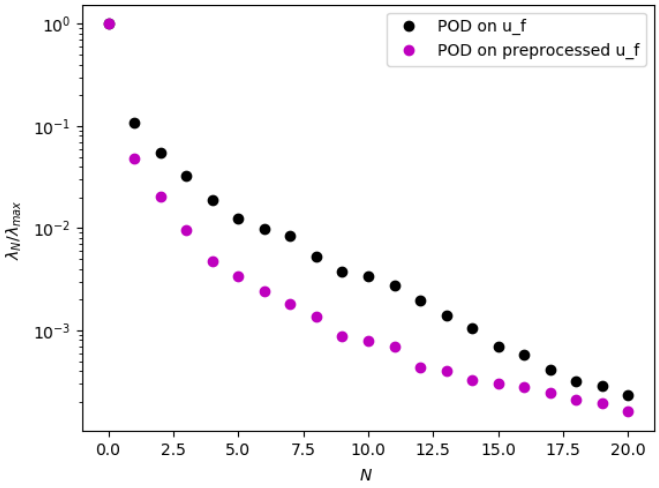
The preprocessing procedure is carried out, for every value of the parameter μ in the discretized parameter space, in the same way it was carried out in the case with no physical parameter: $\forall k \in \{\mu_1, \dots, \mu_N\}$, and for every timestep t_i we compute $\theta_i^k = \theta(t_i, \mu_k)$, which is the angle spanned by the direction of propagation of the vortex and the horizontal axis. Then the deformation map $F^{-1}_{t_i,\mu_k}(x, y)$ is defined as:

$$F^{-1}_{t_i,\mu_k}(x, y) = \begin{pmatrix} \cos(\theta_i^k) & -\sin(\theta_i^k) \\ \sin(\theta_i^k) & \cos(\theta_i^k) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}.$$

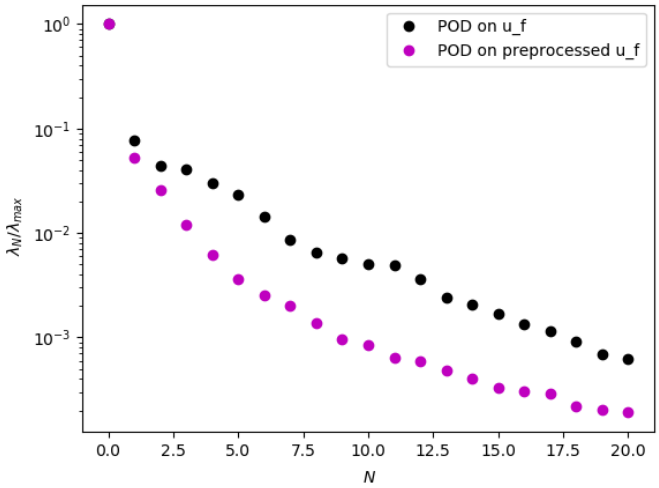
Also in this case the idea is to rotate back to a horizontal direction of propagation all the vortexes. We remark that the preprocessing procedure is carried out on the snapshots restricted to the domain Ω_{int}, exactly as we did for the problem in Section 3. Figure 4.17 shows the results that we obtain for three different values of the Reynolds number. As we can see, there is an improvement in the rate of decay of the eigenvalues, with results showing a difference of one order of magnitude for $Re = 145$ (right column) with just 20 modes.

4.6 Conclusions

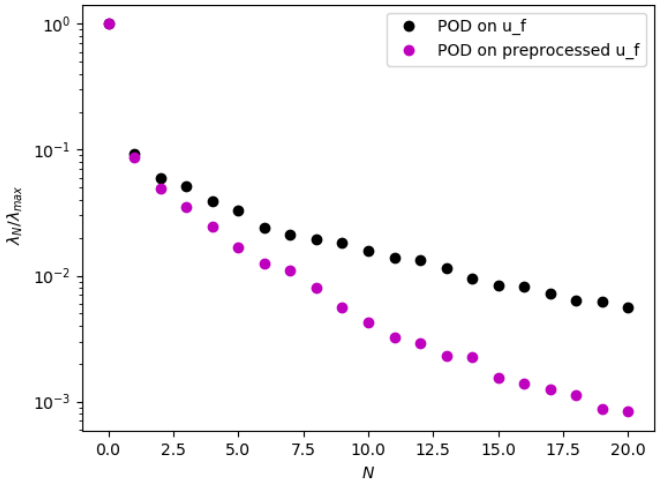
In this Chapter we used a preprocessing of the snapshots during the offline stage of the reduced basis method to improve the rate of decay of the Kolmogorov n-width of the solution manifold of the problem of interest. The general idea of the preprocessing procedure employed is to perform a geometrical transformation over the domain of interest, in order to get rid, or minimize, any kind of transportation phenomenon, rotation phenomenon, and so on. The method is known to perform very well for one dimensional problems [35], and in this Chapter we focused on two dimensional problems in a non-periodic setting. We can say that by adopting this preprocessing procedure of the set of snapshots, the saves from a computational point of view in terms of the dimension of the set of basis functions needed to reach a certain approximation accuracy are evident. The results that we obtained are promising: for the FSI test case, to reach a magnitude of 10^{-3} for the eigenvalues related to the POD on the pressure, we need 10 less modes with respect to the standard situation with no preprocessing, and we can say the same about other components (fluid displacement, fluid velocity, lagrange multipliers), thus lowering the dimension of the system to be solved in the online phase of the reduced method of at least 50. The plots in Figure 4.10 tell us also that the energy retained by the first modes after the preprocessing has been applied is by far higher with respect to the energy retained by the first modes computed from a POD on the original solution manifold. We saw that, in this particular test case, to learn the parameter $\gamma(t)$ that defines the family of deformation mappings $\mathcal{F}_\gamma$ it is sufficient to adopt a polynomial interpolation of order one. For future developements regarding the combination of the preprocessing procedure to FSI



(a) Results for Reynolds number equal to 47.



(b) Results for Reynolds number equal to 82.72.



(c) Results for Reynolds number equal to 145.

FIGURE 4.17: Rate of decay of the eigenvalues with and without preprocessing of the snapshots, for $\mu = 47$ (a), $\mu = 82.72$ (b) and for $\mu = 145$ (c).

problems, an interesting idea would be to combine this procedure with a segregated approach during the online stage: indeed, this would allow to decrease even more the dimension of the system by discarding the Lagrange multipliers. It would be interesting to compare the speedup obtained with the preprocessing technique combined with segregated approaches, compared to the performance of a standard monolithic approach combined with the preprocessing of the snapshots.
We obtained promising results also in the fluid dynamics test case. In the non parametric problem, results for the fluid velocity show that the eigenvalues after the preprocessing decay with almost two orders of magnitude faster than the standard case: to reach a magnitude of 10^{-3} in the standard case we need almost 30 modes whereas in the preprocessed case we need 10 modes. In the parametric case we analyzed the results for three different values of the physical parameter: all the three cases showed good results.

Chapter 5

A CutFEM Model Order Reduction for nonlinear fluid flows

In this Chapter we adopt an alternative point of view for what concerns the discretization of the problem of interest at the Finite Element level: indeed, in the previous two Chapters, the model order reduction techniques that we have proposed and tested have been based on a classical FE discretization. We are now interested in presenting and analyzing a reduction procedure that is instead based on an unfitted cut Finite Element discretization: these methods are particularly appealing because of their capability of handling large deformations of parametrized domains. We will see that the use of an embedded method, together with a reduced order method, allows for fast evaluation of parametrized problems, without the need of remeshing and without the need of a reference domain formulation, as it has been done in the previous Chapters in this thesis. The material presented in this Chapter has been submitted for journal publication, see [101].

5.1 Motivation

In the current Chapter, we are interested in geometrically parameterized steady and unsteady Navier–Stokes equation in a Eulerian framework. An approach based on unfitted mesh Finite Element Method shows its flexibility and capability in situations where domain changes occur, especially if the domain is subject to large deformations, and classical methods such as the Finite Element Method (FEM) fail. This work extends the approaches of [102, 105, 106] to nonlinear CutFEM discretization as well as to evolutionary in time fluid flow systems. We construct and investigate a unified and geometry independent reduced basis which overcomes many barriers and complications of the past. These complications may occur whenever geometrical morphings are taking place, and by using a geometry independent reduced basis, we are able to avoid remeshing, to avoid transformations to reference geometries and we can handle complex geometries. This combination of a fixed background mesh in a fixed extended background geometry with reduced order techniques appears beneficial and advantageous in many industrial and engineering applications, which could not be resolved efficiently in the past. In general, new computational tools have been invented and studied over the past years to solve numerically Navier-Stokes problems. The difficulties that arise from a mathematical point of view when we are interested in solving numerically these systems are many. FEM is a powerful computational tool to discretize the physical domain of interest and simulate the behavior of the solution of the system, and its efficiency has been proven over the years, in a wide range

of applications. Nonetheless, FEM capability to handle geometrically parametrized problems comes to a limit, this limit being given by extremely complex geometries, but also by situations where large deformations, fractures, contact points occur: in [20], the authors propose a modified ALE formulation that can be applied to problems where large deformation occurs. As an alternative to classical FEM, we can consider Finite Element (FE) approximations of the physical fields that are not fitted to the actual physical geometry. The FE approximations are then cut at the boundaries and interfaces: this gives rise to the Cut Finite Element Method (CutFEM). For a more precise idea and for more rigorous definitions of what "cutting" a physical field means, and for a detailed introduction to CutFEM, we refer to [32] and references therein.

The repeated solution of parametrized problems discretized by CutFEM is an expensive task (whose cost essentially depends on the size of the underlying background mesh), especially in complex geometries. It is precisely at this point that the Reduced Basis Method (RBM) comes into play. It is well known that the Reduced Basis Method is an extremely powerful tool to obtain a speedup in the simulation of the behavior of the solution of the system. The method relies on a set of already computed solutions (snapshots) for different parameter values: see, for example, [90, 156, 86]. Therein, these snapshots are FE approximations of the truth solution, thus the RBM relies on the FEM. Even though in general there are several methods that can be employed to project the full order system to a reduced system, see for example [156, 44, 98, 143, 45, 51], in the present work we will employ the Proper Orthogonal Decomposition (POD). We use a fixed background geometry and mesh: this approach leads to important advantages whenever a geometry deformation occurs [102], and it overcomes several related limitations in efficiency, compared with traditional FEM, see e.g. [10, 104, 105].
The aim of this Chapter is to implement a Reduced Basis Method for the stationary and evolutionary Navier–Stokes equation, that relies not on FEM but on CutFEM instead: already existing results on RBM applied to Navier Stokes problems with standard FEM can be found for example in [92, 144, 159], as well as results concerning RBM based on embedded FEM for different kind of problems, see for example [104, 105], as well as [106] for Navier-Stokes with the Shifted Boundary Method.

Our starting motivation is to apply this technique to CutFEM nonlinear fluid flow problems, but in order to do so, we test and study the procedure step by step: we first begin with the parametrized steady Navier–Stokes equation, then we will move to the time–dependent parametrized fluid problem. At the end of this Chapter we consider an unsteady problem, with a time–dependent geometrical configuration: this test case is of great interest, since it represents a first and preliminary study that potentially allows to think about a future developement of the work, in the direction of FSI applications; a work on this future line of research and developement is in preparation [100].

This Chapter is structured as follows: in Section 5.3.1 we introduce first a steady Navier–Stokes equation with incompressibility constraint in a fluid domain where the shape of some of the boundaries is described through a levelset function depending on a geometrical parameter; we move then to the unsteady Navier–Stokes equation, formulated over the same physical domain. Finally we solve the unsteady Navier–Stokes equation in a domain, where the levelset function is now time–dependent, i.e. it changes with time. Afterwards, in Section 5.4 we present the reduced order approach for each of the aforementioned problems. Numerical results are described

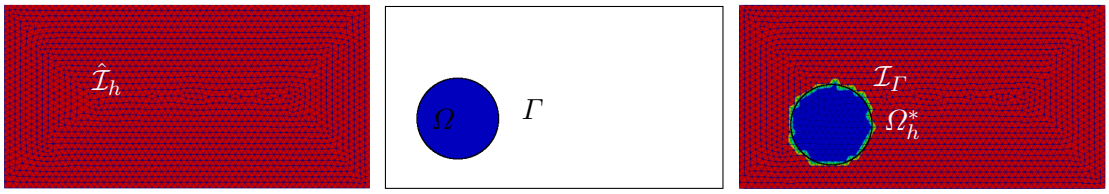


FIGURE 5.1: Representation of some important definitions in Domain Decomposition. Left: the background mesh $\hat{\mathcal{I}}_h$ and the background geometry (a rectangle in red). Center: the actual physical domain Ω in blue, immersed in the background geometry. Right: the fictious domain Ω_h^* (in blue and green), and the cut elements of $\mathcal{I}_\Gamma$ in green.

in Section 5.5 at both the high fidelity level and the reduced order level. Conclusions and perspectives are provided in Section 5.6.

5.2 CutFEM: terminology and definitions

In the following we recall some definitions and terminology that are commonly used for Domain Decomposition techniques: Domain Decomposition is a useful tool that allows to tailor the approximation approaches to the different computational domains at hand.

Let Ω be the physical domain over which our problem is formulated, and let $\hat{\mathcal{I}}_h$ be a background mesh of mesh size $h > 0$ covering Ω. The *active mesh* $\mathcal{I}_h$ is the mesh constituted by elements T of the background mesh, that are intersected by the physical domain:

$$\mathcal{I}_h := \{T \in \hat{\mathcal{I}}_h : \Omega \cap T \neq \emptyset\}.$$

Then, starting from the active mesh we can define a domain Ω_h^* as follows:

$$\Omega_h^* = \bigcup_{T \in \mathcal{I}_h} T.$$

We clarify that $\mathcal{I}_h$ is a

- *boundary fitted mesh* if $\overline{\Omega_h^*} = \overline{\Omega}$;
- *unfitted mesh* if $\overline{\Omega} \subsetneq \overline{\Omega_h^*}$. In this case Ω_h^* is the so called *fictitious domain*.

For unfitted meshes an important role is played by those elements of the active mesh that intersect the boundary Γ of the physical domain Ω:

$$\mathcal{I}_\Gamma := \{T \in \mathcal{I}_h : T \cap \Gamma \neq \emptyset\}.$$

Related to the set $\mathcal{I}_\Gamma$ we can also define the set of facets that belong to elements intersected by the boundary:

$$\mathcal{F}_\Gamma := \{F \in \mathcal{F} : F \text{ is a facet of an element } T \in \mathcal{I}_\Gamma\};$$

this set will become important in the next paragraph, when we introduce the Ghost Penalty (GP) stabilization. Figure 5.1 helps to understand and visualize some of the main concepts that are fundamental to understand the cut Finite Element Method.

Finally we define the approximation space

$$\mathcal{H}_h := \{v_h \in C_0(\Omega_h^*) \colon v_h|T \in \mathcal{P}^k(T), \quad \forall T \in \mathcal{I}_h\},$$

which is the space of continuous functions over Ω_h^*, which are polynomials of degree k on each element of the active mesh.

Nitsche's Method

Nitsche's Method [131] is a technique to weakly impose boundary conditions. While Neumann boundary conditions can be imposed naturally in the standard Galerkin formulation of the problem without introducing any source of instability, things are slightly more complicated for Dirichlet boundary conditions, or mixed boundary conditions such as Robin-type. The most straightforward way to weakly impose Dirichlet boundary conditions is to use a Lagrange multiplier, thus choosing an inf–sup stable pair of approximation spaces (Λ_h, V_h), one for the Lagrange multiplier and one for the primal unknown. In this framework, Nitsche's Method can be viewed as a particular Lagrange multiplier method, where the multiplier variable is replaced by its physical interpretation, or better, its numerical approximation: one of the reasons for the popularity of this technique is that it avoids to introduce any further unknown in the system. Originally the Nitsche's Method was employed by Nitsche to impose Dirichlet boundary conditions, but it has also been adapted for Robin and Neumman type boundary conditions [97]. For the simplest problem $-\Delta u = f \quad \text{in } \Omega$, with boundary conditions $u = g \quad \text{in } \Gamma$, the weak Nitsche's Method formulation reads as follows: find $u_h \in V_h$ such that for every $v_h \in V_h$ it holds:

$$\mathcal{L}(u_h, v_h) = F(v_h), \tag{5.1}$$

where:

$$\mathcal{L}(u_h, v_h) = (\nabla u_h, \nabla v_h)_\Omega - (\nabla u_h \cdot n, v_h)_\Gamma \mp (u_h, \nabla v_h \cdot n)_\Gamma + (\tfrac{\gamma}{h} u_h, v_h)_\Gamma, \tag{5.2}$$

$$F(v_h) = (f, v_h)_\Omega \mp (g, \nabla v_h \cdot n)_\Gamma + (\tfrac{\gamma}{h} g, v_h)_\Gamma. \tag{5.3}$$

In (5.2), the first two terms come from standard integration by part, as $v_h \neq 0$ in Γ; the third term enforces weakly the condition $u_h - g = 0$ in Γ, using as Lagrange multiplier the boundary flux (its numerical approximation) $\nabla v_h \cdot n$. According to the choice of the sign we have a *symmetric Nitsche's Method* $(-)$ or a *non-symmetric Nitsche's Method* $(+)$. The last term in (5.2) is the so called *penalty term*, and γ is a user defined parameter. If $\gamma = 0$, the method is penalty free. In the original work [131] the proposed method is a symmetric method, with penalty term γ to be choosen carefully to ensure coercivity, and hence inf-sup stability, of the formulation.

Ghost Penalty stabilization

It can be shown that, for unfitted meshes, the choice of the parameter γ is strictly related to the position of the interface Γ with respect to the element of the mesh. In order to overcome the stability issue of unfitted meshes, namely dependency of the stability and a priori estimates on the position of the interface, and the overall ill-conditioning of the global system matrix due to bad intersections, Burman et al. introduced a stabilization technique called Ghost Penalty, see [32, 34, 33, 31]. Ghost

Penalty consists in adding weakly consistent operators with the aim of having a better control on the solution in $\overline{\Omega_h^*} \setminus \overline{\Omega}$. Let us call *ghost nodes* the nodes of the unfitted mesh that are located in $\overline{\Omega_h^*} \setminus \overline{\Omega}$; the objective of Ghost Penalty is to control discrete polynomials on these ghost nodes. The idea is to do so by defining a smooth extension of discrete polyniomials in the boundary zone. We can distinguish two different types of extensions: inter-element face jump penalties and patch-wise L^2 projection–based penalties.

Inter-element face jump penalty This operator is particularly suited for k–order polynomials, and is defined as follows:

$$g(u_h, v_h) = \sum_{F \in \mathcal{F}_\Gamma} \sum_{0 \leq j \leq k} h_F^{2j+1} \langle [\![\partial_n^j u_h]\!], [\![\partial_n^j v_h]\!] \rangle_F, \tag{5.4}$$

where u_h and v_h are piecewise polynomials of order at most k. The operator defined in (5.4) introduces therefore a control on all the derivatives of the polynomial, up to order k, thus “smoothing" the behaviour of the polynomial near the boundary.

Patch-wise L^2 projection–based penalty For polynomials of higher order, a stabilization like the one defined in (5.4) would require a lot of computational effort. A good alternative is given by an L^2 projection–based penalty, which aims at penalizing fluctuations of discrete polynomials u_h between elements T, and L^2 projection onto patches. A patch can be defined as the union of two neighbouring elements T_F^+ and T_F^- sharing a common face F: $P_F = T_F^+ \cup T_F^-$. The L^2 projection–based penalty operator is defined as follows:

$$g(u_h, v_h) = \sum_{F \in \mathcal{F}_T} (u_h - \pi_F u_h, v_h)_{P_F}, \tag{5.5}$$

where $(\cdot, \cdot)_{P_F}$ is the L^2 inner product on P_F.
Both stabilization techniques have their drawbacks: for the operator defined in (5.4), the computational cost of evaluating jumps becomes relevant, when high order discrete polynomials are used; the operator also requires topological information on the mesh, such as identifying neighbouring elements, which can be challenging. On the other hand the operator defined in (5.5) requires the definition of functional spaces on patches, or at least the evaluation of polynomials on neighbouring elements. Nevertheless, even though the two operators require the introduction of additional data structures, the advantage in terms of stabilization is much more important than these drawbacks. Throughout this Chapter we will be using the inter–element face jump penalty.

5.3 Full order discretization by CutFEM

In this paragraph we introduce the problem formulation for the steady and unsteady Navier–Stokes equation.

In the time–dependent levelset geometry test case, we assume that the deformation in time of the levelset geometry is given: this can be seen as a first step towards coupled problems in fluid dynamics, which we are not considering herein and which will be part of a future work.

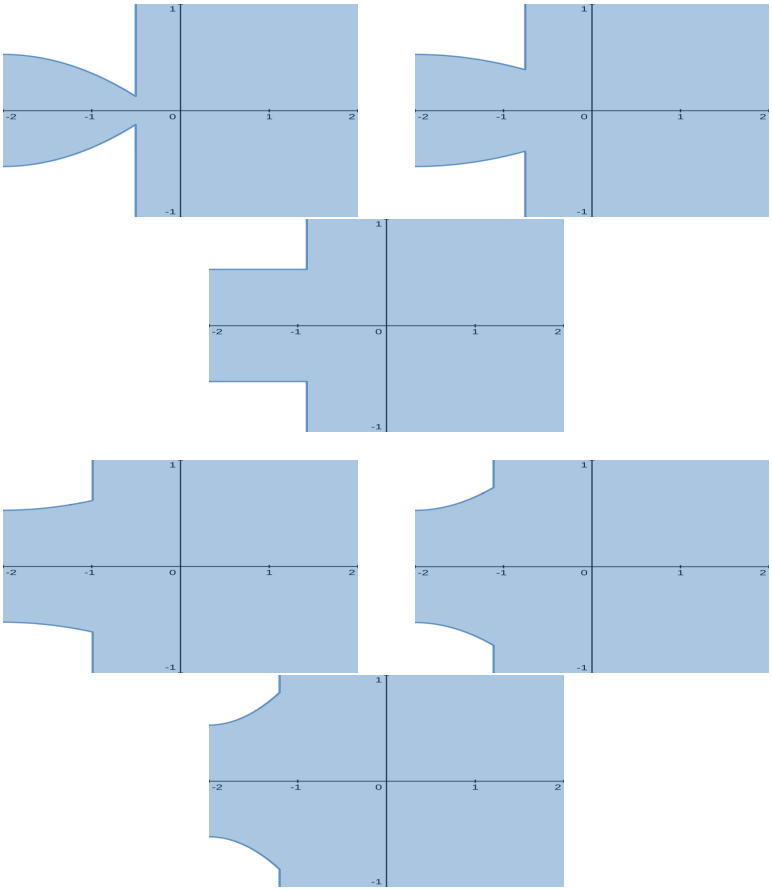


FIGURE 5.2: Six examples of solid walls, described by the levelset $\{\Phi_\mu = 0\}$. From left to the right, the levelset for $\mu = -0.1, -0.06, 0, 0.18, 0.37, 0.50$.

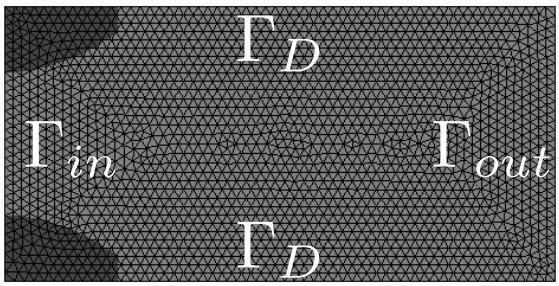


FIGURE 5.3: Mesh and geometry of the problem: the fluid domain (light grey) has an inlet boundary Γ_{in} on the left and an outlet boundary Γ_{out} on the right. The rest of the boundaries (Γ_D) are Dirichlet type boundaries. The solid domain is depicted in dark grey. Its shape is described through a levelset function and varies according to a parameter μ. This instance corresponds to $\mu = 0.37$.

