Inverse problems in hemodynamics. Fast estimation of blood flows from medical data

Felipe Galarce Marin

Inverse problems in Hemodynamics. Fast estimation of blood flows from medical data.

THÈSE DE DOCTORAT

Présentée par

Felipe Galarce

pour obtenir le grade de

DOCTEUR DE
SORBONNE UNIVERSITÉ

Spécialité : Mathématiques Appliquées

Soutenue publiquement le 09/04/2021 devant le jury composé de :

Jean-Frédéric Gerbeau	Directeur de thèse
Damiano Lombardi	Directeur de thèse
Olga Mula	Directrice de thèse
Yvon Maday	Président du jury
Christian Vergara	Rapporteur
Marcela Szopos	Rapporteur
Alfonso Caiazzo	Examinateur
Stéphanie Salmon	Examinateur

Thèse préparée au sein de l'équipe-projet COMMEDIA
Laboratoire Jacques-Louis Lions
Sorbonne Université
et **Centre de Recherche Inria de Paris**
2 rue Simone Iff
75589 Paris Cedex 12

Acknowledgments

I would like to express the deepest and most sincere gratitude to Damiano Lombardi. Such a scientific mind and devotion to knowledge are enough reasons to justify my migration to Paris. The exact same gratitude I show to Olga Mula and Jean-Frédéric Gerbeau.

Additionally, I convey my gratitude to the members of the jury. To Yvon Maday, the president of the jury. To both Christian Vergara and Marcela Szopos for their time reading my thesis draft and for their pertinent and useful remarks. To Alfonso Caiazzo and Stéphanie Salmon as well for assisting to the defense and triggering interesting discussions.

It has been a pleasure to write my thesis inside the environment generated by the COMMEDIA team at INRIA Paris. The relation with each team member has been special, either in the coffee break dynamics or with those I can call friends.

I allow myself as well to thank my family and friends for their emotional support.

Abstract: This thesis presents a work at the interface between applied mathematics and biomedical engineering. The work's main subject is the estimation of blood flows and quantities of medical interest in diagnosing certain diseases concerning the cardiovascular system. We propose a complete pipeline, providing the theoretical foundations for state estimation from medical data using reduced-order models, and addressing inter-patient variability. Extensive numerical tests are shown in realistic 3D scenarios that verify the potential impact of the work in the medical comunnity.

Keywords: Inverse problems, State estimation, Quantities of Interest, Non-parametric domains, Inter-patient variability, Model reduction.

Résumé:

Cette thèse présente un travail à l'interface entre les mathématiques appliquées et l'ingénierie biomedicale. Le sujet principal en est l'estimation des écoulements sanguins et de quantités d'intérêt pour le diagnostic de certaines maladies cardiovasculaires. Nous proposons une procédure complète, dont nous détaillons les fondements théoriques, permettant l'estimation d'état à partir de données médicales en utilisant des techniques de réduction de modèle, et en prenant en compte la problématique de la variabilité inter-patients. De nombreux test numériques en 3D sont exposés afin de vérifier le potentiel de cette étude dans le domaine médical.

Mots-clés : Problèmes inverses, Estimation d'état, Quantités d'intérêt, Domaines non paramétriques, Variabilité inter-patient, Réduction de modèle.

Contents

Chapter 1

Introduction

This thesis is about inverse problems in bio-medical engineering. We aim to design tools for the enrichment of medical data using PDE models from classical mechanics and model reduction techniques. We mainly address the problem of, given ultrasound 2D Doppler data, to reconstruct three dimensional velocity fields and quantities of interest (QoI), such as pressure drops and wall shear stress, all of them states of uttermost importance for the diagnosis of a number of pathologies. Although it is not the unique technique implemented, we work in this thesis mainly with the parameterized background data weak approach (PBDW) [Maday et al., 2015b]. Last but not least, a problem about non linear model reduction on non-parametric domains is addressed with promising numerical results towards the deploy of reconstruction techniquess in the clinical practice.

Chapters 2 and 3 share a common goal: the reconstruction of blood flows and QoI in the common carotid arteries from Doppler synthetic data. In chapter 2 we explore PBDW variations using a battery of approaches to build up reduced models at offline phase. Data-driven non linear methods, partitioned POD, and classical greedy approaches are some of those variations. An extensive benchmark is presented. Chapter 3 takes the best among the reconstruction methods presented in chapter 2 in order to give a step towards the application, where QoI such as the velocity vorticity and pressure fields are estimated for synthetically generated noisy velocity data. Those QoIs are typically related to several diseases in the medical community. We claim that this chapters are in a sense complete because, on the one hand, numerical experiments are provided over a three dimensional geometry segmentated from magnetic resonance images, and on the other hand, theoretical results are shown, that coherently background the numerical outcomes.

Chapter 4 is devoted solely to the topic of state estimation in non parametric domains. The challenging problematic of estimating a state into an *a priori* unknown working domain via offline pre-built reduced models is addressed. This leads to the presentation of a systematic methodology that involves the consistent mapping between divergence free spaces, the introduction of model-based metrics, and the use of machine learning techniques to map geometries in

low dimensional representations, or embeddings, that in some manner reproduces a parameter space (thought the geometry does not need to be parameterized).

In chapter 5 we show a detailed description of the software MAD, developed specially for this thesis, which is competitive in terms of performance and scope, providing tools for, among other matters, the approximation of forward fluid dynamics problems solutions with finite elements in parallel, linear and non linear model order reduction techniques, such as greedy methods, principal component analysis and non Euclidean multi-dimensional scaling, and geometry registrations via model induced metrics, LDDMM and Piola transforms.

This introduction gives a general framework of the work, starting in next section by introducing the medical context that encompasses the applications which are shown in all the thesis chapters. A short overview of inverse problems is shown and a discussion on state estimation is added in order to locate PBDW between other typical methodologies. This leads naturally to a discussion about the nature of the data we want to use for our inverse problem. To this end, a brief section on ultrasound echography is added in order to justify the models which are used for the measures.

1.1 Medical framework

The initiative of using tools from scientific computing in the medical practice has been in the air since decades. There are two main ingredients that have converged during the second half of the XX century to make this possible. First, the accelerated capabilities of computers, typically measured in the ability to manufacture computing central processing units (CPU) that are smaller as time goes by, and second, the development of a strong mathematical background coming from, among other fields: numerical analysis, approximation theory, numerical linear algebra, partial differential equations, statistics and model reduction. Both ingredients combined allow us to simulate complex systems, implement artificial intelligence tools and assimilate data in order to deliver predictions on states or quantities of interest that are useful in the diagnosis, treatment and follow-up of several diseases of importance for the medical doctor's community.

The cardiovascular tree is a representative example of a complex system by excellence. The interaction of several phenomena, the intrinsic morphological difficulties, and the numerous restrictions for data acquisition make the study of the circulation a quite challenging task. The field dedicated to the mathematical modeling and numerical simulation of blood flows is called haemodynamics. The word *in silico*[1] is adopted wherever computational experiments takes place in the medical practice, in contrast with the much older Latin expressions *in vitro* and *in vivo*. The ambition of the community, materialized for instance in big projects such as the Virtual

[1] likely coming from *in silicium*, referring to the main element used to manufacture transistors.

Physiological Human institute (VPHi) based at Belgium, aims to bring terms like *digital twins* close to the daily basis medical practice.

First and foremost, the reason behind the attention that the research in this field drags, is based on its direct impact on the society welfare. According to the *Global Health Estimates 2016: Deaths by cause, Age, Sex, by Country and by Region, 2000-2016*, published in 2018 by the World Health Organisation, a report with desegregated data about cause of death in humans in the developed world, cardiovascular malfunction is the first death cause in the planet, as shown in Figure 1.1. It is thus visible that any advance towards the understanding of the cardiovascular function could heavily impact those statistics. The proposal from the scientific computing community is driven by the heuristic that, given the current state of the art in medical diagnosis, the incorporation of mechanistic principles can contribute to enhance the understanding of the underlying physiology. This heuristic, proven astonishingly successful in fields like physics, has a delimited perimeter in other disciplines, such as economy and other social sciences, in which a blinded mechanistic approach could lead to miserable failures, so that we can not be dogmatic, we have to be modest, and we have thus to apply this principle carefully. In addition, statistical tools such as artificial intelligence and machine learning are required to process and assimilate data.

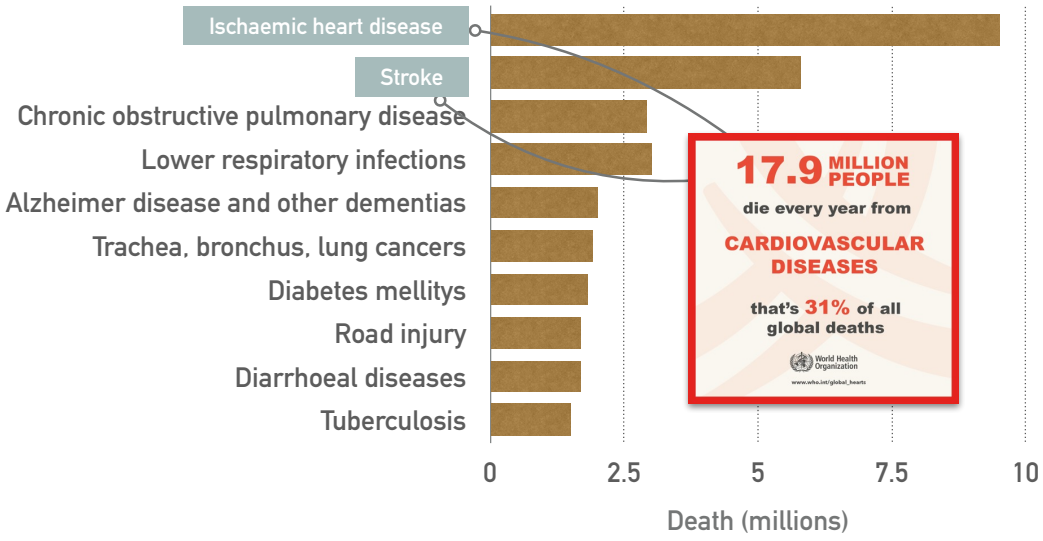


Figure 1.1: Death cause of humans in millions. Cardiovascular malfunctions leads the list averaging 17.9 millions deaths per year, around 31% of all global deaths.

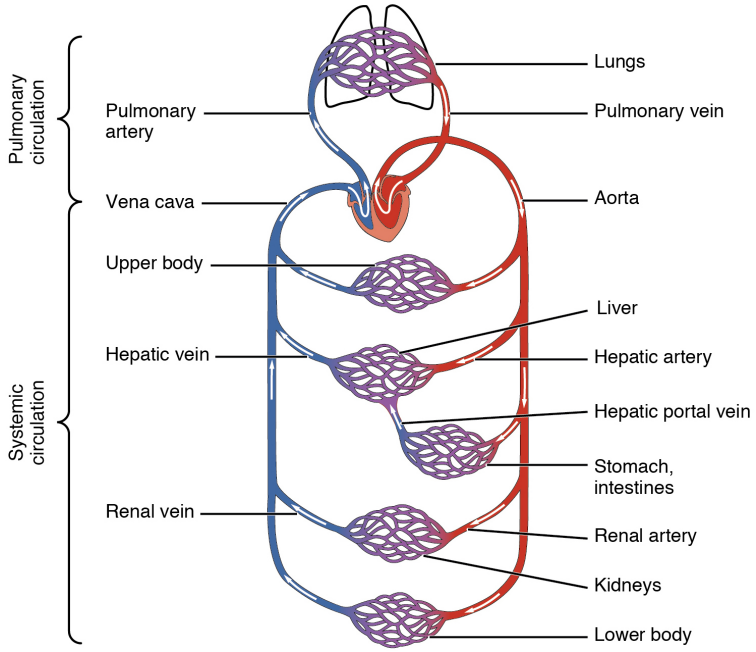


Figure 1.2: Scheme of the cardiovascular tree. Red stands for oxygenated blood, blue stands for de-oxygenated blood, and we see purple in regions where exchange of fluid, nutrients, electrolytes, hormones and other substances is done.

1.1.1 The cardiovascular tree

Let us see a scheme of the human cardiovascular tree[2] in Figure 1.2. The blood circulation is typically categorized as either systemic or pulmonary. The former concerns the blood delivery and recovery to all the body tissues, whereas the latter concerns the interaction with the lungs in order to re-oxygenate the blood.

In the pulmonary circulation, the heart pumps blood to the lungs. There, the blood has to be first detached from the carbon dioxide and hydrogen protons that it carries, and second, the blood is oxygenated interacting with the pulmonary system. At the meso-scale, small cells of biconcave shape, so called red blood cells (RBCs), are in charge of the transport of chemicals. RBCs can carry carbon dioxide, hydrogen protons and oxygen, thanks to a protein called hemoglobin that is capable of bind them to itself. This chemical mechanism is at the core of the system capability to keep the overall homeostasis. The hemoglobin affinity with hydrogen protons is fundamental for the oxygen diffusion, since the blood pH plays a role in the saturation threshold for the oxygen in the protein. Finally, the blood, which is saturated with oxygen, is transported back to the heart so that it can be delivered to all the body tissues.

[2] A rigorous medical discussion about the cardiovascular system can be found in [Guyton and Hall, 2011]

Systemic circulation starts as the heart pumps blood from the left ventricle to the aorta, bifurcating to the lower and upper parts of the body. The blood pass through by the arterial system going from arteries, to arterioles, and then to capillaries, where the diffusion of oxygen, hormones, electrolytes and other substances is done. The pressure potential imposed by the heart pumping is almost decreased to zero close to the diffusion zone. In fact, for a person in resting conditions (lying in a bed) the velocity averages about 33 cm/s in the aorta, and about 0.3 mm/s in the capillaries. In the diffusion zone, the hemoglobin takes the residuals to be transported back to the heart via the venous system, so that another cycle of the pulmonary circulation can re-start.

The whole cycle comprehending pulmonary and systemic circulation occurs around 72 times per minute for a person in resting conditions. About 5 L/min is the average flow that the circulatory system holds in such conditions, but it can adapt largely for stressing scenarios that require quicker muscle oxygenation by augmenting the overall flow up to 30 times the resting value.

1.1.2 The carotid arteries

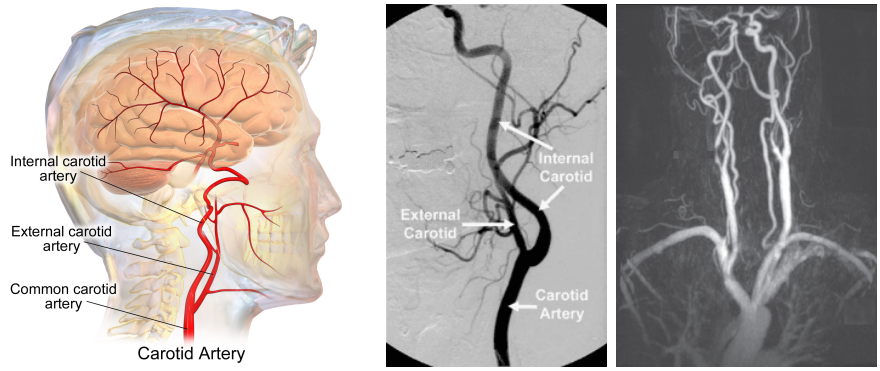
The heart pumps blood to the aorta. The aortic arc bifurcates to, among others, two carotid arteries, which are shown schematically and from medical data in Figure 1.3. In chapter 2 and chapter 3 the carotid artery will be our working domain to solve inverse problems. For a person in resting conditions without morphological nor physiological pathologies, the average velocity in these vessels is about 30 cm/s. Every human has two carotids, and each one of them bifurcates to irrigate the neck and to irrigate zones close to the brain, via the internal carotids or carotid siphons.

A number of pathologies can be understood by studying the flow mechanics in the aforementioned vessels. Instead of following a typical clinical exposition in which the tendency is to show a battery of diseases correlating in numerous ways with certain clinical indexes in ways that are usually independent among them, we prefer to dedicate a section to some malfunctions of the carotid and the cardiovascular system in general in the next section, keeping in mind the already brought up heuristic that classical mechanics could provide up to some extent a theoretical unifying framework.

1.1.3 Velocity and quantities of interest in diagnosis

Vessel coarctation is among the more studied pathologies in haomedynamics. Let us depict in Figure 1.4 the angiography of a case in which the carotid coarctation (or stenosis), is critical[3].

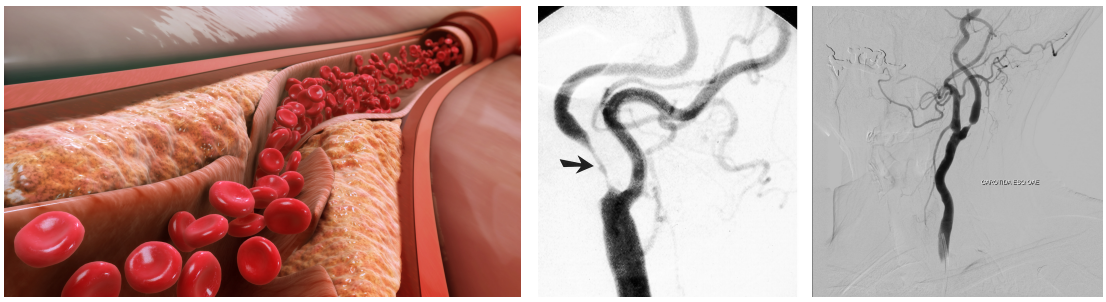
[3] Angiograms in this Figure are courtesy of Dr. Bruno Di Muzio, Radiopaedia.org, rID: 31740



(a) The carotids (b) Angiograms of the carotid arteries

Figure 1.3: The common carotid arteries branches and their bifurcations to the brain.

This problem is not exclusive of the carotid arteries, as it is often present in the aorta and other vessels as well. Among the reasons behind this issue there is, for example, a disease called atherosclerosis, in which an accumulation of lipids and other materials in the vessel walls changes the effective radius of the vessels preventing the normal transit of oxygenated blood to the tissues. A critical example scenario in which this pathology becomes a matter of life and death is the sickle cell disease in kids. As it was stated in previous section, the circulation relies at the cellular level on the affinity that hemoglobin has with oxygen and other chemicals. The perfect storm could come here in a patient with sickle cell disease, i.e., a patient with dysfunctional and deformed red blood cells, which has also an arterial coarctation, and that is also a kid, i.e., a patient whose vessels are of small diameter.



(a) Scheme of vessel coarctation (b) Angiograms of coarcted carotids

Figure 1.4: Pathological carotid arteries.

Either if it is the unlucky case from above or it is one of the desegregated possibilities, the thing is that from the practical point of view it is necessary to assess the patient risk of tissue de-oxygenation, over-fatigue of the vessel walls, risk of cardiovascular accident, need of blood transfusions, etc. On aorta coarctations, for instance, the stenosis severity is assessed by

measuring pressure jumps between certain regions of interest, by means of inserting a catheter inside the vessel that both perturbs the natural flow and second, is incredibly invasive.

It is easy to see as well that all of the narrated phenomena will manifest not only in pressure jumps, but also in the blood velocity and its spatial derivatives. A huge amount of literature correlating cardiovascular diseases with the wall shear stress, vorticity, peak velocities and other derived quantities, supports this statement, e.g., [Konstantinos et al., 2009, Heo et al., 2014, Zarins et al., 1983, Bluestein et al., 2010, Gibson et al., 1993]. As a consequence, another huge amount of literature is devoted to estimate those quantities of interest from several types of medical data, e.g., [Garcia et al., 2013, Hatle et al., 1978, Hatle et al., 1980]. It is important then to provide tools to understand the natural interrelationships that are well known to exists among all the enlisted quantities.

For this thesis, part of the goal is to build up methods to solve inverse problems in haemodynamics, so that state estimations of velocity field and other derived quantities can be assessed fast from medical data, and of uttermost importance, in a non-invasive way. The development of a fast CFD solver that exploits high performance computing capabilities is thus a fundamental middle step to be done. Last but not least, the goal of this thesis is to tackle the crucial hot problem of domain variability, which is of importance in many fields but in particular in bio-medical engineering, as patients, even though sharing a lot in common, have unique vessel structures. As we will see in chapter 4, this problem is what prevents fast reconstruction methods to be clinically useful. A technique to tackle this in a systematic way is explored and tested with admittedly simple in silico experiments, but with successful results.

1.2 Overview of inverse problems and data assimilation

Inverse problems are about inferring a disease by looking at medical data, or (just to make justice to the typical way to introduce this topics in books) about being able to hear the shape of a drum membrane, or about solving the riddle of a crime scene, etc. The term is developed in contrast to forward problem, which can be defined as a way to approximate a quantitative output of a system given the value of the parameters and the data, and exploiting a relationship that quantitatively describe the causality which is supposed to take place. In inverse problems we infer the data and parameters that likely produced a measured output. We refer to [Kaipio and Somersalo, 2005, Stuart, 2010, Law et al., 2015, Dashti and Stuart, 2017] for surveys and detailed texts on inverse problems.

The impact of bringing tools from the inverse problems theory to the clinical applications is remarkably big, even compared to other fields in which many contributions are done, such as mechanical engineering and soil mechanics. The reason behind this arises easily: observing

a state or some QoI in a human being is hugely restricted. Medical imaging brings a solution that allows physicians to diagnose. Nevertheless, although powerful, the medical data is often incomplete, noisy, and hard to interpret. Inverse problems comes here naturally to the table in order to enrich the data and strengthen the medical decision making.

The idea of *non invasive* diagnosis is here at the center of the discussion. A very simple example can shed light on this. Let us thing about the measurement of pressure drops in human aortas, necessary in the clinical practice to assess the severity of vessel coarctations (or stenosis) in patients that suffers, for instance, atherosclerosis. The tradition proposes to compute this QoI by introducing a catheter in the vessel, that first and foremost, requires an incredibly invasive procedure, and second, the mere act of observing perturbs the acquired data. Thus, a non-invasive solution to this would be looked with good eyes in the medical community, and by non-invasive here we are immediately driven to inverse problems, with which we claim we can, among other things, to estimate QoIs by only looking at medical data, that is to say, without the need of even to touch the patient.

1.2.1 State estimation

Let us illustrate some fundamentals about inverse problems with a typical introduction in finite dimension for the estimation of a *state* (a function in this case) $u \in V$, with V a N-dimensional Hilbert space, with inner product $\langle \cdot, \cdot \rangle$ and norm $\| \cdot \|$. We aim to find an *algorithm* (a function) $A : W \to V$ that maps observations $\omega \in W$, an $m-$dimensional space, to a function in V.

Definition 1. Let X be an $\mathcal{N}$-dimensional Hilbert space, with inner product $\langle \cdot, \cdot \rangle$ and norm $\| \cdot \|$, Let $\rho_1, \ldots, \rho_{\mathcal{N}}$ be an orthonormal basis of X. We denote the orthogonal projection of some $y \in Y$, such that $X \subset Y$, *into* X, by:

$$P_X(y) \stackrel{\text{def}}{=} \sum_{i=1}^{\mathcal{N}} \langle y, \rho_i \rangle \rho_i = \inf_{x \in X} \|x - y\|. \tag{1.1}$$

We can interpret u as the state describing a complex system, such as deformation fields in structural engineering, a velocity field, etc. W, typically refereed to as space of observations, can be interpreted as a space generated with m sensors of the state.

Let us denote $\hat{u} = (\hat{u}_1, \ldots, \hat{u}_N)$ the vector of coefficients of u in the basis $\rho_1, \ldots, \rho_N$, i.e., following the definition 1.1 we have $u = \sum_i \hat{u}_i \rho_i$. So, say that we are given m *observations* (or measures) in a vector $l \in \mathbb{R}^m$, which is, analogously to $\hat{u}$, a vector of coefficients in a not yet defined orthonormal basis for W. Assume that we can model the action of observing the state u with a linear operator $O : V \to W$, which admits a matrix representation $\mathbb{A}^{m \times N}$. The key of inverse problems is to study the equation

$$\mathbb{A}u = l. \tag{1.2}$$

We can immediately discard the utopia in which $m = N$ and $\mathbb{A}$ admits an inverse (otherwise the field of inverse problems would end here). We have typically m much smaller than N. Having said so, we can pose the problem in terms of a least-squares fitting, so we look for

$$u^* = \arg\min_{v \in \mathbb{R}^N} \frac{1}{2} \|\mathbb{A}v - l\|^2$$

whose Euler-Lagrange optimality conditions leads to the normal equations

$$\mathbb{A}^T \mathbb{A} u^* = \mathbb{A}^T l. \tag{1.3}$$

Naturally, $\mathbb{A}^T\mathbb{A}$ does not admit an inverse. The problem is not well-posed by construction (we assume $N > m$). We are therefore in the need for a method able to enforce some regularity in the equations so that we get a solution out of this. There are certainly many other ways of addressing the regularization of least-squares problems. Among them we can count regularization via truncated conjugate gradients (or in general by truncated Krylov methods) and Tikhonov regularization [Tikhonov, 1943]. Those are not exposed here for the sake of brevity and because they are not used during the thesis, with the exception of Krylov sub-space methods, but as iterative solvers (which is what they were conceived for) for discretized PDEs, not as regularizers. The discussion is finished in section 1.2.4.2 after introducing the singular value decomposition.

The least-squares (LSQ) approach (1.3) is a fitting that takes only into account the measures information, is a linear regression. Thought very illustrative, this approach is usually not effective. In the next section we introduce how to account for the knowledge that comes from governing laws (arising, for instance, from classical mechanics) in the pipeline. Before doing so, we have to remark that there are indeed methods that are capable to perform state estimation solely relying on data, so called purely *data-driven* approaches. We can mention two: one is of course machine-learning/deep learning/auto-encoder approaches, which are non-linear layered regressions of the data. Another possibility is to perform for instance a Kriging approach, in which the problem is written as $\min_{v \in \omega \oplus W^\perp} \|v\|^2_{\text{Kring}}$, i.e., we do an energy minimization that enforces the estimated state to satisfy the measures. One could argue that there are some sort of physics behind this method in the sense that big part of the effectiveness of this approach relies on a good choice for the metric $\|\cdot\|_{\text{Kring}}$, nevertheless, we will try to set a (illusory) line between physical and non physical state estimations. The former will be characterized by the use of mechanistic equations to describe phenomena. The line is illusory in the sense that, regardless that *laws* of physics are perceived sometimes as a religion, they were comprehended by their authors either by experimentation, or by a huge philosophical perspective, or by using logical relationships starting from a base ground of axioms that comes from intuition and community agreement as well. As a critic, the term *data-driven* physics should induce a huge shared public opinion on the scientific community about the fact that what they are reading when looking at

this phrase is nothing but a pleonasm, a redundancy. Physical models, sciences and laws are about data, not about anything else.

1.2.2 Governing laws and PBDW

This section introduces the main state estimator used in this thesis document: the Parameterized Background Data Weak (PBDW) approach [Maday et al., 2015b]. The method mix up models and measures and it can be seen as a least squares approach fitting between data and model, plus a corrector capable to account the model bias up to some extent. We briefly describe the methodology here, but the method is formally discussed in chapter 2.

In the context of PBDW, we have two building blocks:

1. We are required to model the m sensor devices by linear functionals $\ell_i(u) : V \to \mathbb{R}$. The space of observations is built up as follows: $W = \text{span}\{\omega_1, \ldots, \omega_m\}$, where ω_i denotes the unique Riesz representers of the $i-$th sensor, i.e., $\hat{\ell}_i(u) = \langle \omega_i, u \rangle$. We write $\hat{\ell}_i(u)$ just to denote an scaling of $\ell_i(u)$ such that the representers satisfies $\langle \omega_i, \omega_j \rangle = \delta_{ij}$, where δ_{ij} is a Kronecker delta.

2. The physical laws to which the state is subjected to can be written via a parameterized PDE $\mathcal{P}(u; y)$, for $y \in Y$, a space of parameters, typically a box in $\mathbb{R}^p$, for p the number of parameters of the system. We can define the set of solutions $\mathcal{M} = \{u \in V;\ \mathcal{P}(u, y) = 0\}$.

PBDW does a fitting between ω and a compressed version of $\mathcal{M}$, a $n-$dimensional space $V_n \subset V$ with an approximation bound $\text{dist}\,(u, V_n) \leq \varepsilon$. The fitting is done by means of solving the following optimization problem: find u^* such that:

$$\begin{aligned} \inf_{u \in (V_n \oplus (V_n^\perp \cap W))} & \frac{1}{2}\|u - P_{V_n} u\|^2, \\ \text{s.t } & P_W\,(u) = \omega. \end{aligned} \tag{1.4}$$

Definition 2. Let X and Y be two closed sub-spaces of V. We define:

$$\beta(X, Y) = \inf_{x \in X} \frac{\|P_Y\,(x)\,\|}{\|x\|}. \tag{1.5}$$

The following estimation bound holds when estimating u using (1.4):

$$\|u - u^*\| \leq \beta(W, V_n)^{-1}\,\text{dist}\left(u, V_n \oplus \left(W \cap V_n^\perp\right)\right).$$

1.2.3 Kolmogorov n-widths

We dedicate a section to this core subject of approximation theory in order to establish a vocabulary to asses the ability of compressing data-bases via model order reduction techniques.

The scope of this theoretical framework is vastly broader as it also touch subjects such as function approximation in Galerkin projections via finite element methods for partial differential equations (PDE).

Let V be a Hilbert space with norm $\|\cdot\|$, and let V_n be an $n-$dimensional subspace of V. We can define a distance of some point $x \in V$ to the space V_n as

$$E(x, V_n; V) \stackrel{\text{def}}{=} \inf_{x_n \in V_n} \|x - x_n\|.$$

Having the *point to space* distance, we can now introduce the error of approximating a compact set $\mathcal{M} \subset V$ with an $n-$dimensional space V_n

$$E(\mathcal{M}, V_n; V) \stackrel{\text{def}}{=} \sup_{y \in \mathcal{M}} \inf_{x_n \in V_n} \|x_n - y\|.$$

which gives some sort of a measure of how good the space V_n approximates the worst element of $\mathcal{M}$. Proposed by Andreï Kolmogorov in [Kolmogorov, 1936], the n-width, seen by us in many cases as a measure of how *compressible* an space is, is defined as:

$$d_n(\mathcal{M}; V) \stackrel{\text{def}}{=} \inf\{E(\mathcal{M}, V_n, V) : V_n \text{ a n-dimensional sub-space of } V\}.$$

This formal way of measuring how well we can approximate an space with sub-spaces of a certain dimension is of uttermost importance to endow model order reduction techniques with mathematical rigor.

Whenever a space $\mathcal{M}$ is said to have a fast decaying Kolmogorov n-width, we mean that $d_n(Y; X) \to 0$ *fast* (exponentially, for instance) as n grows. In chapters 2 and 3, moderately slow decaying n-width Kolmogorov function spaces[4] are explored in a bio-medical numerical experiment. In chapter 4, very fast decaying n-width Kolmogorov function spaces are explored in non parametric domains. We see in those chapters numerical outcomes that verifies what it has been exposed in this section. In fact, for the function spaces of chapters 2 and 3, set of solutions of the Navier-Stokes equations on a carotid domain, we see that in order to guarantee a good error approximation via model reduction using singular value decompositions we need around $\sim$35 basis functions for the reduced models, where as $\sim$6 basis functions are required in the examples of chapter 4, where non convective flows on simple Venturi domains are the sets of solutions to be compressed.

1.2.4 The singular value decomposition

The matrix $\mathbb{A}$ admits the following (unique) decomposition $\mathbb{A} = \hat{U}\hat{S}\hat{V}^T$, called full singular value decomposition (SVD), where $\hat{U} \in \mathbb{R}^{m\times m}$, $\hat{S} \in \mathbb{R}^{m\times N}$ and $\hat{V} \in \mathbb{R}^{N\times N}$, satisfies the orthogonality

[4] by convention of the model order reduction community, in chapters 2, 3 and 4, the function spaces, typically sets of solutions of some PDE, are refereed to as *solution manifolds*, in spite of the sometimes impossibility to show for instance, the Hausdorff axiom for neighborhoods for those sets

properties $\hat{U}^T\hat{U} = I_{N\times N}$, and $\hat{V}^T\hat{V} = I_{m\times m}$. $\hat{S}$ is filled with zeros excepting its diagonal, with entries $\sigma_1 > \sigma_2 > \dots \sigma_m$, singular values of $\mathbb{A}$.

Let us denote by $\mathbb{A} = USV^T$ the truncated decomposition (see [Trefethen and Bau, 1997] for details). We can geometrically interpret the SVD applying $\mathbb{A}$ to a unitary ball in $\mathbb{R}^m$ (recall we have assumed at previous section that $N > m$): an ellipsoid with axis defined by vectors pointing towards the columns of $U = \{u_1, \dots, u_m\}$, whose semi-axis sizes are the entries in the diagonal of S: $\sigma_1, \dots, \sigma_m$.

1.2.4.1 SVD for manifold compression

For any matrix X, let us denote X_r the sub-matrix of X containing the first r columns of X. In the context of linear model reduction, SVD is the gold standard, as it gives an optimal approximation of the space spanned by the columns of $\mathbb{A}$ in the sense that

$$U_r = \underset{X \text{ s.t. } \mathrm{Ran}(X)=r}{\arg\min} \|\mathbb{A} - XX^T\mathbb{A}\|_{\mathrm{F}},$$

where we denote by $\|\cdot\|_{\mathrm{F}}$ the Frobenius norm. Otherwise stated, we get the best r-dimensional basis for the space spanned by the columns of $\mathbb{A}$. We have stated the SVD for the matrix $\mathbb{A}$, so the streamline of ideas with the least-squares introduction to this overview is not interrupted. In practice, the matrix to be decomposed will have (see chapter 2) finite element solutions of the system governing dynamics.

1.2.4.2 SVD as a regularization technique

We can come back to equation (1.2) by using the SVD as a regularization technique. Since $\mathbb{A}$ is not invertible, it has zero singular values. Let $r < m$ be the number of non-zero singular values. We can approximate $\mathbb{A} \approx \hat{\mathbb{A}} = U_r S_r V_r^T$, so the solution to (1.3) reads

$$\begin{aligned} u^* &\approx \left(\hat{\mathbb{A}}^T\hat{\mathbb{A}}\right)^{-1}\hat{\mathbb{A}}^T l \\ &= \left((U_r S_r V_r^T)^T U_r S_r V_r^T\right)^{-1}\left(U_r S_r V_r^T\right)^T l \\ &= \left(V_r S_r U_r^T U_r S_r V_r^T\right)^{-1} V_r S_r U_r^T l \\ &= V_r S_r^{-2} V_r^T V_r S_r U_r^T l \\ &= V_r S_r^{-1} U_r^T l \end{aligned}$$

We adopt the notation $\hat{\mathbb{A}}^\dagger = \left(\hat{\mathbb{A}}^T\hat{\mathbb{A}}\right)^{-1}\hat{\mathbb{A}}^T$ for the Moore-Penrose pseudo-inverse of $\hat{\mathbb{A}}$.

1.2.5 Forecasting and parameter estimation

State estimation is one among some of the branches of inverse problems. We extend briefly the discussion with two bullets:

1. So far we have not even mention the word *time*, regardless that $\mathcal{P}$ in the previous section could describe a non-stationary problem. When addressing time in a particular way, and not seeing it as a parameter, we can talk about data assimilation instead of state estimation. The example by excellence of methods dealing with time for data assimilation of model and measures is Kalman filtering [Kalman, 1960], in any of its dynamic variants: linear filtering, unscented filtering or extended filtering. What this assimilation provides is a forecasting, and daily basis examples of the use of this technique in technology are abundant.

2. In some inverse problems engineers might decide that it would be better to mimic the forward problem from the measures by first estimating the system parameters and then running a simulation. Or, furthermore, maybe the application of interest does not need the state as the parameter space contains by itself relevant information. That is the case of the field in bio-medical engineering called elastography, dedicated to assess elastic parameters of body tissues from medical data, typically from magnetic resonance imaging (MRI).

1.3 Ultrasound echography

This section is devoted to discuss some fundamentals about medical imaging. In particular, we will extend the discussion about ultrasound (US) imaging since is the one we are going to synthetically generate for the numerical experiments of chapters 3, 4 and 5. We include this section to understand the data we want to assimilate, but also and since this section does not have encyclopedic purposes, we give enough notions about image acquisition in order to justify the physical models that we are going to use to reconstruct velocity fields, the linear functionals $\ell_i(u)$ defined in the section dedicated to PBDW 1.2.2.

1.3.1 Image acquisition

Ultrasound imaging (or echography) has several advantages with respect to other image modalities: it is cheap, fast, it has a good time sampling and it is portable. Instead of digging down in more aspects about US and other generalities, we describe here only the necessary fundamentals. The reader is refereed to the book [Barrie and Webb, 2011], which has inspired the following paragraphs and it is the source of all the figures and tables of this section.

US imaging is about sending mechanical pulses to human tissue with frequencies ranging from 10^6 Hz and 1.5×10^7 Hz. In order to generate those pulses, an electrical potential is induced to an array of piezo-electric sheets that allows the conversion of electromagnetic waves to mechanical ones. The array, normally called *transducer*, vibrates and induces the longitudinal movement of the particles of the tissues, a movement that ranges around orders of magnitude of 10^{-9} m. For

the scales we are interested in, the tissue movement is negligible, and we can therefore assume that the action of sending US waves to the media does not perturb the outcome, an assumption of importance in general, and of importance in particular from a data assimilation point of view, since it permits to state that the measurement process is decoupled from the system evolution.

The key of ultrasound imaging relies on the velocity of the particles in the tissue. Let us denote by u_x the longitudinal velocity of the tissue at a certain tissue depth, and let p denotes the pressure wave intensity. We have this sort of Ohm's law

$$Z = \frac{p}{u_x},$$

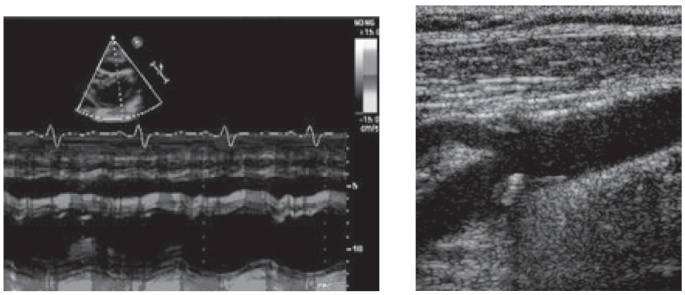
a basic elastic property, here refereed to as acoustic impedance. In fact, it is easy to see that $Z = \sqrt{\rho/\kappa}$, where ρ is the organ density and κ is the organ compressibility (a rephrase of the stiffness, standing for how much the particles resist to the movement parallel to the longitudinal waves sent by the US machine). Different tissues has different acoustic impedance, so we have, explained in a nutshell, a method for tissue characterization from mechanical waves. The transducer can, due to the piezo-electric effect, receive the pressure waves that are scattered back by the region of interest and transform it back to an electric signal. The complete underlying mechanism is much deeper but we are omitting that discussion here. See table 1.1 for averages values of the acoustic impedance Z in some tissues.

Table 1.1: Acoustic impedance as a way to characterize tissues.

	Air	Blood	Bone	Fat	Brain	Muscle	Liver	Kidney
$Z \times 10^5$ gr cm$^{-2}s^{-1}$	0.00043	1.59	7.8	1.38	1.58	1.7	1.65	1.62

So, as already brought up, the tissue back-scatterer the pressure signals so that we can post-process them. After dealing with the raw data, it is possible to distinguish three image modalities:

- (Amplitude) A-mode imaging: this mode acquires a one-dimensional *line-image* which plots the amplitude of the back-scattered echo time evolution. A tipycal application of A-mode imaging is to measure the eye corneal thickness.
- (Motion) M-mode imaging: a continuous series of A-mode lines are obtained and they are displayed as a function of time (see Figure 1.5(a)). The brightness of the displayed M-mode signal represents the amplitude of the back-scattered echo.
- (Brightness) B-mode imaging: the output for this mode is a two-dimensional image (see e.g. Figure 1.5(b)). Each line in the image is an A-mode line, with the intensity of each echo being represented by the brightness on the two dimensional scan.



(a) US M-mode of a heart wall (b) US B-mode of a carotid

Figure 1.5: Ultrasound image modalities.

1.3.2 Doppler ultrasound data

An application of US is the velocity estimation in blood vessels. Red blood cells are moving scatterers and they change the frequency of the pulses that comes to them. We can use that frequency shift to calculate the velocity in the direction parallel to the longitudinal waves (the beam), defined by an unitary vector $b \in \mathbb{R}^3$.

From the Doppler effect we know that the effective frequency perceived by the transducer from a moving scatterer in the blood (see Figure 1.6) can be written as follows:

$$f_i^{\text{eff}} = f_{\text{i}} \frac{c + v\cos(\theta)}{c}, \tag{1.6}$$

where f_i is the frequency of the longitudinal wave emitted by the transducer, θ is the angle

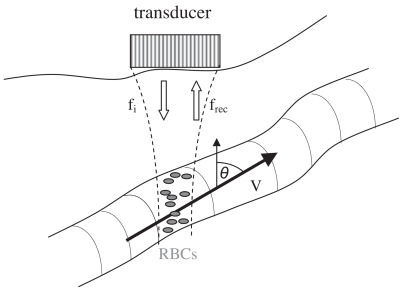


Figure 1.6: Conventional velocity estimation scheme.

between the beam direction and the flow direction, c is the speed of sound, assumed constant and equal to 1540 m/s, and v is the blood velocity. The same process occurs during the reception of the signal back in the transducer, so the received frequency f_{rec} is given by

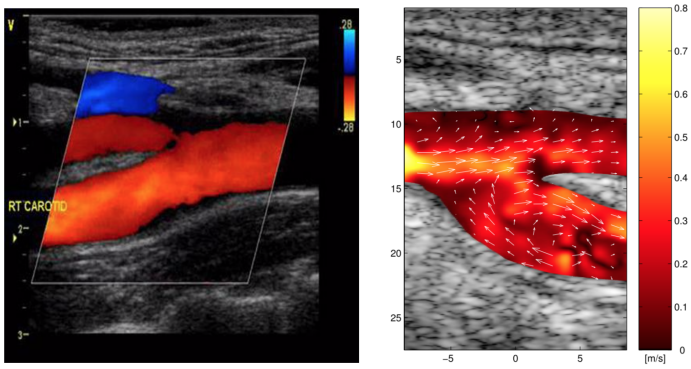
$$f_{\text{rec}} = f_i + \frac{2 f_i v\cos(\theta)}{c} + \frac{f_i v^2 \cos^2(\theta)}{c^2}, \tag{1.7}$$

where we can discard the quadratic term so that the overall frequency shift is given by

$$f_p = \frac{2|v|\cos(\theta)}{c} f_0. \tag{1.8}$$

An accurate measurement of blood velocity can only be achieved if the angle θ is known and is kept under certain threshold value, typically recommended under 60 degrees. That is why usually B-mode scans are acquired at the same time. Newer techniques that combine data from different beams also allow the obtention of two components of the velocity.

In summary, Doppler ultrasound provides two-dimensional mappings of one or two components of the velocity field. Those mappings are called *color flow image* (CFI) and *vector flow image* (VFI), respectively. An example of both is seen in Figure 1.7.



(a) Color flow image (b) Vector flow image

Figure 1.7: Doppler flow imaging in the common carotid branches.

Back to the language of section 1.2.2, a CFI provides a set of linear functionals $\{\ell_i(u)\}_{i=1}^m$, defined over m disjoint voxels $\{\Omega_i\}_{i=1}^m$, with $\Omega_i \subset \mathbb{R}^3$, for $i = 1, \ldots, m$

$$\ell_i(u) = \int_{\Omega_i} u \cdot b \, \mathrm{d}s.$$

1.4 Summary of contributions

Publications

- F. Galarce, J.-F. Gerbeau, D. Lombardi, O. Mula. 2020. *Fast Reconstruction of 3D Blood Flows from Doppler Ultrasound Images and Reduced Models*. **Computer Methods in Applied Mechanics and Engineering**. Elsevier.
- F. Galarce, D. Lombardi, O. Mula. 2020. *Reconstructing Haemodynamics Quantities of Interest from Doppler Ultrasound Imaging*. **International Journal for Numerical Methods in Bio-medical Engineering**. John Wiley & Sons.

- F. Galarce, D. Lombardi, O. Mula. *Inverse Problems on Non-Parametric Domains.Flow Reconstruction from Medical Data Using Non Linear Dimensionality Reduction.* **Ongoing work with expected preprint for early 2021**.

MAD

The comprehensive software MAD, developed in C++ during the current Ph.D. thesis, and further described in chapter 5, is a contribution for the automatization of model order reduction techniques, 3D parallel solvers for the Navier-Stokes equations, geometry registration routines and machine learning tools.

Presentation of thesis work in conferences

- CSE2021: accepted to participate in contributed talk at *SIAM Conference on Computational Science and Engineering*. Title: Inverse Problems on Non-Parametric Domains. Flow Reconstruction from Medical Data Using Non Linear Dimensionality Reduction. March 2021, Fort Worth, Texas, U.S. To be held online due to covid.

- VPH2020: Contributed talk at *Virtual Physiological Human conference*. Title: Fast assimilation of Doppler measures for state estimation of blood flows and quantities of medical interest. August, 2020, Paris, France. Held online due to covid.

- Enumath 2019: Contributed talk at *European Numerical Mathematics and Advanced Applications Conference*. Title: 3D flow reconstruction from Doppler ultrasound data. October 2019, Egmond aan Zee, The Netherlands.

- CSMA 2019: Contributed talk at *14ème Colloque National en Calcul des Structures*. Title: Reconstruction of blood flows from Doppler images. May 2019, Presqu'ile de Giens, France.

- Eccomas 2018: Contributed talk at 6th European Conference on Computational Mechanics (ECCM 6) & 7th European Conference on Computational Fluid Dynamics (ECFD 7). Title: Optimal reconstruction of flows from Doppler measurements. June 2018, Glasgow, UK.

- MoRePas 2018. Poster presentation at *conference for Model Reduction for Parameterized Systems*. Title: Enhancing Hemodynamics Measurements with Mathematical Modelling. April 2018, Nantes, France.

Chapter 2

Fast reconstruction of 3D blood flows from Doppler ultrasound images and reduced models

---

The content shown in this chapter inspired the following scientific publication: F. Galarce, J.F. Gerbeau, D. Lombardi and O. Mula. (2020). *Fast reconstruction of 3D blood flows from Doppler ultrasound images and reduced models*. **Computer Methods in Applied Mechanics and Engineering**. Elsevier.

This chapter is focused on presenting techniques for the state estimation of an element that belongs to a certain Hilbert space V from partial observations. One simple and efficient strategy is the so-called Parameterized Background Data-Weak approach (PBDW, see [Maday et al., 2015b]). It is a linear mapping that consists in a least squares fit between the measurement data and a linear reduced model to which a certain correction term is added. However, in the original approach, the reduced model is built from forward reduced modeling and independently of the reconstruction task (typically with a proper orthogonal decomposition or a greedy algorithm). The methodologies will thus differ on the reduced basis construction, where we introduce a data-driven technique more adapted to the reconstruction task. We will compare the performance of the proposed approaches on a realistic 3D scenario of velocity recovery in a human artery. Nevertheless, the interest will remain close to the theory in this chapter, reason why we have allowed ourselves to choose $V = [L^2(\Omega)]^3$ for our tests. Additionally, only noiseless measures are tested in this chapter. Those two hypothesis will be reconsidered in chapter 3, in which the medical usability of our results will be the main focus.

Contents

2.1 Introduction

Developing artificial intelligence-driven tools to assist doctors in medical decisions and diagnosis requires to solve efficiently and in a reliable manner data assimilation and inverse problems from biomedical applications. These problems share the following general common features, which will guide our subsequent developments:

- The available data are often corrupted by noise and obtained with medical imaging techniques, which have the advantage of being *non-invasive*.
- There may be morphological constraints that prevent from measuring at specific locations.
- In some cases, the device may not be able to measure directly the desired quantity of interest (QoI), and a complex post-processing may be required to obtain an estimate of it.
- Sometimes, the desired QoI is a prediction or a forecast ahead in time, which requires a specific treatment to be inferred.

In this chapter, we develop state estimation techniques which aim at taking the above features into account, and which involve reduced modeling of parameterized Partial Differential Equations (PDEs). As an illustrative example of how the reconstruction techniques that we deploy address some of the above points, in this chapter, we consider a haemodynamics inverse problem: the real-time reconstruction of the full 3D blood velocity field in an artery from Doppler ultrasound images taken on a restricted portion of the artery. This application is of interest in its own right since a real-time reconstruction of the full flow would enrich the available information for medical diagnosis.

Haemodynamics forward and inverse problems is a wide multidisciplinary topic with a long history, and a complete overview of the numerous existing contributions would go beyond the scope of the present chapter. However, as a brief summary of the state of the art for our application on haemodynamics reconstructions using medical images, we could start by citing [Aderson, 1998], where a method based on quadratures is used to estimate the velocity from Doppler ultrasound. In [Moireau et al., 2013, Hu et al., 2007, Caiazzo et al., 2017, Müller et al., 2018, Adib et al., 2016] sequential data estimation methods are used to provide a description of the haemodynamics in different portions of the vascular tree. A patient specific fluid-structure interaction simulation of the left ventricle is proposed in [Lassila et al., 2012] based on Magnetic Resonance Imaging data. The calibration of computational fluid mechanics simulations was also investigated in [Koltukluoğlu, 2019]. A multi-fidelity approach and a Bayesian framework are detailed in [Perdikaris and Karniadakis, 2016]. Recently, some works have explored kernel and deep learning based approaches to perform data assimilation [Koeppl et al., 2018, Huttunen et al., 2020, Kissas et al., 2020].

In this work, we use methods based on reduced-order modelling to obtain reconstructions in close to real-time. This approach has attracted considerable attention and numerous mathematical developments and applications have been developed, see [Khalil et al., 2007, Astrid et al., 2008, Buffoni et al., 2008, Leroux et al., 2013, Maday and Mula, 2013, Maday et al., 2015a, Raiola et al., 2015, Argaud et al., 2017, Kärcher et al., 2018, Cohen et al., 2019]. These methods share connections with other well-known approaches such as 3D and 4D-Var (see [Lorenc, 1981, Le Dimet and Talagrand, 1986]) or the Partial Spline Model ([Wahba, 1990, Chapter 9]). We refer to [Taddei, 2017, Kärcher et al., 2018] for further details on this point. Regarding inverse haemodynamics problems, prior contributions involving reduced-order modeling are, e.g., [Lassila et al., 2011, Pant et al., 2017].

Our starting point is the Parameterized Background Data-Weak originally introduced in [Maday et al., 2015b]. The method consists in a linear mapping which is built from a least squares fit between the measurements and a given reduced space and an additional term that is capable of correcting model bias to some extent. In the original approach of PBDW, the reduced model is built using forward reduced modeling and independently of the reconstruction task (typically with a Proper Orthogonal Decomposition or a greedy algorithm). In this chapter, we investigate the construction of other reduced spaces which are built to be better adapted to the reconstruction task and which result in mappings that are sometimes nonlinear. We compare the performance of the different algorithms in the context of the fast reconstruction of the velocity field in an artery. The results illustrate the superiority of the proposed alternatives to the classical approach involving linear reduced bases.

The chapter is organized as follows, in section 2.2 we describe mathematically what we understand by state estimation problems and reconstruction algorithms (section 2.2.1). We next introduce several ways of building a reduced order space, some of which take the measurements into account, henceforth leading to non-linear reconstruction methods (section 2.2.2). We give details on the practical implementation and computational costs in section 2.3. We next apply the methodology to the reconstruction of 3D blood flows from Doppler images (section 2.4). The example is semi-realistic, meant to be a first (still idealized) step towards realistic applications. The main assumptions are the following. First, the Doppler images are noiseless and synthetically computed but they have been generated with a realistic model with respect to the real involved physics and measurement devices. This assumption is mainly due to our lack of real Doppler measurements and also because of the very involved space-time structure of the noise in Doppler images (which is a research topic in itself [Ledoux et al., 1997, Bjaerum et al., 2002, Demené et al., 2015]). Second, we assume that there is no model misfit even if the method can correct it to some extent. However, we use a rather realistic model for flows in large arteries based on incompressible Navier-Stokes equations, with

boundary conditions used in haemodynamics simulations (see [Formaggia et al., 2009]). The precise parametric PDE model upon which we build our reduced models is explained in section 2.4.1. Section 2.4.2 gives some details on Doppler ultrasound imaging and the way in which we have incorporated it to our methodology. Finally, sections 2.4.3 and 2.4.4 present numerical results in two test cases. They illustrate the superiority of the proposed reconstruction algorithms with respect to the classical linear PBDW. In addition to this, our second example also shows that the method can be used to estimate quantities of interest which could be helpful in the detection of an arterial blockage.

