UNIVERSITÀ
DEGLI STUDI
DI PADOVA

Sede Amministrativa: Università degli Studi di Padova

Dipartimento di Matematica

Scuola di dottorato di ricerca in Scienze Matematiche
Indirizzo: Matematica Computazionale
Ciclo XXIV

Computational Parabolic Inverse Problems

Direttore della Scuola: Ch.mo Prof. Paolo Dai Pra

Coordinatore d'indirizzo: Ch.mo Prof. Paolo Dai Pra

Supervisore: Ch.mo Prof. Fabio Marcuzzi

Dottoranda: Giulia Deolmi

Some mathematician has said pleasure lies not in discovering truth,
but in seeking it.

(L. N. Tolstoy)

Abstract

This thesis presents a general approach to solve numerically parabolic Inverse Problems, whose underlying mathematical model is discretized using the Finite Element method. The proposed solution is based upon an adaptive parametrization and it is applied specifically to a geometric conduction inverse problem of corrosion estimation and to a boundary convection inverse problem of pollution rate estimation, as explained below.

In Part I the convection-diffusion-reaction equation and the heat equation are presented. These models are used to describe both a pollutant being transported along a stream and the temperature field of a material. In chapter 3 convection-dominated problems are analyzed: it is well known that this kind of equations need special care when discretized by using the Finite Element method. In section 3.6 a novel discretization strategy, originally presented in (59), is analyzed and substantially improved: the so called *Best Approximation Weighted Residuals (BAWR)* method (20). It is formulated in a Petrov-Galerkin context: the corresponding weighting function space is built such that the BAWR solution is optimal in the L^2 norm. It is demonstrated that it performs substantially better, compared to the Galerkin method. Moreover, it is a parameter-free method and, using a localization technique for the weighting functions, it is also computationally efficient.

Since in general problems from realistic situations have a large amount of degrees of freedom and consequently a high computational cost, in Part II, Model Order Reduction is considered. A largely used approach, adopted to solve also problems describing complicated dynamics, is the Proper Orthogonal Decomposition (POD). This method is highly problem dependent,

since it is based upon the Singular Value Decomposition of a matrix of trajectories. In section 6.6 the POD reduction of Navier Stokes equations is studied.

Finally Part III is about computational parabolic Inverse Problems. In particular the attention is focused on two problems, which are both solved estimating a vector of parameters, using a particular Gauss Newton approach. The first one consists in corrosion estimation: it is solved using an Infrared Thermographic Inspection. More precisely, supposing to know the field of temperature on a face of the material, the profile of corrosion of the unknown opposite face is estimated using a novel *Predictor-Corrector* strategy, originally presented in (155) and here substantially improved (138). The second problem consists in a pollution rate estimation: the pollutant is released in a fluid through a part of the boundary and its concentration is modeled by the convection-diffusion-reaction equation. Supposing to know the concentration at the outflow, the inverse problem consists both in localizing the part of the boundary where immision occurs and in estimating its intensity. It is solved with a novel algorithm, which uses both an *adaptive parametrization* and *time localization* (139). To solve the problem also POD reduction is studied.

Abstract

In questa tesi viene presentato un approccio numerico volto alla risoluzione di problemi inversi parabolici, basato sull'utilizzo di una parametrizzazione adattativa. Come descritto in seguito, l'algoritmo risolutivo viene descritto per due specifici problemi: mentre il primo consiste nella stima della corrosione di una faccia incognita del dominio, il secondo ha come scopo la quantificazione di inquinante immesso in un fiume.

La parte I della tesi si apre con la presentazione sia dell'equazione di convezione - diffusione - reazione sia dell'equazione del calore: entrambe sono usate nel seguito per descrivere rispettivamente un inquinante trasportato dalla corrente e la temperatura di un materiale. Nel capitolo 3 invece vengono analizzati problemi dominati dalla convezione: è risaputo che questo tipo di equazioni necessitano di particolare attenzione quando sono discretizzate usando il metodo degli Elementi Finiti. Nella sezione 3.6 una nuova strategia, originariamente presentata in (59), viene analizzata e perfezionata: essa è il metodo *Best Approximation Weighted Residuals (BAWR)* (20). In un contesto di tipo Petrov-Galerkin, lo spazio di funzioni peso è costruito in modo che la soluzione BAWR sia ottimale nella norma L^2. E' dimostrato che questo metodo dà risultati sostanzialmente migliori rispetto a quello di Galerkin. Inoltre non è parametrico e usando una tecnica di localizzazione delle funzioni di peso, è anche computazionalmente efficiente.

Poichè in genere la discretizzazione numerica di problemi che descrivono situazioni reali hanno molti gradi di libertà, e quindi elevato costo computazionale, nella parte II della tesi viene considerata la Riduzione di Modello. Un metodo largamente usato, anche per problemi che descrivono dinamiche complesse, è la Proper Orthogonal Decomposition (POD). Tuttavia essa è strettamente legato al problema, poichè si basa sulla decompo-

sizione ai valori singolari di una matrice di traiettorie. Nella sezione 6.6 è considerata la riduzione POD delle equazioni di Navier Stokes.

Per concludere, la parte III della tesi riguarda problemi inversi computazionali parabolici. Più nel dettaglio, due particolari problemi vengono analizzati e risolti stimando un vettore di parametri, mediante un approccio di tipo Gauss Newton. Il primo consiste nella stima della corrosione. Esso è risolto usando un test termografico all'infrarosso: supponendo di conoscere il campo di temperatura corrispondente ad una faccia del dominio, viene stimato il profilo di corrosione che descrive la faccia opposta, che è supposta incognita. L'algoritmo risolutivo consiste nell'utilizzo di una strategia *Predictor-Corrector*, originariamente presentata in (155) e qui perfezionata (138). Il secondo problema consiste nella stima della quantità di inquinante rilasciato in un fiume attraverso una parte del dominio. La sua concentrazione è modellata usando l'equazione di convezione-diffusione-reazione. Supponendo che essa sia nota all'outflow, il problema inverso consiste sia nel localizzare la parte del bordo del dominio dove avviene l'immisione sia nella stima della quantità di inquinante rilasciata. L'algoritmo risolutivo utilizza sia una parametrizzazione adattativa sia la localizzazione nel tempo (139). Nell'analisi del problema è studiata anche la riduzione POD.

To my family

Acknowledgements

I can no other answer make, but, thanks, and thanks.

(W. Shakespeare)

Here we are, at the end of this adventure: it's time to say "Thank you".

I thank my advisor, Prof. Fabio Marcuzzi, for encouraging me to apply for the doctorate school, for guiding me during these years and for giving me the freedom to choose what to study. I thank him for all the discussions and for all the possibilities that I have had during these years.

I would like to thank also Prof. Caterina Calgaro, for welcoming me in Lille, at INRIA and at the University of Lille1, for teaching me what she knows about Navier Stokes discretization and for the time we spent together studying the Proper Orthogonal Decomposition. A special thank also for letting me taste the real Belgian chocolate!

I thank also Prof. Dai Pra, director of the doctorate school, and all the school Council for giving me all the opportunity that I have had.

A special thank goes to Prof. Paola Mannucci, for the time she spent to help me with the proofs of Lemmas 8.4.1 and 9.5.1.

I thank also Prof. Massimo Fornasier for all his advises and Dr. Sergio Marinetti for proposing us the problem of corrosion estimation, for interesting discussions, and for the possibility to cooperate with the CNR (National Research Council).

A special thank goes also to Dr. Silvia Poles for the possibility to cooperate with Enginsoft Spa, solving the problem of corrosion detection. But above all, I thank her for her friendship, for her advices, for her time when I need help, for encouraging me and for the enjoyable time we spent together at conferences!

I would like to thank also all my friends and workmates at the department, with whom I shared this experience and my lunches. I expecially thank Paola, Gabriella, Alice and Cecilia, Francesca, Paolo, Daniele, Stefano, Giulio, Silvia, Valentina, Gabriele, Giovanni, Marco, Mirco, Enrico, Vittorio, Federico, Marco.

I thank also all old and new friends who supported me during these years. A special thank goes to Annalisa, with whom I shared my French adventure.

A very special thank goes to my family: my grandmother Elisabetta and my aunts Teresa and Linda. I thank also my godmother Marie Louise for her advices and for our trips to Paris! And last but not least, I express all my gratitude to my parents, Marica and Augusto, for encouraging me when I was down, for all the opportunities they gave me and for being always present in my life. I dedicate this work to them.

Enjoy the reading!

Contents

1

Introduction

The beauty of mathematical analysis is the elegant description of physical phenomena through the definition of rigorous mathematical models that summarize their essential aspects using partial differential equations (PDE's). Given a model problem, first of all it is important to study its well-posedness: suitable functional spaces are chosen to prove the existence of a solution and, if possible, its uniqueness.

Usually it is difficult to find an explicit analytical solution of the problem: this is the reason why *numerical techniques* are so important: they could be adopted to approximate the infinite dimensional solution space, using ad hoc algorithms, like Finite Differences (FD), Spectral Methods (SM), Finite Elements (FE) and Finite Volumes (FV).

This thesis focuses on *parabolic mathematical models* coming from applications and *discretized using FE*. These kind of models are the starting point to describe more complicate ones, called *Inverse Problems* (IP), which in general are not well-posed and must be solved adopting regularization techniques. In this thesis a solution approach is presented, based upon the Gauss Newton method, to solve both a geometric conduction inverse problem of corrosion estimation and a boundary convection inverse problem of pollution rate estimation.

Since in general problems from realistic situations have a large amount of degrees of freedom and consequently a high computational cost, also *Model Order Reduction* (MOR) techniques are studied.

1.1 Thesis outline

In more detail, Part I of this thesis introduces the parabolic models which are considered in the following parts. First of all in chapter 9 the *convection-diffusion-reaction* equation is analyzed. It may model both the *concentration of a pollutant* being transported by a fluid and the *temperature* of a material, as a particular case. In chapter 3 convection-dominated problems are analyzed: since the FE method could present spurious oscillations, stabilization methods are presented. Moreover a novel discretization strategy, originally presented in (59), is analyzed and substantially improved: the so called *Best Approximation Weighted Residuals (BAWR)* method (20). In chapter 4 *incompressible Navier-Stokes* equations are presented: they model the motion of an incompressible viscous flow. The well-posedness of the problem is discussed and its FE discretization is briefly described.

The second part of this thesis presents MOR techniques: discretizing PDE's models describing real problems means solving high dimensional algebraic systems. MOR techniques tries to reduce their dimensions, keeping as much information as possible. A largely used approach is to project the original system on a suitable subspace, the choice of which characterizes different reduction methods. In chapter 5 a general overview of linear and nonlinear strategies is presented. In this thesis we will focus on *Proper Orthogonal Decomposition* (POD), based upon a Singular Value Decomposition (SVD) of a matrix of trajectories of the unreduced model: this strategy is presented in chapter 6.Then the reduction of Navier Stokes equations is studied.

Finally, the third part is about *parabolic inverse problems*, which can be described as situations where the answer is known, but not the question, or where the results, or consequences are known, but not the cause. A general introduction to these kind of problems is given in Chapter 7. In the following both a geometric conduction inverse problem of corrosion estimation and a boundary convection inverse problem of pollution rate estimation will be presented: the solution strategy consists in estimating a vector of parameters using a Gauss Newton algorithm.

The corrosion estimation problem is described in chapter 8. It is based upon an Infrared Thermographic Inspection: the known surface is heated with a flash, and ex-

perimental temperatures are collected through a thermographic camera to reconstruct the eventual corrosion on the opposite unknown surface, as drawn in figure 1.1. This

Figure 1.1: *Infrared thermographic inspection: the known surface is heated with a flash, and experimental temperatures are collected through a thermographic camera.*

inverse problem is based upon the heat equation and it is solved using a novel *Predictor-Corrector* strategy, originally presented in (155) and here substantially improved (138).

The problem of pollution rate estimation introduced in chapter 9 is described by the convection-diffusion-reaction model: given the concentration at the outflow, the problem consists both in localizing the part of the boundary where immision occurs and in estimating it, as depicted in figure 1.2. To solve this inverse problem a novel algorithm

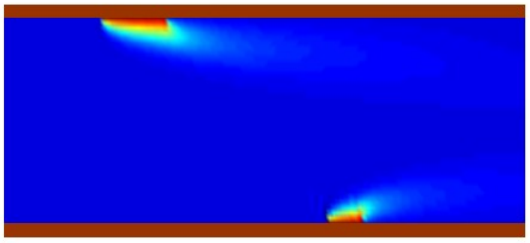

Figure 1.2: *Given the concentration at the outflow, localize the immision boundary and quantify the concentration put in.*

is formulated, considering both *adaptive parametrization* and *time localization* (139).

The model can be generalized describing the velocity field of the fluid with Navier-Stokes equations, as briefly described in chapter 9. For the pollution problem also POD reduction is considered, to decrease its computational cost.

1.2 Thesis contributions

In this thesis a new stabilization method for convection dominated problems, the BAWR strategy, originally presented in (59), is analyzed and substantially improved (20).

The corrosion estimation problem is solved with a novel strategy, the *Predictor-Corrector* method, originally presented in (155) and here substantially improved (138). Moreover also for the pollution rate estimation problem a novel algorithm is presented, based upon both the adaptive parametrization and time localization (139). To solve this inverse problem also the POD reduction is studied.

Part I

Parabolic models

[The universe] cannot be read until we have learnt the language and become familiar with the characters in which it is written. It is written in mathematical language, and the letters are triangles, circles and other geometrical figures, without which means it is humanly impossible to comprehend a single word.

(G. Galilei)

In this part, the following parabolic problems are introduced: the *convection-diffusion-reaction* equation, the *heat* equation and the *Navier-Stokes* equation. The mathematical problem is described and its Finite Element discretization is presented. Moreover convection dominated problems are considered and a novel stabilization technique, originally presented in (59), is analyzed and substantially improved: the so called *Best Approximation Weighted Residual (BAWR)* method (20).

These models will be used in the following parts of this thesis.

2

Convection Diffusion Reaction equation

2.1 Convection - Diffusion - Reaction equation

Let Ω be an open, limited and lipschitz continuous boundary subset of $\mathbb{R}^n$, $n \geq 1$. Assume that it is sufficiently regular.

Consider the following *convection - diffusion - reaction* equation

$$\frac{\partial \Theta}{\partial t} - div(k\nabla\Theta) + \mathbf{u} \cdot \nabla\Theta + \sigma\Theta = f \qquad . \tag{2.1}$$

As described in (23), the unknown function Θ may represent the *concentration of a pollutant* being transported along a stream, moving at velocity $\mathbf{u}$, subject to diffusive effects ($k > 0$): σ models its production ($\sigma > 0$) or destruction ($\sigma < 0$) by chemical reaction, while f describes fixed sources or sinks.

Alternatively, (2.1) may model the *temperature* Θ of a material $\mathcal{M}$, moving with velocity $\mathbf{u}$: the so called *convective heat equation*. In this case, $\mathbf{u}$ represents the velocity

of $\mathcal{M}$ and $\mathbf{f}$ external sources of heat. Moreover σ is supposed to be zero. This equation can be derived from the conservation of energy. A particular case is the *heat equation*

$$\frac{\partial}{\partial t}\Theta - k\Delta\Theta = 0. \tag{2.2}$$

In this thesis, while in the pollutant case we will rename Θ with c (cfr. chapter 9), which stands for *concentration*, when Θ represents a temperature it will be denoted by T (cfr. chapter 8). Since in this chapter we are not considering any particular situation, we will denote the unknown variable with Θ.

2.2 Variational formulation and Finite Element discretization

In this section we will briefly discuss the wellposedness of the general problem (2.1).

Let V and W be two Hilbert spaces on Ω, respectively, the *trial* (or *solution*) space and the *weighting* (or *test*) space. We denote with $\partial\Omega$ the boundary of Ω, $\partial\Omega = \Gamma_d \cup \Gamma_n$, $\Gamma_d \cap \Gamma_n = \emptyset$, and with $\mathbf{n}$ the outward normal to $\partial\Omega$. We also denote with $\frac{\partial u}{\partial n} = \nabla u \cdot \mathbf{n}$ the *conormal derivative* of Θ.

Let $\mathcal{L}$ be a *linear elliptic n-dimensional differential operator* on Ω

$$\mathcal{L}\Theta := -div(k\nabla\Theta) + \mathbf{u}\cdot\nabla\Theta + \sigma\Theta, \qquad \Theta \in V \qquad . \tag{2.3}$$

In the following we will denote with $(\cdot,\cdot)$ the inner product in $L^2(\Omega)$. Suppose that $k \in L^\infty(\Omega)$, $k(x) \geq k_0 > 0\ \forall x \in \Omega$, $\sigma \in L^\infty(\Omega)$, $\sigma(x) \geq 0$ a.e. in Ω, $\mathbf{u} \in [L^\infty(\Omega)]^n$, $div(\mathbf{u}) \in L^2(\Omega)$.

2.2.1 Stationary case

Let us consider the class of *linear boundary-value problems*: find $\Theta \in X \subset V$ such that

$$\begin{cases} \mathcal{L}\Theta &= f & in \quad \Omega \\ \Theta &= \Theta_d & on \quad \Gamma_d \\ \frac{\partial\Theta}{\partial n} &= \Theta_n & on \quad \Gamma_n \end{cases} \tag{2.4}$$

where $f \in L^2(\Omega)$, X is a suitable functional space and $\Theta_d \in H^{\frac{1}{2}}(\Gamma_d)$ and $\Theta_n \in L^2(\Gamma_n)$ are assigned functions.

2.2 Variational formulation and Finite Element discretization

If $\Gamma_d = \emptyset$, $\mathbf{u} = \mathbf{0}$ and $\sigma = 0$, the following compatibility condition must be satisfied: $\int_\Omega f d\omega = -\int_{\Gamma_n} \Theta_n d\gamma$ and uniqueness is possible if $\sigma(x) > 0$ (64).

Let us now derive the weak (variational) formulation of the differential problem (2.4) for $\mathcal{L}$ defined in (2.3).

Since $\int_\Omega -div(k\nabla\Theta)wd\omega = \int_\Omega -div(k\nabla\Theta w)d\omega + \int_\Omega k\nabla\Theta\nabla w d\omega = (\textit{Divergence Theorem}) = \int_{\partial\Omega} -k\frac{\partial\Theta}{\partial n}wd\gamma + \int_\Omega k\nabla\Theta\nabla w d\omega$, it follows that:

$$(\mathcal{L}\Theta, w) = \int_{\partial\Omega} -k\frac{\partial\Theta}{\partial n}wd\gamma + \int_\Omega k\nabla\Theta\nabla w d\omega + \int_\Omega \mathbf{u}\cdot\nabla\Theta w d\omega + \int_\Omega \sigma\Theta w d\omega = (f, w). \quad (2.5)$$

So the weak problem is well posed, that is the integrals are defined, if we consider V and W subspaces of $H^1(\Omega)$, such that they contain $H_0^1(\Omega)$ as a subspace, and we interpret derivatives in a distributional sense (9, 64). In particular we choose $W \subseteq H^1_{\Gamma_d}(\Omega)$, i.e. we impose that $w = 0$ on Γ_d.

Now, by imposing the boundary conditions (2.4), we are ready to define the weak (variational) formulation of (2.4):

$$\text{find } \Theta \in V \text{ s.t. } a(\Theta, w) = F(w), \qquad \forall w \in W, \quad (2.6)$$

where $a(\cdot,\cdot)$ is a continuous bilinear form $a : V \times W \to \mathbb{R}$, defined as $a(\Theta, w) := \int_\Omega k\nabla\Theta\nabla w d\omega + \int_\Omega \mathbf{u}\cdot\nabla\Theta w d\omega + \int_\Omega \sigma\Theta w d\omega$, and $F(\cdot)$ is a continuous linear operator $F: W \to \mathbb{R}$, $F(w) := (f, w) + \int_{\Gamma_n} k\Theta_n w d\gamma$.

To obtain an approximate numerical solution, related to a *finite element* discretization of the domain Ω whose refinement level is characterized by a parameter h, identifying the computational domain Ω_h with Ω, we will choose also two N_h finite-dimensional families of Hilbert subspaces $\{V_h\}_{h>0}$, $V_h \subset V$ and $\{W_h\}_{h>0}$, $W_h \subset W$.

For simplicity we assume now that $\Gamma_n = \emptyset$, i.e. $W \subseteq H_0^1(\Omega)$, that is we impose that $w = 0$ on $\partial\Omega$.

Suppose moreover that $W = V = H_0^1(\Omega)$, and redefine

$$F(w) = (f, w) + \int_{\Gamma_n} k\Theta_n w d\gamma - a(G, w),$$

where G is the Dirichlet lift of a boundary data g, i.e. $G \in H^1(\Omega)$ and $G\,|_{\Gamma_d} = g$. It is possible to show (64) that, if $-\frac{1}{2}div(\mathbf{u}) + \sigma \geq 0$ *a.e. in* Ω, the bilinear form a is coercive with a constant $C = \frac{k_0}{1+C_\Omega^2}$, where C_Ω is the constant appearing in the Poincarè inequality ($\|w\|_{L^2(\Omega)} \leq C_\Omega \|\nabla w\|_{L^2(\Omega)}$). Moreover a is also continuous with constant $\gamma = \|k\|_{L^\infty(\Omega)} + \|\mathbf{u}\|_{L^\infty(\Omega)} + \|\sigma\|_{L^\infty(\Omega)}$. Then Lax Milgram Lemma (Theorem A.2.1) holds.

If we formulate the *Galerkin* approximation (A.3) of (2.6), applying Theorem A.3.1, the following estimate holds:

$$|\Theta - \Theta_h|_{H^1} \leq \frac{\gamma}{C} \inf_{v_h \in V_h} \|\Theta - v_h\|_{H^1} \qquad .$$

Observe now that $\frac{\gamma}{C} = \frac{\|k\|_{L^\infty(\Omega)} + \|\mathbf{u}\|_{L^\infty(\Omega)} + \|\sigma\|_{L^\infty(\Omega)}}{k_0} \left(1 + C_\Omega^2\right)$, i.e. if the convection coefficient $\mathbf{u}$ or the reaction one σ are much bigger than the diffusion one k_0, the constant in the estimate is big; thus it become useless and the approximation Θ_h could be unsatisfactory. It is possible to show that in this kind of *convection or reaction dominated problems* the Galerkin solution Θ_h presents spurious oscillations around the real one, also when the last one is monotone (64, 65). Observe that in this case the bilinear form a is *not* symmetric and then the Galerkin solution is not still the optimal one. This is the reason that motivates the study of *stabilization methods*, as presented in chapter 3.

Consider now a *Generalized Galerkin* approximation (A.6) of (2.6), supposing that a_h and F_h are approximants of a and F, s.t. $\sup_{w_h \in V_h, w_h \neq 0} \frac{|a(v_h, w_h) - a_h(v_h, w_h)|}{\|w_h\|}$ and $\sup_{w_h \in W_h, w_h \neq 0} \frac{|F(w_h) - F_h(w_h)|}{\|w_h\|}$ are small enough. Applying Theorem A.3.3 we deduce that the Galerkin convergence rate will be improved choosing a_h s.t. it is uniformly coercive and C^* is s.t. $\frac{1}{C^*}$ and $1 + \frac{\gamma}{C^*}$ are small. We will see examples of these methods in chapter 3.

Finally consider the *Petrov-Galerkin* formulation (A.4) of the problem (2.6), with $a_h = a$ and $F_h = F$, applying Theorem A.3.2 the following estimate holds:

$$|\Theta - \Theta_h|_V \leq \inf_{v_h \in V_h} \left(1 + \frac{\gamma}{C_h}\right) |\Theta - v_h|_V \,, \tag{2.7}$$

which still depends on the coefficients of $\mathcal{L}$ as the Galerkin method, and on W_h and V_h. Thus also a general Petrov Galerkin method could be *inaccurate*. Neverless it can be shown that this kind of methods can be used to *stabilize* the problem: in chapter 3 standard stabilization methods are presented, while in section 3.6 a novel one is analyzed, the so called *Best Approximation Weighted Residual (BAWR)* method.

2.2.2 Unstationary case

Until now we have discussed the steady problem: in this section we will introduce the time-dependence. A general parabolic problem has the following structure

$$\begin{cases} \frac{\partial \Theta}{\partial t} + \mathcal{L}\Theta = f, & in \quad Q_T := (0,T) \times \Omega \\ B\Theta = g & on \quad \Sigma_T := (0,T) \times \partial\Omega \\ \Theta\,|_{t=0} = \Theta_0, & on \quad \Omega, \end{cases} \tag{2.8}$$

where $f = f(t, \mathbf{x})$, $g = g(t, \mathbf{x})$, $\Theta_0 = \Theta_0(\mathbf{x})$ are known data and $B\Theta = g$ denotes boundary conditions (e.g. Dirichlet, Neumann, mixed, Robin).

As for the elliptic case, choose V a closed subspace of $H^1(\Omega)$ s.t. $H_0^1(\Omega) \subset V \subset H^1(\Omega)$, which depends on $\mathcal{L}$ and B.

Define the following Banach spaces (24)

$$L^2(0,T;V) := \left\{ \Theta : \ (0,T) \to V \text{ s.t. } \Theta \text{ is measurable and } \int_0^T \|\Theta(t)\|_V^2 \, dt < \infty \right\}$$

and

$$C^0([0,T];L^2(\Omega)) := \left\{ \Theta : \ [0,T] \to L^2(\Omega) \text{ s.t. } \Theta \text{ is measurable and } \int_0^T \|\Theta(t)\|_2^2 \, dt < \infty \right\},$$

where $\Theta(t) := \Theta(t, \cdot)$. Suppose that $B\Theta = g$ represents homogeneous boundary conditions and consider $V = H_0^1(\Omega)$: thus the weak formulation of (2.8) is the following: given $f \in L^2(Q_T)$ and $\Theta_0 \in L^2(\Omega)$, find $\Theta \in L^2(0,T;V) \cap C^0([0,T];L^2(\Omega))$ s.t.

$$\begin{cases} \frac{d}{dt}(\Theta(t), v) + a(\Theta(t), v) &= (f(t), v) \quad \forall v \in V \\ \Theta(0) &= \Theta_0 \qquad on \quad \Omega, \end{cases} \tag{2.9}$$

where a is a suitable bilinear form depending on $\mathcal{L}$.