5.3.1 Steady Navier–Stokes

Strong formulation

In the following, the incompressible Navier–Stokes equation is formulated within an Eulerian formalism. Let $\mathcal{R}$ be a background rectangular domain in $\mathbb{R}^2$, and let $\mathcal{D}(\mu) \subset \mathcal{R}$ be a bounded subset of $\mathcal{R}$, whose boundary is described through a levelset function $\{\Phi_\mu = 0\}$, where Φ_μ is an implicit function depending on a geometrical parameter μ. The physical domain over which our problem is formulated is $\Omega(\mu) := \mathcal{R} \setminus \mathcal{D}(\mu)$. We denote by $\mathcal{P}$ the parameter space to which μ belongs. Under these assumptions, our problem of interest reads: for every $\mu \in \mathcal{P}$, find $\boldsymbol{u}(\mu)\colon \Omega(\mu) \mapsto \mathbb{R}^2$ and $p(\mu)\colon \Omega(\mu) \mapsto \mathbb{R}$ such that:

$$\begin{cases} -\nu \Delta \boldsymbol{u}(\mu) + \nabla p(\mu) + (\boldsymbol{u}(\mu) \cdot \nabla)\boldsymbol{u}(\mu) = \boldsymbol{f}(\mu) \quad \text{in } \Omega(\mu), \\ \text{div}\boldsymbol{u}(\mu) = 0 \quad \text{in } \Omega(\mu), \\ \boldsymbol{u}(\mu) = \boldsymbol{u}_{in}(\mu) \quad \text{on } \Gamma_{in}(\mu). \end{cases} \tag{5.6}$$

Geometrical parametrization

Figure 5.3 shows the background domain $\mathcal{R}$, with the background mesh $\hat{\mathcal{I}}_h$, the physical domain $\Omega(\mu)$ (light grey) and the domain $\mathcal{D}(\mu)$ (dark grey). For the problem considered in this section the expression of the levelset function is the following:

$$\begin{aligned} \Phi_\mu(x,y) = & -\Big(|A(x) + B(x) - 1| + |A(x) - B(x) - 2| + D(x)\Big) \\ & \cdot \Big(|A(x) + C(x) - 1| + |A(x) - C(x) - 2| + D(x)\Big), \end{aligned} \tag{5.7}$$

where $A(x) = \sqrt{k_1}\,|x - k_3|$, $B(x) = \sqrt{k_2}\,|y - k_4|$, $C(x) = \sqrt{k_2}\,|y - k_5|$ and $D(x) = e^{-\mu}(k_1(x - k_3)^2)\mu - 4$. The values of the constants k_1, k_2, k_3, k_4, k_5 are reported in Table 5.1. To have a better idea of how the shape of the walls changes by varying the parameter μ, the reader is referred to Figure 5.2.

TABLE 5.1: Values for the constants in the levelset equation in the Navier–Stokes problem.

Constant	Value
k_1	10
k_2	10
k_3	-2
k_4	-1
k_5	1

Discrete weak formulation and algebraic formulation

As we can see from Figure 5.3, the background mesh $\hat{\mathcal{I}}_h$ is a rectangular mesh made by triangular elements. By choosing an unfitted method, once we have defined a background mesh, there is no need to remesh every time the parameter μ changes (and hence everytime the shape of the levelset in dark grey in Figure 5.3 changes). Since we are in an unfitted framework, we need some stabilization terms in the discretized weak formulation, in order to control polynomials on the cut elements.
In order to state the discretized weak formulation of the problem, let us introduce the following discrete approximation spaces:

$$V_{h,k}(\mu) := \{\boldsymbol{v}_h \in C_0(\Omega_h^*(\mu))^2 \colon \boldsymbol{v}_h|_T \in (\mathcal{P}^k(T))^2, \quad \forall T \in \mathcal{I}_h(\mu)\},$$
$$Q_{h,1}(\mu) := \{q_h \in C_0(\Omega_h^*(\mu)) \colon q_h|_T \in \mathcal{P}^1(T), \quad \forall T \in \mathcal{I}_h(\mu)\},$$

where $\mathcal{I}_h(\mu)$ and $\Omega_h^*(\mu)$ are the active mesh and the fictitious domain respectively, as defined in Section 5.2. We remark that these spaces depend on μ, since the levelset geometry changes according to the parameter μ. The discretized weak formulation of the problem, with Nitsche's Method and Ghost Penalty terms then reads: find $(\boldsymbol{u}_h(\mu), p_h(\mu)) \in V_{h,1}(\mu) \times Q_{h,1}(\mu)$ such that for all test functions $(\boldsymbol{v}_h(\mu), q_h(\mu)) \in V_{h,1}(\mu) \times Q_{h,1}(\mu)$:

$$\mathcal{A}(\boldsymbol{u_h}(\mu), p_h(\mu), \boldsymbol{v}_h(\mu), q_h(\mu)) = \mathcal{L}(\boldsymbol{v}_h(\mu)), \tag{5.8}$$

where:

$$\begin{aligned}
\mathcal{A}(\boldsymbol{u_h}(\mu), p_h(\mu), \boldsymbol{v_h}(\mu), q_h(\mu)) &= a(\boldsymbol{u_h}(\mu), \boldsymbol{v_h}(\mu); \mu) + d(\boldsymbol{u_h}(\mu), \boldsymbol{u_h}(\mu), \boldsymbol{v_h}(\mu); \mu) \\
&\quad - b(p_h(\mu), \boldsymbol{v_h}(\mu); \mu) + b(q_h(\mu), \boldsymbol{u_h}(\mu); \mu) \\
&\quad - \hat{b}(q_h(\mu), \boldsymbol{u_h}(\mu); \mu) + j_{\mathrm{NIT}}(\boldsymbol{u_h}(\mu), \boldsymbol{v_h}(\mu); \mu) \\
&\quad + j^u_{\mathrm{IP}}(\boldsymbol{u_h}(\mu), \boldsymbol{v_h}(\mu); \mu) - j^p_{\mathrm{IP}}(p_h(\mu), q_h(\mu); \mu), \\
\mathcal{L}(\mathbf{v}_h) &= \int_{\Omega_h^*(\mu)} f(\mu)\mathbf{v}_h(\mu).
\end{aligned}$$

TABLE 5.2: Constants values for the weak formulation of the steady Navier–Stokes problem

Constant	Value	Constant	Value
ν	0.05	h	0.07
γ_1	10	γ_2	10
α	0.1	g_u	0.001
γ_p	0.1	γ_g	0.1
λ_s	10	γ_s	$(0.1, 0.01)$

More precisely we have the following terms:

$$
\begin{aligned}
a(\boldsymbol{u}_h(\mu), \boldsymbol{v}_h(\mu); \mu) &= \int_{\Omega_h^*(\mu)} \nu \nabla \boldsymbol{u}_h(\mu) \colon \nabla \boldsymbol{v}_h(\mu) - \sum_{T \in \mathcal{I}_\Gamma(\mu)} \int_T \nu \partial_n \boldsymbol{u}_h(\mu) \cdot \boldsymbol{v}_h(\mu) \\
&\quad - \sum_{T \in \mathcal{I}_\Gamma(\mu)} \int_T \nu \partial_n \boldsymbol{v}_h(\mu) \cdot \boldsymbol{u}_h(\mu), \\
d(\boldsymbol{u_h}(\mu), \boldsymbol{w_h}(\mu), \boldsymbol{v_h}(\mu); \mu) &= \int_{\Omega_h^*(\mu)} ((\boldsymbol{u_h}(\mu) \cdot \nabla) \boldsymbol{w_h}(\mu)) \cdot \boldsymbol{v_h}(\mu), \\
b(p_h(\mu), \boldsymbol{v_h}(\mu); \mu) &= \int_{\Omega_h^*(\mu)} p_h(\mu) \mathrm{div} \boldsymbol{v_h}(\mu), \\
\hat{b}(q_h(\mu), \boldsymbol{u_h}(\mu); \mu) &= \sum_{T \in \mathcal{I}_\Gamma(\mu)} \int_T q_h(\mu) \boldsymbol{n} \cdot \boldsymbol{u_h}(\mu), \\
j_{\mathrm{NIT}}(\boldsymbol{u_h}(\mu), \boldsymbol{v_h}(\mu); \mu) &= \nu \frac{\gamma_1}{h} \sum_{T \in \mathcal{I}_\Gamma(\mu)} \int_T \boldsymbol{u_h}(\mu) \boldsymbol{v_h}(\mu) + \\
&\quad + \frac{\gamma_2}{h} \sum_{T \in \mathcal{I}_\Gamma(\mu)} \int_T (\boldsymbol{u_h}(\mu) \cdot \boldsymbol{n})(\boldsymbol{v_h}(\mu) \cdot \boldsymbol{n}), \\
j^u_{\mathrm{IP/GP}}(\boldsymbol{u_h}(\mu), \boldsymbol{v_h}(\mu); \mu) &= \sum_{F \in \mathcal{F}_T(\mu)} \alpha \nu h^2 ||\boldsymbol{u_h}(\mu)||_\infty \int_F [\![\mathrm{div}(\boldsymbol{u_h})(\mu)]\!] [\![\mathrm{div}(\boldsymbol{v_h}(\mu))]\!] \\
&\quad - \sum_{F \in \mathcal{F}_T(\mu)} g_u \nu h^2 \int_F [\![\partial_n (\boldsymbol{u_h}(\mu))]\!] \cdot [\![\partial_n (\boldsymbol{v_h}(\mu))]\!] \\
&\quad + \sum_{F \in \mathcal{F}_T(\mu)} \gamma_u \nu h \int_F [\![\partial_n (\boldsymbol{u_h}(\mu))]\!] [\![\partial_n (\boldsymbol{v_h}(\mu))]\!], \\
j^p_{\mathrm{IP/GP}}(p_h(\mu), q_h(\mu); \mu) &= \sum_{F \in \mathcal{F}_T(\mu)} \frac{\gamma_p h^3}{\nu} \frac{1}{\max(||\boldsymbol{u}(\mu)||_\infty, 1)} \int_F [\![\partial_n p_h(\mu)]\!] [\![\partial_n q_h(\mu)]\!].
\end{aligned}
$$

The terms $a(\boldsymbol{u_h}(\mu), \boldsymbol{v_h}(\mu); \mu)$, $b(p_h(\mu), \boldsymbol{v_h}(\mu); \mu)$, $d(\boldsymbol{u_h}(\mu), \boldsymbol{u_h}(\mu), \boldsymbol{v_h}(\mu); \mu)$, and $\hat{b}(q_h(\mu), \boldsymbol{u_h}(\mu); \mu)$ come from the weak formulation of the incompressible Navier–Stokes equation, with Nitsche terms to impose the Dirichlet boundary condition.
The expressions $j_{\mathrm{NIT}}(\boldsymbol{u_h}(\mu), \boldsymbol{v_h}(\mu); \mu)$, $j^u_{\mathrm{IP/GP}}(\boldsymbol{u_h}(\mu), \boldsymbol{v_h}(\mu); \mu)$ and $j^p_{\mathrm{IP/GP}}(p_h(\mu), q_h(\mu); \mu)$ contain stabilization terms, that are necessary whenever the discretization spaces do not satisfy the inf–sup condition (e.g. with equal order spaces), Interior Penalty terms and Ghost Penalty terms. The values of the constants appearing in the previous equations are reported in Table 5.2.
As we can see from the previous definitions, $\boldsymbol{u}_h(\mu)$ belongs to a space that is μ–dependent, namely $V_{h,1}(\mu)$, and the same goes for the pressure $p_h(\mu) \in Q_{h,1}(\mu)$.

In order to define solutions on the whole background mesh, we employ a *natural smooth extension* of both velocity and pressure. In this way we obtain snapshots $(\hat{\boldsymbol{u}}_h(\mu), \hat{p}_h(\mu))$ that are defined on the common background mesh $\hat{\mathcal{I}}_h$; the reader interested in a detailed discussion on the natural smooth extension and on alternative techniques to extend the snapshots to the background mesh is referred to [102]. Such extension defines a pair of velocity–pressure snapshots $(\hat{\boldsymbol{u}}_h(\mu), \hat{p}_h(\mu))$ belonging to μ*–independent* discrete spaces:

$$\begin{aligned}\hat{V}_{h,k} &:= \{\hat{\boldsymbol{v}}_h \in C_0(\mathcal{R})^2 \colon \hat{\boldsymbol{v}}_h|_T \in (\mathcal{P}^k(T))^2, \quad \forall T \in \hat{\mathcal{I}}_h\},\\ \hat{Q}_{h,1} &:= \{\hat{q}_h \in C_0(\mathcal{R}) \colon \hat{q}_h|_T \in \mathcal{P}^1(T), \quad \forall T \in \hat{\mathcal{I}}_h\}.\end{aligned}$$

In order to state the algebraic formulation equivalent to (5.8), let us introduce the following bijection between $\hat{V}_{h,1}$ and $\mathbb{R}^{N_u^h}$ (respectively $\hat{Q}_{h,1}$ and $\mathbb{R}^{N_p^h}$), where N_u^h and N_p^h are the dimensions of the discrete spaces $\hat{V}_{h,1}$ and $\hat{Q}_{h,1}$:

$$\begin{cases}\underline{\hat{\mathbf{v}}}_h = (\hat{v}_h^1, \dots, \hat{v}_h^{N_u^h})^T \in \mathbb{R}^{N_u^h} \iff \hat{\boldsymbol{v}_h} = \sum_{i=1}^{N_u^h} \hat{v}_h^i \boldsymbol{\varphi}^i \in \hat{V}_{h,1},\\ \underline{\hat{q}}_h = (\hat{q}_h^1, \dots, \hat{q}_h^{N_p^h})^T \in \mathbb{R}^{N_p^h} \iff \hat{q}_h = \sum_{i=1}^{N_p^h} \hat{q}_h^i \zeta^i \in \hat{Q}_{h,1},\end{cases} \tag{5.9}$$

where $\boldsymbol{\varphi}^i$ and ζ^i are the *parameter–independent* basis functions of the FE spaces $\hat{V}_{h,1}$ and $\hat{Q}_{h,1}$ respectively. Thanks to this bijection we can define the following matrices:

$$\begin{aligned}\boldsymbol{A}(\mu)_{ij} &:= a(\boldsymbol{\varphi}^i, \boldsymbol{\varphi}^j, \mu) + j_{\mathrm{NIT}}(\boldsymbol{\varphi}^i, \boldsymbol{\varphi}^j, \mu) + j^u_{\mathrm{IP/GP}}(\boldsymbol{\varphi}^i, \boldsymbol{\varphi}^j, \mu),\\ \boldsymbol{N}(\underline{\hat{\boldsymbol{u}}}_h(\mu); \mu)_{ij} &:= \sum_{k=1}^{N_u^h} \hat{u}_h^k(\mu) d(\boldsymbol{\varphi}^k, \boldsymbol{\varphi}^i, \boldsymbol{\varphi}^j; \mu),\\ \boldsymbol{B}(\mu)_{ij} &:= -b(\zeta^i, \boldsymbol{\varphi}^j, \mu),\\ \hat{\boldsymbol{B}}(\mu)_{ij} &:= \hat{b}(\zeta^i, \boldsymbol{\varphi}^j, \mu).\\ \boldsymbol{C}(\mu)_{ij} &:= j^p_{\mathrm{IP/GP}}(\zeta^i, \zeta^j; \mu)\end{aligned}$$

Thanks to the introduced notation, we can conclude that equation (5.8) is equivalent to the following algebraic system:

$$R(\hat{U}_h(\mu), \mu) := \begin{bmatrix}\boldsymbol{A}(\mu) + \boldsymbol{N}(\underline{\hat{\mathbf{u}}}_h(\mu); \mu) & \boldsymbol{B}^{\boldsymbol{T}}(\mu)\\ \boldsymbol{B}(\mu) + \hat{\boldsymbol{B}}(\mu) & C(\mu)\end{bmatrix} \begin{bmatrix}\underline{\hat{\mathbf{u}}}_h(\mu)\\ \underline{\hat{p}}_h(\mu)\end{bmatrix} - \begin{bmatrix}\boldsymbol{F}_1(\mu)\\ \boldsymbol{F}_2(\mu)\end{bmatrix} = \begin{bmatrix}0\\0\end{bmatrix},$$

where $\hat{U}_h(\mu) = (\underline{\hat{\mathbf{u}}}_h(\mu), \underline{\hat{p}}_h(\mu))$, $(\boldsymbol{F}_1(\mu))_i := \int_{\Omega_h^*(\mu)} \boldsymbol{\varphi}^i \cdot \boldsymbol{f}\, dx$, and $\boldsymbol{F}_2(\mu) = \mathbf{0}$.

We point out that the aforementioned formulation, defined on the whole background mesh, is actually required only by the ROM procedure. The CutFEM formulation is defined only on the active mesh $\mathcal{I}_h$ and the extended geometry Ω_h^*.

5.3.2 Unsteady Navier–Stokes

Strong formulation

We now extend the previous treatment by introducing the time evolution term $\partial_t \boldsymbol{u}(\mu)$ in system (5.6). Given a time interval of interest $[0, T]$, the strong formulation

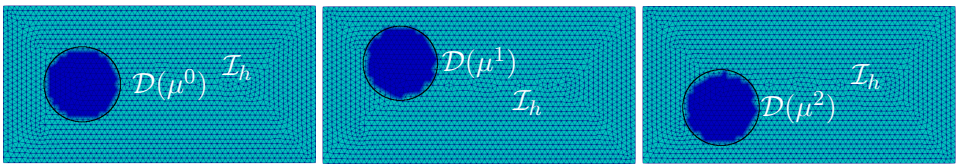


FIGURE 5.4: Time-dependent levelset: The physical domain at different times t_0, t_1, t_2. The black circle delimits the cylinder $\mathcal{D}(\mu)$ immersed in the fluid: the cylinder is moving up and down, so that the levelset function defining $\mathcal{D}(\mu)$ is time dependent. $\mathcal{I}_h$ corresponds to the active mesh.

of the problem reads as follows: for every $\mu \in \mathcal{P}$, and for every $t \in [0,T]$, find $\boldsymbol{u}(t;\mu)\colon \Omega(\mu) \mapsto \mathbb{R}^2$ and $p(t;\mu)\colon \Omega(\mu) \mapsto \mathbb{R}$ such that:

$$\begin{cases} \partial_t \boldsymbol{u}(\mu) - \nu \Delta \boldsymbol{u}(\mu) + \nabla p(\mu) + (\boldsymbol{u}(\mu)\cdot\nabla)\boldsymbol{u}(\mu) = f(\mu) \quad \text{in } \Omega(\mu)\times[0,T], \\ \text{div}\boldsymbol{u}(\mu) = 0 \quad \text{in } \Omega(\mu)\times[0,T], \\ \boldsymbol{u}(\mu) = \boldsymbol{u}_{in} \quad \text{on } \Gamma_{in}(\mu)\times[0,T], \\ \boldsymbol{u}(\boldsymbol{x},0;\mu) = \boldsymbol{u}^0(\boldsymbol{x},\mu) \quad \text{in } \Omega(\mu), \end{cases} \tag{5.10}$$

with geometrical parameterization identical to that in the previous subsection. We remark that here the geometrical parametrization does not evolve in time, i.e. $\frac{d\mu}{dt} = 0$. We will consider the case of $\frac{d\mu}{dt} \neq 0$ in the next Section 5.3.3.

Space discretization, time–stepping scheme and algebraic formulation

We discretize in time by an implicit–explicit Euler approach, see e.g. [152]. We discretize the time interval $[0,T]$ with the following partition:

$$0 = t^0 < \ldots < t^{N_t} = T,$$

where every interval $(t^n, t^{n+1}]$ has measure $\tau_{n+1} = t^{n+1} - t^n$, $n = 0, ..., N_t - 1$. The discrete version of the initial condition $\boldsymbol{u}^0(x;\mu)$ is denoted by $\boldsymbol{u}_h^0(x;\mu)$; we denote $\boldsymbol{u}_h^n$ the discrete fluid velocity at time step t^n, and similar notation is used for the pressure.
We will treat the nonlinear term explicitly, and the linear terms implicitly. After having applied a time stepping scheme, the space discretized weak formulation of the problem reads as follows: for every $n = 0, \ldots, N_t - 1$, we seek a discrete velocity $\boldsymbol{u}_h^{n+1}(\mu) \in V_{h,2}(\mu)$ and discrete pressure $p_h^{n+1}(\mu) \in Q_{h,1}(\mu)$, such that for every $(\boldsymbol{v}_h(\mu), q_h(\mu)) \in V_{h,2}(\mu) \times Q_{h,1}(\mu)$, it holds:

$$m(\boldsymbol{u}_h^{n+1}(\mu) - \boldsymbol{u}_h^n(\mu), v_h) + \tau_{n+1}\mathcal{A}(\boldsymbol{u}_h^{n+1}(\mu), p_h^{n+1}(\mu), \boldsymbol{v}_h(\mu), q_h(\mu)) = \tau_{n+1}\mathcal{L}(\boldsymbol{v}_h(\mu)),$$

where

$$m(\boldsymbol{w}_h, \boldsymbol{v}_h) := \int_{\Omega_h^*(\mu)} \boldsymbol{w}_h \cdot \boldsymbol{v}_h \, dx,$$

and where $\mathcal{A}(\boldsymbol{u_h}^{n+1}(\mu), p_h^{n+1}(\mu), \boldsymbol{v_h}(\mu), q_h(\mu))$ and $\mathcal{L}(\boldsymbol{v}_h(\mu))$ are defined as in Section 5.3.1. We employ again a natural smooth extension, as stated in the previous subsection. Keeping the notation previously introduced, and using the bijection (5.9), we introduce the *mass matrix* $\boldsymbol{M}_{ij} := m(\boldsymbol{\varphi}^i, \boldsymbol{\varphi}^j)$. Let now $\hat{U}_h^{n+1}(\mu)$ be defined as $\hat{U}_h^{n+1}(\mu) := (\underline{\hat{\mathbf{u}}}_h^{n+1}, \underline{\hat{p}}_h^{n+1})$; using the notation introduced for the steady problem, the

resulting fully-discrete time dependent cut Finite Element Navier–Stokes algebraic equation is

$$\begin{bmatrix} \boldsymbol{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \hat{U}_h^{n+1}(\mu) + \tau_{n+1} R(\hat{U}_h^{n+1}(\mu); \mu) = \begin{bmatrix} \boldsymbol{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \hat{U}_h^{n}(\mu).$$

5.3.3 Unsteady Navier–Stokes with time dependent geometry

We extend here the previous unsteady formulation to the case where the parameter μ is *time dependent*, i.e $\partial_t \mu \neq 0$. For this test case we assume that $\mathcal{D}(\mu)$ represents a cylinder immersed in the fluid domain, and therefore we denote herein $\Gamma_{cyl}(\mu) = \partial\mathcal{D}(\mu)$.

Strong formulation.

The problem reads as follows: for every $t \in [0, T]$ and for every $\mu(t) \in \mathcal{P}$, find $\boldsymbol{u}(t; \mu(t))\colon \Omega(\mu(t)) \mapsto \mathbb{R}^2$, $p(t; \mu(t))\colon \Omega(\mu(t)) \mapsto \mathbb{R}$ such that:

$$\begin{cases} \partial_t \boldsymbol{u}(\mu(t)) - \nu \Delta \boldsymbol{u}(\mu(t)) + \nabla p(\mu(t)) + \\ \qquad\qquad + (\boldsymbol{u}(\mu(t)) \cdot \nabla)\boldsymbol{u}(\mu(t)) = f(\mu(t)) \quad \text{in } \Omega(\mu(t)) \times [0, T], \\ \text{div}\boldsymbol{u}(\mu(t)) = 0 \quad \text{in } \Omega(\mu(t)) \times [0, T], \\ \boldsymbol{u}(\mu(t)) = \boldsymbol{u}_{in} \quad \text{on } \Gamma_{in}(\mu(t)) \times [0, T], \\ \boldsymbol{u}(\mu(t)) = \partial_t \mu \quad \text{on } \Gamma_{cyl}(\mu(t)) \times [0, T], \\ \boldsymbol{u}(\boldsymbol{x}, 0; \mu(t)) = \boldsymbol{u}^0(\boldsymbol{x}, \mu(t)) \quad \text{in } \Omega(\mu(t)), \end{cases} \tag{5.11}$$

where $\partial_t \mu$ denotes the velocity with which the cylinder $\mathcal{D}(\mu(t))$ moves in the domain.

Geometrical parametrization

The obstacle immersed in the fluid domain in our problem is a parametrized circle, defined through the *time dependent* levelset function:

$$\phi(x, y, \mu_1(t), \mu_2(t)) = (x - \mu_1(t))^2 + (y - \mu_2(t))^2 - R^2,$$

where $\mu(t) = (\mu_1(t), \mu_2(t))$ denotes the position of the center of the cylinder in the domain, and R is the radius of the circle.
As we can see from the strong formulation of the problem, the motion of the cylinder is assumed to be known, i.e. it is not an unknown of the system, as it would be instead for a fully coupled Fluid–Structure Interaction problem. For our simulations, we assume the motion of the cylinder to be periodic, i.e

$$\mu(t) = \mu(0) + A\sin(8\pi\omega(t))\underline{\boldsymbol{j}}, \tag{5.12}$$

where A denotes the amplitude of the oscillation of the cylinder, $\omega(t)$ is a function of time and $\underline{\boldsymbol{j}}$ is a versor in the vertical direction: the cylinder moves only vertically in our simulations. With this assumption, we can explicitly write the velocity of the

cylinder $\partial_t\mu$, in fact:

$$\partial_t\mu(t) = A\frac{d\omega(t)}{dt}8\pi\cos(\omega(t)8\pi)\underline{\boldsymbol{j}}.$$

Figure 5.4 shows the physical domain $\Omega(\mu)$ at different times t_0, t_1, t_2, for the time dependent levelset function $\phi(x, y, t)$ defining $\mathcal{D}(\mu)$.