We conclude this introduction by emphasizing that the assumptions in our computations with no model misfit and synthetic, noiseless measurements are not inherent limitations of the present approach. The focus lies rather on the reduced space construction and the haemodynamics application. The model misfit is automatically corrected and measurement noise can be added to the recostruction pipeline (see [Taddei, 2017, Argaud et al., 2017, Gong et al., 2019] for works on measurement noise). We also emphasize the novelty that the current approach represents for the field of blood flow reconstruction. Indeed, estimations performed routinely by medical doctors usually do not involve full reconstructions (perhaps due to the lack of efficient techniques). They often reduce to simple pointwise estimations of the peak and average velocities of the imaged anatomical part. The development of more refined estimations of the velocity field are actually an active research topic, especially for flows in the heart cavities. The only fast technique that we are aware of is based solely on the divergence equation ($\nabla \cdot u = 0$) on the imaging plane [Ohtsuki and Tanaka, 2006, Uejima et al., 2010]. Despite its rapidity and simplicity, the contribution of the out-of-plane velocity and the momentum conservation are neglected, leading to very non-physical approximations. Another existing alternative are classical inverse problems in which a joint state-parameter estimation is performed. Their main drawback is the large computational times, which is incompatible to the clinical practice.

2.2 Reconstruction methods

We start by introducing the state reconstruction methods that we use in the present work. We focus on explaining alternative ways of building reduced basis that are better tailored for the task of state estimation than the usual reduced bases of forward problems. Our presentation is done for noiseless measurements and assumes that there is no model error. As already brought up, this is due to the fact that we lack from real measurements for our targeted application.

2.2.1 State estimation and recovery algorithms

Let Ω be a domain of $\mathbb{R}^d$ for a given dimension $d \geq 1$ and let V be a Hilbert space defined over Ω, with inner product $\langle \cdot, \cdot \rangle$ and norm $\| \cdot \|$. Our goal is to recover an unknown function $u \in V$ from m measurement observations

$$\ell_i(u), \quad i = 1, \ldots, m, \tag{2.1}$$

where the ℓ_i are linearly independent linear forms over V. In many applications, each ℓ_i models a sensor device which is used to collect the measurement data $\ell_i(u)$. If the observations come in the form of an image as in the application of this thesis, each ℓ_i may represent the response of the system in a given pixel. The Riesz representers of the ℓ_i are denoted by ω_i and span an m-dimensional space

$$W_m = \operatorname{span}\{\omega_1, \ldots, \omega_m\} \subset V.$$

The observations $\ell_1(u), \ldots, \ell_m(u)$ are thus equivalent to knowing the orthogonal projection

$$\omega = P_{W_m} u. \tag{2.2}$$

In this setting, the task of recovering u from the measurement observation ω can be viewed as building a recovery algorithm

$$A : W_m \mapsto V$$

such that $A(P_{W_m} u)$ is a good approximation of u in the sense that $\|u - A(P_{W_m} u)\|$ is small.

Recovering u from the measurements $P_{W_m} u$ is a very ill-posed problem since there are infinitely many $v \in V$ such that $P_{W_m} v = \omega$. It is thus necessary to add some a priori information on u in order to recover the state up to a guaranteed accuracy.

We are motivated by the setting where u is a solution to some parameter-dependent PDE of the general form

$$\mathcal{P}(u, y) = 0,$$

where $\mathcal{P}$ is a differential operator and y is a vector of parameters that describes some physical property and lives in a given set $\mathrm{Y} \subset \mathbb{R}^p$. Therefore, our prior on u is that it belongs to the set

$$\mathcal{M} \stackrel{\text{def}}{=} \{u(y) \in V \ : \ y \in \mathrm{Y}\}, \tag{2.3}$$

which is sometimes referred to as the *solution manifold*. The performance of a recovery mapping A is usually quantified in two ways:

- If the sole prior information is that u belongs to the manifold $\mathcal{M}$, the performance is usually measured by the worst case reconstruction error

$$E_{\mathrm{wc}}(A, \mathcal{M}) = \sup_{u \in \mathcal{M}} \|u - A(P_{W_m} u)\| .$$

- In some cases u is described by a probability distribution p on V supported on $\mathcal{M}$. This distribution is itself induced by a probability distribution on Y that is assumed to be known. In this Bayesian-type setting, the performance is usually measured in an average sense through the mean-square error

$$E_{\text{ms}}^2(A, \mathcal{M}) = \mathbb{E}\left(\|u - A(P_{W_m} u)\|^2\right) = \int_V \|u - A(P_{W_m} u)\|^2 dp(u)\,,$$

  and it naturally follows that $E_{\text{ms}}(A, \mathcal{M}) \leq E_{\text{wc}}(A, \mathcal{M})$.

Let us take a look to figure 2.1. The problem to address is about receiving an element from W_m, and complement it with the compressed model data in V_n. Notice how the angle between both spaces has intuitively an impact in the reconstruction quality, confirmed with the theoretical bound shown in the next section.

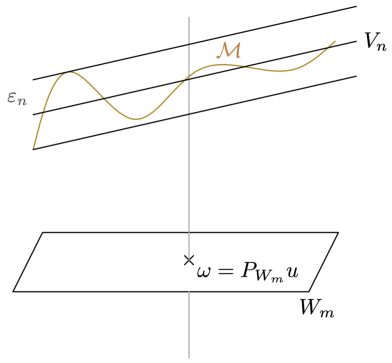


Figure 2.1: Geometrical interpretation for the reconstruction problem of finding $A : W_m \to V$.

2.2.2 Linear and nonlinear algorithms using reduced modeling

Reduced models are a family of methods that produce each a hierarchy of spaces $(V_n)_{n\geq 1}$ that approximate the solution manifold well in the sense that

$$\varepsilon_n \overset{\text{def}}{=} \sup_{u\in\mathcal{M}} \operatorname{dist}(u, V_n)\,, \quad \text{or} \quad \delta_n^2 \overset{\text{def}}{=} \mathbb{E}\left(\operatorname{dist}(u, V_n)^2\right)$$

decays rapidly as n grows for certain classes of PDEs. Several methods exist to build these spaces among which stand the reduced basis method (see [Rozza et al., 2007]), the (Generalized) Empirical Interpolation Method (see [Barrault et al., 2004, Maday and Mula, 2013, Maday et al., 2016]), Proper Orthogonal Decomposition (POD, [Sirovich, 1987, Berkooz et al., 1993]) and low-rank methods (see [Cohen et al., 2011, Cohen and DeVore, 2015]).

Linear reconstruction algorithms that make use of reduced spaces V_n are the Generalized Empirical Interpolation Method (GEIM) introduced in [Maday and Mula, 2013] and further analyzed in [Maday et al., 2015a, Maday et al., 2016] and the Parameterized Background Data-Weak Approach (PBDW) introduced in [Maday et al., 2015b] and further analyzed in [Binev et al., 2017a]. Note that some modified versions have been proposed to address measurement noise (see, e.g., [Argaud et al., 2017, Taddei, 2017]) and other recovery algorithms involving reduced modelling have also been recently proposed (see [Kärcher et al., 2018]).

2.2.2.1 PBDW, a linear recovery algorithm

Given a measurement space W_m and a reduced model V_n with $1 \leq n \leq m$, the PBDW algorithm

$$A_{m,n}^{(\mathrm{pbdw})} : W_m \to V$$

gives for any $\omega \in W_m$ a solution of

$$\min_{u \in \omega + W^\perp} \mathrm{dist}(u, V_n).$$

Denoting

$$\beta(X,Y) \stackrel{\mathrm{def}}{=} \inf_{x \in X} \sup_{y \in Y} \frac{\langle x, y \rangle}{\|x\| \, \|y\|} = \inf_{x \in X} \frac{\|P_Y x\|}{\|x\|} \in [0,1] \tag{2.4}$$

for any pair of closed subspaces (X,Y) of V, the above optimization problem has a unique minimizer

$$A_{m,n}^{(\mathrm{pbdw})}(\omega) = u_{m,n}^*(\omega) \stackrel{\mathrm{def}}{=} \operatorname*{arg\,min}_{u = \omega + W^\perp} \mathrm{dist}(u, V_n). \tag{2.5}$$

as soon as $n \leq m$ and $\beta(V_n, W_m) > 0$. We adhere to these two assumptions in the following.

As proven in section 2.3.1, an explicit expression of $u_{m,n}^*(\omega)$ is

$$u_{m,n}^*(\omega) = v_{m,n}^*(\omega) + \omega - P_{W_m} v_{m,n}^*(\omega) \tag{2.6}$$

with

$$v_{m,n}^*(\omega) = \left(P_{V_n|W_m} P_{W_m|V_n}\right)^{-1} P_{V_n|W_m}(\omega), \tag{2.7}$$

where, for any pair of closed subspaces (X,Y) of V, $P_{X|Y} : Y \to X$ is the orthogonal projection into X restricted to Y. The invertibility of the operator $P_{V_n|W_m} P_{W_m|V_n}$ is guaranteed under the above conditions.

Formula (2.6) shows that $A_{m,n}^{(\mathrm{pbdw})}$ is a bounded linear map from W_m to $V_n \oplus (W_m \cap V_n^\perp)$. Depending on whether V_n is built to address the worst case or mean square error, the reconstruction performance is bounded by

$$e_{m,n}^{(\mathrm{wc,\ pbdw})} = E_{\mathrm{wc}}(A_{m,n}^{(\mathrm{pbdw})}, \mathcal{M}) \leq \beta^{-1}(V_n, W_m) \max_{u \in \mathcal{M}} \mathrm{dist}(u, V_n \oplus (V_n^\perp \cap W_m)) \leq \beta^{-1}(V_n, W_m)\, \varepsilon_n, \tag{2.8}$$

or

$$e_{m,n}^{(\text{ms, pbdw})} = E_{\text{ms}}(A_{m,n}^{(\text{pbdw})}, \mathcal{M}) \overset{\text{def}}{=} \mathbb{E}\left(\|u - A_n^{(\text{pbdw})}(P_{W_m}u)\|^2\right)^{1/2}$$
$$\leq \beta^{-1}(V_n, W_m)\mathbb{E}\left(\text{dist}(u, V_n \oplus (V_n^{\perp} \cap W_m)^2\right)^{1/2}$$
$$\leq \beta^{-1}(V_n, W_m)\,\delta_n, \tag{2.9}$$

Note that $\beta(V_n, W_m)$ can be understood as a stability constant. It can also be interpreted as the cosine of the angle between V_n and W_m. The error bounds involve the distance of u to the space $V_n \oplus (V_n^{\perp} \cap W_m)$ which provides slightly more accuracy than the reduced model V_n alone. In the following, to ease the reading we will write errors only with the second type of bounds that do not involve this correction part on $V_n^{\perp} \cap W_m$.

An important observation is that for a fixed measurement space W_m (which is the setting in our numerical tests), the errors $e_{m,n}^{(\text{wc, pbdw})}$ and $e_{m,n}^{(\text{ms, pbdw})}$ reach a minimal value $e_{m,n_{\text{wc}}^*}^{(\text{wc, pbdw})}$ and $e_{m,n_{\text{ms}}^*}^{(\text{ms, pbdw})}$ as the dimension n varies from 1 to m. This behavior is due to the trade-off between the increase of the approximation properties of V_n as n grows and the degradation of the stability of the algorithm, given here by the decrease of $\beta(V_n, W_m)$ to 0 as $n \to m$. As a result, the best reconstruction performance with PBDW is given by

$$e_{m,n_{\text{wc}}^*}^{(\text{wc, pbdw})} = \min_{1\leq n\leq m} e_{m,n}^{(\text{wc, pbdw})}, \quad \text{or} \quad e_{m,n_{\text{ms}}^*}^{(\text{ms, pbdw})} = \min_{1\leq n\leq m} e_{m,n}^{(\text{ms, pbdw})}.$$

We finish this section with two remarks:

1. We do not consider measurement noise and model error. However, the PBDW algorithm can correct to some extent the model misfit (through the term η^*, see 2.3.1). Also, some extensions of the current setting have been proposed to address measurement noise (see [Taddei, 2017, Argaud et al., 2017, Gong et al., 2019]).

2. Note that in the present setting the measurement space W_m is fixed and we will adhere to this assumption in the rest of the thesis. This is reasonable since usually the nature and location of the sensors is fixed in our application following the experience of the medical doctors. A different, yet related problem, would be to optimize the choice of the measurement space W_m and we refer to [Maday et al., 2016, Binev et al., 2018, Bensoussan, 1972, Aidarous et al., 1975, Cannon and Klein, 1971, Yu and Seinfeld, 1973] for works on this topic.

2.2.2.2 Alternative reconstruction algorithms

Taking PBDW as a starting point, we next describe two different strategies to build recovery algorithms. They all incorporate an affine extension of PBDW that we explain next. One

motivation to introduce it is because it can give more accuracy when the solution manifold $\mathcal{M}$ is not localized near the origin. This typically happens when the state u is a perturbation of a nominal state $\bar{u} \in V$.

Affine PBDW: Given an average or nominal state $\bar{u} \in V$, we can formulate the equivalent affine version of PBDW, which reads

$$A_{m,n}^{(\mathrm{aff})}(\omega) \stackrel{\mathrm{def}}{=} \underset{u \in \omega + W^{\perp}}{\arg\min}\ \mathrm{dist}(u, \bar{u} + V_n), \tag{2.10}$$

As shown in 2.3.1, this algorithm can also be written as

$$A_{m,n}^{(\mathrm{aff})}(\omega) = \bar{u} + A_{m,n}^{(\mathrm{pbdw})}(\omega - \bar{\omega}),$$

where $\bar{\omega} = P_{W_m}\bar{u}$ and $A_{m,n}^{(\mathrm{pbdw})}$ is the linear PBDW algorithm of the previous section. We can easily see through the last expression that $A_{m,n}^{(\mathrm{aff})}$ is an affine mapping from W_m to $\bar{u} + V_n \oplus (W_m \cap V_n^{\perp})$. An interesting feature of this affine extension is that if $\omega = 0$, the algorithm yields a nonzero reconstruction in ${W_m}^{\perp}$ based on the nominal state $\bar{u}$, $A_{m,n}^{(\mathrm{aff})}(0) = \bar{u} - A_{m,n}^{(\mathrm{pbdw})}(\bar{\omega})$. This is in contrast to the linear version where $A_{m,n}^{(\mathrm{pbdw})}(0) = 0$. We also have $A_{m,n}^{(\mathrm{aff})}(\overline{\omega}) = \overline{u}$.

Proceeding similarly as before, for a given $u \in \mathcal{M}$, the error is bounded by (see [Cohen et al., 2019])

$$\begin{aligned} \|u - A_{m,n}^{(\mathrm{aff})}(\omega)\| &\leq \beta^{-1}(V_n, W)\,\mathrm{dist}(u, \bar{u} + V_n \oplus (W_m \cap {V_n}^{\perp})) \\ &\leq \beta^{-1}(V_n, W)\,\mathrm{dist}(u, \bar{u} + V_n) \end{aligned} \tag{2.11}$$

and

$$e_{m,n}^{(\mathrm{wc,\ aff})} \stackrel{\mathrm{def}}{=} E_{\mathrm{wc}}(A_{m,n}^{(\mathrm{aff})}, \mathcal{M}) \leq \beta^{-1}(V_n, W_m)\,\varepsilon_n^{(\mathrm{aff})}, \tag{2.12}$$

or

$$e_{m,n}^{(\mathrm{ms,\ aff})} \stackrel{\mathrm{def}}{=} E_{\mathrm{ms}}(A_{m,n}^{(\mathrm{aff})}, \mathcal{M}) \stackrel{\mathrm{def}}{=} \mathbb{E}\left(\|u - A_{m,n}^{(\mathrm{aff})}(P_{W_m}u)\|^2\right)^{1/2} \leq \beta^{-1}(V_n, W_m)\,\delta_n^{(\mathrm{aff})}, \tag{2.13}$$

where

$$\varepsilon_n^{(\mathrm{aff})} \stackrel{\mathrm{def}}{=} \sup_{u \in \mathcal{M}} \mathrm{dist}(u, \bar{u} + V_n)\,, \quad \text{or} \quad (\delta_n^{(\mathrm{aff})})^2 \stackrel{\mathrm{def}}{=} \mathbb{E}\left(\mathrm{dist}(u, \bar{u} + V_n)^2\right)$$

In the following, to simplify notation, we will use ε_n and δ_n to denote either the error in the linear PBDW or its affine version since the reasoning and the estimates that will be derived next have the same form.

Partition of $\mathcal{M}$: Our first strategy stems from the fact that we may have very few observations (m small) and since we must have $n \leq m$ for reconstruction, we may not have enough approximation power in the interval where n can range. Also, the approximation errors ε_n or δ_n provided by reduced basis may not always decrease rapidly to zero (the Kolmogorov n-width of $\mathcal{M}$ may decrease slowly). However, the physical structure of the problem could give a natural decomposition of the manifold $\mathcal{M}$ into different subdomains $\mathcal{M}^{(k)}$

that are better adapted for model reduction in the sense that the errors $\varepsilon_n^{(k)}$ or $\delta_n^{(k)}$ may decrease faster. Works based on that partition principle have been developed in the context of model reduction of forward problems [Carlberg, 2015, Amsallem and Haasdonk, 2016, Amsallem et al., 2012, Peherstorfer et al., 2014, Cohen et al., 2020]) and we propose to adapt it for data assimilation. A simple setting where we can decompose the manifold in our context is when we work in an application for which it is possible to know exactly a subset of entries in y for any target $u(y)$ during the online phase, say $\overline{y} \in \mathbb{R}^{\overline{p}}$ (with $\overline{p} < p$). Given $\overline{y}$ and a data-base generated from the governing PDE we may produce a disjoint union of K subsets $\mathrm{Y}^{(k)}$ which yields a decomposition of $\mathcal{M}$ into subsets $\mathcal{M}^{(k)} = u(\mathrm{Y}^{(k)})$. We can thus build reduced models $V_n^{(k)}$ for each subset $\mathcal{M}^{(k)}$ and then reconstruct with the linear or affine PBDW.

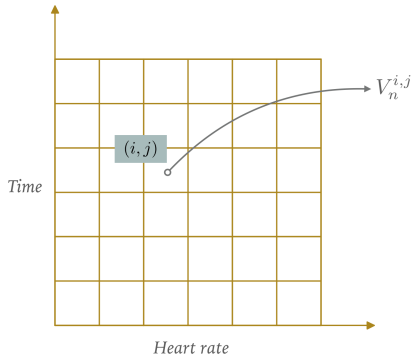


Figure 2.2: Manifold splitting and reduced models V_n^k on each partition. In our application is possible to access to entries of the parameter space during the online phase: Heart rate and time.

Proceeding similarly as in the previous section, the reconstruction performance on subset $\mathcal{M}^{(k)}$ is

$$e_{m,n}^{(\text{wc, aff, k})} = E_{\text{wc}}(A_{m,n}^{(\text{aff})}, \mathcal{M}^{(k)}) \leq \beta^{-1}(V_n^{(k)}, W_m)\, \varepsilon_n^{(k)},$$

or

$$e_{m,n}^{(\text{ms, aff, k})} = E_{\text{ms}}(A_{m,n}^{(\text{aff})}, \mathcal{M}^{(k)}) \overset{\text{def}}{=} \mathbb{E}\left(\|u - A_{m,n}^{(\text{aff})}(P_{W_m} u)\|^2\right)^{1/2} \leq \beta^{-1}(V_n^{(k)}, W_m)\, \delta_n^{(k)}.$$

The best reconstruction performance for $\mathcal{M}^{(k)}$ is thus

$$e_{m,n_{\text{wc}}^*(k)}^{(\text{wc, aff, k})} = \min_{1\leq n\leq m} e_{m,n}^{(\text{wc, aff, k})}, \quad \text{or} \quad e_{m,n_{\text{ms}}^*(k)}^{(\text{ms, aff, k})} = \min_{1\leq n\leq m} e_{m,n}^{(\text{ms, aff, k})}.$$

It follows that the performance in $\mathcal{M} = \cup_{k=1}^K \mathcal{M}^{(k)}$ is

$$e_m^{(\text{wc, aff})} = \max_{1\leq n\leq m} e_{m,n_{\text{ms}}^*(k)}^{(\text{wc, aff, k})}, \quad \text{or} \quad e_{m,n_{\text{ms}}^*(k)}^{(\text{ms, aff, k})} = \sum_{k=1}^K \omega_k e_{m,n_{\text{ms}}^*(k)}^{(\text{ms, aff, k})},$$

where $\omega_k = p(u \in \mathcal{M}^{(k)})$.

Data-based reduced models: The second strategy is motivated by the observation that the reduced models V_n or $V_n^{(k)}$ of the previous approaches are built independently of the given measurement space W_m and also of the given measurement observation ω. For a given $\omega \in W_m$, we can build a data-driven $V_n(\omega)$ with an orthogonal matching pursuit greedy algorithm (OMP) that we explain next. Once this reduced model has been computed, we reconstruct with the data-driven affine version of PBDW,

$$A_{m,n}^{(\mathrm{dd})}(\omega) \stackrel{\mathrm{def}}{=} \underset{u \in \omega + W^{\perp}}{\arg\min}\ \mathrm{dist}(u, \bar{u} + V_n(\omega)), \tag{2.14}$$

where the difference with respect to (2.10) is that now V_n depends on ω. The reconstruction performance of this algorithm is bounded by

$$e_{m,n}^{(\mathrm{wc,\ dd})} = E_{\mathrm{wc}}(A_{m,n}^{(\mathrm{dd})}, \mathcal{M}) \leq \sup_{u \in \mathcal{M}} \beta^{-1}(V_n(P_{W_m} u), W)\, \mathrm{dist}(u, \bar{u} + V_n(P_{W_m} u)),$$

in the worst case setting. Similarly as before, $e_{m,n}^{(\mathrm{wc,\ dd})}$ reaches a minimum $e_{m,n^*_{(\mathrm{dd})}}^{(\mathrm{wc,\ dd})}$ when n varies from 1 to m. Since V_n is now adapted to the measurement observations, we expect that the current algorithm performs better than its classical linear counterpart.

We now present the OMP greedy algorithm that we propose. Let

$$\mathcal{D} \stackrel{\mathrm{def}}{=} \{v = u/\|u\| \ :\ u \in \mathcal{M}\}$$

be the set of normalized functions from $\mathcal{M}$.

If $\bar{u} = 0$, the first element φ_1 is chosen as

$$\varphi_1 = \frac{1}{\#\mathcal{D}} \sum_{u \in \mathcal{D}} u. \tag{2.15}$$

For $n > 1$, given $V_n = \mathrm{span}\{\varphi_1, \ldots, \varphi_n\}$, we select

$$\varphi_{n+1} \in \underset{v \in \mathcal{D}}{\arg\max} \left| \left\langle \omega - P_{P_{W_m} V_n} \omega, \frac{P_{W_m} v}{\|P_{W_m} v\|} \right\rangle \right| \tag{2.16}$$

where $P_{W_m} V_n = \mathrm{span}\{P_{W_m}\varphi_1, \ldots, P_{W_m}\varphi_n\}$. We set $V_{n+1} = \mathrm{span}\{V_n, \varphi_{n+1}\}$.

Note that all operations in this algorithm are done in the space W_m. Hence we can do all calculations in $\mathbb{R}^m$, which makes this algorithm be very fast since it does not involve computations with functions from the whole space V (see 2.3.2 for further details on its implementation).

In the case $\bar{u} \neq 0$, we introduce $\bar{\omega} = P_{W_m}\bar{u}$ and the shifted set

$$\delta_{\bar{u}}\mathcal{D} = \left\{ v = \frac{u - \bar{u}}{\|u - \bar{u}\|} \ :\ u \in \mathcal{M},\, u \neq \bar{u} \right\}.$$

Now it suffices to apply the previous greedy algorithm to the target function $\omega - \bar{\omega}$ instead of w and do the search over $\delta_{\bar{u}}\mathcal{D}$ instead of $\mathcal{D}$.

Remark 2.2.1. *Note that both nonlinear approaches are very different in nature: the first is based on localization and the second is purely data-driven. Their performance will thus be very dependent on the physical problem under consideration and the measurement setting: the nature of W_m, the nature of the actual measurements $\omega = P_{W_m} u$, the nature of the manifold and its alignment with W_m locally at ω. Note also that both approaches can be combined and we can consider a piece-wise data-driven algorithm (its performance has actually been tested in the numerical tests below).*

2.3 Computational details of the methods

In the previous section and for the sake of clarity, we have given an idealized description of PBDW and the other reconstruction methods where we do not mention certain discretization aspects that inevitably come into play. To start with, the snapshot functions $u \in \mathcal{M}$ cannot be computed exactly but only up to some tolerance. One typical instance (which is used in our work) is the finite element method which gives an approximation $u_h \in V_h$ of $u \in V$ where V_h is a finite element space with $\mathcal{N}$ degrees of freedom. Therefore the computable states are elements of the perturbed manifold

$$\mathcal{M}_h \stackrel{\text{def}}{=} \{u_h(y) \in V_h \, : \, y \in \mathrm{Y}\}.$$

The computation of reduced models V_n involves a finite training subset $\widetilde{\mathcal{M}}_{\text{train}} \subset \mathcal{M}_h$ of snapshots, and we denote by $\#\widetilde{\mathcal{M}}_{\text{train}}$ its cardinality. As a consequence, all the reduced models are low dimensional subspaces of V_h. Note that the fact that the true states do not belong to $\mathcal{M}_h$ can be interpreted as a model bias.

The methodology also requires the computation of the Riesz lifts ω_i in order to define the measurement space W_m. Since we work in the space V_h, these can be defined as elements of this space satisfying

$$\langle \omega_i, v \rangle = \ell_i(v), \quad v \in V_h,$$

thus resulting in a measurement space $W \subset V_h$.

In what follows, we keep our idealized discussion and do not systematically recall that all computations take place in a background discretization space V_h.

2.3.1 Algebraic formulation of PBDW

Here we derive the algebraic formulation of $u^*_{m,n}(\omega)$, the function given by the linear PBDW algorithm. Let X and Y be two finite dimensional subspaces of V and let

$$\begin{aligned} P_{X|Y} : Y &\to X \\ y &\mapsto P_{X|Y}(y) \end{aligned}$$

be the orthogonal projection into X restricted to Y. That is, for any $y \in Y$, $P_{X|Y}(y)$ is the unique element $x \in X$ such that

$$\langle y - x, \tilde{x} \rangle = 0, \quad \forall \tilde{x} \in X.$$

Lemma 2.3.1. *Let W_m and V_n be an observation space and a reduced basis of dimension $n \leq m$ such that $\beta(V_n, W_m) > 0$. Then the linear PBDW algorithm is given by*

$$u_n^*(\omega) = \omega + v_n^* - P_W v_n^*, \tag{2.17}$$

with

$$v_n^* = \left(P_{V_n|W_m} P_{W_m|V_n}\right)^{-1} P_{V_n|W_m}(\omega). \tag{2.18}$$

Proof. By formula (2.5), $u_n^*(\omega)$ is a minimizer of

$$\begin{aligned}
\min_{u \in \omega + {W_m}^\perp} \operatorname{dist}(u, V_n)^2 &= \min_{u \in \omega + {W_m}^\perp} \min_{v \in V_n} \|u - v\|^2 \\
&= \min_{v \in V_n} \min_{\eta \in {W_m}^\perp} \|\omega + \eta - v\|^2 \\
&= \min_{v \in V_n} \|\omega - v - P_{{W_m}^\perp}(w - v)\|^2 \\
&= \min_{v \in V_n} \|\omega - v + P_{{W_m}^\perp}(v)\|^2 \\
&= \min_{v \in V_n} \|\omega - P_{W_m}(v)\|^2.
\end{aligned}$$

The last minimization problem is a classical least squares optimization. Any minimizer $v_n^* \in V_n$ satisfies the normal equations

$$P^*_{W_m|V_n} P_{W_m|V_n} v_n^* = P^*_{W_m|V_n} w,$$

where $P^*_{W_m|V_n} : V_n \to W_m$ is the adjoint operator of $P_{W_m|V_n}$. Note that $P^*_{W_m|V_n}$ is well defined since $\beta(V_n, W_m) = \min_{v \in V_n} \|P_{W_m|V_n} v\| / \|v\| > 0$, which implies that $P_{W_m|V_n}$ is injective and thus admits an adjoint. Furthermore, since for any $w \in W_m$ and $v \in V_n$, $\langle v, w \rangle = \langle P_{W_m|V_n} v, w \rangle = \langle v, P_{V_n|W_m} w \rangle$, it follows that $P^*_{W_m|V_n} = P_{V_n|W_m}$, which finally yields that the unique solution of the least squares problem is

$$v_n^* = \left(P_{V_n|W_m} P_{W_m|V_n}\right)^{-1} P_{V_n|W_m} w.$$

Therefore $u_n^* = w + \eta_n^* = w + v_n^* - P_{W_m} v_n^*$. □

As a direct consequence of this Lemma, we derive the following formulation for the affine PBDW algorithm.

Corollary 2.3.2. *Let W_m and V_n be an observation space and a reduced basis of dimension $n \leq m$ such that $\beta(V_n, W_m) > 0$. Then the affine PBDW algorithm with respect to a nominal state $\bar{u}$ is given by*

$$A_{m,n}^{(aff)}(\omega) = \bar{u} + u_{m,n}^*(w - \bar{w}) \tag{2.19}$$

where $\bar{w} = P_{W_m}\bar{u}$ and $u_{m,n}^*$ is the reconstruction with the linear PBDW method.

Proof. The result follows by following the same lines as in the linear case for the shifted minimization problem

$$\min_{u \in \omega + W_m^{\perp}} \operatorname{dist}(u, \bar{u} + V_n)^2 = \min_{u \in \omega + W_m^{\perp}} \min_{v \in V_n} \|u - \bar{u} - v\|^2$$

□

We next derive the symmetric linear system of equations to be solved in order to compute v_n^* in expression (2.18). Let F and H be two finite-dimensional spaces of V of dimensions n and m respectively in the Hilbert space V and let $\mathcal{F} = \{f_i\}_{i=1}^n$ and $\mathcal{H} = \{h_i\}_{i=1}^m$ be a basis for each subspace respectively. The Gram matrix associated to $\mathcal{F}$ and $\mathcal{H}$ is

$$\mathbb{G}(\mathcal{F}, \mathcal{H}) = (\langle f_i, h_j \rangle)_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}}.$$

These matrices are useful to express the orthogonal projection $P_{F|H} : H \mapsto F$ in the bases $\mathcal{F}$ and $\mathcal{H}$ in terms of the matrix

$$\mathbb{P}_{F|H} = \mathbb{G}(\mathcal{F}, \mathcal{F})^{-1}\mathbb{G}(\mathcal{F}, \mathcal{H}).$$

As a consequence, if $\mathcal{V}_n = \{v_i\}_{i=1}^n$ is a basis of the space V_n and $\mathcal{W}_m = \{\omega_i\}_{i=1}^m$ is the basis of W_m formed by the Riesz representers of the linear functionals $\{\ell_i\}_{i=1}^m$, the coefficients $\mathbf{v}_n^*$ of the function v_n^* in the basis $\mathcal{V}_n$ are the solution to the normal equations

$$\mathbb{P}_{V_n|W_m}\mathbb{P}_{W_m|V_n}\mathbf{v}_n^* = \mathbb{P}_{V_n|W_m}\mathbb{G}(\mathcal{W}_m, \mathcal{W}_m)^{-1}\mathbf{w}, \tag{2.20}$$

where

$$\mathbb{P}_{V_n|W_m} = \mathbb{P}_{V_n|W_m}^T$$

since $P_{W_m|V_n}^* = P_{W_m|V_n}$ and $\mathbf{w}$ is the vector of measurement observations

$$\mathbf{w} = (\langle u, \omega_i \rangle)_{i=1}^m.$$

Usually $\mathbf{v}_n^*$ is computed with a QR decomposition or any other suitable method. Once $\mathbf{v}_n^*$ is found, the vector of coefficients $\mathbf{u}_n^*$ of u_n^* easily follows.

In our application, the manifold $\mathcal{M}$ will be a family of incompressible fluid flow solutions of a parametric incompressible Navier-Stokes equation. In this case, V_n is usually built such that all functions $v \in V_n$ satisfy the divergence-free condition $\nabla \cdot v = 0$ As a result, a reconstruction with only $v^*_{m,n}$ from the normal equations 2.20 will yield a divergence-free approximation of the flux. Note however that the full PBDW reconstruction $u^*_{m,n}$ does not guarantee this property since the model bias corrector $\omega - P_{W_m} v^*_{m,n}$ may not be divergence free. The mass conservation of the reconstruction could be enforced, for instance, by considering its projection on divergence-free fields. That is to say, let us take a reconstructed vector field $u^*_{m,n}$. A Helmholtz decomposition leads to the vector field $v_{\text{Helmholtz}}$ and an scalar field $\phi_{\text{Helmholtz}}$ such that $u^*_{m,n} = \nabla \times v_{\text{Helmholtz}} - \nabla \phi_{\text{Helmholtz}}$. One might solve therefore the following Laplacian problem: Find $\phi_{\text{Helmholtz}} \in H^1(\Omega)$ such that:

$$\begin{cases} -\Delta \phi_{\text{Helmholtz}} = \nabla \cdot u^*_{m,n} & \text{in } \Omega, \\ \phi_{\text{Helmholtz}} = 0 & \text{on } \Gamma_{\text{in}}, \\ \nabla \phi_{\text{Helmholtz}} \cdot n = 0 & \text{on } \Gamma_{\text{out}^1}, \\ \nabla \phi_{\text{Helmholtz}} \cdot n = 0 & \text{on } \Gamma_{\text{out}^2}, \\ \nabla \phi_{\text{Helmholtz}} \cdot n = 0 & \text{on } \Gamma_{\text{w}}. \end{cases} \tag{2.21}$$

The solenoidal field is thus $u^*_{m,n} + \nabla \phi_{\text{Helmholtz}}$. Notice that the inlet boundary condition is arbitrary but nonetheless it does not influence the results since we are interested on the gradient of the field.

2.3.2 Practical implementation of OMP

Introducing the set $\mathcal{W} = \{P_{W_m} v \ : \ v \in \mathcal{D}\}$, the recursive step of the OMP algorithm (2.16) can equivalently written in terms of functions of W_m as follows. For $n > 1$, given $V_n = \text{span}\{\varphi_1, \ldots, \varphi_n\}$, we select (see equation (2.16))

$$z_{n+1} \in \arg\max_{z \in \mathcal{W}} \left| \left\langle \omega - P_{P_{W_m}(V_n)} \omega, \frac{z}{\|z\|} \right\rangle \right|. \tag{2.22}$$

For the chosen z_{n+1}, we take one of the corresponding functions φ_{n+1} from $\mathcal{D}$ that satisfy $P_{W_m} \varphi_{n+1} = z_{n+1}$. We then set $V_{n+1} = \text{span}\{V_n, \varphi_{n+1}\}$. The computation of $P_{P_{W_m} V_n} \omega$ is a straightforward solving of a linear problem. We look for the coefficients $c = (c_i)_{i=1}^n$ such that $P_{P_{W_m}(V_n)} \omega = \sum_{i=1}^n c_i P_{W_m} \varphi_i$. Since

$$\langle P_{P_{W_m}(V_n)} \omega, P_{W_m} \varphi_i \rangle = \langle \omega, P_{W_m} \varphi_i \rangle, \quad \forall i = 1, \ldots, n \tag{2.23}$$

it follows that

$$\sum_{j=1}^n c_j \langle P_{W_m} \varphi_i, P_{W_m} \varphi_j \rangle = \langle \omega, P_{W_m} \varphi_i \rangle, \quad 1 \leq i \leq n,$$

which is a linear system of the form

$$A^{\mathrm{OMP}} c = g^{\mathrm{OMP}}. \tag{2.24}$$

Denoting $\{\mathrm{w}_1, \ldots, \mathrm{w}_\mathrm{m}\}$ an orthonormal basis of W_m, we have

$$A^{\mathrm{OMP}} = (a_{i,j})_{1\le i,j\le n}, \quad a_{i,j} = \langle P_{W_m}\varphi_i, P_{W_m}\varphi_j\rangle = \sum_{k=1}^{m} \langle \mathrm{w_k}, \varphi_\mathrm{j}\rangle \langle \mathrm{w_k}, \varphi_\mathrm{i}\rangle$$

and

$$g^{\mathrm{OMP}} = (g_i)_{i=1}^{n}, \quad g_i = \langle \omega, P_{W_m}\varphi_i\rangle = \sum_{k=1}^{m} \langle u, \mathrm{w_k}\rangle \langle \varphi_i, \mathrm{w_k}\rangle .$$

2.3.3 Summary of the methods investigated and their computational cost.

Methods: The methods that we have implemented in our numerical tests and their computational cost are:

1. *Linear PBDW (labelled* POD-lin*):* We build the linear spaces V_n from a classical Proper Orthogonal Decomposition of the whole training set $\widetilde{\mathcal{M}}_{\mathrm{train}}$. This is the classical PBDW approach explained in section 2.2.2.1 and we consider it as the reference that our proposed methods should outperform.

2. *Nonlinear algorithm with manifold partitioning:* We partition the training set $\widetilde{\mathcal{M}}_{\mathrm{train}}$ in a number of non-intersecting subsets (see section 2.2.2.2). For each subset $\widetilde{\mathcal{M}}^{(k)}$ of the partition, we build in an offline phase a reduced model. Two constructions have been tested:

   - *P-POD-aff*: A Proper Orthogonal Decomposition on each partition $\widetilde{\mathcal{M}}^{(k)}$.
   - *P-Greedy-aff*: A greedy algorithm: for $n = 1$, we set $V_1^{(k)} = \mathrm{span}\{u_1^{(k)}\}$ with

   $$u_1^{(k)} = \frac{1}{\#\widetilde{\mathcal{M}}^{(k)}} \sum_{u\in\widetilde{\mathcal{M}}^{(k)}} u.$$

   For $n > 1$, we select

   $$u_n^{(k)} \in \arg\max_{u\in\widetilde{\mathcal{M}}^{(k)}} \|u - P_{V_{n-1}} u\|, \tag{2.25}$$

   and set $V_n^{(k)} = \mathrm{span}\{V_{n-1}^{(k)}, u_n^{(k)}\}$.

3. *P-DB-aff, Data-driven nonlinear algorithm with manifold partitioning:* Since each ultrasound image can be seen as an observation $\omega \in W_m$, we run the OMP algorithm of section 2.2.2.2 to build $V_n(\omega)$ and do the reconstruction. Note that the greedy search has to be

Table 2.1: Computational cost of each method. First two rows corresponds to computations during offline phase, where as the reconstruction row refers to the computational costs during online phase.

Cost - Method	*POD-lin*	*P-POD-aff*	*P-Greedy-aff*	*P-DB-aff*
Database const.	$\mathcal{O}(\mathcal{N}^\alpha \# \widetilde{\mathcal{M}}_{\text{train}})$	$\mathcal{O}(\mathcal{N}^\alpha \# \widetilde{\mathcal{M}}_{\text{train}})$	$\mathcal{O}(\mathcal{N}^\alpha \# \widetilde{\mathcal{M}}_{\text{train}})$	$\mathcal{O}(\mathcal{N}^\alpha \# \widetilde{\mathcal{D}}_{\text{train}})$
MOR	$\mathcal{O}(\mathcal{N} \# \widetilde{\mathcal{M}}^2_{\text{train}})$	$\mathcal{O}(\mathcal{N}(\# \widetilde{\mathcal{M}}^{(k)}_{\text{train}})^2)$	$\mathcal{O}(n\mathcal{N} \# \widetilde{\mathcal{M}}^{(k)}_{\text{train}})$	$\mathcal{O}(\mathcal{N} \# \widetilde{\mathcal{D}}^{(k)}_{\text{train}} + m^3)$
Reconstruction	$\mathcal{O}(n^2 + n\mathcal{N})$	$\mathcal{O}(n^2 + n\mathcal{N})$	$\mathcal{O}(n^2 + n\mathcal{N})$	$\mathcal{O}(n^2 + nm \# \widetilde{\mathcal{D}}^{(k)}_{\text{train}} + n\mathcal{N})$

done online since we need the knowledge of the measurement. To speed-up computations, instead of searching in the whole training set

$$\delta_{\bar{u}} \widetilde{\mathcal{D}} = \left\{ v = \frac{u - \bar{u}}{\|u - \bar{u}\|} \ : \ u \in \widetilde{\mathcal{M}}_{\text{train}} \right\},$$

we restrict the search to the partition

$$\delta_{\bar{u}} \widetilde{\mathcal{D}}^{(k)} = \left\{ v = \frac{u - \bar{u}}{\|u - \bar{u}\|} \ : \ u \in \widetilde{\mathcal{M}}^{(k)} \right\}.$$

Computational costs: We outline the computational cost of the methods in Table 2.1. Note that all of them can be decomposed into an online and an offline phase. The cost of generating the snapshots database is of the same order in all methods, namely $\mathcal{O}(\mathcal{N}^\alpha \# \widetilde{\mathcal{M}}_{\text{train}})$ where α is the scaling behavior of the linear solver (usually $\alpha = 2$ for iterative methods). The cost of building the reduced model in *POD-lin* and *P-POD-aff* is the one to compute the SVD of the correlation matrix of the snapshots. For *POD-lin*, the matrix is of size $\# \widetilde{\mathcal{M}}^2_{\text{train}}$. For *P-POD-aff*, we have to compute the SVD of K matrices of size $(\# \widetilde{\mathcal{M}}^{(k)})^2$ for $k = 1 \dots, K$. If these SVD computations are done in parallel, we can obtain important time reductions compared to *POD-lin* since in general $\# \widetilde{\mathcal{M}}^{(k)} \ll \# \widetilde{\mathcal{M}}_{\text{train}}$. Regarding the cost of the online phase, if we store the QR decomposition of normal equations (2.20) (at a cost of $\mathcal{O}(n^3)$ operations, not reported on the table), the first three methods only need to solve the associated linear system, which costs $\mathcal{O}(n^2)$ operations. In the case of *P-DB-aff*, we additionally have to run a greedy algorithm in $\mathbb{R}^m$ as explained in section 2.3.2, hence the additional cost reported in the table. Note however that the time required for this extra step is not significantly slowing down computations thanks to the fact that we work in $\mathbb{R}^m$ and m is usually moderate. This is in contrast to the greedy algorithm of the offline phase of *P-Greedy-aff*, which takes place in $\mathbb{R}^{\mathcal{N}}$.

2.4 Reconstruction of 3D blood flows from Doppler data

We apply the above described methodology to reconstruct a 3D blood velocity field in $L^2(\Omega)$ on a human carotid artery from Doppler ultrasound images. The images are synthetically generated

and the use of data from real patients is deferred to a future work. The main goal of the tests is twofold:

1. The first goal is to compare the different strategies to construct the space V_n and the recovery algorithms. The method that can be considered a sort of a baseline to be compared to is the POD.

2. Second, we aim at assessing the ability of these recovery strategies to estimate the velocity in a semi-realistic idealised setting. Methods failing in achieving a 10% accuracy on the peak velocity in this setting should not be used in more realistic scenarios.

The section is organized as follows. First, we present the parameter-dependent model that will define the manifold $\mathcal{M}$ on which we will rely to compute different reduced models. Second, we explain how to define a measurement space W_m from a Doppler velocity image. Finally, we present results on the comparison of the different methods based on their reconstruction performance.

2.4.1 Incompressible Navier-Stokes equations

Let Ω be a spatial bounded domain of $\mathbb{R}^3$ with the shape of a human carotid artery as given in figure 2.3. The boundary $\Gamma \overset{\text{def}}{=} \partial\Omega$ is the union of the inlet part Γ_{in} where the blood is entering the domain, the outlets $\Gamma_{o,1}$ and $\Gamma_{o,2}$ where the blood is exiting the domain after a bifurcation, and the walls Γ_{w}.

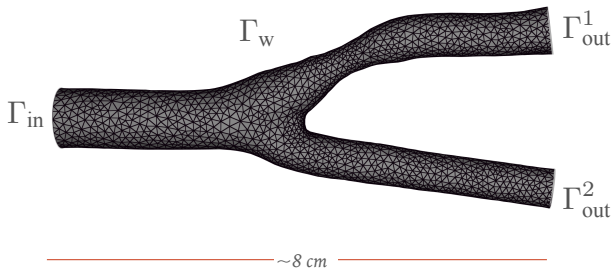


Figure 2.3: Domain Ω used in the simulations. Note the small stenosis in the upper part of the bifurcation.

We recall the Navier-Stokes equations already introduced in section 5.3 on Ω and over the time interval $[0, T]$ for $T > 0$. For a fluid with density $\rho \in \mathbb{R}^+$ and dynamic viscosity $\mu \in \mathbb{R}^+$,

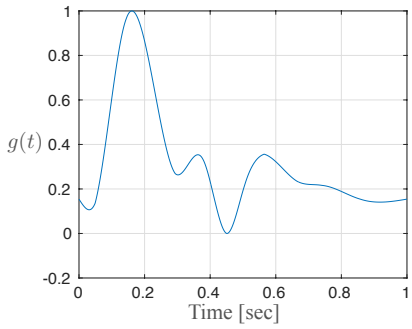


Figure 2.4: Geometry and inlet boundary function of the test case.

we search for the velocity $u \in [H^1(\Omega)]^3$ and pressure $p \in L^2(\Omega)$ that:

$$\begin{cases} \rho \dfrac{\partial u}{\partial t} + \rho\ u\nabla u - \mu\Delta u + \nabla p = 0, & \text{in } \Omega \\ \nabla \cdot u = 0, & \text{in } \Omega. \end{cases} \tag{2.26}$$

These equations are closed by adding a zero initial condition and the following boundary conditions:

1. No-slip condition for the vessel wall, that is, $u = (0, 0, 0)^T$ on Γ_{w}.

2. The inlet boundary Γ_i lies in the xz plane and we apply a Dirichlet condition $u = [0, u_{\text{in}}, 0]^T$. The component b is a function $u_{\text{in}}(t, x, z) = u_0\ g(t) f(x, z)$, where:

- $u_0 \in \mathbb{R}^+$ is an scaling factor. The function $g(t)$ is built by using flow data in the common carotid area taken from [Blanco et al., 2015]. Its behavior is given in figure 2.4.1.

- The function f is a 2D logit-normal distribution

$$f(x) = \frac{1}{x(1-x)z(1-z)} \exp\left\{-0.5\left(\log\left(\frac{x}{1-x}\right) - s\right)^2 - 0.5\left(\log\left(\frac{z}{1-z}\right)\right)^2\right\}, \tag{2.27}$$

  where the parameter, $s \in \mathbb{R}^+$, controls the axial symmetry of the inlet flow.

3. For the outlet boundaries Γ^1_{out} and Γ^2_{out}, we use the 0D model introduced in section 5.4.2. The average pressure over each Γ^k_{out} is computed as follows,

$$\bar{p}_{o,k} = p^k_d + R^k_p \int_{\Gamma^k_{\text{out}}} u \cdot n \ \text{d}s, \quad k = 1,\ 2$$

   where $p^k_d \in \mathbb{R}$ are the solutions to the ordinary differential equation:

$$\begin{cases} C^k_d \dfrac{dp^k_d}{dt} + \dfrac{p^k_d}{R_d} & = \int_{\Gamma^k_{\text{out}}} u \cdot n \ \text{d}s \\ p^k_d(t = 0) & = p_{d,k} \text{ given.} \end{cases} \tag{2.28}$$

where $C_d^k \in \mathbb{R}$, $R_p^k \in \mathbb{R}$ and $R_d^k \in \mathbb{R}$.

Spatial discretization of the carotid geometry leads to a tetrahedron mesh of 42659 vertices and 225196 tetrahedra. The meshing process is in charge of MMG [Dapogny et al., 2014]. Time is discretized with a semi-implicit backward Euler scheme with time-step $\delta t = 2 \cdot 10^{-3} s$, which means that the convective term in the Navier-Stokes equations is written with the velocity explicitly and the velocity gradient implicitly in order to circumvent the non-linearity of the system. An explicit scheme is used to numerically solve the ODE on the distal pressure in the Windkessel model. In addition, a backflow stabilization is added in order to address potential instabilities in the outlet boundaries (see, e.g., [Bertoglio et al., 2017a]). Data visualization is done by using ParaView [Ayachit, 2015] and Vizir [Loseille and Feuillet, 2018]. Standard SUPG stabilization for convection dominated flows is used [Brooks and Hughes, 1982].

The coupled problem for velocity and pressure is discretized using $\mathbb{P}_1 - \mathbb{P}_1$ Lagrange elements, with a monolithic approach. In order to avoid the inf-sup constraint imposed by the saddle point nature of the problem we use the Brezzi-Pitkaranta stabilization approach (for further details, we refer to [Ern and Guermond, 2013] and [Brezzi and Pitkaranta, 1984]), that basically perturbs the weak form of the mass conservation equation with a *stiffness-like* term that scales with the square of the elements size. Each forward simulation has $\mathcal{N} = 127977$ degrees of freedom at each time step. The linear systems are solved by means of a GMRES [Saad and Schultz, 1986] method with additive Schwarz preconditioning.

Note that one could use more sophisticated models involving, for instance, fluid-structure interactions or more refined Winkessel models for the pressure. Our model is a trade-off between its degree of realism and the difficulty and time to solve it. We refer to [Formaggia et al., 2009] for a detailed overview of cardiovascular modeling.

Let us define the manifold of solutions that we consider in our numerical experiments. We set the following coefficients to a fixed value

$$\begin{cases} \rho & = 1 \text{ g/cm}^3 \\ \mu & = 0.03 \text{ Poise} \\ C_d^k & = 1.6 \times 10^{-5} \text{ for } k = 1, 2 \\ R_p^k & = 7501.5 \text{ for } k = 1, 2 \\ p_d^k & = 1.06 \times 10^5 \text{ for } k = 1, 2 \\ R_d^1 & = 60012 \end{cases} \tag{2.29}$$

The ratio of the distal resistances for the Windkessel model at the outlets is introduced:

$$\eta \overset{\text{def}}{=} R_d^1 / R_d^2 = 60012 / R_d^2.$$

This parameter plays an important role since it impacts on how the blood flow splits between the two branches. When $\eta \to 0$ or ∞, one branch is obstructed and the blood tends to flow through the other branch. In the following, we call this situation an arterial blockage. When $\eta \approx 1$, the flow splits more or less equally and there is no blockage.

We define the heart rate as the number of cardiac cycles per minute, that is,

$$\mathrm{HR} \stackrel{\text{def}}{=} 60/T_c,$$

where $T_c > 0$ is the cardiac cycle duration expressed in seconds. We have $T_c = T_{sys} + T_{dia}$, where T_{sys} and T_{dia} are the duration of the systole and diastole respectively.

The manifold $\mathcal{M}$ is generated by the variations of the following six parameters

$$\begin{cases} t & \in [0, T] \\ \mathrm{HR} & \in [48, 120] \\ s & \in [0, 0.2] \\ T_{sys} & \in [0.2863, 0.3182] \text{ s.} \\ u_0 & \in [17, 20] \text{ cm/s} \\ \eta & \in [0.05, 0.2] \cup [0.5, 1.5] \cup [5, 20] \end{cases} \tag{2.30}$$

Note that the time t is also seen as a parameter. The parameter set is thus

$$\mathrm{Y} = \{(t, \mathrm{HR}, s, T_{sys}, u_0, \eta) \in \mathbb{R}^6 \,:\, t \in [0, T],\, \mathrm{HR} \in [48, 120],\, s \in [0, 0.2],\, \dots\} \subset \mathbb{R}^6$$

and the manifold of solutions is

$$\mathcal{M} \stackrel{\text{def}}{=} \{u(y) \in [H^1(\Omega)]^3 \,:\, y \in \mathrm{Y}\}.$$

At this point, several comments are in order:

- Note that we only consider velocity fields since in the present work we are only concerned by the reconstruction of the blood flow velocity. The reconstruction of other quantities of interest, such as the pressure, will be addressed in a forthcoming work.