Theorem 2.2.1 *(J.L. Lions) If a is continuous and coercive, then, given $f \in L^2(Q_T)$ and $\Theta_0 \in L^2(\Omega)$, there exists a unique $\Theta \in L^2(0,T;V) \cap C^0([0,T];L^2(\Omega))$ solution of (2.9). Moreover $\frac{\partial \Theta}{\partial t} \in L^2(0,T;V')$ and the following energy estimate holds:*

$$\max_{t \in [0,T]} \|\Theta(t)\|_0^2 + \alpha \int_0^T \|\Theta(t)\|_V^2 \, dt \leq \|\Theta_0\|_0^2 + \frac{1}{\alpha} \int_0^T \|f(t)\|_0^2 \, dt.$$

For a proof cfr. e.g. (24, 65).

2.2.3 Discretization of (2.9)

The idea is to approximate the solution of (2.9) with the *method of lines*, which consists in a first approximation in space applying the finite element method to (2.9) and then in the numerical solution of an ordinary differential equation whose solution $\Theta_h(t)$ is an approximation of the exact solution for each $t \in [0,T]$. Consider a family of subspaces of V, $\{V_h, \ h > 0\}$, $V_h = X_h^k \cap H_0^1(\Omega)$ (if we are not dealing with homogeneous Dirichlet boundary conditions we choose $V_h = X_h^k$) and denotes with $\Theta_{0,h} \in V_h$ a suitable approximation of $\Theta_0 \in L^2(\Omega)$. The semi-discrete approximate problem is the following: given $f \in L^2(Q_T)$ and $\Theta_{0,h} \in V_h$, for each $t \in [0,T]$ find $\Theta_h(t) \in V_h$ s.t.

$$\begin{cases} \frac{d}{dt}(\Theta_h(t), v_h) + a(\Theta_h(t), v_h) &= (f(t), v_h) \quad \forall v_h \in V_h, \ t \in (0,T) \\ \Theta_h(0) &= \Theta_{0,h} \qquad on \quad \Omega. \end{cases} \tag{2.10}$$

Remark 2.2.1 *Observe that writing* $\Theta_h(t) = \sum_j \alpha_j(t)\phi_j$, $\{\phi_j\}_{j=1,\dots,N_h}$ *basis of* V_h, *and* $\Theta_{0,h} = \sum_j \alpha_{0,j}\phi_j$, *we obtain the equivalent ordinary differential problem*

$$\begin{cases} M\frac{d}{dt}\boldsymbol{\alpha}(t) + A\boldsymbol{\alpha}(t) &= \boldsymbol{F}(t) \\ \boldsymbol{\alpha}(0) &= \boldsymbol{\alpha}_0, \end{cases} \tag{2.11}$$

where $M_{ij} = (\phi_i, \phi_j)$, $A_{ij} = a(\phi_j, \phi_i)$, $F_i(t) = (f(t), \phi_i)$, $i, j = 1, \dots, N_h$. *Since* M *is positive definite, there exists a unique solution* $\boldsymbol{\alpha}(t)$ *of the system.*

Consider now a uniform subdivision of $[0, T]$ of step Δt, whose nodes are

$$t_n := n\Delta t,\ n = 0, \dots, \left[\frac{T}{\Delta t}\right].$$

The idea is to construct a sequence $\Theta_h^n(\mathbf{x})$, discretizing the ordinary differential equation (2.11), to approximate the exact solution $\Theta(t_n, \mathbf{x})$ of (2.9).

Various methods can be used to solve it numerically, e.g. multi-step or Runge-Kutta methods (cfr. e.g. (66) for an introduction to these methods). Here we present only the $\theta-$*method*, which consists in defining a sequence $\{\Theta_h^n\}_{n=0,\dots,\left[\frac{T}{\Delta t}\right]}$ s.t.

$$\begin{cases} \frac{1}{\Delta t}(\Theta_h^{n+1} - \Theta_h^n, v_h) + a(\vartheta\Theta_h^{n+1} + (1-\vartheta)\Theta_h^n, v_h) &= (\vartheta f(t_{n+1}) + (1-\vartheta)f(t_n), v_h) \qquad \forall v_h \in V_h, \\ \Theta_h^0 &= \Theta_{0,h}, \end{cases} \tag{2.12}$$

for each n. When $\vartheta = 0$ or $\vartheta = 1$ this scheme is called *forward Euler* or *backward Euler* method respectively, for $\vartheta = \frac{1}{2}$ it is called *Crank-Nicolson* method. Observe that this method is absolutely stablefor $\vartheta \geq \frac{1}{2}$ (66).

Now we summarize the main results about stability and convergence of the (totally) discretized method, cfr. (24, 65) for proofs and more details.

Theorem 2.2.2 *(Stability) Assume that* a *is coercive and that* $t \to \|f(t)\|_0$ *is bounded in* $[0, T]$. *When* $0 \leq \vartheta < \frac{1}{2}$ *assume, moreover,*

$$\Delta t(1 + Ch^{-2}) < \frac{2\alpha}{(1-2\vartheta)\gamma^2}.$$

Then Θ_h^n *satisfies*

$$\|\Theta_h^n\|_0 \leq C_\vartheta \left(\|\Theta_{0,h}\|_0 + \max_{t\in[0,T]} \|f(t)\|_0\right),\ n = 0, \dots, \left[\frac{T}{\Delta t}\right],$$

where C_ϑ *is a non-decreasing function of* α^{-1}, γ *and* T.

Theorem 2.2.3 *(Convergence) Assume that a is coercive and that $\frac{\partial \Theta_h}{\partial t}(0) \in L^2(\Omega)$, $f \in L^2(Q_T)$, $\frac{\partial f}{\partial t} \in L^2(Q_T)$. When $0 \leq \vartheta < \frac{1}{2}$ assume, moreover,*

$$\Delta t(1 + Ch^{-2}) < \frac{2\alpha}{(1-2\vartheta)\gamma^2}.$$

Then

$$\|\Theta_h^n - \Theta_h(t_n)\|_0 \leq C_\vartheta \Delta t \left(\left\| \frac{\partial \Theta_h}{\partial t}(0) \right\|_0^2 + \int_0^T \left\| \frac{\partial f}{\partial t}(r) \right\|_0^2 dr \right)^{\frac{1}{2}}, \ n = 0, \ldots, \left[\frac{T}{\Delta t} \right],$$

where C_ϑ is a non-decreasing function of α^{-1}, γ and T.

If $\vartheta = \frac{1}{2}$, $\frac{\partial^2 f}{\partial t^2} \in L^2(Q_T)$ and $\frac{\partial^2 \Theta_h}{\partial t^2}(0) \in L^2(\Omega)$, then

$$\|\Theta_h^n - \Theta_h(t_n)\|_0 \leq C_\vartheta (\Delta t)^2 \left(\left\| \frac{\partial^2 \Theta_h}{\partial t^2}(0) \right\|_0^2 + \int_0^T \left\| \frac{\partial^2 f}{\partial t^2}(r) \right\|_0^2 dr \right)^{\frac{1}{2}}, \ n = 0, \ldots, \left[\frac{T}{\Delta t} \right].$$

3

Stabilization of convection dominated problems

3.1 Introduction

Consider problem (2.4): as mentioned e.g. in (54), if $\Gamma_n = \emptyset$, and convection dominates diffusion, i.e. when $\|\mathbf{u}\| >> k$, the solution Θ will vary rapidly in a layer of width $O(k)$ at the *outflow boundary* $\partial\Omega^+ = \{\mathbf{x} \in \partial\Omega \; s.t. \; \mathbf{n}(\mathbf{x}) \cdot \mathbf{u}(\mathbf{x}) \geq 0\}$. Thus a classical problem in numerical analysis is to construct a finite difference or finite element method for

solving this kind of problems, using a mesh with mesh length h skew to the streamline direction. It is required that the scheme is higher-order accurate *and* has good stability properties without requiring h to be smaller than k (e.g. classical monotone upwind schemes obtained by adding an artificial diffusion term, which are only first order accurate).

More precisely convection dominated problems need to be *stabilized* when discretized using the Finite Element Method (FEM), see e.g. (26, 48). As mentioned in (12),

> "Convection-diffusion operators have perplexed numerical analysts for decades. Historically they have been treated by methods which have either compromised stability (e.g. central differences) or accuracy (e.g. upwinding). In fact, one often hears in some circles of computational fluid dynamicists that stability and accuracy are in competition, and that one must be sacrificed to attain the other. Stabilized methods represent a refutation of this ancient religion.".

To *stabilize* means to counteract a priori the effect that the small-scale, under-resolved features of the solution would have on the discrete solution. For this reason, the stabilization techniques are recently called sub grid modeling, see e.g. (10, 34). The practical aim of the stabilizing techniques is to prevent the formation of spurious oscillations that usually appear in the Galerkin solution for convection-dominated problems.

As observed in (8), many alternative variational formulations have been proposed with the goal of recovering at least some advantages of the Galerkin formulation in a more general setting. There are mainly two classes of alternative variational formulations:

- for a given PDE problem, one may *modify* the variational principle with the goal of defining better quasi-projections in a Generalized Galerkin (A.6) or Petrov-Galerkin (A.4) context. One possible solution is to define stabilized methods, such that the modified bilinear form a_h is strongly coercive, or at least satisfy the discrete inf-sup condition of Theorem A.3.2 for *arbitrary* discrete conforming subspaces V_h and W_h. In this Chapter we analyze this kind of strategies which can be classified into two different families: residual and non-residual based.

- *replace* the variational formulation by an externally defined one based on minimizing the residuals of the PDE problem (*Least Squares FEM* (8)).

Residual-based methods belonging to the first class usually add a *stabilizing term* to the weak formulation of the boundary value problem, in a Generalized Galerkin context. Traditional methods are the Streamline Upwind Petrov Galerkin (SUPG) (48), also known as the Streamline Diffusion Finite Element Method (SDFEM), the Generalized Least Squares (GLS) method, the Douglas Wang (DW) method (22) and the residual-free bubbles (12) (cfr. e.g. (65) for a general overview of these methods). Common aspects of these methods are both a residual-based and a parameter dependent formulation. More recently sub grid scale modeling (SGS) has been introduced, trying to extend and generalize the previous ones (39). Although they are very effective, it is well known that often the parameter tuning process is problematic.

In this chapter we will use the steady diffusion-convection-reaction operator (2.3), introduced in section 2.2, as a model problem to study some standard stabilization methods, following (64, 65). Finally in section 3.6 we present a stable FEM approximation following the Petrov-Galerkin approach: the so called *Best-Approximation Weighted-Residuals (BAWR)*, introduced in (59). The weighting function space is built such that the corresponding BAWR solution is optimal in the L^2 norm. Since it is an approximation in a least-squares sense, it oscillates but without spurious oscillations (60). However, as it will be demonstrated, for convection dominated problems it performs substantially better, compared to the Galerkin method. Moreover, it is a *parameter-free* method and it will be demonstrated that, using a localization technique for the weighting functions, it is also computationally efficient. In the recent literature there is a renewed interest in optimal Petrov-Galerkin methods, e.g. the Nearly-Optimal Petrov Galerkin (NOPG) method (35), focusing on H^1-semi norm estimates.

3.2 Generalized Galerkin Methods

In this section we present stabilization strategies formulated as Generalized Galerkin methods (A.6), choosing

$$a_h(\Theta_h, w_h) = a(\Theta_h, w_h) + b_h(\Theta_h, w_h),$$

$$F_h(w_h) = F(w_h) + G_h(w_h).$$

The *stabilization terms* b_h and G_h are operators chosen to limit Galerkin's spurious oscillations.

3.2.1 Artificial Diffusion and Streamline diffusion methods

Consider the simpler diffusion-convection problem

$$\begin{cases} -k\Delta\Theta + \mathbf{u}\cdot\nabla\Theta &= f \quad in \quad \Omega \\ \Theta &= 0 \quad on \quad \partial\Omega \end{cases} \tag{3.1}$$

whose weak formulation is (2.6), with $a(\Theta, w) := \int_\Omega k\nabla\Theta\nabla w d\omega + \int_\Omega \mathbf{u}\cdot\nabla\Theta w d\omega$ and $F(w) := (f, w)$.

Then the *artificial diffusion method* consists in a Generalized Galerkin one, such that $b_h(\Theta_h, w_h) = Qh\int_\Omega \nabla\Theta_h\nabla w_h d\omega$, which corresponds to solve problem (3.1) with the Galerkin method, using $k+hQ$ instead of k, where $Q > 0$ is a parameter. Otherwise this approach introduces diffusion in all directions, and *not* only along the vectorial space generated by $\mathbf{u}$. We refer to (67) for other upwind methods for the diffusion-convection-reaction operator.

The *streamline diffusion method* consists in choosing $b_h(\Theta_h, w_h) = Qh\int_\Omega \mathbf{u}\cdot\nabla\Theta_h\ \mathbf{u}\cdot\nabla w_h d\omega$, which corresponds to add to the original problem (3.1) $-Qhdiv\left((\nabla\Theta\cdot\mathbf{u})\mathbf{u}\right)$, i.e. to introduce artificial diffusion only along the streamlines (40).

Observe that the error estimate for the artificial diffusion method can be obtained applying proposition A.3.1, using $V = H^1(\Omega)$, (65):

$$\|\Theta - \Theta_h\| \le c(C, \gamma, Q)\left(\inf_{w_h\in V_h}\|\Theta - w_h\| + h\|\Theta\|\right),$$

where $c(C, \gamma, Q) > 0$ is a constant depending on a and Q.

Figure 3.1 compares the artificial diffusion and the Galerkin methods.

Finally observe that, using definition A.3.1, streamline and artificial diffusion methods are only *consistent*, in fact

$$a_h(\Theta, w_h) - F(w_h) = a_h(\Theta, w_h) - a(\Theta, w_h) = \begin{cases} Qh(\nabla\Theta, \nabla w_h), & \text{artificial diffusion} \\ Qh(\nabla\Theta\cdot\mathbf{u}, \nabla w_h\cdot\mathbf{u}), & \text{streamline diffusion,} \end{cases} \tag{3.2}$$

thus $\tau_h(\Theta) = O(h)$. This means that they are accurate of order h, *no matter how large the degree of the finite element space is.*

3.2.2 Strongly Consistent Stabilized Finite Element methods

Consider the differential diffusion-convection-reaction operator $\mathcal{L}\Theta = f$ introduced in section 2.2, with homogeneous Dirichlet boundary conditions on Ω. Let $V = H_0^1(\Omega)$: we recall that its continuous weak formulation is

$$\text{find } \Theta \in V \text{ s.t. } a(\Theta, w) = F(w), \qquad \forall w \in V. \tag{3.3}$$

3. STABILIZATION OF CONVECTION DOMINATED PROBLEMS

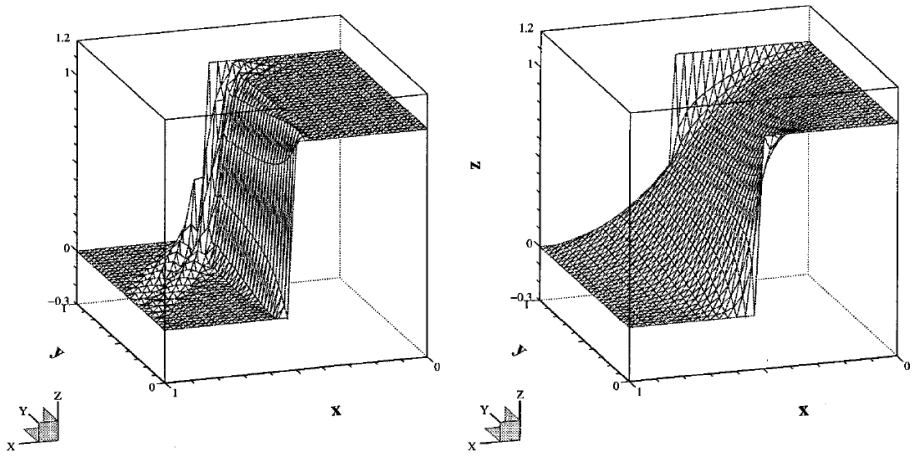

Figure 3.1: *This pictures represents the Galerkin (left) and the artificial diffusion (right) solutions for the two dimensional convection-dominated operator (3.1). They has been taken from (65). Artificial diffusion is more stable, but less accurate in the boundary layer.*

A *strongly consistent stabilization method* is a Generalized Galerkin one, and consists in adding to the standard Galerkin method variational terms that are *mesh-dependent, consistent* and *numerically stabilizing*: find $\Theta_h \in V_h$ s.t.

$$\begin{array}{rcll} a(\Theta_h, w_h) + \mathcal{L}_h(\Theta_h, f; w_h) & = & F(w_h), & \forall w_h \in V_h, \\ \mathcal{L}_h(\Theta, f; w_h) & = & 0, & \forall w_h \in V_h. \end{array} \tag{3.4}$$

A possible choice is

$$\begin{array}{rcl} \mathcal{L}_h(\Theta_h, f; w_h) & = & \mathcal{L}_h^\rho(\Theta_h, f; w_h) = \sum_{K \in \mathcal{T}_h} \tau_K (\mathcal{L}\Theta_h - f, S_K^{(\rho)}(w_h))_{L^2(K)}, \\ S_K^{(\rho)}(w_h) & = & \frac{h_K}{|\mathbf{u}|} \left(\mathcal{L}_S w_h + \rho \mathcal{L}_{SS} w_h \right), \end{array} \tag{3.5}$$

where $\mathcal{T}_h$ denotes a discretization of Ω with elements K whose diameter is h_K, τ_K are parameters which have the dimension of time and $\mathcal{L}_S$ and $\mathcal{L}_{SS}$ are respectively the symmetric and the skewsymmetric part of $\mathcal{L}$.

Define now

$$a_h(\Theta_h, w_h) := a(\Theta_h, w_h) + \sum_{K \in \mathcal{T}_h} \tau_K (\mathcal{L}\Theta_h, S_K^{(\rho)}(w_h))_{L^2(K)}$$

and

$$F_h(w_h) := F(w_h) + \sum_{K \in \mathcal{T}_h} \tau_K (f, S_K^{(\rho)}(w_h))_{L^2(K)},$$

then, if Θ is the real solution of (3.3), it follows that

$$a_h(\Theta, w_h) - F_h(w_h) = \mathcal{L}_h(\Theta, f; w_h) = \sum_{K \in \mathcal{T}_h} \tau_K (\mathcal{L}\Theta - f, S_K^{(\rho)}(w_h))_{L^2(K)} = 0 :$$

the trick consists in using the residual $\mathcal{L}\Theta_h - f$ which is zero in Θ. So these methods are *strongly consistent*.

Let us see some examples (ρ-methods):

- *Galerkin Least Squares (GLS)*: ($\rho = 1$, (47))

  $$S_K^{(1)}(w_h) = \frac{h_K}{|\mathbf{u}|} \mathcal{L} w_h;$$

  (Least Square control of the residual)

- *Streamline Upwind Petrov Galerkin (SUPG)*: ($\rho = 0$, (48, 49))

  $$S_K^{(0)}(w_h) = \frac{h_K}{|\mathbf{u}|} \mathcal{L}_{SS} w_h;$$

  (control of the convective part of the residual). It is also called *Streamline Diffusion Finite Element Method (SDFEM)*, (67). More details on SUPG can be found e.g. in (41, 42, 53, 62, 67, 72).

- *Douglas Wang (DW)*: ($\rho = -1$, it was introduced in (22) for the Stokes problem and generalized for the diffusion-convection operator in (31))

  $$S_K^{(-1)}(w_h) = -\frac{h_K}{|\mathbf{u}|} \mathcal{L}^* w_h.$$

  It is also called *unusual stabilized FEM* (USFEM), and can be interpreted as static condensation of bubbles added to V_h (27). For an error analysis cfr. (28).

In (31) it is underlined that DW presents nicer stability characteristics than GLS, employing high order interpolation. In fact, for the operator (2.3), if $\sigma = 0$ and V_h is the linear finite element space, then all the previous methods *coincide*: on every K the laplacian term is zero. The superiority of DW for the Stokes problem, can be quantified: in fact GLS has a more restrictive condition on τ to ensure stability than DW (31).

Although these are Generalized Galerkin methods, as explained in (65) proposition A.3.1 cannot be applied because the corresponding bilinear form a_h does not satisfy the

continuity requirement. Moreover results of Theorem A.3.3 do not imply convergence for them. So these methods need an ad hoc analysis.

In particular, consider $\eta > 0$ such that $\eta \leq -\frac{1}{2} div\mathbf{u} + \sigma$ and define the following norm:

$$\|v\|_{(\rho)} = \left[k \|\nabla v\|^2_{L^2(\Omega)} + \|\eta v\|^2_{L^2(\Omega)} + \sum_K \tau_K ((\mathcal{L}_{SS} + \rho \mathcal{L}_S)v, S_K^{(\rho)}(v))_{L^2(K)} \right]^{\frac{1}{2}}.$$

Observe that for large Peclet numbers, $Pe = \frac{\mathbf{u}L}{2k}$, where L is a characteristic length of the domain, the character of the solution Θ is dominated by its behavior along the streamlines. Therefore, this norm, in which the streamline derivative $\mathbf{u} \cdot \nabla v$ plays an important rule, through $\sum_K \tau_K ((\mathcal{L}_{SS} + \rho \mathcal{L}_S)v, S_K^{(\rho)}(v))_{L^2(K)}$, is a more meaningful measure than using only $\|\nabla v\|^2_{L^2(\Omega)}$ (23).

Then there exist $C^* > 0$ and $c > 0$ such that the following stability and convergence estimates can be proven (64, 65):

$$\begin{array}{rcl} \|\Theta_h\|_{(\rho)} & \leq & \frac{\gamma}{C^*} \|f\|_{L^2(\Omega)} \\ \|\Theta - \Theta_h\|_{(\rho)} & \leq & \tilde{c} h^{r+\frac{1}{2}} |\Theta|_{H^{r+1}(\Omega)}, \end{array} \tag{3.6}$$

where $\tilde{c}$ and r are constants independent of h. While the Galerkin method has order of convergence h^{r+1}, these methods behave a little bit worse with respect to h ($O(h^{r+\frac{1}{2}})$), but they prevent spurious oscillation, i.e. $\tilde{c}$ will be smaller than the Galerkin one. It is important to note also that an higher order of polynomial approximation corresponds to a better approximation (unlike only consistent methods). As an example cfr. figure 3.2.

It is interesting to note (54) that for SUPG,

$$\|v\|_{(0)} = \left[k \|\nabla v\|^2_{L^2(\Omega)} + \|\eta v\|^2_{L^2(\Omega)} + \sum_K \frac{\tau_K h_K}{|\mathbf{u}|} \|\mathbf{u} \cdot \nabla v\|^2_{L^2(K)} \right]^{\frac{1}{2}},$$

and $\sum_K \frac{\tau_K h_K}{|\mathbf{u}|} \|\mathbf{u} \cdot \nabla v\|^2_{L^2(K)}$ means that the streamline diffusion method has an improved stability for the *streamline derivative* $\mathbf{u} \cdot \nabla$, as compared to the standard Galerkin method.

3.2.3 The choice of τ

A problematic issue using these methods is the quantification of the parameter τ_K, which measures the amount of artificial viscosity introduced in the formulation. As mentioned in (67), the optimal choice of τ_K for ρ-methods is still an open question: only in some special cases they can be derived from problem data.

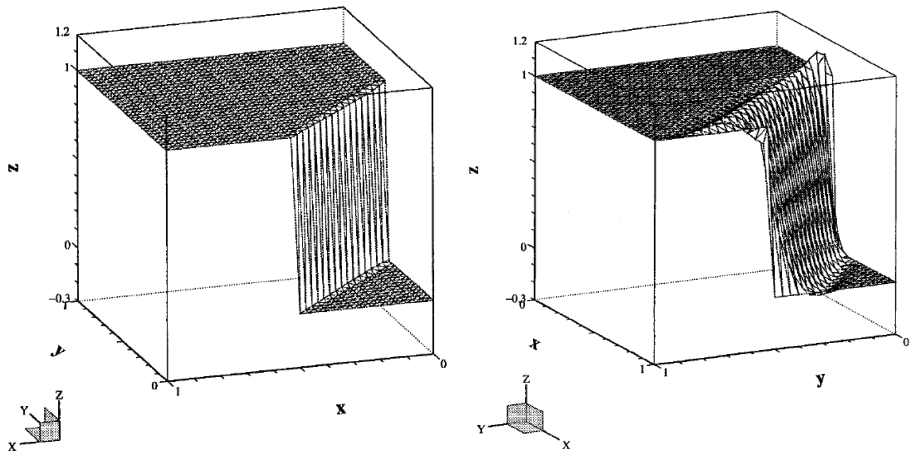

Figure 3.2: *These pictures represent the GLS solution when* $\boldsymbol{u}$ *is parallel to the discontinuity line (left) and when is not (right). In the latter case the scheme is diffusive and there are some under- and over-shootings, which is a prove of the non monotonicity of the scheme (they can be avoided introducing a shock-capturing non linear viscosity term, (42)). Figures has been taken form (65).*

As a special case, consider $n = 1$, $\sigma = 0$ and k and u constant. Suppose moreover that $\Omega = [0, 1]$, $\Gamma_n = \emptyset$ and that homogeneous Dirichlet boundary conditions are imposed. If the partition of Ω is uniform with step h, then it is possible to show that, using linear finite elements, the SUPG solution is *nodally exact* (viz. *superconvergence*) if $\tau = \frac{\alpha h}{2u}$, with the *upwind function* $\alpha = \alpha(Pe^{loc}) = coth(Pe^{loc}) - \frac{1}{Pe^{loc}}$ (16). This analysis can be extended to a one dimensional convection-diffusion-reaction operator, as explained in (35) for GLS.