Weak formulation and time discretization

We now want to state the weak formulation of the original problem after discretization in space and after having applied a time stepping scheme. As far as the time discretization concerns, we use the time stepping scheme adopted in the previous subsection for the unsteady Navier–Stokes problem: we discretize the time interval $[0, T]$ in sub-intervals $(t^n, t^{n+1}]$ of measure $\tau_{n+1} = t^{n+1} - t^n$, for $n = 0, \dots, N_t - 1$. Let us denote with $\mu^n = \mu(t^n)$ the value of the geometrical parameter at time t^n.
For the space discretization, let us introduce the following discrete spaces at time-step t^{n+1}:

$$V_{h,2}(\mu^{n+1}) := \{\boldsymbol{v}_h \in (C_0(\Omega_h^*(\mu^{n+1})))^2 \colon \boldsymbol{v}_h|_T \in (\mathcal{P}^2(T))^2, \quad \forall T \in \mathcal{I}_h(\mu^{n+1})\},$$
$$Q_{h,1}(\mu^{n+1}) := \{q_h \in C_0(\Omega_h^*(\mu^{n+1})) \colon q_h|_T \in \mathcal{P}^1(T), \quad \forall T \in \mathcal{I}_h(\mu^{n+1})\}.$$

As we can see from the previous definitions, the discrete spaces $V_{h,2}(\mu^{n+1})$ and $Q_{h,1}(\mu^{n+1})$ depend now not only on the geometrical parameter μ, but also on time t; in fact, even though the background mesh $\hat{\mathcal{I}}_h$ remains fixed, the active mesh $\mathcal{I}_h$, the fictitious domain Ω_h^*, the cut elements in $\mathcal{I}_\Gamma$ and other relevant entities, change in time according to the change of the levelset function. After applying a time stepping scheme, the weak formulation of the original problem reads: at time–step t^{n+1}, find $(\boldsymbol{u}_h^{n+1}(\mu^{n+1}), p_h^{n+1}(\mu^{n+1})) \in V_{h,2}(\mu^{n+1}) \times Q_{h,1}(\mu^{n+1})$ such that, for all $(v_h, q_h) \in V_{h,2}(\mu^{n+1}) \times Q_{h,1}(\mu^{n+1})$:

$$\begin{aligned} &m(\boldsymbol{u}_h^{n+1}(\mu^{n+1}) - \boldsymbol{u}_h^n(\mu^n), v_h(\mu^{n+1})) + \\ &+ \tau_{n+1}\mathcal{A}(\boldsymbol{u}_h{}^{n+1}(\mu^{n+1}), p_h^{n+1}(\mu^{n+1}), \boldsymbol{v}_h(\mu^{n+1}), q_h(\mu^{n+1})) = \tau_{n+1}\mathcal{L}(\boldsymbol{v}_h(\mu^{n+1})), \end{aligned} \tag{5.13}$$

where we have kept the same notation as in the previous subsection. The problem now is that all the integrals appearing in equation (5.13) are evaluated over the fictitious domain at time-step t^{n+1}, namely $\Omega_h^*(\mu^{n+1})$: in general $\Omega_h^*(\mu^{n+1}) \neq \Omega_h^*(\mu^n)$, and thus $V_h^{n+1}(\mu^{n+1}) \neq V_h^n(\mu^n)$, so in general $\boldsymbol{u}_h^n(\mu^n) \notin V_h(\mu^{n+1})$. To overcome this problem we employ again a *natural smooth extension* of the snapshots $(\boldsymbol{u}_h^{n+1}(\mu^{n+1}), p_h^{n+1}(\mu^{n+1}))$, as already introduced in Section 5.3.1. We therefore recall the definition of the global discrete spaces defined on the background domain $\mathcal{R}$:

$$\hat{V}_{h,2} := \{\hat{\boldsymbol{v}}_h \in C_0(\mathcal{R})^2 \colon \hat{\boldsymbol{v}}_h|_T \in (\mathcal{P}^2(T))^2, \quad \forall T \in \hat{\mathcal{I}}_h\},$$
$$\hat{Q}_{h,1} := \{\hat{q}_h \in C_0(\mathcal{R}) \colon \hat{q}_h|_T \in \mathcal{P}^1(T), \quad \forall T \in \hat{\mathcal{I}}_h\},$$

and, keeping the same notation previously introduced, we can recover the algebraic formulation of the problem, which reads: for every $n = 0, \dots, N_t - 1$, find $\hat{U}_h^{n+1}(\mu^{n+1}) := (\underline{\hat{\mathbf{u}}}_h^{n+1}(\mu^{n+1}), \underline{\hat{p}}_h(\mu^{n+1}))$ such that:

$$\begin{bmatrix} \boldsymbol{M} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \hat{U}_h^{n+1}(\mu^{n+1}) + \tau_{n+1}R(\hat{U}_h^{n+1}(\mu^{n+1}); \mu) = \begin{bmatrix} \boldsymbol{M} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} \hat{U}_h^n(\mu^n).$$

5.4 Proper Orthogonal Decomposition-Galerkin Model Reduction

In the present Chapter, the Proper Orthogonal Decomposition (POD) method is applied to parameter-dependent matrices which have been derived from solution instances. The POD method consists of two phases: one *offline* and one *online*. During the offline phase, we compute the solution of the problem of interest, for different values of the parameters. These parameters are collected from a training set $\mathcal{P}_{train}$, and the corresponding solutions are stored into a matrix, the so-called snapshots matrix. This matrix is then processed in order to extract the reduced basis. Afterwards, in the online phase, we employ these basis functions in a way that reduces the dimension of the original problem, and in a way that is computationally efficient for (in our case) geometrically parametrized systems.

We remind that for POD-Galerkin ROMs for incompressible Navier–Stokes equations, instabilities in the approximation of the pressure may occur. We refer to [38, 70, 155] for a more detailed analysis of the problem, while for such instabilities on transient problems we refer to [94, 2, 23, 158, 60]. For SUPG and PSPG kind of stabilization we refer to [12, 157, 161, 160].

5.4.1 Steady case

Next we denote by $\mu^{(j)}$ each parameter in a finite dimensional training set $\mathcal{P}_{train} = \{\mu^{(1)}, \dots, \mu^{(M)}\}$ for a large number M. We recall that the number of degrees of freedom considering the full order problem are denoted by N_u^h and N_p^h for the velocity and the pressure respectively. The collected snapshots matrices $\boldsymbol{\mathcal{S}_u}$ and $\boldsymbol{\mathcal{S}_p}$, are then defined as follows:

$$\boldsymbol{\mathcal{S}_u} = [\underline{\hat{\mathbf{u}}}_h(\mu^{(1)}), \dots, \underline{\hat{\mathbf{u}}}_h(\mu^{(M)})] \in \mathbb{R}^{N_u^h \times M}, \quad \boldsymbol{\mathcal{S}_p} = [\underline{\hat{p}}_h(\mu^{(1)}), \dots, \underline{\hat{p}}_h(\mu^{(M)})] \in \mathbb{R}^{N_p^h \times M}, \tag{5.14}$$

where $\underline{\hat{\mathbf{u}}}_h$ and $\underline{\hat{p}}_h$ are vectors defined by the bijection (5.9).
In order to make the pressure approximation stable at the reduced order level we also introduce a velocity supremizer variable $\boldsymbol{s_h}$: see [12, 155, 157] for a more detailed introduction to the supremizer enrichment for Navier–Stokes equation. We start with a Poisson formulation for the supremizer $\boldsymbol{s_h}(\mu)$:

$$(\nabla \boldsymbol{s_h}(\mu), \nabla \boldsymbol{v_h}(\mu)) = -p_h(\mu) \mathrm{div} \boldsymbol{v_h}(\mu),$$

and by adding Nitsche terms and stabilization terms we obtain the following formulation:

$$\begin{aligned}(\nabla \boldsymbol{s_h}(\mu^{(i)}), \nabla \boldsymbol{v_h}(\mu^{(i)})) - (\nabla \boldsymbol{s_h}(\mu^{(i)})\boldsymbol{n}, \boldsymbol{v_h})(\mu^{(i)}) - (\nabla \boldsymbol{v_h}(\mu^{(i)})\boldsymbol{n}, \boldsymbol{s_h}(\mu^{(i)}))+ \\ + \frac{\lambda_s}{h}(s_h(\mu^{(i)}), v_h(\mu^{(i)})) + g^{GP}(\boldsymbol{s_h}(\mu^{(i)}), \boldsymbol{v_h}(\mu^{(i)})) = -p_h(\mu^{(i)}) \mathrm{div} \boldsymbol{v_h}(\mu^{(i)}),\end{aligned}$$

where the Ghost Penalty term is given by:

$$g^{GP}(\boldsymbol{s_h}(\mu^{(i)}), \boldsymbol{v_h}(\mu^{(i)})) = \sum_{0 \le j \le k} \gamma_s^j h^{2j+1} \langle [\![\partial_n^j \boldsymbol{s_h}(\mu^{(i)})]\!], [\![\partial_n^j \boldsymbol{v_h}(\mu^{(i)})]\!] \rangle,$$

for all $j = 1, \dots, M$, where we recall that k is the degree of the piecewise polynomial FE basis functions on each element of the active mesh. The values of the constants used are reported in Table 5.2.

The formulation that we use for the supremizer enrichment is a special instance of a Poisson problem, and therefore, we use the CutFEM Poisson discretization for its approximation, component by component. We employ the same natural smooth extension (and the same extended FE space used for velocity) also for the supremizer, thus obtaining the extendend snapshots $\hat{\boldsymbol{s}}_h$. These snapshots are then collected in the snapshot matrix

$$\boldsymbol{\mathcal{S}}_s = [\hat{\underline{\mathbf{s}}}_h(\mu^{(1)}), \dots, \hat{\underline{\mathbf{s}}}_h(\mu^{(M)})] \in \mathbb{R}^{N_u^h \times M},$$

We then carry out a compression by POD on the snapshots matrices, namely $\boldsymbol{\mathcal{S}}_u$, $\boldsymbol{\mathcal{S}}_s$ and $\boldsymbol{\mathcal{S}}_p$, following e.g. [108]. This derives an eigenvalue problem, that for the velocity for example reads:

$$\boldsymbol{\mathcal{C}}^u \boldsymbol{Q}^u = \boldsymbol{Q}^u \boldsymbol{\Lambda}^u, \qquad \text{for } \boldsymbol{\mathcal{C}}^u_{ij} = (\hat{\boldsymbol{u}}_h(\mu^{(i)}), \hat{\boldsymbol{u}}_h(\mu^{(j)}))_{L^2(\hat{\mathcal{I}}_h)},\ i,j = 1, \dots, M,$$

where $\boldsymbol{\mathcal{C}}^{\boldsymbol{u}}$ is the correlation matrix derived from the μ-independent snapshots, $\boldsymbol{Q}^u$ is an eigenvectors square matrix and $\boldsymbol{\Lambda}^u$ is a diagonal matrix of eigenvalues. Similar eigenvalue problems can be derived for the supremizer and for the pressure.
We then obtain a set $\{\boldsymbol{\Phi}^u_1, \dots, \boldsymbol{\Phi}^u_N, \boldsymbol{\Phi}^s_1, \dots, \boldsymbol{\Phi}^s_N\}$ of $2N$ basis functions for the reduced order approximation of the velocity, and a set $\{\Phi^p_1, \dots, \Phi^p_N\}$ of N basis functions for the reduced order approximation of the pressure. We define: $\hat{V}_N$, the enriched reduced basis space for the velocity, and $\hat{Q}_N$, the reduced basis space for the pressure:

$$\hat{V}_N = \text{span}\{\boldsymbol{\Phi}^{u,s}_1, \dots, \boldsymbol{\Phi}^{u,s}_{2N}\}, \qquad \hat{Q}_N = \text{span}\{\Phi^p_1, \dots, \Phi^p_N\}$$

where $N < M$ is chosen according to the eigenvalue decay of $\boldsymbol{\Lambda}^u_{ii}$ and $\boldsymbol{\Lambda}^p_{ii}$, see for instance [156, 22]. We introduce the online velocity $\boldsymbol{u}_N(\mu)$ and the online pressure $p_N(\mu)$:

$$\boldsymbol{u}_N(\mu) := \sum_{i=1}^{2N} \underline{\mathbf{u}}^i_N(\mu) \boldsymbol{\Phi}^{u,s}_i = \boldsymbol{L}_{u,s} \underline{\mathbf{u}}_N(\mu), \tag{5.15}$$

$$p_N(\mu) := \sum_{i=1}^{N} \underline{p}^i(\mu) \Phi^p_i = \boldsymbol{L}_p \underline{p}_N(\mu), \tag{5.16}$$

where $\boldsymbol{L}_{u,s}$ and $\boldsymbol{L}_p$ are rectangular matrices containg the FE degrees of freedom of the basis of $\hat{V}_N$ and $\hat{Q}_N$. The parameter *dependent* solution vector $\underline{\mathbf{u}}_N(\mu)$, $\underline{p}_N(\mu) \in \mathbb{R}^N$ and the parameter *independent* reduced basis functions $\boldsymbol{\Phi}^{u,s}_i$, Φ^p_i are the key ingredients necessary to perform a Galerkin projection of the full system onto the aforementioned reduced basis space. By introducing $U_N(\mu) = (\underline{\mathbf{u}}_N(\mu), \underline{p}_N(\mu))$, the formulation, at the reduced order level, of the steady Navier–Stokes problem, reads as follows:

$$\begin{bmatrix} \boldsymbol{L}^T_{u,s}(\boldsymbol{A}(\mu) + \boldsymbol{N}(\underline{\mathbf{u}}_N(\mu); \mu))\boldsymbol{L}_{u,s} & \boldsymbol{L}^T_{u,s}\boldsymbol{B}^{\boldsymbol{T}}(\mu)\boldsymbol{L}_p \\ \boldsymbol{L}^T_p(\boldsymbol{B}(\mu) + \hat{\boldsymbol{B}}(\mu))\boldsymbol{L}_{u,s} & \boldsymbol{L}^T_p C(\mu)\boldsymbol{L}_p \end{bmatrix} U_N(\mu) - \begin{bmatrix} \boldsymbol{L}^T_{u,s}\boldsymbol{F}_1(\mu) \\ \boldsymbol{L}^T_p \boldsymbol{F}_2(\mu) \end{bmatrix} = 0.$$

For the sake of the exposition, let us denote the vector on the left hand side of the above

equation (the reduced residual) as $R_N(U_N(\mu);\mu)$. In the above POD-ROM solution, we clarify that we have to assemble the matrices of the high fidelity system. For a "cheaper" in time execution and less computation resources costs, one could achieve further improvement employing hyper reduction techniques as in [183, 17, 40, 162].

5.4.2 Unsteady case

Similarly to what has been done in the previous paragraph, in the time dependent case an *offline/online* procedure will be employed, that will lead to the generation of a proper reduced basis set. Since the system is both (geometrical) parameter and time-dependent, we sample not only the geometrical parameter μ, but also the time t, with the sample points $t^k \in \{t^0, \dots, t^{N_t}\} \subset [0, T]$. This procedure is computationally more expensive and results in a much larger total number of snapshots to be collected with respect to the static system: the total number of snapshots that we collect is now equal to $\widehat{M} = M \cdot N_t$. The snapshots matrices $\boldsymbol{\mathcal{S}_u}$, $\boldsymbol{\mathcal{S}_s}$ and $\boldsymbol{\mathcal{S}}_p$ are then given by:

$$\boldsymbol{\mathcal{S}}_u = [\underline{\hat{\boldsymbol{u}}}_h(\mu^{(1)}, t^0), \dots, \underline{\hat{\boldsymbol{u}}}_h(\mu^{(1)}, t^{N_t}), \dots, \underline{\hat{\boldsymbol{u}}}_h(\mu^{(M)}, t^0), \dots, \underline{\hat{\boldsymbol{u}}}_h(\mu^{(M)}, t^{N_t})] \in \mathbb{R}^{N_u^h \times \widehat{M}}, \tag{5.17}$$

$$\boldsymbol{\mathcal{S}}_s = [\underline{\hat{\boldsymbol{s}}}_h(\mu^{(1)}, t^0), \dots, \underline{\hat{\boldsymbol{s}}}_h(\mu^{(1)}, t^{N_t}), \dots, \underline{\hat{\boldsymbol{s}}}_h(\mu^{(M)}, t^0), \dots, \underline{\hat{\boldsymbol{s}}}_h(\mu^{(M)}, t^{N_t})] \in \mathbb{R}^{N_u^h \times \widehat{M}}, \tag{5.18}$$

$$\boldsymbol{\mathcal{S}}_p = [\underline{\hat{p}}_h(\mu^{(1)}, t^0), \dots, \underline{\hat{p}}_h(\mu^{(1)}, t^{N_t}), \dots, \underline{\hat{p}}_h(\mu^{(M)}, t^0), \dots, \underline{\hat{p}}_h(\mu^{(M)}, t^{N_t})] \in \mathbb{R}^{N_p^h \times \widehat{M}}, \tag{5.19}$$

and we solve an eigenvalue problem like the one introduced in the previous subsection 5.4.1.

Finally, adopting the notation of subsection 5.4.1 we end up with the reduced basis spaces

$$\hat{V}_N = \text{span}\{\boldsymbol{\Phi}_1^{u,s}, \dots, \boldsymbol{\Phi}_{2N}^{u,s}\}, \qquad \hat{Q}_N = \text{span}\{\Phi_1^p, \dots, \Phi_N^p\},$$

Let us now denote by $(\boldsymbol{u}_N^n(\mu), p_N^n(\mu))$ the reduced solution at time-step t^n, for $n = 0, \dots, N_t$, where $\boldsymbol{u}_N^n(\mu)$ and $p_N^n(\mu)$ are defined as in (5.15) and in (5.16), respectively. Through a Galerkin projection of the full-order system of equations onto the POD reduced basis spaces we can derive the subsequent reduced algebraic system for the unknown $U_h^{n+1}(\mu) = (\boldsymbol{u}_N^{n+1}(\mu), p_N^{n+1}(\mu))$:

$$\begin{bmatrix} \boldsymbol{L}_{u,s}^T \boldsymbol{M} \boldsymbol{L}_{u,s} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} U_N^{n+1}(\mu) + \tau_{n+1} R_N(U_N^{n+1}(\mu);\mu) = \begin{bmatrix} \boldsymbol{L}_{u,s}^T \boldsymbol{M} \boldsymbol{L}_{u,s} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} U_N^n(\mu). \tag{5.20}$$

5.4.3 Unsteady case with time–dependent geometry

For the unsteady Navier–Stokes with time–dependent geometry we sample the time interval $[0, T]$ with the sample points $\{t^0, \dots, t^{N_t}\}$; for each sample point t^n we then compute the corresponding geometrical parameter μ^n, for $n = 0, \dots, N_t$. Thus, in

this case, the snapshots matrices $\boldsymbol{\mathcal{S}_u}$, $\boldsymbol{\mathcal{S}_s}$ and $\boldsymbol{\mathcal{S}}_p$ are:

$$\boldsymbol{\mathcal{S}}_u = [\underline{\hat{\boldsymbol{u}}}_h(\mu^0, t^0), \dots, \underline{\hat{\boldsymbol{u}}}_h(\mu^{N_t}, t^{N_t})] \in \mathbb{R}^{N_u^h \times N_t}, \tag{5.21}$$

$$\boldsymbol{\mathcal{S}}_s = [\underline{\hat{\boldsymbol{s}}}_h(\mu^0, t^0), \dots, \underline{\hat{\boldsymbol{s}}}_h(\mu^{N_t}, t^{N_t})] \in \mathbb{R}^{N_u^h \times N_t}, \tag{5.22}$$

$$\boldsymbol{\mathcal{S}}_p = [\underline{\hat{p}}_h(\mu^0, t^0), \dots, \underline{\hat{p}}_h(\mu^{N_t}, t^{N_t})] \in \mathbb{R}^{N_p^h \times N_t}, \tag{5.23}$$

Again, by adopting the notation of subsection 5.4.1 we end up with the reduced basis spaces

$$\hat{V}_N = \text{span}\{\boldsymbol{\Phi}_1^{u,s} \dots, \boldsymbol{\Phi}_{2N}^u\}, \qquad \hat{Q}_N = \text{span}\{\Phi_1^p, \dots, \Phi_N^p\},$$

We denote by $(\boldsymbol{u}_N^n(\mu^n), p_N^n(\mu^n))$ the reduced solution at time–step t^n, for $n = 0, \dots, N_t$, defined as in (5.15) and (5.16). We employ again a Galerkin projection of the full-order system of equations onto the POD reduced basis spaces; we derive the subsequent reduced algebraic system of equations for the unknown $U_h^{n+1}(\mu) = (\boldsymbol{u}_N^n(\mu^n), p_N^n(\mu^n))$:

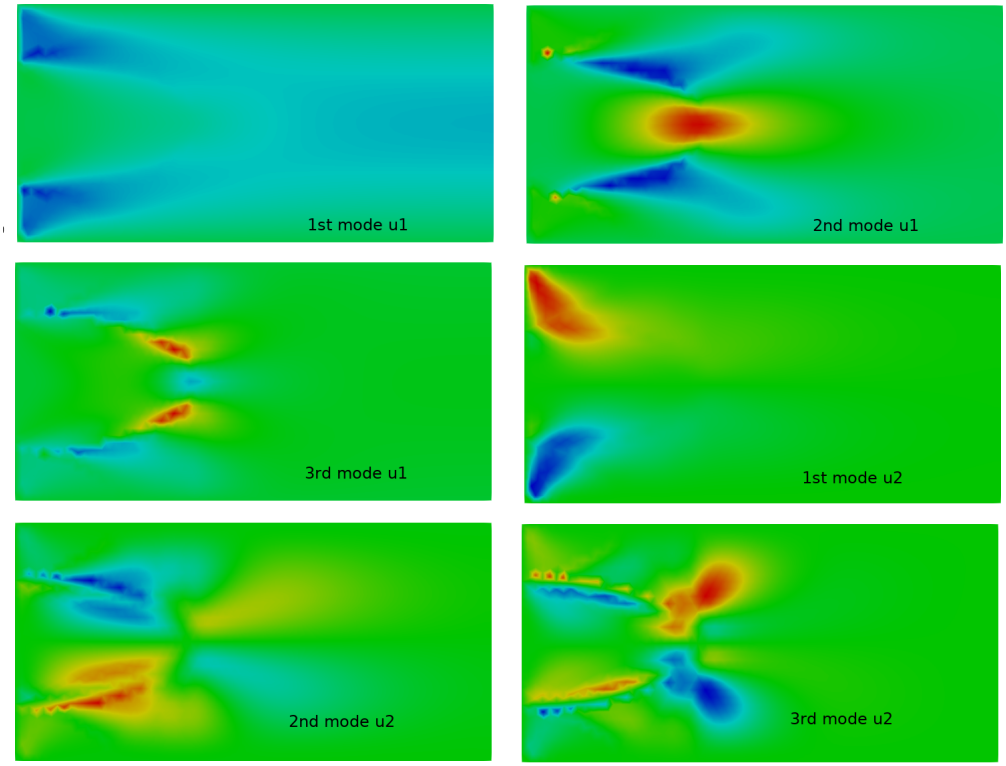
$$\begin{bmatrix} \boldsymbol{L}_{u,s}^T \boldsymbol{M} \boldsymbol{L}_{u,s} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} U_N^{n+1}(\mu^{n+1}) + \tau_{n+1} R_N(U_N^{n+1}(\mu^{n+1}); \mu^{n+1}) = \begin{bmatrix} \boldsymbol{L}_{u,s}^T \boldsymbol{M} \boldsymbol{L}_{u,s} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} U_N^n(\mu^n).$$

Remark 5.4.1. Here we have underlined the dependence of the geometrical parameter μ on time t because in the strong formulation of the original problem the geometry of the domain is time dependent; nevertheless we would like to remember the reader that we chose to implement a natural smooth extension of the discrete CutFEM solution, therefore all the snapshots, as well as the reduced solution, are defined on the whole background mesh.

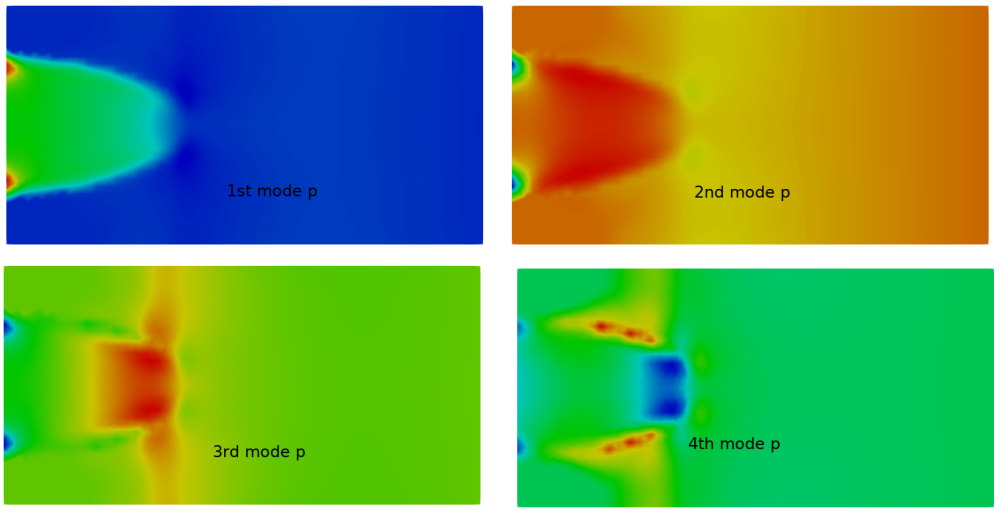
5.5 Numerical results

5.5.1 Steady Navier–Stokes

In this paragraph we present the results obtained by applying the aforementioned reduction techniques to our model problem. For our simulation, the fluid viscosity is $\nu_f = 0.05\ cm^2/s$ and the fluid density is $\rho_f = 1\ g/cm^3$. We impose a constant velocity profile at the inlet, $\boldsymbol{u}_{\text{in}} = (1, 0)$, and we impose a homogeneous condition on the vertical component of the velocity at the top and bottom walls of the rectangle. The reduced basis have been obtained with a Proper Orthogonal Decomposition on the set of snapshots: this reduction technique, although costly in computational terms, is very useful as it gives an insight on the rate of decay of the eigenvalues related to each component of the solution. We take $N_{\text{train}} = 150$, and we generate randomly N_{train} uniformly distributed values for the parameter μ. We then run a POD on the collected set of snapshots and we obtain our basis functions, with which we are going to compute the reduced solutions $(\boldsymbol{u}_N(\mu^{(i)}), p_N(\mu^{(i)}))$, where $i = 1, \dots, N_{\text{test}}$, and N is the number of basis functions that we use. Figures 5.5(a) and 5.5(b) give an example of the first modes that we obtain with this procedure, whereas in Figure 5.6 we report the decay of the eigenvalues for all the components of the solution and for the supremizer. To test the reduced order model we generate randomly $N_{\text{test}} = 30$ uniformly distributed values values for $\mu \in \mathcal{P}_{\text{test}}$. We are interested in the behavior of the relative approximation error that we obtain by changing the number of basis functions N used to build the reduced solution. In order to do this we let N vary in a discrete set $\mathcal{N}$: for a fixed value of $N \in \mathcal{N}$, and for each $\mu^{(i)}$, $i = 1, \dots, N_{\text{test}}$,



(a) First six modes for the velocity in the steady Navier–Stokes problem.