- For each $y \in \mathrm{Y}$, the velocity $u(y)$ is a function of $[H^1(\Omega)]^3$. In the following, we will view it as a function from

  $$V \stackrel{\text{def}}{=} \mathcal{L} = [L^2(\Omega)]^3,$$

  which, endowed with the inner product,

  $$\langle (v_1, v_2, v_3), (w_1, w_2, w_3) \rangle \stackrel{\text{def}}{=} \sum_{i=1}^{3} \langle v_i, w_i \rangle_{L^2(\Omega)}, \quad \forall (v, w) \in [L^2(\Omega)]^3,$$

  defines a Hilbert space.

- Since the time variable has been included as a parameter, a simple way to build nonlinear reduced models is to set a window parameter $\tau > 0$ and consider the subset $\mathcal{M}^{(k)} = \mathcal{M}_{[t_k-\tau,t_k+\tau]} \subset \mathcal{M}$, where t is restricted to the interval $[t_k - \tau, t_k + \tau]$ of size 2τ centered around a given time t_k. We can then build reduced models to reconstruct this specific time interval. As we will see in the numerical experiments, this strategy is very effective in our problem because the velocity presents two regimes given by the systole and diastole periods.

- The computation of reduced models involves a discrete training subset $\widetilde{\mathcal{M}}_{\text{train}} \subset \mathcal{M}_h$ which, in the experiments below, involves $\#\widetilde{\mathcal{M}}_{\text{train}} = 78528$ snapshots $u(y)$. The parameters are chosen from a uniform random distribution and we only save the solutions during the second cardiac cycle of each simulation in order to capture a flow behavior that is close to a realistic periodic regime.

- For the purposes of illustrating the potential of the method for diagnoses, we introduce a notion of sickness in terms of the arterial blockage in the following way.

  Definition 3 (Sick patient)**.** We say that the output of the simulation corresponds to a healthy patient when $\eta \in [0.5, 1.5]$. Outside of this range, simulations correspond to sick patients.

  We thus have the partition

$$\mathcal{M} = \mathcal{M}_{\text{healthy}} \cup \mathcal{M}_{\text{sick}}, \qquad \mathcal{M}_{\text{healthy}} \cap \mathcal{M}_{\text{sick}} = \emptyset$$

  with $\mathcal{M}_{\text{healthy}} \overset{\text{def}}{=} \mathcal{M}_{\eta\in[0.5,1.5]}$ and $\mathcal{M}_{\text{sick}} = \mathcal{M}_{\eta\in[0.05,0.2]\cup[5,20]}$.

2.4.2 Doppler data in the context of reconstruction algorithms

We have assumed a strong hypothesis up to here, which is that we account with m linear functionals to model the image. Every scenario in which the aforementioned methodology is proposed to be used should be examined carefully in order to understand this point of uttermost importance. The question that follows is, is this a valid assumption in our application? The fundamental aspects of the ultrasound image acquisition must be reviewed in order to provide a solid argument behind this hypothesis, and validate the proximity of this reconstruction tool to the medical practice. We refer to section 1.3 to support the working assumptions. To summarize, in a regular partition of $[0, T]$, we are given Doppler ultrasound images that contain information on the blood velocity on a subdomain of the carotid. For typical ultrasound machines with acquisition time of 0.1 milliseconds, a CFI is built with, for instance, 32 consecutive B-mode frames, which leads to a measurement sampling of 32 milliseconds. From the image, the

observations $\ell_i(u)$ are extracted and used to build a complete time-dependent 3D reconstruction of the blood velocity in the whole carotid Ω.

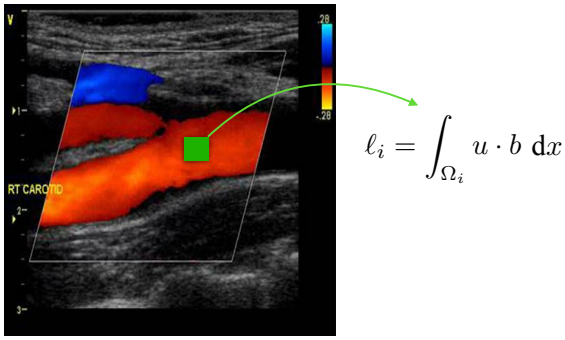


Figure 2.5: Schematic way of looking at the CFI measures as a set of elements in V^*.

In the following, we work with a model of CFI images based on the ultrasound physics shown in section 1.3. For each time t, a given image is a local average in space of the velocity projected into the direction in which the ultrasound probe is steered. More specifically, we consider a partition of $\Omega = \cup_{i=1}^m \Omega_i$ into m disjoint subdomains (voxels) Ω_i. Then, from each color flow image (CFI) image we collect

$$\ell_i(u) = \int_{\Omega_i} u \cdot b \, \mathrm{d}x, \quad 1 \leq i \leq m, \tag{2.31}$$

where $b \in \mathbb{R}^3$ is a unitary vector giving the direction of the ultrasound beam. From (2.31), it follows that the Riesz representers of the ℓ_i in V are simply

$$\omega_i = \chi_i b,$$

where χ_i denotes the characteristic function of the set Ω_i. Thus the measurement space is

$$W_m = W_m{}^{(CFI)} \stackrel{\text{def}}{=} \operatorname{span}\{\omega_i\}_{i=1}^m.$$

Since the voxels Ω_i are disjoint from each other, the functions $\{\omega_i\}_{i=1}^m$ are orthogonal and therefore having a CFI image is equivalent to having

$$\omega = \mathbb{P}_{W_m} u = \sum_{i=1}^m \langle \omega_i, u \rangle \omega_i = \sum_{i=1}^m \ell_i(u) \omega_i. \tag{2.32}$$

Remark 2.4.1. *The case of vector flow images (VFI, see figure 1.7(b)) can be treated similarly. This imaging mode gives $2m$ measurements*

$$\ell_i(u) = \int_{\Omega_i} u \cdot b \, \mathrm{d}x, \quad 1 \leq i \leq m,$$

and

$$\ell_{m+i}(u) = \int_{\Omega_i} u \cdot b_\perp \, \mathrm{d}x, \quad 1 \leq i \leq m,$$

where b is the unitary vector giving the direction of the ultrasound beam and $b_\perp$ is the unitary vector perpendicular to it and contained in the image plane. Therefore, ${W_m}^{(VFI)} = {W_m}^{(CFI)} \oplus \mathrm{span}\{\chi_{\Omega_i} b_\perp\}_{i=1}^m$ which is a space of dimension $2m$. This clearly shows that the additional direction $b_\perp$ enriches the quality of the measurements in the sense that for any $u \in V$, the approximation error $\|u - P_{W_m} u\|$ will be smaller with the VFI mode than with the CFI one. As a result, the CFI mode which we consider in our examples is a more challenging case since the measurements contain less information.

2.4.3 Reconstruction on a first example with healthy patients

To validate our method, we first consider a simple example where we only work with healthy patients, so the manifold is $\mathcal{M}_{\text{healthy}}$. One color flow image contains information of the velocity averaged over 552 voxels. As explained in section 2.4.2, this sets the dimension of the observations space to $m = 552$ (whereas for VFI, $m = 1104$). As a result, each image can be seen as an observation $\omega \in W_m$. The dimension $m = 552$ may seem quite large but it is representative of the one provided by modern pulsated ultrasound devices. 297 healthy patients are simulated in order to build the training set. For each one of them a number of snapshots containing the second cardiac cycle is stored so there are multiple snapshots per patient due to time marching.

In figure 2.6, the synthetic image is depicted. The ultrasound device is placed in such a way we measure a mid plane of the working domain. The ultrasound wave forms an angle of $\pi/4$ respect to the dominant fluid direction in the main carotid branch, that is to say, $b = \left[\sqrt{2}/2, \sqrt{2}/2, 0\right]$.

This leads to a training set composed of $\#\widetilde{\mathcal{M}}_{\text{train}} = 56383$ snapshots. The performance of the algorithms is tested on a test set $\widetilde{\mathcal{M}}_{\text{test}}$ of 32 healthy patients each one with a parameter configuration in range but different from those inside $\widetilde{\mathcal{M}}_{\text{train}}$.

For the comparison, we implement the algorithms outlined in section 2.3.3. We recall them below and give specific details related to the current test case:

1. *Linear PBDW:* The space V_n is built from a classical POD of the whole training set $\widetilde{\mathcal{M}}_{\text{train}}$.

2. *Nonlinear algorithm with manifold partitioning:* In real medical examinations, the heart rate HR of the patient and the time t in which the ultrasound image is taken are known. We exploit this fact to decompose $\widetilde{\mathcal{M}}_{\text{train}}$ into $K = IJ$ subsets

$$\widetilde{\mathcal{M}}_{\text{train}} = \bigcup_{\substack{k=(i,j) \\ (i,j)\in\{1,\dots,I\}\times\{1,\dots,J\}}} \widetilde{\mathcal{M}}^{(k)}, \tag{2.33}$$

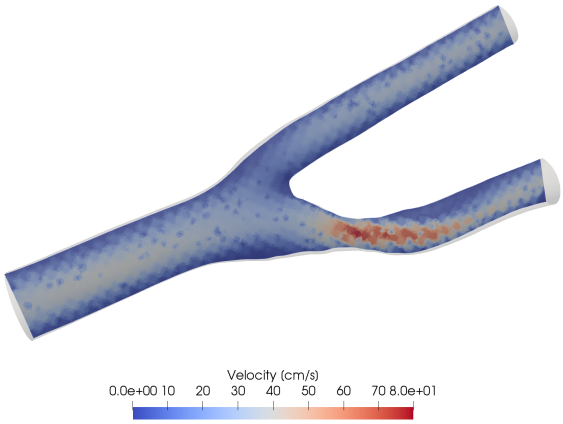


Figure 2.6: Example of synthetic CFI measures used in first example. The image leads to an space W_m of dimension $m = 552$

where, for each $k = (i, j)$ in $\{1, \ldots, I\} \times \{1, \ldots, J\}$,

$$\widetilde{\mathcal{M}}^{(k)} = \{u \in \widetilde{\mathcal{M}}_{\text{train}} : t \in [t_i - \tau, t_i + \tau],\ \text{HR} \in [\text{HR}_j - \delta_{\text{HR}}, \text{HR}_j + \delta_{\text{HR}}]\}. \tag{2.34}$$

Among the possibilities to build the manifold partitioning, our approach is driven by the reconstruction error of the elements inside $\widetilde{\mathcal{M}}_{\text{test}}$. That is to say, we have computed the reconstruction u_i^* of u_i for $i = 1, \ldots, \#\widetilde{\mathcal{M}}_{\text{test}}$ for a set of couples $(\delta_{\text{HR}}, \tau)$ with a fixed dimension for the reduced model ($n = 30$). Hence, we can pick the optimal values in this sense:

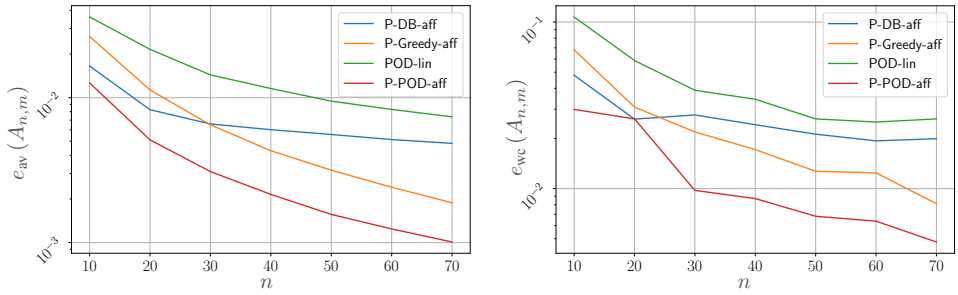
$$(\delta_{\text{HR}}^*, \tau^*) = \underset{\delta_{\text{HR}}, \tau}{\arg\min} \left(\frac{1}{\#\widetilde{\mathcal{M}}_{\text{test}}} \sum_{i=1}^{\#\widetilde{\mathcal{M}}_{\text{test}}} \frac{\| u_i - u_i^*(\delta_{\text{HR}}, \tau) \|}{\|u_i\|} \right)$$

Recalling that the cardiac cycle duration is denoted by T_c, we conclude in this experiment to the sizes $\tau = T_c/10$ and $\delta_{\text{HR}} = 5$ beats per minute.

For each subset $\widetilde{\mathcal{M}}^{(k)}$, a reduced space is built. Two constructions have been tested:

- POD of $\widetilde{\mathcal{M}}^{(k)}$. We call this approach *P-POD-aff*, Partitioned-POD-affine, in the plots.
- A greedy algorithm. We call this approach *P-Greedy-aff*, Partition-Greedy-affine, in the plots

During the online reconstruction, given t and HR, we select the appropriate subset $\widetilde{\mathcal{M}}^{(k)}$ that includes t and HR and reconstruct with a linear PBDW with the reduced model corresponding to $\widetilde{\mathcal{M}}^{(k)}$.



(a) Average error

(b) Worst error

Figure 2.7: Benchmark for first numerical example. The accuracy is evaluated using the error (2.35)

3. *Data-driven nonlinear algorithm with manifold partitioning:* This approach is labeled in the plots as *P-DB-aff*, Partitioned-Data-Based-affine. Since each CFI image can be seen as an observation $\omega \in W_m$, we run the Orthogonal Matching Pursuit algorithm of section 2.2.2.2 to build $V_n(\omega)$ and do the reconstruction. Note that the greedy search has to be done online since we need the knowledge of the measurement.

For each state $u \in \widetilde{\mathcal{M}}_{\mathrm{test}}$, we compute the relative error:

$$e(u, A_{n,m}) = \frac{\|u - A_{m,n}(P_{W_m} u)\|}{\|u\|} \tag{2.35}$$

where $A_{m,n}(P_{W_m} u)$ denotes any of the above four reconstruction algorithms. A patient in the training set is represented by a sequence of simulated states during the second cardiac cycle. This raises the interest in evaluating the reconstruction quality by looking at the following relative error in time:

$$e(t, u, A_{m,n}) = \frac{\|u(t) - A_{m,n}(P_{W_m} u(t))\|}{\left(\int_{T_c}^{2T_c} \|u\|^2 \mathrm{d}t\right)^{1/2}} \tag{2.36}$$

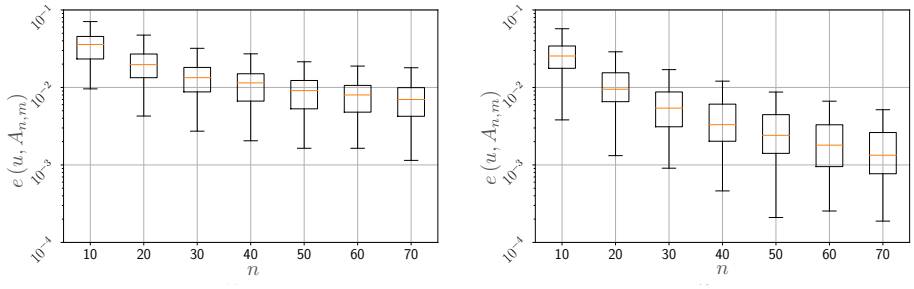
where we normalize by the total energy in the cardiac cycle $\left(\int_{T_c}^{2T_c} \|u(t)\|^2 \mathrm{d}t\right)^{1/2}$.

Figures 2.7(a) and 2.7(b) give the average and worst case performance of the four methods,

$$e_{\mathrm{av}}(A_{n,m}) = \frac{1}{\#\widetilde{\mathcal{M}}_{\mathrm{test}}} \sum_{u \in \widetilde{\mathcal{M}}_{\mathrm{test}}} e(u, A_{n,m}), \quad e_{\mathrm{wc}}(A_{n,m}) = \max_{u \in \widetilde{\mathcal{M}}_{\mathrm{test}}} e(u, A_{n,m}).$$

Note that as n increases, the error decreases for all methods, except perhaps for the *P-DB-aff* approach where the error tends to stagnate for large values of n. This could be due to the fact that *P-DB-aff* heavily relies on the measurement information, which, in the present application, might not deliver enough information to learn reduced models $V_n(\omega)$ that improve

the accuracy as n grows. For instance, the Doppler measurements are close to zero in the diastole of the cardiac cycle so the information that they provide may be insufficient to build a good $V_n(\omega)$. We also see from the figure that the nonlinear method *P-POD-aff* outperforms the rest in the sense that it delivers a given target accuracy with a smaller dimension n of the reduced model. For instance, if we fix a target accuracy on the average performance to 10^{-2}, we see that the *POD-lin* requires 40 modes to achieve it, *P-Greedy-aff* requires 20, *P-DB-aff* requires 17 and *P-POD-aff* requires only 10 (see figure 2.7(a)). Figure 2.8 shows also the error as n changes, but with standard box-plots that give a glance of the low standard deviation in the reconstruction that we register for all the techniques, ensuring there is no *outliers patients* where the reconstruction is not doable.



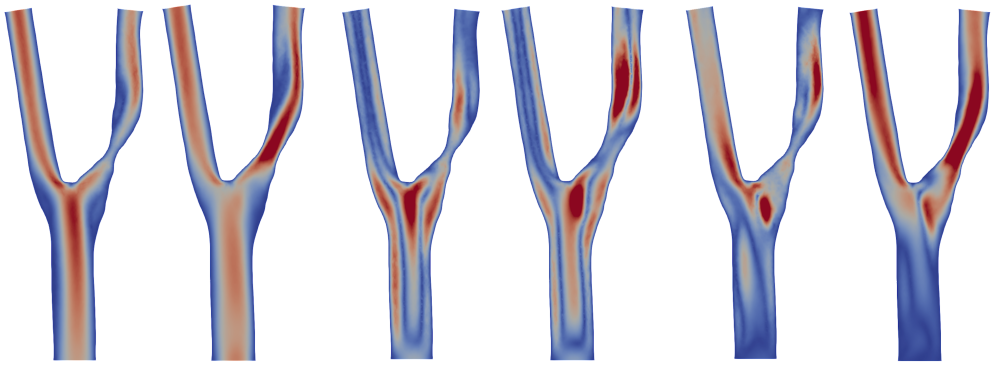
(a) *POD-lin* (b) *P-Greedy-aff*

(c) *P-POD-aff* (d) *P-DB-aff*

Figure 2.8: Reconstruction error $e(u, A_{n,m})$ for patients $u \in \widetilde{\mathcal{M}}_{\text{test}}$ in first example. In the box-plots, data within the first and third quartiles lie in the boxes. The plot also shows the median (in red), maximum and minimum values (outside the box). We observe a regular distribution of the data for the four basis-construction methods. The total amount of snapshots in the test set is 56383.

Let us take a look to the first 3 POD modes for our set of snapshots in figure 2.9. We observe the difference between using a linear space and an affine one. The plot depicted in figure 2.10 shows that using an affine space effectively improves the approximations, but just a bit more

than a marginal amount.



(a) First Mode (b) Second Mode (c) Third Mode

Figure 2.9: First three POD modes. We might notice how they capture the information of the manifold. On the left of each sub-figure we see the *POD-lin* approach, whereas the *POD-aff* one is depicted at the right.

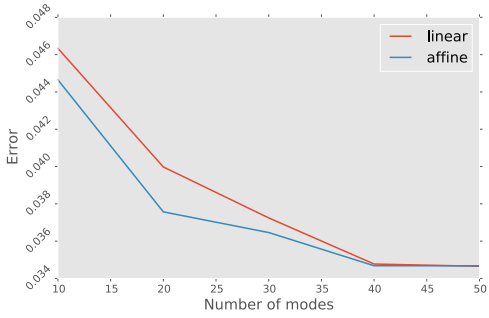


Figure 2.10: Average error as n grows. We observe a small superiority of the affine approach, as was to be expected.

We next fix $n = 30$ and study the error in time $e(t, u, A_{m,30})$ on figure 2.11. We observe that the reconstruction tends to be better during the late diastole phase of the cardiac cycle.

As discussed in section 2.2.2, the inf-sup constant $\beta(V_n, W_m)$ might yield to stability issues when $n \to m$ since its value tends to zero (see equations (2.8) and (2.9)). Figures 2.13(a) and 2.13(b) show its behavior for the four methods during the systole and diastole period. We observe that the four methods perform similarly in terms of stability for the peak systole reconstruction. For the diastole phase, we observe that the inf-sup constant in *P-DB-aff* performs slightly worse than the rest. We think that this could be due to the fact that the measurement space W_m is not rich enough to allow *P-DB-aff* to properly learn reduced models when n becomes large.

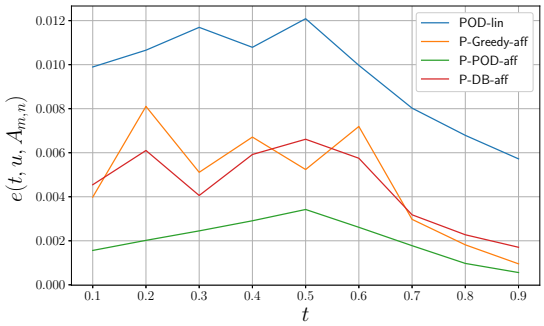


Figure 2.11: Average error comparison for test case described in section 2.4.3. The benchmark shows the temporal evolution of the quantity (2.36) during the cardiac cycle. The dimension of V_n is set to 30.

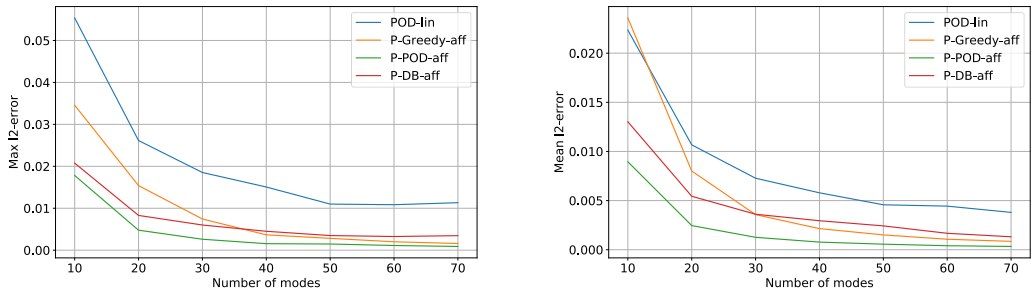
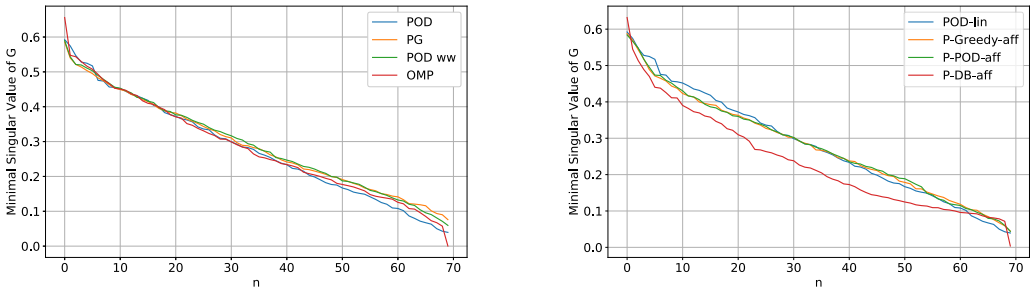


Figure 2.12: Comparison of reconstruction error for peak systole. Max error in the left and mean error in the right.



(a) Peak systole

(b) Late diastole

Figure 2.13: Behavior of $\beta(V_n, W_m)$ as a function of n.

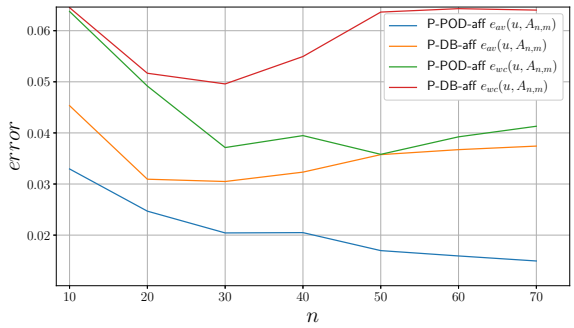


Figure 2.14: Average and worst reconstruction errors as a function of n.

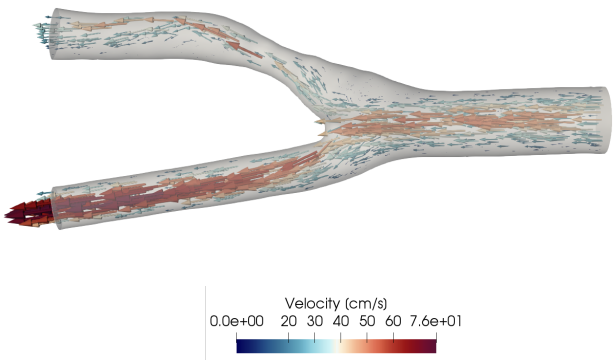
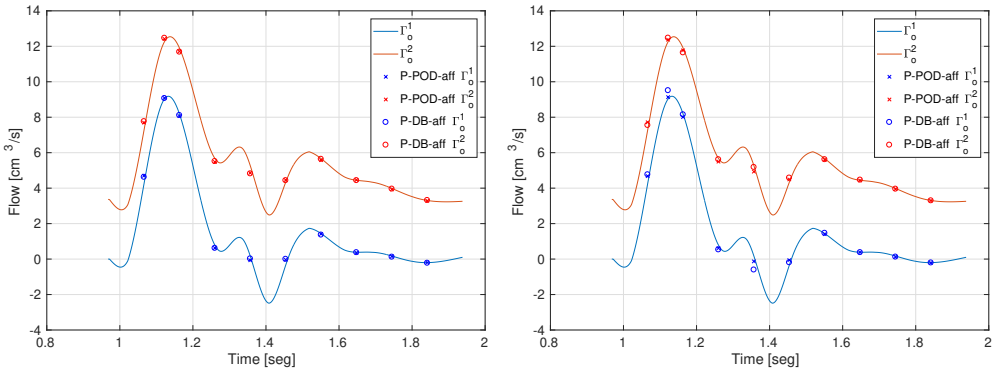
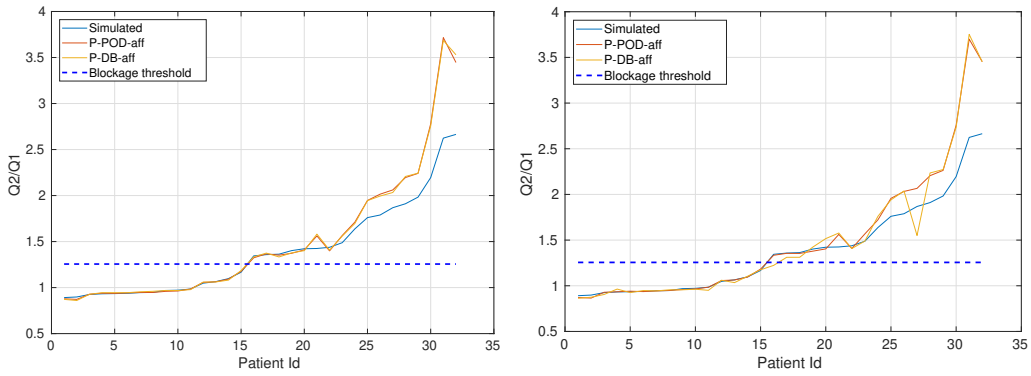


Figure 2.15: Forward simulation and ground truth for one of our test-cases. This field belongs to the data base of patients with blockage.



(a) CFI (b) VFI

Figure 2.16: Comparison of simulated and reconstructed flow at the outlets of the domain.



(a) CFI (b) VFI

Figure 2.17: Flow ratio reconstructed for blockage detection. In this test the number of modes is fixed to 30.

2.4.4 Application to arterial blockage detection

In this example we illustrate that even when the Doppler images do not give information on the whole carotid, we can nevertheless reconstruct the velocity field in the whole domain with our methodology. This is important for actual practice since doctors do not have images in the whole carotid due to morphological constraints. We also show in our example that the method has potential to efficiently estimate in real time relevant quantities of interest.

We illustrate these ideas in the following example: we consider the same setting as before but now the Doppler image does not provide information about the flow in the carotid bifurcations. Therefore, the image does not *see* the flow split in the common carotid downstream (see figure 2.19(a)). In this example we have tested the impact of working with CFI ($m = 233$) or VFI images ($m = 466$).

We train our reconstruction methods on a set $\widetilde{\mathcal{M}}_{\text{train}}$ containing sick and healthy patients. Here, we only work with two of our previous nonlinear algorithms:

1. *P-POD-aff*
2. *P-DB-aff*

Figure 2.14 shows the average and worst case errors

$$e_{\text{av}}(A_{n,m}) = \sum_{u\in\widetilde{\mathcal{M}}_{\text{test}}} e(u, A_{n,m}), \quad e_{\text{wc}}(A_{n,m}) = \max_{u\in\widetilde{\mathcal{M}}_{\text{test}}} e(u, A_{n,m}),$$

as a function of the dimension n of V_n. Like in the previous example, both methods are delivering a very satisfactory accuracy: the average error is below $5\cdot 10^{-2}$ for both methods for all values of n. The method *P-POD-aff* consisting in a partition of the manifold outperforms the data-based one, *P-DB-aff*.

We next show that the method is efficient to assist in the detection of arterial blockages that may cause severe health problems like a stroke. Since a blockage alters the distribution of the velocity field after the bifurcation, a quantity of interest that could serve as a clinical index is the ratio

$$r = \frac{Q_2(t_{peak})}{Q_1(t_{peak})} \tag{2.37}$$

where

$$Q_i(t) \overset{\text{def}}{=} \int_{\Gamma_o^i} u(t)\cdot n \, \mathrm{d}s$$

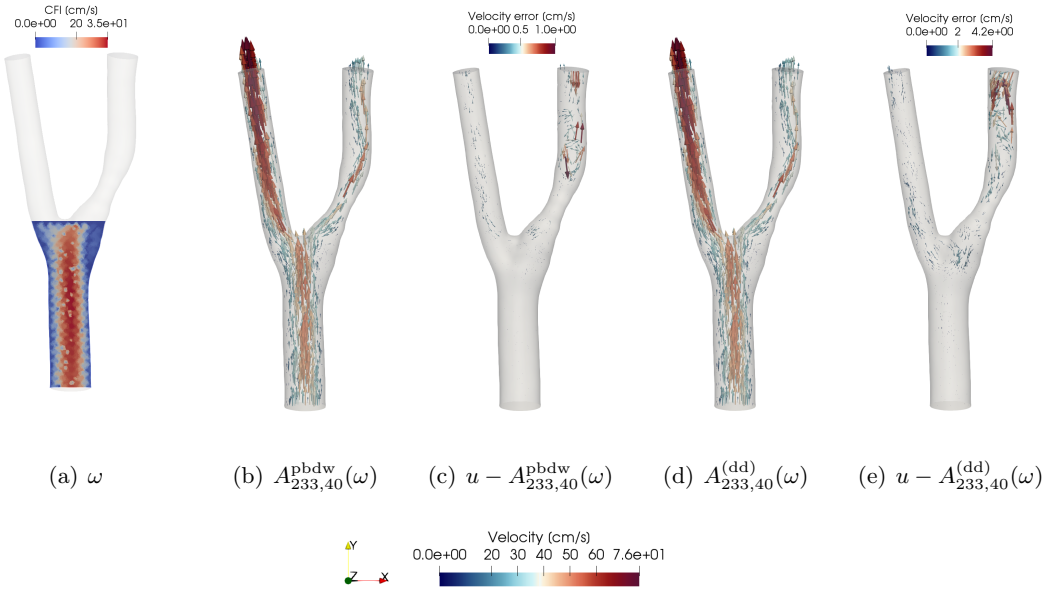
is the blood flow at the outlet Γ_o^i, $i = 1, 2$, and t_{peak} is the peak systole instant. Figure 2.16 shows the evolution of $Q_i(t)$ in time for a sick patient and its approximation with our two reconstruction methods. We observe that, regardless of the image format (CFI or VFI), both methods deliver very satisfactory predictions of the flow.

In figure 2.17, we compare the value of the exact ratio r with the reconstructed one for sick and healthy patients $u \in \widetilde{\mathcal{M}}_{\text{test}}$.

To define a threshold ratio r^* to decide whether the patient has arterial blockage or not, we can take the average of the flow ratios between the healthiest of the patients in the sick group and the sickest of the patients in the healthy group, namely,

$$r^* \overset{\text{def}}{=} \frac{\min_{v\in\widetilde{\mathcal{M}}_{\text{sick}}} r(v) + \max_{u\in\widetilde{\mathcal{M}}_{\text{healthy}}} r(u)}{2} \tag{2.38}$$

where $r(u)$ denotes the flow ratio associated to the velocity field u, as defined in (2.37). In our data-base, we obtain, $r^* = 1.25$, so any patient for which $r > 1.25$ will be considered as presenting high blockage risk. Note that the approximation is very close to the real value for moderate values of r regardless of the image modality. However, we tend to overestimate the



(a) ω (b) $A^{\text{pbdw}}_{233,40}(\omega)$ (c) $u - A^{\text{pbdw}}_{233,40}(\omega)$ (d) $A^{(\text{dd})}_{233,40}(\omega)$ (e) $u - A^{(\text{dd})}_{233,40}(\omega)$

Figure 2.18: Reconstruction example of a CFI partial observation at peak systole for blockage case using 40 modes. No flow information in the carotid branches is observed. Nevertheless, the algorithms are capable of reconstructing the velocity field in the whole computational domain.

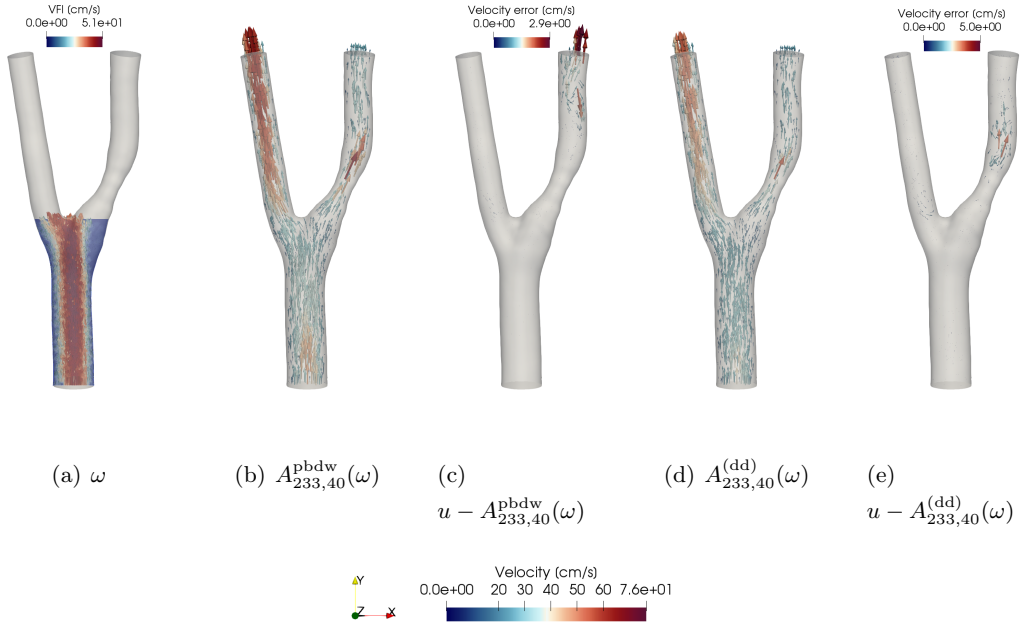
value for $r > 1.7$. In presence of a blockage, r becomes significantly larger than one so the overestimation is by far more preferable than an underestimation. Indeed, the overestimation makes our method conservative and, in the worst case, we will conclude with a false positive. However, the method will not lead to a false negative diagnosis, which would leave a sick patient without treatment/surgery.

To illustrate the reconstruction for both CFI and VFI, let us consider the forward simulation of a peak systole blood flow as depicted in figure 2.15. Assuming this field as a ground truth, we can see the reconstruction with *P-POD-aff* and *P-DB-aff* in figures 2.18 and 2.19.

2.5 Improving the piece-wise linear approximations

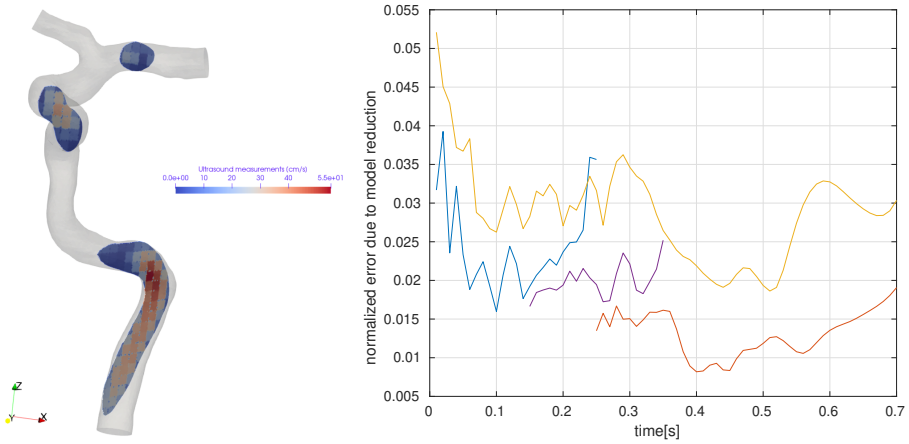
The reader might have noticed that our partitioned methods may bring higher errors at the windows boundaries. This is intuitive since is easy to see that at the boundaries we are maybe neglecting useful information for the reconstruction that can come from the surrounding neighbourhoods.

The strategy to explore is to overlap windows, in such a way that we guarantee enough information for any snapshot we want to reconstruct. We show a preliminary numerical example of velocity reconstruction in a carotid siphon. As an exploratory work, this is part of the



(a) ω (b) $A^{\text{pbdw}}_{233,40}(\omega)$ (c) $u - A^{\text{pbdw}}_{233,40}(\omega)$ (d) $A^{(\text{dd})}_{233,40}(\omega)$ (e) $u - A^{(\text{dd})}_{233,40}(\omega)$

Figure 2.19: Reconstruction example of a VFI partial observation with same information of that of figure 2.18. The method provides a slightly better reconstruction quality than that of the CFI image.



(a) Carotid Siphon Synthetic Measures (b) Projection error

Figure 2.20: Time window overlapping example.

perspectives, and this results are not used in the following chapters.

In figure 2.20.b we see the proper orthogonal decomposition projection error in time for a carotid siphon, as depicted in figure 2.20.a. We observe there two things: The time windowing improves the error, and we see how an overlapped basis in an intermediate time interval compresses useful information for the reconstruction in the late systole phase.

2.6 Conclusions

We have investigated the construction of state estimation techniques with different reduced models. We have compared their accuracy for the problem of reconstructing the blood flow velocity field for which partial information is given by Doppler ultrasound images. The numerical tests were synthetically generated and mimic a real context in medical applications. The results show that the classical linear PBDW method with the classical POD reduced model can be outperformed by other models that are built more specifically for the reconstruction task. In particular, the nonlinear reconstruction method built by manifold partitioning and a local POD basis on each partition outperforms all the other bases choices. It presents a good trade-off between simplicity and efficiency and its accuracy is in some cases even ten times better than the classical PBDW. The data-driven approach gave less performant results than the one with local POD bases but superior to the classical approach. We think that the specific application may have had an important impact on the observed performance. Since we build a space $V_n(\omega)$ from the Doppler measurements ω, they may not deliver enough information to learn a good reduced model, especially in the diastole phase where ω is close to zero. As a result, it seems important to investigate further the performance of this method in other reconstruction problems in order to gain better knowledge on its range of applicability.

In summary, we have set up a solid foundation for the chapters to come, in which the state estimation will be the starting point of a pipeline that will allow us to exploit the observations in order to compute medical quantities of interest.

Chapter 3

Estimation of Haemodynamics Quantities of Interest from Doppler data

---

The content shown in this chapter inspired the following scientific publication: F. Galarce, D. Lombardi and O. Mula. 2020. *Reconstructing Haemodynamics Quantities of Interest from Doppler Ultrasound Imaging*. **International Journal for Numerical Methods in Biomedical Engineering**.

We present a continuation of chapter 2, exploiting the proposed partitioned POD technique further in order to enlarge the possibilities of prediction based on Doppler data. Towards a realistic scenario, we set our ambient space this time to $V = [H^1(\Omega)]^3$. Also, we perturb the synthetic data with white noise.

Beyond the blood flow velocity estimation, we extend the method to estimate quantities that are currently used in the medical practice to perform diagnosis and treatment. Those quantities will be introduced alternating between mathematical and medical frameworks.

The quantities of interest (QoI) are the wall shear stress (WSS), vorticity, flow rate, pressure fields and pressure jumps. For the latter we provide a methodology based also on PBDW, with a theoretical analysis that will come to justify the good numerical results we obtain compared to other state-of-the-art methods.

Contents

3.1 Introduction

In biomedical engineering, most realistic applications have to deal with data assimilation. The problem to be solved consists in providing predictions on Quantities of Interest (QoI) given observations of the system which are often partial and noisy. The present work is a contribution to this topic and focuses on the reconstruction of haemodynamics QoI by exploiting Doppler Ultrasound Imaging.

In several applications related to the cardiovascular system, the QoI to be predicted are:

1. *The complete 3D velocity field* and some quantities associated to it, say, for instance, the maximal velocity or flow rate ([Hata et al., 1987], [Galarce et al., 2020]).

2. *Pressure and pressure drop* ([Hatle et al., 1978], [Hatle et al., 1979], [Hatle et al., 1980], [Mates et al., 1978], [Funamoto and Hayase, 2013]): this is particularly relevant, since it is one of the main indicators of the severity of stenoses and eventual arterial blockages. The direct measure of a pressure (or even a pressure drop) could be performed by implanting a catheter, hence in a rather invasive way.

3. *The vorticity* ([Mehregan et al., 2014], [Hirtler et al., 2016], [Charonko et al., 2013], [Sotelo et al., 2018]): this quantity is monitored especially in the heart cavity and around cardiac valves. A too large vorticity could induce, for instance, haemolysis.

4. *The wall shear stress* ([Gibson et al., 1993], [Reneman et al., 2006], [Shojima et al., 2004]): this is related to the mechanical stress that the blood exerts on the endothelial cells, of paramount importance in aneurysms and plaque formation.

The QoI listed have to be reliably estimated *in vivo*, with the additional constraint of being estimated fast, ideally in real time. For this, two main approaches are available. The first one consists in a purely data-driven strategy where learning techniques are used to build an approximation of the observable-to-QoI map given a sufficiently large dataset. The second one consists in using an *a priori* description of the physics involved by means of a mathematical model, often given in the form of a Partial Differential Equation (PDE), and then solve an inverse problem. On the one hand, since we are dealing with space fields estimation, the pure data-driven learning approach will in general need an exceedingly large data set to meet the accuracy constraints of the application. On the other hand, discretising the system of Partial Differential Equations and solving the inverse problem will in general result in a prohibitive computational cost, thus leading to unacceptable computing times. These facts motivate the use of mixed approaches combining an *a priori* knowledge coming from an available, potentially

inexact physical model of the system, and the *a posteriori* knowledge coming from the data. One recent example in this direction is [Kissas et al., 2020], where a physics-informed machine learning approach to estimate pressure in blood vessels from MRI was proposed. In this work, we use a different methodology based on reduced-order modelling of parameterized PDEs.

Our contribution is to propose a systematic methodology to estimate the above five QoI on quasi-real time involving reduced modelling techniques, and to assess its feasibility in non trivial numerical examples involving the carotid artery. However, due to our lack of real ultrasound images, our experiments present certain limitations: we have worked with synthetically generated images and have used an admittedly simple Gaussian modelling of the ultrasound noise (Doppler ultrasound images present a very involved space-time structure which is not the main topic of our work and we refer to [Ledoux et al., 1997, Bjaerum et al., 2002, Demené et al., 2015] for further details on this matter). The PDE model considered to describe the haemodynamics is the system of incompressible Navier-Stokes equations, which is generally acknowledged to be accurate for large vessels such as the carotid artery. We therefore assume that there is no model error and that the true system is governed by these equations. Note in addition that this assumption also comes from the fact that it is not possible to study the impact of the model error without real measurements.

The structure of the chapter is as follows. In Section 2.2 we describe the reconstruction method which we use. The method is very general and its main mathematical foundations have been established in previous works (see [Maday and Mula, 2013, Maday et al., 2015a, Maday et al., 2015b, Maday et al., 2016, Binev et al., 2017a, Argaud et al., 2017, Taddei, 2017]). We make a presentation that alternates between a summary of the general mathematical theory, and its particular application to the problem of interest. One relevant point to remark is that so far the methodology has mainly focused on reconstructing spatial fields from observable quantities. In the present case, this concerns the reconstruction of the 3D velocity field. One relevant novelty with to previous contributions is that we show that it is possible to reconstruct unobserved quantities such as the pressure field or the pressure drop in our problem. This is possible by making a *joint reconstruction of observable and unobservable fields*, which are velocity and pressure in the present. We explain this idea in section 3.2. The reconstruction of the wall shear stress and the vorticity are discussed in Sections 3.4 and 3.5 respectively. The numerical experiments on a carotid bifurcation are given in Section 3.6 for the case of noiseless images. We examine the effect of noise in Section 3.7.

Thus, the task is to use Doppler velocity measurements taken from a fluid flow and to reconstruct:

- **Partially observable quantities:** the full 3D velocity flow in Ω and related quantities

such as the wall shear stress and vorticity.

- **Non-observable quantities:** the full 3D pressure flow in Ω and the pressure drop.

The proposed strategy to address this task consists essentially in two steps:

- We apply the piecewise linear reconstruction algorithm of section 2.2.2 where the key is to do a *joint reconstruction* of 3D velocity and pressure.
- We then derive the related quantities of interest as a simple by-product (wall shear stress and vorticity).

3.2 Joint reconstruction of velocity and pressure

For the reasons explained in section 5.3, the couple (u, p) of velocity and pressure belongs to the Cartesian product

$$V = U \times P = [H^1(\Omega)]^3 \times L^2(\Omega),$$

and is assumed to be the solution to the parameter-dependent Navier-Stokes equations (2.26) for some parameter $y \in Y$. Some elements y^{obs} are observed but others are not so we cannot directly solve (2.26) with the parameters set to y. We therefore use the piecewise linear reconstruction of section 2.2.2. For this, it is necessary to endow V with the external direct sum and product structure to build a Hilbert space. That is, for any two elements (u_1, p_1) and (u_2, p_2) of $V = U \times P$ and any scalar $\alpha \in \mathbb{R}$,

$$(u_1, p_1) + (u_2, p_2) = (u_1 + u_2, p_1 + p_2), \quad \alpha(u_1, p_1) = (\alpha u_1, \alpha p_1)$$

The inner product is defined as the sum of component-wise inner products

$$\langle (u_1, p_1), (u_2, p_2) \rangle_V = \langle u_1, u_2 \rangle_U + \langle p_1, p_2 \rangle_P , \tag{3.1}$$

and it induces a norm on V,

$$\|(u, p)\| \stackrel{\text{def}}{=} (\langle (u, p), (u, p) \rangle_V)^{1/2}, \quad \forall (u, p) \in V.$$

When we are given partial information on (u, p) from Doppler velocity measures, we are given the projection

$$\omega = P_{W_m}(u, p)$$

where W_m is the observation space

$$W_m \stackrel{\text{def}}{=} W_m^{(u)} \times \{0\} = \text{span}\{\omega_1, \ldots, \omega_m\} \times \{0\} \subset V$$

and the ω_i are the Riesz representers in U of each voxel $\ell_i \in U'$,

$$\langle \omega_i, v \rangle_U = \ell_i(v) = \int_{\Omega_i} v \cdot b \, \mathrm{d}x, \quad \forall v \in U.$$

We are now in position to apply directly the reconstruction algorithms from section 2.2 to do the joint reconstruction of (u, p) with the current particular choice of Hilbert space V and observation space W_m. We briefly instantiate here the main steps. Let us assume that we have a reduced model

$$V_n \overset{\text{def}}{=} \operatorname{span}\{(u_1, p_1), \ldots, (u_n, p_n)\}$$

of dimension $n \leq m$ that approximates

$$\mathcal{M} \overset{\text{def}}{=} \{(u(y), p(y)) \in V \, : \, y \in \mathrm{Y}\}$$

with accuracy

$$\varepsilon_n \overset{\text{def}}{=} \sup_{(u,p)\in\mathcal{M}} \operatorname{dist}((u,p), V_n), \quad \text{or} \quad \delta_n^2 \overset{\text{def}}{=} \mathbb{E}\left(\operatorname{dist}((u,p), V_n)^2\right) \tag{3.2}$$

and which is such that $\beta(V_n, W_m) > 0$. Then, we can reconstruct with the linear PBDW method (see equation (2.5)) which, in the present case, reads

$$A_{m,n}^{(\text{pbdw})}(\omega) = (u_{m,n}^*(\omega), p_{m,n}^*(\omega)) \overset{\text{def}}{=} \operatorname*{arg\,min}_{(u,p)=\omega+W^\perp} \|(u,p) - P_{V_n}(u,p)\|. \tag{3.3}$$

The worst and average reconstruction errors are bounded like in estimates (2.8) and (2.9), that is

$$e_{m,n}^{(\text{wc, pbdw})} = \max_{(u,p)\in\mathcal{M}} \|(u,p) - (u_{m,n}^*(\omega), p_{m,n}^*(\omega))\| \leq \beta^{-1}(V_n, W_m)\, \varepsilon_n, \tag{3.4}$$

or

$$e_{m,n}^{(\text{ms, pbdw})} = \mathbb{E}\left(\|u - A_{m,n}^{((\text{pbdw}))}(P_{W_m} u)\|^2\right)^{1/2} \leq \beta^{-1}(V_n, W_m)\, \delta_n, \tag{3.5}$$

If we build a partition of the manifold $\mathcal{M}$ based on observed parameters, we can reconstruct with the piecewise linear algorithm of section 2.2.2.

3.3 Pressure and pressure prop estimation

Decomposing the domain boundary $\partial\Omega$ of a generic arterial bifurcation into the inlet, the wall and the outlet parts

$$\partial\Omega = \Gamma_{\text{in}} \cup \Gamma_w \cup \Gamma_{\text{out}}^1 \cup \ldots \cup \Gamma_{\text{out}}^l,$$

the quantities to retrieve are

$$\delta p_i = \frac{1}{|\Gamma_{\text{in}}|} \int_{\Gamma_{\text{in}}} p \, \mathrm{d}s - \frac{1}{|\Gamma_{\text{out}}^i|} \int_{\Gamma_{\text{out}}^i} p \, \mathrm{d}s, \tag{3.6}$$

for the outlet labels $i = 1, \ldots, l$.

The pressure drop is a quantity that has traditionally been of high interest to the medical community since it serves to assess, for instance, the severity of stenosis in large vessels due to the accumulation of fat in the walls, aortic regurgitation or arteriosclerosis. Figure 3.1 provides well known pressure profiles for a number of cardiovascular diseases. The potential predictive power of the results will depend whether we are capable to differentiate among the different profiles for a given patient with only the Doppler data.

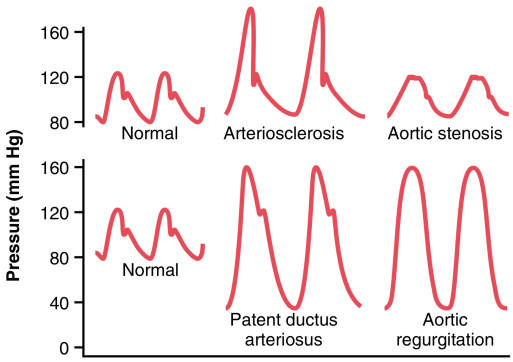


Figure 3.1: Aortic pressure pulse signals for several pathologies. Image source: [Guyton and Hall, 2011].

3.3.1 Joint Reconstruction

If we reconstruct (u_n^*, p_n^*), we can easily approximate the pressure drop by

$$\delta p_i^* = \frac{1}{|\Gamma_{\text{in}}|} \int_{\Gamma_{\text{in}}} p_n^* \, \mathrm{d}s - \frac{1}{|\Gamma_{\text{out}}^i|} \int_{\Gamma_{\text{out}}^i} p_n^* \, \mathrm{d}s,$$

for $i = 1, \ldots, l$.

As we will see in the numerical results, the pressure drop is approximated at very high accuracy with δp_i^* . We next provide a theoretical justification.