A more general expression for τ_K in SUPG, which accounts both $n > 1$ and nonlinear finite element approximations is the following (23):

$$\tau_K = \begin{cases} \frac{h_K}{2\|\mathbf{u}\|}\left(1 - \frac{1}{Pe_K}\right), & if\ Pe_K > 1 \\ 0, & if\ Pe_K \in [0, 1], \end{cases}$$

where $Pe_K = \frac{\mathbf{u}h_K}{2k}$. Another possible choice for τ for DW, can be found in (27, 31):

$$\begin{aligned}
\tau_K &= \tfrac{h_K}{2|u|_p}\xi(Pe_K), \\
Pe_K &:= \tfrac{m_K|u|_p h_K}{2k}, \\
\xi(Pe_K) &:= \begin{cases} Pe_K, & Pe_K \in [0,1) \\ 1, & Pe_K \geq 1 \end{cases}, \\
|u|_p &:= (\textstyle\sum_{i=1}^n |u_i|^p)^{\frac{1}{p}}, \ p \geq 1, \\
m_K &= \min\left\{\tfrac{1}{3}, 2C_K\right\}, \ C_K \ s.t. \ C_K \textstyle\sum_K h_K^2 \|\Delta v\|^2_{L^2(K)} \leq \|\nabla v\|^2_{L^2(\Omega)}, \ \forall v \in H_0^1(\Omega).
\end{aligned} \tag{3.7}$$

It is evident that it is quite complicated, although the definition of τ is a crucial point for method's performance. The usual definition of Peclet number is a little bit modified by the presence of m_K, which accounts for the specific finite element polynomial employed.

We conclude this paragraph underlining that the SUPG solution and these stabilization methods in general limit the damage caused by poor resolution of layers and leads to a more accurate solution outside them, where the solution is not varying rapidly. Neverless in general they are *not* free of oscillations (68): in fact in (23) it is demonstrated that if τ is large enough there are not oscillations, but the solution is overly diffusive, with boundary layers which are much wider than those displayed by the exact solution. An alternative is to choose a smaller τ whose solution presents small oscillation, but is less diffusive. Moreover these oscillations may be controlled by nonlinear modifications, e.g. it is possible to add a *shock-capturing non linear viscosity term* to (3.4) which guarantees extra control in directions different form the streamline one, over the strong gradients that causes oscillations (cfr. e.g. (41, 42) and (46) for the multimensional case).

3.3 Bubble functions

The *Bubble functions method* consists in applying the classical Galerkin method to (3.3), using an *enriched discretizing space*

$$V_h^b := V_h \oplus B,$$

as approximating space. Thus (3.3) is equivalent to

$$\text{find } \Theta_h + \Theta_B \in V_h^b \text{ s.t. } a(\Theta_h+\Theta_B, w_h+w_B) = F(w_h+w_B), \qquad \forall w_h+w_B \in V_h^b, \tag{3.8}$$

where V_h is a standard finite elements space while the finite dimensional space of *bubble functions* is $B := \oplus B_K$, $B_K \subset V$ is a finite dimensional subspace whose dimension

depends on K. The idea is to enhances the quality of the discrete solution on V_h incorporating information interior to elements, where the Galerkin method provides no information.

Let us see briefly how a particular kind of B_K can be defined: *the residual free one* (29).

Suppose that $V = H_0^1(\Omega)$; since a and F are linear, (3.8) is equivalent to find $\Theta_h + \Theta_B \in V_h^b$ s.t.

$$\begin{array}{rcll} a(\Theta_h + \Theta_B, w_h) & = & F(w_h), & \forall w_h \in V_h, \\ a(\Theta_h + \Theta_B, w_B) & = & F(w_B), & \forall w_B \in B. \end{array} \tag{3.9}$$

Since $B := \oplus B_K$ we can rewrite the last equation

$$a(\Theta_{B,K}, w_{B,K})_K = -(a(\Theta_h, \cdot)_K - F(\cdot)_K)(w_{B,K}), \qquad \forall w_{B,K} \in B_K,$$

where we have restricted functions and integrals over K. The *residual-free space* B_K, derives the bubbles from certain element-level boundary-value problems, i.e. it is defined s.t. $\Theta_{B,K} \in H_0^1(K)$ is a solution of

$$a(\Theta_{B,K}, w)_K = -(a(\Theta_h, \cdot)_K - F(\cdot)_K)(w), \qquad \forall w \in H_0^1(K), \tag{3.10}$$

viz. such that $\Theta_h + \Theta_{B,K}$ solves exactly equation (3.8) in the interior of K. Observe that B_K can be thought as the image of the affine operator $R_K : V_h \mid_K \to H_0^1(K)$, s.t. $R_K(\Theta_h \mid_K) = \Theta_{B,K}$, where $V_h \mid_K$ is the restriction of V_h over K. A basis of B_K can be defined starting from a basis of V_h (29).

The conceptual viewpoint of bubble functions method, is to attack the original problem first with the Galerkin method involving standard and simple polynomial finite element spaces and *correct* any deficiencies with regard to stability, by systematically enriching the space with residual-free bubbles (12).

3.4 Subgrid-scale (or variational multiscale) methods

The Subgrid-scale methods (SGS) were first introduced in (39) and compared to other stabilization methods in (16) and (37): they can be viewed as an extension of ρ-methods and also of stabilization methods based on the introduction of bubble functions to the finite element space. The idea is resumed in (12): since the standard Galerkin method with simple polynomial space is an inadequate numerical paradigm for many practically important problems (in particular, those involving fine scale features that are numerically unresolvable due to the length scale of elements composing the mesh) a

new method is presented which accounted for fine scales, in order to accurately calculate the coarse scales. The variational multiscale procedure consists in two steps: a first subproblem is solved for the fine scales in terms of the coarse scales; then the result is substituted into a second subproblem involving *only* the coarse scales (*subgrid-scale model*), solvable with the standard Galerkin method.

Figure 3.3: *Example of resolved (left) and unresolved scales (right). This picture is taken from (39)*

Consider the general scalar linear problem (2.4) with $\Gamma_n = \emptyset$ and the corresponding variational formulation

$$\text{find } \Theta \in V \text{ s.t. } a(\Theta, w) = F(w), \qquad \forall w \in V, \tag{3.11}$$

where $a(\Theta, w) = (\mathcal{L}\Theta, w)$ and $F(w) = (f, w)$, for a suitable space V (which accounts for Dirichlet boundary conditions, e.g. $V = H_0^1(\Omega)$). Suppose that $V = \bar{V} \oplus V'$, i.e. the unknown

$$\Theta = \bar{\Theta} + \Theta',$$

(cfr. figure 3.3) where $\bar{\Theta}$ (*resolvable (or coarse) scale*) is the part which can be described using the finite element mesh, whereas Θ' represents the *unresolvable (or subgrid, or fine) scales* of Θ, i.e. the variations that cannot be reproduced: e.g. in the case of convection-dominated diffusion phenomena Θ' consists in thin layers with steep gradients (39). In general the coarse and the fine scales may overlap or be disjoint, and the fine scales may be globally or locally defined.

A strong hypothesis is the following: *it is assumed that Θ' vanishes on the boundaries of the elements*, viz. $\Theta' = 0$ on ∂K, for all $K \in \mathcal{T}_h$. This means that $V' = \oplus_K H_0^1(K)$. Then Θ' is a solution of

$$\begin{cases} \mathcal{L}\Theta' &= f - \mathcal{L}\bar{\Theta}, \quad in \quad K \\ \Theta' &= 0 \qquad\quad on \quad \partial K. \end{cases} \tag{3.12}$$

The aim is not to describe unresolvable scales in details, instead we wish to compute their *effect* on resolvable scales. Observed that (3.11) leads to two subproblems, as-

suming $\Theta = \bar{\Theta} + \Theta^{'}$ and $v = \bar{v} + v^{'}$:

$$\begin{array}{rcll} a(\bar{\Theta},\bar{w}) + a(\Theta^{'},\bar{w}) & = & F(\bar{w}), & \forall \bar{w} \in \bar{V} \\ a(\bar{\Theta},w^{'}) + a(\Theta^{'},w^{'}) & = & F(w^{'}), & \forall w^{'} \in V^{'} \end{array} \Leftrightarrow \begin{array}{rcll} a(\bar{\Theta},\bar{w}) + (\Theta^{'},\mathcal{L}^{*}\bar{w}) & = & F(\bar{w}), & \forall \bar{w} \in \bar{V} \\ (\mathcal{L}\bar{\Theta},w^{'}) + (\mathcal{L}\Theta^{'},w^{'}) & = & F(w^{'}), & \forall w^{'} \in V^{'}. \end{array} \tag{3.13}$$

Let $g(x,y):\ \Omega \times \Omega \to \mathbb{R}$ be the *Green's function* of (3.12), i.e., for every fixed $y \in \Omega$ it is a solution of

$$\left\{ \begin{array}{rcll} \left(\mathcal{L}g(\cdot,\mathbf{y})\right)(\mathbf{x}) & = & \delta(\mathbf{x}-\mathbf{y}) & for \quad \mathbf{x} \in K \\ g(\cdot,\mathbf{y})(\mathbf{x}) & = & 0 & for \quad \mathbf{x} \in \partial K, \end{array} \right. \tag{3.14}$$

where $\delta(\mathbf{x}-\mathbf{y}) := \left\{ \begin{array}{ll} 0, & \mathbf{x} \neq \mathbf{y}, \\ 1, & \mathbf{x} = \mathbf{y}, \end{array} \right.$ is the *Dirac delta* distribution. If $\mathcal{L}^{*}$ is the adjoint of $\mathcal{L}$, then (67)

$$\left\{ \begin{array}{rcll} \left(\mathcal{L}^{*}g(\mathbf{x},\cdot)\right)(\mathbf{y}) & = & \delta(\mathbf{x}-\mathbf{y}), & for \quad \mathbf{y} \in K \\ g(\mathbf{x},\cdot)(\mathbf{y}) & = & 0, & for \quad \mathbf{y} \in \partial K. \end{array} \right. \tag{3.15}$$

Then it is possible to write the solution of (3.12) in terms of g:

$$\Theta^{'}(\mathbf{y}) = -\sum_{K \in \mathcal{T}_h} \int_K g(\mathbf{x},\mathbf{y}) \left(\mathcal{L}\Theta^{'}\right)(\mathbf{x}) d\omega(\mathbf{x}) = -\sum_{K \in \mathcal{T}_h} \int_K g(\mathbf{x},\mathbf{y}) \left(f - \mathcal{L}\bar{\Theta}\right)(\mathbf{x}) d\omega(\mathbf{x}).$$

Let us define the bounded integral operator $M:\ H^{-1}(\Omega) \to H_0^1(\Omega)$ s.t.

$$M(s)(\cdot) := -\sum_{K \in \mathcal{T}_h} \int_K g(\mathbf{x},\cdot) s(\mathbf{x}) d\omega(\mathbf{x})$$

for every $L^2(\Omega)$ function s. Then

$$\Theta^{'}(\mathbf{y}) = M\left(\mathcal{L}\bar{\Theta} - f\right)(\mathbf{y}).$$

Observed that *"the subgrid scales $\Theta^{'}$ are driven by the residual of the resolved scales"* (39).

Then, using (3.13), $\bar{\Theta}$ must satisfy

$$a(\bar{\Theta},\bar{v}) + (\mathcal{L}^{*}(\bar{v}), M\left(\mathcal{L}\bar{\Theta} - f\right)) = F(\bar{v}),$$

which is a restatement of the continuous problem (3.11). As Hughes says (39):

> "The multiscale interpretation amounts to assuming that unresolvable, fine-scale behavior exists within each element, but not on element boundaries. Up to this assumption, the effect of the unresolved, fine scales on the resolved coarse-scales behavior is exactly accounted for".

To discretize it, consider $\Theta_h \in \bar{V}_h \subseteq \bar{V}$ and $v_h \in \bar{V}_h$, finite element approximations of $\bar{\Theta}$ and $\bar{v}$ respectively. Moreover observe that in general also the Green's function is unknown, thus it must be approximated, using M_h instead of M obtaining:

$$\text{find } \Theta_h \in \bar{V}_h \text{ s.t. } a(\Theta_h, v_h) + (\mathcal{L}^*(v_h), M_h\left(\mathcal{L}\Theta_h - f\right)) = F(v_h),\ \forall v_h \in \bar{V}_h. \tag{3.16}$$

As underlined in (16), different approximations of M, i.e. of g, will lead to different subgrid scale models. A general picture of variational methods is given in (50), where all previous results are collected and generalized. It is presented a paradigm for a variational multiscale method, not assuming that $u^{'}$ is zero at boundary elements ∂K. This is a strong hypothesis which is equivalent to state that the subgrid scales are completely confined within element interiors, being only locally significative, although in the physical situation they are nonlocal and affect (*pollute*) all the solution. A way to overcome this assumption is to interpret equation

$$(\mathcal{L}\bar{\Theta}, w^{'}) + (\mathcal{L}\Theta^{'}, w^{'}) = F(w^{'}),\ \forall w^{'} \in V^{'},$$

in (3.13) as an L^2 projection on $V^{'}$, i.e. considering

$$\begin{cases} \Pi^{'}\mathcal{L}\Theta^{'} &= -\Pi^{'}(\mathcal{L}\bar{\Theta} - f), \quad in \quad \Omega \\ \Theta^{'} &= 0, \qquad\qquad\qquad on \quad \partial\Omega, \end{cases} \tag{3.17}$$

where $\Pi^{'} : H^{-1}(\Omega) \to V^{'}$ is the L^2 projection and we suppose that $\bar{\Theta}$ is exact on $\partial\Omega$, for a linear second order operator $\mathcal{L}$, with $\Gamma_n = \emptyset$.

Then consider the following Green's function problem:

$$\begin{cases} \Pi^{'}\left(\mathcal{L}^* g^{'}(\mathbf{x}, \cdot)\right)(\mathbf{y}) &= \Pi^{'}\delta(\mathbf{x} - \mathbf{y}), \quad for \quad \mathbf{y} \in \Omega \\ g^{'}(\mathbf{x}, \cdot)(\mathbf{y}) &= 0, \qquad\qquad\qquad for \quad \mathbf{y} \in \partial\Omega. \end{cases} \tag{3.18}$$

This is not the usual Green's function and it is called *fine scales Green's function.*

Then

$$\Theta^{'}(\mathbf{y}) = -\int_{\Omega} g^{'}(\mathbf{x}, \mathbf{y})\left(\mathcal{L}\bar{\Theta} - f\right)(\mathbf{x}) d\omega(\mathbf{x}) = M^{'}\left(\mathcal{L}\bar{\Theta} - f\right),$$

for a suitable integral operator $M^{'}$. Finally the problem to be solved is finding $\bar{\Theta} \in \bar{V}$ such that

$$a(\bar{\Theta}, \bar{v}) + (\mathcal{L}^*(\bar{v}), M^{'}\left(\mathcal{L}\bar{\Theta} - f\right)) = F(\bar{v}),$$

for every $\bar{v} \in \bar{V}$, *where the effect of the fine scales on the coarse scales are nonlocal.* This construction holds if $\bar{\Theta}$, $\bar{v}$, $\Theta^{'}$ and $v^{'}$ are smooth functions, which is too restrictive if finite element spaces are used, since $\bar{\Theta}$, $\bar{v}$ are smooth only on elements interiors, but

have slope discontinuities across element boundaries. These means that when we split the integral over Ω in the sum of integrals over elements K and apply integration-by-parts, we must consider *nonvanishing* element boundary terms.

For a comparison of all these methods in the one-dimensional case, cfr. e.g. (37).

3.5 Some other stabilization methods

For solutions that requires a stronger control of the gradients the *Galerkin \ Gradient-Least-Squares method (GGLS)* has been introduced: in (3.4) $\mathcal{L}_h$ is defined as

$$\mathcal{L}_h(\Theta_h, f; w_h) = \sum_K \tau_K(\nabla(\mathcal{L}\Theta_h - f), \nabla(\mathcal{L}w_h)),$$

if $\mathcal{L}$ is a diffusion-convection operator (cfr. (27) and references therein). To ensure more control, especially when both convection and reaction dominate diffusion, another method, *Galerkin-Least-Squares \ Gradient-Least-Squares method (GLSGLS)*, was presented in (35) and consists in combining GLS and GGLS. For the one dimensional problem

$$\mathcal{L}_h(\Theta_h, f; w_h) = \sum_K \tau_K(\mathcal{L}\Theta_h - f, \mathcal{L}w_h) + \sum_K \gamma_K(\nabla(\mathcal{L}\Theta_h - f), \nabla(\mathcal{L}w_h)).$$

As presented in (68) *local projection stabilization methods*, are based on the observation that for SUPG, only the term

$$\sum_K \tau_k(\mathbf{u} \cdot \nabla\Theta_h, \mathbf{u} \cdot \nabla v_h)_K$$

in a_h in (3.4) is responsible for increased stability. The idea is to use two finite element spaces V_h and D_h and subtract from $\mathbf{u} \cdot \nabla\Theta_h$ its L_2-projection π_h into D_h, obtaining

$$\sum_K \tau_k(\mathbf{u} \cdot \nabla\Theta_h - \pi_h(\mathbf{u} \cdot \nabla\Theta_h), \mathbf{u} \cdot \nabla v_h - \pi_h(\mathbf{u} \cdot \nabla v_h))_K.$$

Observe that V_h and D_h can live on different meshes or on the same mesh. These *projection techniques* are also applied to the Stokes problem (18, 21).

Another method presented in (68) is *Edge Stabilization (or Continuous Interior Penalty)* method: the idea is to add to the Galerkin bilinear form a certain jump terms, on interior edges $\{E\}$, viz.

$$b_h(\Theta_h, v_h) = \sum_E \tau_E(\mathbf{u} \cdot [\nabla\Theta_h]_E, \mathbf{u} \cdot [\nabla v_h]_E)_E,$$

and incorporating boundary conditions in the weak sense. Also the *Discountinuos Galerkin* method and its variants are used for stabilization: substantially they uses discontinuous weighting functions, i.e.

$$b_h(\Theta_h, v_h) = \sum_E \tau_E([\Theta_h]_E, [v_h]_E)_E.$$

As mentioned in (16), for the transient diffusion-convection-reaction problem, the *Characteristic Galerkin method* (which coincides with SUPG under some restrictions) and the *Taylor-Galerkin method* (considered as the finite element counterpart of the Lax-Wendroff scheme for finite difference methods) are alternatives to the previously introduced methods.

It is also possible to stabilize a problem using adaptive techniques, which are based on *a posteriori error analysis*, viz. as a function of the residual of the problem. The basic principle is to refine the mesh wherever an a posteriori error estimator indicates the presence of large local errors in the computed solution. In this way hopefully are identified regions affected by local singularities, shocks or interior layers. The aim is to achieve a balance between refined and unrefined regions so that good global accuracy is attained without introducing too mesh points (for an introduction to these methods cfr. e.g. (59, 67)).

In (68) it is underlined that stabilized methods provide good approximations in subdomains that exclude layers. To resolve them, it is possible to use *layer-adapted meshes*, constructed a priori based on precise information on the structure of the layer (67). In (55) SUPG (or SDFEM) behavior in layer regions is analyzed, in presence of anisotropic layer-adapted meshes.

Finally the *Adjoint Weighted Equation method* (AWE) has been presented for advection-reaction problems (63) and extended to the heat equation (6) and incompressible plane-stress elasticity (2). Observe that the Least Squares method can be derived from the Galerkin-Least Squares, for $\tau \to \infty$: the AWE method can be interpreted as the same limit of the DW method.

For example consider the linear first order advection-reaction operator

$$\mathcal{L} := \mathbf{u} \cdot \nabla\Theta + \sigma\Theta$$

and define $a(u, v) := (\mathcal{L}u, \mathcal{L}^*v)$, where $\mathcal{L}^* = -\mathbf{u} \cdot \nabla\Theta + \sigma\Theta$ is the *adjoint operator*. The *Adjoint Weighted Equation* method consists in finding $\Theta \in V$ s.t.

$$a(\Theta, v) = (f, \mathcal{L}^*v), \qquad \forall v \in V. \tag{3.19}$$

Given an Hilbert space $V_h \subset V \subset L^2(\Omega)$ the discrete method consists in finding $\Theta_h \in V_h$ s.t.

$$a(\Theta_h, v_h) = (f, \mathcal{L}^* v_h), \qquad \forall v_h \in V_h. \tag{3.20}$$

As for the Least Squares method, stability and convergence results are based on *Lax Milgram lemma* (Theorem A.2.1).

AWE has been used also for solving inverse problems, based on heat equation (6) and incompressible plane-stress elasticity (2).

3.6 The Best Approximation Weighted Residual (BAWR) method

In this section we will follow (20).

Consider the boundary value problem (2.4).Given two Hilbert spaces $V = H^1(\Omega)$ and $W = H^1_{\Gamma_d}(\Omega)$, where $H^1_{\Gamma_d}(\Omega) := \left\{v \in H^1(\Omega),\ v_{|\Gamma_d} = 0\right\}$, the weak (variational) formulation of (2.4) is the following (cfr. section 2.2):

$$\text{find } \Theta \in V \text{ s.t. } a(\Theta, w) = F(w), \qquad \forall w \in W, \tag{3.21}$$

where $a(\cdot,\cdot)$ is a bilinear form $a : V \times W \to \mathbb{R}$, defined as

$$a(\Theta, w) := \int_\Omega k \nabla\Theta \nabla w d\omega + \int_\Omega \mathbf{u} \cdot \nabla\Theta w d\omega + \int_\Omega \sigma\Theta w d\omega,$$

and $F(\cdot)$ is a continuous linear operator $F :\ W \to \mathbb{R}$,

$$F(w) := (f, w) + \int_{\Gamma_n} k\Theta_n w d\gamma.$$

Observe that defining $\tilde{F}(w) := F(w) - a(G, w)$, where $G \in H^1(\Omega)$ is the Dirichlet lift of g, i.e. $G\,|_{\Gamma_d} = g$, and considering $V = W = H^1_{\Gamma_d}(\Omega)$ we obtain a variational formulation equivalent to (3.21):

$$\text{find } \Theta \in W \text{ s.t. } a(\Theta, w) = \tilde{F}(w), \qquad \forall w \in W. \tag{3.22}$$

Then for the well-posedness of the continuous problem (2.4) it suffices Lax Milgram's Lemma (64).

Consider now a finite element discretization of the domain Ω, whose refinement level is characterized by a parameter h, and identify the computational domain Ω_h with Ω. To obtain an approximate numerical solution of (3.21), two N_h finite-dimensional

Hilbert subspaces $V_h \subset V$ and $W_h \subset W$ are chosen. The *Petrov Galerkin* finite element approximation of (3.21) is formulated as follows:

$$\text{find } \Theta_h \in V_h \text{ s.t. } a(\Theta_h, w_h) = F(w_h), \qquad \forall w_h \in W_h. \tag{3.23}$$

Observing that $F(w_h) - a(\Theta_h, w_h)$ is equivalent, in the distributional sense, for suitable a, F, V and W to $(f - \mathcal{L}\Theta_h, w_h)$, the discrete problem (3.23) can be restated equivalently in the following way:

$$\text{find } \Theta_h \in V_h \; s.t. \; 0 = (f - \mathcal{L}\Theta_h, w_h) = (\mathcal{L}(\Theta - \Theta_h), w_h) \qquad \forall w_h \in W_h. \tag{3.24}$$

(3.24) is called also a *weighted-residuals method* (71): the solution will satisfy an orthogonality condition between the residual of the strong form of the differential problem and the space of weighting-functions.

The aim of this section is to present a parameter-free, analytic method of choosing the space of weighting-functions W_h that brings to the best-approximation in the norm induced by the inner-product adopted in the weighted-residuals formulation, and to propose an efficient numerical realization for this strategy. We call it the *Best Approximation Weighted Residuals (BAWR)* method. Note that this optimality is always achievable, while the standard Galerkin method is optimal in a stronger norm but only for problems dominated by diffusion.

To derive the method, first homogeneous boundary conditions are considered in (2.4), i.e. $\Theta_d = 0 = \Theta_n$. The general case, with inhomogeneous boundary conditions, will be discussed later on. Suppose that $D(\mathcal{L}) \subset V$ is a dense subset of V identified by boundary conditions on $\mathcal{L}$, then the *adjoint operator of* $\mathcal{L}$: $D(\mathcal{L}) \subset V \to W^{'}$, is a differential operator, $\mathcal{L}^*$: $D(\mathcal{L}^*) \subset W \to V^{'}$, such that for every $v \in V$, $w \in W$ the *Lagrange identity* holds, i.e.

$$(\mathcal{L}v, w) = (v, \mathcal{L}^* w)\,, \tag{3.25}$$

where $D(\mathcal{L}^*)$ is chosen such that (3.25) is verified (33, 57). Observe that we are identifying V and W with their dual spaces respectively $V^{'}$ and $W^{'}$ in the definition of $\mathcal{L}^*$: this is justified by the Riesz theorem, since we are dealing with Hilbert spaces (57). Moreover we are identifying the duality pairing between $V^{'}$ and W or $W^{'}$ and V with the L_2 scalar product on Ω: $\langle \mathcal{L}v, w\rangle = (\mathcal{L}v, w)$ and $\langle v, \mathcal{L}^* w\rangle = (v, \mathcal{L}^* w)$; this is possible because we are dealing with L^2 functions.

The following Theorem 3.6.1 defines the BAWR method, i.e. it tells how to choose W_h in (3.24) in such a way that the approximation error $\Theta - \Theta_h$ is L_2-orthogonal to the space V_h of the approximating functions.