(b) First four modes for the pressure in the steady Navier–Stokes problem.

FIGURE 5.5: Steady system: Some reduced basis modes for velocity and pressure for a geometrically patrametrized Navier–Stokes system.

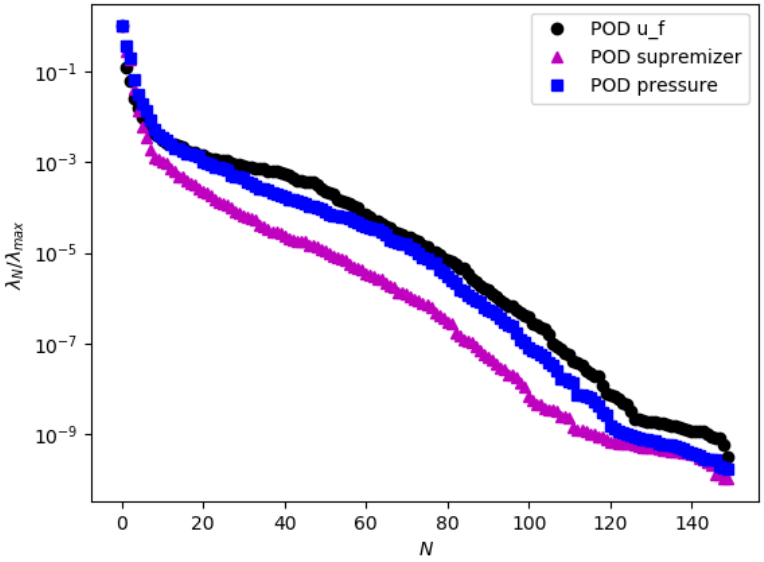


FIGURE 5.6: Steady case: POD eigenvalues decay for the fluid velocity $\boldsymbol{u}$ (black), the fluid pressure p (blue), and the fluid supremizer $\boldsymbol{s}$ (magenta), for a set of $N_{train} = 150$ snapshots.

we compute both the reduced solution $(\boldsymbol{u}_N(\mu^{(i)}), p_N(\mu^{(i)}))$ and the corresponding full order solution $(\boldsymbol{u}_h(\mu^{(i)}), p_h(\mu^{(i)}))$. We compute the L^2 relative error $\epsilon_u^{N,i}$ for the velocity and the relative error $\epsilon_p^{N,i}$ for the pressure; then we compute the average approximation errors $\bar{\epsilon}_u^N$ and $\bar{\epsilon}_p^N$ for every $N \in \mathcal{N}$, defined as:

$$\bar{\epsilon}_u^N = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \epsilon_u^{N,i}.$$

Figure 5.7 shows the relative approximation errors plotted against the number N of basis functions used, with the use of the supremizer enrichment at the reduced order level and without the supremizer enrichment, respectively. As we can see, using a supremizer enrichment at the reduced level allows us to obtain a better approximation of the pressure: we have almost one order of magnitude of difference in the relative error with $N = 20$ basis functions for the pressure, with or without the supremizer. Figure 5.8(b) shows the approximation error for the pressure, for a given test value of the parameter μ, without the supremizer enrichment; the error has been calculated in the L^2 norm. Figure 5.8(a) shows the approximation error for the pressure, for the same parameter value, with the supremizer enrichment: as we can see, the supremizer is useful in obtaining a much more accurate approximation of the reduced order pressure. It is worth to mention that the approximation error tends to concentrate near the cut between the physical domain and the background mesh, similar to to experiments in the works of [102, 105, 106, 103], phenomenon which will be studied in a future work.

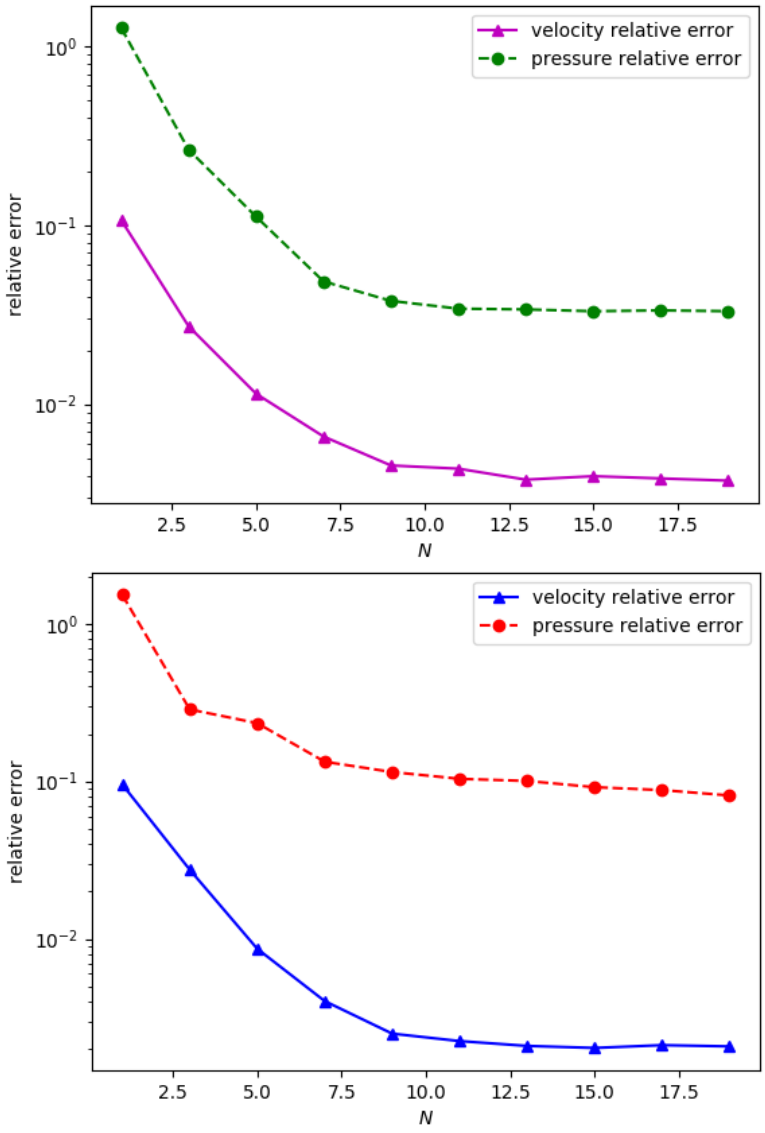
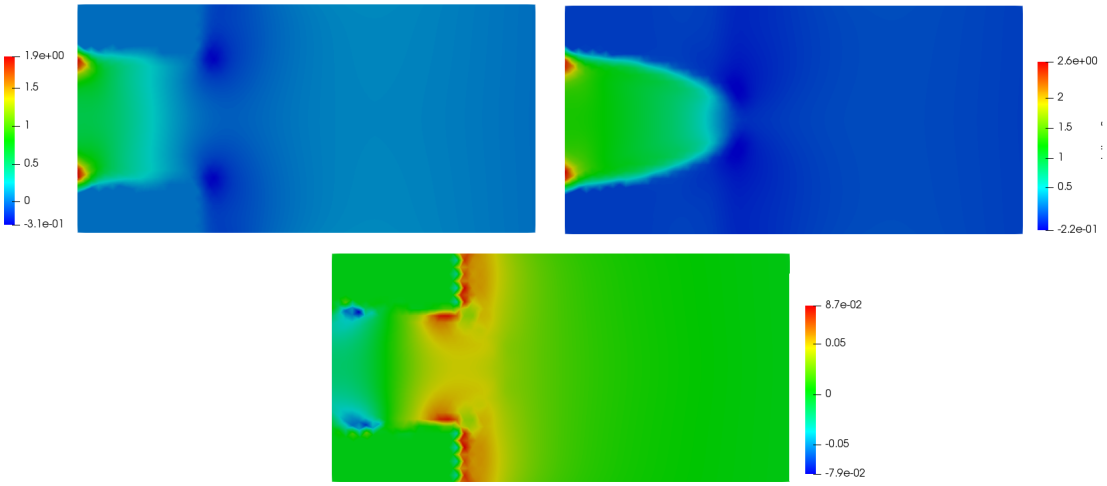
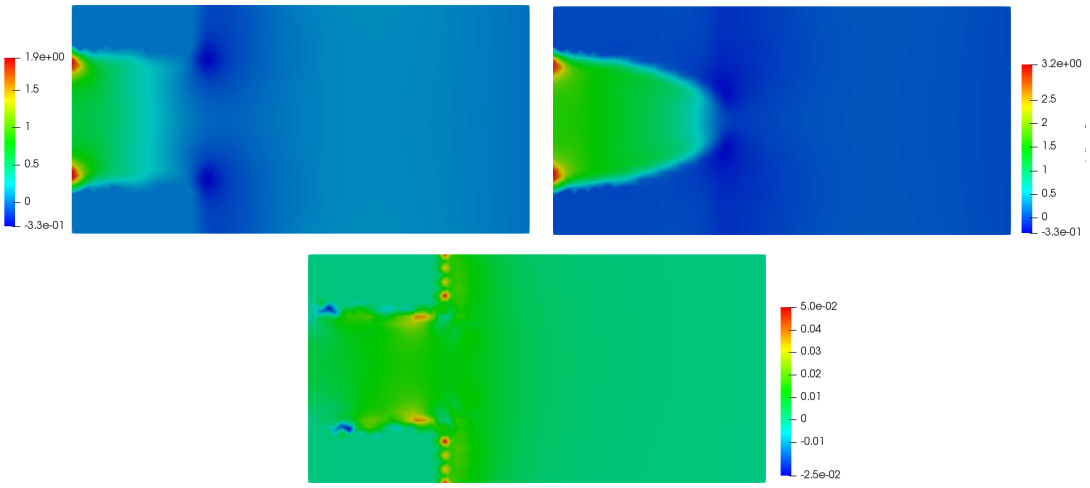


FIGURE 5.7: Steady case: Mode dependent errors between high fidelity and reduced order approximation, with (top) and without (bottom) the supremizer enrichment.



(a) Without supremizer enrichment.



(b) With supremizer enrichment.

FIGURE 5.8: Steady case: Uncut geometry and the high fidelity pressure solution for parameter $\mu = -0.015854$ (left), reduced order solution for the same μ (right) and approximation error (middle)

5.5.2 Unsteady Navier–Stokes

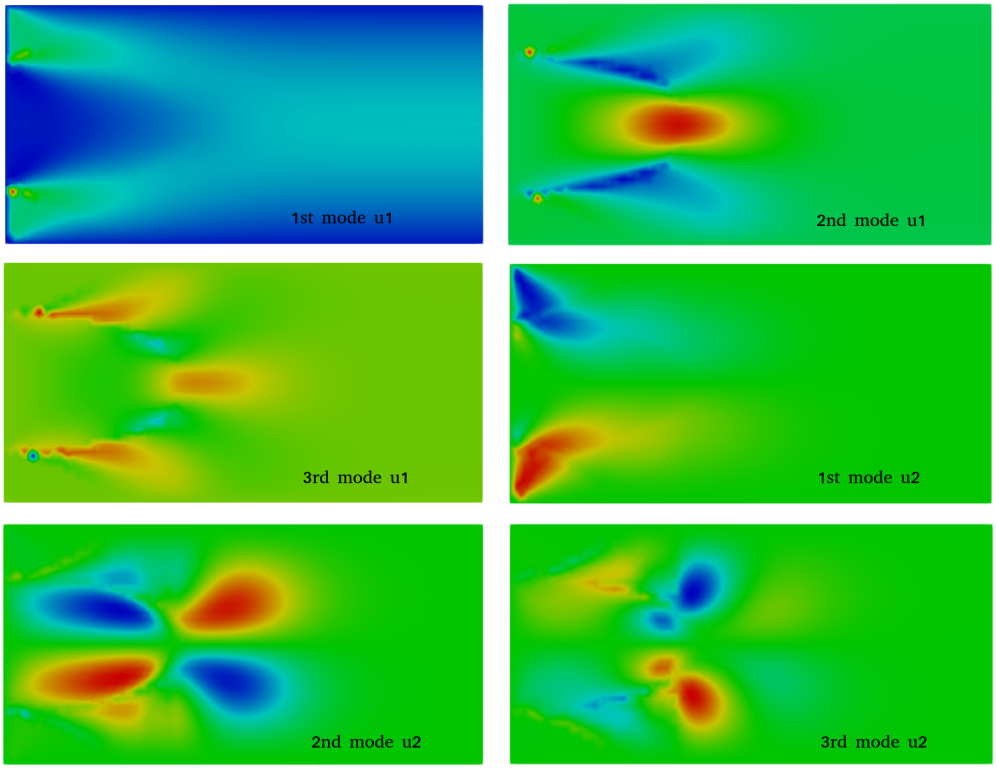
1) Stationary in time geometry case

In this paragraph we present the results obtained by applying the proposed reduction technique to a time dependent case. The time-step used in our simulation is $\tau = 0.011s$, and the final time is $T = 0.7s$. The fluid viscosity is $\nu_f = 0.05\ cm^2/s$ and the fluid density is $\rho_f = 1\ g/cm^3$. We impose a constant inlet velocity $\boldsymbol{u}_{\text{in}} = (1, 0)$, and we impose a homogeneous boundary condition for the vertical component of the fluid velocity at the top and bottom walls of the rectangle. We now take $N_{\text{train}} = 200$, and we generate randomly N_{train} uniformly distributed values for the parameter μ. We also remind that we sample the time interval $[0, T]$ with an equispaced sampling $\{t_0, \dots, t_{N_t}\}$. We then run a POD on the set of snapshots collected, and we obtain our basis functions with which we are going to compute the reduced solutions $(\boldsymbol{u}_N(t, \mu_i), p_N(t, \mu_i))$, where $i = 1, \dots, N_{\text{test}}$, and N is the number of basis functions that we use. Figures 5.9 gives an example of the first modes that we obtain with this procedure, whereas in Figure 5.10 we show the rate of decay of the eigenvalues for all the components of the solution and for the supremizer. To test the reduced order model we generate randomly $N_{\text{test}} = 30$ uniformly distributed values for $\mu \in \mathcal{P}_{\text{test}}$.
We are again interested in the behavior of the relative approximation error as a function of the number N of basis functions used at the reduced order level. We therefore let N vary in a discrete set $\mathcal{N}$: for a fixed value of $N \in \mathcal{N}$, and for each μ_i, $i = 1, \dots, N_{\text{test}}$, we compute both the reduced solution $(\boldsymbol{u}_N(t, \mu_i), p_N(t, \mu_i))$ and the corresponding high order solution $(\boldsymbol{u}_h(t, \mu_i), p_h(t, \mu_i))$. We calculate the L^2 relative error $\epsilon_{u,t_k}^{N,i}$ for the velocity and the relative error $\epsilon_{p,t_k}^{N,i}$ for the pressure at time t_k, by taking an average of these relatives error we obtain the mean approximation error $\epsilon_u^{N,i}$ for $\boldsymbol{u}$ and $\epsilon_p^{N,i}$ for p, for each $\mu_i \in \mathcal{P}_{\text{test}}$. Finally we compute the average approximation errors $\overline{\epsilon}_u^N$ and $\overline{\epsilon}_p^N$ for every $N \in \mathcal{N}$, defined as:

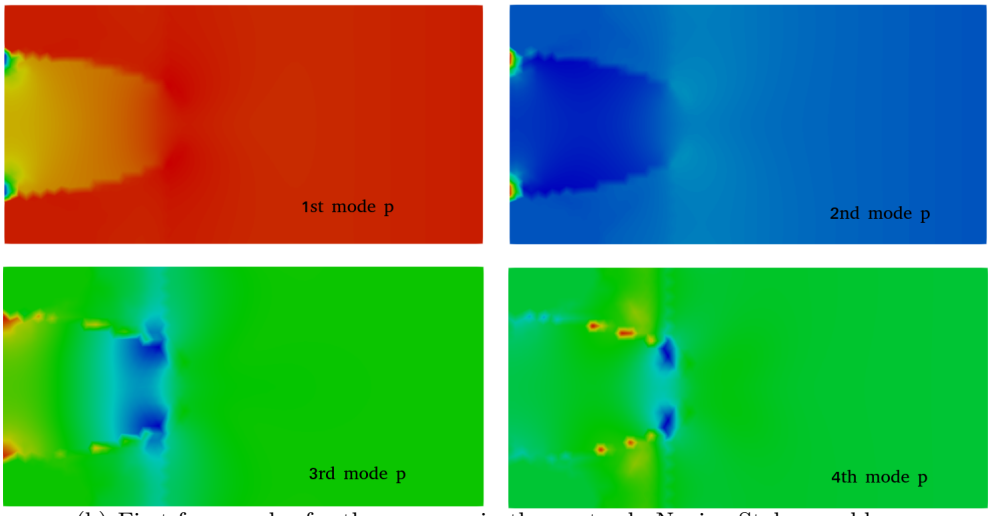
$$\overline{\epsilon}_u^N = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \epsilon_u^{N,i}.$$

Figure 5.11 shows the relative approximation errors plotted against the number N of basis functions used, with and without of the supremizer enrichment at the reduced order level, respectively. As we can see, using a supremizer enrichment at the reduced level allows us to obtain a better approximation of the pressure: for example with $N = 40$ basis functions we have an approximation error $\overline{\epsilon}_p^N$ that is almost one order of magnitude smaller with respect of the approximation error that we obtain without the supremizer enrichment.

Figure 5.12(a) shows the approximation error for the pressure, for a given value of the test parameter μ, without the supremizer enrichment; the error has been calculated in the L^2 norm. Figure 5.12(b) shows the approximation error for the pressure, for the same parameter value, with the supremizer enrichment: as we can see, by employing a supremizer enrichment, we obtain a much more accurate approximation of the reduced order pressure. Figure 5.13 shows the approximation error for the fluid velocity $\boldsymbol{u}_f$ for a given value of the test parameter μ, at the final time-step of the simulation.



(a) First six modes for the velocity in the unsteady Navier–Stokes problem.



(b) First four modes for the pressure in the unsteady Navier–Stokes problem.

FIGURE 5.9: Unsteady system with time–independent geometry: some reduced basis modes for velocity and pressure.

TABLE 5.3: Values for the constants in the equation (5.12) of motion of the cylinder .

Constant	**Value**
$\mu(0)$	$(-1.5, 0.25)$
A	0.015625
$\mu(t)$	$\left(\frac{\mu_M - \mu_m}{T}\right)t + \mu_m$
μ_M	0.5
μ_m	-0.5

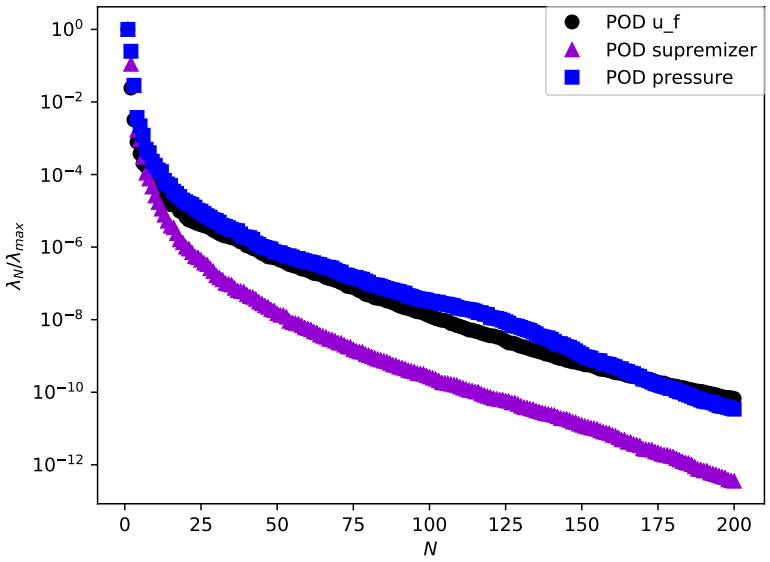


FIGURE 5.10: POD eigenvalues decay in the time dependent Navier–Stokes problem, for the fluid velocity $\boldsymbol{u}$ (black), the fluid pressure p (blue), and the fluid supremizer $\boldsymbol{s}$ (magenta), for a set of $N_{train} = 200$ snapshots.

2) Evolutionary in time geometry case

In this subsection we present the results obtained by applying the proposed reduction technique to a time dependent case, with evolutionary in time level set geometry. The values of the quantities that determine the oscillatory motion of the cylinder immersed in the fluid are reported in Table 5.3. The time-step used in our simulation is $\tau = 0.011s$, and the final time is $T = 400\tau = 4.4s$. Similarly to paragraph 5.5.2, we take the fluid viscosity $\nu_f = 0.05\ cm^2/s$, the fluid density $\rho_f = 1\ g/cm^3$, inlet velocity $\boldsymbol{u}_{\text{in}} = (1, 0)$ and the same boundary conditions. We choose $N_{\text{train}} = N_t = 400$ time instances, and we run a POD on the set of the collected snapshots, and we obtain our basis functions with which we are going to compute the reduced solutions $(\boldsymbol{u}_N(t^i, \mu^i), p_N(t^i, \mu^i))$, where $i = 1, \dots, N_{\text{t}}$, and N is the number of basis functions that we use. Figures 5.14 represent some of the first modes that are derived with the POD procedure, while in Figure 5.15 we show the rate of decay of the eigenvalues for the velocity, pressure snapshot solutions and for the supremizer. Finally, we are interested in the behavior of the relative approximation error as a function of the number N of basis functions used at the reduced solution stage. We again let N vary in a discrete set $\mathcal{N}$: for a fixed value of $N \in \mathcal{N}$, and for each μ^i, $i = 1, \dots, N_{\text{t}}$, we compute both the reduced solution $(\mathbf{u}_N(t^i, \mu^i), p_N(t^i, \mu^i))$ and the corresponding high fidelity solution $(\boldsymbol{u}_h(t^i, \mu^i), p_h(t^i, \mu^i))$. We compute the L^2 relative error $\epsilon^{N,i}_{u,t_k}$ for the velocity and the relative error $\epsilon^{N,i}_{p,t_k}$ for the pressure at time t_k; by taking the time average of these relative errors we obtain the mean approximation error $\epsilon^{N,i}_{u}$ for $\boldsymbol{u}$ and $\epsilon^{N,i}_{p}$ for p, for each $\mu^i \in \mathcal{P}_{\text{t}}$.