For this, we remark that we can view δp_i as a bounded linear mapping from $V = U \times P$ to $\mathbb{R}$ defined as

$$\delta p_i((u,p)) = \frac{1}{|\Gamma_{\text{in}}|} \int_{\Gamma_{\text{in}}} p \, \mathrm{d}s - \frac{1}{|\Gamma_{\text{out}}^i|} \int_{\Gamma_{\text{out}}^i} p \, \mathrm{d}s, \quad \forall (u,p) \in V.$$

Thus the reconstruction error is given by

$$|\delta p_i((u,p)) - \delta_i p((u_n^*, p_n^*))|.$$

Exploiting the linearity of δp_i, one can derive the simple bound

$$|\delta p_i((u,p)) - \delta p_i((u_n^*, p_n^*))| = |\delta p_i\left((u,p) - (u_n^*, p_n^*)\right)| \tag{3.7}$$
$$\leq \|\delta p_i\|_{V'} \|(u,p) - (u_n^*, p_n^*)\| \tag{3.8}$$
$$\leq \|\delta p_i\|_{V'} \beta(V_n, W_m) \|(u,p) - P_{V_n}(u,p)\| \tag{3.9}$$
$$\leq \|\delta p_i\|_{V'} \beta(V_n, W_m) \varepsilon_n, \tag{3.10}$$

where we have used (2.8) between the second and the third line and where

$$\|\delta p_i\|_{V'} \stackrel{\text{def}}{=} \sup_{(u,p)\in V} \frac{|\delta p_i(u,p)|}{\|(u,p)\|} \geq 1.$$

As we will see below, this estimate is too coarse to account for the high reconstruction accuracy which is observed because the values $\beta(V_n, W_m)$ are close to zero and the product $\beta(V_n, W_m)\varepsilon_n$ is only moderately small. It is necessary to find a sharper estimate that involves finer constants in front of ε_n to account for the good reconstruction results. For this, observing that, by construction of (u_n^*, p_n^*),

$$\mathrm{P}_{W_m}(u,p) = \mathrm{P}_{W_m}(u_n^*, p_n^*),$$

we have

$$(u,p) - (u_n^*, p_n^*) \in {W_m}^{\perp}$$

so we can derive the new estimate

$$|\delta p_i((u,p)) - \delta p_i((u_n^*, p_n^*))| \leq \kappa_{m,n} \|(u,p) - (u_n^*, p_n^*) - P_{V_n}\left((u,p) - (u_n^*, p_n^*)\right)\| \tag{3.11}$$
$$\leq 2\kappa_{m,n}\varepsilon_n, \tag{3.12}$$

with

$$\kappa_{m,n} \stackrel{\text{def}}{=} \sup_{(u,p)\in {W_m}^{\perp}} \frac{|\delta p_i((u,p))|}{\|\operatorname{dist}\left((u,p), V_n\right)\|}. \tag{3.13}$$

As we illustrate in our numerical tests, the value of $\kappa_{m,n}$ is moderate and significantly smaller than the factor $\|\delta p_i\|_{V'}\beta(V_n, W_m)$ of the previous estimate. As a result, the product $\kappa_{m,n}\varepsilon_n$ is small, and we guarantee a reconstruction of the pressure with good accuracy.

3.3.2 Virtual work principle from the reconstruction of u_n^*

As an alternative to the joint reconstruction strategy, we can use a method introduced in [Bertoglio et al., 2017b] called Integral Momentum Relative Pressure estimator. As a starting point, it requires to work with a reconstruction u_n^* of the velocity which, in our work, will be given by the PBDW method applied only to the reconstruction of the velocity field without

pressure. We then estimate the pressure drop using the Navier-Stokes equations as follows. Assuming that u_n^* satisfies perfectly the momentum conservation (2.26), we test by a virtual and divergence free velocity field $v \in U$,

$$\underbrace{\rho \int_\Omega \partial_t u_n^* \cdot v \, \mathrm{d}x}_{K(u_n^*, v)} + \underbrace{\rho \int_\Omega (u_n^* \cdot \nabla u_n^*) \cdot v \, \mathrm{d}x}_{I_{conv}(u_n^*, v)} + \underbrace{\int_\Omega \nabla p \cdot v \, \mathrm{d}x}_{I_{press}(p, v)} \underbrace{- \mu \int_\Omega \Delta u_n^* \cdot v \, \mathrm{d}x}_{I_{visc}(u_n^*, v)} = 0. \tag{3.14}$$

Using Green's identities, we can write

$$\begin{aligned} I_{\mathrm{conv}}(u_n^*, v) &= \rho \int_{\partial\Omega} (u_n^* \cdot n)(u_n^* \cdot v) \, \mathrm{d}s - \rho \int_\Omega (u_n^* \cdot \nabla v) \cdot u_n^* \, \mathrm{d}x. \\ I_{\mathrm{visc}}(u_n^*, v) &= \mu \int_\Omega \nabla u_n^* : \nabla v \, \mathrm{d}x - \mu \int_{\partial\Omega} (\nabla u_n^* \cdot n) \cdot v \, \mathrm{d}s. \\ I_{\mathrm{press}}(u_n^*, v) &= \int_{\partial\Omega} p(v \cdot n) \, \mathrm{d}s - \int_\Omega p(\nabla \cdot v) \, \mathrm{d}x. \end{aligned} \tag{3.15}$$

The current strategy requires to assume that the pressure field is constant over the inlet and outlets. Notice that, since $\nabla \cdot v = 0$, the following identity holds,

$$I_{\mathrm{press}}(p, v) = p \int_{\partial\Omega} v \cdot n \, \mathrm{d}s = p_{in} \int_{\Gamma_{\mathrm{in}}} v \cdot n \, \mathrm{d}s + \sum_{i=1}^{l} p_{out}^i \int_{\Gamma_{\mathrm{out}}^i} v \cdot n \, \mathrm{d}s, \tag{3.16}$$

where p_{in} is the average pressure over Γ_{in} and p_{out}^i is the average pressure over the i-th outlet Γ_{out}^i. For $j = 1, \ldots, l$, we consider a function $v_j \in V$ satisfying $\nabla \cdot v_j = 0$ and $v_j = 0$ in Γ_w. Mass conservation for incompressible regimes implies

$$\int_{\Gamma_{\mathrm{in}}} v_j \cdot n \, \mathrm{d}s + \sum_{i=1}^{l} \int_{\Gamma_{\mathrm{out}}^i} v_j \cdot n \, \mathrm{d}s = 0, \tag{3.17}$$

for $j = 1, \cdots, l$. As a result, it is possible to recover the mean pressure drop $x_j = p_{out}^j - p_{in}$ for each outlet $j = 1, \cdots, l$ by solving an $l \times l$ system of equations

$$Fx = H(u_n^*), \tag{3.18}$$

where $F \in \mathbb{R}^{l \times l}$ has entries

$$F_{ij} = \int_{\Gamma_{\mathrm{out}}^j} v_i \cdot n \, \mathrm{d}s, \tag{3.19}$$

and,

$$H_i(u_n^*) = -\left(I_{\mathrm{visc}}(u_n^*, v_i) + I_{\mathrm{conv}}(u_n^*, v_i) + I_{\mathrm{kin}}(u_n^*, v_i) \right). \tag{3.20}$$

The inversion of the system (3.18) is made trivial when the v_i are chosen so that F becomes diagonal (with nonzero entries). We remark that this is achieved if we choose the v_i to be divergence-free and to have outgoing zero flux in all the outlets Γ_{out}^j for $j \neq i$. For each

$i = 1, \ldots, l$, v_i can be characterized as the unique solution to the following Stokes problems: Find $v_i \in U$ and $\lambda \in L^2(\Omega)$ auxiliary function such that:

$$\begin{cases} -\Delta v_i + \nabla \lambda = (0,0,0) & \text{in } \Omega, \\ \nabla \cdot v_i = 0 & \text{in } \Omega, \\ v_i = (0,0,0) & \text{on } \Gamma_{\text{w}}, \\ v_i = [(1,1,1) \cdot n]n & \text{on } \Gamma_{\text{in}}, \\ v_i = (0,0,0) & \text{on } \Gamma^j_{\text{out}} \quad \forall j \neq i, \\ \frac{1}{2}\left(\nabla v_i + \nabla^T v_i\right) n - \lambda = 0 & \text{on } \Gamma^i_{out}. \end{cases} \tag{3.21}$$

In order to ensure good stability when doing the time integration of (3.14), we use the Cranck-Nicholson scheme.

We may note that this method requires the knowledge of the flow viscosity and density, and it assumes constant pressure over the inlets and outlets. This is contrast to the joint reconstruction approach which does not need these assumptions.

3.4 Wall shear stress

The wall shear stress (WSS) has been proposed as an index of damage in vascular endothelial cells and atherosclerosis, a disease in which the blood coagulates close to the vessel walls. The works [Konstantinos et al., 2009], [Heo et al., 2014] or [Zarins et al., 1983] can serve as a reference. In figure 3.2 we see schematized the influence that plaque formation has in the blood flow and therefore its correlation with WSS.

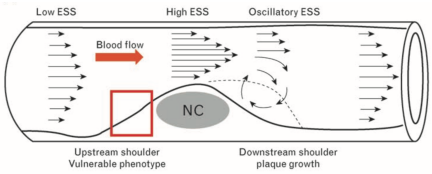


Figure 3.2: Wall shear stress alteration due to human carotid artery plaque, here refered as endothelial shear stress (ESS). Image source: [Konstantinos et al., 2009].

The WSS is a mapping $\mathcal{S} : U \to [H^{-1/2}(\Gamma_{\text{w}})]^3$, $\mathcal{S} : \Omega \mapsto \mathbb{R}^3$ that returns the tangential component of the force that the blood applies on the vessel wall

$$\mathcal{S}(u) := 2\mu \left\{ I - n \otimes n \right\} \left(\frac{\nabla u + \nabla u^T}{2} n \right), \text{ on } \Gamma_{\text{w}}. \tag{3.22}$$

Note that $\mathcal{S}(u) \in [H^{-1/2}(\Gamma_{\text{w}})]^3$ because a velocity solution u of the Navier-Stokes equations satisfies $(\nabla u)n \in [L^2(\Omega)]^3$ and, by Green's formula, we can prove that $(\nabla u)n|_{\Gamma_{\text{w}}} \in [H^{-1/2}(\Gamma_{\text{w}})]^3$.

Our goal is therefore to compute the reconstruction error $\|\mathcal{S}(u) - \mathcal{S}(u^*)\|_{[H^{-1/2}(\Gamma_w)]^3}$. First, we have that

$$\|\mathcal{S}(u) - \mathcal{S}(u^*)\|_{[H^{-1/2}(\Gamma_w)]^3} \lesssim \|\lambda_1\|_{[H^{-1/2}(\Gamma_w)]^3} + |\overline{\mathcal{S}(u)} - \overline{\mathcal{S}(u^*)}|$$

where $\overline{\mathcal{S}(u)} \coloneqq \int_{\Gamma_w} \mathcal{S}(u)(x)\, \mathrm{d}x$ and λ_1 is a vector field defined as

$$\lambda_1 \coloneqq \mathcal{S}(u) - \mathcal{S}(u^*) - \left(\overline{\mathcal{S}(u)} - \overline{\mathcal{S}(u^*)}\right) \in [H^{-1/2}(\Gamma_w)]^3$$

is a function of zero mean. We next prove that there exists a constant $C > 0$ such that

$$\|\lambda_1\|_{[H^{-1/2}(\Gamma_w)]^3} \leq C\|\phi_{\lambda_1}|_{\Gamma_w}\|_{[L^2(\Gamma_w)]^3} \tag{3.23}$$

where $\phi_{\lambda_1} \in [H^1(\Omega)]^3$ is the unique solution to the following homogeneous Laplace equation with Neumann boundary condition: Find $\phi_{\lambda_1} \in [H^1(\Omega)]^3$ with $\overline{\phi_{\lambda_1}} = 0$ such that

$$\int_\Omega \nabla\phi_{\lambda_1} : \nabla v \, \mathrm{d}x = \int_{\Gamma_w} \lambda_1 \cdot v \, \mathrm{d}s = \langle \lambda_1, \mathrm{Tr}(v)\rangle_{[H^{-1/2}(\Gamma_w)]^3,[H^{1/2}(\Gamma_w)]^3}, \quad \forall v \in [H^1(\Omega)]^3, \tag{3.24}$$

where $\mathrm{Tr} : [H^1(\Omega)]^3 \to [H^{1/2}(\Omega)]^3$ is the trace operator. By applying the Cauchy-Schwarz inequality in (3.24), we derive

$$\langle \lambda_1, \mathrm{Tr}(v)\rangle_{[H^{-1/2}(\Gamma_w)]^3,[H^{1/2}(\Gamma_w)]^3} \leq \|\nabla\phi_{\lambda_1}\|_{[L^2(\Omega)]^3}\|\nabla v\|_{[L^2(\Omega)]^3} \leq \|\phi_{\lambda_1}\|_{[H^1(\Omega)]^3}\|v\|_{[H^1(\Omega)]^3},$$

from which we deduce that

$$\|\lambda_1\|_{[H^{-1/2}(\Gamma_w)]^3} \leq \|\phi_{\lambda_1}\|_{[H^1(\Omega)]^3}.$$

Finally, by continuity of the mapping $\lambda_1 \in [L^2(\Gamma_w)]^3 \mapsto \phi_{\lambda_1} \in [H^1(\Omega)]^3$, there exists a constant $C > 0$ such that $\|\phi_{\lambda_1}\|_{[H^1(\Omega)]^3} \leq C\|\lambda_1\|_{[L^2(\Gamma_w)]^3}$. This yields our final, computable bound

$$\|\mathcal{S}(u) - \mathcal{S}(u^*)\|_{[H^{-1/2}(\Gamma_w)]^3} \leq C(\|\mathrm{Tr}(\phi_{\lambda_1})\|_{[L^2(\Gamma_w)]^3} + |\overline{\mathcal{S}(u)} - \overline{\mathcal{S}(u^*)}|).$$

Thus, to evaluate the reconstruction quality of the WSS at a given time t we compute:

$$e_{\mathrm{wss}}(t) = \frac{\|\mathrm{Tr}(\phi_{\lambda_1})\|_{[L^2(\Gamma_w)]^3} + |\overline{\mathcal{S}(u(t))} - \overline{\mathcal{S}(u^*(t))}|}{\|\mathrm{Tr}(\phi_{\lambda_2})\|_{[L^2(\Gamma_w)]^3} + |\overline{\mathcal{S}(u(t))}|} \tag{3.25}$$

where $\lambda_2 = \mathcal{S}(u(t)) - \overline{\mathcal{S}}(u(t))$.

3.5 Vorticity

The vorticity is defined as $\Theta = \nabla \times u$. The relative L^2 error in time for the vorticity reconstruction $\Theta^* = \nabla \times u^* = \nabla \times A(P_{W_m} u)$ is given by

$$e_{\text{vorticity}}(t) = \frac{\|\Theta(t) - \Theta^*(t)\|}{\left(\int \|\Theta(t)\|^2 \, \mathrm{d}t\right)^{1/2}}, \tag{3.26}$$

Since $u \in U = [H^1(\Omega)]^3$ with $\nabla \cdot u = 0$ (incompressible flow) and since we have the identity (see, e.g., [Guermond et al., 2005])

$$\|\nabla \times u\|^2_{L^2(\Omega)} = \|\nabla u\|^2_{L^2(\Omega)} + \|\nabla \cdot u\|^2_{L^2(\Omega)} = \|\nabla u\|^2_{L^2(\Omega)}, \tag{3.27}$$

it follows by (2.8) that

$$\|\Theta - \Theta^*\|_{L^2(\Omega)} \leq \|u - u^*\|_U \leq \beta^{-1}(V_n, W_m)\|u - P_{V_n} u\|_U \tag{3.28}$$

What motivates the computation of this QoI is that it provides information about the shear layer thickness, which has been correlated with thrombus formation and hemolysis [Bluestein et al., 2010]. In general, vorticity is connected to the assessment of the cardiovascular function. There have been also efforts to reconstruct it from magnetic resonance images (see, for instance: [Garcia et al., 2013]).

3.6 Noise-free numerical test in a carotid geometry

This section will present extensive test on QoI computations up to what extent we can get accuracy under noiseless conditions. Our experiment will be exactly the same as that of section 2.4.3 for the common carotid artery branches.

3.6.1 Optimal partitioning and optimal dimension of V_n

As was already brought up in the numerical example of chapter 2, we can use our piece-wise linear approach due to the fact that during ultrasound examination, we have access to the patient's heart rate HR and the time t of the cardiac cycle. We can therefore decompose the vector y of parameters as

$$y = (y^{\text{obs}}, y^{\text{unobs}}), \quad y^{\text{obs}} = (t, \text{HR}), \quad y^{\text{unobs}} = (s, T_{sys}, u_0, \eta).$$

and use the piecewise reconstruction algorithm introduced in section 2.2.2.

Let us recall that the strategy that we have followed consists in computing first a training subset $\widetilde{\mathcal{M}}$ of snapshots. We next consider a splitting of the time interval into $K \in \mathbb{N}^*$ uniform subintervals

$$[0, T] = \cup_{k=0}^{K-1} \tau_k, \quad \text{with } \tau_k = [kT/K, (k+1)T/K[$$

We proceed similarly for the heart rate's interval and split it into $K' \in \mathbb{N}^*$ uniform subintervals,

$$[48, 120] = \cup_{k'=0}^{K'-1} h_{k'}, \quad \text{with } h_{k'} = [48 + 72k'/K', 48 + 72(k'+1)/K'[.$$

For fixed (K, K'), we have the partition in the parameter domain (see figure 2.2)

$$Y^{\text{obs}} = \bigcup_{(k,k')\in\{0,\dots,K-1\}\times\{0,\dots,K'-1\}} \tau_k \times h_{k'}, \qquad Y = \bigcup_{(k,k')\in\{0,\dots,K-1\}\times\{0,\dots,K'-1\}} \tau_k \times h_{k'} \times Y^{\text{unobs}}$$

and the induced partition in the manifold

$$\mathcal{M} = \bigcup_{(k,k')\in\{0,\dots,K-1\}\times\{0,\dots,K'-1\}} \mathcal{M}^{(k,k')}$$

We have performed test changing the dimension of the reduced model. Nevertheless, we need a systematic way to choose a unique n to do the computations to optimize on the reconstruction quality. We next proceed to introduce an optimality criteria to choose this number.

For each $\mathcal{M}^{(k,k')}$, we can compute reduced models $V_n^{(k,k')}$. If we measure the reconstruction error in the worst case sense, we can estimate the reconstruction performance with this splitting by computing

$$e\left(K, K'\right) = \max_{(k,k')\in\{0,\dots,K-1\}\times\{0,\dots,K'-1\}} \min_{1\le n\le m} \max_{u\in\widetilde{\mathcal{M}}_{(k,k')}} \frac{\text{dist}(u, V_n^{(k,k')})}{\beta(V_n^{(k,k')}, W_m)} \tag{3.29}$$

We then look for the optimal partition when K and K' range between 1 and 7, that is, we select

$$(K_{\text{opt}}, K'_{\text{opt}}) \in \underset{(K,K')\in\{1,\dots,7\}\times\{1,\dots,7\}}{\arg\min} e\left(K, K'\right).$$

When we consider only the velocity u as a target quantity, we obtain a 5×5 partitioning of Y^{obs}. In figure 3.3, we show the behavior with n of the stability constant $\beta(V_n^{(k,k')}, W_m)$ and the error $\max_{u\in\widetilde{\mathcal{M}}^{(k,k')}} \text{dist}(u, V_n^{(k,k')})$ for each element of this optimal partition.

We proceed similarly to derive the optimal partition for the couple velocity-pressure (u, p) in $V = U \times P$. We also obtain a 5×5 partition and figure 3.4 shows the behavior of $\beta(V_n^{(k,k')}, W_m)$ and the error $\max_{(u,p)\in\widetilde{\mathcal{M}}^{(k,k')}} \text{dist}((u,p), V_n^{(k,k')})$ for this case. Note that the value of the stability constant is very low, and this is due to the fact that our measurement space allows only to sense in the velocity.

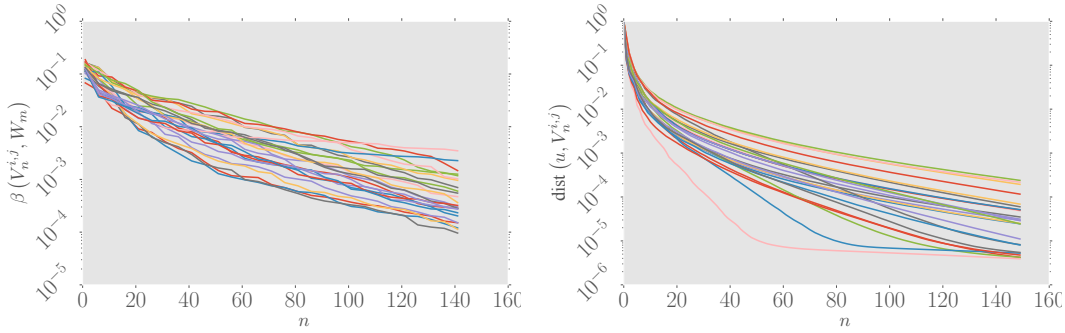
Once the optimal partition has been found, for each subset $\mathcal{M}^{(k,k')}$, we select the optimal dimension n^* as

$$n^*_{(k,k')} \in \underset{n=1,\dots,m}{\arg\min} \frac{\text{dist}(u, V_n^{(k,k')})}{\beta(V_n^{(k,k')}, W_m)}. \tag{3.30}$$

This procedure for the selection of $n^*_{(k,k')}$ is referred to as *the multi-space* approach in [Binev et al., 2017b].

3.6.1.1 Impact of ambient space choice in the reconstruction quality

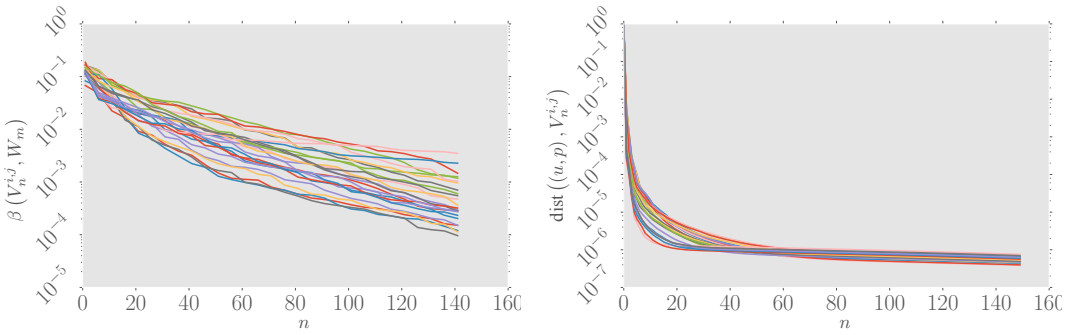
It might be natural to think that the inner product introduced with (3.1) could not be the optimal choice, since it is well known that the orders of magnitude of both pressure and velocity



(a) Stability constant $\beta\left(V_n^{i,j}, W_m\right)$ for each manifold partition

(b) Model error dist $\left(u, V_n^{i,j}\right)$ for each manifold partition

Figure 3.3: Behavior of stability constant and model error with respect to the dimension of V_n for reconstruction of velocity only. Optimal partitioning is $K = K' = 5$.



(a) Stability constant $\beta\left(V_n^{i,j}, W_m\right)$ for each manifold partition

(b) Model error dist $\left((u,p), V_n^{i,j}\right)$ for each manifold partition

Figure 3.4: Behavior of stability constant and model error respect to the dimension of V_n for joint reconstruction of section 3.2. Optimal partitioning is $(K = K' = 5)$.

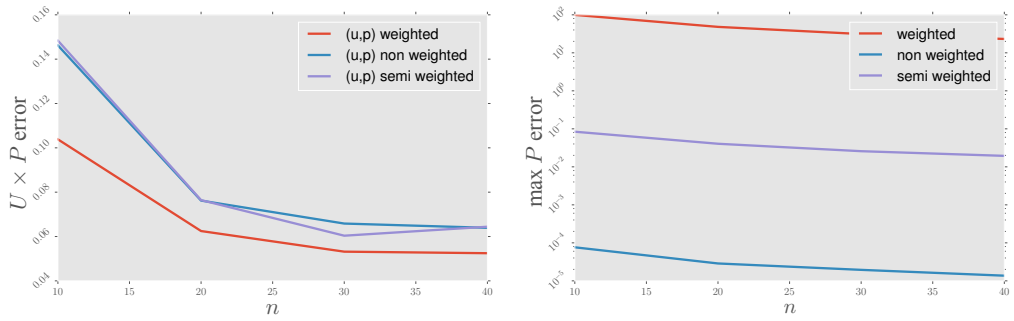
are very different. Thus, one could try to compensate this difference by adopting the following inner product for elements $(u, p) \in U \times P$:

$$\langle (u,p),(u,p) \rangle = \langle u,u \rangle_U + \alpha \langle p,p \rangle_P \tag{3.31}$$

$\alpha > 0$ is introduced in order to account for this aforementioned difference in the orders of magnitude of the pressure field in the mixed space. One would expect that, the lower the value of α, the closer we get to the reconstruction stability and result of (2.6). Different options for this parameter are going to be tested. Among them, one that properly weights pressure and velocity importance would be:

$$\alpha = \frac{\arg\max_{u \in \widetilde{\mathcal{M}}_{\text{train}}} \|u\|_U}{\arg\max_{p \in \widetilde{\mathcal{M}}_{\text{train}}} \|p\|_P}. \tag{3.32}$$

where in this context $\widetilde{\mathcal{M}}_{\text{train}}$ corresponds to the training set made up by a finite number of samples of FEM solutions of the governing laws.



(a) Error in (u, p)

(b) Worst case scenario L^2 error in pressure

Figure 3.5: Impact of the weighting parameter in the inner product that defines the search space. In the legend, non weighted stands for $\alpha = 1$, weighted stands for α as expressed in (3.32) and semi-weighted stands for α as the square root of (3.32)

Nevertheless, we can verify in figure 3.5 that, against the intuition we have just brought to the table, the results suggest using $\alpha = 1$. The absolute lack of information about the pressure in W_m could explain this, we speculate.

3.6.2 Reconstruction results for velocity field and derived quantities

In the rest of the chapter, we use the piecewise linear approach with the optimal splitting and the optimal dimension $n^*_{(k,k')}$ for the reduced models $V_n^{(k,k')}$. To simplify notation, we will write V_n instead of $V_n^{(k,k')}$ when no confusion arises and (u^*, p^*) instead of $(u^*_{n^*_{(k,k')}}, p^*_{n^*_{(k,k')}})$. In addition,

depending on the context, V_n denotes either the reduced model for the velocity reconstruction or the reduced model for the joint reconstruction described in section 3.2.

Similarly to the numerical examples of chapter 3, the ultrasound device will be placed in such a way we measure the half of a mid plane of the working domain. The ultrasound wave will form an angle of $\pi/4$ to the dominant fluid direction in the main carotid branch, that is to say, $b = \left[\sqrt{2}/2, \sqrt{2}/2, 0\right]$. An example of a noiseless CFI during the systole phase can be depicted in figure 3.6

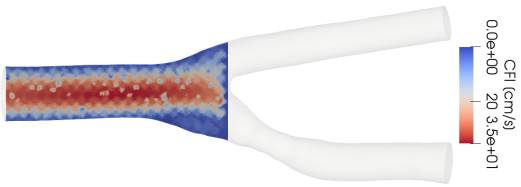


Figure 3.6: Synthetic CFI of the common carotid branch with 233 voxels. We simulate a device incapable of recovering velocity data after the artery bifurcation downstream, which includes an interesting extra challenge to the numerical example.

Figure 3.7 shows the relative error in time in the velocity reconstruction

$$e(u(t))^2 = \frac{\|u(t) - u^*(t)\|_U^2}{\int \|u(t)\|_U^2 dt}. \tag{3.33}$$

in norm $U = [H^1(\Omega)]^3$. We observe that there is no field over 10% error. One can further examine the error by studying separately the $L^2(\Omega)$ reconstruction error of the velocity and its gradient, as shown in figure 3.8. The reconstruction plots show that there are small error peaks around the region where one time window ends and the next one begins. Section 2.5 show preliminary results of a technique to overcome this issue.

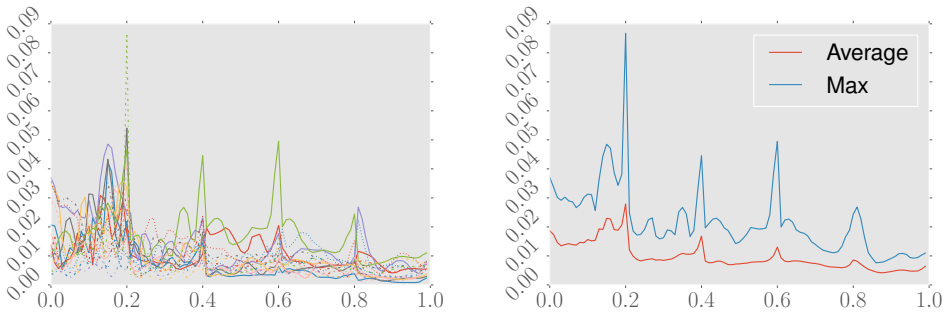
Figure 3.9 shows the approximation error due to the projection in V_n,

$$e_{V_n}(u(t))^2 = \frac{\|u(t) - P_{V_n} u(t)\|^2}{\int \|u(t)\|^2 dt}.$$

We see that this error does not interfere with the reconstruction quality in the sense that it stays much lower than (3.33).

An example of the velocity reconstruction for one patient during the early systole period is shown in figure 3.10, where we observe that the larger errors occur in the stenosis zone. This behavior is observed during the whole cardiac cycle.

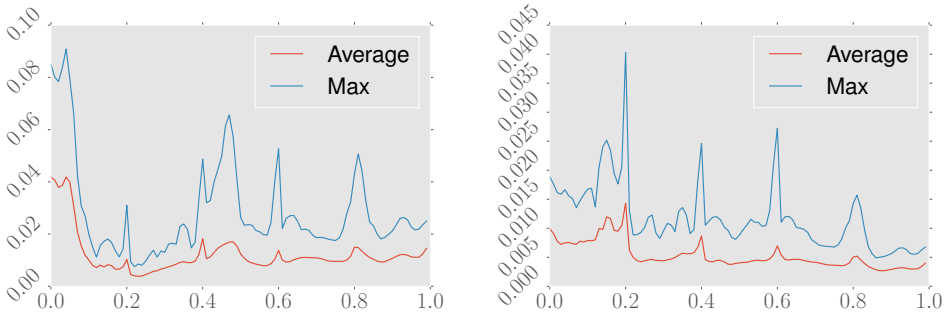
Concerning the WSS and vorticity, we can see the time evolution of the errors (3.25) and (3.26) in figure 3.11. As an illustration, figure 3.12 and figure 3.13 shows a three-dimensional



(a) $e\,(u\,(t))$ for each patient

(b) $e\,(u\,(t))$ averaged and max

Figure 3.7: Reconstruction error in $U = [H^1(\Omega)]^3$ of the velocity field for 16 patients. Notice the small jumps at each time window interface. The vertical axis shows the error as expressed in (3.33). The horizontal axis shows the normalized time for one cardiac cycle.



(a) $e((u(t))$ in norm L^2

(b) $e((\nabla u(t)))$ in norm L^2

Figure 3.8: L^2 error in velocity reconstruction. We observe that the accuracy for both u^* and ∇u^* stays in the same orders of magnitude. The vertical axis shows the error as expressed in (3.33). The horizontal axis shows the normalized time for one cardiac cycle.

reconstructed vorticity and wall shear stress fields, ively. The error in the stenosis area tends to be *propagated* from the velocity reconstruction, as can be expected.

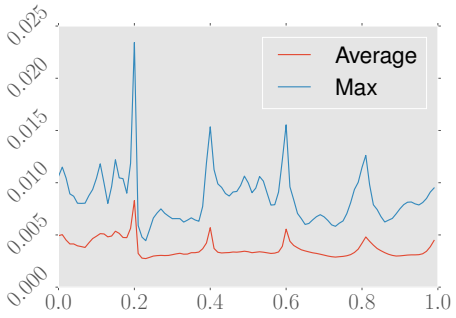
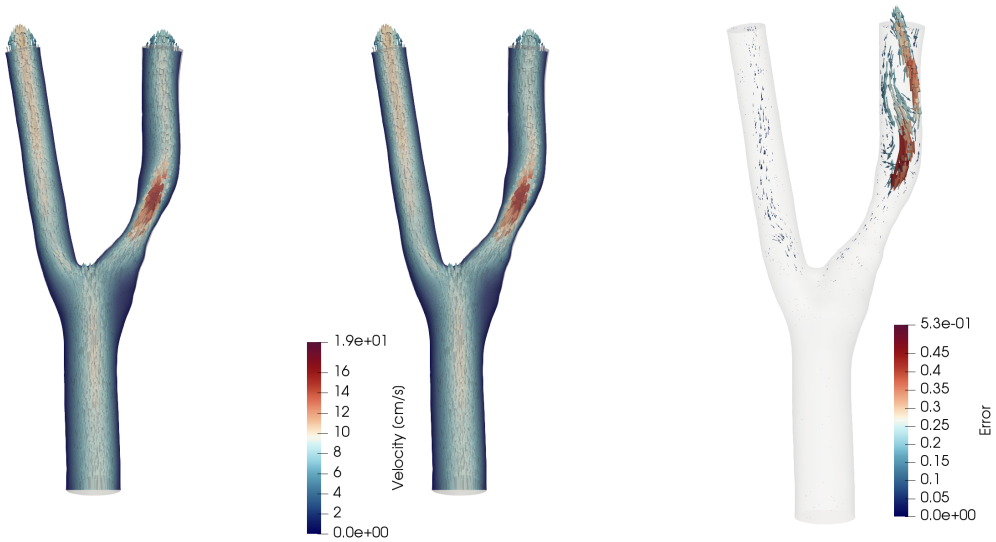


Figure 3.9: Model error $e_{V_n}(u(t))$. The horizontal axis shows the normalized time for one cardiac cycle.

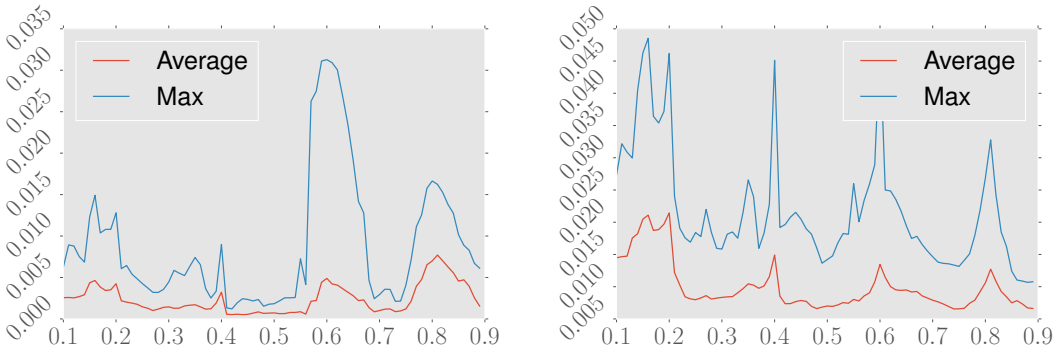


(a) u. (b) u^*. (c) $u - u^*$.

Figure 3.10: Example of reconstruction of the velocity during the early systole period. We observe a zone of high error close to the stenosis.

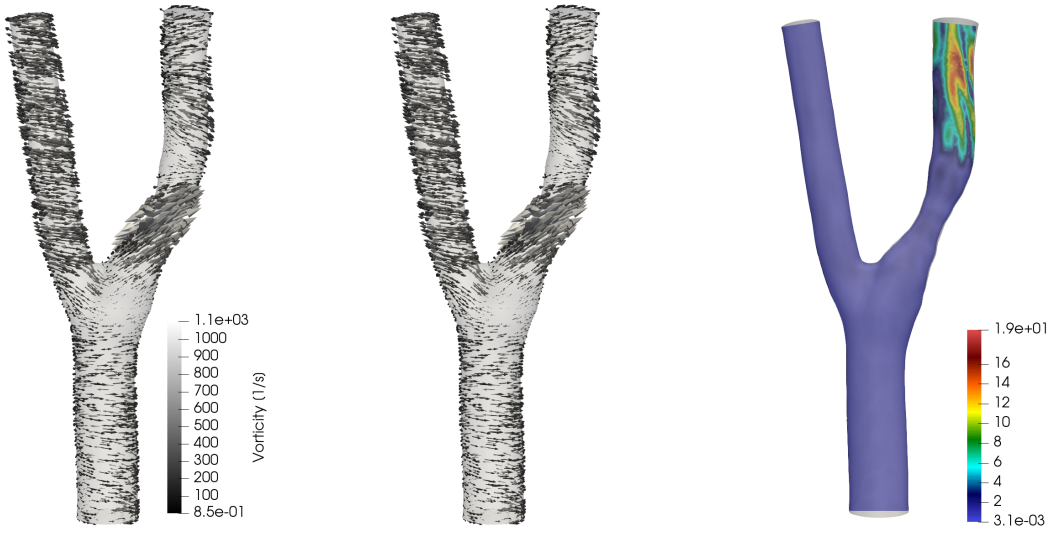
3.6.3 Pressure Drop Estimation Results

This section is devoted to comparing the two reconstruction methods for the pressure drop introduced in section 3.3.



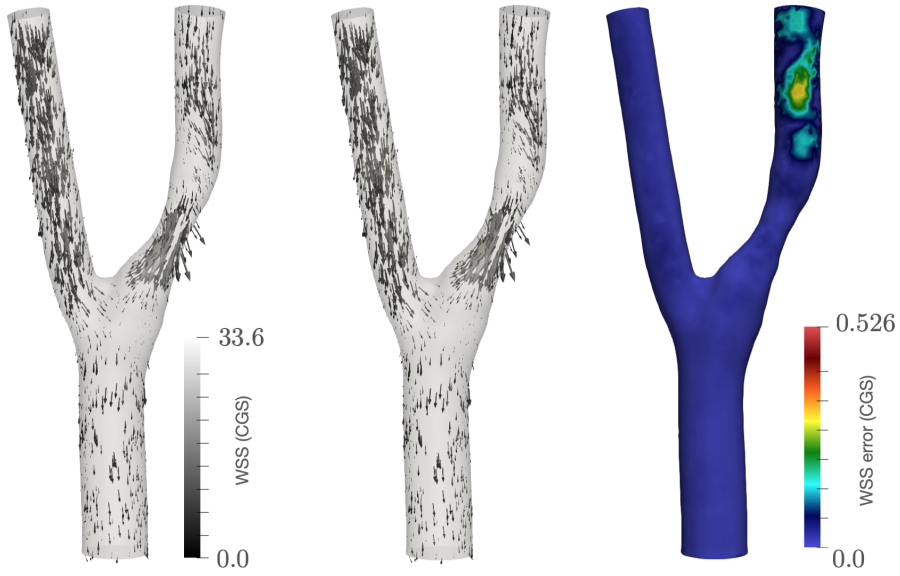
(a) $e_{wss}(t)$ (b) $e_{vorticity}(t)$

Figure 3.11: Relative errors (3.26) and (3.25) in time. The horizontal axis shows the normalized time for one cardiac cycle.



(a) Θ. (b) Θ^*. (c) $\Theta - \Theta^*$

Figure 3.12: Example of reconstruction of the vorticity during the early systole period. We observe how the error from the velocity reconstruction is reproduced close to the stenosis.



(a) $\mathcal{S}$. (b) $\mathcal{S}^*$. (c) $\mathcal{S} - \mathcal{S}^*$.

Figure 3.13: Example of reconstruction of the wall shear stress during the early systole period. We observe how the error from the velocity reconstruction is reproduced close to the stenosis.

In the first method, we first compute the joint reconstruction of velocity-pressure with the piecewise linear algorithm, and then compute the pressure drop with formula (3.6). Figure 3.14 shows the evolution of the estimated pressure drop in 4 simulations and compares it with the evolution of the exact pressure drop. The figure shows that the methods delivers a very high accuracy.

We can justify the obtained high accuracy by estimating the value of the stability parameter $\kappa_{m,n}$ defined in (3.13). For this, we approximate the space $W^\perp$ with

$$\widetilde{W}^\perp = \text{span}\{\Psi_1, \dots, \Psi_N\} \subset W^\perp,$$

where $\{\Psi_1, \dots, \Psi_N\}$ is an orthonormal set of functions of $W^\perp$. These functions are obtained, for example, by first computing a singular value decomposition with $N \gg n$ functions $\phi_i = (u_i, p_i)$ from the manifold $\mathcal{M}$. In our case we set $N = 250$. We can then orthonormalize them with to $W_m = \text{span}\{\omega_1, \dots, \omega_m\}$, which yields the desired

$$\Psi_i = \phi_i - P_{W_m}\phi_i. \tag{3.34}$$

We can next expand any function $\eta \in \widetilde{W}^\perp$ as

$$\eta = \sum_{i=1}^{N} \eta_i \Psi_i, \qquad \text{with } \eta_i = \langle \eta, \Psi_i \rangle \tag{3.35}$$

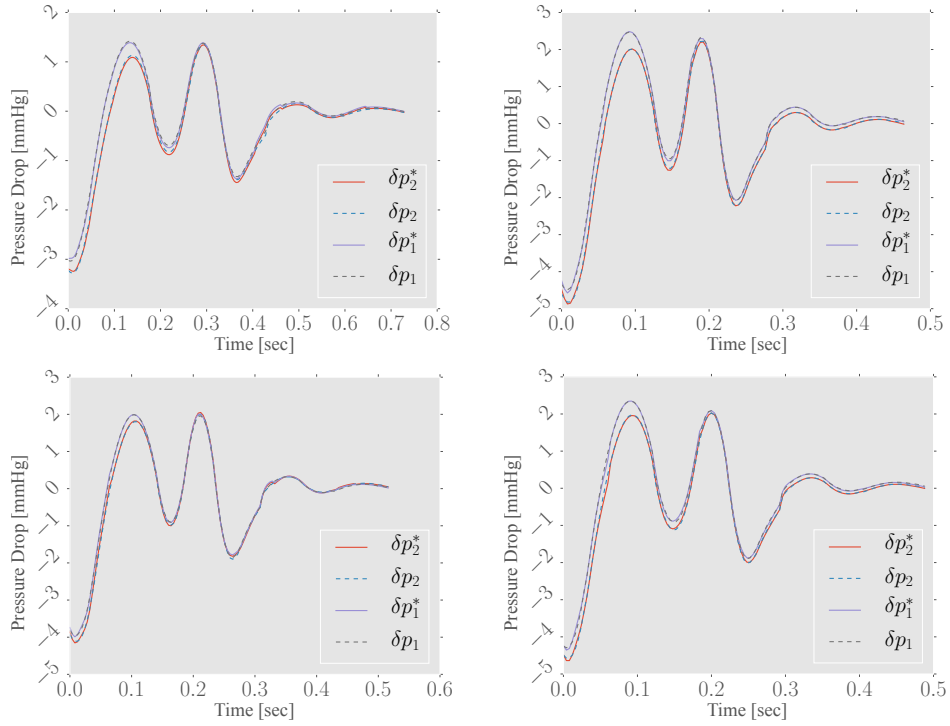
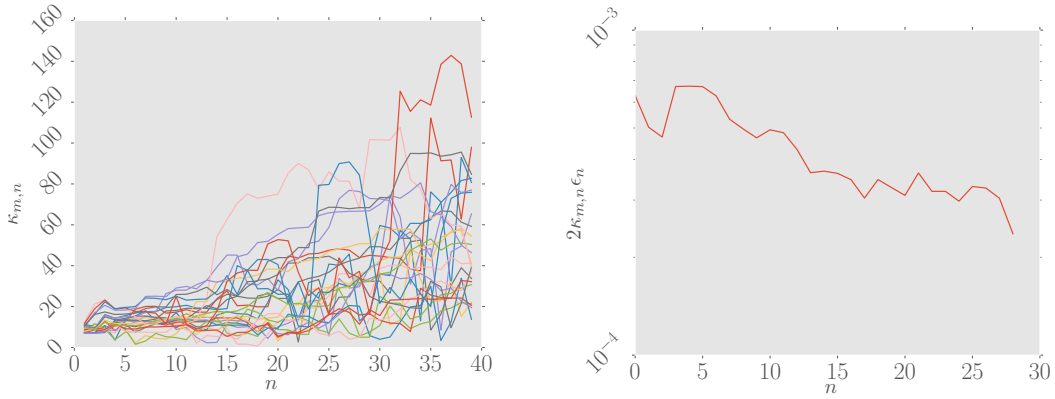


Figure 3.14: Pressure drops δp_1^* and δp_2^* in four simulations using the joint reconstruction method of section 3.2 with the piecewise linear algorithm. Dashed lines shows the ground truth δp_1 and δp_2. The vertical axis shows the pressure drop in [mmHg] and the horizontal axis the time in seconds.



(a) $\kappa_{m,n}$ as a function of n.

(b) Upper bound of pressure drop reconstruction error as a function of n.

Figure 3.15: Behavior of $\kappa_{m,n}$ with to dimension of V_n for each manifold partition in the method for joint reconstruction of section 3.2. In the right side we see the plot of $\max_i\{2\kappa_{m,n}\epsilon_n\}$ for $i = 1, \ldots, 25$ denoting the worst among the 25 windows in the piecewise linear approach. this quantity is an upper bound of the pressure drop reconstruction error (see inequality (3.12)).

The discrete version of equation (3.13) leads to the optimization problem

$$\begin{aligned} \max_{\eta \in \mathbb{R}^N} \quad & \eta^T Q \eta \\ \text{s.t.} \quad & \eta^T M \eta = 1, \end{aligned}$$

where

$$M_{ij} = \langle \Psi_i - P_{V_n}\Psi_i, \Psi_j - P_{V_n}\Psi_j \rangle,$$

and

$$Q_{ij} = Q(\Psi_i)Q(\Psi_j).$$

This problem is equivalent to the generalized eigenvalue problem of finding $\eta \in \mathbb{R}^N$ and the largest $\lambda \in \mathbb{R}_+$ such that

$$Q\eta = \lambda M \eta \tag{3.36}$$

As a result, we can estimate the value of $\kappa_{m,n}$ with the largest eigenvalue λ of problem (3.36). Figure 3.15(a) shows the estimated value of $\kappa_{m,n}$ as a function of the dimension n of the reduced model V_n. Since we use a 5×5 partition of the manifold, we plot the 25 associated curves. From the figure, we deduce that $\kappa_{m,n} \leq 160$ for all partitions. As we see from figure 3.15(b), the product $2\epsilon_n\kappa_{m,n}$ stays lower than 10^{-3} for any dimension n. As proven in (3.12), this quantity is an upper bound of the reconstruction error and rigorously confirms the quality of the approach.

We can next examine in figure 3.16 the performance of the second method involving virtual works discussed in section 3.3. Although we observe good results like for the previous method, a mismatch is observed for one of the common carotid branches in the systolic phase. This error is not observed with the joint reconstruction, and it is probably due to the fact that the method involves less assumptions on the nature of the flow and pressure.

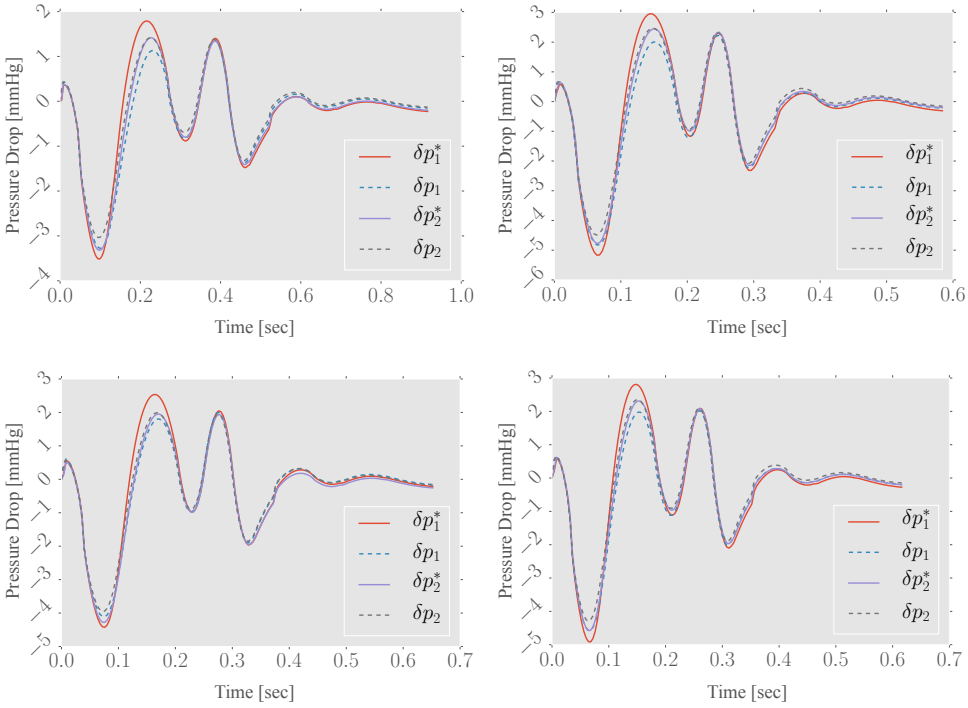


Figure 3.16: Pressure drops δp_1^* and δp_2^* with virtual work principle for noise-free measures. We observe the time evolution for 4 test cases in the two carotid branches, and a comparison with the ground truths δp_1 and δp_2. Vertical axis shows the drop in [mmHg], whereas the horizontal axis shows the time evolution in seconds.

3.7 Noisy measurements

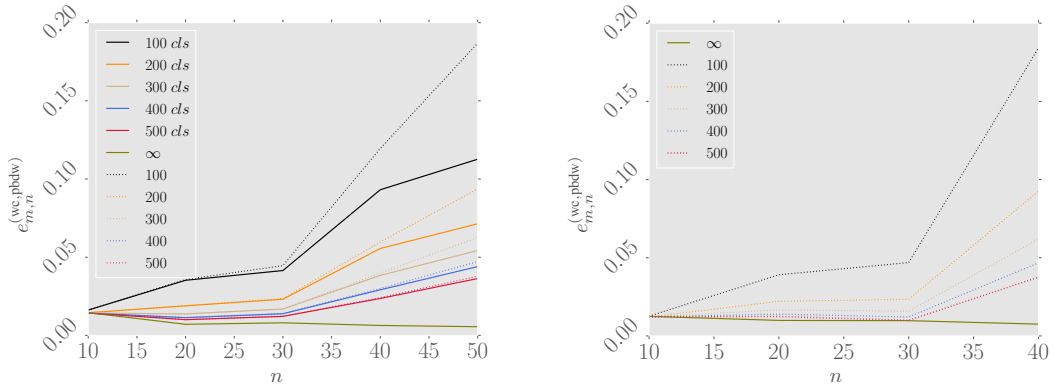
There are multiple sources of noise in CFI images, e.g., fake echo, reverberation, speckle, side lobes, ghosting, which result in a complicated space-time structure of the noise (see [Ledoux et al., 1997, Bjaerum et al., 2002, Demené et al., 2015]). Here we study the effect of noise in the admittedly simple case where we assume a gaussian perturbation of our observations of the form

$$z_i = \ell_i(u) + \eta_i \tag{3.37}$$

where $\eta_i \sim \mathcal{N}(0, \sigma^2)$. The noise is then independent at each voxel, averaged on the perfect measures. The standard deviation is chosen relative to the synthetic measures as

$$\sigma = \frac{\max_t \max_{i=1,\dots,m} \ell_i(u(t))}{\tilde{\alpha}},$$

where $\tilde{\alpha} \in \mathbb{R}$ is a parameter that steers the *noise level*.



(a) $V = U$.

(b) $V = U \times P$.

Figure 3.17: Reconstruction error (2.8) in one of the manifold partitions. Dots: unconstrained approach (3.40). Full line: constrained approach (3.42). Curves for the constrained and unconstrained approach overlap for the joint reconstruction in $V = U \times P$, showing that the constraints do not bring any improvement.