3.6 The Best Approximation Weighted Residual (BAWR) method

Theorem 3.6.1 *Given the model problem (2.4), with homogeneous boundary conditions, i.e.* $\Theta_d = 0 = \Theta_n$*, and the numerical method (3.24), let* $\left\{\phi_h^i\right\}_{i=1,\dots,N_h}$ *be a basis of* $V_h \subset V$*. Let* $\mathcal{L} : V \to W$ *be a one-to-one linear differential operator (isomorphism). Consider the following adjoint boundary-value-problems:*
for each $\phi_h^i \in V_h$*,* $i = 1, \dots, N_h$*, find* $w^i \in W$ *such that*

$$\mathcal{L}^* w^i = \phi_h^i \tag{3.26}$$

where $\mathcal{L}^*$ *is the adjoint of* $\mathcal{L}$ *and* $D(\mathcal{L}^*)$ *is defined imposing adjoint boundary conditions. Define* $W_h := span\left\{w^i\right\}_{i=1,\dots,N_h}$.

Then W_h *is a finite* N_h*-dimensional Hilbert space. Moreover using it as weighting space in (3.24),* Θ_h *is the* L_2*-projection of* $\Theta \in V$ *onto* V_h*, i.e.*

$$(v_h, \Theta - \Theta_h) = 0 \qquad \forall v_h \in V_h \qquad . \tag{3.27}$$

Proof. Since $\mathcal{L}^*$ is an isomorphism (9), the existence of w^i in (3.26) is guaranteed for every $i = 1, \dots, N_h$.

Let us first demonstrate that $\left\{w^i\right\}_i$ are linearly independent, i.e. they form a basis of W_h. In fact, choosing $\alpha_i \in \mathbb{R}$ for all i, $\sum_i \alpha_i w^i = 0 \Leftrightarrow$ (by the linearity of $\mathcal{L}^*$)$\Leftrightarrow$ $\sum_i \alpha_i \mathcal{L}^* w^i = 0 \Leftrightarrow$ (3.26) $\Leftrightarrow \sum_i \alpha_i \phi_h^i = 0 \Leftrightarrow \alpha_i = 0 \ \forall i$, since $\left\{\phi_h^i\right\}_i$ is a basis of V_h.

As a consequence $dim W_h = N_h$. Moreover as a finite dimensional vectorial subspace of W, it is also an Hilbert space, with the induced norm.

Considering now (3.24) and applying *Lagrange identity* (3.25), there exists $D(\mathcal{L}^*)$, defined imposing *adjoint boundary conditions* (33), s.t. the following holds:

$$\text{find } \Theta_h \in V_h \text{ s.t. } 0 = \left(\mathcal{L}^* w_h, \Theta - \Theta_h\right), \forall w_h \in W_h. \tag{3.28}$$

Since $\left\{w^i\right\}_i$ is a basis of W_h, there exists $\beta \in \mathbb{R}^{N_h}$ s.t. $w_h = \sum_i \beta_i w^i$. Using the linearity of $\mathcal{L}^*$, $0 = \left(\mathcal{L}^* w_h, \Theta - \Theta_h\right) = \sum_i \beta_i \left(\mathcal{L}^* w^i, \Theta - \Theta_h\right)$.

Using now (3.26) for every $i = 1, \dots, N_h$, (3.28) is equivalent to

$$\text{find } \Theta_h \in V_h \text{ s.t. } 0 = \sum_i \beta_i \left(\phi_h^i, \Theta - \Theta_h\right), \forall \beta \in \mathbb{R}^{N_h}. \tag{3.29}$$

Since $\left\{\phi_h^i\right\}_i$ is a basis of V_h, there exists β s.t. every $v_h \in V_h$ can be written as $\sum_i \beta_i \phi_h^i$. Thus (3.29) is equivalent to

$$\text{find } \Theta_h \in V_h \text{ s.t. } 0 = \left(v_h, \Theta - \Theta_h\right), \forall v_h \in V_h. \tag{3.30}$$

□

The above Theorem holds in general for weighted-residuals methods. Here we will consider the case of V_h being a space of Finite Elements and W_h a space of continuous test functions.

3.6.1 Application to the steady diffusion-convection-reaction equation

Consider the model problem (2.4). First of all we need to define the adjoint operator of $\mathcal{L}$, $\mathcal{L}^*$, applying both the identity (3.25) and the *Green's formula*.

$$\begin{array}{rcl} \int_\Omega -div(k\nabla\Theta)wd\omega & = & (\textit{Green's formula}) = \int_{\partial\Omega} -k\frac{\partial\Theta}{\partial n}wd\gamma + \int_\Omega k\nabla\Theta\nabla wd\omega \\ & = & \int_{\partial\Omega} -k\frac{\partial\Theta}{\partial n}wd\gamma + \int_\Omega div(k\Theta\nabla w)d\omega - \int_\Omega \Theta div(k\nabla w)d\omega \\ & = & (\textit{Green's formula}) = \int_{\partial\Omega} k\left[-\frac{\partial\Theta}{\partial n}w + \frac{\partial w}{\partial n}\Theta\right]d\gamma - \int_\Omega \Theta div(k\nabla w)d\omega \end{array} .$$

$$\begin{array}{rcl} \int_\Omega \mathbf{u}\cdot\nabla\Theta w d\omega & = & \int_\Omega div(\mathbf{u}w\Theta)d\omega - \int_\Omega div(\mathbf{u}w)\Theta d\omega \\ & = & (\textit{Green's formula}) = \int_{\partial\Omega} w\Theta\mathbf{u}\cdot\mathbf{n}d\gamma - \int_\Omega div(\mathbf{u}w)\Theta d\omega \end{array} .$$

$$(\mathcal{L}\Theta, w) = \int_{\partial\Omega}\left[-k\frac{\partial\Theta}{\partial n}w + k\frac{\partial w}{\partial n}\Theta + w\Theta\mathbf{u}\cdot\mathbf{n}\right]d\gamma + \int_\Omega \left[-div(k\nabla w) - div(\mathbf{u}w) + \sigma w\right]ud\omega. \tag{3.31}$$

If we define $\mathcal{L}^*w = -div(k\nabla w) - div(\mathbf{u}w) + \sigma w$, so

$$(\mathcal{L}\Theta, w) = \int_{\partial\Omega}\left[-k\frac{\partial\Theta}{\partial n}w + k\frac{\partial w}{\partial n}\Theta + w\Theta\mathbf{u}\cdot\mathbf{n}\right]d\gamma + (\Theta, \mathcal{L}^*w)\,. \tag{3.32}$$

3.6.1.1 The case of homogeneous boundary conditions

To satisfy the *Lagrange identity* (3.25) it is necessary to choose $D(\mathcal{L}) \subset V$ and $D(\mathcal{L}^*) \subset W$ such that
$\int_{\partial\Omega}\left[-k\frac{\partial\Theta}{\partial n}w + k\frac{\partial w}{\partial n}\Theta + w\Theta\mathbf{u}\cdot\mathbf{n}\right]d\gamma = 0$ (33). To do this we consider the following homogeneous boundary-value problem

$$\left\{\begin{array}{rcll} \mathcal{L}\Theta & = & f, & on \quad \Omega \\ \Theta & = & 0, & on \quad \Gamma_d \\ \frac{\partial\Theta}{\partial n} & = & 0, & on \quad \Gamma_n. \end{array}\right. \tag{3.33}$$

where $f \in L^2(\Omega)$ is assigned. Choosing $D(\mathcal{L}) \subset V = H^1(\Omega)$ such that every $v \in D(\mathcal{L})$ satisfies boundary conditions of the problem (3.33), we obtain the condition:

$$\int_{\Gamma_n}\left[k\frac{\partial w}{\partial n} + w\mathbf{u}\cdot\mathbf{n}\right]\Theta d\gamma = 0.$$

So we can define the *adjoint operator* associated to (3.33):

$$\left\{\begin{array}{rcll} \mathcal{L}^*w & = & -div(k\nabla w) - div(\mathbf{u}w) + \sigma w & \\ w & = & 0 & on \quad \Gamma_d \\ k\frac{\partial w}{\partial n} + (\mathbf{u}\cdot\mathbf{n})w & = & 0 & on \quad \Gamma_n \end{array}\right. \tag{3.34}$$

where $D(\mathcal{L}^*) \subset W = H^1(\Omega)$ is the set of functions w that satisfy the boundary conditions of (3.34). Now we have all the ingredients to apply the BAWR method to (3.33):

$\forall i = 1, \ldots, N_h$,

$$\begin{array}{rcl} \text{find } \Theta_h \in V_h \text{ s.t. } 0 & = & (w^i, f - \mathcal{L}\Theta_h) \\ & = & (w^i, \mathcal{L}(\Theta - \Theta_h)) = (\mathcal{L}^* w^i, \Theta - \Theta_h) = (\phi_h^i, \Theta - \Theta_h) \end{array} \tag{3.35}$$

using as $W_h = span\left\{w^i\right\}_{i=1,\ldots,N_h}$ the $H^1(\Omega)$ N_h- finite dimensional subspace defined in Theorem 3.6.1.

3.6.1.2 The case of non homogeneous boundary conditions

Our aim is now to extend results derived for the homogeneous boundary conditions case (3.33) to the general non homogeneous linear differential boundary-value problem (2.4). As mentioned in (33), in the general non homogeneous case we cannot construct an adjoint differential operator, i.e. an operator like (3.34) with its own boundary conditions, such that *Lagrange identity* (3.25) holds. Consider the corresponding homogeneous problem (3.33) and its adjoint operator (3.34). Suppose that u and w satisfy (2.4)'s and (3.34)'s boundary conditions respectively. Then (3.32) is equivalent to:

$$(\mathcal{L}\Theta, w) = \int_{\Gamma_n} -k\Theta_n w d\gamma + (\Theta, \mathcal{L}^* w) + \int_{\Gamma_d} k\frac{\partial w}{\partial n}\Theta_d d\gamma. \tag{3.36}$$

For inhomogeneous boundary conditions, using (3.36), the BAWR method can be restated as follows: $\forall i = 1, \ldots, N_h$

$$\begin{array}{rcl} \text{find } \Theta_h \in V_h \text{ s.t.0} & = & (w^i, \mathcal{L}(\Theta - \Theta_h)) \\ & = & \int_{\Gamma_n} -k(\Theta_n - \Theta_{n_h})w^i d\gamma + \left(\Theta - \Theta_h, \mathcal{L}^* w^i\right) + \int_{\Gamma_d} k\frac{\partial w^i}{\partial n}(\Theta_d - \Theta_{d_h})d\gamma \\ & = & \int_{\Gamma_n} -k(\Theta_n - \Theta_{n_h})w^i d\gamma + \left(\Theta - \Theta_h, \phi_h^i\right) + \int_{\Gamma_d} k\frac{\partial w^i}{\partial n}(\Theta_d - \Theta_{d_h})d\gamma, \end{array} \tag{3.37}$$

where $\Theta_{n_h},\ \Theta_{d_h} \in V_h$ are approximations on the h-step grid of boundary data Θ_n and Θ_d respectively. The term $\int_{\Gamma_n} -k(\Theta_n - \Theta_{n_h})w^i d\gamma + \int_{\Gamma_d} k\frac{\partial w^i}{\partial n}(\Theta_d - \Theta_{d_h})d\gamma$ quantifies the *deviation from orthogonality*. This is meaningful because it tells us that to control it, it is sufficient to guarantee an accurate approximation of the boundary terms Θ_n and Θ_d:

$$\begin{array}{rcl} \left|\int_{\Gamma_n} -k(\Theta_n - \Theta_{n_h})w^i d\gamma\right| & \leq & \|k\|_\infty \|\Theta_n - \Theta_{n_h}\|_{L^2(\Gamma_n)} \left\|w^i\right\|_{L^2(\Gamma_n)} \\ & \leq & C(\Omega)\|k\|_\infty \|\Theta_n - \Theta_{n_h}\|_{L^2(\Gamma_n)} \left\|w^i\right\|_{H^1} \\ \left|\int_{\Gamma_d} -k\frac{\partial w^i}{\partial n}(\Theta_d - \Theta_{d_h})d\gamma\right| & \leq & \|k\|_\infty \|\Theta_d - \Theta_{d_h}\|_{L^2(\Gamma_d)} \left\|\nabla w^i \cdot \mathbf{n}\right\|_{L^2(\Gamma_d)} \\ & \leq & C'(\Omega)\|k\|_\infty \|\Theta_d - \Theta_{d_h}\|_{L^2(\Gamma_d)} \left\|w^i\right\|_{H^1} \end{array},$$

where $C(\Omega)$ and $C'(\Omega)$ are constants and we have used the Cauchy Schwarz inequality and the continuity of the trace operator (15).

3.6.2 BAWR stability and convergence estimates

3.6.2.1 Optimality in L^2-norm

Lemma 3.6.1 *Under the assumptions of Theorem 3.6.1, the BAWR solution Θ_h is optimal in the L^2-norm.*

Proof. This is an immediate consequence of Theorem 3.6.1, which shows that Θ_h is the L^2-projection of Θ onto V_h, thus it verifies the following L^2 convergence estimate of minimal distance:

$$\|\Theta - \Theta_h\|_{L^2} = \inf_{v_h \in V_h} \|\Theta - v_h\|_{L^2} . \tag{3.38}$$

The well known *Projection Theorem* (19) guarantees that Θ_h is the best approximation of Θ in the L_2-norm, and this best-approximation always exists, in the hypotheses made, and is unique.

□

3.6.2.2 Convergence and stability estimates using H^1-norm

In the following part of this section we analyze BAWR's convergence in H^1- norm.

As stated in the generalization of the *Lax Milgram's Lemma* for Petrov Galerkin problems, i.e. Theorem A.3.2 (4, 65), if W_h and V_h in (3.23) are chosen such that a verifies

$$\begin{array}{l} \sup_{\Theta_h \in V_h} |a(\Theta_h, w_h)| > 0, \ \forall w_h \in W_h, \ w_h \neq 0, \\ \exists \ C_h > 0 \ s.t. \ \sup_{w_h \in W_h, \|w_h\|_W \neq 0} \frac{|a(\Theta_h, w_h)|}{\|w_h\|_W} \geq C_h \|\Theta_h\|_V \ \ \forall \Theta_h \in V_h \end{array} , \tag{3.39}$$

then it follows existence and uniqueness of the solution of (3.23) and the following stability and convergence estimates hold:

$$\begin{array}{l} \|\Theta_h\|_V \leq \frac{\|F\|_{W'}}{C_h} \\ \|\Theta - \Theta_h\|_V \leq \left(1 + \frac{\gamma}{C_h}\right) inf_{v_h \in V_h} \|\Theta - v_h\|_V \end{array} . \tag{3.40}$$

It is important to note that in estimates (3.40) the constants are functions of the coefficients of the differential operator (2.3) and depends on the choice of V_h and W_h. In the following it will be proved that estimates (3.40) hold for the BAWR method.

Lemma 3.6.2 *Under the assumptions of Theorem 3.6.1,* $\sup_{\Theta_h \in V_h} |a(\Theta_h, w_h)| > 0, \ \forall w_h \in W_h, \ w_h \neq 0.$

Proof. First of all observe that, since $w_h \in W_h$, there exists $\alpha \in \mathbb{R}^{N_h}$ such that $w_h = \sum_{i=1}^{N_h} \alpha_i w^i$. Then $a(\Theta_h, w_h) = \sum_i \alpha_i a(\Theta_h, w^i) = \sum_i \alpha_i (\mathcal{L}\Theta_h, w^i) = \sum_i \alpha_i (\Theta_h, \mathcal{L}^* w^i)$ = (Theorem 3.6.1) $= \sum_i \alpha_i (\Theta_h, \phi_h^i)$, for every $w_h \in W_h$ and $\Theta_h \in V_h$.

Thus $\sup_{\Theta_h \in V_h} |a(\Theta_h, w_h)| = \sup_{\Theta_h \in V_h} \left|\sum_i \alpha_i (\Theta_h, \phi_h^i)\right| = \sup_{\Theta_h \in V_h} \left|(\Theta_h, \sum_i \alpha_i \phi_h^i)\right|$. Define now $v_h := \sum_i \alpha_i \phi_h^i$: observe that $v_h = 0$ iff $\alpha_i = 0$, $\forall i$ iff $w_h = 0$. Thus $\sup_{\Theta_h \in V_h} |a(\Theta_h, w_h)| = \sup_{\Theta_h \in V_h} |(\Theta_h, v_h)| \geq |(v_h, v_h)| = \|v_h\|_2^2 > 0$, for every $w_h \in W_h$, $w_h \neq 0$.

□

Lemma 3.6.3 *Under the assumptions of Theorem 3.6.1, there exists a constant $C_h > 0$ s.t.*

$$\sup_{w_h \in W_h, \|w_h\|_1 \neq 0} \frac{|a(\Theta_h, w_h)|}{\|w_h\|_1} \geq C_h \|\Theta_h\|_1 \ \ \forall \Theta_h \in V_h. \tag{3.41}$$

Proof. For all $w_h \in W_h$, there exists $\alpha \in \mathbb{R}^{N_h}$ such that $w_h = \sum_{i=1}^{N_h} \alpha_i w^i$: then, as proved in Lemma 3.6.2, $a(\Theta_h, w_h) = \sum_i \alpha_i (\Theta_h, \phi_h^i)$, for every $w_h \in W_h$ and $\Theta_h \in V_h$.

Using this fact we can restate (3.41) in the following equivalent way: there exists a constant $C_h > 0$ s.t.

$$\sup_{\alpha \in \mathbb{R}^{N_h}, \alpha \neq 0} \frac{\left|\sum_i \alpha_i (\Theta_h, \phi_h^i)\right|}{\|\sum_i \alpha_i w^i\|_1} \geq C_h \|\Theta_h\|_1 \ \ \forall \Theta_h \in V_h. \tag{3.42}$$

First observe that for every $C_h > 0$, $\Theta_h = 0$ satisfies inequality (3.41), since $a(\Theta_h, w_h) = 0$. Thus it can be assumed that $\Theta_h \neq 0$.

To prove the existence of a suitable constant C_h, first of all observe that, using the triangular inequality

$$\left\| \sum_i \alpha_i w^i \right\|_1 \leq \sum_i |\alpha_i| \left\|w^i\right\|_1,$$

it holds

$$\sup_\alpha \frac{\left|\sum_i \alpha_i (\Theta_h, \phi_h^i)\right|}{\|\sum_i \alpha_i w^i\|_1} \geq \sup_\alpha \frac{\left|\sum_i \alpha_i (\Theta_h, \phi_h^i)\right|}{\sum_i |\alpha_i| \|w^i\|_1} \tag{3.43}$$

Now for all $\Theta_h \in V_h$, choose $\bar{\alpha} \in \mathbb{R}^{N_h}$ such that for all $i = 1, \ldots, N_h$

$$\bar{\alpha}_i = \begin{cases} 1, & if\ (\Theta_h, \phi_h^i) \geq 0, \\ -1, & if\ (\Theta_h, \phi_h^i) < 0 \end{cases}$$

Thus the following inequality holds:

$$\sup_{\alpha} \frac{\left|\sum_i \alpha_i(\Theta_h, \phi_h^i)\right|}{\sum_i |\alpha_i| \left\|w^i\right\|_1} \geq \frac{1}{\sum_i \left\|w^i\right\|_1} \sum_i \left|(\Theta_h, \phi_h^i)\right|. \tag{3.44}$$

Finally define $D_h(\Theta_h) := \frac{1}{\sum_i \|w^i\|_1} \sum_i \left|(\Theta_h, \phi_h^i)\right|$: since $\Theta_h \neq 0$ $D_h(\Theta_h) > 0$.

If C_h is sufficiently small such that $C_h \leq \inf_{\Theta_h \in V_h, \Theta_h \neq 0} \frac{D_h(\Theta_h)}{\|\Theta_h\|_1}$, then for all $\Theta_h \in V_h$, $\Theta_h \neq 0$

$$D_h(\Theta_h) \geq C_h \left\|\Theta_h\right\|_1,$$

thus, using (3.43) and (3.44), it holds

$$\sup_{\alpha} \frac{\left|\sum_i \alpha_i(\Theta_h, \phi_h^i)\right|}{\left\|\sum_i \alpha_i w^i\right\|_1} \geq C_h \left\|\Theta_h\right\|_1 \tag{3.45}$$

for all $\Theta_h \neq 0$, i.e. (3.42).

To conclude the proof, it remains to prove that

$$\inf_{\Theta_h \in V_h, \Theta_h \neq 0} \frac{D_h(\Theta_h)}{\left\|\Theta_h\right\|_1} > 0. \tag{3.46}$$

For all $\Theta_h \in V_h$, there exists a vector $\beta \in \mathbb{R}^{N_h}$ such that $\Theta_h = \sum_i \beta_i \phi_h^i$. Then (3.46) is equivalent to

$$\inf_{\beta \neq 0} \Phi(\beta) > 0, \tag{3.47}$$

defining $\Phi(\beta) := \frac{\sum_i \left|(\sum_j \beta_j \phi_h^j, \phi_h^i)\right|}{\sum_i \|w^i\|_1 \left\|\sum_j \beta_j \phi_h^j\right\|_1}$. Observe that $\Phi : \mathbb{R}^{N_h} \setminus \{0\} \to 0$ is a continuous positive function. Consider the bidimensional case ($N_h = 2$): $\Phi = \Phi(\beta_1, \beta_2)$ and restrict Φ along straight lines, i.e. consider $\psi(m)(\beta_1) := \Phi(\beta_1, \beta_2)_{|\beta_2 = m\beta_1}$, $m \in \mathbb{R}$. Since $\psi(m)(\beta_1) = \psi(m) \in \mathbb{R}$, Φ is constant along each straight line of slope m passing trought the origin. Moreover $\psi(m)$ is a positive continuous function with respect to m and has an infimum greater than zero. Thus $\inf_{\beta \neq 0} \Phi(\beta) > 0$.

□

Proposition 3.6.1 *Under the assumptions of Theorem 3.6.1, estimates (3.40) hold, i.e.*

$$\begin{array}{rcl} \left\|\Theta_h\right\|_1 & \leq & \frac{\|F\|_1}{C_h} \\ \left\|\Theta - \Theta_h\right\|_1 & \leq & \left(1 + \frac{\gamma}{C_h}\right) inf_{v_h \in V_h} \left\|\Theta - v_h\right\|_1 \end{array}, \tag{3.48}$$

where C_h is a suitable positive constant.

Proof. Since Lemmas 3.6.2 and 3.6.3 hold, it is sufficient to apply the generalization of the *Lax Milgram's Lemma* for Petrov Galerkin problems (64, 65).

□

Lemma 3.6.4 *Under the assumptions of Proposition 3.6.1, denote with r the degree of polynomials used in the approximation finite element space. If there exists $p \geq r$ such that $u \in H^{p+1}(\Omega)$, then*

$$\|\Theta - \Theta_h\|_1 \leq \tilde{C}_{h,r} h^r \left|\Theta\right|_{r+1}, \tag{3.49}$$

for a suitable $\tilde{C}_{h,r} > 0$, function of h and r.

Proof. It is a consequence of proposition 3.6.1 and of interpolation estimates (64, 65).

More precisely, let $\Pi_h^r \Theta$ be the r-degree polynomial interpolant of Θ upon the discretization of Ω characterized by h, then, for a suitable constant $C_r > 0$, it holds

$$\begin{aligned} \|\Theta - \Theta_h\|_1 &\leq \left(1 + \tfrac{\gamma}{C_h}\right) \inf_{v_h \in V_h} \|\Theta - v_h\|_1 \leq \left(1 + \tfrac{\gamma}{C_h}\right) \|\Theta - \Pi_h^r \Theta\|_1 \\ &\leq (64,\, 65) \leq \left(1 + \tfrac{\gamma}{C_h}\right) C_r h^r \left|u\right|_{r+1} = \tilde{C}_{h,r} h^r \left|\Theta\right|_{r+1}, \end{aligned}$$

for $\tilde{C}_{h,r} := \left(1 + \frac{\gamma}{C_h}\right) C_r$.

□

3.6.2.3 Numerical study of the order of convergence

An analytic study of (3.49) is not simple, since it is not evident how the constant $\tilde{C}_{h,r}$ depends on h. Thus, to conclude this section, the order of convergence in H^1-norm of BAWR is estimated numerically, using as a test case the example of section 3.6.4.1, solved using the implementation described in section 3.6.3 and P1 finite elements (i.e. $r = 1$). For completeness also the L^2-norm estimate of the order of convergence has been included.

As it can be seen in Figure 3.4, the order of convergence of BAWR is asyntotically two and one, using respectively L^2 and H^1 norms to compute the error. A Galerkin approximation computed on a much finer grid is used as real solution Θ.

Moreover, Figure 3.4 shows the optimality of BAWR in the L^2 norm and a behaviour much more independent from h for the constant $\tilde{C}_{h,r}$ (i.e. the error curve in logarithmic scale is more straight), with respect to the Galerkin method. It is interesting to note that the BAWR method performs better also for small h steps, and the gap between the two methods becomes more evident as the Peclet number grows (i.e. as ν increases).

Moreover, as a *Petrov Galerkin method* in which the operators a and F are not approximated, the BAWR method is strongly consistent.