Figures 5.17 and 5.18 visualizes the approximation error for the pressure, at final time T, with supremizer enrichment as well as the velocity error; the error has been calculated in the L^2 norm. In Figure 5.16 the mean with respect time relative errors

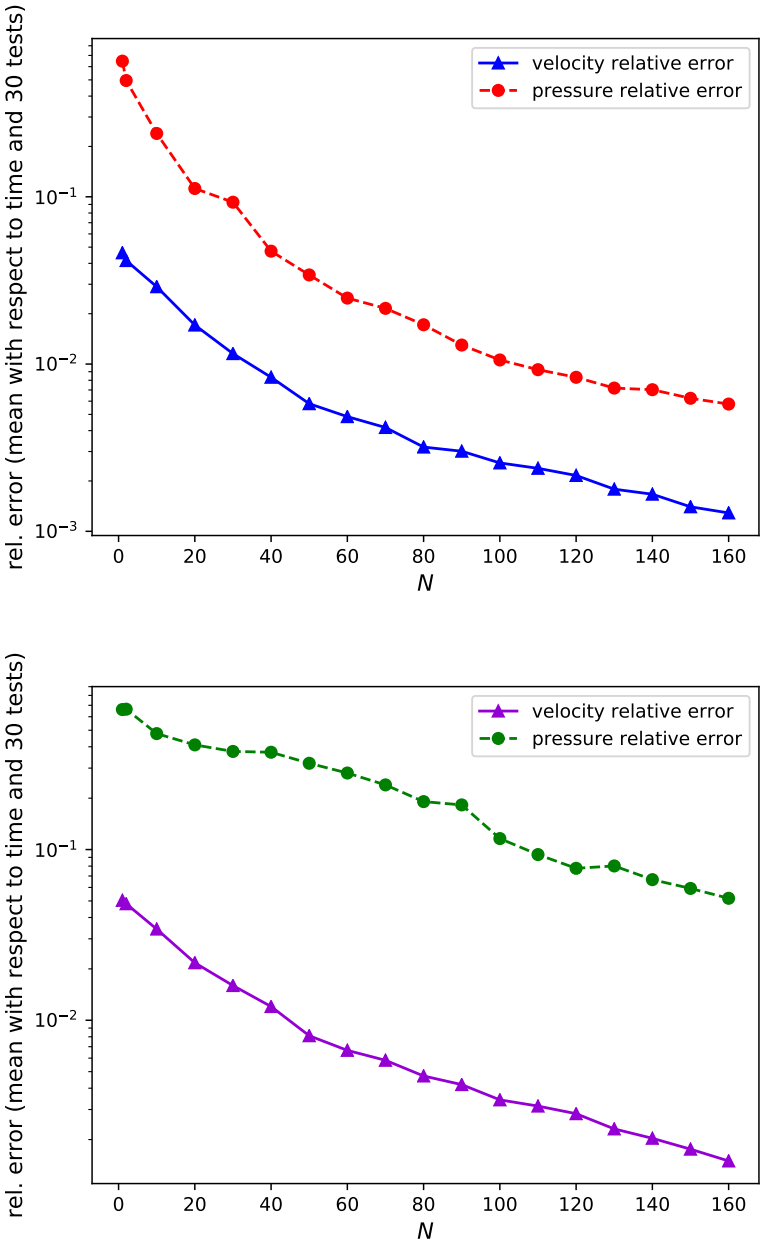
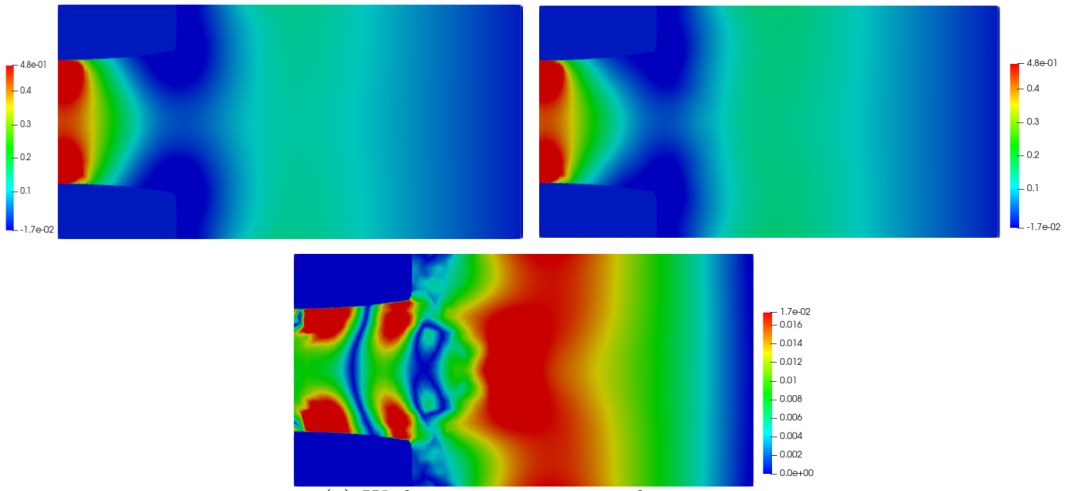
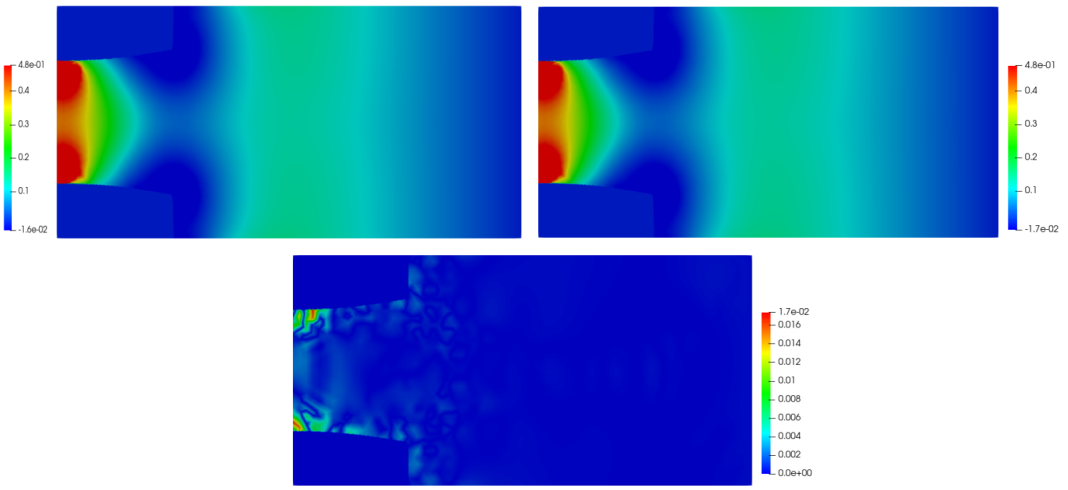


FIGURE 5.11: Unsteady case: mode dependent errors between high fidelity and reduced order approximation, with and without the supremizer enrichment.



(a) Without supremizer enrichment.



(b) With supremizer enrichment.

FIGURE 5.12: Unsteady case: cut geometry and the high fidelity pressure solution at final time at final time $T = 0.7$ for parameter $\mu = 0.050014$ (left), reduced order solution for the same μ (right) and approximation error (bottom).

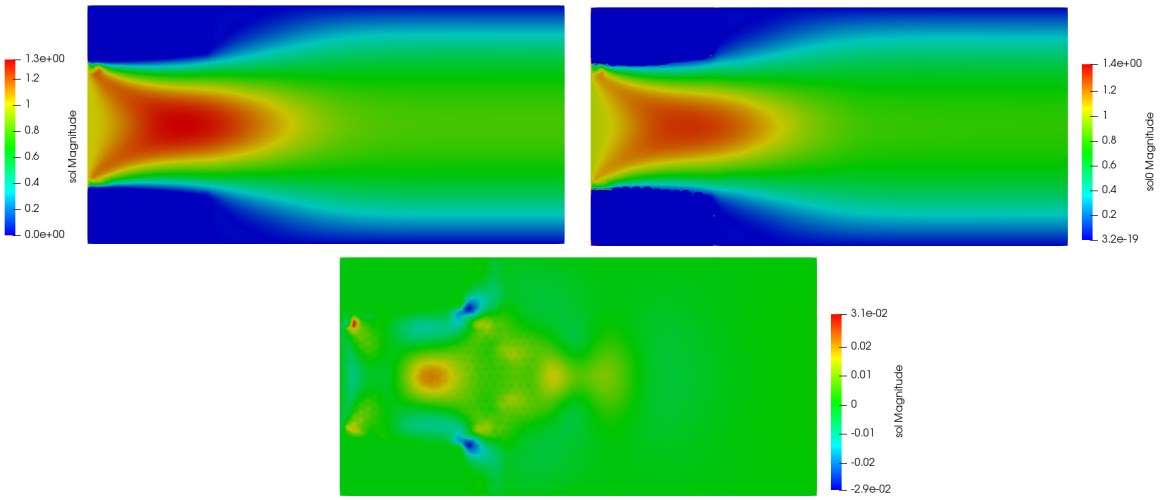


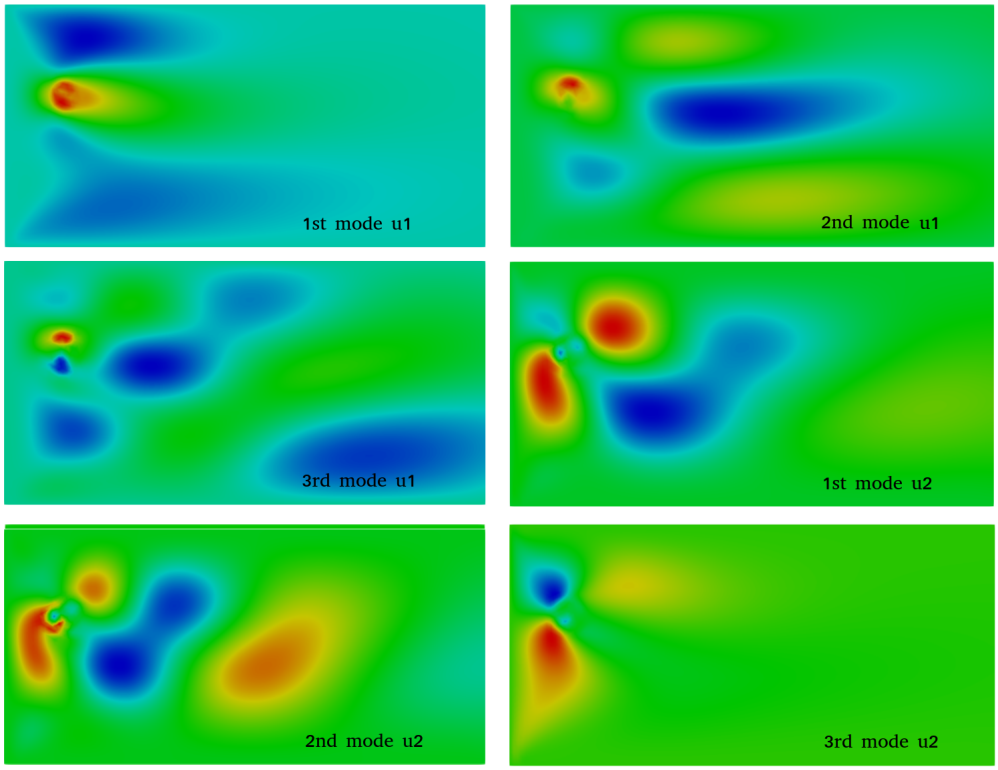
FIGURE 5.13: Unsteady case: cut geometry and high fidelity fluid velocity at final time $T = 0.7$ for parameter $\mu = 0.045406$ (left), reduced order solution for the same μ (right) and approximation error (bottom).

for various number of modes are visualized.

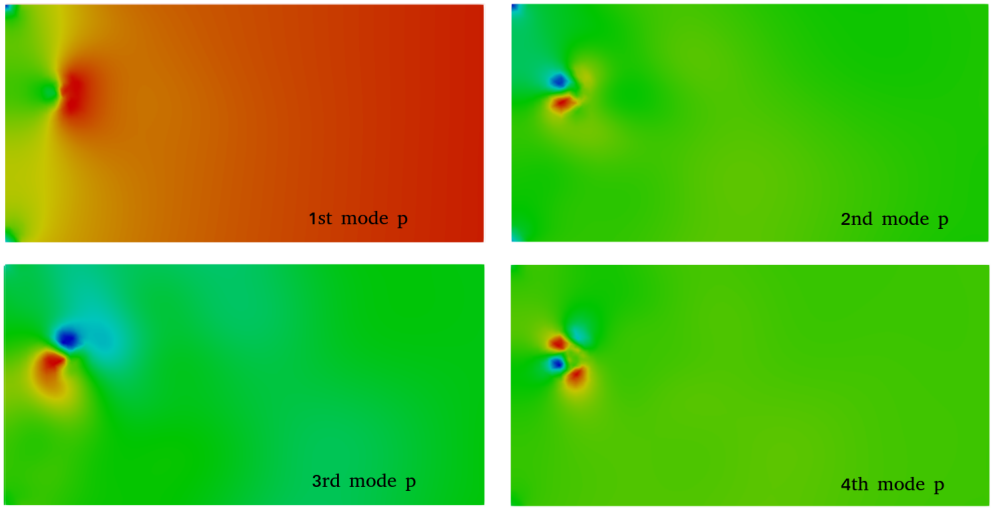
With this test case we can appreciate deeply the advantage that we obtain by employing the cutFEM reduced order model that we proposed. Indeed, at every new time–step t^i corresponds a different domain configuration, depending on the parameter μ^i; without a cutFEM formulation, we would have to remesh at every time–step, making the offline phase of a reduced order model prohibitive from the computational cost point of view. Moreover, results show that we are able to obtain good results at the reduced order level even without employing a snapshot transportation during the offline phase.

5.6 Conclusions

In this Chapter we have introduced a POD–Galerkin ROM approach for a geometrically parametrized two dimensional Navier–Stokes equation, both in the steady and in the unsteady case, extending the study also to a problem where the levelset geometry changes in time. The procedure that we have proposed shows many of the advantages that characterize CutFEM and reduced order methods. First of all, by choosing an unfitted mesh approach, we have shown that it is possible to work with geometries that can potentially change significantly the shape of (part of) the domain, as we can see from the examples in Figure 5.2. By employing an unfitted CutFEM approach at the full order level, we can let the geometrical parameter μ vary in a large interval of values: this helps to overcome one of the limitations of the standard Finite Element discretization, where a re–meshing would have been needed. At the reduced order level, we have extended the supremizer enrichment technique, widely used in the reduced basis community, to a CutFEM setting: the results that we have obtained confirm that the supremizer does actually significantly help in obtaining more accurate approximation of the fluid pressure allowing an inf-sup stable reduced basis preserving the stabilization effects.
A first future perspective is the extension of the work presented in this Chapter to



(a) First six modes for the velocity in the time dependent geometry problem.



(b) First four modes for the pressure in the time dependent geometry.

FIGURE 5.14: Unsteady system with evolutionary in time geometry: some reduced basis modes for velocity and pressure for the evolutionary in time Navier–Stokes system.

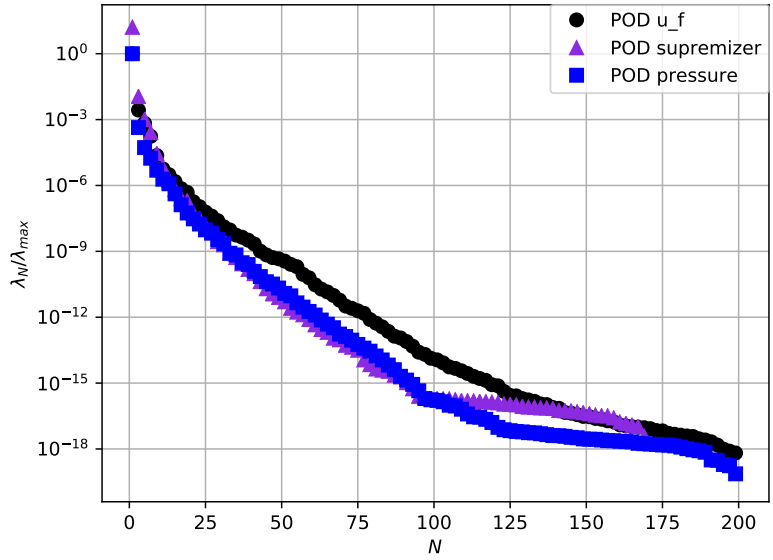


FIGURE 5.15: Evolutionary in time geometry: the POD eigenvalues decay for the fluid velocity $\mathbf{u}$ (black), the fluid pressure p (blue), and the fluid supremizer $\mathbf{s}$ (magenta), for a set of $N_{train} = N_t = 400$ (time instances) snapshots.

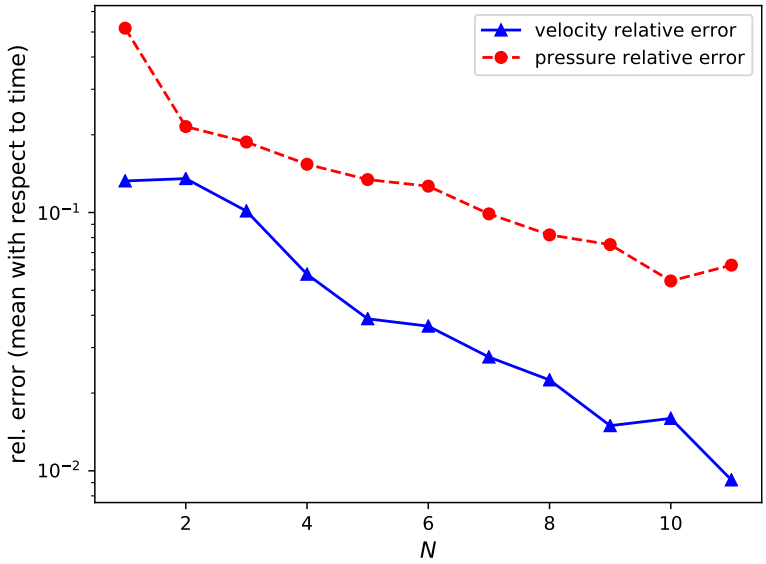


FIGURE 5.16: Evolutionary in time geometry: Mode dependent errors between high fidelity and reduced order approximation, with the supremizer enrichment.

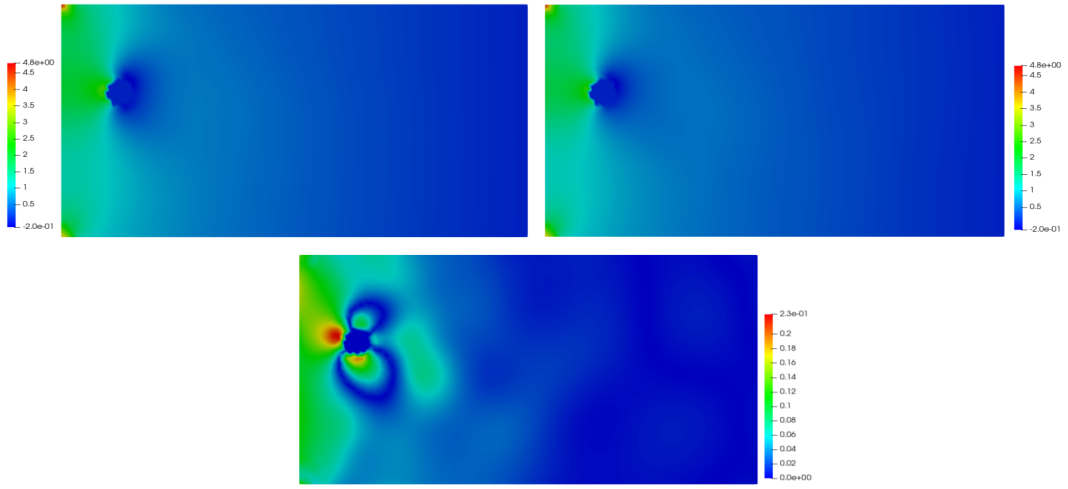


FIGURE 5.17: Unsteady case with evolutionary in time geometry: The high fidelity pressure solution at final time $t = T = 400\tau$ (left), reduced order solution for the same t (right) and approximation error (bottom). The results have been calculated with supremizer enrichment.

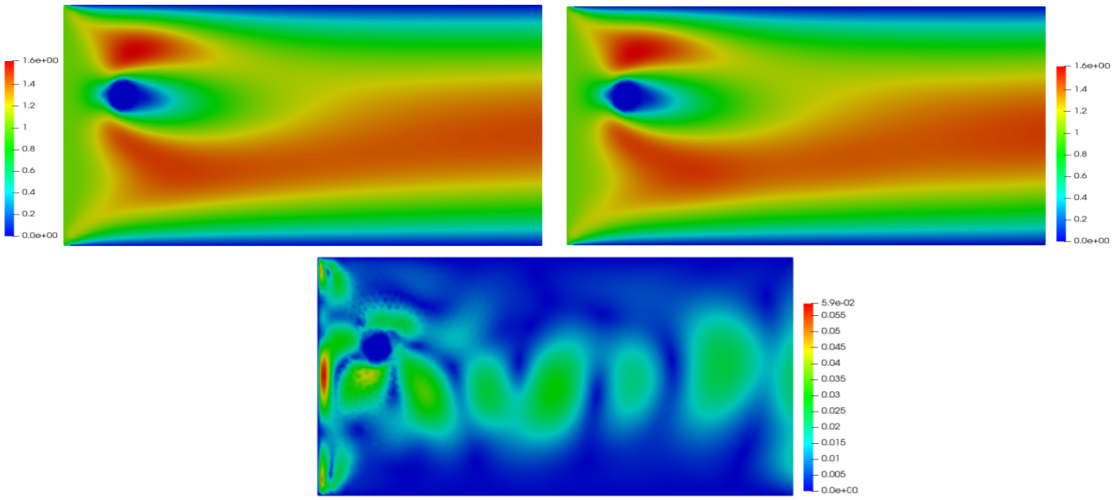


FIGURE 5.18: Unsteady case with evolutionary in time geometry: The high fidelity fluid velocity at final time $t = T = 400\tau$ (left), reduced order solution for the same t (right) and approximation error (bottom). The results have been calculated with supremizer enrichment.

multiphisics problems, i.e. in the situation where the deformation of the domain is governed by an unknown motion. The integration of a reduction procedure for coupled problems with a preprocessing technique such as the one presented in Chapter 4 would also be very interesting in the design of an algorithm that combines all the advantages of using an embedded Finite Element discretization and the advantages of reducing the Kolmogorov n–width of the solution manifold of the problem of interest.

Chapter 6

Conclusions and future perspectives

The goal of this thesis has been to analyze, test and adapt the widely used Reduced Basis Method to various situations; in particular, the main interest in this work has been the application of the RBM to Fluid–Structure Interaction problems. Through the study of different test cases, we have seen how the Reduced Basis Method can be coupled with various discretization techniques and with various algorithms, in order to obtain reduction procedures that can be applied to several situations: coupled problems where we want to reduce the dimensionality of the system to be solved, advection dominated problems and problems with a significant change in the physical domain. For all the considered cases, we have obtained promising results.

After the theoretical preliminaries presented in Chapter 1 and Chapter 2, in Chapter 3 we have introduced a particular class of algorithms used for solving Fluid–Structure Interaction problems, namely the partitioned (or segregated) procedures. With this algorithms we have seen that it is possible to decouple the fluid and the structure problem, and in this way we can use two different solvers for the fluid and for the solid. By adapting the Reduced Basis Method, we have designed a reduced order model that is a partitioned procedure as well: the main advantage of this reduction technique is given by the fact that, by solving separately the fluid and the solid problem, we are lowering the dimension of the systems to be solved in the online phase. Moreover, thanks to the introduction of some change of variables, we have seen that it is also possible to avoid the use of Lagrange multipliers to impose non–homogeneous boundary conditions, and this is a further advantage that makes this procedure very appealing. The results that we have obtained have a good level of accuracy, both in the geometrical parametrization case and in the non parametrized case. While testing this algorithm, we have seen that a partitioned procedure is demanding from the computational time point of view: this drawback is represented by the fact that, in the imposition of the coupling conditions through a Robin boundary condition, the constant α_{ROB} that makes the procedure more stable is heavily dependent on the time–step used. If we choose a time–step that is too big, the coupling becomes very weak and the algorithm becomes somehow insensitive to the effect of the interaction between the two physics. Aside from this aspect, which is peculiar to segregated approaches, the designed Reduced Basis Method performs very well, and thanks to the choice of a pressure Poisson formulation and to the derivation the basis functions for the fluid displacement directly from the basis functions of the solid displacement, we have been able to limit the number of unknowns in the online stage.

The entire work of Chapter 4 has been motivated by a Fluid–Structure Interaction test case, where the solution behaves like a travelling wave: this problem clearly does not satisfy one of the very basics and fundamental assumptions at the core of the reduction method, namely that the solution manifold can be well represented by linear spaces of small dimension. In this framework, we have decided to test a modified Reduced Basis Method, where in the offline phase we have added a preprocessing of the snapshots. We focused on two different test cases, that are interesting for this study: a fluid dynamics problem where a Karman vortex developing in the wake of a cylinder changes its direction of propagation due to the rotation of the cylinder, and a multiphysics problem, where the solution behaves like a travelling wave. One of the good aspects of the method presented is that, in the online phase, the problem is solved in the original domain, with the original mesh: indeed, we did not recast the problem formulation on the domain with the deformed mesh, where everything is aligned (the peak of the wave, the fluid vortex). This is a big advantage for the procedure, compared for example to an ALE method like the one proposed in [171], where the problem in the online phase is solved on the reference domain. For the FSI problem that we have studied, indeed, solving on the reference mesh could cause some issues, especially when the peak of the wave is quite far from the point where we want to align everything, as the re–mapping causes some triangles of the mesh to become degenerate. The results that we have obtained with this study are promising, and they confirm the necessity of some preprocessing procedure, in order to lower the dimension of the solution manifold.

In Chapter 5 we have designed a POD–Galerkin ROM approach for a geometrically parametrized CFD problem, both in the steady and in the unsteady case. Starting from a steady Navier–Stokes problem, formulated over a domain whose shape depends on a geometrical parameter, we moved to the unsteady Navier–Stokes equation, and we finally arrived to an unsteady problem, formulated over a domain whose shape changes in time, according to some prescribed law. As we have mentioned in the conclusion of Chapter 5, the combination of CutFEM discretization techniques with the Reduced Basis Method creates an extremely powerful tool in the framework of model order reduction, that can be employed also for those problems and those applications in which the geometry under consideration is very complicated, and where the geometrical parametrization can lead to a potentially big change in the physical domain. In our work we have adopted a natural smooth extension of the snapshots on the whole background mesh: this guarantees that we can work with basis functions and with functions spaces that do not depend on the geometrical parameter or on time. Another avdantage of the method presented is given by the fact that we have employed a supremizer enrichment technique, in order to obtain a more stable approximation of the pressure at the reduced order level: we have adapted the Poisson formulation for the supremizer to be suited for a CutFEM approach.