As already observed in previous works, a naive reconstruction with the PBDW method with the noisy measurements $(z_i)_{i=1}^m$ is not asymptotically robust in the sense that when number m of observations increases, the error bounds degrade essentially like $\sqrt{m}\sigma$. This has motivated the search for more stable formulations, and several approaches based on different types of regularization and thresholding have been proposed (see [Taddei, 2017, Argaud et al., 2017, Gong et al., 2019]). Here we consider a simple variant based on a thresholding technique for $v^*_{m,n}$ (see equation (2.7)) in the spirit of [Argaud et al., 2017]. To explain it, we first need to recall that in the noiseless case, $v^*_{m,n}$ is the unique minimizer of:

$$v^*_{m,n} = \arg\min_{v \in V_n} \frac{1}{2} \|P_{W_m} u - P_{W_m} v\|^2. \tag{3.38}$$

A slightly different approach is to find $\tilde{v}^*_{m,n} \in V_n$ as

$$\tilde{v}^*_{m,n} = \arg\min_{v \in V_n} \frac{1}{2} \sum_{i=1}^{m} |\ell_i(u) - \ell_i(v)|^2_{\ell(\mathbb{R}^m)}. \tag{3.39}$$

We may note that, in general, $\tilde{v}^*_{m,n} \neq v^*_{m,n}$ except if $\{\omega_i\}_{i=1}^m$ is an orthonormal family in V. In presence of noise, we measure z_i and not $\ell_i(u)$ so the minimization becomes

$$\min_{v \in V_n} \frac{1}{2} \sum_{i=1}^{m} |z_i - \ell_i(v)|^2_{\ell(\mathbb{R}^m)}. \tag{3.40}$$

To make this reconstruction more robust againt noise, instead of minimizing over the whole space V_n, we can use the structure of the PDE solution manifold $\mathcal{M}$ and minimize over its "footprint" on V_n, that is,

$$\mathcal{K}_n = P_{V_n}\mathcal{M} \overset{\text{def}}{=} \{P_{V_n} u \,:\, u \in \mathcal{M}\}.$$

The resulting minimization reads

$$\hat{v}^*_{m,n} = \arg\min_{v \in \mathcal{K}_n} \frac{1}{2} \sum_{i=1}^{m} |z_i - \ell_i(v)|^2_{\ell(\mathbb{R}^m)}. \tag{3.41}$$

In practice, if $\{v_i\}_{i=1}^n$ is an orthonormal basis of the space V_n, we can compute the coefficients $\mathbf{c}^* \in \mathbb{R}^n$ of $\hat{v}^*_{m,n}$ in this basis by solving the constrained least-squares problem

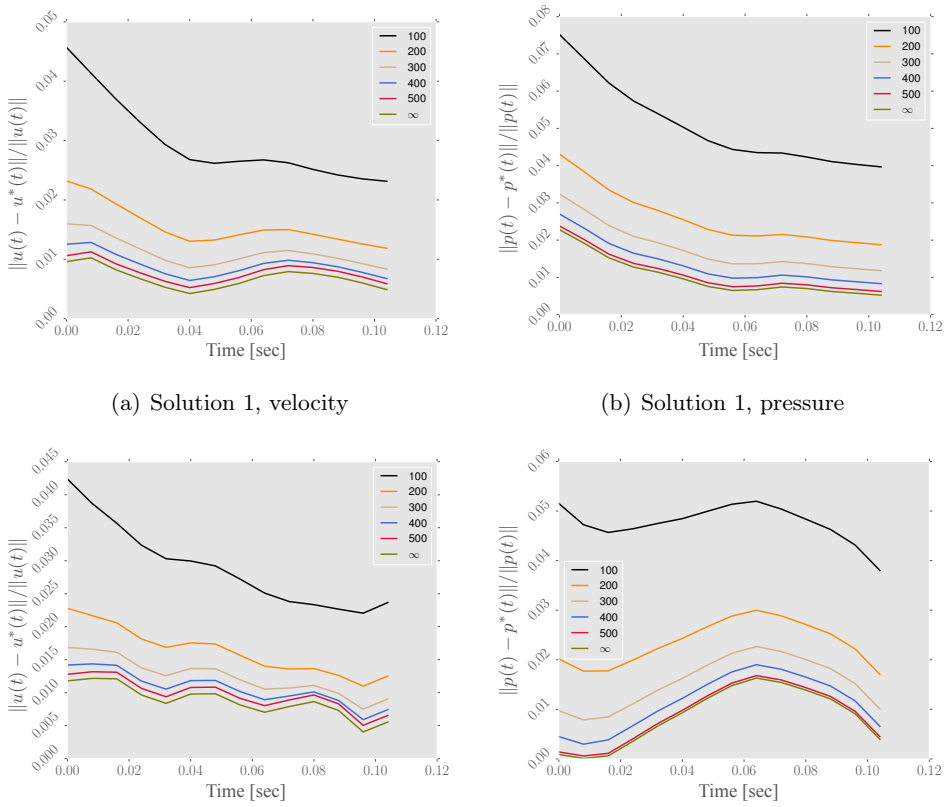
$$\begin{aligned} &\min_{\mathbf{c} \in \mathbb{R}^n} \frac{1}{2} \sum_{i=1}^{m} |z_i - \sum_{j=1}^{n} c_j \ell_i(v_j)|^2 \\ &\text{s.t. } |c_j| \leq \max_{u \in \mathcal{M}} |\langle u, v_j \rangle| \overset{\text{def}}{=} \delta_j, \quad j = 1, \dots n. \end{aligned} \tag{3.42}$$

This optimization problem is quadratic, with linear constraints and with a convex feasible region, a *box*. The optimality conditions for this kind of problems were first introduced by William Karush in his master's thesis in 1939 [Karush, 1939]. The steps to get the minimizer $c^*_{rls} \in \mathbb{R}^n$ starts with testing the trivial case, i.e., the case in which:

$$G^T G c^*_{rls} = G^T l. \tag{3.43}$$

where we have introduced the matrix of dimensions $m \times n$, a cross-Gramian between the spaces W_m and V_n. $l \in \mathbb{R}^m$ is a vector containing the Doppler data. We then should verify if $\forall i$, $|c^*_{rls,i}| < \delta_i$. If so, there are non active restriction, which leads to state that the following bullets are equivalent:

- c^*_{rls} is an interior point of $[-\delta_1, \delta_1] \times ... \times [-\delta_n, \delta_n]$.
- $\mu_i = 0$ for $i = 1, \dots, n$.
- The solution of the constrained problem is that of the pure least-squares (3.39). Thus, the model bias correction would be missing for this strategy.



(a) Solution 1, velocity

(b) Solution 1, pressure

(c) Solution 13, velocity

(d) Solution 13, pressure

Figure 3.18: Relative errors in joint reconstruction for two time dependent solutions. Figure 3.17(b) suggests us to use 30 modes in the reconstruction.

Notice that there are $2n$ restrictions, but at most n of them can be active simultaneously. If any of the constraints are violated, then we start by assuming that n of them, the conditions $c_i < \delta_i$, for $i = 1, \ldots, n$, *bind*. In general, assuming that there are k binding constraints, we can introduce a vector of KKT multipliers $\mu \in \mathbb{R}^k$ and write the Lagrangian,

$$\mathcal{J} = \|l - Gu\|^2_{\ell^2} + \mu^T(c - \delta)$$

Optimality conditions leads to the following saddle point problem:

$$\begin{bmatrix} G^TG & I_{n\times k} \\ I_{k\times n} & 0 \end{bmatrix} \begin{bmatrix} c \\ \mu \end{bmatrix} = \begin{bmatrix} G^Tl \\ \delta \end{bmatrix} \tag{3.44}$$

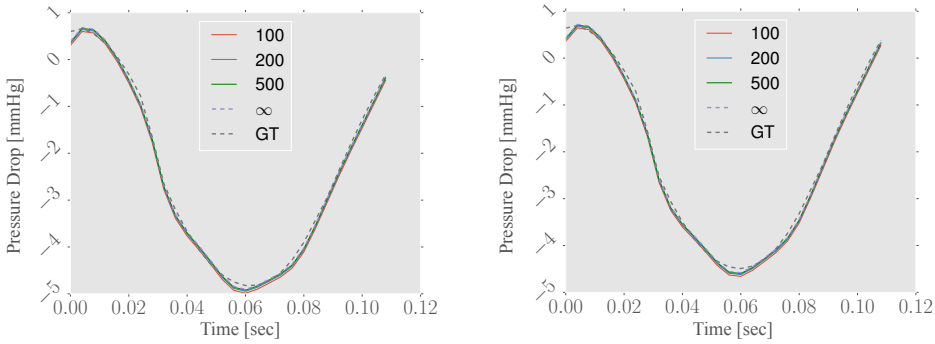
We next study the reconstruction error with the unconstrained and constrained approaches (3.40) and (3.42). Figures 3.17(a) and 3.17(b) show ively the error against the dimension n of V_n for the velocity reconstruction and velocity-pressure reconstruction. For each value of n, we compute the average error over 100 realizations of the noisy measurements for different levels $\tilde{\alpha}$ of the noise. The noiseless case is labeled $\alpha = \infty$ in the plots. The test case is focused on the first time partition, during the systolic phase of the cardiac cycle, and for patients in the lower heart rate partition. As expected, the quality of the reconstruction degrades when the level of the noise increases ($\tilde{\alpha}$ decreases). We observe that both constrained and unconstrained methods behave very similarly for a low number of modes. For the ambient space $V = U$, the constrained approach is able to grant a better reconstruction as we increase the dimension of the space V_n. However, for the ambient space $V = U \times P$, the constrained approach does not bring any improvement with respect to the unconstrained one.

Figures 3.18 and 3.19 show the reconstruction error for velocity and pressure fields, and the pressure drop computed from the joint reconstruction in $V = U \times P$. As in the noise-free numerical experiment, the reconstruction output is very satisfactory for all quantities. We observe that the pressure drop reconstruction is more robust to noise than the reconstruction of the full 3D pressure field although the reconstructed pressure drop is derived from the reconstructed 3D field. For all the cases, the reconstruction was done with a dimension $n = 30$ for V_n.

3.8 Conclusions and perspectives

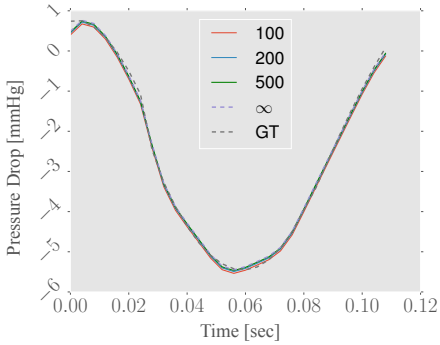
We conclude this chapter by summarizing the main topic and contributions. We have proposed a systematic methodology involving reduced modelling to give quick and reliable estimations of QoI in biological fluid flows. We have assessed the feasibility of the approach in non trivial numerical examples involving the carotid artery. The numerical examples include:

- The reconstruction of velocity related quantities such as vorticity and wall shear stress from Doppler data, both holding average errors below the 5% in an H^1 sense. At worst,

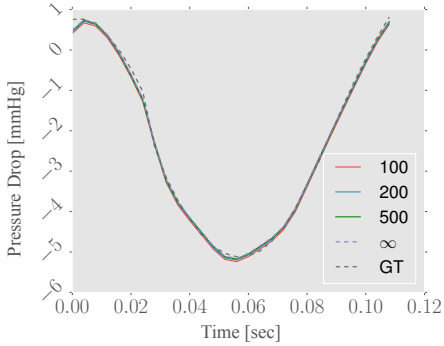


(a) Patient 1, Γ_o^1

(b) Patient 1, Γ_o^2



(c) Patient 13, Γ_o^1



(d) Patient 13, Γ_o^2

Figure 3.19: Reconstruction of pressure drop for 3 noise levels in two patients. The results are presented for the early systole phase. Vertical axis shows the presure drop in [mmHg] and the horizontal axis the time in seconds.

we have observed a maximal error during the cardiac cycle for one of the test cases below the 10%.

- The reconstruction of unobserved QoI from Doppler data such as pressure fields and pressure drops, with comparisons with other state-of-the-art techniques [Bertoglio et al., 2017b].
- The simulation of semi-realistic measures by considering white noise in the input signals.
- A theoretical study of the reconstruction error in all QoI. In particular, the numerical results confirm that the bound for the pressure drop estimation is rather sharp.

Although the present results are promising, they remain a proof-of-concept since the Doppler images and the flows serving as the ground truth are synthetically generated. To go further, we need to validate the methodology with real flows and real ultrasound images. This step poses however a certain number of challenges that we will eventually addressed in a collaboration involving medical doctors and experts in 3D printing. The main roadmap is: (i) to manufacture arteries with similar mechanical properties as biological ones, and favorable optical properties to collect ultrasound and PIV measurements. (ii) Once this is done, we will collect the ultrasound images and feed our reconstruction algorithms. We will compare our reconstruction with PIV images, which will serve as the ground truth.

Chapter 4

Inverse problems on non-parametric domains. Application to fluid mechanics using non linear dimensionality reduction

---

Solving in real time inverse problems for biomedical applications might require learning techniques that takes simulations and databases from different patients which inevitably involve anatomical variations. We present a state estimation method which allows to take this variability into account without needing any *a priori* knowledge on a parametrization of the anatomical differences. We rely on morphometric techniques involving Multidimensional Scaling and couple them with reconstruction techniques that make use of reduced modeling. We prove the potential of the method on a simple application inspired from the reconstruction of blood flows and quantities of medical interest with Doppler ultrasound imaging.

Contents

4.1 Introduction

In both, forward and inverse problems, MOR techniques relies on pre-built (*offline*) data bases that encodes the governing dynamics of the states. In Chapters 3 and 4 we have saw the efficiency of MOR techniques to address problems for a fixed working domain. We explore here how to deal with the scenario in which the targeted application does not allow us to know working domains *a-priori*. That is certainly the case of most of the bio-medical applications. For example, in [Manzoni et al., 2018], we see efficient model order reduction techniques for the mono-domain equations in the heart, but which are blocked by the challenge that is called there as *inter-patient variability*.

Inter-patient variability (or domain variability in general) is an urgent subject to address in order to make methodologies such as those of Chapter 3 and Chapter 4 to go from academic proof of concepts to medical or other applications in the industry. Furthermore, there is special attention on this matter in the scientific community, as its scope goes much beyond the applications listed in this thesis.

There is a number of works dedicated to non-paremetric domains. Nonetheless not much of the available literature concerns inverse problems. In spite that they address forward modeling, the problem of registration between geometries is fundamentally the same. In [Taddei, 2020], a registration method between parameterized working domains is proposed, which provides a systematic tool to compute mappings so that MOR techniques can be used in a domain known *a-priori*. Among its features, the registration method is said to be general in the sense that works independently of the governing dynamics. In contrast, we propose a methodology that gives prominent importance to the PDE that is used to generate the data base. We claim that is possible to build up metrics with this in mind in order to address inverse problems in non-parametric domains.

In [Akkari et al., 2019], we see an application in fluid dynamics in which the domain variations can be localized with an indicatrix function, so that high fidelity simulations are run inside certain zones of the domain, and reduced models are used wherever no domain variations are expected. This approach, proved up to a certain accuracy successful, is not well-adapted for bio-medical applications in which the geometrical variabilities are not that predictable. In [Guibert et al., 2013] we see forward modeling with reduced basis that, in spite of providing predictive power on quantities of interest such as flow rate or pressure jumps, the methodology lacks enough accuracy to recover the whole state.

The novelty of this Chapter can be summarized then to a method that propose to address the points of the previous paragraphs in order to solve inverse problems in non-parametric domains providing and sometimes theoretically guarantying state estimations up to accuracies that, as

far the author knows, has not been reported yet.

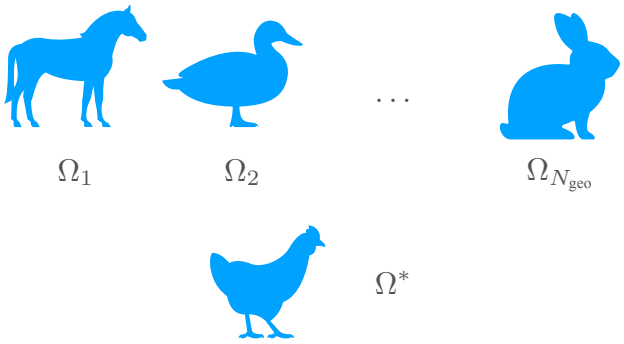


Figure 4.1: The problem of state estimation from pre-computed models in templates.

We introduce a pipeline to systematically solve inverse problems over non-parametric domains for blood flows. *In a nutshell*, we propose to circumvent the aforementioned problematic by building data-bases over a set of working domains, so called *templates*, that represent a broad spectra of configurations. The underlying idea is that given a new geometry, there will be at least one among those templates that will provide good reconstructions. Each building block of the methodology has to carefully address the following three points (see an scheme in figure 4.2):

1. First, a mechanism to map fields between geometries in such a way that certain physical properties are preserved has to be discussed.

2. Second, we need a proper way to identify a correct template given a new working domain. This requires to question the notion of distance between spatial domains by enforcing a metric space that somehow encodes geometrical information about euclidean proximity between cloud of points but also information about the underlying governing dynamics that are used to produce the data base on each template.

3. Third, a methodology about how to represent geometries in low-dimensional spaces that hopefully resembles a parameter space is introduced. We borrow here techniques from machine learning in order to find those representations. We can then build linear mappings to find out a template given a new working domain.

A final step in the pipeline is thus to connect the previous blocks with the reconstruction algorithms of previous chapters. Having all of that we present a 3D numerical example of low Reynolds non-convective flows reconstruction from synthetic Doppler data.

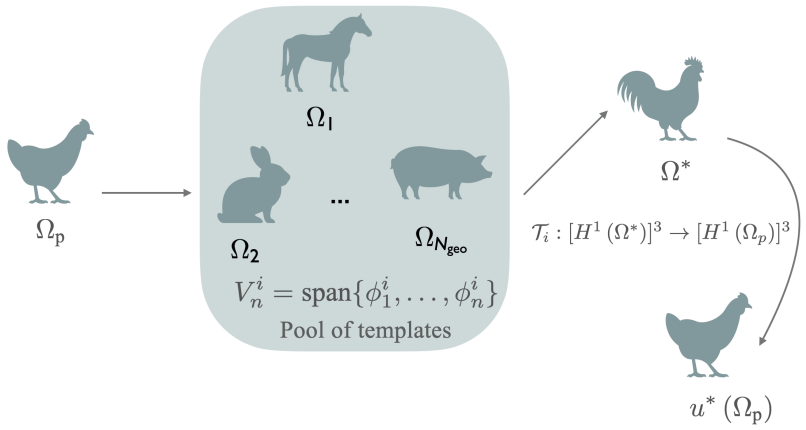


Figure 4.2: Multi-domain problem pipeline scheme.

4.1.1 Motivation and metrics induced by governing dynamics

Suppose we are given on the one hand partial observations about a state in a certain spatial domain $\Omega^* \in \mathbb{R}^d$ $(d = 2, 3)$, illustrated in figure 4.1. We have, on the other hand, model information on a pool of $N_{\text{geo}} \in \mathbb{N}$ domains $\{\Omega_1, \ldots, \Omega_{N_{\text{geo}}}\}$ called templates. The challenge is to *identify* which, among the templates, the duck, the horse, or the rabbit could somehow provide accurate information about the state on the given chicken. What is the state we are interested in? Say for instance that the word *bird* is what we would like to recover. Having said so is clear that duck is going to be the perfect match. Deep down, what we want to introduce here is the idea that every template contains some abstract information, sometimes provided with a model, that we would like to use to build up those *matches*.

Let us put ideas down with a fluid mechanics thought experiment designed to naturally dissociate the notions of geometrical distances from a model distance. In figure 4.3 we observe three domains in which we can imagine a flow is entering from the bottom boundary. It is easy to see that there is a certain geometrical symmetry of Ω_1 and Ω_3 with respect to Ω_2. Nevertheless, if the state to be recovered in a new geometry was the fluid flow, a purely geometrical criteria would lead us potentially to the wrong template. This small example should facilitate the discussion that we will bring up in this chapter in which we set up metrics based on distances between sets of solutions of fluid governing dynamics in the templates.

4.2 An operator between divergence free spaces

Let us recall the space of divergence free fields in $\Omega \subset \mathbb{R}^3$: $H(\text{div}; \Omega) = \{u \in [H^1(\Omega)]^3;\ \nabla \cdot u = 0\}$. Incompressibility in fluid flows is a fundamental hypothesis we have taken and is present inside

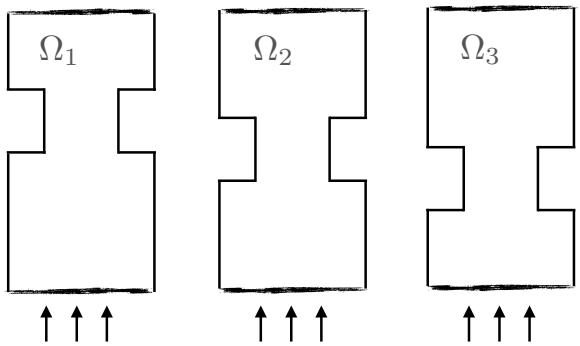


Figure 4.3: In a fluid mechanics problem is easy to see the discrepancy between flows in Ω_1, Ω_2 and Ω_3 in spite of the geometrical symmetry.

all the simulations of this document. It is then mandatory to have a reliable way to build an operator $\mathcal{T}_{i\mapsto j} : H(\mathrm{div};\Omega_i) \to H(\mathrm{div};\Omega_j)$, $u \mapsto (\mathcal{T}_{i\mapsto j}(u))$, so that mass conservation is ensured when moving a field from one geometry to another. We can divide the process of building $\mathcal{T}_{i\mapsto j}$ in four sub-steps, a methodology lended from [Guibert et al., 2013]:

- An optimization problem to find an invertible function $f_{i\mapsto j} : \partial\Omega_i \to \partial\Omega'$ that maps $\partial\Omega_i$ to a *auxiliary* surface $\partial\Omega'$, hopefully similar in a sense to $\partial\Omega_j$. This mapping should be designed in order to satisfy some suitable properties, such a certain smoothness and enough flexibility (non linearity) for large deformations. We have chosen to realize this step via the large deformation diffeomorphic metric matching (LDDMM) approach (see, for instance [Vaillant et al., 2004]).

- The extension of $f_{i\mapsto j}$ to the full domain Ω_i via the harmonic extension of $f_{i\mapsto j}$, say a map $\mathcal{A}_{i\mapsto j} : \Omega_i \to \Omega'$. This allows to define an operator $d_v : H(\mathrm{div};\Omega_i) \to \left[H^1(\Omega')\right]^3$, $u \mapsto \hat{u} \overset{\text{def}}{=} d_v(u)$, such that $d_v\left(u\left(x\right)\right) = u(\mathcal{A}^{-1}_{i\mapsto j}\left(x\right))$, for any $x \in \Omega'$.

- Build an interpolator $\mathcal{I}_{i\mapsto j} : \left[H^1(\Omega')\right]^3 \to \left[H^1(\Omega_j)\right]^3$, $\hat{u} \mapsto \mathcal{I}_{i\mapsto j}(\hat{u})$. The operator $\mathcal{I}_{i\mapsto j} \circ d_v$ is typically referred in computational anatomy as *3D shape registration* [Pears et al., 2012].

- The computation of a Piola transform $\mathcal{P} : \left[H^1(\Omega_j)\right]^3 \to \left[H^1(\Omega_j)\right]^3$ to enforce incompressibility [Rognes et al., 2009].

This steps will be discussed in sections 4.2.1, 4.2.2, and 4.2.3. Thus the desired operator is the composition $\mathcal{T}_{i\mapsto j} = \mathcal{P} \circ \mathcal{I}_{i\mapsto j} \circ d_v$.

4.2.1 Large Deformation Diffeomorphic Metric Matching Approach

Let us consider the tetrahedron tessellations Ω_i^h and Ω_j^h, meshes for Ω_i and Ω_j, respectively. For any tessellation Ω^h we denote $\partial\Omega^h$ the triangulation of the mesh surface. Let us characterize the triangulation $\partial\Omega_i^h$ with its vertices $\{x_k^{(i)}\}_{k=1}^{N_i}$, the centers of each triangle $\{c_k^{(i)}\}_{k=1}^{N_i^e}$, and its orientations (unitary normal vectors pointing outwards the working domain) $\{s_k^{(i)}\}_{k=1}^{N_i^e}$. $N_i \in \mathbb{N}^+$ and $N_i^e \in \mathbb{N}^+$ are the number of vertices and triangles on the surface mesh Ω_i^h, respectively. The notation for vertices, centers and orientations for Ω_j^h is analogous.

In addition, let us consider finite dimensional approximations of $\left[H^1(\partial\Omega_i)\right]^3$ and $\left[H^1(\partial\Omega_j)\right]^3$, $\mathcal{V}_i$ and $\mathcal{V}_j$, respectively. Although is not necessary, we will keep the discussion simple and close to what the computational implementation of MAD permits by choosing piece-wise linear Lagrange elements, so $\mathcal{V}_i = \mathbb{P}_1(\partial\Omega_i^h)$ and $\mathcal{V}_j = \mathbb{P}_1(\partial\Omega_i^h)$. We can project $f_{i\mapsto j}$ on $\mathcal{V}_i$, so we approximate

$$(f_{i\mapsto j})_l \approx \left(f_{i\mapsto j}^h\right)_l = \{P_{\mathcal{V}_i}(f_{i\mapsto j})\}_l = \sum_{k=1}^{N_i} \{f_{i\mapsto j}^{h,k}\}_l \mathcal{L}_k^b,$$

where $\mathcal{V}_i = \text{span}\{\mathcal{L}_1^b, \ldots, \mathcal{L}_{N_i}^b\}$, with $\{\mathcal{L}_k\}_{k=1}^{N_i}$ a $\mathbb{P}_1(\Omega_i^h)$ basis. We will write down LDDMM as a method for finding the vectors of coefficients

$$\left(\hat{f}^h\right)_l = \{\left(f_{i\mapsto j}^{h,1}\right)_l, \ldots, \left(f_{i\mapsto j}^{h,N_i}\right)_l\}^T,$$

for $l = 1, 2, 3$. In fact, LDDMM search for the pair position $q : \Omega_i^h \to \mathbb{R}^3$ and momentum $p : \Omega_i^h \to \mathbb{R}^3$ that minimizes:

$$\begin{aligned}\mathcal{L}(p,q) = &\sum_{k=1}^{N_i}\sum_{l=1}^{N_i} G(c_k^{(i)}, c_k^{(i)}) s_k^{(i)} \cdot s_k^{(i)} + \sum_{k=1}^{N_j^e}\sum_{l=1}^{N_j^e} G(c_k^{(j)}, c_k^{(j)}) s_k^{(j)} \cdot s_k^{(j)} \\ &- 2\sum_{k=1}^{N_i^e}\sum_{l=1}^{N_j^e} G(c_k^{(i)}, c_k^{(j)}) s_k^{(i)} \cdot s_k^{(j)} + \frac{\gamma}{2}\sum_{k=1}^{N_i}\sum_{l=1}^{N_j} p_i^T G(q_i, q_j) p_j,\end{aligned} \tag{4.1}$$

where, for any x and y in $\mathbb{R}^3$: $G(x,y) = e^{\frac{\|x-y\|}{\sigma^2}}$ and $x \cdot y = x^T y$. There are two free parameters, $\gamma > 0$, and $\sigma > 0$ can be seen as a way to control how rigid we allow the transformation to be. The underlying mechanics are the equations:

$$\begin{cases} \dfrac{dp}{dt} = -\dfrac{\partial H(q,p)}{\partial q}, \\ \dfrac{dq}{dt} = \dfrac{\partial H(q,p)}{\partial p}, \end{cases} \tag{4.2}$$

which are used in practice to update the position and momentum of the deformed mesh at each iteration, so t here is to be understood as an artificial time. The Hamiltonian H is defined as:

$$H(q,p) = \frac{1}{2} p^T e^{\|q_0 - q\|/\sigma^2} p.$$

Here q_0 denotes the first iteration of the displacement field, typically chosen to be the position vector of the vertices of the starting mesh $\partial\Omega_i^h$, i.e., $(q_0)_k = x_k$, $k = 1, \ldots, N_i$. Once the minimizer, say q^* is found, our displacement field is thus $\hat{f}^h = q^0 + q^*$. This problem is also referred to in the literature as *landmark matching*. A less general version of the presented method can be found in [Joshi and Miller, 2000].

4.2.2 Volumetric extension of LDDMM and mesh interpolation

In order to have a volumetric mapping, we consider an harmonic extension of the surface mapping $f_{i\mapsto j}$ provided by LDDMM. This reads as follows: find $d \in [H^1(\Omega_i)]^3$ such that:

$$\begin{cases} \Delta d = 0 & \text{in } \Omega_i, \\ d = f_{i\mapsto j}(x) & \text{on } \partial\Omega_i. \end{cases} \tag{4.3}$$

As in Chapter 2, here we sometimes do not distinguish the notation when changing from continuum to discrete when there is no place for confusion. Thus,

$$\mathcal{A}_{i\mapsto j}(x) = x + d(x), \text{ for } x \in \Omega_i. \tag{4.4}$$

The interpolation of a generic vector field $u : \Omega_i \to \mathbb{R}^3$ reads as follows:

1. We impose $\hat{u} \overset{\text{def}}{=} d_v(u) = u(\mathcal{A}_{i\mapsto j}^{-1}(x))$, $\forall x \in \Omega'$. Up to here we have a field expressed in the *auxiliary* mesh Ω'. ,

2. We interpolate the field to Ω_j: for any $x_k \in \Omega_j^h$ we can locate the tetrahedron in Ω' that contains it, with vertices $\{v_1^k, v_2^k, v_3^k, v_4^k\}$. Using standard Lagrange $\mathbb{P}_1$ interpolation functions $\{S_i\}_{i=1}^4$ for those vertices, we get the new approximation $\mathcal{I}_{i\mapsto j}(\hat{u})(x_k) = \sum_l S_l \hat{u}(v_l^k)$. This idea is schematically depicted in figure 4.4.

4.2.3 Transport of mass conservative fields

It is easy to see that in general $\mathcal{I}_{i\mapsto j} \circ d_v(u)$ is not divergence free even if u is. The pipeline needs a final step to assemble $\mathcal{T}_{i\mapsto j}$ in which we take this matter into account. Having said so, we introduce the Piola trasform $\mathcal{P} : \left[H^1(\Omega_j)\right]^3 \to \left[H^1(\Omega_j)\right]^3$:

$$\mathcal{P}(u) = \frac{\{I_{3\times 3} + \nabla\left[\mathcal{I}_{i\mapsto j} \circ d_v(d)\right]\}\mathcal{I}_{i\mapsto j} \circ d_v(u)}{\det\left(I_{3\times 3} + \nabla\left[\mathcal{I}_{i\mapsto j} \circ d_v(d)\right]\right)}, \tag{4.5}$$

for $u \in \left[H^1(\Omega_j)\right]^3$. This transformation ensures that (see, for instance [Ciarlet, 1988]),

$$\det\left(I_{3\times 3} + \nabla\left[\mathcal{I}_{i\mapsto j} \circ d_v(d)\right]\right) \nabla \cdot u = \nabla \cdot \left(\mathcal{I}_{i\mapsto j} \circ d_v(u)\right).$$

The operator can be finally written: $\mathcal{T}_{i\mapsto j}(u) = \mathcal{P} \circ \mathcal{I}_{i\mapsto j} \circ d_v(u)$.

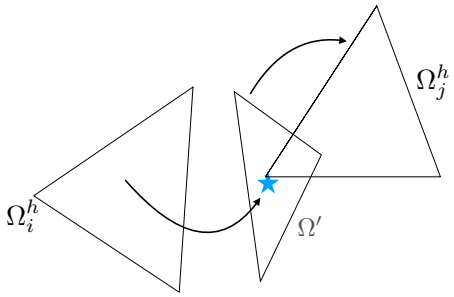


Figure 4.4: Scheme for $\mathbb{P}_1$ interpolation of fields from Ω' to Ω_j^h. We can locate the blue star (and all the other vertices) inside some element in Ω' and use the field information in it to interpolate.

4.2.4 Numerical example

Let us build a pool of geometries to test the methodology exposed in this section but that will serve as well for the numerical experiments of the whole Chapter. Consider a 3D Venturi tube as that of figure 4.5. The geometry is $L = 5$ cm long and it has a diameter $D = 0.4$ cm. We can parameterize the Venturi stenosis radius $S_r \in \mathbb{R}^+$, its length $S_l \in \mathbb{R}^+$ and its position $S_x \in \mathbb{R}^+$. A grid of $N_{\text{geo}} = 60$ working domains is generated by moving those 3 parameters in the following ranges: $S_r \in [1.4, 2.6]$ mm, $S_l \in \left[0.8\frac{L}{2}, 1.2\frac{L}{2}\right]$ and $S_x \in [5, 11]$ mm.

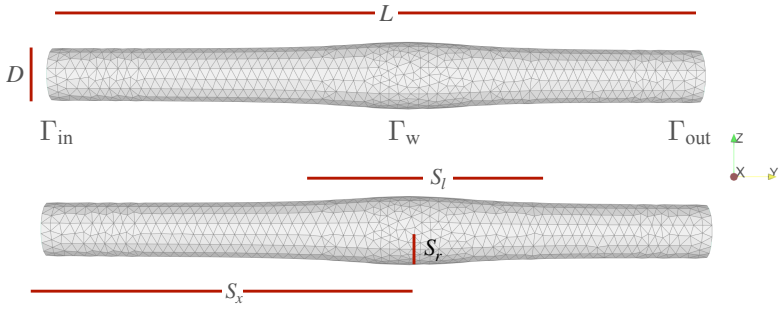


Figure 4.5: Working domain of the numerical experiment of this section and all the sections to come in this chapter. We see the domain labels and the domain parameters used to generate the templates. Mesh size is set to $h = 0.08$ cm.

We select two working domains to compute $\mathcal{T}_{i\mapsto j}$, those depicted in figure 4.6. The first step is to compute the coefficients of the projection of the surface registration $f_{i\mapsto j}$ in $\mathbb{P}_1(\partial\Omega_i)$. To do so, the functional (4.1) is minimized using one among the many extensions of Newton's algorithm: the Broyden−Fletcher−Goldfarb−Shanno (BFGS) method. The decay of the loss for fours cases is shown in figure 4.7. The case of interest for our numerical experiment is remarked

in the plot with blue.

(a) Ω_i (b) Ω_j

Figure 4.6: We have chosen two geometries from the pool of templates to test the numerical construction of $\mathcal{T}_{i\mapsto j}$.

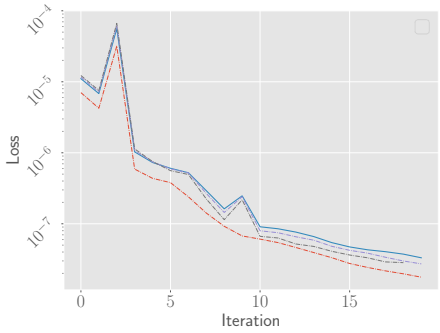
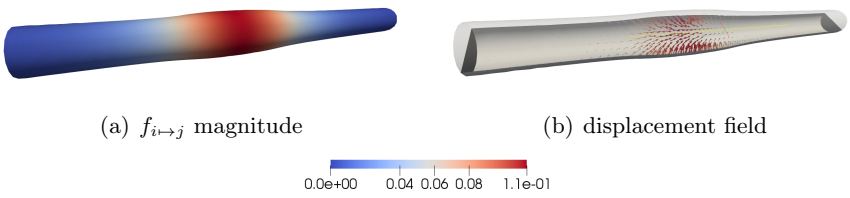


Figure 4.7: Loss (4.1) decay when computing $f_{i\mapsto j}$ for four pair of geometries. In particular, we see in the blue curve the decay for the domains we have chosen for this experiment, Ω_i and Ω_j from figure 4.6.

In addition, we can observe the displacement field obtained from the minimization in figure 4.8. The displacement field has some wished features: for instance, it is able to move the domain whereby the coarctation changes, whereas the rest of the domain remains untouched. This is not trivial to achieve from a numerical point of view. LDDMM has, as a drawback, the fact that it has free parameters. For instance, a bad choice for the *stiffness* σ in equation (4.1) could lead to rigid transformations that would not capture the geometrical variations. For our numerical experiment we have found that a good choice is $\sigma = 0.05$ and $\gamma = 0.0$. In addition, the input mesh Ω_i^h is scaled by a factor 0.1 in order to get the results shown in the figures.



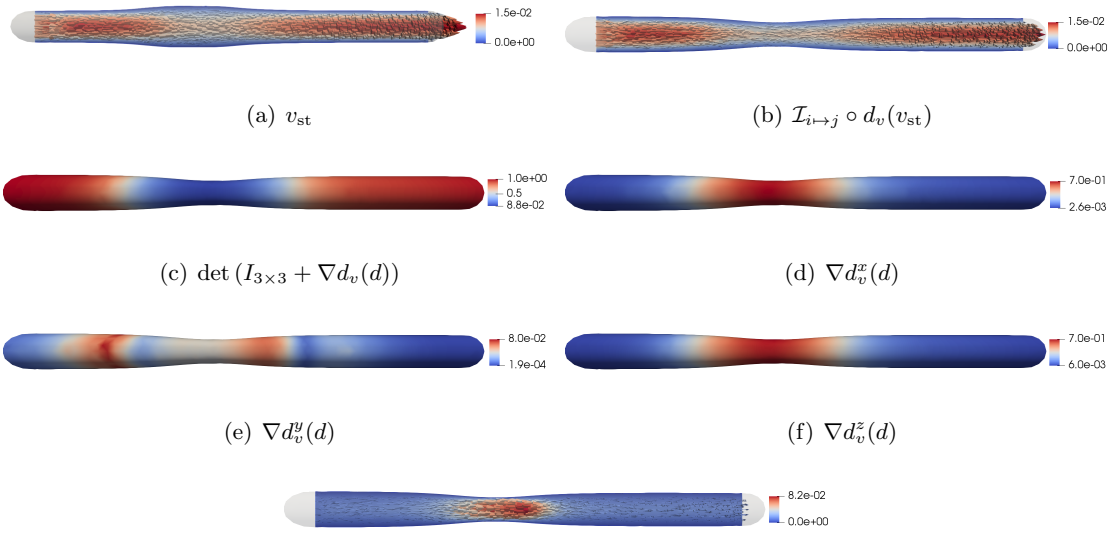
(a) $f_{i\mapsto j}$ magnitude (b) displacement field

Figure 4.8: $f_{i\mapsto j}$ projected on $\mathbb{P}_1\left(\partial\Omega_i^h\right)$.

In order to test the pipeline, let us compute a solution of the following Stokes problem: find $v \in [H^1(\Omega)]^3$ and $\lambda \in L^2(\Omega)$ such that:

$$\begin{cases} \partial_t v - \Delta v + \nabla \lambda = 0 \text{ in } \Omega_i, \\ \nabla \cdot v = 0 \text{ in } \Omega_i, \\ v = (0,0,0) \text{ on } \Gamma_{\text{w}}, \\ v = (0,1,0) \text{ on } \Gamma_{\text{in}}, \\ \left(\dfrac{\nabla v + \nabla^T v}{2} - \lambda I_{3\times 3} \right) \cdot n = (0,0,0) \text{ on } \Gamma_{\text{out}}. \end{cases} \tag{4.6}$$

We can test the full pipeline by computing the operator $\mathcal{T}_{i \mapsto j}(v_{\text{st}})$ for v_{st}, a solution of the problem stated above once steady regime is reached. In short, in figure 4.9 we see three things: The steady field $v_{\text{st}} \in H(\text{div}; \Omega_i)$, its interpolation in $\mathcal{I}_{i \mapsto j} \circ d_v(v_{\text{st}}) \in \left[H^1(\Omega_j)\right]^3$ and its Piola transform $\mathcal{T}_{i \mapsto j}(v_{\text{st}}) = \mathcal{P} \circ \mathcal{I}_{i \mapsto j} \circ d_v(v_{\text{st}}) \in H(\text{div}; \Omega_j)$. To find the time marching solution of the problems we use MAD with the incremental fractional step method exposed in section 5.4.1.3 and a BDF2 time scheme.



(a) v_{st} (b) $\mathcal{I}_{i \mapsto j} \circ d_v(v_{\text{st}})$

(c) $\det\left(I_{3\times 3} + \nabla d_v(d)\right)$ (d) $\nabla d_v^x(d)$

(e) $\nabla d_v^y(d)$ (f) $\nabla d_v^z(d)$

(g) $\mathcal{P} \circ \mathcal{I}_{i \mapsto j} \circ d_v(v_{\text{st}})$

Figure 4.9: Full pipeline of field transport v_{st} from Ω_i to Ω_j.

In figure 4.9 we see the full pipeline. First, the steady field, solution of (4.6) is depicted in 4.9(a). Next, the interpolation via LDDMM plus harmonic extension is seen in 4.9(b). We see with figures 4.9(c), 4.9(d), 4.9(e), 4.9(f) the mechanism of the Piola transform with the components of $\nabla d_v(d)$ and the scaling factor $\det(I_{3\times 3} + \nabla d_v(d))$. By looking at the components of $\nabla d_v(d)$ it is possible to see that $\nabla d_v^x(d)$ and $\nabla d_v^z(d)$ stands for the radius r_c variation from

one geometry to the other, where as $\nabla d_v^y(d)$ reveals a change in the position S_x of the section alteration.

In short, we see how the transformation manages to capture and counter the zones in which the geometry variations would induce a loss of mass conservation. As a consequence, we get a divergence free field as depicted in figure 4.9(g). This is very easy to see when thinking the conservation laws with more rudimentary but useful equations. Let us call $\bar{v}_{\text{in}}$ and $\bar{A}_{\text{in}}$ the average velocity and section at the inlet boundary, and let us call $\bar{v}_{\text{mid}}$ and $\bar{A}_{\text{mid}}$ the average velocity and section at a cut orthogonal to the fluid direction at the tubes coarctation/dilatation. If we keep in mind that what we expect is $\bar{v}_{\text{mid}}\bar{A}_{\text{mid}} \approx \bar{v}_{\text{in}}\bar{A}_{\text{in}}$, then it is easy to see the quality of the transported field in terms of its consistency with the governing dynamics of the flow.

This numerical experiment has been carried out with MAD interfacing with PyKeops/Torch for the computation of the LDDMM map.

4.3 Discussion about distance between working domains

This section is devoted to provide tools to evaluate distances between geometries by means of Hilbert spaces defined on them. Different alternatives to evaluate the proximity between sets of solutions in different working domains are explored. The discussion is kept abstract until the next section, where the models are going to be explicitly introduced via singular value decompositions of low Reynolds number flows.

Let us recall that the core idea of this Chapter is to introduce a metric based on some underlying governing dynamic of the state we are aiming to recover. Consider, again, two working domains $\Omega_i \subset \mathbb{R}^3$ and $\Omega_j \subset \mathbb{R}^3$. Let us consider n-dimensional ($n < \infty$) sub-sets E and F of a metric space (V, d). We might think of E and F as sets of solutions in Ω_i and Ω_j, respectively, or reduced models of those sets of solutions as well.

In section 4.2.3 we have already developed a method to manipulate F as a subset of the ambient space V (or vice-versa). If it turns out that we want to write down F, a finite dimensional space, as a subset of V, the task would be as straightforward as using operators $\mathcal{T}_{i\mapsto j}$ and $\mathcal{T}_{j\mapsto i}$ for some basis of F. Otherwise stated, given a basis $\{f_k\}_{k=1}^n$ of F, we denote:

$$\mathcal{T}_{j\mapsto i}(F) \stackrel{\text{def}}{=} \text{span}\{\mathcal{T}_{j\mapsto i}(f_1), \ldots, \mathcal{T}_{j\mapsto i}(f_n)\}.$$

which is assumed to be re-orthonormalized at the numerical level via modified Gram-Schmidt (mGS) algorithm.

4.3.1 Finite dimensional Grassmannian distances

A number of choices are possible when it comes to evaluate the proximity between two finite-dimensional sub-spaces E and $\mathcal{T}_{j\mapsto i}(F)$ of an ambient space V. A *Grassmannian* $\mathrm{Gr}(n,\mathcal{N})$, refers to hyperplanes of dimension n in larger sub-spaces embedded in $\mathbb{R}^{\mathcal{N}}$, with $n<\mathcal{N}$. We can view our linear reduced models, the spaces generated by the reduced basis on each templates, as some of those hyperplanes. A complete discussion about Grassmannians and generalizations to infinite-dimensional cases can be found in [Ye and Lim, 2016].

Let us consider matrices $A\in\mathbb{R}^{\mathcal{N}\times n}$ and $B\in\mathbb{R}^{\mathcal{N}\times n}$, whose columns are $\mathcal{N}$-dimensional basis vectors of E and $\mathcal{T}_{j\mapsto i}(F)$, respectively. We are interested in the singular value decomposition of the cross-Gramian

$$G(A,B)=A^TMB, \tag{4.7}$$

where $M\in\mathbb{R}^{\mathcal{N}\times\mathcal{N}}$ denotes a discretization matrix encoding the inner product of the ambient space V. For instance, in ℓ^2 we have $M_{kl}=\delta_{kl}$ or in $\left[H^1(\Omega)\right]^3$ we have $M_{kl}=\int\mathcal{L}_k\mathcal{L}_l\,\mathrm{d}x+\int\nabla\mathcal{L}_k\cdot\nabla\mathcal{L}_l\,\mathrm{d}x$, for $\{\mathcal{L}_k\}_{k=1}^{\mathcal{N}}$ basis functions of some $\mathcal{N}$-dimensional approximation of V.

Let us consider the full singular value decomposition:

$$G(A,B)=USV^T,$$

with $U^TU=I_{n\times n}$ and $V^TV=I_{n\times n}$, and assume that $S_{ii}>0$ for $i=1,\dots,n$. We can encode the proximity between the spaces by looking at the principal angles:

$$\theta_i=\cos^{-1}S_{ii}.$$

for $i=1,\dots,n$.

Table 4.1: Distance between sub-spaces expressed in terms of principal angles.

Asimov	$d^{\alpha}(E,\mathcal{T}_{j\mapsto i}(F))\overset{\text{def}}{=}\theta_n$
Binet-Cauchy	$d^{\beta}(E,\mathcal{T}_{j\mapsto i}(F))\overset{\text{def}}{=}\left(1-\Pi_{i=1}^{n}\cos^2(\theta_i)\right)^{1/2}$
Chordal	$d^{\gamma}(E,\mathcal{T}_{j\mapsto i}(F))\overset{\text{def}}{=}\left(\sum_{i=1}^{n}\sin^2(\theta_i)\right)^{1/2}$
Fubini-Study	$d^{\epsilon}(E,\mathcal{T}_{j\mapsto i}(F))\overset{\text{def}}{=}\cos^{-1}\left(\Pi_{i=1}^{n}\cos(\theta_i)\right)$
Martin	$d^{\delta}(E,\mathcal{T}_{j\mapsto i}(F))\overset{\text{def}}{=}\left(\log\Pi_{i=1}^{n}1/\cos^2(\theta_i)\right)^{1/2}$
Procrustes	$d^{\zeta}(E,\mathcal{T}_{j\mapsto i}(F))\overset{\text{def}}{=}2\left(\sum_{i=1}^{n}\sin^2(\theta_i/2)\right)^{1/2}$
Projection	$d^{\eta}(E,\mathcal{T}_{j\mapsto i}(F))\overset{\text{def}}{=}\sin\theta_n$
Spectral	$d^{\iota}(E,\mathcal{T}_{j\mapsto i}(F))\overset{\text{def}}{=}2\sin\left(\theta_n/2\right)$

All distances enlisted in Table 4.1 provide some sort of intuition concerning the orientation of one space with respect to the other. Nonetheless, these distances are not equivalent and there

are some subtle differences between all of them. For instance, there are some *Hausdorff-like* distances such as d^{α} and d^{η}, in which the distance criteria relies solely on how close $G(A,B)$ is to the space of singular matrices. On the contrary, d^{γ} and d^{ζ} give some global information, more closer to the trace of the matrix S, in which even spaces that are very far at high frequencies can be close in the kind of big picture this distances gives.

4.3.2 Hausdorff distance in the context of recovery algorithms

The Hausdorff distance between E and $\mathcal{T}_{j\mapsto i}(F)$, sub-spaces of the metric space (V,d) is defined as:

$$d_H(E,\mathcal{T}_{j\mapsto i}(F)) \stackrel{\text{def}}{=} \max\{\sup_{x\in E}\inf_{y\in\mathcal{T}_{j\mapsto i}(F)} d(x,y), \sup_{x\in\mathcal{T}_{j\mapsto i}(F)}\inf_{y\in E} d(x,y)\}. \tag{4.8}$$

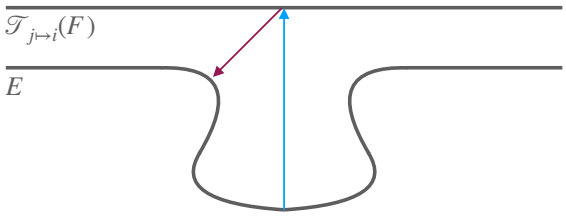


Figure 4.10: Hausdorff distance scheme.

In figure 4.10 we can grasp the idea behind this definition. It is about looking at the shortest distance between each point of one space to the other space as a whole and taking the larger among those shortest distances. Then repeat the process in the inverse sense and pick the larger value. Let us think about the Hausdorff distance between a straight line and a curve. In figure 4.10, the light blue arrow shows the *larger shortest* path from E to $\mathcal{T}_{j\mapsto i}(E)$, whose *length* is:

$$\sup_{x\in E}\inf_{y\in\mathcal{T}_{j\mapsto i}(F)} d(x,y).$$

where as the maroon arrow has a length

$$\sup_{x\in\mathcal{T}_{j\mapsto i}(F)}\inf_{y\in E} d(x,y),$$

Let us recall the following definition from section 2.2.1: given two closed normed sub-spaces X and Y of (V,d), we define:

$$\beta(X,Y) = \inf_{v\in X}\frac{\|P_Y v\|}{\|v\|}.$$

Proposition 4.3.1. *The Hausdorff distance* (4.8) *between unitary spheres* $S_X = \{x \in X;\ \|x\|_X = 1\}$ *and* $S_Y = \{y \in Y;\ \|y\|_Y = 1\}$ *can be written as:*

$$d_H(S_X,S_Y) = \max\{\sqrt{1-\beta^2(S_X,S_Y)},\ \sqrt{1-\beta^2(S_Y,S_X)}\}. \tag{4.9}$$

Proof. We can work out the proof with one of the terms inside the max brackets and then proceed analogously for the other. We can see that

$$\begin{aligned}\sup_{x\in S_X}\inf_{y\in S_Y} d^2(x,y) &= \sup_{x\in S_X}\|x - P_{S_Y}x\| \\ &= 1 - \inf_{x\in S_X}\|P_{S_Y}x\|^2 \\ &= 1-\beta^2(S_X,S_Y),\end{aligned}$$

from which (4.9) follows. First row is just to rephrase the distance from a point to a space in terms of orthogonal projections. Second row comes naturally since $\|x\|$, $\|x-P_{S_Y}y\|$ and $\|P_{S_Y}x\|$ is a Pythagorean triplet by construction. Next we use typical properties of sup in set theory and the fact that the elements belongs to unitary spheres. □

It is easy to see why we would like to work with distances between spheres when putting ideas down with a POD. In such case, what we would be doing is to compare how different are the ellipsoids mapped from the unity sphere in different working domains. This measure of distance is very consistent with our final target which is the construction of recovery algorithms using linear reduced models of non linear PDEs.

We next provide a theoretical foundation to link the distance between templates and the reconstruction quality of inverse problems using PBDW [Maday et al., 2015b]. The interest relies in finding a map $A : W_m \to V$, with W_m a space of observations. Let us assume A is the algorithm introduced in section 2.2.2.1. The natural question that arises at this point is: Given an upper bound for the distance between two models $(E, \mathcal{T}_{j\mapsto i}(F)) < \delta$, can we ensure some sort of reconstruction quality when doing the model reduction with $\mathcal{T}_{j\mapsto i}(F)$?