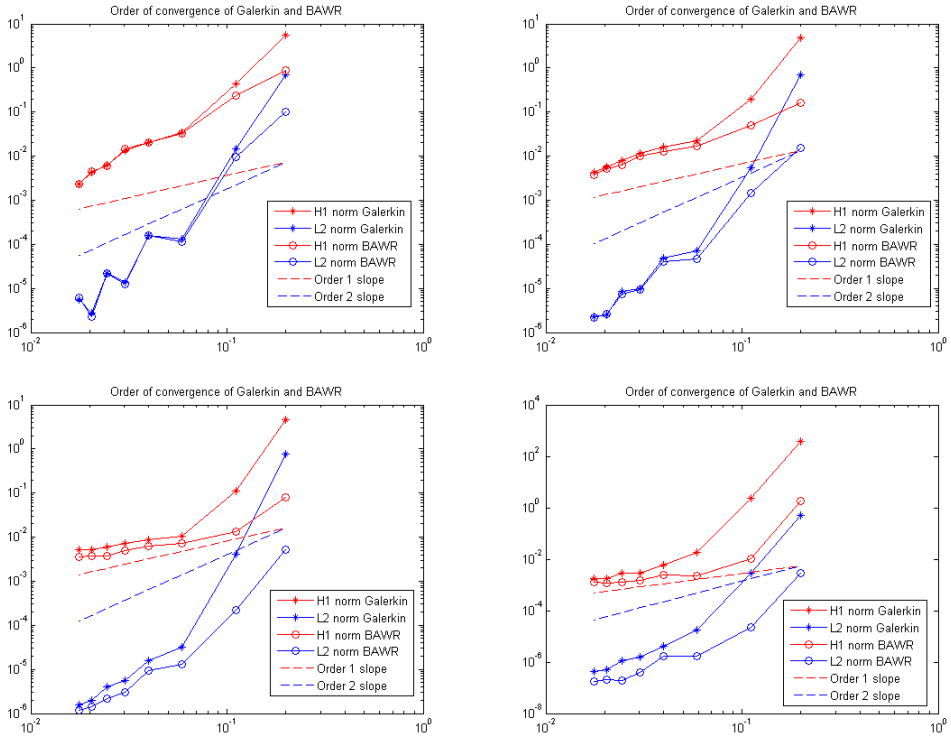

Figure 3.4: *Numerical estimate of the order of convergence of BAWR and of Galerkin, obtained solving example of section 3.6.4.1 for different values of the mesh step h. Using logarithmic scales: step size h vs. L^2 and H^1 norms of BAWR and Galerkin errors. A Galerkin approximation computed on a much finer grid is used as real solution Θ. From left to right, from the top to the bottom: example 3.6.4.1 solved for different values of the convection parameter ν: 200, 600, 2000, 10000.*

3.6.3 BAWR finite elements

The crucial part of the BAWR method is the definition of W_h, which consists in solving the following N_h *adjoint problems*: for each $\phi_h^i \in V_h$, $i = 1, \ldots, N_h$, find $w^i \in W$ such that

$$\begin{cases} \mathcal{L}^* w^i &= \phi_h^i \quad in\ \Omega \\ w^i &= 0 \quad on \quad \Gamma_d \\ k\frac{\partial w^i}{\partial n} + (\mathbf{u}\cdot\mathbf{n})w^i &= 0 \quad on \quad \Gamma_n. \end{cases} \tag{3.50}$$

These are boundary-value problems defined on the same domain of the original one, but with an highly localized forcing term. For all $i = 1, \ldots, N_h$, the numerical approximation of w^i can be obtained e.g. using a standard Galerkin finite element

approximation, i.e. finding $w_h^i \in V_h^w \subset W$, such that for all $\chi_h \in V_h^w$, $\chi_h \mid_{\Gamma_d} = 0$ we have

$$\int_\Omega \left[k\nabla w_h^i \nabla \chi_h + \mathbf{u}\cdot w_h^i \nabla \chi_h + \sigma w_h^i \chi_h\right] d\omega = \int_\Omega \phi_i \chi_h d\omega. \tag{3.51}$$

To obtain useful weighting functions $\{w_h^i\}$ there are two possibilities. The first is to adopt for V_h^w a finer discretization than that used for V_h, e.g. $V_h^w = V_{\frac{h}{2}}$ or $V_h^w = V_{\frac{h}{4}}$, while the second is to use higher order polynomials on the same discretization.

Then W_h is approximated by $span\{w_h^i\}$. Observe that, to define it, it is necessary to solve (3.50), for all i, i.e. N_h distinct adjoint problems all defined on Ω. Clearly this is not efficient, thus an alternative implementation of the method is proposed in the following section.

3.6.3.1 Efficient computation of the weighting functions

In this section we derive an efficient numerical method to approximate w^i, $i = 1, \ldots, N_h$. To have an idea of what will be presented, observe that from the computational point-of-view, the strong locality of the right-hand-side in the adjoint problems (3.50) suggests that also the solution could be only locally meaningful, thus requiring a reduced computational cost. Moreover, the N_h adjoint problems (3.50) are very similar among them: the forcing term is very local and always the same; it only changes its point of application. These observations suggest at the same time to localize the problem, i.e. to approximate w_h^i which in general have support Ω with a function only with a local support, and also to compute only one reference weighting function. Let's analyze this idea.

Assume that $\left\{\phi_h^i\right\}_i$ is a *lagrangian basis* of V_h. For every $i = 1, \ldots, N_h$ we approximate $w^i \in W$, solution of (3.50), with a compactly supported function $\hat{w}^i \in W$. Without loss of generality, in this section we suppose that $\Omega_{loc}^i := \text{supp}\hat{w}^i = \text{supp}\phi_h^i$, for all i, and we denote its boundary with $\Gamma_{loc}^i = \partial\Omega_{loc}^i$. In general it is sufficient that $\Omega_{loc}^i \supseteq \text{supp}\phi_h^i$.

Define $\hat{w}^i$ in the following way:

$$\hat{w}^i = \begin{cases} \tilde{w}^i, & on \quad \Omega_{loc}^i \\ 0, & on \quad \Omega \setminus \Omega_{loc}^i, \end{cases} \tag{3.52}$$

where $\tilde{w}^i$ is the solution of the i-th *localized adjoint problem*:

$$\begin{cases} \mathcal{L}^*\tilde{w}^i = \phi_h^i \mid_{\Omega_{loc}^i} & on \quad \Omega_{loc}^i \\ \tilde{w}^i = 0 & on \quad \Gamma_{loc}^i \end{cases} . \tag{3.53}$$

It is important to note that although the adjoint problems (3.50) depend by construction on the boundary conditions on u imposed in the original problem (3.33), the *localized* problems (3.53) are independent from these conditions: they only depend on the PDE coefficients and the choice of the space of approximating functions ϕ_h^i. This is important because if the PDE coefficients are constant on Ω we can compute only *one* (reference) $\tilde{w}^i$ and translate it to define $\{\hat{w}^k\}$. In fact for every $k = 1, \ldots, N_h$ consider a projection function Π_k s.t. $\Pi_k \Omega_{loc}^k \subseteq \Omega_{loc}^i$ is the part of Ω_{loc}^i corresponding to the k-th node: $\Pi_k \Omega_{loc}^k \subset \Omega_{loc}^i$ if k is a boundary node, whereas $\Pi_k \Omega_{loc}^k \equiv \Omega_{loc}^i$ if k is an internal node (cfr. Figure 3.5).

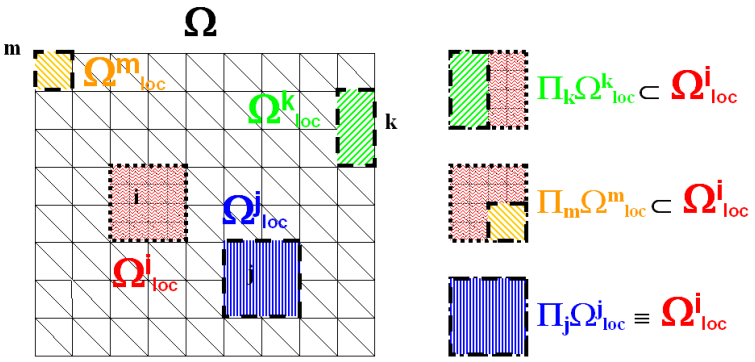

Figure 3.5: *Examples of the projection $\Pi_k \Omega_{loc}^k$: it coincides with Ω_{loc}^i for internal nodes, whereas it is a part of it for boundary nodes.*

Define now

$$\hat{w}^k = \begin{cases} \tilde{w}^i_{|\Pi_k \Omega_{loc}^k}, & on \quad \Omega_{loc}^k \\ 0, & on \quad \Omega \setminus \Omega_{loc}^k \end{cases}. \tag{3.54}$$

The *approximated weighting function space* is defined as $\hat{W}_h = \text{span}\{\hat{w}^k\}_k$ and we denote with $\hat{u}_h$ the corresponding approximate solution. Moreover for every $i = 1, \ldots, N_h$ we approximate $\tilde{w}^i$ with $\tilde{w}_h^i$, Galerkin's solution of (3.53), obtained using a finer discretization than that corresponding to V_h. Then for every $k = 1, \ldots, N_h$ we define

$$\hat{w}_h^k = \begin{cases} \tilde{w}_h^i{}_{|\Pi_k \Omega_{loc}^k}, & on \quad \Omega_{loc}^k \\ 0, & on \quad \Omega \setminus \Omega_{loc}^k. \end{cases} \tag{3.55}$$

Thus we approximate $\hat{W}_h$ with *span* $\{\hat{w}_h^k\}$, for simplicity in the following we will identify them.

Now, it is important to identify the error that we introduce in the BAWR solution by approximating the space W_h, defined by Theorem 3.6.1, with $\hat{W}_h$. We denote by

$\hat{u}_h$ the corresponding BAWR approximate solution, obtained using $\hat{W}_h$ as weighting function space in (3.23).
Using identity (3.24) we obtain $\forall\ i = 1, \dots, N_h$

$$\begin{aligned}
0 &= \left(\hat{w}^i, \mathcal{L}(\Theta - \hat{\Theta}_h)\right) = \int_\Omega \mathcal{L}(\Theta - \hat{\Theta}_h)\hat{w}^i d\omega = \int_{\Omega^i_{loc}} \mathcal{L}(\Theta - \hat{\Theta}_h)\hat{w}^i d\omega + \int_{\Omega \setminus \Omega^i_{loc}} \mathcal{L}(\Theta - \hat{\Theta}_h)\hat{w}^i d\omega \\
&= (3.52) = \int_{\Omega^i_{loc}} \mathcal{L}(\Theta - \hat{\Theta}_h)\tilde{w}^i \\
&= \int_{\Omega^i_{loc}} -div(k\nabla(\Theta - \hat{\Theta}_h))\tilde{w}^i d\omega + \int_{\Omega^i_{loc}} \mathbf{u}\cdot\nabla(\Theta - \hat{\Theta}_h)\tilde{w}^i d\omega + \int_{\Omega^i_{loc}} \sigma(\Theta - \hat{\Theta}_h)\tilde{w}^i d\omega \\
&= (3.31) = I_{\Gamma^i_{loc}} + \int_{\Omega^i_{loc}} \left[-div(k\nabla\tilde{w}^i) - \mathbf{u}\cdot\nabla\tilde{w}^i - div(\mathbf{u})\tilde{w}^i + \sigma\tilde{w}^i\right](\Theta - \hat{\Theta}_h)d\omega \\
&= I_{\Gamma^i_{loc}} + \left(\mathcal{L}^*\tilde{w}^i, \Theta - \hat{\Theta}_h\right)_{L^2(\Omega^i_{loc})},
\end{aligned}$$

denoting with

$$\begin{aligned}
I_{\Gamma^i_{loc}} &:= \int_{\Gamma^i_{loc}} \left[-k\tfrac{\partial(\Theta - \hat{\Theta}_h)}{\partial n}\tilde{w}^i + k\tfrac{\partial\tilde{w}^i}{\partial n}(\Theta - \hat{\Theta}_h) + \tilde{w}^i(\Theta - \hat{\Theta}_h)\mathbf{u}\cdot\mathbf{n}\right] d\gamma \\
&= (3.53) = \int_{\Gamma^i_{loc}} k\tfrac{\partial\tilde{w}^i}{\partial n}(\Theta - \hat{\Theta}_h)d\gamma
\end{aligned} \tag{3.56}$$

the *localization error*.
Moreover, observing that

$$\begin{aligned}
\left(\mathcal{L}^*\hat{w}^i, \Theta - \hat{\Theta}_h\right)_{L^2(\Omega)} &= (3.52) = \left(\mathcal{L}^*\tilde{w}^i, \Theta - \hat{\Theta}_h\right)_{L^2(\Omega^i_{loc})} = (3.53) = \\
&= \left(\phi^i_h, \Theta - \hat{\Theta}_h\right)_{L^2(\Omega^i_{loc})} = \left(\phi^i_h, \Theta - \hat{\Theta}_h\right)_{L^2(\Omega)},
\end{aligned}$$

we obtain the following approximation of the BAWR method (3.37): $\forall i = 1, \dots, N_h$

$$\text{find } \hat{\Theta}_h \in V_h \text{ s.t.} 0 = \left(\hat{w}^i, \mathcal{L}(\Theta - \hat{\Theta}_h)\right) = \left(\mathcal{L}^*\hat{w}^i, \Theta - \hat{\Theta}_h\right) + I_{\Gamma^i_{loc}} = \left(\phi^i_h, \Theta - \hat{\Theta}_h\right) + I_{\Gamma^i_{loc}}. \tag{3.57}$$

If we are dealing with homogeneous boundary conditions (i.e. $\Theta_d = 0 = \Theta_n$), we observe that although the BAWR formulation (3.35) gives the optimal L_2-solution, the approximation BAWR method (3.57) introduce a global *localization error* $\sum_{i=1}^{N_h} I_{\Gamma^i_{loc}}$ which is proportional to the exactness of $\hat{\Theta}_h$ on Γ^i_{loc}, for all i. Consider Θ' such that $\Theta = \hat{\Theta}_h + \Theta'$: observe that Θ' represents the unresolved scales (50). Assuming that $\Theta = \hat{\Theta}_h$ on Γ^i_{loc}, is equivalent to consider $\Theta' = 0$ on Γ^i_{loc}: this is similar to the assumption made by Hughes in (39), presenting the *subgrid-scale model* (cfr. section 3.4), where Θ' is chosen equal to zero on the boundary of every element of Ω. Observe moreover that this hypothesis can be relaxed using the approximation BAWR method: in fact it is possible to reduce the localization error, choosing a bigger support for $\hat{w}^i$, for all i. We estimate the global localization error $\sum_{i=1}^{N_h} I_{\Gamma^i_{loc}}$ numerically, using as a test case the example described in section 3.6.4.1: the error estimates are obtained comparing the BAWR solution on an h-step grid with the Galerkin approximation on a $\frac{h}{8}$-grid, which is assumed to be the reference true solution. For different increasing Peclet values, obtained considering ν equal to 50, 200, 600, 1000, we obtain respectively 0.0011159, 0.00018404, $4.6108 \cdot 10^{-5}$, $4.0047 \cdot 10^{-5}$ as estimates of the global localization error.

In particular, suppose now that $\Gamma_n \neq \emptyset$. Solving (3.52) for every i means that we are imposing homogeneous *Dirichlet* boundary conditions on $\Gamma^i_{loc} \cap \partial\Omega$, for all i, i.e. we are neglecting the Neumann component on Γ_n, i.e.

$$(k\frac{\partial w^i}{\partial n} + (\mathbf{u}\cdot\mathbf{n})w^i = 0) \;\approx\; \left(w^i = 0\right). \tag{3.58}$$

in (3.50). Observe that for convection dominated problems it is proper to neglect $k\frac{\partial w^i}{\partial n}$ at least where the convection field $\mathbf{u}$ is not tangential to the boundary.

To understand the consequences of neglecting Neumann boundary conditions on $\Gamma^i_{loc} \cap \Gamma_n$ for the localized adjoint problems (cfr. approsimation 3.58), choose Ω^i_{loc} s.t. $\Gamma^i_{loc} \cap \Gamma_n \neq \emptyset$ and let $\tilde{w}^i_{loc}$ be the solution of the following problem, which is a local problem on Ω^i_{loc} derived from (3.50):

$$\begin{cases} \mathcal{L}^*\tilde{w}^i_{loc} &= \phi_i\,|_{\Omega^i_{loc}} \quad in \quad \Omega^i_{loc} \\ \tilde{w}^i_{loc} &= 0 \quad on \quad \Gamma^i_{loc}\cap\Gamma_d \quad \bigcup \quad \Gamma^i_{loc}\setminus\partial\Omega \\ k\frac{\partial\tilde{w}^i_{loc}}{\partial n} + (\mathbf{u}\cdot\mathbf{n})\tilde{w}^i_{loc} &= 0 \quad on \quad \Gamma^i_{loc}\cap\Gamma_n. \end{cases} \tag{3.59}$$

Comparing boundary conditions of problems (3.59) defined on Ω^i_{loc} and (3.53), we observe that they are different iff $\Gamma^i_{loc} \cap \Gamma_n \neq \emptyset$. Through a numerical example, we analyze numerically this difference, choosing a square $\Omega^i_{loc} = [-h, h] \times [-h, h]$, $h = 0.0625$, $\Gamma^i_{loc}\cap\Gamma_n = \{(\xi, h),\ (h, \zeta),\ \xi, \zeta \in [0, h]\}$. The resulting $\tilde{w}^i_{loc}$ are shown in Figure 3.6: they could be compared with the corresponding $\tilde{w}^i_h$, i.e. the one with Dirichlet homogeneous boundary conditions (cfr. Figure 3.8). As explained in section 3.6.4, all weighting functions are the P1-Galerkin solutions of local problems (3.59) and (3.53) on Ω^i_{loc}, whose uniform local grid has step size $\frac{h}{4}$. Note that in Figure 3.6 the error becomes negligible as Peclet number grows, i.e., dealing with convection dominated problems, using $\hat{W}_h$ is proper also when $\Gamma_n \neq \emptyset$.

If we are dealing with inhomogeneous boundary conditions (i.e. $u_d \neq 0 \neq u_n$), comparing the approximate BAWR method (3.57) with the BAWR one (3.37), it can be observed that the *deviation from orthogonality term* in (3.37) is replaced by the *localization error* in (3.57).

In section 3.6.4 $\hat{W}_h$ has been used, instead of the real BAWR weighting function space W_h.

Finally, before describing numerical results, we observe that thanks to the localization and the use of a single reference test function, there is only a relatively small increase in the cost of the BAWR method compared to the standard Galerkin method, analogous to the increase in the computational cost required by other existing stabilization methods. More precisely, the computation of the reference test function $\tilde{w}^i_h$ on

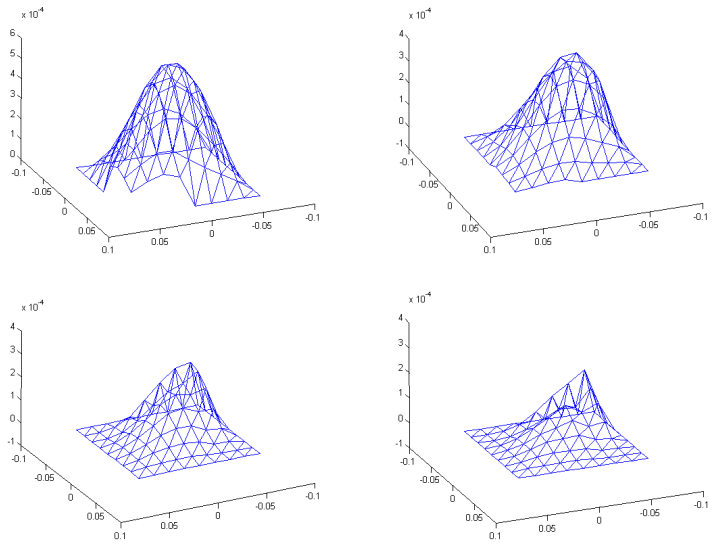

Figure 3.6: *The function $\tilde{w}^i_{h_{loc}}$ on the domain Ω^i_{loc} chosen s.t. $\Gamma^i_{loc} \cap \Gamma_n \neq 0$. Different values of the convection parameter ν are considered: 50, 200, 600, 1000.*

Ω^i_{loc} is negligible even for middle-small sized problems, while an $O(N_h)$ increase occurs in the computation of the element matrices, since the test function $\tilde{w}^i_h$ is defined on a finer mesh and thus each h-grid element integral must be computed on this finer mesh.

3.6.4 Numerical examples

In this section we present few examples to demonstrate the effectiveness of the BAWR method for convection dominated problems. Consider the two dimensional convection-diffusion equation (2.4) on $\Omega = [0,1] \times [0,1]$, with constant coefficients $k = 1$, $\sigma = 0$, $\mathbf{u} = \nu(\cos\vartheta, \sin\vartheta)$, where $\nu > 0$ is constant in space and varying across the experiments to test different values of the ratio between convection and diffusion (the Peclet number $Pe := \frac{\nu}{k}$). Suppose that $P1$ elements are used for both BAWR and Galerkin methods. Weighting functions are computed in an approximated way (cfr. section 3.6.3.1), choosing $\Omega^i_{loc} = [-h, h] \times [-h, h]$, and considering a grid of step $\frac{h}{4}$ over it, denoting with h Ω's uniform grid step (cfr. Figure 3.7). Thus only one reference adjoint problem (3.53) is solved on Ω^i_{loc}, using a $P1$ Galerkin method, on a finer discretization of step $\frac{h}{4}$ (i.e. $V^w_h = V_{\frac{h}{4}}$).

The chosen step $h = 0.0625$ is not too small, to show that the BAWR method behaves better than the Galerkin one on coarser grids, as already observed through the

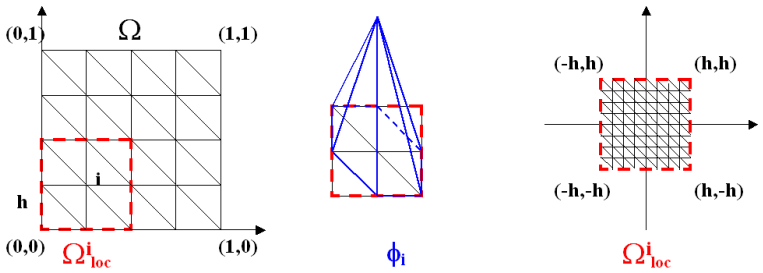

Figure 3.7: *The domain Ω (left), a $P1$ shape function on an internal node of Ω (center) and the local domain Ω^i_{loc} for the adjoint problems. In this example $\Omega^i_{loc} \supset supp\phi^i$*

convergence analysis (cfr. Figure 3.4).

3.6.4.1 Dirichlet homogeneous boundary conditions and point wise forcing term

Consider $\vartheta = \frac{\pi}{4}$ and a point-load f, applied at the point $(0.71, 0.79)$.

In this test we apply homogeneous Dirichlet boundary conditions in all $\partial\Omega$, i.e. $\Gamma_d = \partial\Omega$ and $\Theta_d = 0$, thus the localized weighting functions $\tilde{w}^i$, solutions of (3.53), satisfy exactly the corresponding adjoint boundary conditions (3.50) on $\partial\Omega$. An analogous example is given also in (59), where BAWR is compared in 1D problems with other stabilized methods, in particular SUPG (originally presented in (48)).

In Figure 3.8 are represented the reference $\tilde{w}^i_h$ on Ω^i_{loc} for different values of the convection parameter ν: 50, 200, 600, 1000. Note that, with large Peclet numbers, $\tilde{w}^i_h$ becomes oscillatory. This is not surprising, since $\tilde{w}^i_h$ is a Galerkin approximation (on a locally refined grid). It is noteworthy that the corresponding BAWR solution (Figure 3.9) does not suffer from such instability. This is another way to see that the BAWR solution $\hat{u}_h$ is more stable than the Galerkin one.

In Table 3.1 we report L^2 and H^1 error estimates obtained solving this example, for different values of the convection parameter ν: 50, 200, 600, 1000 (column 1). The values in the table are the error results in the L_2-norm and H_1-norm, obtained on an h-step grid with the BAWR method $(\Theta - \hat{\Theta}_h)$ and the Galerkin method $(\Theta - \Theta^{Gal}_h)$. The error is computed assuming as true solution a Galerkin approximation computed on an $\frac{h}{8}$-grid. The comparison between the second and the third columns of the table is useful to confirm the L^2-optimality of BAWR stated in Lemma 3.6.1. Moreover it can be seen that the BAWR solution results progressively more accurate than the Galerkin one, as the Peclet number grows, even using H^1-norm.