To conclude, the proposed methodologies have been tested for the following problems: an unsteady Fluid–Structure Interaction problem that couples an incompressible fluid with an elastic structure, also in the presence of geometrical parametrization of part of the domain; an unsteady Fluid–Structure Interaction problem that couples an incompressible fluid with a one dimensional string and that is dominated by an advection phenomenon; an unsteady fluid dynamics problem, also in the presence of a physical parameter, that exhibits a change in the direction of propagation of a vortex in the wake of a rotating cylinder, and lastly three fluid dynamics problems (steady and unsteady case), formulated over a domain whose shape changes significantly according to a parameter of our choice. In all the situations considered, the methodologies

presented in this thesis have performed very well, showing all the advantages of using the proposed modified reduction techniques in these different frameworks.

6.1 Overall perspectives

The work presented in this thesis represents an improvement in the framework of pursuing a model order reduction for multiphysics problems: indeed, we have improved the design of partitioned techniques by implementing a Reduced Basis Method that can be efficiently applied to FSI problems with a thick, two dimensional structure, also in the presence of a geometrical parametrization. We have enriched the offline phase of the Reduced Basis Method with a preprocessing procedure that leads to the design of a reduction technique that is well suited for advection dominated problems. Finally, we have designed a Reduced Basis Method that is based on an embedded Finite Element discretization, and that performs very well for problems with a significant change in the shape of the domain. For all the reduction methods proposed here, we have relied on a Proper Orthogonal Decomposition, or on a Proper Orthogonal Decomposition combined with a pseudo Greedy algorithm. When using a classical Finite Element discretization, we have employed an Arbitrary Lagrangian Eulerian formulation, i.e. we have rewritten the coupled problem on a reference configuration. On the contrary, by using an embedded Finite Element discretization, we have avoided the formulation on a reference configuration and we have focused on the original domain. All these results represent an improvement in the state of the art of reduction techniques for coupled problems, in different situations: large deformations of the structure, large transport phenomenon, large dimension of the system to be solved in the online phase.
For what concerns the various techniques that we have used, these are some opinions that the author of this thesis has formulated during the developement of the algorithms proposed: monolithic algorithms are more stable with respect to partitioned procedures. They do not suffer from instabilities due to the physical constants of the fluid and of the solid, and, moreover, they allow for a bigger time step in the simulation. They however require a larger number of unknowns in the system. On the other hand, partitioned approaches are well suited for reducing the dimensionality of the system to be solved in the online phase, but they are sensitive to the properties of the materials under consideration. If the goal is to focus on a particular problem of interest for a long time interval, the author believes that monolithic approaches are more suited, especially because of the improvement that has been proposed in this thesis at the level of reducing the Kolmogorov n–width of the problem.
The incorporation of a snapshot transportation, shift or preprocessing procedure in the offline phase of the RBM represents without any doubt a benefit for the method, especially in combination with other advantageous features of the RBM, such as EIM for example.
The embedded approach is extremely helpful: the author believes that, if the focus is on biomedical applications, then the deformation of the biological tissue under consideration is quite small, so there is no need to use a CutFEM approach in these situations. For other industrial applications, where large deformations occur, using an embedded formulation seems the most natural choice, even though the mathematical formulation of the problem becomes more involved.
Several improvements and several further studies are possible, for the proposed reduced order models: we mention herafter a few of them.

- *Partitioned ROMs*: in this thesis we have proposed a partitioned reduced order model that relies on a Chorin–Temam projection scheme, with semi–implicit coupling between the fluid and the solid problem. The algorithm has been tested for an unsteady problem, where we couple an incompressible fluid with an elastic, two dimensional structure, and for the geometrically parametrized version of the same problem. A next step in the direction of testing partitioned model order reduction techniques is to take into consideration a test case that comprises both a geometrical and a physical parametrization; for example, it would be interesting to study, for the FSI test case that has been considered in this Chapter, the influence of the leaflet length in combination with some physical parameters such as the solid stiffness and the fluid Reynolds number.
  A future developement for this reduction technique is the implementation of the Empirical Interpolation Method (EIM), in order to guarantee an efficient offline–online decoupling for nonlinear problems such as the one considered in this thesis. With the incorporation of the EIM we pursue a further speedup of the online part and we design a partitioned reduction method that can be efficiently applied in situations where we have a non–affine parameter dependence. An investigation of a different, more efficient separation at the reduced order level of the parabolic and the hyperbolic component of the coupled problem is another future step in the developement of partitioned reduced order methods. It would also be beneficial to try to overcome the fact that the Robin coupling condition in the implicit step is heavily time dependent: alternative coupling techniques should be studied and tested, in order to allow a bigger time–step in the simulation.

- *ROMs in the framework of advection dominated problems*: this line of research has become more and more active and productive in the last years, and we are aware that there are several interesting alternatives to the preprocessing technique that we have used, and several aspects in which the work that has been presented in this thesis can be improved. For future perspectives indeed, we do believe that there are several possibilities to further develop this technique, especially in the parametric setting. For example, for the CFD test case with physical parametrization of Section 4.5, it would be an interesting idea to use an Artificial Neural Network (ANN), in order to efficiently explore the parameter space and in order to efficiently learn the deformation parameter $\gamma(t)$ that identifies the map used in the preprocessing procedure; computing $\gamma(t)$ at every time t is not feasible, especially if the deformation map is identified by multiple parameters. While for the FSI problem that we have considered, a linear interpolation was enough, we expect more difficulties for more complex phenomena, that do not behave linearly. To conclude, future research work concerning the results presented in this Chapter includes the efficient evaluation of the online phase, and, in addition, application of this reduced order method coupled with the preprocessing technique in the framework of inverse problems [120, 64, 167, 166, 184].

- *CutFEM ROMs*: all the work that has been presented in Chapter 5 lays the groundwork for the application of ROMs based on CutFEM to Fluid–Structure Interaction problems: we have started to prepare the basis for a CutFEM-RB procedure that, thanks to a Galerkin projection and with the use of a supremizer enrichment, will ideally be used to obtain accurate approximations of solutions of very complex problems, such as fully coupled multiphysics problem, where large displacements occur. Therefore, as future perspectives, we would like to

test the performance of this approach with time dependent Fluid–Structure Interaction problems, geometrically shallow water flows as well as with phase flow Navier-Stokes systems. Furthemore, from the model reduction point of view, we will pursue further developments in hyper-reduction techniques [183, 17, 40, 162] tailored for unfitted discretizations. Another interesting direction of developement is the incorporation of a snapshots transportation technique during the offline phase, in order to obtain a reduction method in the CutFEM framework, that can be applied also to problems that show a slow decay of the Kolmogorov n–width of the solution manifold. A combination of the work presented in Chapter 4 and in this Chapter is thus a possible future line of research.

As a concluding observation we remark that all the examples that have been considered in this thesis are in the low Reynolds number regime, i.e. the fluid has a laminar behaviour. A general perspective is to adapt the reduction procedures presented also for problems and applications where the Reynolds number of the fluid under consideration is high [25]. In the framework of turbulent models, Finite Volumes (FV) and Spectral Element Methods (SEM) represent the most suited and most used discretization methods: examples of already existing reduction procedures based on FV can be found in [163, 67, 66, 71, 162], whereas for SEM we refer to [88, 89].
To conclude, we believe that the future inspection of reduced order models for problems with varying Reynolds number allows to further develop and to extend the work presented in this thesis to general multiphysics problems that are not necessarily given by biomedical applications, such as, for example, aeronautical applications, as well as another class of coupled problems, namely multiphase flows problems.

6.2 Acknowledgements

I acknowledge my supervisor, Professor Gianluigi Rozza, and my co–advisor, Dr. Francesco Ballarin, for the constant help, the endless support that they demonstrated during my journey as a PhD student.
The collaboration with Professor Yvon Maday has been a wonderful motivation for the work presented in Chapter 3 and 4. The collaboration with Dr. Efthymios Karatzas has been fundamental for the developement of the work presented in Chapter 5: to him, my most profound thank you, for the effort put in the developement of the code and the constant support during the writing of that portion of the thesis.
The numerical results presented in this thesis have been obtained with FEniCS (`https://fenicsproject.org/`), for the Finite Element part: FEniCS is an open sourse computational platform based on Python and C++. For the Fluid–Structure Interaction problems, the Python based library multiphenics (`https://github.com/mathLab/multiphenics`) has been used, and for the reduced basis simulations, the Python based library RBniCS (`https://github.com/mathLab/RBniCS`) has been used. Both multiphenics and RBniCS have been developed in the Sissa MathLab group, which I heartily acknowledge.
The numerical results of Chapter 5, in the cut Finite Element part, have been obtained with NGSolve (`https://github.com/NGSolve/ngsolve`), a C++ Finite Element tool; in particular, the NGSolve extension for unfitted discretizations, *ngxfem* (`https://github.com/ngsxfem/ngsxfem`) has been employed.

Appendix A

Robin coupling conditions in a projection scheme

We propose here all the calculation that leads to the formulation of Robin boundary conditions for a projection–based semi–implicit coupling scheme. The final goal is to compute the constant α_{ROB} that improves the stability of the implicit coupling step, for a problem where we couple an incompressible fluid with a thick elastic structure. All the calculations presented here are inspired by the work of [58, 16]. We will proceed by steps: we start with the coupling of the fluid with a thin structure (generalized string equation); in this case, the coupling between the fluid and the solid is given by the nature of the equation describing the motion of the structure. Then we will move to the coupling with a thick elastic structure. The final step will be to recast everything within the ALE framework.

A.1 Coupling of an incompressible fluid with a thin structure

We consider a simplified model, where the fluid is described by the Stokes equation on a *fixed* domain Ω, and the structure, which is assumed to be one dimensional, is described by a Reissner-Mindlin shell model. Let Ω be the fluid domain, and let Σ be the portion of the boundary $\partial\Omega$ that represents the FSI interface. The problem formulation is the following: find $\boldsymbol{u}_f \colon \Omega \mapsto \mathbb{R}^2$, $p \colon \Omega \mapsto \mathbb{R}$ and $\boldsymbol{d}_s \colon \Sigma \mapsto \mathbb{R}^2$ such that:

$$\begin{cases} \rho_f \partial_t \boldsymbol{u}_f - \mathrm{div}\sigma_f(\boldsymbol{u}_f, p_f) = 0 \quad \text{in } \Omega, \\ \mathrm{div}\boldsymbol{u}_f = 0 \quad \text{in } \Omega, \\ \rho_s h_s \partial_{tt} \boldsymbol{d}_s + L(\boldsymbol{d}_s) = -\sigma_f(\boldsymbol{u}_f, p_f)\boldsymbol{n} \quad \text{in } \Sigma, \end{cases} \tag{A.1}$$

where we assume that $L(\cdot)$ is a linear operator for $\boldsymbol{d}_s$. We complete system (A.1) with the coupling condition:

$$\boldsymbol{u}_f = \partial_t \boldsymbol{d}_s \quad \text{on } \Sigma, \tag{A.2}$$

and with some suitable boundary conditions. From equation $(A.1)_3$ and coupling condition (A.2) it follows that:

$$\rho_s h_s \partial_t \boldsymbol{u}_f + \sigma_f(\boldsymbol{u}_f, p_f)\boldsymbol{n} = -L(\boldsymbol{d}_s), \quad \text{on } \Sigma. \tag{A.3}$$

Condition (A.3) can be seen as a Robin condition for the fluid at the interface, and now we want to incorporate it in the fluid projection step of our partitioned procedure. First of all, let us summarize the partitioned procedure hereafter.

Projection-based partitioned scheme

Let Δt denote the timestep for our discretization; for a function f we choose to discretize $D_t f^{i+1} = \frac{f^{i+1}-f^i}{\Delta t}$, and $D_{tt} f^{i+1} = D_t(D_t f^{i+1})$. Our partitioned scheme reads as follows: for $i = 0, \dots, N_T = \frac{T}{\Delta t}$:

Fluid explicit step:

find $\tilde{\boldsymbol{u}}_f^{i+1} \colon \Omega \to \mathbb{R}^2$ such that:

$$\begin{cases} \rho_f \frac{\tilde{\boldsymbol{u}}_f^{i+1} - \boldsymbol{u}_f^i}{\Delta_t} - 2\mu_f \mathrm{div}\varepsilon(\tilde{\boldsymbol{u}}_f^{i+1}) = 0 \quad \text{in } \Omega, \\ \tilde{\boldsymbol{u}}_f^{i+1} = D_t \boldsymbol{d}_s^i \quad \text{on } \Sigma \end{cases} \tag{A.4}$$

Implicit step:

1. **fluid projection substep:** find $p^{i+1} \colon \Omega \to \mathbb{R}$ and $\boldsymbol{u}_f^{i+1} \colon \Omega \to \mathbb{R}^2$ such that:

$$\begin{cases} \rho_f \frac{\boldsymbol{u}_f^{i+1} - \tilde{\boldsymbol{u}}_f^{i+1}}{\Delta_t} + \nabla p^{i+1} = 0 \quad \text{in } \Omega, \\ \mathrm{div}(\boldsymbol{u}_f)^{i+1} = 0 \quad \text{in } \Omega, \end{cases} \tag{A.5}$$

2. **structure projection substep:** find $\boldsymbol{d}_s^{i+1} \colon \Sigma \to \mathbb{R}$ such that:

$$\rho_s h_s D_{tt} \boldsymbol{d}_s^{i+1} + L(\boldsymbol{d}_s^{i+1}) = \sigma_f(\tilde{\boldsymbol{u}}_f^{i+1}, p^{i+1})\boldsymbol{n} \quad \text{on } \Sigma. \tag{A.6}$$

In order to add some coupling conditions to the procedure, first of all we decompose equation (A.3), by observing that the fluid pressure only contributes to the normal component of the fluid stress $\sigma_f(\boldsymbol{u}_f, p)\boldsymbol{n}$. Therefore, just the normal component of the vectorial equation (A.3) is going to contribute to the pressure projection step (A.5). The previous observation motivates the following discretized and "decoupled" conditions:

$$\begin{cases} 2\mu_f \varepsilon(\tilde{\boldsymbol{u}}_f^{i+1})\boldsymbol{n} + \frac{\rho_s h_s}{\Delta t}\tilde{\boldsymbol{u}}_f^{i+1} = \frac{\rho_s h_s}{\Delta t}\dot{\boldsymbol{d}}_s^i - L(\boldsymbol{d}_s^{i+1,\star}) \cdot \tau \\ -p^{i+1} + \frac{\rho_s h_s}{\Delta t}\boldsymbol{u}_f^{i+1} \cdot \boldsymbol{n} = \frac{\rho_s h_s}{\Delta t}\tilde{\boldsymbol{u}}_f^{i+1} \cdot \boldsymbol{n} - L(\boldsymbol{d}_s^{i+1,\star}) \cdot \boldsymbol{n}, \end{cases} \tag{A.7}$$

where τ is the vector tangent to Σ and $\boldsymbol{n}$ is the outward unit normal to Σ. In the previous system, the superscript $\star$ indicates an extrapolation of the quantity of interest. The key point of the previous system is that we have a tangential component of the Robin condition (A.3), that takes care of the viscous part of the fluid and that incorporates the continuity of the velocity at the interface, and we have a normal component of the equation, that incorporates the contribution of the pressure.
Usually the tangential component of the equation (A.3) is used as a Robin boundary condition for the fluid viscous step (fluid explicit step). We re–write the tangential

component of equation (A.3), because we will use it further on to write in a suitable way the Robin boundary condition for the pressure implicit step.

From the structure implicit step (A.6) it follows that:

$$-L(\boldsymbol{d}_s^{i+1,\star}) = \rho_s h_s D_{tt}\boldsymbol{d}_s^{i+1,\star} + \sigma_f(\tilde{\boldsymbol{u}}_f^{i+1,\star}, p^{i+1,\star})\boldsymbol{n}.$$

So:

$$\begin{aligned} -L(\boldsymbol{d}_s^{1+1,\star})\cdot\boldsymbol{n} &= \rho_s h_s D_{tt}\boldsymbol{d}_s^{i+1,\star}\cdot\boldsymbol{n} + \sigma_f(\tilde{\boldsymbol{u}}_f^{i+1,\star}, p^{i+1\star})\boldsymbol{n}\cdot\boldsymbol{n} \\ &= \rho_s h_s D_{tt}\boldsymbol{d}_s^{i+1,\star}\cdot\boldsymbol{n} - p^{i+1,\star} + 2\mu_f\varepsilon(\tilde{\boldsymbol{u}}_f^{i+1,\star})\boldsymbol{n}\cdot\boldsymbol{n}. \end{aligned} \tag{A.8}$$

But now taking the scalar product of $(\text{A.7})_1$ with $\boldsymbol{n}$ we find that:

$$2\mu_f\varepsilon(\tilde{\boldsymbol{u}}_f^{i+1,\star})\boldsymbol{n}\cdot\boldsymbol{n} = \frac{\rho_s h_s}{\Delta t}(\dot{\boldsymbol{d}}_s^{i,\star} - \tilde{\boldsymbol{u}}_f^{i+1,\star})\cdot\boldsymbol{n},$$

so substituting this term in equation (A.8), and re–writing the right hand side of $(\text{A.7})_2$ leads us to:

$$\begin{aligned} -p^{i+1} + \frac{\rho_s h_s}{\Delta t}\boldsymbol{u}_f^{i+1}\cdot\boldsymbol{n} &= \frac{\rho_s h_s}{\Delta t}\tilde{\boldsymbol{u}}_f^{i+1}\cdot\boldsymbol{n} - \frac{\rho_s h_s}{\Delta t}(\dot{\boldsymbol{d}}_s^{i+1,\star} - \dot{\boldsymbol{d}}_s^{i,\star})\cdot\boldsymbol{n} - p^{i+1,\star} + \\ &+ \frac{\rho_s h_s}{\Delta_t}(\dot{\boldsymbol{d}}_s^{i,\star} - \tilde{\boldsymbol{u}}_f^{i+i,\star})\cdot\boldsymbol{n} \\ &= \frac{\rho_s h_s}{\Delta t}\tilde{\boldsymbol{u}}_f^{i+1}\cdot\boldsymbol{n} - p^{i+1,\star} + \rho_s h_s D_{tt}\boldsymbol{d}_s^{i+1,\star}\cdot\boldsymbol{n}. \end{aligned} \tag{A.9}$$

The previous equation is the Robin BC for the pressure projection step in the Darcy formulation: if we want to work with the pressure projection Poisson formulation, we just have to substitute the term $\boldsymbol{u}^{i+1}$ with the term $\boldsymbol{u}^{i+1} = \tilde{\boldsymbol{u}}_f^{i+1} - \frac{\Delta t}{\rho_f}\nabla p^{i+1}$, and we retrieve the famous coefficient $\alpha_{ROB} = \frac{\rho_s h_s}{\rho_f}$ and the Robin BC for the Poisson problem.

A.2 Coupling of an incompressible fluid with a thick structure

In this Section we try to find suitable Robin conditions, in a situation where the structure is no longer considered to be one dimensional, and its behavior is described by a linear elasticity equation. What follows is inspired by the idea proposed in [16]. We first begin with a simple test case: the deformation of the structure is so small, that we can consider the domain as being fixed. Let Ω_f be the fluid domain, Ω_s the solid domain, and Γ_{FSI} the fluid–structure interface.
The fluid problem reads as follows: find $\boldsymbol{u}_f\colon \Omega_f \mapsto \mathbb{R}^2$, $p\colon \Omega_f \mapsto \mathbb{R}$ and $\boldsymbol{d}_s\colon \Omega_s \mapsto \mathbb{R}^2$ such that

$$\begin{cases} \rho_f\partial_t\boldsymbol{u}_f - \mathrm{div}\sigma_f(\boldsymbol{u}_f, p) = 0 \quad \text{in } \Omega_f, \\ \mathrm{div}\boldsymbol{u}_f = 0 \quad \text{in } \Omega_f, \\ \rho_s\partial_{tt}\boldsymbol{u}_s - \mathrm{div}P(\boldsymbol{d}_s) = 0 \quad \text{in } \Omega_s, \end{cases} \tag{A.10}$$

where $P(\boldsymbol{d}_s)$ is defined as

$$P(\boldsymbol{d}_s) = \lambda_s(\nabla\cdot\boldsymbol{d}_s)\boldsymbol{I} + \mu_s\varepsilon(\boldsymbol{d}_s), \qquad \varepsilon(\boldsymbol{d}_s) = (\nabla\boldsymbol{d}_s + \nabla^T\boldsymbol{d}_s).$$

Here $\boldsymbol{I}$ is the 2×2 identity matrix, and λ_s, μ_s are the Lamè coefficients of the solid. The coupling conditions are:

$$\begin{cases} \boldsymbol{u}_f = \partial_t \boldsymbol{d}_s, & \text{on } \Sigma, \\ \sigma_f(\boldsymbol{u}_f, p)\boldsymbol{n} = P(\boldsymbol{d}_s)\boldsymbol{n}, & \text{on } \Sigma. \end{cases}$$

The key ingredient is the observation that the solid problem (A.10) is an hyperbolic PDE, that has two wave velocities:

$$c_p = \sqrt{\frac{\lambda_s + 2\mu_s}{\rho_s}} \qquad c_s = \sqrt{\frac{\mu_s}{\rho_s}}$$

If one writes equation (A.10) in the normal component with respect to Γ_{FSI}, then it is possible to compute the characteristics of the corresponding wave equation, and the characteristic variables: the characteristic variables represent, basically, the information that is carried by the wave, in the normal component with respect to Γ_{FSI}.
These are the characteristic variables that are *outgoing*, i.e. that are transferred from the solid to the fluid:

$$\mathcal{B}(P, \partial_t \boldsymbol{d}_s) = P(\boldsymbol{d}_s)\boldsymbol{n} \cdot \boldsymbol{n} + z_p \partial_t \boldsymbol{d}_s \cdot \boldsymbol{n}, \qquad \mathcal{B}_\tau(P, \partial_t \boldsymbol{d}_s) = P(\boldsymbol{d}_s)\boldsymbol{n} \cdot \tau + z_s \partial_t \boldsymbol{d}_s \cdot \tau,$$

whereas the characteristic variables that are *incoming* to the solid are:

$$\mathcal{A}(P, \partial_t \boldsymbol{d}_s) = P(\boldsymbol{d}_s)\boldsymbol{n} \cdot \boldsymbol{n} - z_p \partial_t \boldsymbol{d}_s \cdot \boldsymbol{n}, \qquad \mathcal{A}_\tau(P, \partial_t \boldsymbol{d}_s) = P(\boldsymbol{d}_s)\boldsymbol{n} \cdot \tau - z_s \partial_t \boldsymbol{d}_s \cdot \tau,$$

where $z_p = \rho_s c_p$ and $z_s = \rho_s c_s$ are the solid impedances, while $\boldsymbol{n}$ and τ are, respectively, the unit normal to Γ_{FSI} pointing outward the fluid domain, and τ is the tangential unit vector to Γ_{FSI}.
The key point, therefore, is to impose the balance of these informations at the interface Γ_{FSI}:

$$\mathcal{B}(\sigma_f^f, \boldsymbol{u}_f) = \mathcal{B}(P(\boldsymbol{d}_s), \partial_t \boldsymbol{d}_s), \qquad \mathcal{B}_\tau(\sigma_f^f, \boldsymbol{u}_f) = \mathcal{B}_\tau(P(\boldsymbol{d}_s), \partial_t \boldsymbol{d}_s) \tag{A.11}$$

and

$$\mathcal{A}(\sigma_f^f, \boldsymbol{u}_f) = \mathcal{A}(P(\boldsymbol{d}_s), \partial_t \boldsymbol{d}_s), \qquad \mathcal{A}_\tau(\sigma_f^f, \boldsymbol{u}_f) = \mathcal{A}_\tau(P(\boldsymbol{d}_s), \partial_t \boldsymbol{d}_s). \tag{A.12}$$

From equation (A.11) it follows in particular that:

$$P(\boldsymbol{d}_s)\boldsymbol{n} \cdot \boldsymbol{n} + z_p \partial_t \boldsymbol{d}_s \cdot \boldsymbol{n} = \sigma_f^f(\boldsymbol{u}_f, p)\boldsymbol{n} \cdot \boldsymbol{n} + z_p \boldsymbol{u}_f \cdot \boldsymbol{n}, \tag{A.13}$$

and keeping the structure of the semi-implicit coupling scheme in mind, we can re–write the discretized in time version of the previous equation as follows:

$$\lambda_s \mathrm{tr}\varepsilon(\boldsymbol{d}_s^{i+1,\star}) + \mu_s \varepsilon(\boldsymbol{d}_s^{i+1,\star})\boldsymbol{n} \cdot \boldsymbol{n} + z_p D_t \boldsymbol{d}_s^{i+1,\star} \cdot \boldsymbol{n} = -p^{i+1} + \mu_f \varepsilon(\tilde{\boldsymbol{u}}_f^{i+1})\boldsymbol{n} \cdot \boldsymbol{n} + z_p \boldsymbol{u}_f^{i+1} \cdot \boldsymbol{n}. \tag{A.14}$$

Now let us remember that we choose to impose strongly the coupling condition:

$$\sigma_f(\tilde{\boldsymbol{u}}_f^{i+1}, p^{i+1})\boldsymbol{n} = P(\boldsymbol{d}_s^{i+1})\boldsymbol{n} \quad \text{on } \Gamma_{FSI},$$

so it follows that:

$$\sigma_f(\tilde{\boldsymbol{u}}_f^{i+1,\star}, p^{i+1,\star})\boldsymbol{n} = P(\boldsymbol{d}_s^{i+1,\star})\boldsymbol{n} \quad \text{on } \Gamma_{FSI},$$

and so we have the following equation, obtained projecting the previous coupling condition along $\boldsymbol{n}$:

$$-p^{i+1,\star} + \mu_f \varepsilon(\tilde{\boldsymbol{u}}_f^{i+1,\star})\boldsymbol{n}\cdot\boldsymbol{n} = \lambda_s \mathrm{tr}\varepsilon(\boldsymbol{d}^{i+1,\star}) + \mu_s \varepsilon(\boldsymbol{d}_s^{i+1,\star})\boldsymbol{n}\cdot\boldsymbol{n}.$$

Now, remembering that $\tilde{\boldsymbol{u}}_f^{i+1,\star}$ is an extrapolation of $\tilde{\boldsymbol{u}}_f^{i+1}$ and that $\tilde{\boldsymbol{u}}_f^{i+1,\star}$ has already been computed at iteration $i+1$, according to the structure of our partitioned algorithm, it follows, by substituting the previous equation into equation (A.14), that:

$$-p^{i+1} + z_p \boldsymbol{u}_f^{i+1}\cdot\boldsymbol{n} = -p^{i+1,\star} + z_p D_t \boldsymbol{d}_s^{i+1,\star}\cdot\boldsymbol{n}.$$

Again, if we adopt a pressure Poisson formulation in our algorithm, then we substitute $\boldsymbol{u}_f^{i+1} = \tilde{\boldsymbol{u}}_f^{i+1} - \frac{\Delta t}{\rho_f}\nabla p^{i+1}$, and we arrive to the final expression of the Robin coupling condition:

$$-\alpha_{rob} p^{i+1} - \nabla p^{i+1}\cdot\boldsymbol{n} = -\alpha_{rob} p^{i+1,\star} + \rho_f D_{tt}\boldsymbol{d}_s^{i+1,\star}\cdot\boldsymbol{n}. \tag{A.15}$$

Thus the expression of the Robin coupling conditions remains unchanged, the only thing that changes is the Robin constant α_{rob}, which now is $\alpha_{rob} = \frac{\rho_f}{z_p \Delta t}$: the coupling constant now depends also on the timestep, unlike in the case of a thin structure. Therefore if we choose a timestep that is too big, α_{rob} goes to zero and thus the algorithm becomes insensitive to the effects of the interaction between fluid and structure.