Proposition 4.3.2. *Let us consider two PBDW reconstruction algorithms $A_E : W_m \to V$, and $A_{\mathcal{T}_{j\mapsto i}(F)} : W_m \to V$, and let $u \in V$ be the state we would like to recover when we are given $\omega = P_{W_m}u$, so that $A_E : \omega \mapsto A_E(\omega)$ and $A_{\mathcal{T}_{j\mapsto i}(F)} : \omega \mapsto A_{\mathcal{T}_{j\mapsto i}(F)}(\omega)$.*

Provided $\operatorname{dist}(u,E) \le \varepsilon$ *and* $d_H(E,\mathcal{T}_{j\mapsto i}(F)) \le \delta$, *with* $\delta > 0$ *and* $\varepsilon > 0$, *it holds that, for* $\rho \in [0,1]$*:*

$$\|u - A_{\mathcal{T}_{j\mapsto i}(F)}(\omega)\| \le \varepsilon\left(1 - \{(1+\rho)\,\delta + (1+\rho^{-1})\left(1-\beta^2(E,W_m)\right)\}\right)^{1/2}(1+2\delta) \tag{4.10}$$

Proof. From (2.8) we know that:

$$\begin{aligned}\|u - A_E(P_{W_m}u)\| &\le \beta^{-1}(E,W_m)\operatorname{dist}(u,E), \text{ and} \\ \|u - A_{\mathcal{T}_{j\mapsto i}(F)}(P_{W_m}u)\| &\le \beta^{-1}(\mathcal{T}_{j\mapsto i}(F),W_m)\operatorname{dist}(u,\mathcal{T}_{j\mapsto i}(F)),\end{aligned}$$

The problem is thus reduced to find an upper bound for $\operatorname{dist}(u,\mathcal{T}_{j\mapsto i}(F))$ and a lower bound for $\beta(\mathcal{T}_{j\mapsto i}(F),W_m)$, both in function of the known quantities δ and ε. The former is straightfor-

ward. Using triangular inequality we get,

$$\begin{aligned}\operatorname{dist}(u, \mathcal{T}_{j\mapsto i}(F)) &\leq \|u - P_{\mathcal{T}_{j\mapsto i}(F)}u\| + \|P_E u - P_{\mathcal{T}_{j\mapsto i}(F)}u\| \\ &\leq \|u - P_{\mathcal{T}_{j\mapsto i}(F)}u\| + 2\delta\|P_{E\oplus\mathcal{T}_{j\mapsto i}(F)}u\| \\ &\leq \varepsilon(1+2\delta).\end{aligned}$$

For the latter, it follows from definition that:

$$\beta^2\left(\mathcal{T}_{j\mapsto i}(F), W_m\right) = 1 - \sup_{v\in\mathcal{T}_{j\mapsto i}(F)} \frac{\|v - P_{W_m}v\|^2}{\|v\|^2}. \tag{4.11}$$

The convexity of the supremum argument allows to use Jensen's inequality [Jensen, 1906]. Then, for $\rho \in [0,1]$ it is true that,

$$\frac{\|v - P_{W_m}v\|^2}{\|v\|^2} \leq (1+\rho)\frac{\|v - z - P_{W_m}(v-z)\|^2}{\|z\|^2} + (1+\rho^{-1})\frac{\|z - P_{W_m}z\|^2}{\|v\|^2}, \quad \forall z \in E.$$

Since we can do the following trick:

$$\begin{aligned}\frac{\|z - P_{W_m}z\|^2}{\|v\|^2} &\leq \sup_{z\in E}\frac{\|z - P_{W_m}z\|^2}{\|z\|^2}\frac{\|z\|^2}{\|v\|^2}, \quad \forall z \in E, \\ &= (1 - \beta^2(E, W_m))\frac{\|z\|^2}{\|v\|^2}, \quad \forall z \in E,\end{aligned}$$

and in addition:

$$\frac{\|v - z - P_{W_m}(v-z)\|^2}{\|z\|^2} \leq \frac{\|v-z\|^2}{\|z\|^2}, \quad \forall z \in E.$$

we have therefore an inequality to bound the supremum argument in (4.11) valid $\forall z \in E$. If we particularize $z = P_E v$ we finally get:

$$\begin{aligned}\frac{\|v - P_{W_m}v\|^2}{\|v\|^2} &\leq (1+\rho)\,\frac{\|v - P_E(v)\|^2}{\|v\|^2} + \left(1+\rho^{-1}\right)\left(1 - \beta^2(E, W_m)\right)\frac{\|P_E(v)\|^2}{\|v\|^2}, \\ &\leq (1+\rho)\sup_{v\in\mathcal{T}_{j\mapsto i}(F)}\frac{\|v - P_E(v)\|^2}{\|v\|^2} + \left(1+\rho^{-1}\right)\left(1 - \beta^2(E, W_m)\right)\sup_{v\in\mathcal{T}_{j\mapsto i}(F)}\frac{\|P_E(v)\|^2}{\|v\|^2}, \\ &\leq (1+\rho)\left(1 - \beta^2\left(\mathcal{T}_{j\mapsto i}(F), E\right)\right) + \left(1+\rho^{-1}\right)\left(1 - \beta^2(E, W_m)\right).\end{aligned}$$

Thus, plugging the last inequality in (4.11) we have found the lower bound we have searched for. □

4.3.3 Fluid dynamics experiment using singular value decompositions

We set up a numerical experiment to compute reduced models and the distances exposed in this sections. Their efficiency in the reconstruction task will be evaluated in the next section.

The training data base is built using governing PDEs for non-convective 3D blood flows, realizations of the following Stokes problem: find $u \in [H^1(\Omega)]^3$ and $p \in L^2(\Omega)$ such that:

$$\begin{cases} \partial_t u - \mu \Delta u + \nabla p = 0 \text{ in } \Omega_i, \\ \nabla \cdot u = 0 \text{ in } \Omega_i, \\ u = (0,0,0) \text{ on } \Gamma_{\text{w}}, \\ u = u_{\text{in}}\{0, \left(1 - \dfrac{x^2+z^2}{(D/2)^2}\right), 0\} \sin(2\pi t) \text{ on } \Gamma_{\text{in}}, \\ \left(\dfrac{\nabla u + \nabla^T u}{2} - p I_{3\times 3}\right) \cdot n = (0,0,0) \text{ on } \Gamma_{\text{out}}. \end{cases} \tag{4.12}$$

for $t \in [0, 0.5]$ [sec], and for all the pool of geometries constructed in section 4.2.4, i.e., $i = 1, \ldots, N_{\text{geo}}$. In addition, the parameter space includes the dynamic viscosity in the range $\mu \in [0.01, 0.1]$ [Poise] and the maximal value of the inlet paraboloid for the Dirichlet boundary condition: $u_{\text{in}} \in [0.01, 0.1]$ [cm/s].

FEM solutions of (4.12) are obtained with MAD using a time step of 0.02 [sec]. Each simulation will therefore provide 25 snapshots. For every template Ω_i we run 512 simulations, so we have a training set $\mathcal{M}^{(i)}_{\text{train}}$ of $N_s^{(i)} \overset{\text{def}}{=} \#\mathcal{M}^{(i)}_{\text{train}} = 12800$ snapshots, which are used to build reduced model spaces $\{V_1, \ldots, V_{N_{\text{geo}}}\}$. In particular, constraining the set of solutions to belong to $L^2(\Omega)$, we build N_{geo} singular value decompositions, one per working domain.

(a) Ω_7

(b) Ω_{14}

(c) Ω_{32}

(d) Ω_{60}

Figure 4.11: POD modes using $V = L^2(\Omega_i)$ in four working domains. First modes at the left side and second modes at the right side.

Before computing distances between reduced models we need to fix a dimension for them, which we choose to adapt according to the forward modeling approximation error. This has to

be carefully done since we are not considering $\beta(V_i, W_m)$ in the criteria. We have chosen not to use $\beta(V_i, W_m)$ because that would require to fix the space of observations W_m at offline phase, which is too restrictive in view of the medical applications we would like to tackle. Having said so, let $A_i \in \mathbb{R}^{\mathcal{N}^{(i)} \times N_s^i}$ be the snapshot matrix of the working domain Ω_i, whose tessellation has $N^{(i)} \stackrel{\text{def}}{=} \mathcal{N}^{(i)}/3$ vertices. Each column of the matrix correspond to one realization of the data set $\{u_k^{(i)}\}_{k=1}^{N_s^{(i)}}$.

Consider the SVD of $A_i = USV^T$, which is computed in parallel by MAD doing the eigenvalue decomposition of the covariance $A_i A_i^T$, with singular values $\sigma_1 > \sigma_2 > \ldots > \sigma_r > \ldots > 0$ (with $r \leq N_s^{(i)}$). Let us build an n-dimensional base for V_i, so we can generate the space doing $V_i = \text{span}\{\phi_1^{(i)}, \ldots, \phi_n^{(i)}\}$. It is well known that (see, for instance, [Trefethen and Bau, 1997]):

$$\|A_i - \Phi_i \Phi_i^T M A\|_{\text{F}} \leq \left(\sum_{j=n+1}^{r} \sigma_j^2 \right)^{1/2}, \tag{4.13}$$

where we stack the basis in the columns of the matrix $\Phi_i \in \mathbb{R}^{\mathcal{N} \times n}$, and M encodes the discretization of the ambient space, as in equation (4.7). This systematic although risky approach can be seen as a low-pass filter, or a gate function in the domain of the POD coefficients.

We see the first two modes of four geometries in figure 4.11. The goal is now to compute all the distances exposed in this section. The decay of the eigen-values is shown in figure 4.12.

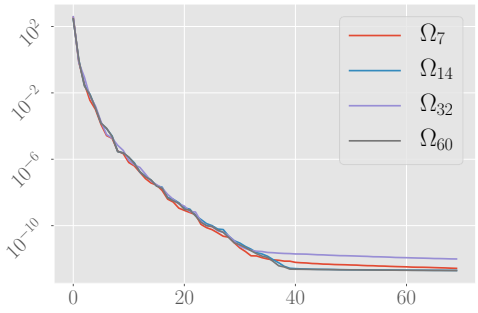


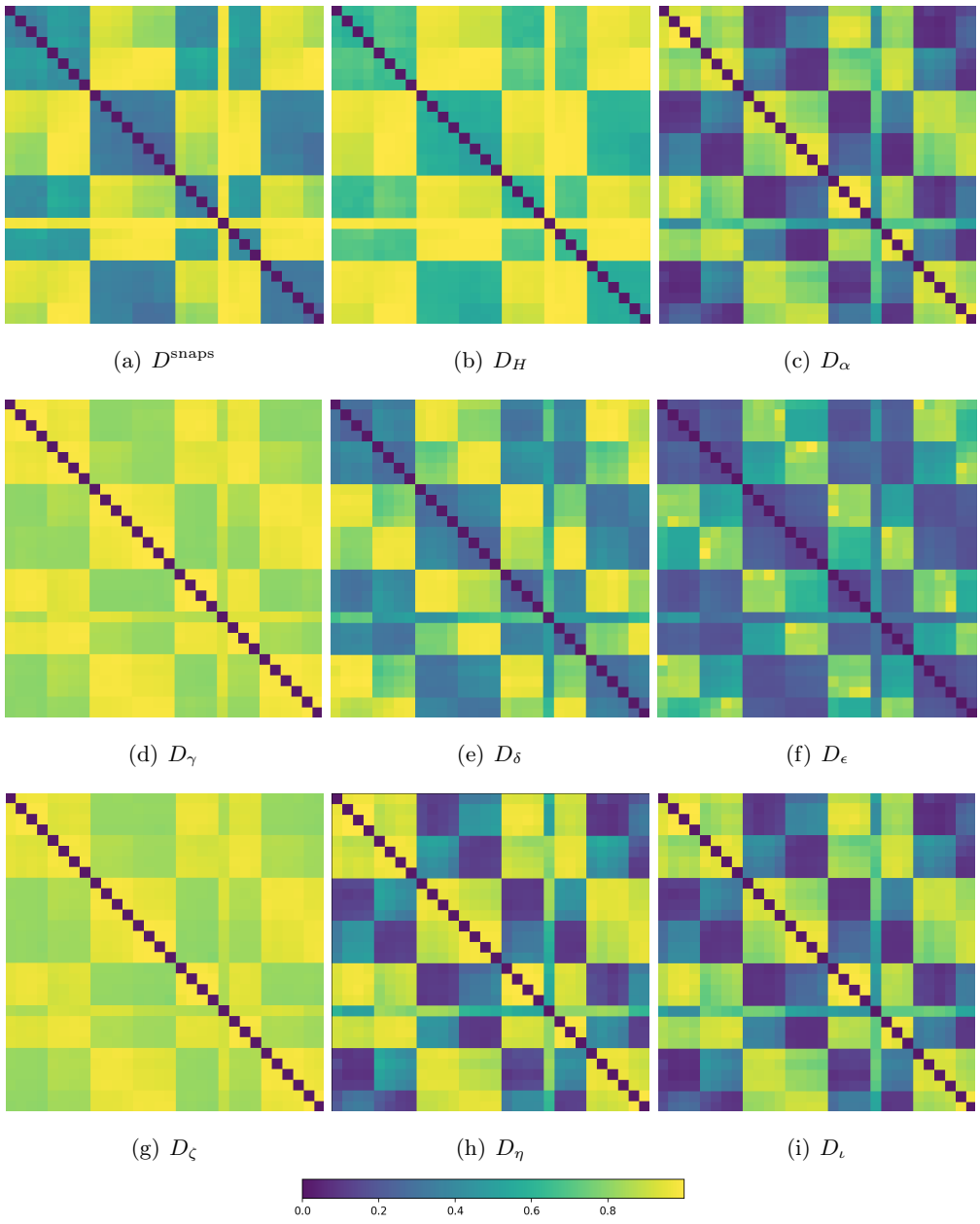
Figure 4.12: Decay of eigen-values for SVD in four geometries.

In addition, we can introduce another distance based on the models. We can compute the distance between two geometries Ω_i and Ω_j as follows:

$$d_{ij}^{\text{snaps}} = \frac{1}{2}\{\sum_{k=1}^{N_s^{(i)}} \frac{\| P_{V_i} \left(u_k^{(i)} - P_{\mathcal{T}_{j \mapsto i}(V_j)} u_k^{(i)} \right) \|}{\| P_{V_i} \left(u_k^{(i)} \right) \|} + \sum_{k=1}^{N_s^{(j)}} \frac{\| P_{V_j} \left(u_k^{(j)} - P_{\mathcal{T}_{i \mapsto j}(V_i)} u_k^{(j)} \right) \|}{\| P_{V_j} \left(u_k^{(i)} \right) \|}\}. \tag{4.14}$$

discarding the snapshots such that $\| P_{V_i} \left(u_k^{(j)} \right) \| = 0$ and $\| P_{V_j} \left(u_k^{(i)} \right) \| = 0$.

We conclude this part of the numerical experiment by visualizing all the distances exposed section 4.3.1 and 4.3.2 in figure 4.13. The dimensions of the spaces V_i are chosen fixing a tolerance 10^{-3} to the error (4.13).



(a) D^{snaps} (b) D_H (c) D_α

(d) D_γ (e) D_δ (f) D_ϵ

(g) D_ζ (h) D_η (i) D_ι

Figure 4.13: Visual representation of distance matrices between 30 working domains. We see computations for (4.9), (4.14) and all the distances from Table 4.1.

4.4 In silico flow reconstruction in non parametric-domains

In previous sections we have prepared the ground to solve the inverse problem of estimating blood flows from synthetic Doppler data. As we have pointed out, a critical point is about identifying a correct template given an incoming patient geometry. For doing so, we will explore two paths: First, we will use the distances discussed in the previous sections to build a low dimensional representation for the geometries. Second, we could try a straightforward method to learn directly how to distinguish between templates by introducing an error measure based on forward modeling. Both approaches will require two additional elements: on the one hand, a fast and systematic way to compare geometries, for which a voxelization method is introduced. On the other hand, we build a linear mapping to, given a new geometry and its voxelization, compute either its error with respect to each template or its representation in a low dimensional parameter space in such a way we can select the template with a nearest neighbors approach using an ℓ^2 metric.

4.4.1 Multidimensional scaling

We can try to find a low dimensional representation of the geometries from pair-wise distances, as those of figure 4.13. A multi-dimensional scaling (MDS) is the natural procedure for addressing this problem (see [Saeed et al., 2018] for a detailed survey). Let us assume that there exists vectors $x^{(1)}, \ldots, x^{(N_{\text{geo}})}$ in an $\mathcal{N}$-dimensional space, representing each one of the templates. We do not know those vectors, but we know the distances between them, encoded in a component-wise squared distance matrix $D^{(2)}$. The core idea is to try to enforce $x_i^T x_j = d_{ij}^2$. Notice that we say enforce due to the fact that this is assuming an ℓ^2 structure of the space that the high dimensional vectors $x^{(i)}$ belongs to. In addition, notice that there are infinitely many combinations of elements such that we get the pair-wise distances. A projector is introduced to add a zero-mean condition.

Formally, let $X \in \mathbb{R}^{\mathcal{N} \times N_{\text{geo}}}$ be a matrix with the geometries $x^{(i)}$ in its columns. So, we enforce the pair-wise distances for every pair of geometries by doing:

$$X^T X = -\frac{1}{2} H D^{(2)} H,$$

where $H = I_{N_{\text{geo}} \times N_{\text{geo}}} - \frac{1}{N_{\text{geo}}} e_{N_{\text{geo}}} e_{N_{\text{geo}}}^T$ is a double centering matrix that makes the low dimensional representation unique. Mathematically, we say that we are looking to embed the data into some space $\mathbb{R}^p$, with p hopefully small, reason why this is typically called Euclidean embedding. $e_{N_{\text{geo}}}$ is a vector with N_{geo} entries filled with ones.

Let us consider the eigen-value decomposition,

$$-\frac{1}{2} H D^{(2)} H = V \Lambda V^T.$$

The Euclidean hypothesis is then verified if the eigen-values are positive. Otherwise, some modifications to the method can be introduced, such as isomaps [Franz et al., 2014]. As we will see in our numerical examples, this extra step won't be required, as the geometry of the space we are trying to represent is intrinsically flat, but nonetheless the isomap can be introduced to the pipeline as the problem complexity increases.

Thus, for a fixed positive integer p we retrieve a parametric representation $Y \in \mathbb{R}^{N_{\text{geo}} \times p}$, with $p < \mathcal{N}$, by truncating the eigen value decomposition as follows:

$$Y = \Lambda_p^{1/2} V_p^T.$$

where Λ_p is the truncation of Λ up to the first p eigen-values and V_p is a matrix containing the first p columns of V. Let y_i be te columns of Y. The embedding is constructed to provide a good pair wise approximation in the sense that $\hat{D} \overset{\text{def}}{=} \|y_i - y_j\| \sim D_{ij}$. In fact, it can be proven that the embedding solves the optimization problem

$$\min_{Y \in \mathbb{R}^{N_{\text{geo}} \times p}} \frac{1}{2} \| H D^{(2)} H - H \hat{D}^{(2)}(Y) H \|_{\text{F}}^2,$$

where we denote $\hat{D}^{(2)}(Y)$ the distance matrix generated by a p-dimensional embedding Y. In many cases, such as the toy example of section 5.5,this problem has a zero minimizer providing an exact representation.

4.4.2 Numerical experiment

Before moving forward, let us check MDS with 16 geometries and $p = 2$. This way we can get a nice visualization of the geometries in a plane, as depicted in figure 4.14. We can see four clouds of points corresponding to clusters of geometries that share the same stenosis radius r_c. For 60 geometries we can depict the representation with $p = 3$ in figure 4.15 using the Hausdorff metric.

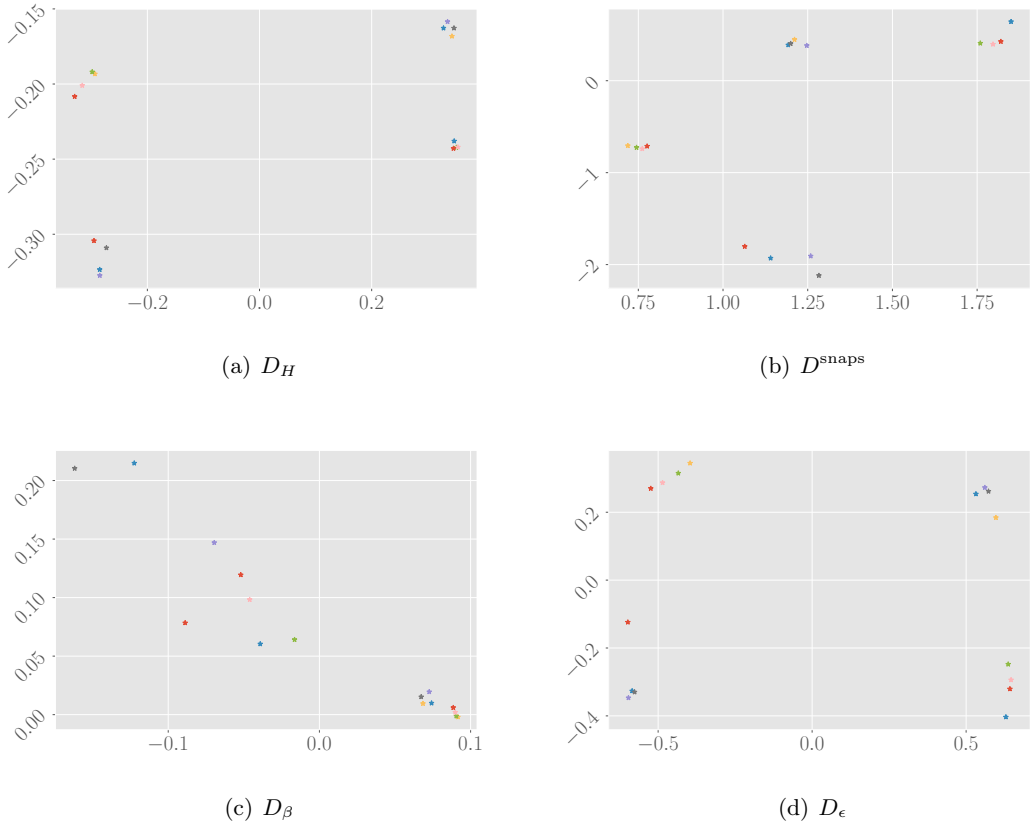
In terms of reproducing the distances, the three best choices to do the embedding are D^{snaps}, D_H and D_δ. We can see the error measure,

$$\frac{\|D - \hat{D}\|_{\text{F}}}{\|D\|_{\text{F}}} \tag{4.15}$$

in figure 4.16 for $p = 1, 2, 3$.

Concerning the distances d^α, d^γ, d^δ, d^ζ, d^η, d^ι, we get spectral decompositions with negative eigen-values. This suggest an underlying non-Euclidean geometry. It could be the case that this distances are able to identify the right template but an additional Isomap step should be added to the pipeline in order to test them, which is put as a perspective for the moment.

Concerning the reconstruction step of the pipeline, all the distances have to be tested, since a better accuracy in the low representation space does not translate necessarily in a better



(a) D_H (b) D^{snaps}

(c) D_β (d) D_ϵ

Figure 4.14: MDS representation of 16 geometries using $p = 2$. We show here three successful embeddings that capture the variations in the parameter space, D_H, D^{snaps} and D_ϵ. Every *island* corresponds to a fixed value of the coarctation radius r_c in the parameter space. D_β, on the other hand, does not manages to separate the data.

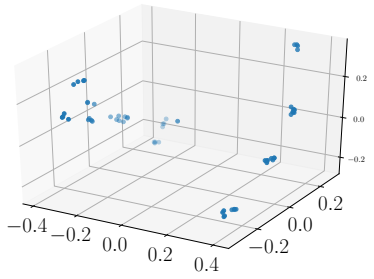


Figure 4.15: 3D representation of the full pool of geometries. In spite that the plot does not provide the same facility to interpret the data as it does for the two dimensional case, MDS indeed captures the intrinsic dimension of the parameter space in the sense that $D_H \sim \hat{D}$.

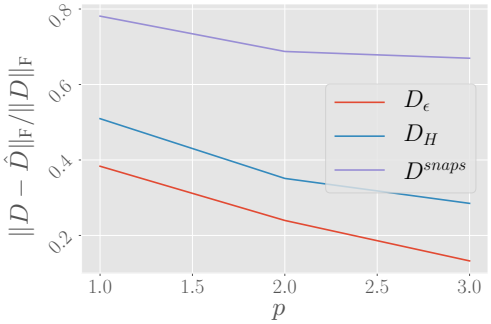


Figure 4.16: Capacity of MDS to reproduce the pair-wise distances as p grows. We see the quantity (4.15). Ω^* stands in the legend for the optimal template among those in the pool, computed *a-posteriori*.

recovery, as it has to be taken in to account that every distance encodes a different notion of proximity between models in the working domains.

4.4.3 Linear mappings for template finding

The next goal in the pipeline is, provided a new geometry Ω^*, to find the template from which we should transport the ROM in order to perform a fast blood flow reconstruction in Ω^*. As already brought up, we have two alternatives to follow. One is to build a map such that when given a new geometry, we can compute its representation on the parameter space. Another possibility is to circumvent the middle-step of computing a low-dimensional representation of the data and introduce the following measure of *error* between geometries:

$$E\left(\Omega_i, \Omega_j\right) = \sum_{k=1}^{N_s^{(i)}} \frac{\|u_k^{(i)} - P_{\mathcal{T}_{j\mapsto i}(V_j)} u_k^{(i)}\|}{\|u_k^{(i)}\|}, \tag{4.16}$$

discarding the snapshots with zero norm.

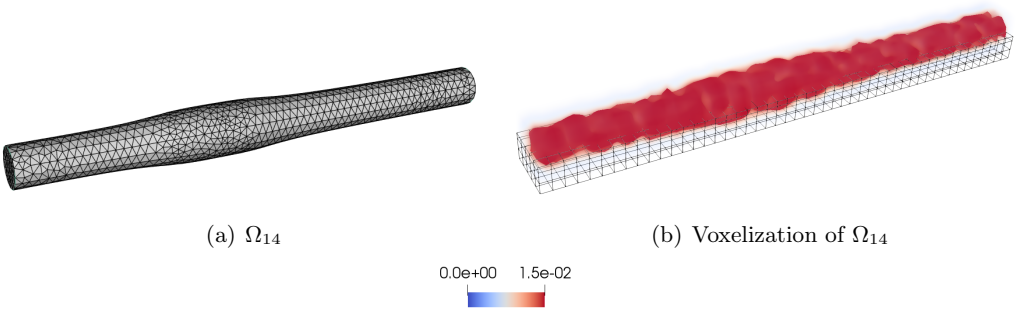
$$g_k = e^T M(\Omega_v^k) e,$$

where $M(\Omega_v^k)$ is a mass matrix for the tessellation of Ω intersected with the interior of the voxel Ω_v^k, that is to say,

$$M(\Omega_v^k)_{ij} = \int_{\Omega_v^k} \mathcal{L}_i \mathcal{L}_j \, \mathrm{d}x,$$

for $\{\mathcal{L}_i\}_{i=1}^N$ piece-wise linear $\mathbb{P}_1$ Lagrange functions on a mesh of Ω with N vertices.

We are in order to construct maps $g_{\text{mds}} : \mathbb{R}^{N_\text{v}} \to \mathbb{R}^p$ and $g_E : \mathbb{R}^{N_\text{v}} \to \mathbb{R}^{N_\text{geo}}$ (if we choose to circumvent the MDS and use (4.16)). Let us start by addressing the latter. A set of linear



(a) Ω_{14} (b) Voxelization of Ω_{14}

Figure 4.17: Voxelization of working domain Ω_{14} from the pool of geometries. Each voxel captures the volume portion of the mesh that there is inside.

mappings can be found in order to *learn* every row of $E(\Omega_i, \Omega_j)$. Basically, what we would like to find is some sort of test voxelizations, or weights, $w_i \in \mathbb{R}^N_{\text{v}}$ such that $w_i^T g_j = E(\Omega_i, \Omega_j)$. Otherwise stated, we can see this as solving N_{geo} optimization problems: find $w_1, \ldots, w_{N_{\text{geo}}}$ such that:

$$\min_{w_k \in \mathbb{R}^N_{\text{v}}} \frac{1}{2} \sum_{i=1}^{N_{\text{geo}}} |w_k^T g_i - E(\Omega_k, \Omega_i)|^2, \tag{4.17}$$

To enrich the input data we add $K_{it} = 100$ iterations of noise to each voxelized geometry, using white noise with deviation equal to 30% of the maximal volume captured by the grid. If we place the pool of voxelized geometries $g_1, \ldots, g_{N_{\text{geo}}}$ in the columns of a matrix $G \in \mathbb{R}^{N_{\text{v}} \times N_{\text{geo}}(K_{it}+1)}$, we can write down the optimization problems as follows:

$$\min_{w_k \in N_{\text{v}}} \frac{1}{2} \|G^T w_k - E(\Omega_k)\|^2_{\ell^2}, \qquad k = 1, \ldots, N_{\text{geo}}. \tag{4.18}$$

We denote $E(\Omega_k)$ the k row of $E(\Omega_k, \Omega_l)$. Trivial optimality conditions of this problems leads to the usually not well posed equations:

$$GG^T w_k = GE(\Omega_k), \qquad k = 1, \ldots, N_{\text{geo}}.$$

Consider the singular value decomposition $G = USV^T$, so we compute the solution to the problem by means of a regularized pseudo-inverse (which is, at best, nothing but a change of basis), i.e., $w_k = US^{-1}V^T E(\Omega_k)$. We have then the weights for N_{geo} linear mappings $g_E^{(i)} : \mathbb{R}^{N_{\text{v}}} \to \mathbb{R}^{N_{\text{geo}}}$.

On the other hand, if what we want to do is to build a mapping to the parameter space, we proceed analogously learning *each parameter at a time*. Otherwise stated, we solve p minimization problems:

$$\min_{w_k \in N_{\text{v}}} \frac{1}{2} \sum_{i=1}^{N_{\text{geo}}} |w_k^T g_i - y_k^i|^2,$$

for $k = 1, \ldots, p$, so that whenever we are given a new voxelization g^* we can guess its representation in a p-dimensional space. From there, the template is chosen via nearest neighbors in a ℓ^2 sense.

4.4.4 Blood flows reconstruction

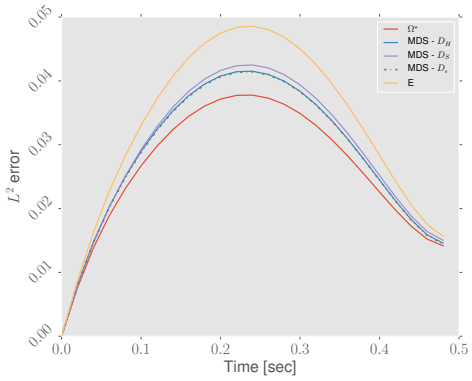


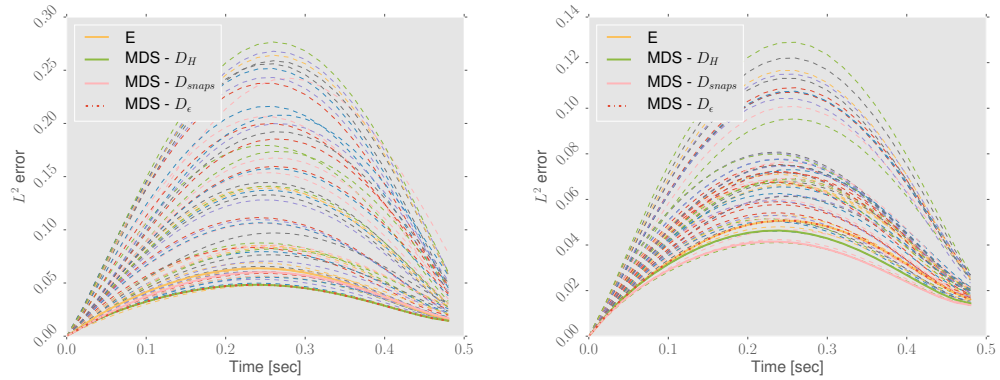
Figure 4.18: Average of averages in blood flows reconstruction. We see all 4 methods are capable of choosing templates that deliver errors below 5%. E stands for the reconstructions choosing the template with (4.17).

We can finalize the numerical experiments of this Chapter by coming back to our main goal: blood flows reconstruction from Doppler data. For doing so, we prepare a set of 16 *target* working domains by choosing randomly the geometrical parameters r_c, S_l and S_r in range with those of the pool of 60 geometries, assuming an uniform distribution. Let us denote those geometries by $\Omega_1^*, \ldots, \Omega_{16}^*$ (we present some of them in figure 4.19). In addition, at each of those 16 geometries we run 16 new simulations to test the reconstructions.

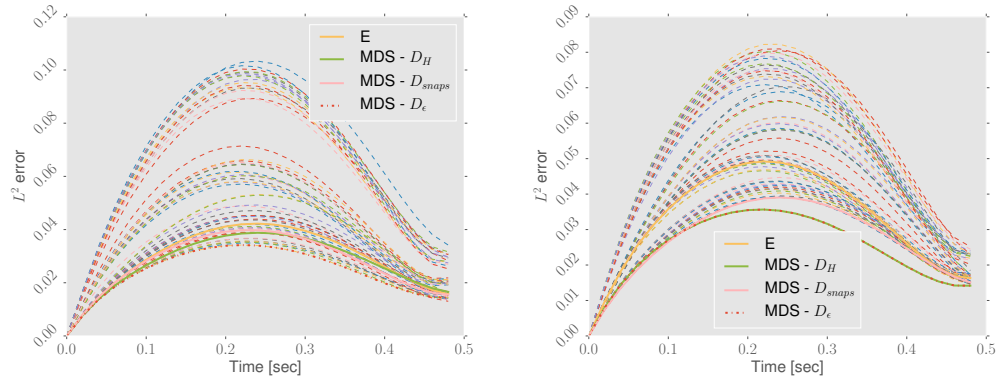
Figure 4.19: 3 of the 16 target working domains produced with a uniform distribution for S_r, S_r and S_x (in-range) to test the reconstruction algorithms.

Let us take a look to four isolated reconstructions corresponding to 6 different target geometries in figure 4.20, where each curve is the average of 16 reconstruction errors computed as:

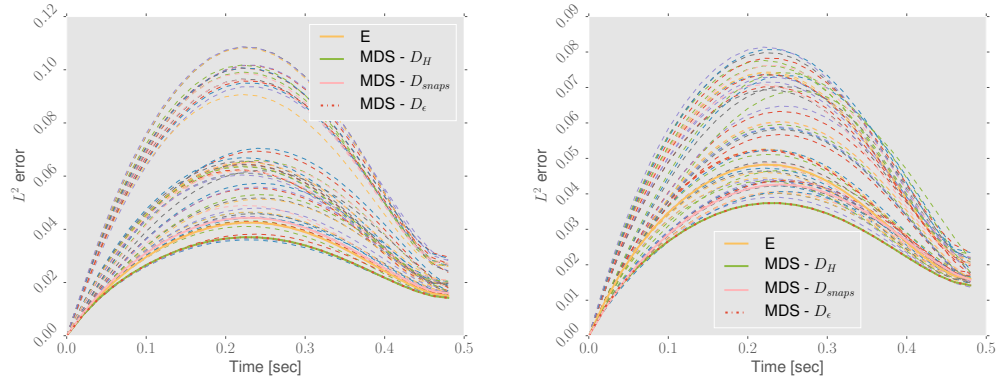
$$e(t) = \frac{\|u - u^*\|_{L^2(\Omega)}}{\int \|u\|^2_{L^2(\Omega)} dt}$$



(a) Ω_1^* (b) Ω_2^*



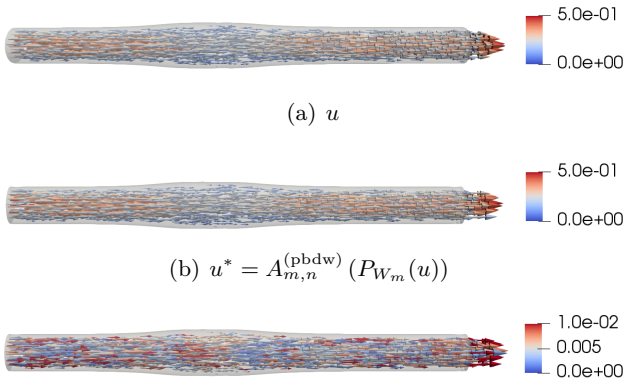
(c) Ω_3^* (d) Ω_4^*



(e) Ω_5^* (f) Ω_6^*

Figure 4.20: Reconstruction errors with all the templates for 6 target geometries. The curves are averages of 16 target simulations.

In addition, in figure 4.18 we see the average behavior among all the target geometries. We denote by u^* the reconstruction of the velocity field from $\omega = P_{W_m}(u)$ using PBDW, i.e., $u^* = A_{m,n}^{(\text{pbdw})}(\omega)$. The space of observations is built simulating an ultrasound transducer in the plane $x = 0$, with an inclination of $\pi/4$ with respect to the y axis. We see the reconstructed and error fields for Ω_{16}^* in figure 4.21



(a) u

(b) $u^* = A_{m,n}^{(\text{pbdw})}(P_{W_m}(u))$

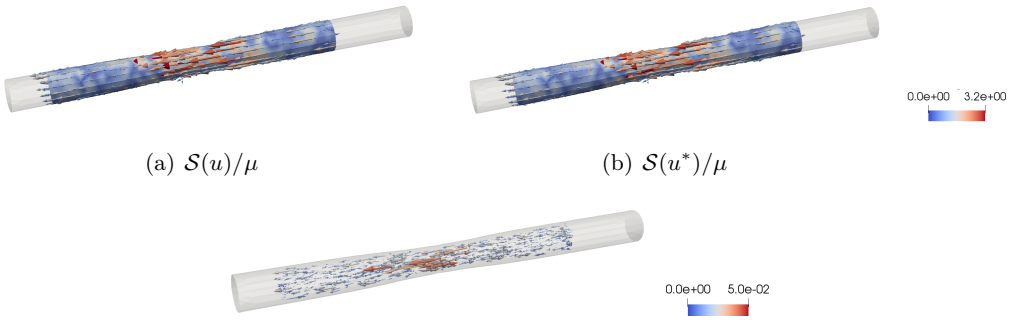
(c) $u - u^*$

Figure 4.21: Example of reconstruction from optimal template POD using as a criteria the mapping learned from (4.18). The parameter space is built using the Hausdorff metric. The snapshot is taken at time $t = 0.25$ [sec].

In addition, we can verify a scaled WSS $\mathcal{S}(u^*)/\mu$ reconstruction for one snapshot in figure 4.22. Extensive tests on QoI reconstruction has to be made in order to drive conclusion about the quality of the results. Nonetheless, recovering a WSS with high accuracy in spite of selecting $V = L^2(\Omega)$ as ambient space is very encouraging. Another pre-liminary test we can do concerning the post-processing of the results, is to check out the energies $E_{\text{kin}} = \int u \cdot u \, \mathrm{d}x$ and $E_{\text{visc}} = \int \nabla u : \nabla u \, \mathrm{d}x$, for the same target, depicted in figure 4.23. As it was verified for the pressure drop reconstruction in Chapter 4, the accuracy to capture this kind of aggregated quantities is higher than capturing the state.

4.4.5 Pressure and pressure drop estimation via STE

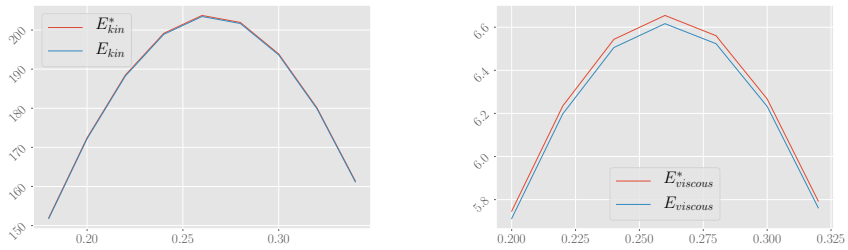
We are going to briefly explore a pressure estimation method to apply in our reconstructions. It relies on using the Navier-Stokes equations to solve a deterministic inverse problem for the pressure field over the whole geometry, so that the pressure drop can be directly computed afterwards. The methodology is called Stokes pressure estimator (STE) and it is introduced in [Švihlová et al., 2016], where further theoretical considerations can be found. Let us write down the method for a generic domain $\Omega \subset \mathbb{R}^3$.



(a) $\mathcal{S}(u)/\mu$ (b) $\mathcal{S}(u^*)/\mu$

(c) $\mathcal{S}(u)/\mu - \mathcal{S}(u^*)/\mu$

Figure 4.22: Scaled WSS reconstruction. The snapshot is taken at time $t = 0.25$ [sec].



(a) E_{kin} (b) E_{visc}

Figure 4.23: Reconstruction of quadratic forms E_{kin} and E_{visc} in a time window close to the velocity peak.

Let us assume that momentum conservation holds for the PBDW reconstruction field $u_{\text{pbdw}} = A(P_{W_m} u) = A(\omega)$

$$\rho\left(\partial_t u_{\text{pbdw}} + (\nabla u_{\text{pbdw}})\, u_{\text{pbdw}}\right) - \mu \Delta u_{\text{pbdw}} + \nabla p_{\text{ppe}} = 0.$$

A natural approach that we could follow is to take the divergence, so we get the Poisson problem: Find $p_{\text{ppe}} \in H^1(\Omega)$:

$$-\Delta p_{\text{ppe}} = \nabla \cdot \left(\rho\left(\partial_t u_{\text{pbdw}} + (\nabla u_{\text{pbdw}})\, u_{\text{pbdw}}\right) - \mu \Delta u_{\text{pbdw}}\right) \quad \text{in } \Omega.$$

plus boundary conditions that can be seen at [Švihlová et al., 2016]. As was already brought up, PBDW does not necessarily provides solenoidal fields, therefore we can not vanish the time derivative in the equation, as we would usually do for an incompressible flow. This methodology to compute pressure fields from velocity data is introduced and further examined in the context of magnetic resonance data [Krittian et al., 2012], and is called Poisson pressure estimator (PPE). To assume that p_{ppe} is the pressure field of the system is risky mostly due to the exaggerated regularity that is imposed when solving the Poisson problem. This is the reason behind that the method typically underestimate pressure jumps, as has been explored for MRI data in [Bertoglio et al., 2017b]. This should be enough argument to justify a technique that relax this regularity condition, searching for a field in $L^2(\Omega)$ instead of a field in $H^1(\Omega)$, so that the functional setting is consistent with the Navier-Stokes equations.

In short, STE can be formulated as: find $w \in [H^1(\Omega)]^3$ and $p_{\text{ste}} \in L^2(\Omega)$:

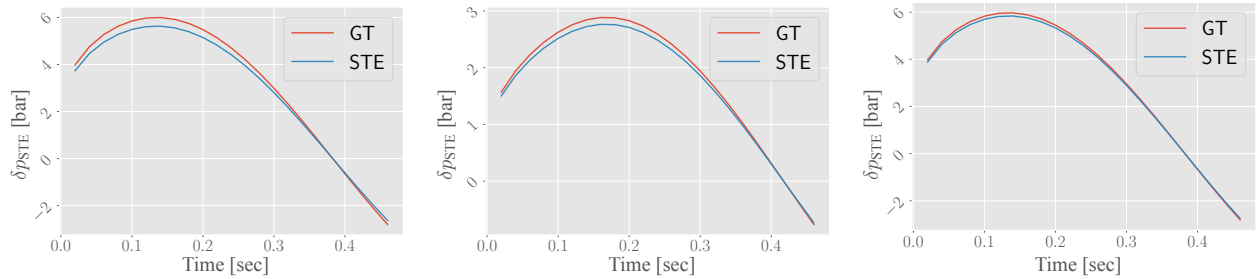
$$\begin{aligned} \Delta w + \nabla p_{\text{ste}} = -\left(\rho\left(\partial_t u_{\text{pbdw}} + (\nabla u_{\text{pbdw}})\, u_{\text{pbdw}}\right) - \mu \Delta u_{\text{pbdw}}\right) \quad & \text{in } \Omega. \\ \nabla \cdot w = 0 \quad & \text{in } \Omega \\ w = (0,0,0) \quad & \text{on } \partial\Omega, \end{aligned}$$

where w can be seen as some sort of auxiliary field, typically close to zero, which has nothing to do with the velocity field nor the reconstructed u^*_{pbdw}. The computation of the pressure drop then is straightforward. For two disjoint sets $\Gamma_{\text{in}} \subset \partial\Omega$ and $\Gamma_{\text{out}} \subset \partial\Omega$ we simply compute the drop

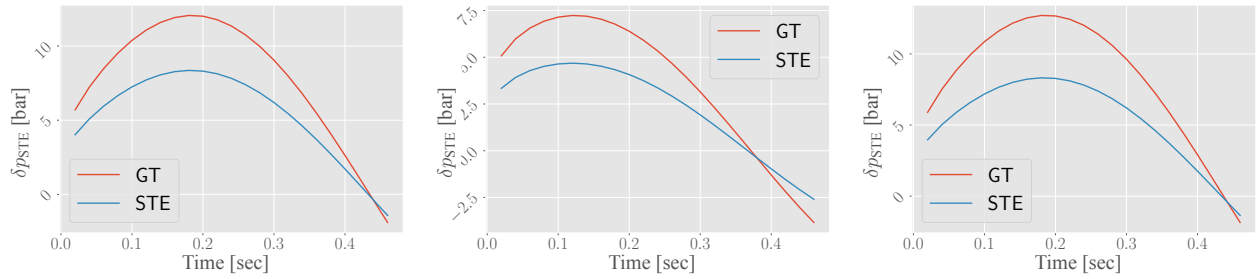
$$\delta p_{\text{ste}} = \frac{1}{|\Gamma_{\text{in}}|} \int_{\Gamma_{\text{in}}} p_{\text{ste}}\, \mathrm{d}s - \frac{1}{|\Gamma_{\text{out}}|} \int_{\Gamma_{\text{out}}} p_{\text{ste}}\, \mathrm{d}s$$

We test STE in the context of reconstruction of non-parametric domains, following the numerical example of the previous section in blood flows.

In figure 4.24 the reconstruction over 6 target working domains is depicted, from what the following observations arises: there are cases in which the pressure drop works up to good precisions, and others in which it does not (regardless that the method manages to capture the pressure drop shape). A theoretical background to understand the perimeter of application



(a) Succesfull reconstructions of pressure drop

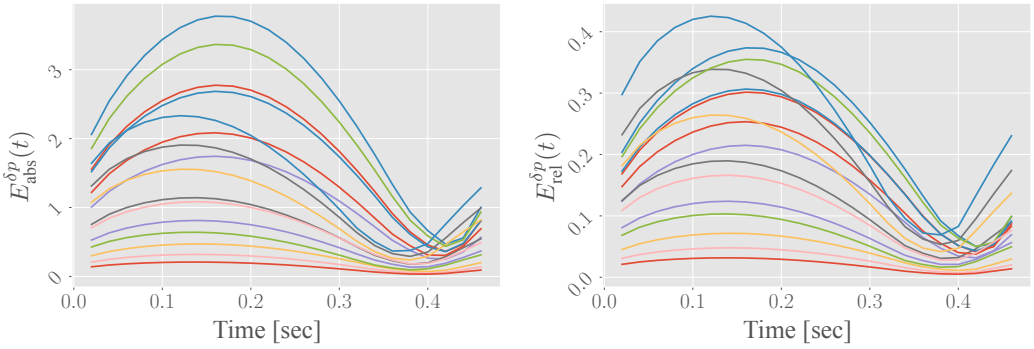


(b) Failed reconstructions of pressure drop

Figure 4.24: Pressure drop reconstruction in non-parametric domains using STE.

of STE in the reconstruction in non-parametric domains is required in order to discriminate when the reconstruction is doable. Empirically, it is clear that the most influential geometrical parameter in the pressure drop reconstruction quality is not surprisingly the coarctation radius S_r. Despite that a deep sensitivity analysis might be performed in order to understand the impact of each parameter on the reconstruction, the numerical experiments suggests that the pressure drop reconstruction results are much better as $S_r \to D/2$ (i.e., tubes with almost no coarctation. The challenge here is admittedly simple.), they are moderately good when $S_r \to 2.6$ mm (i.e., tubes with dilatation) and they become unfortunately wrong as $S_r \to 1.4$ (i.e., tubes with high coarctation). Another thing to be notices is that, as might be expected, the pressure drop reconstruction via STE is much less sensitive to the parameter space in the training set at each geometry. Otherwise stated, if we pick a geometry with suitable geometrical parameters for the reconstruction in the sense of the previous lines, then no matter the choice of the simulation parameters (viscosity, boundary condition intensity and time), the reconstruction stays good as well. We can take the average of each simulation in the target set fixing the target working domain in order to see this more clearly (see figure 4.25).

The results are not encouraging in the sense that they require the geometrical variabilities to be too reduced in order to guarantee good reconstructions. We see that the *worst mesh* in terms of relative errors goes up to 40 percent, a useless result from any point of view. The absolute



(a) $E^{\delta p}_{\text{abs}}(t)$, defined at (4.19)

(b) $E^{\delta p}_{\text{rel}}(t)$, defined at (4.20)

Figure 4.25: Average errors in pressure drop per target geometry.

error in the figure 4.25 is measured as

$$E^{\delta p}_{\text{abs}}(t) = |\delta p(t) - \delta p_{\text{ste}}(t)|, \tag{4.19}$$

and the relative one is measures as

$$E^{\delta p}_{\text{rel}}(t) = \frac{|\delta p(t) - \delta p_{\text{ste}}(t)|}{\max_{t\in[0,0.5]} |\delta p(t)|}. \tag{4.20}$$

To conclude we can also average the relative error among the 16 target geometries, as shown in figure 4.26.

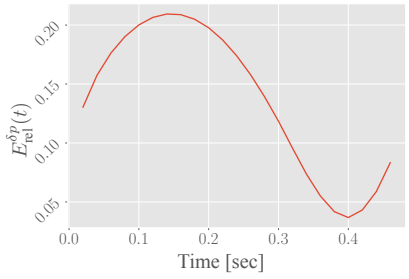


Figure 4.26: Average for the 16 target domains of relative errors (4.20) in pressure drop.

4.5 Conclusion and perspectives

We have developed a full pipeline for state estimation in non-parametric domains. We successfully introduce a methodology based on metrics induced by governing dynamics. We consider this a step towards the clinical implementation of the tools presented in this thesis.

Nevertheless, there is a few points that have to be addressed and are seen as future work, all of them related to increase the challenge of the problem: First, given the promising results of this Chapter, a next test case in patient-specific geometries can be performed. Second, non linear PDEs can be used to generate the training sets, such as the Navier-Stokes equations or even fluid structure interaction coupled equations.

In addition, we have appended a QoI estimation to the chain using a pressure estimator called STE. Nonetheless, the results are ambiguous, and they suggest that other approaches should be tried. The natural perspective this results gives is to use for instance the joint reconstruction introduced in chapter 3. The joint reconstruction will require first the generation of pressure basis in each template, and also a correct mechanism to translate it. In [Guibert et al., 2013], pressure basis are transported by using $\mathcal{I}_{i\mapsto j} \circ d_v$, but maybe some structure preserving transformation can be explored for this matter.

Another work to be addressed in more realistic scenarios is to try a more sophisticated way of building the mappings of section 4.4.3. Non linear regressions such as deep neural networks are a first strategy to be explored.

Chapter 5

Software implementation: *Multi-physics for biomedicAl engineering and Data assimilation*

MAD, which stands for Multi-physics for biomedicAl engineering and Data assimilation, is a software created by Felipe Galarce at INRIA Paris. MAD is developed with a hope to be useful in addressing state-of-the-art problems of the field, but more importantly, it is a way to ensure reproducibility in science for any work based on this code. Among MAD's high-level characteristics, we can distinguish between, say, a project level and a library level. The project level is where full pipe-lines for research projects related to this Ph.D. thesis can be found. The library level refers to three main sets of objects. First, a complete CFD library containing solvers for the Navier-Stokes equations using both monolithic and fractional step approaches, interpolation routines, post-processing tools, and pressure estimators from data. Second, a set of tools for State Estimation, Data assimilation, and Model reduction, covering a broad range of tools such as Kalman filters, variational approaches, and proper orthogonal decomposition. Third, a library oriented to model reduction in non-parametric domains is implemented. For instance, we can find tools to compute feature maps, such as multi-dimensional scaling and isomaps. MAD is paradigm-free in the sense that object-oriented or function-oriented code is used whenever it seems pertinent. The code is developed under LGPL license, and its source code, mostly written in C++ and Python can be accessed by request to the author. Linux and MAC OS are supported.