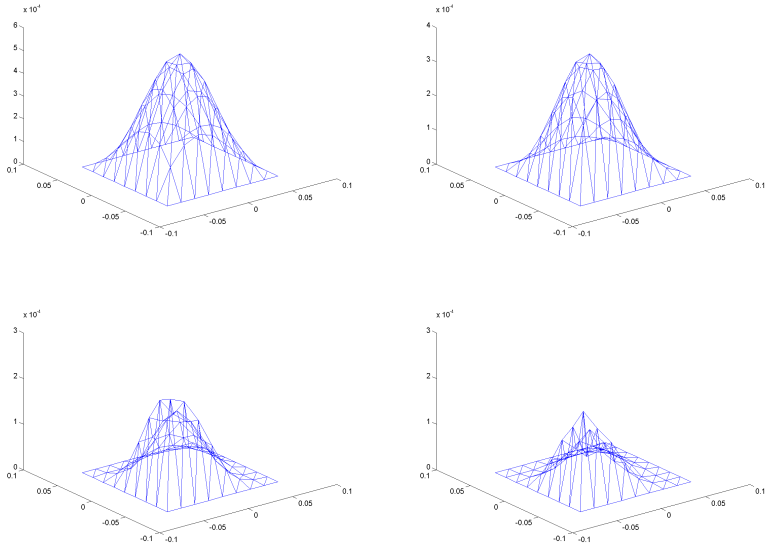

Figure 3.8: *The reference $\tilde{w}_h^i$ on Ω_{loc}^i for different values of the Peclet number: 50, 200, 600, 1000.*

Example 3.6.4.1						
ν	$\left\|\Theta - \hat{\Theta}_h\right\|_2$	$\left\|\Theta - \Theta_h^{Gal}\right\|_2$	$\left\|\Theta - \hat{\Theta}_h\right\|_{H^1}$	$\left\|\Theta - \Theta_h^{Gal}\right\|_{H^1}$	$\|\Theta\|_2$	$\|\Theta\|_{H^1}$
50	0.00021004	0.00023707	0.055512	0.048982	0.0090632	0.1571
200	0.00011652	0.00012898	0.032797	0.034742	0.0014733	0.060604
600	$4.7675{\cdot}10^{-5}$	$6.9811{\cdot}10^{-5}$	0.016595	0.02203	0.00027831	0.023921
1000	$2.8021{\cdot}10^{-5}$	$4.8036{\cdot}10^{-5}$	0.011666	0.016145	0.00012303	0.015057

Table 3.1: *Global approximation error results in the L_2-norm for example 3.6.4.1, for the BAWR solution $\hat{\Theta}_h$ and the Galerkin solution Θ_h^{Gal}, for various Peclet numbers. The exact solution is a Galerkin one on a mesh of size $\frac{h}{8}$.*

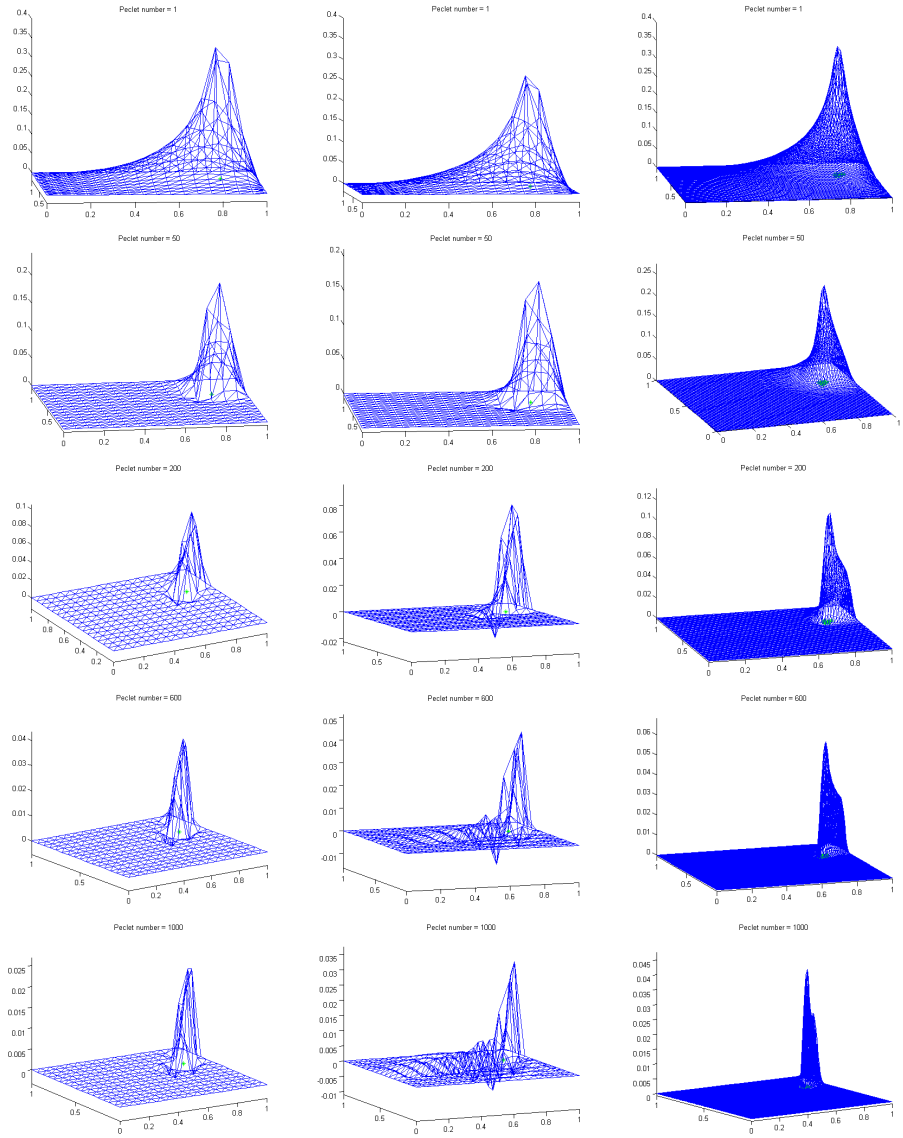

Figure 3.9: *Example 3.6.4.1. Left: BAWR solution for homogeneous boundary conditions for different values of the convection parameter ν (and corresponding Peclet number): 1, 50, 200, 600, 1000. Center: Galerkin solution on the same mesh of step size h. Right: Galerkin solution on a finer mesh of step size $\frac{h}{8}$.*

3.6.4.2 Inhomogeneous dirichlet boundary conditions and null forcing term

In this section few examples are given, similar to those presented e.g. in (14, 27, 35). Boundary conditions are sketched in Figure 3.10.

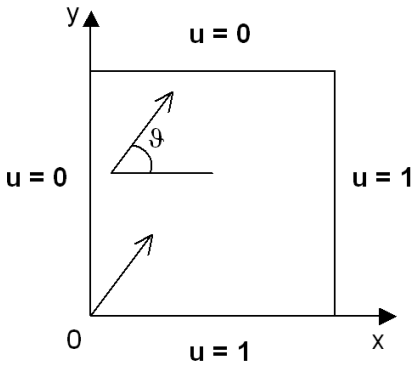

Figure 3.10: *Problem statement of example 3.6.4.2.*

$\vartheta = \frac{\pi}{4}$

As in example of section (3.6.4.1), $\vartheta = \frac{\pi}{4}$, i.e. the velocity field is aligned to the mesh. It is known that this is a more stable choice dealing with convection dominated problems (36). In this example a discontinuity in Dirichlet boundary conditions at the inflow causes the formation of an internal boundary layer. Choosing $f = 0$ and $\nu = 10^6$, the BAWR method works pretty well (cfr. figure 3.12), also compared to SUPG and bubbles (cfr. figure 3.11 taken from (14) for a comparison).

$\vartheta = \frac{\pi}{3}$

This example is analogous to the previous one, but the vector field is not aligned with the mesh. Thus Dirichlet boundary condition at the outflow give rise to an outflow boundary layer (cfr. figure 3.13). Considering $\nu = 10^3$ the error results in the L_2-norm of BAWR and Galerkin, are respectively 0.11478 and 0.53283. The error is computed assuming as the true solution a Galerkin approximation obtained on a $\frac{h}{8}$-grid (cfr. Figure 3.13).

$\vartheta = \frac{2\pi}{3}$

As in the previous example, the vector field is not aligned with the mesh. We consider $\nu = 10^3$ and $\nu = 10^4$ and compare BAWR and Galerkin solutions (cfr. Figures 3.14 and 3.15). For $\nu = 10^3$, the error results in the L_2-norm of BAWR and Galerkin, are respectively 0.50822 and 1.7268. The error is computed assuming as the true solution a Galerkin approximation obtained on a $\frac{h}{8}$-grid (cfr. Figure 3.14). For $\nu = 10^4$

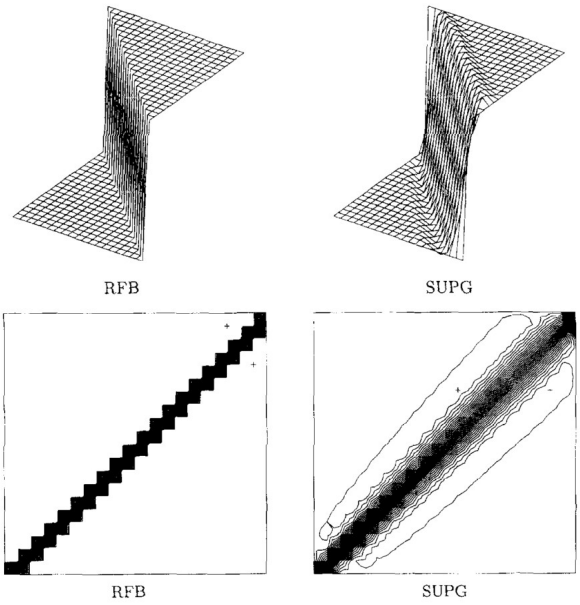

Figure 3.11: *Bubble solution for non homogeneous Dirichlet boundary conditions,* $\vartheta = \frac{\pi}{4}$, $\nu = 10^6$ *(left) and SUPG solution (right). Down: corresponding contour plots.This figure are taken from (14).*

it can be seen also that the BAWR solution on an h-step grid is much more accurate than the Galerkin reference one, computed using $\frac{h}{8}$ as step (cfr. Figure 3.15).

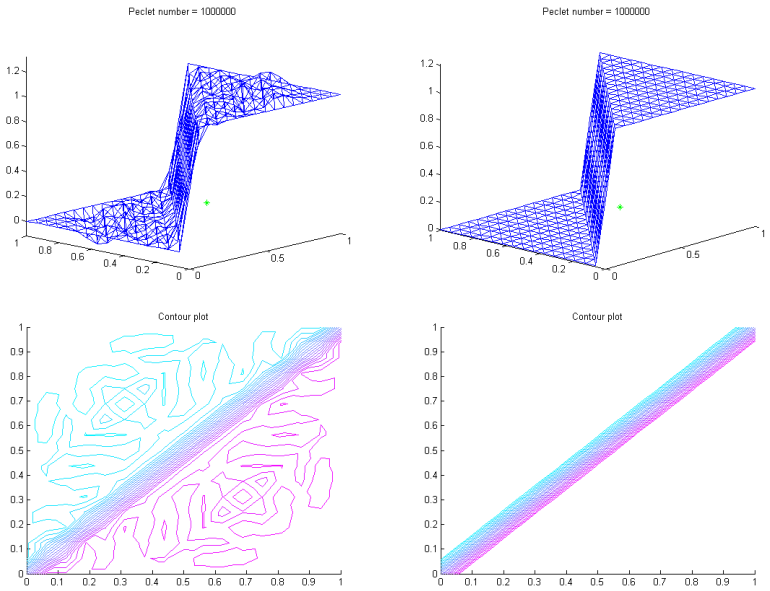

Figure 3.12: *Up: Galerkin solution for non homogeneous Dirichlet boundary conditions, $\vartheta = \frac{\pi}{4}$, $\nu = 10^6$ (left) and BAWR solution (right). Down: corresponding contour plots.*

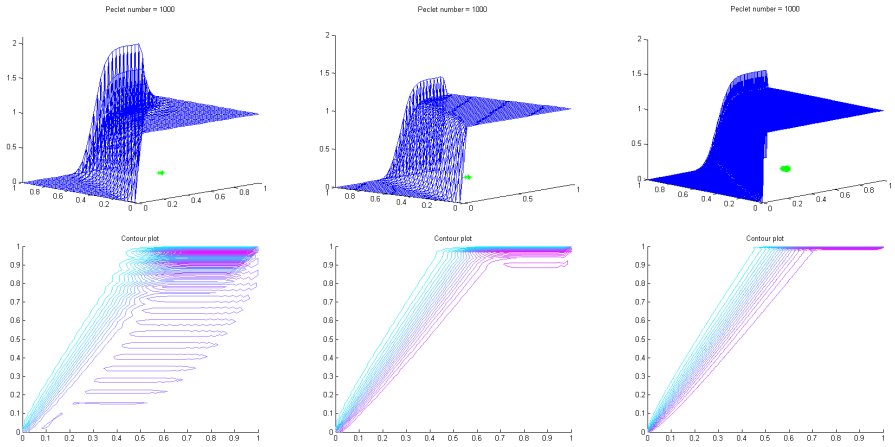

Figure 3.13: *Example $\vartheta = \frac{\pi}{3}$, $\nu = 10^3$. Up: Galerkin solution (left), BAWR solution on the same mesh (center), Galerkin solution on a finer mesh $\frac{h}{4}$ (right). Down: corresponding contour plots.*

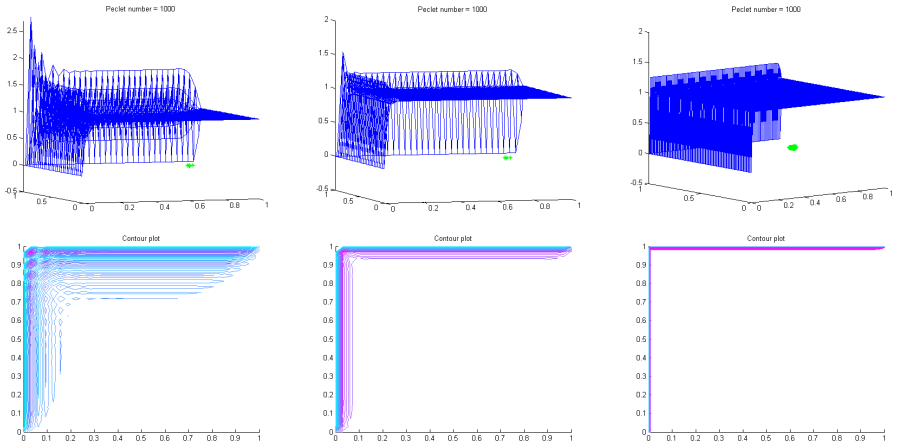

Figure 3.14: *Example $\vartheta = \frac{2\pi}{3}$, $\nu = 10^3$. Up: Galerkin solution (left), BAWR solution on the same mesh (center), Galerkin solution on a finer mesh $\frac{h}{4}$ (right). Down: corresponding contour plots.*

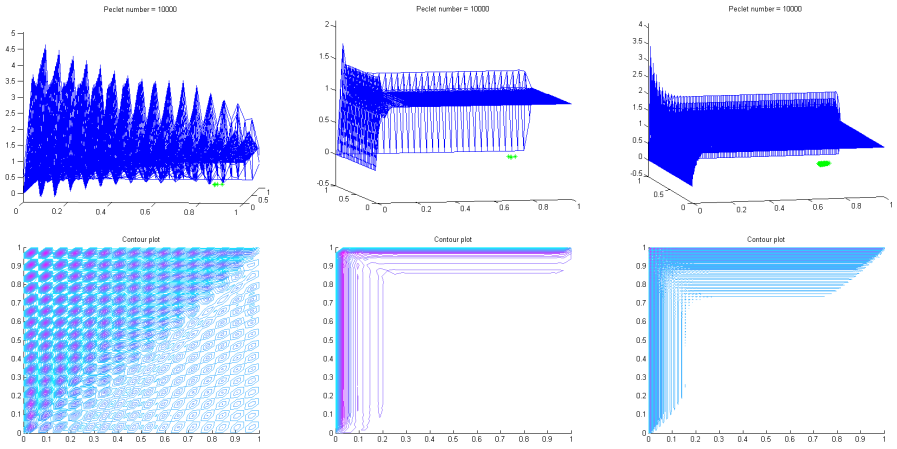

Figure 3.15: *Example $\vartheta = \frac{2\pi}{3}$, $\nu = 10^4$. Up: Galerkin solution (left), BAWR solution on the same mesh (center), Galerkin solution on a finer mesh $\frac{h}{4}$ (right). Down: corresponding contour plots.*

4

Navier Stokes equations

4.1 Navier Stokes equations

Consider a fluid of density ρ which is moving in $\Omega \subset \mathbb{R}^n$, $n \geq 1$, with velocity $\mathbf{u} = \mathbf{u}(t, \mathbf{x})$, $\mathbf{u} = (u_i)_{i=1,\dots,n}$, $t \in [t_0, t_f]$ and denote its pressure with p.

In (23) the mathematical model describing fluid flow motion is derived from the fundamental principles of conservation of mass and momentum: to deepen the physic underlying fluid flow modeling (1) or (5) could be consulted; moreover an introduction of the microscopic one can be found in (58).

4. NAVIER STOKES EQUATIONS

Following (23), the *conservation of mass* can be expressed as follow:

$$\frac{d}{dt}\int_D \rho(t,\mathbf{x})d\omega = -\int_{\partial D} \rho(t,\mathbf{x})\mathbf{u}(t,\mathbf{x})\cdot\mathbf{n}(\mathbf{x})d\gamma,$$

for every fixed $D \subseteq \Omega$: *the rate of change of mass in D equals the amount of fluid flowing into D across ∂D*. Applying the *Green's lemma*

$$0 = \frac{d}{dt}\int_D \rho(t,\mathbf{x})d\omega + \int_D div(\rho(t,\mathbf{x})\mathbf{u}(t,\mathbf{x}))d\omega = \int_D \frac{\partial}{\partial t}\rho(t,\mathbf{x}) + div(\rho(t,\mathbf{x})\mathbf{u}(t,\mathbf{x}))d\omega.$$

Using D arbitrariness, it follows the *conservation equation*:

$$\frac{\partial}{\partial t}\rho + div(\rho\mathbf{u}) = 0 \ in\ \Omega. \tag{4.1}$$

Modeling an *incompressible fluid*, that is a fluid such that any amount of it does not change its volume along the motion, is equivalent to impose the *incompressibility constraint*

$$div\mathbf{u} = 0.$$

Substituting this constraint in (4.1) the following equation holds:

$$\frac{d}{dt}\rho = \frac{\partial}{\partial t}\rho + \nabla\rho\cdot\mathbf{u} = 0 \ in\ \Omega,$$

since $div(\rho\mathbf{u}) = \rho div(\mathbf{u}) + \nabla\rho\cdot\mathbf{u}$.

Consider now the *momentum equation*. First of all observe that $\frac{d}{dt}\mathbf{u}(t,\mathbf{x}) = \left(\frac{d}{dt}u_i(t,\mathbf{x})\right)_{i=1,\dots,n}$ and for every i

$$\frac{d}{dt}u_i(t,\mathbf{x}) = \frac{\partial}{\partial t}u_i + \sum_{j=1}^n \frac{\partial u_i}{\partial x_j}\frac{\partial x_j}{\partial t} = \frac{\partial}{\partial t}u_i + \sum_{j=1}^n \frac{\partial u_i}{\partial x_j}u_j = \frac{\partial}{\partial t}u_i + \nabla u_i\cdot\mathbf{u}.$$

Defining

$$(\mathbf{u}\cdot\nabla)(\bullet) := (\nabla(\bullet_i)\cdot\mathbf{u})_{i=1,\dots,n},$$

$$\frac{d}{dt}\mathbf{u}(t,\mathbf{x}) = \frac{\partial\mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)(\mathbf{u}),$$

represents the *fluid acceleration*, which is nonlinear in $\mathbf{u}$.

Observe that $\frac{d(\bullet)}{dt} = \frac{\partial(\bullet)}{\partial t} + (\mathbf{u}\cdot\nabla)(\bullet)$ is the so called *convective derivative*, and expresses the rate of change of either a scalar quantity (or of each scalar component of a vector quantity) that is "following the fluid".

For a fixed volume $D \subseteq \Omega$, the rate of change of momentum is the product of the mass and the acceleration, i.e.

$$\int_D \rho\left(\frac{\partial\mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)(\mathbf{u})\right)d\omega.$$

This rate must be equal to the sum of forces acting on the fluid.

For a *viscous* fluid, each small volume of fluid D is not only acted on by pressure forces (*normal stresses*), and any external body force $\mathbf{F}$, e.g. gravity, but also by *tangential stresses* (or *shear stresses*). While normal stresses give rise to

$$\int_{\partial D} -p\mathbf{n}d\gamma = \int_{\partial D} -p\mathbb{I}\mathbf{n}d\gamma,$$

where $\mathbb{I}_n$ is the n dimensional *unit diagonal tensor*, the shear stresses act in any direction at different points of ∂D: then a full $n \times n$ tensor $\mathbf{T}$ is needed, and the corresponding force is

$$\int_{\partial D} \mathbf{T}\mathbf{n}d\gamma$$

which is equal to

$$\int_D div\mathbf{T}d\omega,$$

applying the *Green's lemma* and denoting with $div\mathbf{T}$ an n-dimensional vector such that $(div\mathbf{T})_i = \sum_{j=1}^n T_{ij}$.

A *Newtonian fluid* is characterized by the fact that the shear stress tensor is a linear function of the *rate of strain tensor*

$$\mathbf{D} := \frac{1}{2}\left[\nabla\mathbf{u} + (\nabla\mathbf{u})^t\right],$$

where $\nabla\mathbf{u}$ in an $n \times n$ matrix such that $\nabla\mathbf{u}_{ij} = \frac{\partial u_j}{\partial x_i}$. More precisely

$$\mathbf{T} = \mu\mathbf{D} + \left[-p + \lambda Tr(\mathbf{D})\right]\mathbb{I},$$

where $Tr(\mathbf{D}) = \sum_{i=1}^n D_{ii}$ and μ and λ are parameters describing how sticky the fluid is: $\lambda = \zeta - \frac{\mu}{n}$, where μ and ζ are called *shear* and *bulk* viscosity coefficients respectively (cfr. (65) for more details).

For an incompressible Newtonian fluid $Tr(\mathbf{D}) = div\mathbf{u} = 0$, thus

$$\mathbf{T} = \mu\mathbf{D} - p.$$

The *molecular viscosity* μ measures the resistance of the fluid to shearing.

$$\mu\mathbf{D} = \mu\Delta\mathbf{u}.$$

Applying now the Second Law of Motion:

$$\int_D \rho\left(\frac{\partial\mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)(\mathbf{u})\right)d\omega = \int_D -\nabla p + \rho\mathbf{F} + \mu\Delta\mathbf{u}d\omega :$$

thus, for arbitrariness of D, we obtain the *Navier Stokes equations*

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)(\mathbf{u}) &= -\frac{1}{\rho}\nabla p + \mathbf{F} + \nu\Delta\mathbf{u}, \quad in\ \Omega, \\ div\mathbf{u} &= 0, \qquad\qquad\qquad\qquad in\ \Omega, \end{cases} \tag{4.2}$$

where $\nu = \frac{\mu}{\rho}$ is called *kinematic viscosity.*

If U is a reference value for $\mathbf{u}$, e.g. the maximum magnitude of velocity on the inflow, and L is the characteristic length scale for the domain, the relative contributions of convection and diffusion are defined by the *Reynolds number*

$$Re = \frac{UL}{\nu} :$$

if $Re \leq 1$ then the fluid is diffusion dominated and the solution can be shown to be uniquely defined (23), whereas, if $Re > 1$ the fluid is convection dominated. If $Re \to \infty$ we obtain the Euler system. Observe that in general *the larger is Re, the more difficult is the problem to handle.*

A linear simplification of (4.2) is the *Stokes problem*

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} - \nu\Delta\mathbf{u} + \nabla p &= \mathbf{f}, \quad on \quad \Omega \\ div\mathbf{u} &= 0, \quad on \quad \Omega \\ \mathbf{u} &= \mathbf{0}, \quad on \quad \Gamma_d. \end{cases} \tag{4.3}$$

The Stokes problem describes the flow at low Reynolds number of an incompressible fluid: the nonlinear convection term is neglected because it is assumed that the flow is moving with "low-speed", e.g. it is very viscous or tightly confined (e.g. the flow of blood in parts of the human body).

A particular example: Poiseuille flow

As presented in (25), consider the following steady Stokes problem, defined in $\Omega = [0,1]\times[0,1] \subset \mathbb{R}^2$, representing steady horizontal flow in a channel driven by a pressure difference between the two ends:

$$\begin{cases} -\nu\Delta\mathbf{u} + \nabla p &= \mathbf{0}, \quad in \quad \Omega \\ div\mathbf{u} &= 0, \quad in \quad \Omega \\ \mathbf{u} &= \mathbf{g}, \quad on\partial\Omega, \end{cases} \tag{4.4}$$

$\mathbf{g} \in \mathbf{H}^{\frac{1}{2}}(\partial\Omega)$. Suppose moreover that $\mathbf{g} = \begin{pmatrix} g1 \\ g2 \end{pmatrix}$,

$$g_2 = 0,\ g_1 = \begin{cases} 0 & for\ y = 0,\ y = 1,\ x \in (0,1) \\ y - y^2 & for\ x = 0,\ x = 1,\ y \in (0,1). \end{cases}$$

It is possible to show (25) that an *analytical* solution of (4.5) is the following:

$$\mathbf{u} = \begin{pmatrix} y - y^2 \\ 0 \end{pmatrix}, \; p = -2\nu x + \nu,$$

also known as *bidimensional Poiseuille flow*.

Observe that this is also the analytical solution of Navier-Stokes problem

$$\begin{cases} -\nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p &= \mathbf{0}, \quad in \quad \Omega \\ div\mathbf{u} &= 0, \quad in \quad \Omega \\ \mathbf{u} &= \mathbf{g}_1, \quad on\partial\Omega, \end{cases} \tag{4.5}$$

for $\mathbf{g}_1 = \mathbf{g}$ and also for

$$\mathbf{g}_1 = \begin{cases} \mathbf{0} & on \; x \in [0,1], \; y = 0, \; y = 1 \\ p\mathbf{n} - \nu\nabla u\cdot\mathbf{n} = -2\nu\mathbf{n} & on \; x = 0, \; y \in [0,1] \\ p\mathbf{n} - \nu\nabla u\cdot\mathbf{n} = \mathbf{0} & on \; x = 1, \; y \in [0,1]. \end{cases}$$

4.2 Variational formulation and FE discretization

4.2.1 Steady Stokes problem

Following (65), consider the constrained Stokes problem:

$$\begin{cases} a_0\mathbf{u} - \nu\Delta\mathbf{u} + \nabla p &= \mathbf{f}, \quad on \quad \Omega \\ div\mathbf{u} &= 0, \quad on \quad \Omega \\ \mathbf{u} &= \mathbf{0}, \quad on \quad \Gamma_d \\ \nu\frac{\partial\mathbf{u}}{\partial\mathbf{n}} - p\mathbf{n} &= \mathbf{0}, \quad on \quad \Gamma_n \end{cases} \tag{4.6}$$

where $a_0 \geq 0$ and $\nu > 0$ are real constant values. Suppose moreover that $\mathbf{f} \in \mathbf{L}^2(\Omega) := (L^2(\Omega))^n$ (*body force* acting on the fluid) and for simplicity that $\Gamma_n = \emptyset$. Observe that in this case, if we are dealing with non-homogeneous Dirichlet boundary conditions ($\mathbf{u} = \mathbf{g}$, on Γ_d), integrating the incompressibility constraint, we obtain $\int_\Omega div\mathbf{u}d\omega = \int_{\Gamma_d}\mathbf{g}\cdot\mathbf{n}d\sigma$, thus $\mathbf{g}$ must satisfy a compatibility condition.

The fact that the incompressibility constraint does not involve the pressure variable makes the construction of finite element approximations problematic: the discrete spaces used to approximate the velocity and pressure fields cannot be chosen independently of one another (*infsup* condition).

It is possible to derive two variational formulations of (4.6), which lead to two different strategies to solve it numerically. We introduce both of them for completeness, remanding to (65) for more details.