A.3 Coupling of an incompressible fluid in ALE formulation with a thick walled structure

The final step of these calculations is to formulate the proper Robin boundary conditions in the situation where we are incorporating an Arbitrary Lagrangian Eulerian formulation to the problem description.
Let us call $\Omega_f(t)$ the fluid domain at time t, and let $\hat{\Omega}_f$ be a reference configuration. Let $\mathcal{A}_f(t)\colon \hat{\Omega}_f \mapsto \Omega_f(t)$ be the ALE map that maps the reference configuration to the actual physical domain. Let $\boldsymbol{F}$ be the jacobian of $\mathcal{A}_f$ and let J be its determinant. Let $\boldsymbol{u}_f$, p be the fluid velocity and the fluid pressure, defined on the physical domain $\Omega_f(t)$, and let $\hat{\boldsymbol{u}}_f$ and $\hat{p}$ be their counterparts in the reference configuration, let $\hat{\Omega}_s$ be the solid reference domain (which coincides with the Lagrangian domain for the solid), and let $\hat{\Gamma}_{FSI}$ be the FSI interface in the reference configuration. The coupled FSI problem now reads: find $\hat{\boldsymbol{u}}_f\colon \hat{\Omega}_f \mapsto \mathbb{R}^2$, $\hat{p}\colon \hat{\Omega}_f \mapsto \mathbb{R}$, $\boldsymbol{d}_f\colon \hat{\Omega}_f \mapsto \mathbb{R}^2$ and $\boldsymbol{d}_s\colon \hat{\Omega}_s \mapsto \mathbb{R}^2$ such that:

$$\begin{cases} \rho_f J(\partial_t \hat{\boldsymbol{u}}_f + \boldsymbol{F}^{-1}((\hat{\boldsymbol{u}}_f - \partial_t \boldsymbol{d}_f)\cdot\hat{\nabla}\hat{\boldsymbol{u}}_f) - \mu_f \mathrm{div}(J\hat{\varepsilon}(\hat{\boldsymbol{u}}_f)\boldsymbol{F}^{-T}) + J\boldsymbol{F}^{-T}\hat{\nabla}\hat{p} = 0 \quad \text{in } \hat{\Omega}_f, \\ \mathrm{div}(J\boldsymbol{F}^{-1}\hat{\boldsymbol{u}}_f) = 0 \quad \text{in } \hat{\Omega}_f, \\ \rho_s \partial_{tt}\boldsymbol{d}_s - \mathrm{div}P(\boldsymbol{d}_s) = 0 \quad \text{in } \hat{\Omega}_s, \end{cases} \tag{A.16}$$

together with the coupling condition:

$$J\hat{\sigma}_f(\hat{\boldsymbol{u}}_f, \hat{p})\boldsymbol{F}^{-T}\hat{\boldsymbol{n}} = P(\boldsymbol{d}_s)\hat{\boldsymbol{n}} \quad \text{on } \hat{\Gamma}_{FSI},$$

and with some suitable boundary conditions. Here $\hat{\varepsilon}(\hat{\boldsymbol{u}}_f)$ is defined as:

$$\hat{\varepsilon}(\hat{\boldsymbol{u}}_f) = \hat{\nabla}\boldsymbol{u}_f\boldsymbol{F}^{-1} + \boldsymbol{F}^{-T}\hat{\nabla}^T\boldsymbol{u}_f.$$

Before going any further with the computations, let us just remind the partitioned procedure in the case of ALE formalism. Everything is formulated over the reference configuration and therefore, in order to ease the notation, we now drop the $\hat{}$ notation.

Partitioned algorithm in the nonlinear case

For $i = 0, \dots, N_T = \frac{T}{\Delta t}$:

$$\begin{cases} -\Delta \boldsymbol{d}_f^{i+1} = 0 \quad \text{in } \Omega_f, \\ \boldsymbol{d}_f^{i+1} = \boldsymbol{d}_s^i \quad \text{on } \Gamma_{FSI}. \end{cases} \tag{A.17}$$

Fluid explicit step:

find $\tilde{\boldsymbol{u}}_f^{i+1}$ such that:

$$\begin{cases} \frac{J\rho_f}{\Delta t}(\tilde{\boldsymbol{u}}_f^{i+1} - \boldsymbol{u}_f^i) + J\boldsymbol{F}^{-1}((\tilde{\boldsymbol{u}}_f^{i+1} - D_t\boldsymbol{d}_f^{i+1}) \cdot \nabla \tilde{\boldsymbol{u}}_f^{i+1}) - \mu_f \mathrm{div}(J\hat{\varepsilon}(\tilde{\boldsymbol{u}}_f^{i+1})\boldsymbol{F}^{-T}) = 0 \quad \text{in } \Omega_f, \\ \tilde{\boldsymbol{u}}_f^{i+1} = D_t\boldsymbol{d}_f^{i+1} \quad \text{on } \Gamma_{FSI}, \end{cases} \tag{A.18}$$

Implicit step:

1. **fluid projection substep:** find p^{i+1} and $\boldsymbol{u}_f^{i+1}$ such that:

$$\begin{cases} \rho_f J \frac{\boldsymbol{u}_f^{i+1} - \tilde{\boldsymbol{u}}_f^{i+1}}{\Delta t} + J\boldsymbol{F}^{-T}\nabla p^{i+1} = 0 \quad \text{in } \Omega_f, \\ \mathrm{div}(J\boldsymbol{F}^{-1}\boldsymbol{u}_f^{i+1}) = 0 \quad \text{in } \Omega_f \end{cases} \tag{A.19}$$

2. **structure projection substep:** find $\boldsymbol{d}_s^{i+1}$ such that:

$$\rho_s D_{tt}\boldsymbol{d}_s^{i+1} - \mathrm{div}P(\boldsymbol{d}_s^{i+1}) = 0 \quad \text{in } \Omega_s. \tag{A.20}$$

Now to move from the Darcy formulation (A.19) to the Poisson formulation in ALE coordinates, we make the substitution

$$\boldsymbol{u}_f^{i+1} = \tilde{\boldsymbol{u}}_f^{i+1} - \frac{\Delta t}{\rho_f}\boldsymbol{F}^{-T}\nabla p^{i+1},$$

and obtain:

$$-\frac{\rho_f}{\Delta t}\mathrm{div}(J\boldsymbol{F}^{-1}\tilde{\boldsymbol{u}}_f^{i+1}) = -\mathrm{div}(J\boldsymbol{F}^{-1}\boldsymbol{F}^{-T}\nabla p^{i+1}). \tag{A.21}$$

Now if we write explicitly the fluid stress tensor in the reference configuration, we may notice that now the pressure p contributes just to the normal component of the vector $J\sigma_f(\boldsymbol{u}_f, p)\boldsymbol{F}^{-T}\boldsymbol{n}$; but now the normal component in the reference configuration is $J\boldsymbol{F}^{-T}\boldsymbol{n}$. With this subtle yet very important observation, we can perform a pull–back of the equation (A.15) on the reference configuration, to obtain the following Robin coupling condition in the ALE setting:

$$-\frac{\rho_f}{z_p\Delta t}p^{i+1} - \boldsymbol{F}^{-T}\nabla p^{i+1} \cdot J\boldsymbol{F}^{-T}\boldsymbol{n} = -\frac{\rho_f}{z_p\Delta t}p^{i+1,\star} + \rho_f D_{tt}\boldsymbol{d}_s^{i+1,\star} \cdot J\boldsymbol{F}^{-T}\boldsymbol{n}.$$

Bibliography

[1] R. Abgrall, D. Amsallem, and R. Crisovan. Robust model reduction by L^1-norm minimization and approximation via dictionaries: application to nonlinear hyperbolic problems. *Advanced Modeling and Simulation in Engineering Sciences*, 3(1), 2016.

[2] I. Akhtar, A. H. Nayfeh, and C. J. Ribbens. On the stability and extension of reduced-order Galerkin models in incompressible flows. *Theoretical and Computational Fluid Dynamics*, 23(3):213–237, 2009.

[3] B. O. Almroth, P. Stern, and F. A. Brogan. Automatic choice of global shape functions in structural analysis. *AIAA Journal*, 16(5):525–528, 1978.

[4] D. Amsallem, M. J. Zahr, and C. Farhat. Nonlinear model order reduction based on local reduced-order bases. *International Journal for Numerical Methods in Engineering*, 92(10):891–916, 2012.

[5] M. Astorino, F. Chouly, and M. A. Fernández. Robin based semi-implicit coupling in Fluid–Structure Interaction: stability analysis and numerics. *SIAM Journal on Scientific Computing*, 31(6):4041–4065, 2010.

[6] S. Badia, F. Nobile, and C. Vergara. Fluid–structure partitioned procedures based on Robin transmission conditions. *Journal of Computational Physics*, 227(14):7027 – 7051, 2008.

[7] S. Badia, A. Quaini, and A. Quarteroni. Modular vs. non-modular preconditioners for fluid–structure systems with large added-mass effect. *Computer Methods in Applied Mechanics and Engineering*, 197(49):4216 – 4232, 2008.

[8] S. Badia, A. Quaini, and A. Quarteroni. Splitting methods based on algebraic factorization for Fluid–Structure Interaction. *SIAM Journal on Scientific Computing*, 30(4):1778–1805, 2008.

[9] S. Badia, A. Quaini, and A. Quarteroni. Coupling Biot and Navier–Stokes equations for modelling fluid–poroelastic media interaction. *Journal of Computational Physics*, 228(21):7986 – 8014, 2009.

[10] M. Balajewicz and C. Farhat. Reduction of nonlinear embedded boundary models for problems with evolving interfaces. *Journal of Computational Physics*, 274:489–504, 2014.

[11] F. Ballarin, E. Faggiano, S. Ippolito, Manzoni, A. Quarteroni, G. Rozza, and R. Scrofani. Fast simulations of patient-specific haemodynamics of coronary artery bypass grafts based on a POD–Galerkin method and a vascular shape parametrization. *Journal of Computational Physics*, 315:609–628, 2016.

[12] F. Ballarin, A. Manzoni, A. Quarteroni, and G. Rozza. Supremizer stabilization of POD–Galerkin approximation of parametrized steady incompressible Navier–Stokes equations. *International Journal for Numerical Methods in Engineering*, 102(5):1136–1161, 2015.

[13] F. Ballarin and G. Rozza. POD–Galerkin monolithic reduced order models for parametrized Fluid–Structure Interaction problems. *International Journal for Numerical Methods in Fluids*, 82(12):1010–1034, 2016.

[14] F. Ballarin, G. Rozza, and Y. Maday. Reduced-order semi-implicit schemes for Fluid–Structure Interaction problems. In P. Benner, M. Ohlberger, A. T. Patera, G. Rozza, and K. Urban, editors, *Model Reduction of Parametrized Systems*, volume 17 of *MS&A series*, pages 149–167. Springer, Cham., 2017.

[15] E. Balmès. Parametric families of reduced finite element models. theory and applications. *Mechanical Systems and Signal Processing*, 10(4):381 – 394, 1996.

[16] J. W. Banks, W. D. Henshaw, and D. W. Schwendeman. An analysis of a new stable partitioned algorithm for FSI problems. Part I: incompressible flow and elastic solids. *Journal of Computational Physics*, 269:108 – 137, 2014.

[17] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An Empirical Interpolation Method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

[18] S. Basting, A. Quaini, R. Glowinski, and S. Canic. Comparison of time discretization schemes to simulate the motion of an inextensible beam. In A. Abdulle, S. Deparis, D. Kressner, F. Nobile, and M. Picasso, editors, *Numerical Mathematics and Advanced Applications - ENUMATH 2013*, volume 103 of *Lecture notes in Computational Science and Engeneering*, pages 175–183. Springer, Cham., 2015.

[19] S. Basting, A. Quaini, R. Glowinski, and S. Čanić. On the implementation and benchmarking of an extended ALE Method for FSI problems. In S. Frei, B. Holm, T. Richter, T. Wick, and H. Yang, editors, *Fluid-Structure Interaction: Modeling, Adaptive Discretizations and Solvers*, pages 3–39. 2017.

[20] S. Basting, A. Quaini, S. Čanić, and R. Glowinski. Extended ALE method for Fluid–Structure Interaction problems with large structural displacements. *Journal of Computational Physics*, 331:312 – 336, 2017.

[21] S. Behara and S. Mittal. Flow past a circular cylinder at low Reynolds number: oblique vortex shedding. *Physics of Fluids*, 22(5):054102, 2010.

[22] P. Benner, M. Ohlberger, A. T. Patera, G. Rozza, and K. Urban. *Model Reduction of Parametrized Systems*, volume 17 of *MS&A series*. Springer, 2017.

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

[24] F. Bernard, A. Iollo, and S. Riffaud. Reduced-order model for the BGK equation based on POD and optimal transport. *Journal of Computational Physics*, 373:545–570, 2018.

[25] L. Bertagna, A. Quaini, and A. Veneziani. Deconvolution-based nonlinear filtering for incompressible flows at moderately large Reynolds numbers. *International Journal for Numerical Methods in Fluids*, 81(8):463–488, 2016.

[26] L. Bertagna and A. Veneziani. A model reduction approach for the variational estimation of vascular compliance by solving an inverse Fluid–Structure Interaction problem. *Inverse Problems*, 30(5):055006, 2014.

[27] W. Beyn and V. Thümmler. Freezing solutions of equivariant evolution equations. *SIAM Journal on Applied Dynamical Systems*, 3(2):85–116, 2004.

[28] A. Buffa, Y. Maday, A. T. Patera, C. Prud'homme, and G. Turinici. A priori convergence of the greedy algorithm for the parametrized reduced basis method. *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique*, 46(3):595–603, 2012.

[29] T. Bui-Thanh, M. Damodaran, and K. Willcox. Proper orthogonal decomposition extensions for parametric applications in compressible aerodynamics. In *21st AIAA Applied Aerodynamics Conference, Orlando, Florida*, 2003.

[30] M. Bukač, S. Čanić, R. Glowinski, J. Tambača, and A. Quaini. Fluid–Structure Interaction in blood flow capturing non-zero longitudinal structure displacement. *Journal of Computational Physics*, 235:515 – 541, 2013.

[31] E. Burman. Interior Penalty variational multiscale method for the incompressible Navier–Stokes equation: monitoring artificial dissipation. *Computer Methods in Applied Mechanics and Engineering*, 196(41):4045 – 4058, 2007.

[32] E. Burman, S. Claus, P. Hansbo, M. Larson, and A. Massing. CutFEM: discretizing geometry and partial differential equations. *International Journal for Numerical Methods in Engineering*, 104:472–501, 2014.

[33] E. Burman and M. A. Fernández. Continuous Interior Penalty finite element method for the time-dependent Navier–Stokes equations: space discretization and convergence. *Numerische Mathematik*, 107:39–77, 2007.

[34] E. Burman and P. Hansbo. Fictitious domain methods using cut elements: Iii. a stabilized Nitsche method for Stokes' problem. *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique*, 48(3):859–874, 2014.

[35] N. Cagniart. *A few non linear approaches in model order reduction.* PhD thesis, École Doctorale De Sciences Mathématiques De Paris Centre, 2018.

[36] N. Cagniart, R. Crisovan, Y. Maday, and R. Abgrall. Model order reduction for hyperbolic problems: a new framework. hal:01583224, 2017.

[37] N. Cagniart, Y. Maday, and B. Stamm. Model order reduction for problems with large convection effects. In W. Fitzgibbon, Y. Kuznetsov, P. Neittaanmäki, J. Periaux, and O. Pironneau, editors, *Contributions to Partial Differential Equations and Applications*, volume 47 of *Computational Methods in Applied Sciences*, pages 131–150. Springer, Cham., 2019.

[38] A. Caiazzo, T. Iliescu, V. John, and S. Schyschlowa. A numerical investigation of velocity-pressure reduced order models for incompressible flows. *Journal of Computational Physics*, 259:598–616, 2014.

[39] K. Carlberg. Adaptive h-refinement for reduced-order models. *International Journal for Numerical Methods in Engineering*, 102(5):1192–1210, 2015.

[40] K. Carlberg, C. Farhat, J. Cortial, and D. Amsallem. The GNAT method for nonlinear model reduction: Effective implementation and application to computational fluid dynamics and turbulent flows. *Journal of Computational Physics*, 242:623 – 647, 2013.

[41] P. Causin, J. F. Gerbeau, and F. Nobile. Added-mass effect in the design of partitioned algorithms for fluid–structure problems. *Computer Methods in Applied Mechanics and Engineering*, 194(42):4506 – 4527, 2005.

[42] A. Cesmelioglu, H. Lee, A. Quaini, K. Wang, and S. Y. Yi. Optimization-Based Decoupling Algorithms for a Fluid-Poroelastic System. In S. C. Brenner, editor, *Topics in Numerical Partial Differential Equations and Scientific Computing*, pages 79–98. Springer, New York, NY, 2016.

[43] T. Chacón Rebollo, E. Delgado Ávila, M. Gómez Mármol, F. Ballarin, and G. Rozza. On a certified Smagorinsky reduced basis turbulence model. *SIAM Journal on Numerical Analysis*, 55(6):3047–3067, 2017.

[44] F. Chinesta, A. Huerta, G. Rozza, and K. Willcox. Model reduction methods. In E. Stein, R. de Borst, and T. J. R. Hughes, editors, *Encyclopedia of Computational Mechanics, Second Edition*, pages 1–36. John Wiley & Sons, Ltd., 2017.

[45] F. Chinesta, P. Ladeveze, and E. Cueto. A Short Review on Model Order Reduction Based on Proper Generalized Decomposition. *Archives of Computational Methods in Engineering*, 18(4):395, 2011.

[46] E. A. Christensen., M. Brøns, and J. N. Sørensen. Evaluation of Proper Orthogonal Decomposition–based decomposition techniques applied to parameter-dependent nonturbulent flows. *SIAM Journal on Scientific Computing*, 21(4):1419–1434, 1999.

[47] A. Cohen and R. DeVore. Kolmogorov widths under holomorfic mappings. *IMA Journal of Numerical Analysis*, 36(1):1–12, 2015.

[48] C. M. Colciago and S. Deparis. Reduced Numerical Approximation for Fluid–Structure Interaction problems with applications in haemodynamics. *Frontiers in Applied Mathematics and Statistics*, 4, 2018.

[49] A. Derkevorkian, P. Avery, C. Farhat, J. Rabinovitch, and L. Peterson. Effects of structural parameters on the FSI simulation of supersonic parachute deployments. In *AIAA Aviation 2019 Forum, Dallas, Texas*.

[50] J. Donea, S. Giuliani, and J. P. Halleux. An Arbitrary Lagrangian–Eulerian finite element method for transient dynamic Fluid–Structure Interactions. *Computer Methods in Applied Mechanics and Engineering*, 33(1):689 – 723, 1982.

[51] A. Dumon, C. Allery, and A. Ammar. Proper general decomposition (PGD) for the resolution of Navier–Stokes equations. *Journal of Computational Physics*, 230(4):1387–1407, 2011.

[52] V. Ehrlacher, D. Lombardi, O. Mula, and F. X. Vialard. Nonlinear model reduction on metric spaces. application to one-dimensional conservative pdes in wasserstein spaces. arXiv:1909.06626, 2019.

[53] D. Errate, M. J. Esteban, and Y. Maday. Couplage fluid–structure. Un modele simplifiè en dimension 1. *Comptes rendus de l'Académie des sciences. Série* 1*, Mathématique*, 318:275–281, 1994.

[54] C. Farhat and V. K. Lakshminarayan. An ALE formulation of embedded boundary methods for tracking boundary layers in turbulent Fluid–Structure Interaction problems. *Journal of Computational Physics*, 263:53 – 70, 2014.

[55] C. Farhat, K. G. van der Zee, and P. Geuzaine. Provably second-order time-accurate loosely-coupled solution algorithms for transient nonlinear computational aeroelasticity. *Computer Methods in Applied Mechanics and Engineering*, 195(17):1973 – 2001, 2006.

[56] M. A. Fernández. Incremental displacement-correction schemes for incompressible Fluid–Structure Interaction. *Numerische Mathematik*, 123:21 – 65, 2013.

[57] M. A. Fernández, J. F. Gerbeau, and C. Grandmont. A projection semi-implicit scheme for the coupling of an elastic structure with an incompressible fluid. *International Journal for Numerical Methods in Engineering*, 69(4):794–821, 2007.

[58] M. A. Fernández, M. Landajuela, and M. Vidrascu. Fully decoupled time-marching schemes for incompressible fluid/thin-walled structure interaction. *Journal of Computational Physics*, 297:156 – 181, 2015.

[59] M. A. Fernández, J. Mullaert, and M. Vidrascu. Explicit Robin–Neumann schemes for the coupling of incompressible fluids with thin-walled structures. *Computer Methods in Applied Mechanics and Engineering*, 267:566 – 593, 2013.

[60] L. Fick, Y. Maday, A. T. Patera, and T. Taddei. A stabilized POD model for turbulent flows over a range of Reynolds numbers: Optimal parameter sampling and constrained projection. *Journal of Computational Physics*, 371:214 – 243, 2018.

[61] D. Forti, M. Bukac, A. Quaini, S. Canic, and S. Deparis. A Monolithic Approach to Fluid–Composite Structure Interaction. *Journal of Scientific Computing*, 72(1):396–421, 2017.

[62] R. L. Fox and H. Mlura. An approximate analysis technique for design calculations. *AIAA Journal*, 9(1):177–179, 1971.

[63] B. A. Freno and K. T. Carlberg. Machine-learning error models for approximate solutions to parameterized systems of nonlinear equations. *Computer Methods in Applied Mechanics and Engineering*, 348:250 – 296, 2019.

[64] F. Galarce, J. F. Gerbeau, D. Lombardi, and O. Mula. State estimation with nonlinear reduced models. application to the reconstruction of blood flows with doppler ultrasound images. arXiv:1904.13367, 2019.

[65] M. W. Gee, U. Küttler, and W. A. Wall. Truly monolithic algebraic multigrid for fluid–structure interaction. *International Journal for Numerical Methods in Engineering*, 85(8):987–1016, 2011.

[66] S. Georgaka, G. Stabile, G. Rozza, and M. J. Bluck. Parametric POD-Galerkin Model Order Reduction for unsteady–state heat transfer problems. *Communications in Computational Physics*, 27(1):1–32, 2019.

[67] S. Georgaka, G. Stabile, K. Star, G. Rozza, and M. J. Bluck. A hybrid reduced order method for modelling turbulent heat transfer problems. *Computers & Fluids*, 208:104615, 2020.