Contents

5.1 Introduction

This chapter describes some aspects of MAD (Multi-physics for biomedicAl engineering and Data assimilation), its scope, limitations and design choices. MAD is developed with a hope to be useful in addressing state-of-the art problems of the field and to provide a guarantee that ensures reproducibility for the numerical experiments that this thesis contains. The text will deliberately alternate between scientific computing aspects, some mathematical foundations and discussions about the cardiovascular tree. This intends to remark how those three axes are intrinsically related behind the choices made to code MAD, so that an eventual user can easily see what MAD is designed for and what MAD is not designed for.

There has been an effort in making a modular code structure that, as a whole, attempts to be as self-contained as possible. That is to say, the software offers tools for state estimation such as the Parametrized Background Data-weak method or machine learning tools such as multi-dimensional scaling, but also contains a PDE solver to generate realistic 3D simulations that could serve for instance, to feed learning algorithms. In addition, model reduction techniques are part of the code, such as PCA and data-driven greedy methods.

Code modularity allows one to use all the aforementioned features independently, but they can also be seen as parts of larger pipe-lines or projects, where typically a very particular application is tackled. This ideas are much more clearly seen in figure 5.1. The user is therefore free to choose at what level of abstraction to work. Lower-level routines such as linear algebra manipulations or finite elements provides flexibility and they can be accessed if the user see a new possible application of them, so a new project can be created.

Concerning forward modeling, MAD is focused on simulating blood inside large vessels. That said, we remark that there is no intention to make of this program a multi-purpose solver for finite elements, as there are many others that are by far much more complete, such as FreeFem++ [Hecht, 2012] or FEniCS [Logg and Wells, 2010], or other softwares that propose an interface in which the main feature is dealing directly with weak formulations. In this sense, MAD resembles another software developed at INRIA: FeLiScE, in which the purpose is to, fixed a model, to set a user interface in which the interaction is done only at the parameter level, without requiring further mathematical knowledge.

The multiple paralellization possibilities that this program provides relies on the use of the standard MPI and the linear algebra library PETSc [Balay et al., 1997] extended for eigen-value problems with SLEPc [Hernandez et al., 2005]. In addition, the software interfaces with PyKeops [Charlier et al., 2020] to address the computation of mappings between geometries. PyKeops relies on PyTorch, which is used deal with kernels. This is the reason behind including CUDA at the bottom of figure 5.1. Besides the interfaces with PyTorch, MAD does not exploit

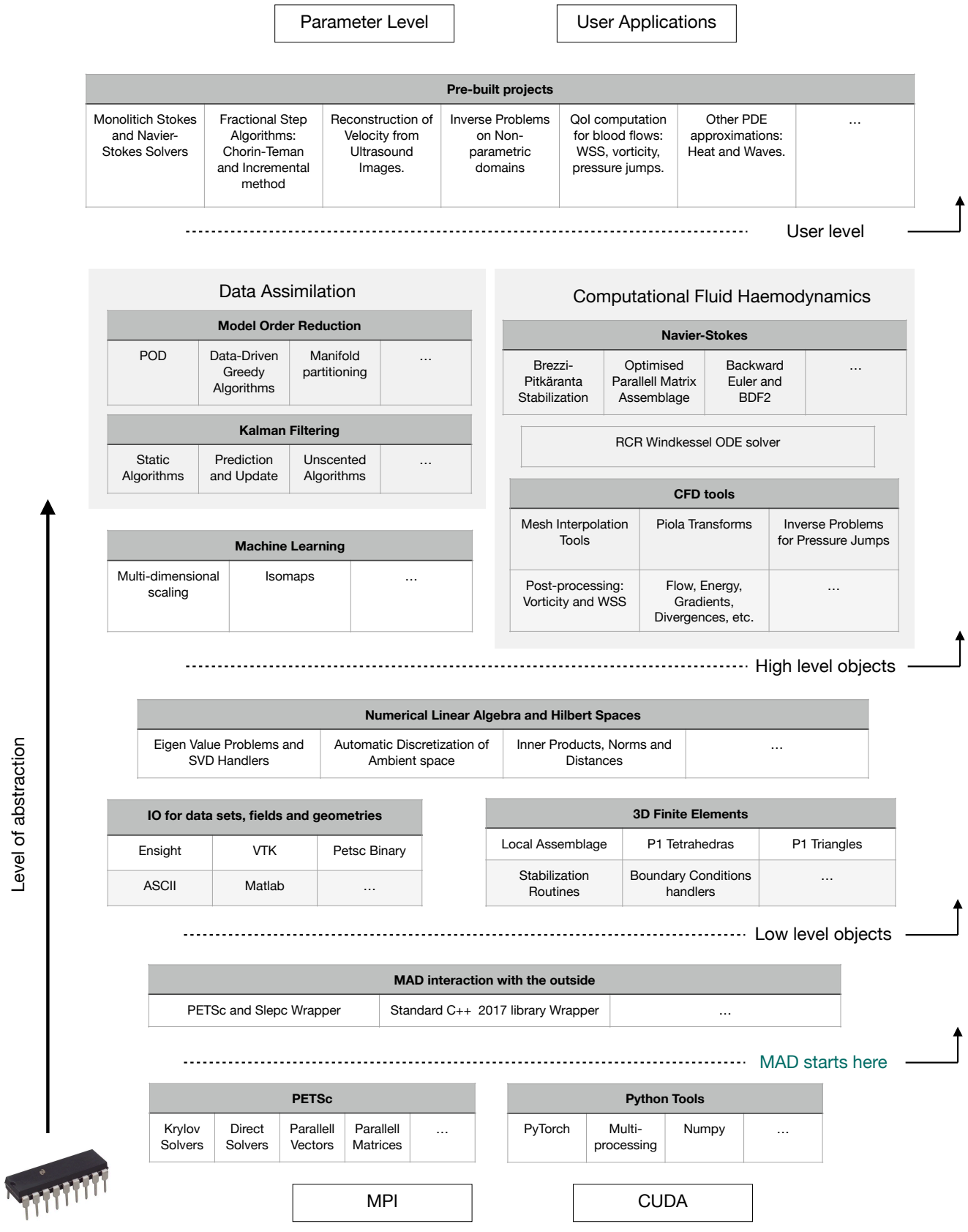


Figure 5.1: Map of MAD features.

graphics processing units.

Despite of the fact that low-level manipulation of MAD allows the user to extend the code beyond the application scope for which MAD is designed for, all the development decisions were made with biomedical applications in mind. The higher the level of abstraction, the closer to the application we stand, and therefore the larger the amount of hypothesis that are made. We include the motivation behind those hypothesis in this chapter. Sections 5.2, 5.3 serves to this purpose. In addition, a typical problem in haemodynamics is discussed in section 5.4.

5.2 Generalities about haemodynamics

One of the main goals of the software is to simulate blood flows. In order to do so, we assume typical governing dynamics coming from Newton's ideas and continuum mechanics. A number of parameters need to be understood from a phenomenological point of view in order to use them properly and carefully in our hypotheses and results. Also, some orders of magnitude of relevant quantities will be exposed in this section in order to ease the result interpretations.

5.2.1 Basic rheology of human blood

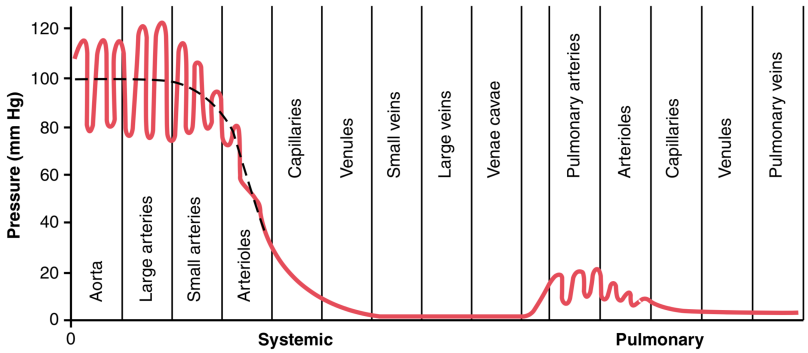


Figure 5.2: Normal values of blood pressure in rest state. Image source: [Guyton and Hall, 2011], chapter 14.

As we will see, the main words concerning the mathematical modeling will be velocity, pressure, density and viscosity. Let us do a survey of normal ranges for those numbers in resting conditions. The velocity average in the thoracic aorta is known to ranges from 100 [cm/s] to 150 [cm/s], whereas is around 0.3 [mm/s] in the capillaries. Concerning the pressure, we can see a detailed curve depending on the system region in figure 5.2. Pressure will be the only exception to magnitudes expressed in CGS in this document.

Although the blood viscosity is far from being constant in space, we will always denote it by

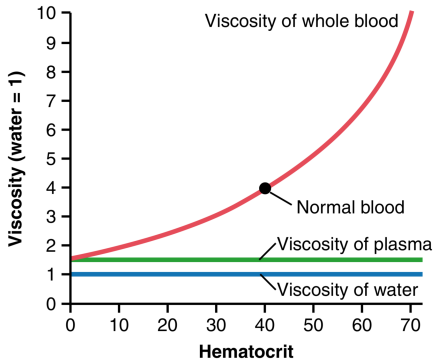


Figure 5.3: Blood viscosity ranges in humans compared to water viscosity. Average hematocrit on men is 42, and 38 in woman. Sources of variations of this parameter are abnormal conditions such as anemia or polycythemia. Image source: [Guyton and Hall, 2011], chapter 14.

$\mu \in \mathbb{R}^+$, which is a good assumption for the scales we work with. In fact, we can understand the underlying phenomena by examining its two main components: red blood cells (RBC) and plasma. Figure 5.3 shows the influence of the proportion of RBC over the total blood, so called *hematocrit*, in the viscosity, viewed as a continuum in the plot. RBCs are not distributed homogeneously in the blood vessel, there will be typically a major concentration close to the vessel center line, mainly in laminar flows, as can be expected.

Constant viscosity is the first hypothesis we introduce. Density is assumed to be close to that of water at ambient temperature.

As we will see in section 5.3, the mathematical modeling at the 3D level will require the aspects narrated in this section. On the other hand, next section provides a discussion where 3D simulations are not doable typically due to computational costs. So called 0D models are introduced and macroscopic variables, resembling Ohm's law's ideas about resistance and capacitance, are used.

5.2.2 Ohm's law for blood flows

To understand phenomena we don't always need sophisticated partial differential equations (PDE). We arrive to them for, among other reasons, the need for accuracy, to understand underlying patterns that are not captured by macroscopic relationships, or for the mere beauty of contemplate that a certain portion of the sensitive world can be almost perfectly described by very fundamental intellectual notions. Cases in which we don't require nor can afford complex models are just too many in engineering. Rarely an hydraulic expert will come up with mass conservation equations written on their differential form to design a few cylindrical pipes.

Thus, before introducing the deep understanding that fluid mechanics can bring to the study

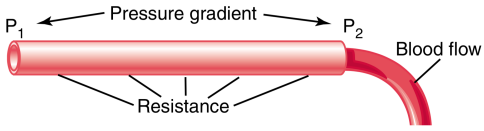


Figure 5.4: Illustration of rudimentary ideas of flow mechanics in blood vessels. Image source: [Guyton and Hall, 2011], chapter 14.

of blood, we will discuss some intuitive and rudimentary ideas of relations between velocity and pressure. Those ideas will be indeed used in our models when necessary, and will be typically referred to as *0D models*.

The easier way to grasp the following lines is to think about electrons flowing (we abuse of the language when it does not touch the main thesis subject!) on a conductor as an analogy to flow and pressure. So, as an electric potential will produce the possibility of electricity to pass, a pressure gradient will do the same for blood flow. On the other hand, the vessel walls oppose to the movement due to the presence of vascular endothelium cells (figure 5.4). This *resistance* R allow us to link flow F and pressure gradient ΔP as simply as,

$$F = \frac{\Delta P}{R},$$

which is the reason why this is typically referred as Ohm's law.

Another concept of importance is to understand the ratio of volume and pressure inside vessels. This ratio is called *vascular compliance*, and is also part of this electrical analogy, burrowing the concept of capacitance. Let Γ be a section of interest where we cant to compute the tension $\overline{p}$. Let $Q(t)$ be the flow across Γ for a given time instant t. A 3 elements Windkessel reads:

$$\overline{p} = p_d + R_p Q(t),$$

where R_p plays the role of a proximal resistance in the circuit and where p_d is a distal pressure and is the solution to the ordinary differential equation (ODE):

$$\begin{cases} C_d \dfrac{dp_d}{dt} + \dfrac{p_d}{R_d} & = Q(t), \\ p_d(0) & = p_d^0. \end{cases} \tag{5.1}$$

where $C_d > 0$ is called distal capacitance, $R_d > 0$ is called distal resistance, and p_d^0 is an initial datum.

5.3 Conservation laws and Hilbertian framework

In a modern context, we will always think about velocity and pressure as elements of a metric space in which all Cauchy sequences converge in a norm $\|\cdot\|$ that is induced by its inner product $\langle\cdot,\cdot\rangle$ doing $\|\cdot\| = \sqrt{\langle\cdot,\cdot\rangle}$.

In particular, three Hilbert spaces will drag our attention during this chapter. Namely,

$$L^2(\Omega) = \{f : \Omega \to \mathbb{R};\ \int_\Omega |f|^2 \,\mathrm{d}x < \infty\},$$

$$\mathcal{H} \overset{\text{def}}{=} \left[H^1(\Omega)\right]^3 = \{f \in \left[L^2(\Omega)\right]^3;\ \frac{\partial f_i}{\partial x_j} \in L^2(\Omega),\ i = 1,\dots,3,\ j = 1,\dots,3\},\ \text{and}$$

$$L^2\left(\partial\Omega\right) = \{f : \partial\Omega \to \mathbb{R};\ \int_{\partial\Omega} |f|^2 \,\mathrm{d}s < \infty\},$$

where $\Omega \subset \mathbb{R}^3$ (MAD only support 3D geometries) and $x = [x_1,\ x_2,\ x_3] \in \Omega$. Thus, from now on, velocity will almost always refer to an element $u \in \mathcal{H}$ and pressure to $p \in L^2(\Omega)$.

In the following, we mix up the previous lines with some fundamentals of continuum mechanics. Let us consider a function $h(x) : \Omega \to \mathbb{R}^d$ ($d = 1$ or $d = 3$ typically) which describe some property (such as density), in Eulerian (or spatial) coordinates. Let $X(t)$ denote Lagrangian (or particle) coordinates. Consider the transformation between both coordinates descriptions $x(t) = \xi(X, t)$. Fluid mechanics almost compels the use of Eulerian Frameworks. Whenever we want to evaluate a material (or Lagrangian) derivative $\frac{D}{Dt}(h)$ we can use the chain rule

$$\frac{D}{Dt}h(x) = \frac{D}{Dt}h\left(\xi\left(X,t\right)\right) = \frac{\partial h(x)}{\partial t} + \sum_{i=1}^{3} \frac{\partial h}{\partial \xi_i}\frac{\partial \xi_i}{\partial t}$$

and get

$$\frac{D}{Dt}h(x) = \frac{\partial h(x)}{\partial t} + u \cdot \nabla h(x). \tag{5.2}$$

where $u = \partial_t \xi$. This crucial relation can be seen as a link between Lagrangian and Eulerian points of view as well. $t > 0$ is time. Now we can briefly sketch how to derive the conservation laws that will be used in this document. First one is mass conservation, which translates to a material derivative of the domain mass vanishing. If $\rho : \Omega \to \mathbb{R}^+$ is a function denoting the domain density then:

$$\frac{D}{Dt}\int_\Omega \rho \,\mathrm{d}x = 0.$$

From 5.2 and taking into account that the domain is arbitrary one can see that this law implies that:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho u) = 0. \tag{5.3}$$

There is no need to use this equation for our models since we will always consider a constant density in time and space, which is a reasonable assumption for the phenomena we are simulating with MAD. Thus, for an incompressible flow:

$$\nabla \cdot u = 0. \tag{5.4}$$

It is hard to grasp the scope of ideas that big characters of science have had. A glance is shown here, where Isaac Newton inspires the following momentum conservation law:

$$\frac{D}{Dt}\int_\Omega \rho u \, \mathrm{d}x = \sum F_{ext},$$

denoting here $\sum F_{ext}$ the external *forces* acting in the domain. Let us assume that the external forces are only produced by surface stresses on the boundary $\Gamma = \partial\Omega$, i.e., body forces are neglected. We can write such surface forces as follows:

$$\sum F_{\text{ext}} = \int_\Gamma \sigma \cdot n \, \mathrm{d}x,$$

where σ is the stress tensor of the material and n is a normal unitary and exterior vector to the domain. Using the divergence theorem we arrive to:

$$\frac{D}{Dt}\int_\Omega \rho u \, \mathrm{d}x - \int_\Omega \nabla \cdot \sigma \, \mathrm{d}x = 0.$$

By means of the same arguments used to write down mass conservation (5.4) on differential form we get:

$$\rho\frac{\partial u}{dt} + \rho\,(u \cdot \nabla u) - \nabla \cdot \sigma = 0. \tag{5.5}$$

We must assume a constitutive law for the blood. In principle, non-linearities are impossible to neglect in small vessels, typically arterioles and capillaries. This type of domains are out of the scope of MAD. On the contrary, for large vessels, to assume a Newtonian flow provides a good model, that is to say, the stress tensor is written as:

$$\sigma = 2\mu\epsilon(u) - pI_{3\times 3}. \tag{5.6}$$

The strain rate tensor ϵ is usually taken as the symmetric part of the velocity field: $\epsilon(u) = \frac{\nabla u + \nabla^T u}{2}$. This leads us to the momentum conservation equation that will be heavily exploited in the following chapters:

$$\rho\frac{\partial u}{\partial t} + \rho\,(u \cdot \nabla u) - \mu\Delta u + \nabla p = 0 \tag{5.7}$$

The coupled system of equations composed by 5.4 and 5.7 was introduced by Claude-Louis Navier and George Gabriel Stokes during the XIX century.

5.4 Standard problem in haemodynamics

This thesis will require to numerically solve some PDEs such as Laplace and Poisson problems, Stokes or Navier−Stokes systems. We are dedicating here a few lines about how we solve a Navier−Stokes problem with MAD. Let us decompose the domain boundary as follows: $\partial\Omega = \Gamma_{\mathrm{w}} \bigcup \Gamma_{\mathrm{in}} \bigcup \Gamma_{\mathrm{out}}$. Those boundary sub-domains are disjoint. The real interval $[0, T]$ will be our time domain.

The problem reads as follows: for any $t \in [0, T]$, given $F_{\mathrm{out}} \in \left[L^2(\Gamma_{\mathrm{out}})\right]^3$, find $u \in \mathcal{H}$ and $p \in L^2(\Omega)$ such that:

$$\begin{cases} \rho \dfrac{\partial u}{\partial t} + \rho\left(u \cdot \nabla u\right) - \mu \Delta u + \nabla p = 0 \text{ in } \Omega, \\ \nabla \cdot u = 0 \text{ in } \Omega, \\ \mu \dfrac{\nabla u + \nabla^T u}{2} \cdot n - p I_{3\times 3} \cdot n = F_{\mathrm{out}} \text{ on } \Gamma_{\mathrm{out}}, \\ u = (0,0,0) \text{ on } \Gamma_{\mathrm{w}}, \\ u = u_{\mathrm{in}} \in \mathbb{R}^3 \text{ on } \Gamma_{\mathrm{in}}. \end{cases} \tag{5.8}$$

5.4.1 Time discretization

This section discusses the choices that MAD offers in terms of time discretization, not only concerning the time derivative in the momentum equation, but also dealing with approaches that split the solution adding an intermediate step that hopefully increases the performance of the overall simulation, the so called *fractional step methods*. For a detailed survey on the several available choices that can be followed the reader is referred to [Guermond et al., 2005].

The time domain $[0, T]$ is uniformly partitioned with a size $\Delta t > 0$. The time discretized problem is then to find functions $\{u^1, \ldots, u^n\}$.

5.4.1.1 Backward difference formulas

Concerning the only time derivative in the equations, we are either going to use an implicit Euler approach, also called first order Backward Difference Formula (BDF1), that is to say:

$$\rho \frac{u^{n+1} - u^n}{\Delta t} + \rho\left(u^n \cdot \nabla u^{n+1}\right) - \mu \Delta u^{n+1} + \nabla p^{n+1} = 0,$$

or a second order approximation (BDF2):

$$\frac{\rho}{2\Delta t}\left(3u^{n+1} - 4u^n + u^{n-1}\right) + \rho\left(u^n \cdot \nabla u^{n+1}\right) - \mu \Delta u^{n+1} + \nabla p^{n+1}.$$

To ease the reading, all the equations in the following sections of this chapter will be shown using BDF1. In practice we will do different choices for space and time discretization according

to the needs of the moment. Those things will be clarified pertinently for every numerical experiment. Notice the *semi-implicit* choice for the convective term, a good trade off between a realistic model and difficulty of solving. This decision, used in every simulation of this thesis, will allow us to use linear solvers at the discrete level.

5.4.1.2 Monolithic approach

If we couple both mass conservation and the aforementioned BDF1 formula for the momentum conservation we get a Monolithic time discretized system:

$$\begin{cases} \rho\dfrac{u^{n+1}-u^n}{\Delta t} + \rho\left(u^n\cdot\nabla u^{n+1}\right) - \mu\Delta u^{n+1} + \nabla p^{n+1} = 0 \text{ in } \Omega, \\ \nabla\cdot u^{n+1} = 0 \text{ in } \Omega. \end{cases}$$

We can anticipate that the reason to introduce the alternative methods of the following paragraphs is going to be the intrinsic saddle point nature of this problem, which leads to a standard *inf-sup* constraint. We will discuss about how to deal with this without using high order finite elements in section 5.4.3.2.

5.4.1.3 Fractional step methods

Up to the author's knowledge, the first attempt to speed up and simplify the numerical solution of the Navier−Stokes equations was brought to the community during 1968 [Chorin, 1968]. The underlying idea is to somehow decouple pressure and velocity by neglecting the pressure force in the momentum conservation equation. After, a correction step, equivalent to a projection in a function space of divergence free functions, would guarantee us physically reliable solutions.

The method reads as follows:

1. Viscous step: find $\tilde{u}\in\mathcal{H}$ such that:
$$\rho\frac{\tilde{u}-u^n}{\Delta t} + \rho\left(u^n\cdot\nabla u^{n+1}\right) - \mu\Delta u^{n+1} = 0,$$
enforcing the boundary conditions over the velocity.

2. Projection step: find $p^{n+1}\in H^1(\Omega)$ such that:
$$\Delta p^{n+1} = \frac{1}{\Delta t}\nabla\cdot\tilde{u},$$
enforcing the boundary conditions over the pressure.

3. Correction step: compute $u^{n+1}\in\mathcal{H}$ such that:
$$u^{n+1} = \tilde{u} + \Delta t\nabla p^{n+1},$$
enforcing the boundary conditions over the velocity.

Another possibility doable with MAD, is to use an incremental pressure correction scheme, which reads as follows:

1. Viscous step: find $\tilde{u} \in \mathcal{H}$ such that:

$$\rho \frac{\tilde{u} - u^n}{\Delta t} + \rho \left(u^n \cdot \nabla u^{n+1} \right) - \mu \Delta u^{n+1} + \nabla p^n = 0,$$

   enforcing the boundary conditions over the velocity.

2. Projection step: find $p^{n+1} \in H^1(\Omega)$ such that:

$$\Delta p^{n+1} = \frac{1}{\Delta t} \nabla \cdot \tilde{u} + \Delta p^n,$$

   enforcing the boundary conditions over the pressure.

3. Correction step: Compute $u^{n+1} \in \mathcal{H}$:

$$u^{n+1} = \tilde{u} + \Delta t \left(\nabla p^{n+1} - \nabla p^n \right).$$

   enforcing the boundary conditions over the velocity.

If we neglect the convective term from the equations (so we have an Stokes problem), it is possible to show [Guermond et al., 2005] that this incremental method improves the approximation accuracy with respect to classical Chorin−Teman.

5.4.2 RCR 0D model for Neumann boundaries

In order to deliver a realistic Neumann boundary condition for the problem, we can parametrize MAD to compute an average pressure $\overline{p}$ on Γ_{out} by means of a 0D model inspired in some of the phenomenological ideas exposed in section 5.2.2. A deep survey on this type of models can be found in [Formaggia et al., 2009].

Basically, we will simulate the flow behavior beyond the domain boundaries with an RCR circuit:

$$\overline{p} = p_d + R_p \int_{\Gamma_{\text{out}}} u^n \cdot n \, \mathrm{d}x, \tag{5.9}$$

where R_p plays the role of a proximal resistance in the circuit and where $p_d : [0, T] \to \mathbb{R}$ is a distal pressure and is the solution to the ordinary differential equation (ODE):

$$\begin{cases} C_d \dfrac{dp_d}{dt} + \dfrac{p_d}{R_d} & = \int_{\Gamma_{\text{out}}} u^n \cdot n \, \mathrm{d}x, \\ p_d(0) & = p_d^0. \end{cases} \tag{5.10}$$

where $p_d^0 \in \mathbb{R}$ is a given initial condition. Naturally, C_d and R_d are named distal capacitance and distal resistance, respectively. This model is typically referred as Windkessel model.

MAD includes a minimal ODE solver that allows us to choose either an explicit approach, or a semi-implicit one. The later translates in nothing but in to compute the distal pressure evolution as:

$$p_d^{n+1} = \frac{\int_{\Gamma_{\text{out}}} u^n \cdot n \, \mathrm{d}x + \frac{C_d}{\Delta t} p_d^n}{\frac{C_d}{\Delta t} + \frac{1}{R_d}}.$$

The coupling of this 0D model with the Navier-Stokes equations will be described in section 5.4.3.1.

5.4.3 Space discretization with finite elements

A long path has been traveled since M.J. Turner *et al.* published their stiffness method in the Journal for Aeronautical Sciences [Turner et al., 1956] during 1956. The intuitive point of view of this epoch has remained untouched in the introductory undergraduate courses for structural analysis. The mathematical foundations that lead the *way of thinking the problems* nowadays are going to be briefly discussed in what follows, in order to understand the underlying mechanism of MAD.

For the sake of compactness we are just drawing a sketch on the discretization of the Monolithic approach.

The weak formulation with BDF1 of (5.8) after integration by parts may be written as:

$$\begin{aligned}\rho \int_\Omega \frac{u^{n+1} - u^n}{\Delta t} \cdot v \, \mathrm{d}x + \rho \int_\Omega (u^n \cdot \nabla u^{n+1}) \cdot v \, \mathrm{d}x + \mu \int_\Omega \nabla u^{n+1} : \nabla v \, \mathrm{d}x - \int_\Omega (\nabla \cdot v) \, p^{n+1} \, \mathrm{d}x \\ + \int_\Omega \left(\nabla \cdot u^{n+1}\right) q \, \mathrm{d}x + \int_{\Gamma_{\text{out}}} \left(\left[-\mu \frac{\nabla u^{n+1} + \nabla^T \left(u^{n+1}\right)}{2} + p^{n+1} I_{3\times 3} \right] \cdot n \right) \cdot v \, \mathrm{d}s = 0,\end{aligned} \tag{5.11}$$

$\forall\, (v, q) \in \mathcal{H}_0 \times L^2(\Omega)$, where $\mathcal{H}_0$ denotes test functions in $\mathcal{H}$ with zero trace on the Dirichlet boundaries.

The spatial domain Ω, assumed to have at least a Lipschitz boundary, is discretized with a mesh tessellation of tetrahedrons $\mathcal{T}_h = \{\mathcal{T}_1, \mathcal{T}_2, \dots \mathcal{T}_{N_e}\}$, with vertices $\{x_1, \dots, x_N\}$. The mesh tessellation is assumed to be uniform, and the sub-script $h \in [0, 1)$ denotes the tetrahedron size. On this geometry we define finite-dimensional spaces in which MAD approximates the unknowns u and p. Let $\mathcal{L}_1(x), \dots \mathcal{L}_N(x)$ be a basis of $\mathbb{P}_1(\mathcal{T}_h)$, the piece-wise linear Lagrange elements, which by construction satisfies $\mathcal{L}_i(x_j) = \delta_{ij}$, $\forall i, j = 1, \dots, N$.

We next project $u = [u_1, u_2, u_3] \in \mathcal{H}$ on $[\mathbb{P}_1(\mathcal{T}_h)]^3$ and $p \in L^2(\Omega)$ on $\mathbb{P}_1(\mathcal{T}_h)$, so $u_1 \approx \tilde{u}_1 = \sum_i^N u_1^i \mathcal{L}_i(x)$, $u_2 \approx \tilde{u}_2 = \sum_i^N u_2^i \mathcal{L}_i(x)$, $u_3 \approx \tilde{u}_3 = \sum_i^N u_3^i \mathcal{L}_i(x)$ and $p \approx \tilde{p} = \sum_i^N p_i \mathcal{L}_i(x)$. When no confusion arises, we will make no notation difference between p and its FEM approximation $\tilde{p}$. The problem has been reduced to finding the coefficients $\{p_i\}_{i=1}^N$, $\{u_1^i\}_{i=1}^N$, $\{u_2^i\}_{i=1}^N$ and $\{u_3^i\}_{i=1}^N$. Details on the approximation errors from the projection and the discretization can be found in books like [Ern and Guermond, 2013].

In MAD, two steps of the FEM solver heavily exploit multi-threading: matrix assemblage and solution of system of linear equations. The former is manually implemented with MPI using the way that PETSc distributes the degrees of freedom $\mathcal{N}$ among the computer processor units (in our delimited configuration we always have $\mathcal{N} = 4N$ for the monolithic approach), whereas the later almost does not require any attention from the MAD development point of view since it takes advantages of the multiple choices of already paralellized solvers inside PETSc.

Concerning the boundary conditions of the PDEs, MAD offers classical matrix blocking for Dirichlet boundaries. No penalization method nor Nitsche approaches are yet implemented. Concerning the non homogeneous Neumann boundary conditions, the implementation is a bit more interesting when dealing with mixed system such as (5.8). We have omitted details on the assembling of the discretization matrices, but we will in particular show in the next section the local assemblage of the Neumann boundary term.

5.4.3.1 3D-0D coupling at the discrete level

To illustrate the local assemblage and some discretization choices made in MAD, we will describe step by step the way in which non-homogeneous boundary conditions are applied. That is to say, we will focus on the integral of the outlet forces,

$$\int_{\Gamma_{\text{out}}} \left(\left[\mu \frac{\nabla u + \nabla^T u}{2} - p_{3\times 3} \right] \cdot n \right) \cdot v \, \mathrm{d}s.$$

We impose in the software a constant diagonal stress tensor on Γ_{out}. We do this by imposing $\left(\nabla u^{n+1} + \left(\nabla^T u^{n+1}\right)\right) \cdot n/2 = 0$ and assuming a constant pressure $p = \overline{p}$ on Γ_{out}. At every time step we will compute $\overline{p}$ from the Windkessel ODE (5.9).

We tessellate the domain outlet $\mathbb{T}_h = \{\mathbb{T}_1, \ldots, \mathbb{T}_{N_e^b}\}$, set of triangles, with $N_e^b \in \mathbb{N}^+$ the number of boundary elements and we denote by N_b the number of vertices on it. Next, let us consider the function $g(x) = (\overline{p} I_{3\times 3} n) \in \left[L^2(\Gamma_{\text{out}})\right]^3$, with components $g = [g_1, g_2, g_3]$. We can proceed to project g and v on $\mathbb{P}_1(\Gamma_{\text{out}})$ similarly to what we did in section 5.4.3, considering a basis $\{\mathcal{L}_1^b, \ldots, \mathcal{L}_{N_b}^b\}$. It is easy to see that this translates into adding to the system right hand side the term $M_b g$, where

$$(M_b)_{ij} = \int_{\Gamma_{\text{out}}} \mathcal{L}_i^b \, \mathcal{L}_j^b \, \mathrm{d}s,$$

is a mass matrix on the boundary triangulation. To assemble it, MAD considers isoparametric transformations $\Lambda_l : \mathbb{T}_l \to T_M$, for $l = 1, \ldots, N_e^b$, and where T_M is a *master* triangle in which quadrature rules are known, described with two coordinates $[\xi_1, \xi_2]$. Let us denote the Jacobians of this transformations by the matrices $J^l \in \mathbb{R}^{3\times 2}$, with entries $J_{ij}^l = \frac{\partial (\Lambda_l)_i}{\partial \xi_j}$. We can then

integrate over every triangle as follows:

$$(M_b)_{ij} = \sum_{k=1}^{N_e^b} \int_{T_M} \mathcal{L}_i^b \, \mathcal{L}_j^b \det\left((J^k)^T J^k\right) \, \mathrm{d}s.$$

The scaling factors $\det\left((J^k)^T J^k\right)$ are typically refereed to as the first fundamental form of differential geometry.

MAD is not different from other FEM softwares when it comes to the implementation of this ideas. Triangles and tetrahedrons are implemented with objects that contain the relevant information presented in this section, such as Jacobians, routines to compute elementary masses, quadrature rules, etc. At a higher level, the CFD library allows us the direct assembly of M_b without digging down on the integration routines.

5.4.3.2 Inf-sup stabilization of Monolithic approach

The reader might have noticed that in our discretization choices for the monolithic approach in MAD we are not ensuring the Ladyzhenskaya−Babuska−Brezzi conditions to address correctly our saddle point problem (something that could be easier to grasp if we think the pressure as a Lagrange multiplier). To avoid using high order elements, we use a traditional technique proposed in [Brezzi and Pitkaranta, 1984]. In short, what we will do is to perturb the mass conservation of the system with a stiffness−like term that scales with the square of the tetrahedron size. That is to say, the weak form of mass conservation (5.4) would look like this:

$$\int_\Omega (\nabla \cdot u)\, q \, \mathrm{d}x + h^2 \int_\Omega \nabla p \cdot \nabla q \, \mathrm{d}x = 0, \qquad \forall q \in L^2(\Omega), \tag{5.12}$$

where the perturbation term is consistent in the sense that it returns an incompressible flow as $h \to 0$.

5.4.4 3D Numerical in silico experiments

Let us run a simulation on a realistic geometry, depicted in figure 5.5. Besides the geometry, all the other simulation parameters are chosen to keep the in silico experiment simple, as a toy example. Having said so, we set $\mu = \rho = u_{\text{in}} = 1.0$. Time discretization is done using a backward Euler scheme with $\Delta t = 0.004$ [sec]. No-slip boundary conditions, i.e., $u = 0$ is set on Γ_{out} and a tension free Neumann condition is set on $\Gamma_{\text{out}} = \Gamma_{\text{out}}^1 \cup \Gamma_{\text{out}}^2 \cup \Gamma_{\text{out}}^3 \cup \Gamma_{\text{out}}^4$.

The discretization matrix without the convective term can be depicted in figure 5.4.4. In the figure we also see the stabilized matrix using the technique of section 5.4.3.2. This matrix is stored in memory by MAD in order to only compute the convective term at each time iteration. Once the final matrix is assembled, by default MAD uses the ASM preconditioner. The solution of the system of equations is done using the iterative solver GMRES.

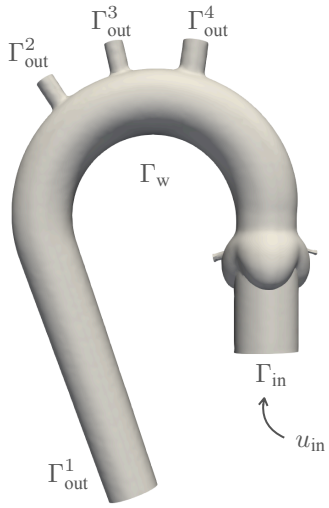
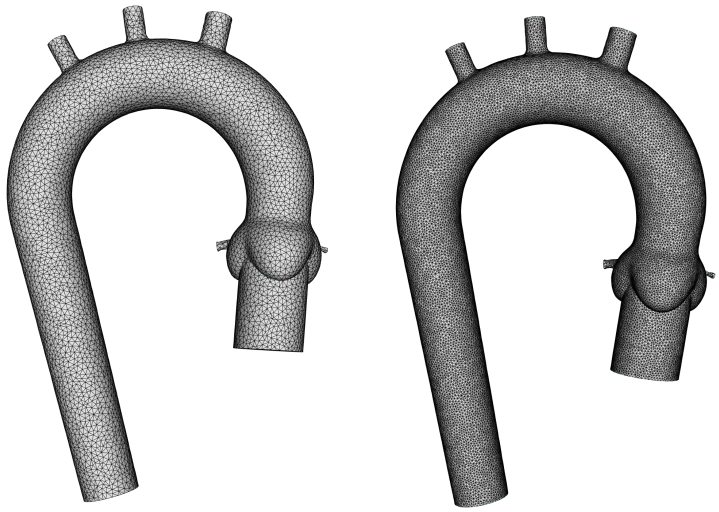
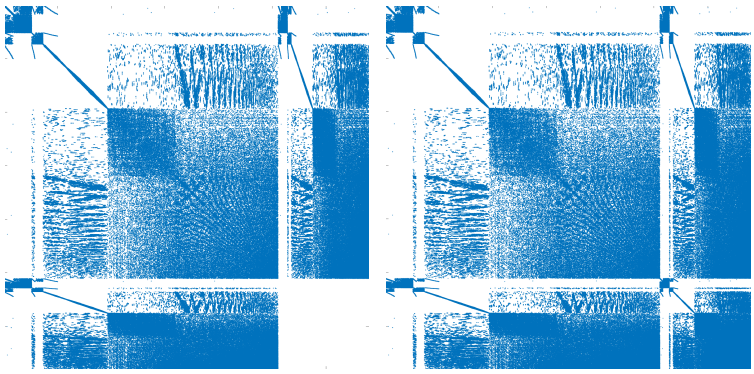


Figure 5.5: Working domain for in silico experiment.



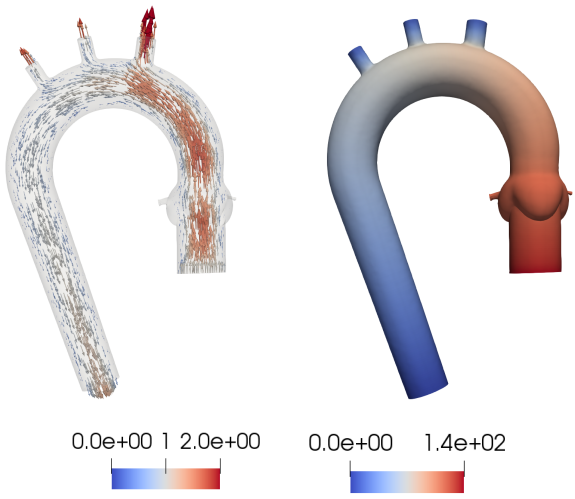
(a) $h = 1.6$ mm (b) $h = 0.8$ mm

Figure 5.6: Two tetrahedron tessellations for the working domain. The coarse mesh ($h = 1.6$ [mm]) is used to benchmark the parallel scaling of MAD, where as all the simulation results are shown with the fine mesh. The fine mesh ($h = 0.8$ [mm]) has $\sim 1.502 \times 10^6$ elements and it translates into a problem of $\sim 1.057 \times 10^6$ degrees of freedom.



(a) Discretization matrix. (b) Perturbed matrix.

Figure 5.7: Matrix of system of equations to be solved for a standard Navier−Stokes problem. We see a singular matrix and a stabilized one. For better visualization, we have chosen to show here the matrices for the mesh of size $h = 1.6$ [mm].



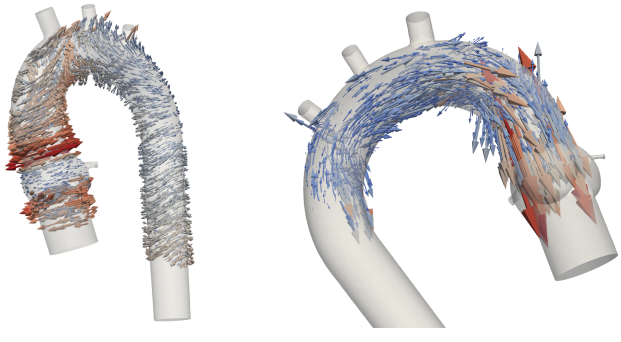
(a) Velocity (cm/s). (b) Pressure (bar).

Figure 5.8: Approximated Navier-Stokes solutions using MAD with 20 processing units.

The $\mathbb{P}_1(\Omega)$ solutions are shown on the geometry in figure 5.4.4, where both the velocity and the pressure fields are depicted once steady regime is attained, at $t = 0.4$ [sec]. In addition, MAD includes routines to compute quantities of interest (QoI), which are sometimes fields or numbers that encrypts meaningful information for the bio-medical engineering community. Among the QoI we can compute with MAD we count pressure jumps (from velocity data), flows, wall shear stress (WSS) and vorticity. For example, the field vorticity and wall shear stress are defined as:

$$\Theta(u) = \nabla \times u, \qquad \mathcal{S}(u) = 2\mu\{I_{3\times 3} - n \otimes n\} \left(\frac{\nabla u + \nabla^T u}{2} \right) \cdot n,$$

and they are shown in figure 5.4.4. This result is achieved by doing a *double interpolation* given the lack of regularity that $\mathbb{P}_1$ elements provide. Otherwise stated, in order to obtain the results shown, we first compute the solution gradient in $\mathbb{P}_0(\Omega)$ (per element), and then we re-interpolate the field to $\mathbb{P}_1(\Omega)$.



(a) Vorticity (b) Wall shear stress

Figure 5.9: Post-processing of velocity field from Navier-Stokes solutions.

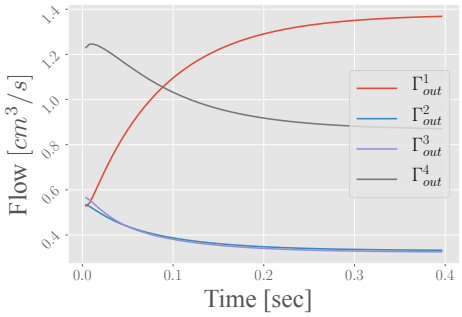
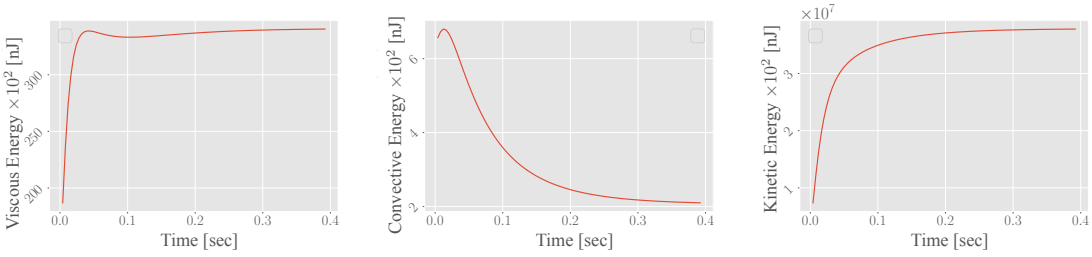


Figure 5.10: Flow on domain outlets

In addition, we can depict the evolution of the flow at the working domain outlets in figure 5.10. The tensor free boundaries explains why the flow is that big on Γ^4_{out} at the beginning of the simulation. In addition, another reason why this example can be called a toy or a semi-toy in silico experiment is due to the non-convective nature of the flow. Let us take a look at the energy of each term in the momentum conservation equation. We can define for this purpose the time dependent quadratic forms:

$$\begin{aligned} E_{\text{vis}}(t_k) &= u_k^T K u_k, \\ E_{\text{kin}}(t_k) &= u_k^T M u_k, \\ E_{\text{con}}(t_k) &= u_k^T C(u_k)\, u_k, \end{aligned} \tag{5.13}$$

that stands for viscous, kinetic and convective energy, respectively. The matrices $K \in \mathbb{R}^{3N\times 3N}$, $M \in \mathbb{R}^{3N\times 3N}$ and $C(u_k) \in \mathbb{R}^{3N\times 3N}$ are defined as $K_{ij} = \int \nabla\mathcal{L}_i \cdot \nabla\mathcal{L}_j \, \mathrm{d}x$, $M_{ij} = \int \mathcal{L}_i\mathcal{L}_j \, \mathrm{d}x$ and $C_{ij} = \int \nabla\mathcal{L}_j \cdot u_k \, \mathrm{d}x$. Their time evolution can be depicted in figure 5.11, where we indeed verify how negligible is the convective term respect to the others.



(a) Viscous energy (b) Convective energy (c) Kinetic energy

Figure 5.11: Time evolution of quadratic forms (5.13) of the velocity field.

Finally, let us perform a small high performance computing (HPC) benchmark to MAD using the aorta mesh of size $h = 1.6$ [mm]. In figure 5.12 we see the CPU time scaling with the number of processors. The benchmark comprehends the simulation from top to bottom, that is to say, from deploying the MAD architecture, the assemblage of the static matrices, the assemblage of the convection matrix, and the solution of one time iteration. We can remark an almost linear scaling. In fact, the CPU time with 10 processors is $\sim$ 106 [sec], and $\sim$ 42 [sec] with 20 processors. Even more astonishing is the difference between serial computing, where times goes up to 8486 [sec], and two processors, where the performance immediately increase to 949 [sec].

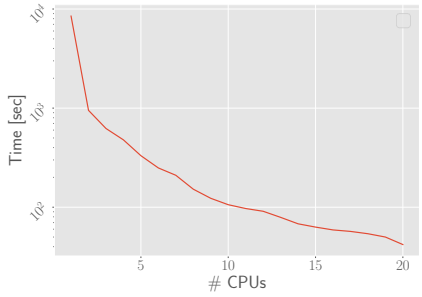


Figure 5.12: MAD and HPC. Parallel CPU time scaling for Navier-Stokes solver.

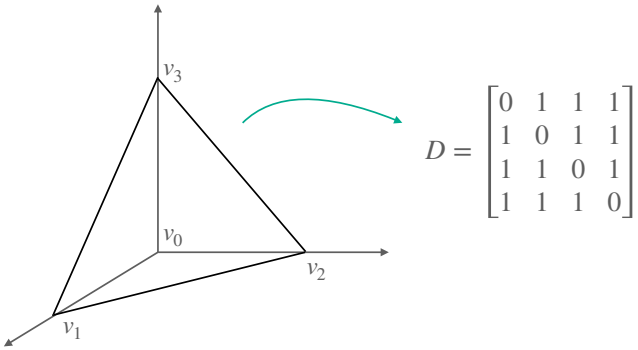


Figure 5.13: Input data set for toy MDS example.

5.5 Multi-dimensional scaling with MAD

A number of model order reduction (MOR) techniques are part of MAD, such as POD, Isomaps, data-driven algorithms and greedy methods. A benchmark between some of those techniques when addressing the data compression of a set of Navier-Stokes solutions can be found in chapter 3. In this section we will show an small example of an Euclidean multi-dimensional scaling (MDS), a non-linear MOR method that is going to be explored for a bio-medical application in chapter 4.

In short, the methodology is conceived to find a representation of high-dimensional data by embedding it in a space that is assumed to have an Euclidean structure. The data-set input is encoded in a dissimilarity matrix $D \in \mathbb{R}^{K\times K}$ with pair-wise distances among K samples in its entries. So, if we denote by X a matrix containing in each column one of those samples, we can enforce the Euclidean structure by writing down the inner product:

$$X^T X = -\frac{1}{2} H D^2 H, \tag{5.14}$$

where $H = I_{K\times K} - \frac{1}{K} e_K \otimes e_K$, a centering matrix that ensures a unique low dimensional

representation. Next, we can do the singular value decomposition:

$$-\frac{1}{2}HD^2H = USV^T,$$

and retrieve a parametric representation $Y \in \mathbb{R}^{K\times p}$, with $p < K$, by truncating the above equation as follows:

$$Y = S_p V_p^T.$$

where S_p is the truncation of S up to the first p singular values and V_p is a matrix containing the first p columns of V.

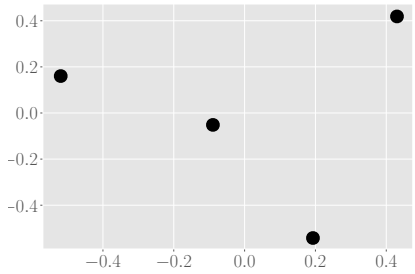


Figure 5.14: 2D representation of a tetrahedron using MDS.

The implementation of this method requires not much more than a correct interface with PETSc and SLEPc.

5.5.1 Example

Multi-dimensional scaling (MDS) is a key feature to run the numerical experiments of chapter 4 on variable domains. We can test MAD for MDS with a toy example by using as data-set the vertices of a tetrahedra, so the samples are the matrix $X^T = [v_1, \ldots, v_4]$, with the vertices in its columns (see figure 5.13 with the input dissimilarity matrix). MDS can retrieve the intrinsic degrees of freedom of the data: 3 of course. This example, in spite of its simplicity, provides a very intuitive glance on the mechanism of the method. The inner product matrix (5.14) for this example is:

$$X^T X = \begin{bmatrix} 0.375 & -0.125 & -0.125 & -0.125 \\ -0.125 & 0.375 & -0.125 & -0.125 \\ -0.125 & -0.125 & 0.375 & -0.125 \\ -0.125 & -0.125 & -0.125 & 0.375 \end{bmatrix}$$

For $p = 3$ we recover an exact representation of the tetrahedra, i.e, for $\hat{D}_{ij} \stackrel{\text{def}}{=} \|y_i - y_j\|$, we verify $D_{ij} = \hat{D}_{ij}$, where y_i are the columns of Y. The Gram matrix is singular and with MAD one retrieves the following eigen-values: $(0.5, 0.5, 0.5, 0.0)$. The MDS two dimensional representation of the tetrahedron can be depicted in figure 5.14.

5.6 Comments and acknowledgments

An overview of MAD have been given in this chapter, with a double goal: to show the software structure and to ambitiously convince the reader about the comprehensive scope of MAD. Complete understanding of the software can only be achieved by looking closer at MAD source code and the documentation that can be accessed by direct request to the author.

We would like to acknowledge Hernan Mella, Sergio Carrasco and Fannie Gerosa for accepting to test MAD on their personal computers.

Chapter 6

Future work and final conclusions

The big challenge to be addressed post-thesis is dealing with real measures. We have all the building blocks to perform reconstructions on patient-specific data. The collaboration with medical doctors and electrical engineers is crucial at this point. The field is intrinsically interdisciplinary.

6.1 Ultrasound data

Even though we have explored the assimilation of Doppler data intensely, even including synthetic noise, we have not yet used data acquired neither from plastic phantoms nor patients. Researchers from the COMMEDIA team and other collaborators have submitted a project whose goal is to validate the methodologies presented in this thesis with 3D-printed carotid arteries. The experimental set-up contemplates a pump with a liquid that mimics the blood's mechanical properties, from which Doppler measurements are going to be taken. This thesis could serve as the foundation for the project's experiments.

6.2 MRI data

The scope of the pipeline might go beyond Doppler data. In fact, as defined in chapter 2, we need not much more than data encoded in linear functionals of the space in which the state we want to reconstruct lives in. Magnetic resonance imaging (MRI) is an example of this. There is an ongoing collaboration with the *Biomedical Imaging Center* (CIB) at the Pontificia Universidad Católica de Chile. A complete set-up is already implemented, and it can be depicted in figure 6.1.

6.2.1 Experimental set-up

According to [J. et al., 2016] (where the experimental set-up is extensively described), the machine is a clinical 1.5T MR scanner (Philips Achieva, Best, the Netherlands) with a four-element phased-array body coil. The phantom can be scanned and posterior segmentation is done to obtain working domains to run simulations and assimilate data, as seen in figure 6.2.