4.2.1.1 Variational formulation of (4.6): *constrained formulation*

Define the Hilbert space $X = \mathbf{V} = \mathbf{H}_0^1(\Omega) := (H_0^1(\Omega))^n$ and $\mathbf{V}_{div} := \{\mathbf{v} \in \mathbf{V} \ s.t. \ div\mathbf{v} = 0\}$: it can be demonstrated (65) that it is a closed subset of $\mathbf{V}$ and it is an Hilbert space for the norm $\|\mathbf{v}\| := \|\nabla\mathbf{v}\|_{L^2(\Omega)}$.

Observe that multiplying the first equation of (4.6) by a test function $\mathbf{v} \in \mathbf{V}$ and applying the *Green's lemma* we obtain

$$a_0(\mathbf{u}, \mathbf{v}) + \nu(\nabla\mathbf{u}, \nabla\mathbf{v}) - (p, div\mathbf{v}) = (\mathbf{f}, \mathbf{v}).$$

and then

$$a_0(\mathbf{u}, \mathbf{v}) + \nu(\nabla\mathbf{u}, \nabla\mathbf{v}) = (\mathbf{f}, \mathbf{v}), \ \forall\mathbf{v} \in \mathbf{V}_{div}.$$

Consider now the bilinear form $a(\mathbf{w}, \mathbf{v}) := a_0(\mathbf{w}, \mathbf{v}) + \nu(\nabla\mathbf{w}, \nabla\mathbf{v}), \ \forall\mathbf{w}, \mathbf{v} \in \mathbf{V}$. It can be proved that it is coercive over $\mathbf{V}_{div} \times \mathbf{V}_{div}$. Moreover $F(\mathbf{v}) := (\mathbf{f}, \mathbf{v})$ is linear and continuous over $\mathbf{V}_{div}$. Then, applying Theorem A.2.1, the problem

$$\text{find } \mathbf{u} \in \mathbf{V}_{div} \text{ s.t. } a(\mathbf{u}, \mathbf{v}) = F(\mathbf{v}), \ \forall\mathbf{v} \in \mathbf{V}_{div} \tag{4.7}$$

admits a unique stable solution.

Theorem 4.2.1 *Let Ω be a bounded domain in $\mathbb{R}^n$, with a Lipschitz continuous boundary, and for each $\boldsymbol{f} \in \boldsymbol{L}^2(\Omega)$ let $\boldsymbol{u}$ be the solution of (4.7). Then there exists a function $p \in L^2(\Omega)$, which is unique up to an additive constant, s.t.*

$$a(\boldsymbol{u}, \boldsymbol{v}) - (p, div\boldsymbol{v}) = (\boldsymbol{f}, \boldsymbol{v}), \ \forall\boldsymbol{v} \in \boldsymbol{V}. \tag{4.8}$$

For a proof cfr. (32, 65). This Theorem is meaningful because it tells us that the pressure is well defined if it is known the velocity field of the problem. Moreover the rule of the pressure field in the weak formulation is substantially to force the velocity field to be solenoidal.

Observe that from (4.7) we obtain $\mathbf{u}$, then p can be derived using (4.8).

Consider now the Galerkin approximation of (4.7) (cfr. section A.3.1):

$$\text{find } \mathbf{u}_h \in \mathbf{V}_{div,h} \text{ s.t. } a(\mathbf{u}_h, \mathbf{v}_h) = F(\mathbf{v}_h), \ \forall\mathbf{v}_h \in \mathbf{V}_{div,h}, \tag{4.9}$$

where $\{\mathbf{V}_{div,h}\}_h$ is a family of finite dimensional subspaces of $\mathbf{V}_{div}$ satisfying the consistency assumption

$$\forall\mathbf{v} \in \mathbf{V}_{div} : \inf_{\mathbf{v}_h \in \mathbf{V}_{div,h}} \|\mathbf{v} - \mathbf{v}_h\| \to 0, \text{ as } h \to 0.$$

Applying Theorem A.3.1 we deduce existence and uniqueness of (4.9)'s solution, which is also stable and convergent.

In practice this formulation is scarcely used because it is difficult to find approximants $\mathbf{V}_{div,h}$ of $\mathbf{V}_{div}$ s.t. the convergence estimate of *Cea Lemma* (cfr. Theorem A.3.1) is useful. Moreover it could be very difficult to construct a basis for $\mathbf{V}_{div,h}$. A possible solution is to substitute it with a space Z_h which is not a subspace of $\mathbf{V}_{div}$ leading to a *non-conforming approximation* to (4.7) (65).

4.2.1.2 Variational formulation of (4.6): *mixed formulation*

Consider another Hilbert space $M = Q = L^2_0(\Omega)$ denoting the space of functions of $L^2(\Omega)$ with zero mean and the bilinear form $b(\mathbf{v}, q) := -(q, div\mathbf{v})$, $\mathbf{v} \in \mathbf{V}$ and $q \in Q$.

Multiply now the first equation of (4.6) by a test function $\mathbf{v} \in \mathbf{V}$ and the second by $q \in Q$ and integrate on Ω obtaining the *weak formulation of (4.6)*: find $\mathbf{u} \in \mathbf{V}$ and $p \in Q$ s.t.

$$\begin{cases} a(\mathbf{u}, \mathbf{v}) + b(\mathbf{v}, p) &= (\mathbf{f}, \mathbf{v}), \quad \forall \mathbf{v} \in \mathbf{V} \\ b(\mathbf{u}, q) &= 0, \qquad \forall q \in Q. \end{cases} \tag{4.10}$$

Observe that from Theorem 4.2.1 we deduce that (4.10) has a unique solution (65). Finally it remains to prove that the solution of (4.10) is also a solution of (4.6), supposing that the last one exists: this can be done using a classical density argument (65).

Consider now two families of finite dimensional subspaces $\mathbf{V}_h \subset \mathbf{V}$ and $Q_h \subset Q$ and approximate (4.10): find $\mathbf{u}_h \in \mathbf{V}_h$ and $p_h \in Q_h$ s.t.

$$\begin{cases} a(\mathbf{u}_h, \mathbf{v}_h) + b(\mathbf{v}_h, p_h) &= (\mathbf{f}, \mathbf{v}_h), \quad \forall \mathbf{v}_h \in \mathbf{V}_h \\ b(\mathbf{u}_h, q_h) &= 0, \qquad \forall q_h \in Q_h. \end{cases} \tag{4.11}$$

Defining the *space of discretely divergence-free functions*

$$\mathbf{Z}_h := \{\mathbf{v}_h \in \mathbf{V}_h \ s.t. \ (q_h, div\mathbf{v}_h) = 0, \ \forall q_h \in Q_h\},$$

the bilinear form a is coercive in $\mathbf{Z}_h$, i.e. there exists $C > 0$ s.t.

$$a_0 \|v_h\|^2_{\mathbf{L}^2} + \nu \|\nabla v_h\|^2_{\mathbf{L}^2} \geq \|v_h\|^2_{\mathbf{V}}, \ \forall \mathbf{v}_h \in \mathbf{Z}_h.$$

Moreover a and b are continuous over $\mathbf{V} \times \mathbf{V}$ and $\mathbf{V} \times Q$ respectively, i.e. there exist $\gamma > 0$ and $\delta > 0$ s.t.

$$|a(\mathbf{w}, \mathbf{v})| \leq \gamma \|\mathbf{w}\| \|\mathbf{v}\|, \ \forall \mathbf{v}, \mathbf{w} \in \mathbf{V}$$

$$|b(\mathbf{v}, q)| \leq \delta \|\mathbf{v}\|_{\mathbf{V}} \|q\|_{L^2}, \ \forall \mathbf{v}, \mathbf{w} \in \mathbf{V}.$$

Suppose moreover that $\mathbf{V}_h$ and Q_h satisfy the *compatibility (or inf-sup, or LBB) condition* (A.16), i.e.

$$\text{there exists } \beta > 0 \text{ s.t. } \forall q_h \in Q_h \ \exists \mathbf{v}_h \in \mathbf{V}_h, \mathbf{v}_h \neq \mathbf{0}\text{: } b(\mathbf{v}_h, q_h) \geq \beta \left\|\mathbf{v}_h\right\| \left\|q_h\right\|. \quad (4.12)$$

Observe that this condition states that the space of discrete velocities $\mathbf{V}_h$ is *sufficiently rich* compared with the one of discrete pressures Q_h. Moreover, as introduced in section A.4, it ensures that no spurious pressure mode is allowed, i.e. there exists no $p_h^* \in Q_h,\ p_h^* \neq 0$ s.t.

$$b(\mathbf{v}_h, p_h^*) = 0,\ \forall \mathbf{v}_h \in \mathbf{V}_h.$$

Avoiding spurious modes is important because they cause spurious oscillation in the computed pressure, destroying the simulation of real dynamics.

Under these assumptions, Theorem A.4.2 yields existence and uniqueness for the solution of (4.11) whereas Theorem A.4.3 guarantees convergence (for Stokes problem σ is the null operator, $\eta_h = p_h$, $\mu_h = q_h$, $X_h = \mathbf{V}_h$ and $M_h = Q_h$):

$$\left\|\mathbf{u} - \mathbf{u}_h\right\|_1 \leq \left(1 + \frac{\gamma}{C}\right) \inf_{\mathbf{v}_h \in Z_h} \left\|\mathbf{u} - \mathbf{v}_h\right\|_1 + \frac{\delta}{C} \inf_{q_h \in Q_h} \left\|p - q_h\right\|_0$$

$$\left\|p - p_h\right\|_0 \leq \frac{\gamma}{\beta}\left(1 + \frac{\gamma}{C}\right) \inf_{\mathbf{v}_h \in Z_h} \left\|\mathbf{u} - \mathbf{v}_h\right\|_1 + \left(1 + \frac{\delta}{\beta} + \frac{\gamma\delta}{C\beta}\right) \inf_{q_h \in Q_h} \left\|p - q_h\right\|_0.$$

Since β is independent of h, the solution is stable and convergence is optimal.

Under some assumptions on the approximating spaces (32), it is possible to improve the convergence estimate in the following way:

$$\left|\mathbf{u} - \mathbf{u}_h\right|_1 + \left\|p - p_h\right\|_0 \leq C h^m \left(\left\|\mathbf{u}\right\|_{m+1} + \left\|p\right\|_m\right),$$

where m depends upon the regularity of $\mathbf{u}$ and p. Moreover if Stokes problem is *regular*, i.e. if $(\mathbf{u}, p) \to -\nu\Delta\mathbf{u} + \nabla p$ is an isomorphism, $\mathbf{u} \in \mathbf{H}^{m+1}(\Omega)$, $p \in H^m(\Omega) \cap L_0^2(\Omega)$, the following error bound holds:

$$\left\|\mathbf{u} - \mathbf{u}_h\right\|_0 + \leq C h^{m+1} \left(\left\|\mathbf{u}\right\|_{m+1} + \left\|p\right\|_m\right).$$

Also a posteriori estimators can be derived: cfr. (23) and references therein.

Finally observe that if the finite dimensional subspaces do not satisfy the *inf-sup condition*, anyway the velocity field can be obtained in a stable an convergent way, because the corresponding estimates given by Theorems A.4.2 and A.4.3 (with $\|\sigma\| = 0$) are independent from β. Thus only the pressure field are affected by spurious modes.

If we define the operator $\mathcal{B}_h :\ V_h \to Q_h^{'}$ such that

$$\langle \mathcal{B}_h v_h, q_h \rangle = b(v_h, q_h),$$

for all $v_h \in V_h$ and $q_h \in Q_h$, the *inf-sup* condition is not satisfied iff $\mathcal{B}_h^* : \ Q_h \to V_h^{'}$ is not injective (i.e. iff the corresponding matrix B^T has not full column rank, cfr. section 4.2.1.3). Equivalently the *inf-sup* condition is not satisfied iff $\mathcal{B}_h$ is not surjective (24).

To guarantee a good approximation of the pressure field, there are different possibilities: first the choice of discrete spaces that satisfy the *inf-sup condition*; an alternative is to *filter* the spurious modes out of the computed pressure (65). Finally it is possible to *stabilize* (4.11) relaxing the incomprimibility constraint on $\mathbf{u}_h$: in fact in this case stability and convergence results can be proved, regardless of the *inf-sup* condition.

4.2.1.3 Algebraic Formulation

Let $\left\{\boldsymbol{\phi}_j\right\}_{j=1,\ldots,N_h^u}$ and $\{\psi_l\}_{l=1,\ldots,N_h^p}$ be bases of $\mathbf{V}_h$ and Q_h respectively. Consider

$$\mathbf{u}_h(\mathbf{x}) = \sum_{j=1}^{N_h^u} u_j \boldsymbol{\phi}_j(\mathbf{x}), \ p_h(\mathbf{x}) = \sum_{l=1}^{N_h^p} p_l \psi_l(\mathbf{x}),$$

then the linear system associated with (4.11) is

$$\begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{0} \end{pmatrix}, \tag{4.13}$$

where

$$A_{ij} = a(\boldsymbol{\phi}_j, \boldsymbol{\phi}_i), \ B_{li} = b(\boldsymbol{\phi}_i, \psi_l), \ f_i = F(\boldsymbol{\phi}_i).$$

Observe that A (*vector-Laplacian* matrix) is an $N_h^u \times N_h^u$ symmetric and positive definite matrix (which corresponds to a_h's coerciveness), while B (*divergence* matrix) is $N_h^p \times N_h^u$. It is important to note that the *compatibility condition* (A.16) on b_h holds iff $ker B^T = \mathbf{0}$ (cfr. e.g. (25)). In this case the global matrix $\begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix}$ is *non singular*. If $N_h^p > N_h^u$ then $\begin{pmatrix} A & B^T \\ B & 0 \end{pmatrix}$ is rank deficient by at least $N_h^p - N_h^u$, since B has rank less or equal to N_h^u (cfr. (23)). This is another proof of the impossibility of choosing too high dimensional approximation for pressure, compared to the velocity one.

In (23) some properties of the system (4.13) are discussed. Observe that the system matrix in (4.13) is neither positive nor definite, although is symmetric. The simplest strategy is to solve the system using a direct method, for example the LU factorization. An alternative is to use *iterative methods* or *penalty (or artificial compressibility) techniques* which replace this system with the perturbed one

$$\begin{pmatrix} A & B^T \\ B & -\epsilon M \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{p} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{0} \end{pmatrix}, \tag{4.14}$$

where $\epsilon > 0$ is the *penalty coefficient* and M is the *pressure mass matrix*, $M_{ij} = a(\psi_j, \psi_i)$. Observe that it corresponds to solve the following problem

$$\left\{ \begin{array}{rcll} a_0\mathbf{u}^\epsilon - \nu\Delta\mathbf{u}^\epsilon + \nabla p^\epsilon &=& \mathbf{f}, & on \quad \Omega \\ div\mathbf{u}^\epsilon &=& -\epsilon p^\epsilon, & on \quad \Omega \\ \mathbf{u} &=& \mathbf{0}, & on \quad \Gamma_d. \end{array} \right. \tag{4.15}$$

Another strategy is the *pressure-matrix method*, which is based on the elimination procedure

$$\mathbf{u} = A^{-1}(\mathbf{f} - B^T\mathbf{p})$$

$$R\mathbf{p} = BA^{-1}\mathbf{f}.$$

the idea is to generate an independent linear system for p after elimination of $\mathbf{u}$.

The *Uzawa method*, given p^0, consists in solving for any $k > 0$ the continuous problem

$$a_0\mathbf{u}^{k+1} - \nu\Delta\mathbf{u}^{k+1} = \mathbf{f} - \nabla p^k,$$

with $\mathbf{u}^{k+1} = \mathbf{0}$ on $\partial\Omega$, and then

$$p^{k+1} - p^k = -\rho div\mathbf{u}^{k+1},$$

where $0 < \rho < 2\nu$ is an *acceleration parameter*. The corresponding discrete algebraic formulation reads

$$\left\{ \begin{array}{rcl} A\mathbf{u}^{k+1} &=& \mathbf{f} - B^T\mathbf{p}^k \\ P(\mathbf{p}^{k+1} - \mathbf{p}^k) &=& \rho B\mathbf{u}^{k+1}, \end{array} \right.$$

where P is a suitable preconditioner for R. For convergence results cfr. (65) and references therein.

Finally the *Augmented-Lagrangian method*, given p^0, consists in solving for any $k \geq 0$

$$a_0\mathbf{u}^{k+1} - \nu\Delta\mathbf{u}^{k+1} + \nabla p^{k+1} = \mathbf{f},$$

$$p^{k+1} - p^k = -\rho div\mathbf{u}^{k+1}.$$

The corresponding algebraic system is

$$\left\{ \begin{array}{rcl} A\mathbf{u}^{k+1} + B^T\mathbf{p}^{k+1} &=& \mathbf{f} \\ J(\mathbf{p}^{k+1} - \mathbf{p}^k) &=& \rho B\mathbf{u}^{k+1}, \end{array} \right.$$

where $J_{lm} = (\psi_l, \psi_m)$, which is non singular. Thus $(A + \rho B^T J^{-1} B)\mathbf{u}^{k+1} = \mathbf{f} - B^T\mathbf{p}^k$, which is a symmetric positive definite system for $\mathbf{u}^{k+1}$. Then $\mathbf{p}^{k+1} = \rho J^{-1} B\mathbf{u}^{k+1} + \mathbf{p}^k$.

4.2.2 Unsteady Navier-Stokes equation

Consider the *unsteady Navier-Stokes* model:

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} - \nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)(\mathbf{u}) + \nabla p &= \mathbf{f}, & in\ Q_T := (0,T)\times\Omega, \\ div\mathbf{u} &= 0, & in\ Q_T, \\ \mathbf{u} &= \mathbf{0} & on\ \Sigma_T := (0,T)\times\partial\Omega \\ \mathbf{u}\,|_{t=0} &= \mathbf{u}_0, & on\ \Omega, \end{cases} \tag{4.16}$$

where $\mathbf{f} = \mathbf{f}(t,x)$ and $\mathbf{u}_0 = \mathbf{u}_0(x)$ are given data and $a_0 = 0$.

The corresponding weak formulation is the following one: given $\mathbf{f} \in L^2(0,T;H_{div})$, $\mathbf{u}_0 \in H_{div}$, find $\mathbf{u} \in L^2(0,T;V_{div}) \cap L^\infty(0,T;H_{div})$ s.t.

$$\begin{cases} \frac{d(\mathbf{u}(t),\mathbf{v})}{dt} + a(\mathbf{u}(t),\mathbf{v}) + c(\mathbf{u}(t);\mathbf{u}(t),\mathbf{v}) = (\mathbf{f}(t),\mathbf{v}), & \forall \mathbf{v} \in \mathbf{V}_{div}, \\ \mathbf{u}(0) = \mathbf{u}_0, \end{cases} \tag{4.17}$$

where the trilinear form c is such that $c(\mathbf{u};\mathbf{z},\mathbf{v}) := \int_\Omega ((\mathbf{u}\cdot\nabla)\mathbf{z})\cdot\mathbf{v}$.

The *existence* of a solution of this problem has been proven by Leray (44) and Hopf (38). *Uniqueness* is still an open problem in the three-dimensional case, whereas for $n = 2$ the solution $\mathbf{u}$ has been shown to belong to $C^0([0,T];H_{div})$ and to be unique (43, 56).

Moreover any solution of (4.17) satisfies the following energy estimate

$$\sup_{t\in(0,T)} \|u(t)\|_0^2 + \nu \int_0^T \|\nabla u(t)\|_0^2 \le \|u_0\|_0^2 + \frac{C_\Omega}{\nu}\int_0^T \|f(t)\|_0^2,$$

where C_Ω is the constant of the Poincarè inequality.

The alternative weak formulation is find $\mathbf{u}(t) \in \mathbf{V}$ and $p(t) \in Q$ such that for a.e. $t \in (0,T)$

$$\begin{cases} \frac{d(\mathbf{u}(t),\mathbf{v})}{dt} + a(\mathbf{u}(t),\mathbf{v}) + c(\mathbf{u}(t);\mathbf{u}(t),\mathbf{v}) + b(\mathbf{v},p) &= (\mathbf{f}(t),\mathbf{v}), & \forall \mathbf{v} \in \mathbf{V} \\ b(\mathbf{u}(t),q) &= 0, & \forall q \in Q \\ \mathbf{u}(0) &= \mathbf{u}_0. \end{cases} \tag{4.18}$$

A complete *stability analysis* of Navier Stokes equations could be found in (69) and (23).

In particular in the 2D context it holds (69):

Theorem 4.2.2 *Let $\boldsymbol{f}$ and u_0 belong to $\boldsymbol{H}_{div}$, then there exists a unique solution of (4.18) such that $\boldsymbol{u} \in \mathcal{C}([0,T];\boldsymbol{H}_{div}) \cap L^2(0,T;V_{div})$, for all $T > 0$. Moreover $\boldsymbol{u}$ is analytic in $t > 0$ with values in $\boldsymbol{H}^2(\Omega) \cap \boldsymbol{V}_{div}$ and $\boldsymbol{u}_0 \mapsto \boldsymbol{u}(t)$ is continuous from $\boldsymbol{H}_{div}$ to $\boldsymbol{H}^2(\Omega) \cap \boldsymbol{V}_{div}$. Finally if $\boldsymbol{u}_0 \in \boldsymbol{V}_{div}$, then $\boldsymbol{u} \in \mathcal{C}([0,T];\boldsymbol{H}_{div}) \cap L^2(0,T;\boldsymbol{H}^2(\Omega) \cap \boldsymbol{V}_{div})$.*

4.2.2.1 Space discretization of (4.17) and (4.18)

Consider (4.17) and choose a finite dimensional subspace of V_{div}, $V_{div,h}$: for each $t \in [0,T]$ find $\mathbf{u}_h(t,\cdot) \in V_{div,h}$ s.t.

$$\begin{cases} \frac{d(\mathbf{u}_h(t),\mathbf{v}_h)}{dt} + a(\mathbf{u}_h(t),\mathbf{v}_h) + c(\mathbf{u}_h(t);\mathbf{u}_h(t),\mathbf{v}_h) &= (\mathbf{f}(t),\mathbf{v}_h), \quad \forall \mathbf{v}_h \in \mathbf{V}_{div,h},\ t \in (0,T) \\ \mathbf{u}_h(0) &= \mathbf{u}_{0,h} \quad , \end{cases} \tag{4.19}$$

where $\mathbf{u}_{0,h} \in V_{div,h}$ is an approximation to the initial data $\mathbf{u}_0$.

To approximate (4.18) choose $V_h \subset V$ and $Q_h \subset Q$: thus for each $t \in [0,T]$ find $\mathbf{u}_h(t,\cdot) \in V_h$ and $p_h(t,\cdot) \in Q_h$ s.t.

$$\begin{cases} \frac{d(\mathbf{u}_h(t),\mathbf{v}_h)}{dt} + a(\mathbf{u}_h(t),\mathbf{v}_h) + c(\mathbf{u}_h(t);\mathbf{u}_h(t),\mathbf{v}_h) + b(\mathbf{v}_h,p_h(t)) &= (\mathbf{f}(t),\mathbf{v}_h), \quad \forall \mathbf{v}_h \in \mathbf{V}_h,\ t \in (0,T) \\ b(\mathbf{u}_h(t),q_h) &= 0, \quad \forall q_h \in Q_h,\ t \in (0,T) \\ \mathbf{u}_h(0) &= \mathbf{u}_{0,h} \quad , \end{cases} \tag{4.20}$$

$\mathbf{u}_{0,h} \in V_h$.

If V_h and Q_h satisfy the *inf-sup condition*, suppose moreover that for all $\mathbf{v} \in V$ and $q \in Q$

$$\inf_{\mathbf{v}_h \in V_h} \|\mathbf{v} - \mathbf{v}_h\|_1 + \inf_{q_h \in Q_h} \|q - q_h\|_0 = O(h).$$

Thus the following error estimate holds

$$\|\mathbf{u}(t) - \mathbf{u}_h(t)\|_0 \leq C_1(t)h^2,$$

$$\|p(t) - p_h(t)\|_0 \leq C_2(t)h,$$

$C_1(t) \leq Ke^{KT}$ and $C_2(t) \leq K\tau(t)^{-\frac{1}{2}}e^{KT}$, $\tau(t) := \min(t,1)$ (65). The estimate can be improved assuming more regularity on boundary and initial data (65):

$$\|\mathbf{u}(t) - \mathbf{u}_h(t)\|_0 \leq C_1(t)h^k,$$

$$\|p(t) - p_h(t)\|_0 \leq C_2(t)h^{k-1},$$

$k = 2,\ldots,5$. In numerical tests, in this thesis we will use P2-P1 approximation, as explained in section 4.3: thus the above error estimates hold with $k = 3$.

As for Stokes problem, the algebraic formulation of (4.20) can be derived. Let $\{\boldsymbol{\varphi}_j\}_{j=1,\ldots,N_h^u}$ and $\{\psi_l\}_{l=1,\ldots,N_h^p}$ be basis of V_h and Q_h respectively. Thus $\mathbf{u}_h(t,x) = \sum_{j=1}^{N_h^u} u_j(t)\boldsymbol{\varphi}_j(x)$, $p_h(t,x) = \sum_{l=1}^{N_h^p} p_l(t)\psi_l(x)$. The system of differential algebraic equations (DAE) is the following:

$$\begin{cases} M\frac{d\mathbf{u}(t)}{dt} + A\mathbf{u}(t) + C(\mathbf{u}(t))\mathbf{u}(t) + B^T\mathbf{p}(t) &= \mathbf{f}(t), \quad t \in (0,T) \\ B\mathbf{u}(t) &= \mathbf{0}, \quad t \in (0,T) \\ \mathbf{u}(0) &= \mathbf{u}_0, \end{cases} \tag{4.21}$$

defining $M_{ij} := (\boldsymbol{\varphi}_i, \boldsymbol{\varphi}_j)$, $A_{ij} := a(\boldsymbol{\varphi}_j, \boldsymbol{\varphi}_i)$, $C(\mathbf{w})_{ij} := \sum_{m=1}^{N_h^u} w_m c(\boldsymbol{\varphi}_m; \boldsymbol{\varphi}_j, \boldsymbol{\varphi}_i)$, $B_{li} := b(\boldsymbol{\varphi}_i, \psi_l)$ and $f_i(t) := (\mathbf{f}(t), \boldsymbol{\varphi}_i)$.