[68] J. F. Gerbeau and D. Lombardi. Approximated lax pairs for the reduced order integration of nonlinear evolution equations. *Journal of Computational Physics*, 265:246–269, 2014.

[69] J. F. Gerbeau, D. Lombardi, and E. Schenone. Reduced order model in cardiac electrophysiology with approximated Lax Pairs. *Advances in Computational Mathematics*, 41(5):1103–1130, 2015.

[70] A. Gerner and K. Veroy. Certified Reduced Basis Methods for Parametrized Saddle Point Problems. *SIAM Journal on Scientific Computing*, 34(5):A2812–A2836, 2012.

[71] M. Girfoglio, A. Quaini, and G. Rozza. A Finite Volume approximation of the Navier-Stokes equations with nonlinear filtering stabilization. *Computers & Fluids*, 187:27–45, 2019.

[72] C. Grandmont, V. Guimet, and Y. Maday. Numerical analysis of some decoupling techniques for the approximation of the unsteady Fluid–Structure Interaction. *Mathematical Models and Methods in Applied Sciences*, 11(08):1349–1377, 2001.

[73] C. Grandmont and Y. Maday. Existence de solutions d'un problème de couplage fluide-structure bidimensionnel instationnaire. *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics*, 326(4):525 – 530, 1998.

[74] C. Grandmont and Y. Maday. Nonconforming grids for the simulation of Fluid–Structure Interaction. *Contemporary Mathematics*, 218:262–270, 1998.

[75] C. Grandmont and Y. Maday. Fluid-Structure Interaction: a theoretical point of view. *Revue Européenne des Éléments Finis*, 9(6-7):633–653, 2000.

[76] C. Greif and K. Urban. Decay of the Kolmogorov n-width for wave problems. *Applied Mathematics Letters*, 96:216 – 222, 2019.

[77] M. A. Grepl, Y. Maday, N. C. Nguyen, and A. T. Patera. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: M2AN*, 41(3):575–605, 2007.

[78] J. L. Guermond and L. Quartapelle. Calculation of incompressible viscous flows by an unconditionally stable projection fem. *Journal of Computational Physics*, 132(1):12 – 33, 1997.

[79] J. L. Guermond and L. Quartapelle. On stability and convergence of projection methods based on pressure poisson equation. *International Journal for Numerical Methods in Fluids*, 26(9):1039–1053, 1998.

[80] J. L. Guermond and L. Quartapelle. On the approximation of the unsteady navier–stokes equations by finite element projection methods. *Numerische Mathematik*, 80:207–238, 1998.

[81] M. D. Gunzburger. *Perspectives in flow control and optimization*. SIAM, 2002.

[82] M. D. Gunzburger, J. S. Peterson, and J. N. Shadid. Reduced-order modeling of time-dependent PDEs with multiple parameters in the boundary data.

Computer Methods in Applied Mechanics and Engineering, 196(4):1030 – 1047, 2007.

[83] M. E. Gurtin. *An Introduction to Continuum Mechanics*. Mathematics in Science and Engineering. Elsevier Science, 1982.

[84] B. Haasdonk. Reduced basis methods for parametrized PDEs – a tutorial introduction for stationary and instationary problems. In P. Benner, A. Cohen, M. Ohlberger, and K. Willcox, editors, *Model Reduction and Approximation: theory and algorithms*, Computational science and engineering, chapter 2, pages 65–136. Siam, Philadelphia, 2017.

[85] B. Haasdonk and M. Ohlberger. Reduced basis method for finite volume approximations of parametrized linear evolution equations. *ESAIM: M2AN*, 42(2):277–302, 2008.

[86] B. Haasdonk and M. Ohlberger. Efficient reduced models and a posteriori error estimation for parametrized dynamical systems by offline/online decomposition. *Mathematical and Computer Modelling of Dynamical Systems*, 17(2):145–161, 2011.

[87] D. Hartman and L. K. Mestha. A deep learning framework for model reduction of dynamical systems. pages 1917–1922, 2017.

[88] M. Hess, A. Quaini, and G. Rozza. Reduced Basis Model Order Reduction for Navier-Stokes equations in domains with walls of varying curvature. *International Journal of Computational Fluid Dynamics*, 34(2):119–126, 2020.

[89] M. W. Hess and G. Rozza. A Spectral Element Reduced Basis Method in Parametric CFD. In F. A. Radu, K. Kumar, I. Berre, J. M. Nordbotten, and I. S. Pop, editors, *Numerical Mathematics and Advanced Applications - ENUMATH 2017*, volume 126, pages 693–701. Springer, Cham., 2019.

[90] J. S. Hesthaven, G. Rozza, and B. Stamm. *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*. Springer Briefs in Mathematics. Springer Cham Heidelberg New York Dordrecht London, 2016.

[91] T. Hughes, W. Liu, and T. Zimmermann. Lagrangian–Eulerian finite element formulation for incompressible viscous flows. *Computer Methods in Applied Mechanics and Engineering*, 29(3):329–349, 1981.

[92] T. J. R. Hughes, G. Scovazzi, and L. P. Franca. Multiscale and stabilized methods. In E. Stein, R. de Borst, and H. T. J. R., editors, *Encyclopedia of Computational Mechanics Second Edition*, pages 1–64. John Wiley & Sons, Ltd., 2017.

[93] D. B. P. Huynh, G. Rozza, S. Sen, and A. T. Patera. A successive constraint linear optimization method for lower bounds of parametric coercivity and inf–sup stability constants. *Comptes Rendus Mathematique*, 345(8):473 – 478, 2007.

[94] A. Iollo, S. Lanteri, and J. A. Désidéri. Stability Properties of POD–Galerkin Approximations for the Compressible Navier–Stokes Equations. *Theoretical and Computational Fluid Dynamics*, 13(6):377–396, 2000.

[95] A. Iollo and D. Lombardi. Advection modes by optimal mass transfer. *Phys. Rev. E*, 89:022923, 2014.

[96] J. F. Gerbeau and M. Vidrascu. A quasi-Newton algorithm based on a reduced model for Fluid–Structure Interaction problems in blood flows. *ESAIM: M2AN*, 37(4):631–647, 2003.

[97] M. Juntunen and R. Stenberg. Nitsche's method for general boundary conditions. *Mathematics of Computation*, 58:1353–1374, 2009.

[98] I. Kalashnikova and M. F. Barone. On the stability and convergence of a Galerkin reduced order model (ROM) of compressible flow with solid wall and far-field boundary treatment. *International Journal for Numerical Methods in Engineering*, 83(10):1345–1375, 2010.

[99] S. Kang, H. Choi, and S. Lee. Laminar flow past a rotating circular cylinder. *Physics of Fluids*, 11(11):3312–3321, 1999.

[100] E. Karatzas, M. Nonino, F. Ballarin, and G. Rozza. A Reduced Order Cut Finite Element Method for Fluid–Structure Interaction problems with large deformations. in preparation, 2020.

[101] E. Karatzas, M. Nonino, F. Ballarin, and G. Rozza. A Reduced Order Cut Finite Element Method for geometrically parametrized steady and unsteady Navier–Stokes. submitted, 2020.

[102] E. N. Karatzas, F. Ballarin, and G. Rozza. Projection-based reduced order models for a cut finite element method in parametrized domains. *Computers & Mathematics with Applications*, 79(3):833 – 851, 2020.

[103] E. N. Karatzas and G. Rozza. Reduced Order Modeling and a stable embedded boundary parametrised Cahn-Hilliard phase field system based on cut finite elements. In preperation, 2019.

[104] E. N. Karatzas, G. Stabile, N. Atallah, G. Scovazzi, and G. Rozza. A Reduced Order Approach for the Embedded Shifted Boundary FEM and a heat exchange system on parametrized geometries. In J. Fehr and B. Haasdonk, editors, *IUTAM Symposium on Model Order Reduction of Coupled Systems, Stuttgart, Germany*, volume 36 of *IUTAM Bookseries*, pages 111–125. Springer, Cham., 2018.

[105] E. N. Karatzas, G. Stabile, L. Nouveau, G. Scovazzi, and G. Rozza. A reduced basis approach for PDEs on parametrized geometries based on the shifted boundary finite element method and application to a Stokes flow. *Computer Methods in Applied Mechanics and Engineering*, 347:568 – 587, 2019.

[106] E. N. Karatzas, G. Stabile, L. Nouveau, G. Scovazzi, and G. Rozza. A Reduced Order Shifted Boundary Method for Parametrized Incompressible Navier-Stokes Equations. *Computer Methods in Applied Mechanics and Engineering*, 370:113–273, 2020.

[107] D. J. Knezevic, N. C. Nguyen, and A. T. Patera. Reduced basis approximation and a psoteriori error estimation for the parametrized unsteady boussinesq equations. *Mathematical Models and Methods in Applied Sciences*, 21(07):1415–1442, 2011.

[108] K. Kunisch and S. Volkwein. Galerkin Proper Orthogonal Decomposition methods for a general equation in fluid dynamics. *SIAM Journal on Numerical Analysis*, 40(2):492–515, 2002.

[109] T. Lassila, A. Manzoni, A. Quarteroni, and G. Rozza. A reduced computational and geometrical framework for inverse problems in hemodynamics. *International Journal for Numerical Methods in Biomedical Engineering*, 29(7):741–776, 2013.

[110] T. Lassila, A. Manzoni, A. Quarteroni, and G. Rozza. Model order reduction in fluid dynamics: challenges and perspectives. In A. Quarteroni and G. Rozza, editors, *Reduced Order Methods for Modeling and Computational Reduction*, volume 9 of *MS&A series*, pages 235–273. Springer, Cham., 2014.

[111] T. Lassila, A. Quarteroni, and G. Rozza. A reduced basis model with parametric coupling for Fluid–Structure Interaction problems. *SIAM Journal on Scientific Computing*, 34(2):1187–1213, 2012.

[112] P. Le Tallec and J. Mouro. Fluid structure interaction with large structural displacements. *Computer Methods in Applied Mechanics and Engineering*, 190(24):3039 – 3067, 2001. Advances in Computational Methods for Fluid–Structure Interaction.

[113] K. Lee and K. T. Carlberg. Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *Journal of Computational Physics*, 404:108973, 2020.

[114] T. Lieu, C. Farhat, and M. Lesoinne. Reduced-order fluid/structure modeling of a complete aircraft configuration. *Computer Methods in Applied Mechanics and Engineering*, 195(41):5730 – 5742, 2006.

[115] M. Lombardi, N. Parolini, A. Quarteroni, and G. Rozza. Numerical simulation of sailing boats: dynamics, FSI, and shape optimization. In G. Buttazzo and A. Frediani, editors, *Variational Analysis and Aerospace Engineering: Mathematical Challenges for Aerospace Design*, volume 66 of *Springer Optimization and its Applications*, pages 339–377. Springer, Boston, MA, 2012.

[116] A. E. Løvgren, Y. Maday, and E. M. Rønquist. The Reduced Basis Element Method: offline-online decomposition in the nonconforming, nonaffine case. In J. S. Hesthaven and E. M. Rønquist, editors, *Spectral and High Order Methods for Partial Differential Equations*, volume 76 of *Lecture notes in Computational Science and Engeneering*, pages 247–254. Springer Berlin Heidelberg, 2011.

[117] L. Machiels, Y. Maday, I. B. Oliveira, A. T. Patera, and D. V. Rovas. Output bounds for reduced-basis approximations of symmetric positive definite eigenvalue problems. *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics*, 331(2):153 – 158, 2000.

[118] Y. Maday. Analysis of coupled models for fluid-structure interaction of internal flows. In L. Formaggia, A. Quarteroni, and A. Veneziani, editors, *Cardiovascular Mathematics: Modeling and simulation of the circulatory system*, pages 279–306. Springer Milan, 2009.

[119] Y. Maday and O. Mula. A generalized Empirical Interpolation Method: application of reduced basis techniques to data assimilation. In F. Brezzi, P. Colli Franzone, U. Giannazza, and G. Gilardi, editors, *Analysis and Numerics of Partial Differential Equations*, Springer INdAM Series, pages 221–235. Springer Milan, 2013.

[120] Y. Maday, O. Mula, A. T. Patera, and M. Yano. The generalized empirical interpolation method: stability theory on Hilbert spaces with an application to

the stokes equation. *Computer Methods in Applied Mechanics and Engineering*, 287:310–334, 2015.

[121] Y. Maday, N. C. Nguyen, A. T. Patera, and S. H. Pau. A general multipurpose interpolation procedure: the magic points. *Communications on Pure and Applied Analysis*, 8:383, 2009.

[122] Y. Maday, A. T. Patera, and J. Peraire. A general formulation for a posteriori bounds for output functionals of partial differential equations; application to the eigenvalue problem. *Comptes Rendus de l'Académie des Sciences - Series I - Mathematics*, 328(9):823 – 828, 1999.

[123] Y. Maday, A. T. Patera, and G. Turinici. Global a priori convergence theory for reduced-basis approximations of single-parameter symmetric coercive elliptic partial differential equations. *Comptes Rendus Mathematique*, 335(3):289 – 294, 2002.

[124] A. Manzoni, A. Quarteroni, and G. Rozza. Computational reduction for parametrized pdes: Strategies and applications. *Milan Journal of Mathematics*, 80:283–309, 2012.

[125] A. Mejdi, T. C. Rebollo, and S. Rubino. A cure for instabilities due to advection-dominance in pod solution to advection-diffusion-reaction equations. arXiv: 1907.05614, 2019.

[126] J. M. Melenk. On n-widths for elliptic problems. *Journal of Mathematical Analysis and Applications*, 247:272–289, 2000.

[127] C. Murea and S. Sy. A fast method for solving Fluid–Structure Interaction problems numerically. *International Journal for Numerical Methods in Fluids*, 60:1149 – 1172, 2009.

[128] N. J. Nair and M. Balajewicz. Transported snapshot model order reduction approach for parametric, steady-state fluid flows containing parameter-dependent shocks. *International Journal for Numerical Methods in Engineering*, 117(12):1234–1262, 2019.

[129] N. C. Nguyen, G. Rozza, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for the time-dependent viscous Burgers' equation. *Calcolo*, 46(3):157–185, 2009.

[130] N. C. Nguyen, K. Veroy, and A. T. Patera. Certified real-time solution of parametrized Partial Differential Equations. In S. Yip, editor, *Handbook of Materials Modeling*, pages 1529–1564. Springer Dordrecht, 2005.

[131] J. Nitsche. Über ein variationsprinzip zur lösung von dirichlet-problemen bei verwendung von teilräumen, die keinen randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 36:9–15, 1971.

[132] M. Nonino, F. Ballarin, G. Rozza, and Y. Maday. Overcoming slowly decaying Kolmogorov n-width by transport maps: application to model order reduction of fluid dynamics and Fluid–Structure Interaction problems. arXiv:1911.06598, 2019.

[133] M. Nonino, F. Ballarin, G. Rozza, and Y. Maday. Partitioned reduced basis method for a biomedical Fluid–Structure Interaction problem. In preparation, 2020.

[134] A. K. Noor. Recent advances in reduction methods for nonlinear problems. In *Computational Methods in Nonlinear Structural and Solid Mechanics*, pages 31–44. Pergamon, 1981.

[135] A. K. Noor. On making large nonlinear problems small. *Computer Methods in Applied Mechanics and Engineering*, 34(1):955–985, 1982.

[136] M. Ohlberger and S. Rave. Nonlinear reduced basis approximation of parameterized evolution equations via the method of freezing. *Comptes Rendus Mathematique*, 351:901–906, 2013.

[137] M. Ohlberger and S. Rave. Reduced basis methods: success, limitations and future challenges. *Proceedings of the Conference Algoritmy*, pages 1–12, 2016.

[138] B. Peherstorfer. Model reduction for transport-dominated problems via online adaptive bases and adaptive sampling. arXiv:1812.02094, 2018.

[139] S. Piperno and C. Farhat. Design of efficient partitioned procedures for the transient solution of aeroelastic problems. *Revue Européenne des Éléments Finis*, 9(6-7):655–680, 2000.

[140] C. Prud'homme, D. V. Rovas, K. Veroy, L. Machiels, Y. Maday, A. T. Patera, and G. Turinici. Reliable real-time solution of parametrized partial differential equations: reduced-basis output bound methods. *Journal of Fluids Engineering*, 124(1):70–80, 2002.

[141] A. Quaini. *Algorithms for Fluid–Structure Interaction problems arising in hemodynamics*. PhD thesis, EPFL, Lausanne, 2009.

[142] A. Quarteroni and L. Formaggia. Mathematical modelling and numerical simulation of the cardiovascular system. *Handbook of Numerical Analysis*, 12:3–127, 2004.

[143] A. Quarteroni, A. Manzoni, and F. Negri. *Reduced Basis Methods for Partial Differential Equations*, volume 92 of *La Matematica per il 3+2 book series*. Springer International Publishing, 2016.

[144] A. Quarteroni and G. Rozza. Numerical solution of parametrized Navier–Stokes equations by reduced basis methods. *Numerical Methods for Partial Differential Equations*, 23(4):923–948, 2007.

[145] A. Quarteroni, G. Rozza, L. Dedè, and A. Quaini. Numerical approximation of a control problem for advection-diffusion processes. In F. Ceragioli, A. Dontchev, H. Futura, K. Marti, and L. Pandolfi, editors, *System Modeling and Optimization*, pages 261–273. Springer US, Boston, MA, 2006.

[146] A. Quarteroni, G. Rozza, and A. Manzoni. Certified reduced basis approximation for parametrized partial differential equations and applications. *Journal of Mathematics in Industry*, 1, 2011.

[147] A. Quarteroni, M. Tuveri, and A. Veneziani. Computational vascular fluid dynamics: problems, models and methods. *Computing and Visualization in Science*, 2:163–197, 2000.

[148] J. Rabinovitch, D. Z. Huang, R. Borker, P. Avery, C. Farhat, A. Derkevorkian, and L. Peterson. Towards a validated fsi computational framework for supersonic parachute deployments. In *AIAA Aviation 2019 Forum*.

[149] S. S. Ravindran. A reduced-order approach for optimal control of fluids using Proper Orthogonal Decomposition. *International Journal for Numerical Methods in Fluids*, 34(5):425–448, 2000.

[150] J. Reiss, P. Schulze, J. Sesterhenn, and V. Mehrmann. The Shifted Proper Orthogonal Decomposition: a mode decomposition for multiple transport phenomena. *SIAM Journal on Scientific Computing*, 40(3):A1322–A1344, 2018.

[151] T. Richter. *Fluid–structure Interactions. Model, Analysis and Finite Element*, volume 118 of *Lecture Notes in Computational Science and Engineering*. Springer International Publishing, 2017.

[152] A. Rokhzadi. *IMEX and Semi-Implicit Runge-Kutta Schemes for CFD Simulations*. PhD thesis, Civil Engineering Department, Faculty of Engineering, University of Ottawa, 2018.

[153] C. W. Rowley, I. G. Kevrekidis, J. E. Marsden, and K. Lust. Reduction and reconstruction for self-similar dynamical systems. *Nonlinearity*, 16(4):1257–1275, 2003.

[154] G. Rozza. Reduced-basis methods for elliptic equations in sub-domains with a posteriori error bounds and adaptivity. *Applied Numerical Mathematics*, 55(4):403 – 424, 2005.

[155] G. Rozza, D. B. P. Huynh, and A. Manzoni. Reduced basis approximation and a posteriori error estimation for Stokes flows in parametrized geometries: Roles of the inf-sup stability constants. *Numerische Mathematik*, 125(1):115–152, 2013.

[156] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: Application to transport and continuum mechanics. *Archives of Computational Methods in Engineering*, 15(3):229–275, 2008.

[157] G. Rozza and K. Veroy. On the stability of the reduced basis method for Stokes equations in parametrized domains. *Computer Methods in Applied Mechanics and Engineering*, 196(7):1244–1260, 2007.

[158] S. Sirisup and G. Karniadakis. Stability and accuracy of periodic flow solutions obtained by a POD-penalty method. *Physica D: Nonlinear Phenomena*, 202(3-4):218–237, 2005.

[159] G. Stabile, F. Ballarin, G. Zuccarino, and G. Rozza. A reduced order variational multiscale approach for turbulent flows. *Advances in Computational Mathematics*, 45:2349–2368, 2019.

[160] G. Stabile, S. Hijazi, A. Mola, S. Lorenzi, and G. Rozza. POD-Galerkin reduced order methods for CFD using Finite Volume Discretisation: vortex shedding around a circular cylinder. *Communications in Applied and Industrial Mathematics*, 8(1):210 – 236, 2017.

[161] G. Stabile and G. Rozza. Finite volume POD-Galerkin stabilised reduced order methods for the parametrised incompressible Navier-Stokes equations. *Computers & Fluids*, 173:273–284, 2018.

[162] G. Stabile, M. Zancanaro, and G. Rozza. Efficient geometrical parametrization for finite-volume-based reduced order methods. *International Journal for Numerical Methods in Engineering*, 121(12):2655–2682, 2020.

[163] K. Star, G. Stabile, G. Rozza, and J. Degroote. A POD–Galerkin reduced order model of a turbulent convective buoyant flow of sodium over a backward-facing step. https://arxiv.org/abs/2003.01114, 2020.

[164] D. Stojković, M. Breuer, and F. Durst. Effect of high rotation rates on the laminar flow around a circular cylinder. *Physics of Fluids*, 14(9):3160–3178, 2002.

[165] D. Stojković, P. Schön, M. Breuer, and F. Durst. On the new vortex shedding mode past a rotating circular cylinder. *Physics of Fluids*, 15(5):1257–1260, 2003.

[166] M. Strazzullo, F. Ballarin, R. Mosetti, and G. Rozza. Model reduction for parametrized optimal control problems in environmental marine sciences and engineering. *SIAM Journal on Scientific Computing*, 40(4):B1055–B1079, 2018.

[167] M. Strazzullo, F. Ballarin, and G. Rozza. POD–Galerkin model order reduction for parametrized time dependent linear quadratic optimal control problems in saddle point formulation. *SIAM Journal on Scientific Computing*, 83(55), 2020.

[168] S. Sy and C. Murea. Algorithm for solving Fluid–Structure Interaction problem on a global moving mesh. *Coupled systems mechanics*, 1(1):99–113, 2012.

[169] T. Taddei. A registration method for model order reduction: data compression and geometry reduction. *SIAM Journal on Scientific Computing*, 42(2):A997–A1027, 2020.

[170] A. R. Teymourtash and S. E. Salimipour. Compressibility effects on the flow past a rotating cylinder. *Physics of Fluids*, 29(1):016101, 2017.

[171] D. Torlo. Model reduction for advection dominated hyperbolic problems in an ALE framework: offline and online phases. https://arxiv.org/abs/2003.13735, 2020.

[172] D. Torlo, F. Ballarin, and G. Rozza. Stabilized weighted reduced basis methods for parametrized advection dominated problems with random inputs. *SIAM/ASA Journal on Uncertainty Quantification*, 6(4):1475–1502, 2018.

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

[174] K. Veroy, C. Prud'homme, D. Rovas, and A. T. Patera. A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. In *16th AIAA Computational Fluid Dynamics Conference, Orlando, Florida*, 2003.

[175] W. Q. Wang and Y. Yan. Strongly coupling of partitioned fluid–solid interaction solvers using reduced-order models. *Applied Mathematical Modelling*, 34(12):3817 – 3830, 2010.

[176] Y. Wang, A. Quaini, and S. Čanić. A higher-order discontinuous galerkin/arbitrary lagrangian eulerian partitioned approach to solving

fluid–structure interaction problems with incompressible, viscous fluids and elastic structures. *Journal of Scientific Computing*, 76:481–520, 2018.

[177] Y. Wang, A. Quaini, S. Čanić, M. Vukicevic, and S. H. Little. 3d experimental and computational analysis of eccentric mitral regurgitant jets in a mock imaging heart chamber. *Cardiovascular Engineering and Technology*, 8(4):419–438, 2017.

[178] G. Welper. h and hp-adaptive interpolation by transformed snapshots for parametric and stochastic hyperbolic pdes. arXiv:1710.11481, 2017.

[179] G. Welper. Interpolation of functions with parameter dependent jumps by transformed snapshots. *SIAM Journal on Scientific Computing*, 39(4):A1225–A1250, 2017.

[180] G. Welper. Transformed snapshot interpolation with high resolution transforms. *SIAM Journal on Scientific Computing*, 42(4):A2037–A2061, 2020.

[181] K. Willcox and J. Peraire. Balanced model reduction via the Proper Orthogonal Decomposition. *AIAA Journal*, 40(11):2323–2330, 2002.

[182] Y. Wu and X. C. Cai. A fully implicit domain decomposition based ALE framework for three-dimensional Fluid–Structure Interaction with application in blood flow computation. *Journal of Computational Physics*, 258:524 – 537, 2014.

[183] D. Xiao, F. Fang, A. G. Buchan, C. C. Pain, I. M. Navon, J. Du, and G. Hu. Non linear model reduction for the Navier–Stokes equations using residual DEIM method. *Journal of Computational Physics*, 263:1–18, 2014.

[184] Z. Zainib, F. Ballarin, S. Fremes, P. Triverio, L. Jimeńez-Juan, and G. Rozza. Reduced order methods for parametric optimal flow control in coronary bypass grafts, towards patient–specific data assimilation. *International Journal for Numerical Methods in Biomedical Engineering*, page e3367, 2020.