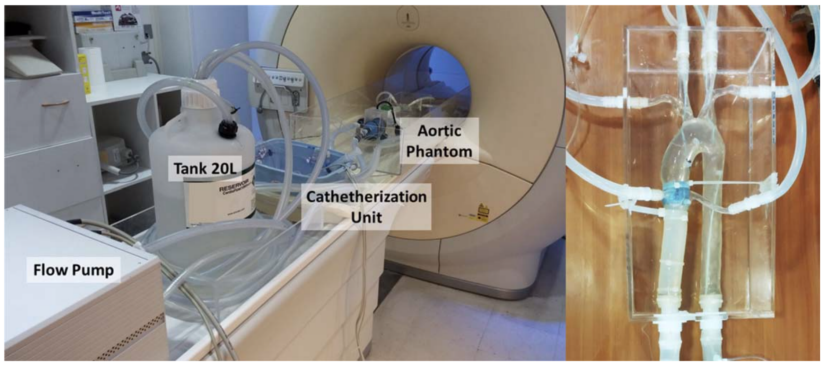
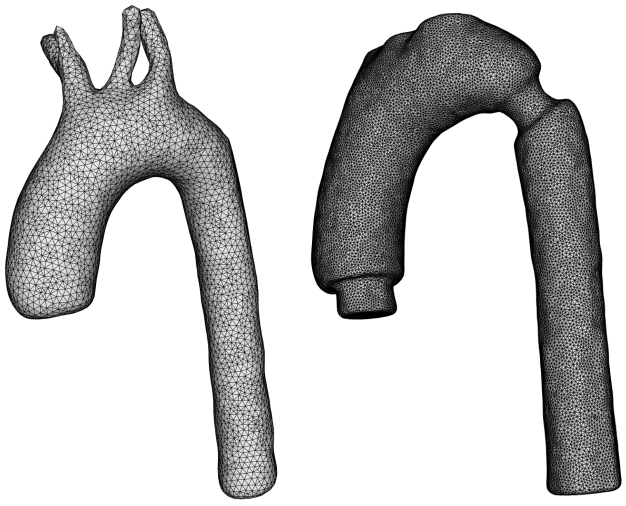


Figure 6.1: Phantom of the aorta and MR scanner, located at the Biomedical Imaging Center, Santiago de Chile. Picture taken from [Montalba et al., 2018]

.



(a) Healthy sample (b) Coarcted sample

Figure 6.2: Two tetrahedron meshes after segmentation of MRI data.

Researchers at CIB have full access to the acquisition pipeline from the scanner, which therefore allows flexibility in terms of defining the signal to be assimilated. Having said so, the problem can be posed in three different variations:

1. Data assimilation of 4D-flow: though very costly, this could improve the quality of the flow data by regularizing the noise, projecting the field in a divergence free space, and increasing the spatial resolution.

2. Data assimilation from 4D-flow slices: velocity data can be accesses in slices, in a similar manner that Doppler data is acquired, but with a lower time resolution, and instead of providing projections of the velocity in the ultrasound probe steering direction, they provide a 3D vector per image voxel.

3. Data assimilation from raw data: probably the more interesting alternative to be explored, we can proceed to assimilate directly the raw data from the scanner before the post-processing that produces the 4D-flow.

We can see an example of two 4D-flow snapshots interpolated over a tetrahedron mesh in figure 6.3. In [Garay et al., 2020], we can see numerical simulations over a phantom with stenosis (the level of stenosis can be parameterized, so that different configurations can be tested). In the document we also see how to tune our models, i.e., the Navier-Stokes equations and the ODE coupling so that we recover realistic simulations.

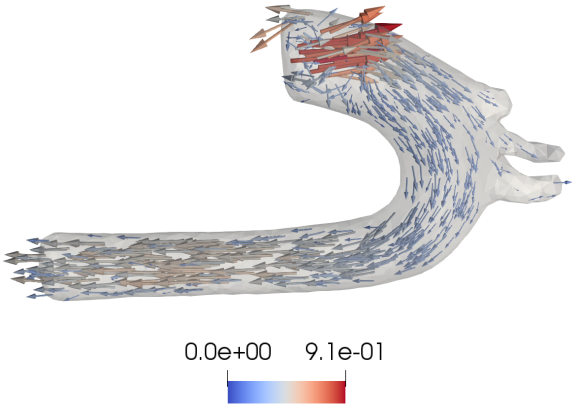
In adittion, the phantom set-up counts with a catheterization unit which allows us to compute pressure jumps as medical doctors do in the regular medical practice. Thus, this is a perfect scenario to test flows reconstruction, estimation of QoI, and our proposed machine-learning techniques for model reduction in non-parametric domains.

6.3 Forecasting

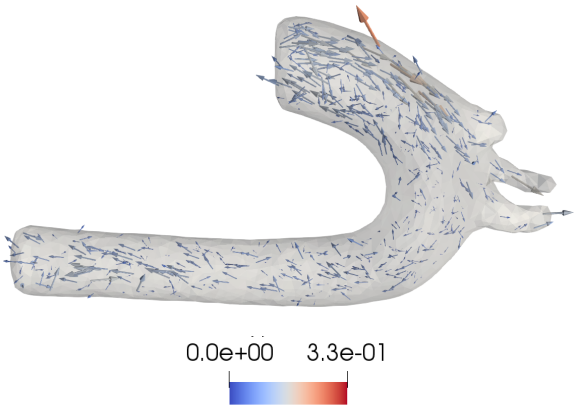
A *prediction and update* (P&U) algorithm is proposed for data assimilation in systems described with parameterized non-stationary PDEs. The methodology can be seen as a Kalman filter. The forecasting is not done thought for the state itself but for its projection coordinates into a reduced model of the governing laws. Therefore, the prediction and update algorithms require an offline phase of data compression, where time is not taken yet into account. The section is shown as part of the perspectives because we present a general idea without extensive numerical experiments.

6.3.1 Data assimilation for non-stationary PDEs

We introduce a method to assimilate data using reduced models of non stationary PDEs. So far, in chapters 2, 3 and 4 we have run numerical experiments using complex non-stationary models.



(a) velocity m/s



(b) velocity m/s

Figure 6.3: Two 4D-flow snapshots, courtesy of Hernan Mella from the *Biomedical Imaging Center*, Santiago de Chile. Upper snapshot is taken at peak systole. Bottom snapshot is taken at diastole.

Nevertheless, time has been taken as part of the parameter space. There is therefore potential to improve the results by using well-known forecasting techniques such as Kalman filters, originally proposed in [Kalman, 1960]. Modern works on filtering strategies for bio-medical applications are abundant. An example of data assimilation for blood flows can be found in [Caiazzo et al., 2017]. A complete study for cardiac mechanics can be found in [Moireau, 2008].

6.3.1.1 Model

We want to estimate $u \in \left[H^1(\Omega)\right]^3 \times [0,T]$ $(T > 0)$, when u is subjected to non-stationary governing dynamics, a PDE $\mathcal{P}(u(t), y) = 0$, where $y \in \mathbb{R}^p$ vector of p parameters in $Y \subset \mathbb{R}^p$. We can define the parameterized set of solutions

$$\mathcal{M} = \{u(t) \in \left[H^1(\Omega)\right]^3;\ \mathcal{P}(u(t); y) = 0,\ y \in Y,\ t \in [0,T]\}.$$

Remark that we have written $u(t)$ to emphasize that time is not part of the parameter space, as it is for chapters 2 and 3. Assume we have a sample of $\mathcal{M}$, a training set of functions each corresponding to a different parameter configuration $u(y_1, t), \ldots, u(y_{N_s}, t)$. We call N_s number of simulations.

Consider a mesh Ω_h of Ω with N vertices. Let $[\mathbb{P}_1(\Omega_h)]^3$ be the piece-wise linear space of Lagrange functions in Ω_h, a finite-dimensional approximation of $\left[H^1(\Omega)\right]^3$. We work with $[\mathbb{P}_1(\Omega_h)]^3$ just because in practice is the space we are going to use in our numerical experiments, it is easy to see that this is not a necessary assumption and the reader might choose another suitable finite-dimensional approximation for the ambient space. Let us project our training set into $[\mathbb{P}_1(\Omega_h)]^3$

$$u(y_i, t) \approx u_h^{(i)}(t) \stackrel{\text{def}}{=} P_{[\mathbb{P}_1(\Omega_h)]^3}(u(y_i, t)) = \sum_{j=1}^{\mathcal{N}} u_j^{(i)}(t)\mathcal{L}_j(x). \tag{6.1}$$

where $\mathcal{N} = 3N$ is the number of degrees of freedom and the functions $\{\mathcal{L}_i\}_{i=1}^{\mathcal{N}}$ are a basis for $[\mathbb{P}_1(\Omega_h)]^3$. Let us consider the uniform time partition of $[0,T]$: $[t_1, \ldots, t_K]$, with $K \stackrel{\text{def}}{=} T/\Delta t$ the number of time iterations, with Δt the partition size. We can define a discrete training set with the expansion of the training functions in $[\mathbb{P}_1(\Omega_h)]^3$, $u^{(i)}(t_k) = \left(u_1^{(i)}(t_k), \ldots, u_{\mathcal{N}}^{(i)}(t_k)\right)^T$, as defined in (6.1). Those coefficients are computable via finite elements for instance. Thus we introduce the set

$$\mathcal{M}_{\text{train}} = \{u^{(i)}(t_k) \in \mathbb{R}^{\mathcal{N}};\ i = 1, \ldots, N_s;\ k = 1, \ldots, K\}. \tag{6.2}$$

6.3.1.2 Measures

Let us denote by $\|\cdot\|$ and $\langle\cdot,\cdot\rangle$ the norm an inner product of the Hilbert space $\left[H^1(\Omega)\right]^3$. We model the m sensor devices by a linear operator $O : W_m \to \left[H^1(\Omega)\right]^3$. Every sensor separately

is modelled by a linear functional $\ell_i(u) : \left[H^1(\Omega)\right]^3 \to \mathbb{R}$. Let us denote by $\lambda_1, \ldots, \lambda_m$ the unique Riesz representers of those functionals, so

$$\ell_i(u) = \int_\Omega u \cdot \lambda_i \, \mathrm{d}x + \int_\Omega \nabla u : \nabla \lambda_i \, \mathrm{d}x, \quad i = 1, \ldots, m,$$

and let us denote by $\omega_1, \ldots, \omega_m$ the orthonormalized Riesz representers of the measures. Let us project them in $[\mathbb{P}_1(\Omega_h)]^3$, so we can see that the finite dimensional version of the operator O admits a matrix representation $W \in \mathbb{R}^{m \times \mathcal{N}}$ in which every column correspond to the coefficients of the expansion $P_{[\mathbb{P}_1(\Omega_h)]^3}(\omega_i)$. The matrix W is orthogonal by construction.

6.3.2 Time independent model compression

At the core of this methodology there are two things: a model reduction for the whole training set (6.2), but that takes into account the time by means of a forecasting algorithm. For this means, it is useful to rephrase the training set and separate it in different cloud of points per time step as follows

$$\mathcal{M}_{\text{train}}(t_k) = \{u^{(i)}(t_k) \in \mathbb{R}^{\mathcal{N}};\ i = 1, \ldots, N_{\text{s}}\}. \tag{6.3}$$

so that $\mathcal{M}_{\text{train}} = \cup_{i=1}^{K} \mathcal{M}_{\text{train}}(t_i)$. Next, we need to introduce a data compression technique for (6.2) and an algorithm to account for time between two contiguous sets (6.3), a *prediction and update* or forecasting that uses information $\mathcal{M}_{\text{train}}(t_k)$ and $\mathcal{M}_{\text{train}}(t_{k+1})$. We start by addressing the former with a singular value decomposition. Consider the snapshot matrix $\mathbb{A} \in \mathbb{R}^{\mathcal{N} \times (N_{\text{s}} K)}$ containing each realization of the training set $\mathcal{M}_{\text{train}}$ in its columns. The model reduction via SVD reads $\mathbb{A} = USV^T$, so we have an n-dimensional basis in the columns of U_n (a sub-matrix of U with the first n columns of U) for the space spanned by the elements in the set $\mathcal{M}_{\text{train}}$, which is optimal in the sense that $\min_{X_n \text{ s.t. } \text{Ran}(X_n)=n} \|\mathbb{A} - X_n X_n^T \mathbb{A}\|_{\text{F}}$ is solved by U_n. Let us denote that base by $\rho_1, \ldots, \rho_n$ and the space that it spans by V_n.

The novelty of this methodology is that we are going to address the time evolution of the state u only looking at the expansion of the training functions in V_n. Remark that this is advantageous due to two things: First, the estimation will require computations in $\mathbb{R}^n$, which is cheap if $\mathcal{P}(u; y)$ has a good decay in terms of its Kolmogorov n-width. Second, we will track the evolution of the state without knowing its parameters. This leads to an online phase estimation which is theoretically much faster than estimation of the state in $\mathbb{R}^{\mathcal{N}}$ and that does not require a middle step for parameter estimation.

6.3.3 Forecasting algorithm

In this section we bring time to the picture. We would like to understand the time evolution of the state from a certain time t_i to another subsequent time t_{i+1} by using the time independent model V_n and the set of solutions samplings $\mathcal{M}_{\text{train}}(t_i)$ and $\mathcal{M}_{\text{train}}(t_{i+1})$.

Assume we have a reconstruction of the target field $u \in \left[H^1(\Omega)\right]^3$ at t_1, say a field $u^*(t_1)$, and let us denote by $a(t_k) = (a_1(t_k), \dots, a_n(t_k))^T \in \mathbb{R}^n$ the coordinates of the projection $P_{V_n} u^*(t_k)$, $k = 1, \dots, K$. In order to compute $u^*(t_1)$ we could follow several paths, such as projecting every sample in $\mathcal{M}_{\text{train}}(t_1)$ and computing a mean, or to use a static algorithm with the given measures. We defer this choice to the numerical experiment of section 6.3.4.

The goal is now to design a prediction and update algorithm for the evolution of the coefficients $a(t_k)$. To this end, we will use a linear propagation model. We can remark that:

Remark 6.3.1. *A linear propagation in the coefficients of the samples when projected into V_n does not mean that the underlying model has to be linear. It is correct to say that the model, encrypted in the parameterized PDE $\mathcal{P}(u(t), y)$ can be non linear, but we compress a sample of this PDE in a linear space that allows us to do propagations that have information about the non-linearities of the governing laws. One could argue about the quality of linear spaces to reduce the PDE, so here is certainly when we can set an intuitive perimeter for this methods: a prediction and update algorithm for non linear PDEs with moderately fast decaying Kolmogorov n-widths.*

In practice, at each time step t_i we are given a vector of measures $l(t_i) \in \mathbb{R}^m$, coordinates of an element in $\omega \in W_m$. We assimilate the data at each time step in two sub-steps, described in the following sections.

6.3.3.1 Prediction

Let us adopt a notation more familiar with Kalman filtering. We call $a_{k|k} = a(t_k)$ the *prior* estimation. Let us project every entry in $\mathcal{M}_{\text{train}}(t_k)$ in V_n and let us place the coordinates of this projections, the vectors $\{a^{(i)}(t_k)\}_{i=1}^{N_s}$, $(a^{(i)}(t_k) \in \mathbb{R}^n)$, in the columns of the matrices $\mathbb{A}(t_k) \in \mathbb{R}^{n \times N_s}$. We propose to do a prediction step with a linear combination of the sampling of $\mathcal{M}$ at t_{k+1}. Otherwise stated, let $w \in \mathbb{R}^K$ ve a vector of weights, our prediction is then $a_{k+1|k} = \mathbb{A}(t_{k+1})w$, and we choose to compute the weights by means of a least squares fitting at t_k:

$$w = \underset{w \in \mathbb{R}^K}{\arg\inf} \|a_{k|k} - \mathbb{A}(t_k)w\|_{\ell^2}^2, \tag{6.4}$$

We assume a good sampling for the set of solutions, that is to say, we assume that K is much bigger than n. Having said so, we can regularize (6.4) with an SVD, $\mathbb{A}(t_k) = USV^T$, so we compute a regularized pseudo-inverse up to the dimension of V_n: $\hat{\mathbb{A}}^\dagger(t_k) = U_{(n)} S_{(n)}^{-1} V_{(n)}^T$ (recall that for a matrix X we denote X_n the sub-matrix with the first n columns of X). The predicted coefficients are thus:

$$a_{k+1|k} = \mathbb{A}(t_{k+1}) \left(\hat{\mathbb{A}}^\dagger(t_k) a_{k|k} \right).$$

The *prior* state has an a-prior covariance $C_{k|k} \in \mathbb{R}^{n\times n}$, we have to propagate it as well, with the same linear dynamics we have used for the state. We introduce the propagation matrix $\mathbb{F} = \mathbb{A}(t_{k+1})\hat{\mathbb{A}}^{\dagger}(t_k)$. Thus,

$$C_{k+1|k} = \mathbb{F}C_{k|k}\mathbb{F}^T + \mathbb{O},$$

where $\mathbb{O} \in \mathbb{R}^{n\times n}$ is assumed to be symmetric and positive definite, to be defined in section 6.3.3.3.

6.3.3.2 Update

We are in order to compute the update $a_{k+1|k+1}$. Let us introduce the *discrepancy* between model and measures, the vector $z \in \mathbb{R}^m$ defined as

$$z = Ga_{k+1|k+1} - l(t_{k+1})$$

where $G \in \mathbb{R}^{m\times n}$ is a cross-Gramian matrix between V_n and W_m, so its entries are $G_{ij} = \langle \omega_i, \rho_j \rangle$. The observation process also is perturbed by noise, and it has therefore an associated covariance matrix $\Sigma \in \mathbb{R}^{m\times m}$.

Notice that both $C_{k+1|k}$ and Σ are symmetric and positive definite by construction. We can define metrics with them. Having said so, the update might be computed by minimizing the joint functional

$$\mathcal{J} = \frac{1}{2}\|a_{k+1|k+1} - a_{k+1|k}\|^2_{C^{-1}_{k+1|k}} + \frac{1}{2}\|z\|^2_{\Sigma^{-1}}. \tag{6.5}$$

Let us optimize it,

$$\begin{aligned}\frac{\partial}{\partial a_{k+1|k+1}}\mathcal{J} = \frac{\partial}{2\partial a_{k+1|k+1}}\{&\left(a_{k+1|k+1} - a_{k+1|k}\right)^T C^{-1}_{k+1|k}\left(a_{k+1|k+1} - a_{k+1|k}\right)\\ &+ \left(Ga_{k+1|k+1} - l(t_{k+1})\right)^T \Sigma^{-1}\left(Ga_{k+1|k+1} - l(t_{k+1})\right)\}\\ &= \left(C^{-1}_{k+1|k} + G^T\Sigma^{-1}G\right)a_{k+1|k+1} - \left(C^{-1}a_{k+1|k} + G^T\Sigma^{-1}l(t_{k+1})\right).\end{aligned}$$

Thus, the update step reads: find $a_{k+1|k+1} \in \mathbb{R}^n$ such that:

$$\left(C^{-1}_{k+1|k} + G^T\Sigma^{-1}G\right)a_{k+1|k+1} = \left(C^{-1}_{k+1|k}a_{k+1|k} + G^T\Sigma^{-1}l(t_{k+1})\right). \tag{6.6}$$

.

We can update the co-variance as well using its definition with the data from $\mathcal{M}_{\text{train}}(t_{k+1})$ biased at $a_{k+1|k+1}$. The state is recovered then by doing $u^*(t_{k+1}) = \sum_{i=1}^n \left(a_{k+1|k+1}\right)_i \rho_i$.

6.3.3.3 Quantification of covariances

We have to choose a method to compute the matrix $\mathbb{O}$ and the matrix Σ. This matrices are typically referred as model and measures covariances of the noise. This could be misleading, as

deep down what we are trying to account with those matrices is how much we *trust model and measures*. This is very easy to see for Σ: imagine that we have an ideal device that provides noise-free measures, but that only is capable of seeing a fraction of the domain, or an aggregated quantity of the state. Will we trust absolutely the observations and do our reconstruction model-blindly $A(\omega) = \omega$? Of course not. We introduce the following systematic manner to account for this:

$$\begin{aligned} \mathbb{O} &= \left(\frac{1}{N_\mathrm{s}} \sum_{i=1}^{N_\mathrm{s}} \| u - P_{V_n}(u) \|^2 \right) I_{n\times n} \\ \Sigma &= \left(\sigma_m^2 + \frac{1}{N_\mathrm{s}} \sum_{i=1}^{N_\mathrm{s}} \| u - P_{W_m}(u) \|^2 \right) I_{m\times m} \end{aligned} \tag{6.7}$$

so that we account for the error in the orthogonal projection into $V_n^\perp$, and we account for how *rich* the space of observations is by doing the same with $W_m^\perp$. σ_m is the standard deviation of the observation noise.

6.3.4 Toy Stokes numerical example

We generate the set $\mathcal{M}_{\text{train}}$ by sampling the Stokes equations over $\Omega \subset \mathbb{R}^3$, the dilated tube seen at 6.4. So, the sampling problem reads: find $u \in [H^(\Omega)]^3$ and $p \in L^2(\Omega)$ such that:

$$\begin{cases} \partial_t u - \mu \Delta u + \nabla p = 0 \text{ in } \Omega, \\ \nabla \cdot u = 0 \text{ in } \Omega, \\ u = (0,0,0) \text{ on } \Gamma_\mathrm{w}, \\ u = u_\mathrm{in}\{0, \left(1 - \dfrac{x^2+z^2}{(D/2)^2}\right), 0\} \sin(2\pi t) \text{ on } \Gamma_\mathrm{in}, \\ \left(\dfrac{\nabla u + \nabla^T u}{2} - p I_{3\times 3} \right) \cdot n = (0,0,0) \text{ on } \Gamma_\mathrm{out}. \end{cases} \tag{6.8}$$

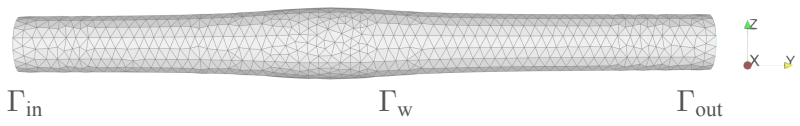


Figure 6.4: Admittedly simple working domain to test out prediction and update algorithm.

The parameter space is set equally as it was done in section 4.3.3. In this context, $N_\mathrm{s} = 500$ and $K = 25$. We propose to compute the first prior to start the prediction and update algorithm by using the least squares term in PBDW, so we get $a_{1|1} \in V_n$. That is to say, the algorithm is

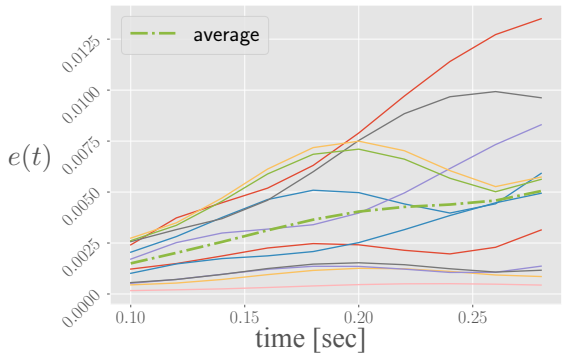
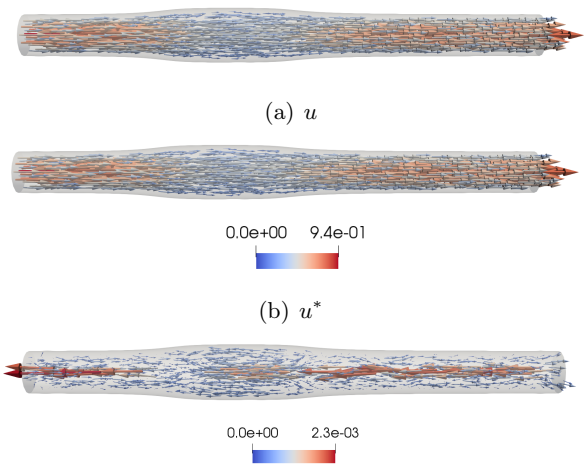


Figure 6.5: $[H^1(\Omega)]^3$ error (6.9) temporal evolution for 12 simulations and their average. The space V_n is selected with a tolerance of 10^{-2}.



(a) u

(b) u^*

(c) $u - u^*$

Figure 6.6: Reconstruction of one field in range with the parameter space to generate the set of solutions. The snapshots is taken at the peak velocity time.

initialized as follows: find $a_{1|1}$ such that:

$$\min_{a_{1|1}\in\mathbb{R}^n} \frac{1}{2}\|Ga_{1|1} - l(t_1)\|^2_{\ell^2}.$$

In addition, we initialize the covariance matrix as follows

$$C_{1|1} = \frac{1}{N_\mathrm{s}} \sum_{i=1}^{N_\mathrm{s}} \left(a^{(i)}(t_1) - \bar{a}(t_1)\right) \otimes \left(a^{(i)}(t_1) - \bar{a}(t_1)\right)$$

where $\bar{a}(t_1) = \frac{1}{N_\mathrm{s}} \sum_{i=1}^{N_\mathrm{s}} a^{(i)}(t_1)$. Figure 6.5 reveals the effectiveness of the algorithm when evaluating the error with

$$e(t) = \frac{\|u - u^*\|^2}{\int_0^{0.5} \|u\|^2 \ \mathrm{dt}} \tag{6.9}$$

A single snapshot with the ground truth, the forecasting reconstruction and the error field can be seen in figure 6.6.

6.4 Inter-patient variability

Last but not least, it is part of the near future work to integrate the content of chapter 4 on inter-patient variability with the state estimation methodologies of chapters 2 and 3, or even of the forecasting methodology proposed in previous section. A scientific communication is expected to be produced with this content.

Bibliography

[Aderson, 1998] Aderson, M. (1998). Multi-dimensional velocity estimation with ultrasound using spatial quadrature. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 45(3):852–861.

[Adib et al., 2016] Adib, M. A. H. M., Ii, S., Watanabe, Y., and Wada, S. (2016). Patient-specific blood flows simulation on cerebral aneurysm based on physically consistency feedback control. pages 334–337. IEEE 16th International Conference on Bioinformatics and Bioengineering (BIBE).

[Aidarous et al., 1975] Aidarous, S. E., Gevers, M. R., and Installe, M. J. (1975). Optimal sensors allocation strategies for a class of stochastic distributed systems. *International journal of Control*, 22(2):197–213.

[Akkari et al., 2019] Akkari, N., Casenave, F., and Ryckelynck, D. (2019). A novel gappy reduced order method to capture non-parameterized geometrical variation in fluid dynamics problems. *Pre-print hal-02344342v2*.

[Amsallem and Haasdonk, 2016] Amsallem, D. and Haasdonk, B. (2016). PEBL-ROM: Projection-error based local reduced-order models. *Advanced Modeling and Simulation in Engineering Sciences*, 3(1):6.

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

[Argaud et al., 2017] Argaud, J. P., Bouriquet, B., Gong, H., Maday, Y., and Mula, O. (2017). Stabilization of (G)EIM in presence of measurement noise: Application to nuclear reactor physics. Spectral and High Order Methods for Partial Differential Equations ICOSAHOM 2016, Rio de Janeiro, Brazil, Springer International Publishing.

[Astrid et al., 2008] Astrid, P., Weiland, S., Willcox, K., and Backx, T. (2008). Missing point estimation in models described by proper orthogonal decomposition. *IEEE Transactions on Automatic Control*, 53(10):2237–2251.

[Ayachit, 2015] Ayachit, U. (2015). *The ParaView guide: a parallel visualization application*. Kitware.

[Balay et al., 1997] Balay, S., Gropp, W. D., McInnes, L. C., and Smith, B. F. (1997). Efficient management of parallelism in object oriented numerical software libraries. In Arge, E., Bruaset, A. M., and Langtangen, H. P., editors, *Modern Software Tools in Scientific Computing*, pages 163–202. Birkhäuser Press.

[Barrault et al., 2004] Barrault, M., Maday, Y., Nguyen, N. C., and Patera, A. T. (2004). An Empirical Interpolation Method: application to efficient reduced-basis discretization of partial differential equations. *C. R. Acad. Sci. Paris, Série I.*, 339(9):667–672.

[Barrie and Webb, 2011] Barrie, N. and Webb, A. (2011). *Introduction to Medical Imaging*. Cambridge University Press.

[Bensoussan, 1972] Bensoussan, A. (1972). Optimization of sensors' location in a distributed filtering problem. pages 62–84. Springer.

[Berkooz et al., 1993] Berkooz, G., Holmes, P., and Lumley, J. L. (1993). The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25(1):539–575.

[Bertoglio et al., 2017a] Bertoglio, C., Caiazzo, A., Y., B., Braack, M., Esmaily-Moghadam, M., Gravemeier, V., Marsden, A., Pironneau, O., Vignon-Clementel, I., and Wall, W. (2017a). Benchmark problems for numerical treatment of backflow at open boundaries. *International Journal for Numerical Methods in Bio-medical Engineering*, 34(2):e2918.

[Bertoglio et al., 2017b] Bertoglio, C., Nunez, R., Galarce, F., Nordsletten, D., and Osses, A. (2017b). Relative pressure estimation from velocity measurements in blood flows: State-of-the-art and new approaches. *International Journal for Numerical Methods in Bio-medical Engineering*, 34(2).

[Binev et al., 2017a] Binev, P., Cohen, A., Dahmen, W., DeVore, R., Petrova, G., and Wojtaszczyk, P. (2017a). Data assimilation in reduced modeling. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):1–29.

[Binev et al., 2017b] Binev, P., Cohen, A., Dahmen, W., DeVore, R., Petrova, G., and Wojtaszczyk, P. (2017b). Data assimilation in reduced modeling. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):1–29.

[Binev et al., 2018] Binev, P., Cohen, A., Mula, O., and Nichols, J. (2018). Greedy algorithms for optimal measurements selection in state estimation using reduced models. *SIAM/ASA journal on Uncertainty Quantification*, 6(3):1101–1126.

[Bjaerum et al., 2002] Bjaerum, S., Torp, H., and Kristoffersen, K. (2002). Clutter filter design for ultrasound color flow imaging. *IEEE transactions on ultrasonics, ferroelectrics, and frequency control*, 49(2):204–216.

[Blanco et al., 2015] Blanco, P., Watanabe, S., Passos, M., Lemos, P., and Feijoo, R. (2015). An anatomically detailed arterial network model for one-dimensional computational hemodynamics. *IEEE transactions on biomedical engineering*, 62(2):736–753.

[Bluestein et al., 2010] Bluestein, D., Chandran, K., and Manning, K. (2010). Towards nonthrombogenic performance of blood recirculating devices. *Annals of Biomedical Engineering*, 38(3):1236–1256.

[Brezzi and Pitkaranta, 1984] Brezzi, F. and Pitkaranta, J. (1984). On the stabilization of finite element approximations of the stokes equations. *Efficient Solutions of Elliptic Systems*.

[Brooks and Hughes, 1982] Brooks, A. and Hughes, T. (1982). Streamline upwind/petrov-galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 32(1–3):199–259.

[Buffoni et al., 2008] Buffoni, M., Camarri, S., Iollo, A., Lombardi, E., and Salvetti, M. (2008). A non-linear observer for unsteady three-dimensional flows. *Journal of Computational Physics*, 227(4):2626–2643.

[Caiazzo et al., 2017] Caiazzo, A., Caforio, F., Montecinos, G., Muller, L., Blanco, P., and Toro, E. F. (2017). Assessment of reduced-order unscented kalman filter for parameter identification in 1-dimensional blood flow models using experimental data. *International journal for numerical methods in biomedical engineering*, 33(8):e2843.

[Cannon and Klein, 1971] Cannon, J. and Klein, R. (1971). Optimal selection of measurement locations in a conductor for approximate determination of temperature distributions. *ASME journal of Dynamic Systems, Measurement, and Control*, 93(3):193–199.

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

[Charlier et al., 2020] Charlier, B., Feydy, J., Glaunès, J. A., Collin, F. D., and Durif, G. (2020). Kernel operations on the GPU, with autodiff, without memory overflows. *E-prints, arXiv:2004.11127*.

[Charonko et al., 2013] Charonko, J., Kumar, R., Stewart, K., Little, W., and Vlachos, P. (2013). Vortices formed on the mitral valve tips aid normal left ventricular filling. *Annals of biomedical engineering*, 41(5):1049–1061.

[Chorin, 1968] Chorin, A. (1968). Numerical solution of the navier stokes equations. *Math. Comput.*, 22(104):745–762.

[Ciarlet, 1988] Ciarlet, P. (1988). *Mathematical Elasticity. Studies in Mathematics and its Applications*. North-Holland Publishing Co.

[Cohen et al., 2019] Cohen, A., Dahmen, W., DeVore, R., Fadili, J., Mula, O., and Nichols, J. (2019). Optimal reduced model algorithms for data-based state estimation. *Pre-print arXiv:1903.07938*.

[Cohen et al., 2020] Cohen, A., Dahmen, W., Mula, O., and Nichols, J. (2020). Nonlinear reduced models for state and parameter estimation. *Pre-print arXiv:2009.02687*.

[Cohen and DeVore, 2015] Cohen, A. and DeVore, R. (2015). Approximation of high-dimensional parametric pdes. *Acta Numerica*, 24:1–159.

[Cohen et al., 2011] Cohen, A., DeVore, R., and Schwab, C. (2011). Analytic regularity and polynomial approximation of parametric and stochastic elliptic PDE's. *Analysis and Applications*, 9(1):11–47.

[Dapogny et al., 2014] Dapogny, C., Dobrzynski, C., and Frey, P. (2014). Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems. *Journal of Computational Physics*, 262:358–378.

[Dashti and Stuart, 2017] Dashti, M. and Stuart, A. M. (2017). *The Bayesian Approach to Inverse Problems*, pages 311–428. Springer International Publishing, Cham.

[Demené et al., 2015] Demené, C., Deffieux, T., Pernot, M., Osmanski, B.-F., Biran, V., Gennisson, J.-L., Sieu, L.-A., Bergel, A., Franqui, S., Correas, J.-M., et al. (2015). Spatiotemporal clutter filtering of ultrafast ultrasound data highly increases doppler and fultrasound sensitivity. *IEEE transactions on medical imaging*, 34(11):2271–2285.

[Ern and Guermond, 2013] Ern, A. and Guermond, J. (2013). *Theory and practice of finite elements*, volume 159. Springer Science & Business Media.

[Formaggia et al., 2009] Formaggia, L., Quarteroni, A., and Veneziani, A. (2009). *Cardiovascular Mathematics. Modeling and simulation of the circulatory system*, volume 1. Springer-Verlag Mailand.

[Franz et al., 2014] Franz, T., Zimmermann, R., Görtz, S., and Karcher, N. (2014). Interpolation-based reduced-order modelling for steady transonic flows via manifold learning. *International Journal of Computation Fluid Dynamics*, 28:106–121.

[Funamoto and Hayase, 2013] Funamoto, K. and Hayase, T. (2013). Reproduction of pressure field in ultrasonic-measurement-integrated simulation of blood flow. *International journal for numerical methods in biomedical engineering*, 29(7):726–740.

[Galarce et al., 2020] Galarce, F., Gerbeau, J.-F., Lombardi, D., and Mula, O. (2020). Fast reconstruction of 3D blood flows from doppler ultrasound images and reduced models. *Accepted on Computer Methods in Applied Mechanics and Engineering*.

[Garay et al., 2020] Garay, J., Mella, H., Sotelo, J., Cárcamo, C., Uribe, S., Bertoglio, C., and Mura, J. (2020). A new mathematical model for verifying the navier-stokes compatibility of 4d flow mri data. *pre-print*, hal-02957167.

[Garcia et al., 2013] Garcia, J., Larose, E., Pibarot, P., and Kadem, L. (2013). On the evaluation of vorticity using cardiovascular magnetic resonance velocity measurements. *Journal of Biomechanical Engineering*, 135(12):124501.

[Gibson et al., 1993] Gibson, C., Diaz, L., Kandarpa, K., Sacks, F., Pasternak, R., Sandor, T., Feldman, C., and Stone, P. (1993). Relation of vessel wall shear stress to atherosclerosis progression in human coronary arteries. *Arteriosclerosis and thrombosis: a journal of vascular biology*, 13(2):310–315.

[Gong et al., 2019] Gong, H., Maday, Y., Mula, O., and Taddei, T. (2019). PBDW method for state estimation: error analysis for noisy data and nonlinear formulation. *Pre-print arXiv:1906.00810*.

[Guermond et al., 2005] Guermond, J., Minev, P., and Shen, J. (2005). An overview of projection methods for incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 195(44–47).

[Guibert et al., 2013] Guibert, R., McLeod, K., Caiazzo, A., Mansi, T., Fernandez, M., Sermesant, M., Pennec, X., Vignon-Clementel, I., Boudjemline, Y., and Gerbeau, J. F. (2013). Group-wise construction of reduced models for understanding and characterization of pulmonary blood flows from medical images. *Medical Image Analysis*, Elsevier(1):63–82.

[Guyton and Hall, 2011] Guyton, A. and Hall, J. (2011). *Textbook of Medical Physiology*. Elsevier.

[Hata et al., 1987] Hata, T., Aoki, S., Hata, K., and Kitao, M. (1987). Intracardiac blood flow velocity waveforms in normal fetuses in utero. *The American journal of cardiology*, 59(5):464–468.

[Hatle et al., 1979] Hatle, L., Angelsen, B., and Tromsdal, A. (1979). Noninvasive assessment of atrioventricular pressure half-time by doppler ultrasound. *Circulation*, 60(5):1096–1104.

[Hatle et al., 1980] Hatle, L., Angelsen, B., and Tromsdal, A. (1980). Non-invasive assessment of aortic stenosis by doppler ultrasound. *Heart*, 43(3):284–292.

[Hatle et al., 1978] Hatle, L., Brubakk, A., Tromsdal, A., and Angelsen, B. (1978). Noninvasive assessment of pressure drop in mitral stenosis by doppler ultrasound. *Heart*, 40(2):131–140.

[Hecht, 2012] Hecht, F. (2012). New development in FreeFem++. *Journal of Numerical Mathematics*, 20(3–4):251–265.

[Heo et al., 2014] Heo, K., Fujiwara, K., and Abe, J. (2014). Shear stress and atherosclerosis. *Molecules and Cells*, 36(6):435–440.

[Hernandez et al., 2005] Hernandez, V., Roman, J., and Vidal, V. (2005). SLEPc: A scalable and flexible toolkit for the solution of eigenvalue problems. *ACM Transactions on Mathematical Software*, 31(3).

[Hirtler et al., 2016] Hirtler, D., Garcia, J., Barker, A., and Geiger, J. (2016). Assessment of intracardiac flow and vorticity in the right heart of patients after repair of tetralogy of fallot by flow-sensitive 4D MRI. *European radiology*, 26(10):3598–3607.

[Hu et al., 2007] Hu, X., Nenov, V., Bergsneider, M., Glenn, T. C., Vespa, P., and Martin, N. (2007). Estimation of hidden state variables of the intracranial system using constrained nonlinear kalman filters. *IEEE transactions on biomedical engineering*, 54(4):597–610.

[Huttunen et al., 2020] Huttunen, J. M., Kärkkäinen, L., Honkala, M., and Lindholm, H. (2020). Deep learning for prediction of cardiac indices from photoplethysmographic waveform: A virtual database approach. *International journal for Numerical Methods in Biomedical Engineering*, 36(3):e3303.

[J. et al., 2016] J., U., Sotelo, J., Springmüller, D., Montalba, C., Letelier, K., Tejos, C., Irarrázaval, P., Andia, M., Razavi, R., Valverde, I., and Uribe, S. (2016). Realistic aortic phantom to study hemodynamics using mri and cardiac catheterization in normal and aortic coarctation conditions. *J. Magn. Reson. Imaging*, 44(3):683–697.

[Jensen, 1906] Jensen, J. (1906). Sur les fonctions convexes et les inégualités entre les valeurs moyennes. In *Acta Mathematica*.

[Joshi and Miller, 2000] Joshi, S. and Miller, M. (2000). Landmark matching via large deformation diffeomorphisms. *IEEE transactions on Image Processing*, 9(8):1357–1370.

[Kaipio and Somersalo, 2005] Kaipio, J. and Somersalo, E. (2005). *Statistical and computational inverse problems*. Springer.

[Kalman, 1960] Kalman, R. (1960). A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering*.

[Kärcher et al., 2018] Kärcher, M., Boyaval, S., Grepl, M. A., and Veroy, K. (2018). Reduced basis approximation and a posteriori error bounds for 4D-Var data assimilation. *Optimization and Engineering*, 19:663–695.

[Karush, 1939] Karush, W. (1939). Minima of functions of several variables with inequalities as side constraints.

[Khalil et al., 2007] Khalil, M., Adhikari, S., and Sarkar, A. (2007). Linear system identification using proper orthogonal decomposition. *Mechanical Systems and Signal Processing*, 21(8):3123–3145.

[Kissas et al., 2020] Kissas, G., Yang, Y., Hwuang, E., Witschey, W., Detre, J., and Perdikaris, P. (2020). Machine learning in cardiovascular flows modeling: Predicting arterial blood pressure from non-invasive 4D flow MRI data using physics-informed neural networks. *Computer methods in applied mechanics and engineering*, 358:112623.

[Koeppl et al., 2018] Koeppl, T., Santin, G., Haasdonk, B., and Helmig, R. (2018). Numerical modelling of a peripheral arterial stenosis using dimensionally reduced models and kernel methods. *International journal for numerical methods in biomedical engineering*, 34(8):e3095.

[Kolmogorov, 1936] Kolmogorov, A. (1936). Über die beste annaäherung van funktionen einer gegebenen funkticnklasse. *Annals of Mathematics*, 37:107–110.

[Koltukluoğlu, 2019] Koltukluoğlu, T. S. (2019). Fourier spectral dynamic data assimilation: Interlacing cfd with 4D flow MRI. pages 741–749. Springer.

[Konstantinos et al., 2009] Konstantinos, C., Yiannis, S., Aaron, B., Elazer, R., Peter, H., and Charles, L. (2009). The role of low endothelial shear stress in the conversion of atherosclerotic lesions from stable to unstable plaque. *Current Opinion in Cardiology*, 24(6):580–590.

[Krittian et al., 2012] Krittian, S., Lamata, P., Michler, C., Nordsletten, D., Bock, J., Bradley, C., Pitcher, A., Kilner, P., Markl, M., and Smith, N. (2012). A finite-element approach to the direct computation of relative cardiovascular pressure from time-resolved MR velocity data. *Medical Image Analysis*, Elsevier(5):1029–1037.

[Lassila et al., 2012] Lassila, T., Malossi, A. C. I., Stevanella, M., Votta, E., Redaelli, A., and Deparis, S. (2012). Multiscale fluid-structure interaction simulation of patient-specific left ventricle fluid dynamics with fictitious elastic structure regularization.

[Lassila et al., 2011] Lassila, T., Manzoni, A., and Rozza, G. (2011). Reduction strategies for shape dependent inverse problems in haemodynamics. pages 397–406. Springer.

[Law et al., 2015] Law, K., Stuart, A., and Zygalakis, K. (2015). Data assimilation: A mathematical introduction. *Pre-print arXiv:1506.07825*.

[Le Dimet and Talagrand, 1986] Le Dimet, F.-X. and Talagrand, O. (1986). Variational algorithms for analysis and assimilation of meteorological observations: theoretical aspects. *Tellus A: Dynamic Meteorology and Oceanography*, 38(2):97–110.

[Ledoux et al., 1997] Ledoux, L. A., Brands, P. J., and Hoeks, A. P. (1997). Reduction of the clutter component in doppler ultrasound signals based on singular value decomposition: A simulation study. *Ultrasonic imaging*, 19(1):1–18.

[Leroux et al., 2013] Leroux, R., Chatellier, L., and David, L. (2013). Application of kalman filtering and partial least square regression to low order modeling of unsteady flows. Begel House Inc., TSFP Digital Library Online.

[Logg and Wells, 2010] Logg, A. and Wells, G. (2010). Dolfin: Automated finite element computing. *ACM transactions on Mathematical Software*, 37(2).

[Lorenc, 1981] Lorenc, A. (1981). A global three-dimensional multivariate statistical interpolation scheme. *Monthly Weather Review*, 109(4):701–721.

[Loseille and Feuillet, 2018] Loseille, A. and Feuillet, R. (2018). Vizir: High-order mesh and solution visualization using OpenGL 4.0 graphic pipeline. AIAA Aerospace Sciences Meeting, AIAA SciTech Forum.

[Maday and Mula, 2013] Maday, Y. and Mula, O. (2013). A Generalized Empirical Interpolation Method: application of reduced basis techniques to data assimilation. In Brezzi, F., Colli Franzone, P., Gianazza, U., and Gilardi, G., editors, *Analysis and Numerics of Partial Differential Equations*, volume 4 of *Springer INdAM Series*, pages 221–235. Springer Milan.

[Maday et al., 2015a] Maday, Y., Mula, O., Patera, A. T., and Yano, M. (2015a). 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.

[Maday et al., 2016] Maday, Y., Mula, O., and Turinici, G. (2016). Convergence analysis of the Generalized Empirical Interpolation Method. *SIAM journal on Numerical Analysis*, 54(3):1713–1731.

[Maday et al., 2015b] Maday, Y., Patera, A. T., Penn, J. D., and Yano, M. (2015b). A parameterized-background data-weak approach to variational data assimilation: formulation, analysis, and application to acoustics. *International journal for Numerical Methods in Engineering*, 102(5):933–965.

[Manzoni et al., 2018] Manzoni, A., Bonomi, D., and Quarteroni, A. (2018). *Reduced Order Modeling for Cardiac Electrophysiology and Mechanics: New Methodologies, Challenges and Perspectives*. Mathematical and Numerical Modeling of the Cardiovascular System and Applications.

[Mates et al., 1978] Mates, R., Gupta, R., Bell, A., and Klocke, F. (1978). Fluid dynamics of coronary artery stenosis. *Circulation research*, 42(1):152–162.

[Mehregan et al., 2014] Mehregan, F., Tournoux, F., Muth, S., Pibarot, P., Rieu, R., Cloutier, G., and Garcia, D. (2014). Doppler vortography: A color doppler approach to quantification of intraventricular blood flow vortices. *Ultrasound in medicine & biology*, 40(1):210–221.

[Moireau, 2008] Moireau, P. (2008). Assimilation de données par filtrage pour les systèmes hyperboliques du second ordre - Applications à la mécanique cardiaque. Ecole polytechnique x.

[Moireau et al., 2013] Moireau, P., Bertoglio, C., Xiao, N., Figueroa, C. A., Taylor, C., Chapelle, D., and Gerbeau, J.-F. (2013). Sequential identification of boundary support parameters in a fluid-structure vascular model using patient image data. *Biomechanics and modeling in mechanobiology*, 12(3):475–496.

[Montalba et al., 2018] Montalba, C., Urbina, J., Sotelo, J., Andia, M., Tejos, C., Irarrazaval, P., Hurtado, D., Valverde, I., and Uribe, S. (2018). Variability of 4D flow parameters when subjected to changes in MRI acquisition parameters using a realistic thoracic aortic phantom. *Magnetic Resonance in Medicine*, 79:1882–1892.

[Müller et al., 2018] Müller, L. O., Caiazzo, A., and Blanco, P. J. (2018). Reduced-order unscented kalman filter with observations in the frequency domain: application to computational hemodynamics. *IEEE transactions on Biomedical Engineering*, 66(5):1269–1276.

[Ohtsuki and Tanaka, 2006] Ohtsuki, S. and Tanaka, M. (2006). The flow velocity distribution from the doppler information on a plane in three-dimensional flow. *Journal of visualization*, 9(1):69–82.

[Pant et al., 2017] Pant, S., Corsini, C., Baker, C., Hsia, T.-Y., Pennati, G., and Vignon-Clementel, I. E. (2017). Inverse problems in reduced order models of cardiovascular haemodynamics: aspects of data assimilation and heart rate variability. *journal of The Royal Society Interface*, 14(126):20160513.

[Pears et al., 2012] Pears, N., Liu, Y., and Bounting, P. (2012). *3D Imaging, Analysis and Applications*. Springer.

[Peherstorfer et al., 2014] Peherstorfer, B., Butnaru, D., Willcox, K., and Bungartz, H.-J. (2014). Localized discrete empirical interpolation method. *SIAM journal on Scientific Computing*, 36(1):A168–A192.

[Perdikaris and Karniadakis, 2016] Perdikaris, P. and Karniadakis, G. E. (2016). Model inversion via multi-fidelity bayesian optimization: a new paradigm for parameter estimation in haemodynamics, and beyond. *Journal of The Royal Society Interface*, 13(118):20151107.

[Raiola et al., 2015] Raiola, M., Discetti, S., and Ianiro, A. (2015). On PIV random error minimization with optimal POD-based low-order reconstruction. *Experiments in Fluids*, 56(4):75.

[Reneman et al., 2006] Reneman, R., Arts, T., and Hoeks, A. (2006). Wall shear stress, an important determinant of endothelial cell function and structure, in the arterial system in vivo. *Journal of vascular research*, 43(3):251–269.

[Rognes et al., 2009] Rognes, E., Kirby, R., and Logg, A. (2009). Efficient assembly of H(div) and H(curl) conforming finite elements. *SIAM Journal on Scientific Computing*, 1(6):4130–4151.

[Rozza et al., 2007] Rozza, G., Huynh, D. B. P., and Patera, A. T. (2007). Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Archives of Computational Methods in Engineering*, 15(3).

[Saad and Schultz, 1986] Saad, Y. and Schultz, M. (1986). GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. ScI. STAT. COMPUT.*, 7(3):856–869.

[Saeed et al., 2018] Saeed, N., Nam, H., Ul, M., and Saqib, D. (2018). A survey on multidimensional scaling. *ACM computing Surveys*, 53(3).

[Shojima et al., 2004] Shojima, M., Oshima, M., Takagi, K., Torii, R., Hayakawa, M., Katada, K., Morita, A., and Kirino, T. (2004). Magnitude and role of wall shear stress on cerebral aneurysm: computational fluid dynamic study of 20 middle cerebral artery aneurysms. *Stroke*, 35(11):2500–2505.

[Sirovich, 1987] Sirovich, L. (1987). Turbulence and the dynamics of coherent structures. i. coherent structures. *Quarterly of applied mathematics*, 45(3):561–571.

[Sotelo et al., 2018] Sotelo, J., Urbina, J., Mura, J., Tejos, C., Irarrazaval, P., Andia, M., Hurtado, D., and Uribe, S. (2018). Three-dimensional quantification of vorticity and helicity from 3D cine PC-MRI using finite-element interpolations. *Magnetic resonance in medicine*, 79(1):541–553.

[Stuart, 2010] Stuart, A. M. (2010). Inverse problems: a Bayesian perspective. *Acta Numerica*, 19:451–559.

[Taddei, 2017] Taddei, T. (2017). An adaptive parametrized-background data-weak approach to variational data assimilation. *ESAIM: Mathematical Modelling and Numerical Analysis*, 51(5):1827–1858.

[Taddei, 2020] Taddei, T. (2020). A registration method for model order reduction: Data compression and geometry reduction. *SIAM journal of Scientific Computing*, 42(2):A997–A1027.

[Tikhonov, 1943] Tikhonov, A. (1943). On the stability of inverse problems. *Doklady Akademii Nauk SSSR*, 5:195–198.

[Trefethen and Bau, 1997] Trefethen, L. and Bau, D. (1997). *Numerical Linear Algebra*. SIAM.

[Turner et al., 1956] Turner, M. J., Clough, R. W., Martin, H. C., and Topp, L. J. (1956). Stiffness and deflection analysis of complex structures. *Journal of the Aeronautical Sciences*, 23(9):805–854.

[Uejima et al., 2010] Uejima, T., Koike, A., Sawada, H., Aizawa, T., Ohtsuki, S., Tanaka, M., Furukawa, T., and Fraser, A. G. (2010). A new echocardiographic method for identifying vortex flow in the left ventricle: numerical validation. *Ultrasound in medicine & biology*, 36(5):772–788.

[Vaillant et al., 2004] Vaillant, M., Miller, M., Younes, L., and Trouve, A. (2004). Statistics on diffeomorphisms via tangent space representations. *NeuroImage*, 23(1):161–169.

[Švihlová et al., 2016] Švihlová, H., Hron, J., Málek, J., Rajagopal, K., and Rajagopal, K. (2016). Determination of pressure data from velocity data with a view toward its application in cardiovascular mechanics. Part 1. Theoretical considerations. *International Journal of Engineering Science*, 105:108–127.

[Wahba, 1990] Wahba, G. (1990). *Spline models for observational data*, volume SIAM.

[Ye and Lim, 2016] Ye, K. and Lim, L. H. (2016). Schubert varieties and distances between subspaces of different dimensions. *SIAM journal on Matrix Analysis and Applications*, 37(3):1176–1197.

[Yu and Seinfeld, 1973] Yu, T. K. and Seinfeld, J. H. (1973). Observability and optimal measurement location in linear distributed parameter systems. *International journal of control*, 18(4):785–799.

[Zarins et al., 1983] Zarins, C., Giddens, D., Bharadvaj, B., Sottiurai, V., Mabon, R., and S., G. (1983). Carotid bifurcation atherosclerosis. *Circulation Research*, 53(4):502–514.