4.2.2.2 Time discretization of (4.21)

ϑ-methods

Defining $t_{n+1} = (n+1)\Delta t$, $n = 1, \ldots, N-1$ a discretization of $[0, T]$ we obtain

$$\begin{cases} M\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\Delta t} + A\mathbf{u}_\vartheta^{n+1} + C(\mathbf{u}_\vartheta^{n+1})\mathbf{u}_\vartheta^{n+1} + B^T\mathbf{p}_\vartheta^{n+1} &= \mathbf{f}(\vartheta t_{n+1} + (1-\vartheta)t_n) \\ B\mathbf{u}^{n+1} &= \mathbf{0}, \end{cases} \tag{4.22}$$

with $\mathbf{v}_\vartheta^{n+1} := \vartheta\mathbf{v}^{n+1} + (1-\vartheta)\mathbf{v}^n$.

Observe that when $\vartheta = 1$ (*backward Euler*) at each time-level we obtain the following nonlinear system

$$\begin{cases} \left(A + \frac{M}{\Delta t}\right)\mathbf{u} + C(\mathbf{u})\mathbf{u} + B^T\mathbf{p} &= \mathbf{G} \\ B\mathbf{u} &= \mathbf{0}, \end{cases} \tag{4.23}$$

where $\mathbf{G}$ is known. Observe that it requires the solution of a nonlinear system at *every* time step. A possible solution is to consider *semi-implicit* methods, linearizing (4.22), e.g.

$$\begin{cases} M\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\Delta t} + A\mathbf{u}_\vartheta^{n+1} + C(\mathbf{u}_\vartheta^{n})\mathbf{u}_\vartheta^{n+1} + B^T\mathbf{p}_\vartheta^{n+1} &= \mathbf{f}(\vartheta t_{n+1} + (1-\vartheta)t_n) \\ B\mathbf{u}^{n+1} &= \mathbf{0}. \end{cases} \tag{4.24}$$

Other algorithms can be obtained using second order accurate methods, instead of first order (for details cfr. e.g. (65)).

Another approach to solve (4.16) is the *fractional-step (or projection) method*, proposed by Chorin and Temam (and presented in (64)). It is based upon an *operator-splitting* technique and on a subdivision of the time interval $[t_n, t_{n+1}]$, considering an intermediate time $\tilde{t}_n$ (e.g. $\tilde{t}_n = t_{n+\frac{1}{2}}$).

1. Solve for $\tilde{\mathbf{u}}^{n+1}$

$$\begin{cases} \frac{\tilde{\mathbf{u}}^{n+1}-\mathbf{u}^n}{\Delta t} - \nu\Delta\tilde{\mathbf{u}}^{n+1} + (\mathbf{u}^* \cdot \nabla)(\mathbf{u}^{**}) + \nabla p &= \mathbf{f}, \quad \textit{in } \Omega, \\ \tilde{\mathbf{u}}^{n+1} &= \mathbf{0}, \quad \textit{on } \partial\Omega \end{cases} \tag{4.25}$$

   where $\mathbf{u}^*$ and $\mathbf{u}^{**}$ could be both $\tilde{\mathbf{u}}^{n+1}$ and $\mathbf{u}^n$ (explicit, implicit or semi-implicit treatment of the convection term).

2. Solve for $\mathbf{u}^{n+1}$

$$\begin{cases} \frac{\mathbf{u}^{n+1}-\tilde{\mathbf{u}}^{n+1}}{\Delta t} + \nabla p^{n+1} &= \mathbf{0}, \quad in\ \Omega, \\ div\mathbf{u}^{n+1} &= 0, \quad in\ \Omega, \\ \mathbf{u}^{n+1}\cdot\mathbf{n} &= 0, \quad on\ \partial\Omega. \end{cases} \tag{4.26}$$

Applying the divergence operator to the first equation we can rewrite it with the equivalent system

$$\begin{cases} -\Delta p^{n+1} &= -div\frac{\tilde{\mathbf{u}}^{n+1}}{\Delta t}, \quad in\ \Omega, \\ div\mathbf{u}^{n+1} &= 0, \quad in\ \Omega, \\ \frac{\partial p^{n+1}}{\partial n} &= 0, \quad on\ \partial\Omega. \end{cases} \tag{4.27}$$

The last system gives p^{n+1}, which can be used to solve (4.26): $\mathbf{u}^{n+1} = \tilde{\mathbf{u}}^{n+1} - \Delta t\nabla p^{n+1}$ in Ω. Observe moreover that in (4.26) we are imposing a condition only on the normal component of $\mathbf{u}^{n+1}$: this causes a *splitting error*, due to the free tangential component.

Other methods are described in (65), while in (70) a scheme (*Navier-Stokes tree*) is presented, summarizing the most used techniques for solving the incompressible unstationary Navier Stokes equation (4.16).

4.3 Numerical simulation of Navier Stokes equation

In this section it is briefly described the numerical discretization of Navier Stokes equation adopted in the simulations of this thesis.

Consider the *nondimensional* version of the Navier-Stokes model problem (4.16):

$$\begin{cases} \frac{\partial\mathbf{u}}{\partial t} - \frac{1}{Re}\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)(\mathbf{u}) + \nabla p &= \mathbf{f}, \quad in\ Q_T := (0,T)\times\Omega, \\ div\mathbf{u} &= 0, \quad in\ Q_T, \\ \mathbf{u} &= \mathbf{0} \quad on\ \Sigma_T := (0,T)\times\partial\Omega \\ \mathbf{u}\,|_{t=0} &= \mathbf{u}_0, \quad on\ \Omega, \end{cases} \tag{4.28}$$

which does not depend directly on the physical sizes.

Solving it with P2-P1 FEM, we obtain the system of ODE's (4.21) that can be written equivalently in the following way

$$\begin{pmatrix} M\dot{\boldsymbol{u}}(t) \\ \mathbf{0} \end{pmatrix} + \begin{pmatrix} A + C((\mathbf{u}(t))) & B^T \\ B & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}(t) \\ \mathbf{p}(t) \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{f}_p \end{pmatrix}, \tag{4.29}$$

or, if $n = 2$, $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2)$ equivalently

$$\begin{pmatrix} M\dot{\boldsymbol{u}}_{\mathbf{1}}(t) \\ M\dot{\boldsymbol{u}}_{\mathbf{2}}(t) \\ \mathbf{0} \end{pmatrix} + \begin{pmatrix} A + C((\mathbf{u}(t))) & 0 & -B_1^T \\ 0 & A + C((\mathbf{u}(t))) & -B_2^T \\ -B_1 & -B_2 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}_1(t) \\ \mathbf{u}_2(t) \\ \mathbf{p}(t) \end{pmatrix} = \begin{pmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_p \end{pmatrix}, \tag{4.30}$$

where $B_{1,li} := (\psi_l, \frac{d}{dx}\boldsymbol{\varphi}_i)$, $B_{2,li} := (\psi_l, \frac{d}{dy}\boldsymbol{\varphi}_i)$, $B = -B_1 - B_2$. In the numerical solution that will be presented, model (4.30) will be used.

4.3.1 Explicit treatment of the nonlinear term

A first discretization technique consists in using *Crank Nicolson* in time ($\theta = \frac{1}{2}$) and *Adams-Bashfort multistep method* for the nonlinear term, obtaining

$$\begin{pmatrix} \frac{A}{2} + \frac{M}{\Delta t} & 0 & -B_1^T \\ 0 & \frac{A}{2} + \frac{M}{\Delta t} & -B_2^T \\ -B_1 & -B_2 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}_1^{n+1} \\ \mathbf{u}_2^{n+1} \\ \mathbf{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_p \end{pmatrix} + \begin{pmatrix} (\frac{M}{\Delta t} - \frac{A}{2} - \frac{3}{2}C(u^n))\mathbf{u}_1^n + \frac{1}{2}C(u^{n-1})\mathbf{u}_1^{n-1} \\ (\frac{M}{\Delta t} - \frac{A}{2} - \frac{3}{2}C(u^n))\mathbf{u}_2^n + \frac{1}{2}C(u^{n-1})\mathbf{u}_2^{n-1} \\ \mathbf{0} \end{pmatrix}. \tag{4.31}$$

Observe that this discretization corresponds to solve at each iteration a Stokes' problem, with a different forcing term:

$$\begin{pmatrix} \mathcal{A} & 0 & -B_1^T \\ 0 & \mathcal{A} & -B_2^T \\ -B_1 & -B_2 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}_1^{(n+1)} \\ \mathbf{u}_2^{(n+1)} \\ \mathbf{p}^{(n+1)} \end{pmatrix} = \begin{pmatrix} \mathbf{F}_1^{(n)} \\ \mathbf{F}_2^{(n)} \\ \mathbf{F}_p^{(n)} \end{pmatrix}. \tag{4.32}$$

To solve it in our simulations we will use the LU factorization.

4.3.2 Semi-implicit treatment of the nonlinear term

Using a *Crank Nicolson algorithm* in time this corresponds to solve

$$\begin{pmatrix} \frac{A}{2} + \frac{C(\mathbf{u}^n)}{2} + \frac{M}{\Delta t} & 0 & -B_1^T \\ 0 & \frac{A}{2} + \frac{C(\mathbf{u}^n)}{2} + \frac{M}{\Delta t} & -B_2^T \\ -B_1 & -B_2 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}_1^{n+1} \\ \mathbf{u}_2^{n+1} \\ \mathbf{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_p \end{pmatrix} + \begin{pmatrix} (\frac{M}{\Delta t} - \frac{A}{2} - \frac{C(\mathbf{u}^n)}{2})\mathbf{u}_1^n \\ (\frac{M}{\Delta t} - \frac{A}{2} - \frac{C(\mathbf{u}^n)}{2})\mathbf{u}_2^n \\ \mathbf{0} \end{pmatrix}. \tag{4.33}$$

Observe that this *semi-implicit discretization in time* is more expensive, since at every iteration both the system matrix and the right term must be computed, but it permits to deal with a bigger temporal step. This is equivalent to solve

$$\begin{pmatrix} \mathcal{A}^{(n)} & 0 & -B_1^T \\ 0 & \mathcal{A}^{(n)} & -B_2^T \\ -B_1 & -B_2 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u}_1^{(n+1)} \\ \mathbf{u}_2^{(n+1)} \\ \mathbf{p}^{(n+1)} \end{pmatrix} = \begin{pmatrix} \mathbf{F}_1^{(n)} \\ \mathbf{F}_2^{(n)} \\ \mathbf{F}_p^{(n)} \end{pmatrix}. \tag{4.34}$$

Remark 4.3.1 *To obtain a more accurate approximation (order two), in previous systems, instead of $C(\boldsymbol{u}^n)$, consider $C(2\boldsymbol{u}^n - \boldsymbol{u}^{n-1})$, using extrapolation.*

As for the explicit algorithm, we solve the system using an LU-factorization of the system matrix.

4.3.3 Equivalent problem using homogeneous Dirichlet boundary conditions

Consider the general case, in which inhomogeneous stationary Dirichlet boundary conditions are applied. Denote with $\bar{\boldsymbol{u}} := (\bar{\boldsymbol{u}}_1, \bar{\boldsymbol{u}}_2)$ the temporal *mean velocity* in each node. Then we can write

$$\mathbf{u}(t) = \tilde{\boldsymbol{u}}(t) + \bar{\boldsymbol{u}},$$

where $\tilde{\boldsymbol{u}}$ satisfies homogeneous Dirichlet boundary conditions (this is useful dealing with reduced systems, cfr. remark 6.4.2).

Then (4.30) is equivalent to the following system:

$$\begin{aligned}
&\begin{pmatrix} M\dot{\tilde{\boldsymbol{u}}}_{\mathbf{1}}(t) \\ M\dot{\tilde{\boldsymbol{u}}}_{\mathbf{2}}(t) \\ \mathbf{0} \end{pmatrix} + \begin{pmatrix} A + C(\tilde{\boldsymbol{u}}(t) + \bar{\boldsymbol{u}}) & 0 & -B_1^T \\ 0 & A + C(\tilde{\boldsymbol{u}}(t) + \bar{\boldsymbol{u}}) & -B_2^T \\ -B_1 & -B_2 & 0 \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{u}}_1(t) \\ \tilde{\boldsymbol{u}}_2(t) \\ \mathbf{p}(t) \end{pmatrix} + \\
&+ \begin{pmatrix} C(\tilde{\boldsymbol{u}}(t) + \bar{\boldsymbol{u}}) & 0 & -B_1^T \\ 0 & C(\tilde{\boldsymbol{u}}(t) + \bar{\boldsymbol{u}}) & -B_2^T \\ -B_1 & -B_2 & 0 \end{pmatrix} \begin{pmatrix} \bar{\boldsymbol{u}}_1 \\ \bar{\boldsymbol{u}}_2 \\ \mathbf{0} \end{pmatrix} = \qquad , \qquad (4.35) \\
&= \begin{pmatrix} \mathbf{f}_1 - A\bar{\boldsymbol{u}}_{\mathbf{1}} \\ \mathbf{f}_2 - A\bar{\boldsymbol{u}}_{\mathbf{2}} \\ \mathbf{f}_p + B_1\bar{\boldsymbol{u}}_{\mathbf{1}} + B_2\bar{\boldsymbol{u}}_{\mathbf{2}} \end{pmatrix}
\end{aligned}$$

$\tilde{\boldsymbol{u}}(0) = \mathbf{u}(0) - \bar{\boldsymbol{u}}$.

Suppose for simplicity of notations that the backward euler method is used in time. If there is an explicit treatment of the convective term then we hand up with

$$\begin{pmatrix} \frac{M}{\Delta t} + A & 0 & -B_1^T \\ 0 & \frac{M}{\Delta t} + A & -B_2^T \\ -B_1 & -B_2 & 0 \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{u}}_1^{n+1} \\ \tilde{\boldsymbol{u}}_2^{n+1} \\ \mathbf{p}^{n+1} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_1 + \frac{M}{\Delta t}\tilde{\boldsymbol{u}}_{\mathbf{1}}^{\boldsymbol{n}} - A\bar{\boldsymbol{u}}_{\mathbf{1}} \\ \mathbf{f}_2 + \frac{M}{\Delta t}\tilde{\boldsymbol{u}}_{\mathbf{2}}^{\boldsymbol{n}} - A\bar{\boldsymbol{u}}_{\mathbf{2}} \\ \mathbf{f}_p + B_1\bar{\boldsymbol{u}}_{\mathbf{1}} + B_2\bar{\boldsymbol{u}}_{\mathbf{2}} \end{pmatrix} - \begin{pmatrix} C(\tilde{\boldsymbol{u}}^{\boldsymbol{n}} + \bar{\boldsymbol{u}})(\tilde{\boldsymbol{u}}_1^n + \bar{\boldsymbol{u}}_1) \\ C(\tilde{\boldsymbol{u}}^{\boldsymbol{n}} + \bar{\boldsymbol{u}})(\tilde{\boldsymbol{u}}_2^n + \bar{\boldsymbol{u}}_2) \\ \mathbf{0} \end{pmatrix}. \tag{4.36}$$

Otherwise, using a semi-implicit treatment

$$\begin{aligned}
&\begin{pmatrix} \frac{M}{\Delta t} + A + C(\tilde{\boldsymbol{u}}^{\boldsymbol{n}} + \bar{\boldsymbol{u}}) & 0 & -B_1^T \\ 0 & \frac{M}{\Delta t} + A + C(\tilde{\boldsymbol{u}}^{\boldsymbol{n}} + \bar{\boldsymbol{u}}) & -B_2^T \\ -B_1 & -B_2 & 0 \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{u}}_1^{n+1} \\ \tilde{\boldsymbol{u}}_2^{n+1} \\ \mathbf{p}^{n+1} \end{pmatrix} = \\
&= \begin{pmatrix} \mathbf{f}_1 + \frac{M}{\Delta t}\tilde{\boldsymbol{u}}_{\mathbf{1}}^{\boldsymbol{n}} - A\bar{\boldsymbol{u}}_{\mathbf{1}} \\ \mathbf{f}_2 + \frac{M}{\Delta t}\tilde{\boldsymbol{u}}_{\mathbf{2}}^{\boldsymbol{n}} - A\bar{\boldsymbol{u}}_{\mathbf{2}} \\ \mathbf{f}_p + B_1\bar{\boldsymbol{u}}_{\mathbf{1}} + B_2\bar{\boldsymbol{u}}_{\mathbf{2}} \end{pmatrix} - \begin{pmatrix} C(\tilde{\boldsymbol{u}}^{\boldsymbol{n}} + \bar{\boldsymbol{u}})\bar{\boldsymbol{u}}_1 \\ C(\tilde{\boldsymbol{u}}^{\boldsymbol{n}} + \bar{\boldsymbol{u}})\bar{\boldsymbol{u}}_2 \\ \mathbf{0} \end{pmatrix}.
\end{aligned} \tag{4.37}$$

As we will see, applying reduction techniques is less costly using an explicit treatment of the convective term and solving the system for the fluctuation velocity $\tilde{\boldsymbol{u}}$: in fact in this way only the right-hand side vector is computed at each iteration and it is not necessary to build the system matrix to impose Dirichlet conditions at each iteration.

4.4 Test problems

4.4.1 Test case: Backward facing step

The flow over a backward facing step is an example of complex flow: in fact it is characterized by a recirculation area between the step and the reattachment point, as sketched in figure 4.1. We will consider the bottom configuration.

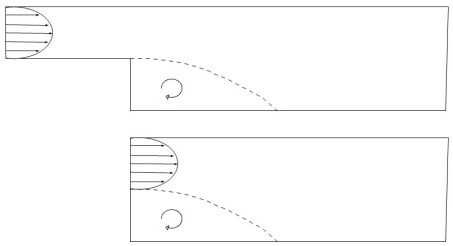

Figure 4.1: *Backward facing step: sketch of two equivalent problems.*

Let Ω be a rectangular domain $[0, L]\times[0, 1]$, and consider the nondimensional Navier Stokes problem:

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} - \frac{1}{Re}\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p &= \mathbf{f}, & on & \Omega\times[0,T] \\ div\mathbf{u} &= 0, & on & \Omega\times[0,T] \\ \mathbf{u} &= \mathbf{u}_d, & on & \Gamma_d\times[0,T] \\ (p\mathbb{I}_2 + \nu\nabla\mathbf{u})\cdot n &= 0, & on & \Gamma_n\times[0,T] \\ \mathbf{u} &= \mathbf{u}_0, & on & \Omega\times\{0\} \end{cases} \tag{4.38}$$

where $\Gamma_n = \{L\}\times[0,1]$, $\Gamma_d = \partial\Omega\backslash\Gamma_n$, $\mathbf{u}:\ \Omega\to\mathbb{R}^2$, $p:\ \Omega\to\mathbb{R}^2$, $\mathbf{f}\in\mathbf{L}^2(\Omega) := (L^2(\Omega))^2$, $\mathbf{f}=\mathbf{0}$, $\mathbf{u}_0$ is the solution of the stationary Stokes equation, depicted in figure 4.3 and

$$\mathbf{u}_d(x,y) = \begin{cases} (ay^2 - \frac{3}{2}ay + \frac{a}{2}, 0), & x = 0,\ y\in[0.5,1], \\ (0,0)^t, & elsewhere\ on\ \Gamma_d \end{cases},$$

$Re = 400$, $L = 10$, $a = -\frac{4}{75}Re$, $T = 25$, $Dt = 0.025$. The grid used for numerical computations is depicted in figure 4.2. The backward facing step is analyzed for example in (7).

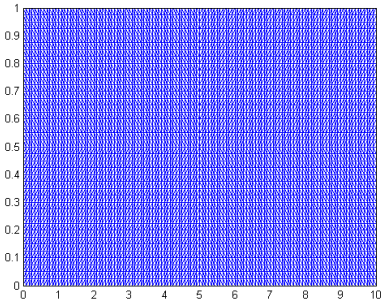

Figure 4.2: *Uniform grid used to discretize the backward facing step problem:* 120 *horizontal segments vs.* 40 *vertical segments).*

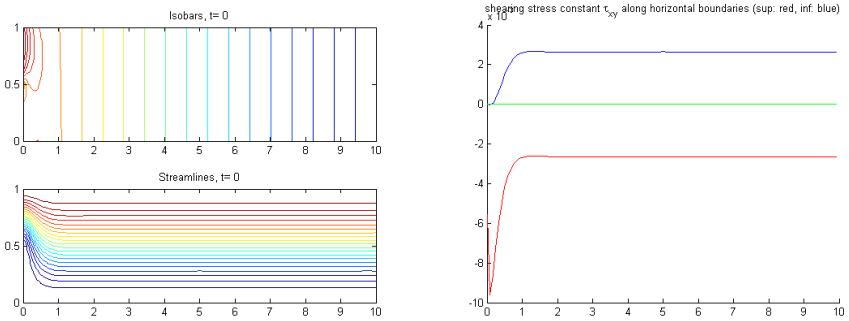

Figure 4.3: *Initial condition of the backward facing step problem: solution of the stationary Stokes problem. Streamlines and Isobars (left),* τ *(right).*

Consider the *shearing stress*

$$\tau_{xy} := \frac{1}{Re}\left(\frac{\partial u_1(x,y)}{\partial y} + \frac{\partial u_2(x,y)}{\partial x}\right),$$

for every $(x, y) \in \Omega$. As can be seen in figure 4.4, before reaching the stationary solution, the dynamic presents small vortexes in both horizontal sides of Ω. The corresponding *attachment points* (x_i, y_i) of $[0, L] \times \{0, 1\}$ are such that $\tau_{x_i y_i} = 0$.

Other interesting quantities to describe the fluid flow are *isobars*, isolines of pressure and *streamlines*, isolines of ψ, where the stream function ψ is a scalar function such that $u_1(x,t) = \frac{\partial \psi}{\partial y}(x,t)$ and $u_2(x,t) = \frac{\partial \psi}{\partial x}(x,t)$, calculated solving a Poisson problem.

As can be seen in figures 4.4 and 4.5, this test problem describes a transitional dynamic, starting from the initial solution, and reaching the stationary Navier-Stokes solution around $T = 25$.

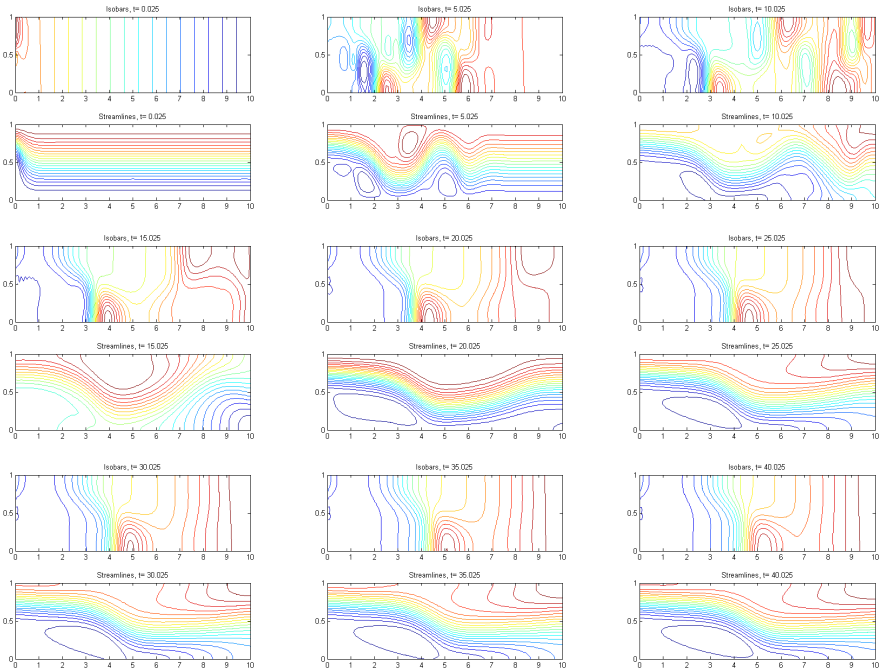

Figure 4.4: *$Re = 400$ streamlines and pressure contour plots at different time instances $[Dt : 5 + Dt : 40 + Dt]$.*

4.4.2 Test case: Square obstacle

Let Ω be the domain of figure 4.6, where the obstacle $\mathcal{O}$ is a square centered in $(0, 0)$ and with side length 1, and consider the nondimensional Navier Stokes problem:

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} - \frac{1}{Re}\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u} + \nabla p &= \mathbf{f}, & on \quad \Omega \times [0,T] \\ div\mathbf{u} &= 0, & on \quad \Omega \times [0,T] \\ \mathbf{u} &= \mathbf{u}_d, & on \quad \Gamma_d \times [0,T] \\ (p\mathbb{I}_2 + \nu\nabla\mathbf{u})\cdot n &= 0, & on \quad \Gamma_n \times [0,T] \\ \mathbf{u} &= \mathbf{u}_0, & on \quad \Omega \times \{0\} \end{cases} \tag{4.39}$$

where $\Gamma_n = \{21.5\} \times [-4.5, 4.5]$, $\Gamma_d = \partial\Omega \setminus \Gamma_n$, $\mathbf{u} : \ \Omega \to \mathbb{R}^2$, $p : \ \Omega \to \mathbb{R}^2$, $\mathbf{f} \in \mathbf{L}^2(\Omega) := (L^2(\Omega))^2$, $\mathbf{f} = \mathbf{0}$, $Re = 100$, and

$$\mathbf{u}_d(x,y) = \begin{cases} (1,0), & x = 0, \ y \in [-4.5, 4.5], \\ (0,0)^t, & elsewhere \ on \ \Gamma_d \end{cases} .$$

To study the dynamic we consider the isolines of the *vorticity* ω, i.e. a scalar function s.t. $\omega(x,t) = \frac{\partial u_2}{\partial x_1}(x,t) - \frac{\partial u_1}{\partial x_2}(x,t)$. Moreover we consider the *drag* C_d and the

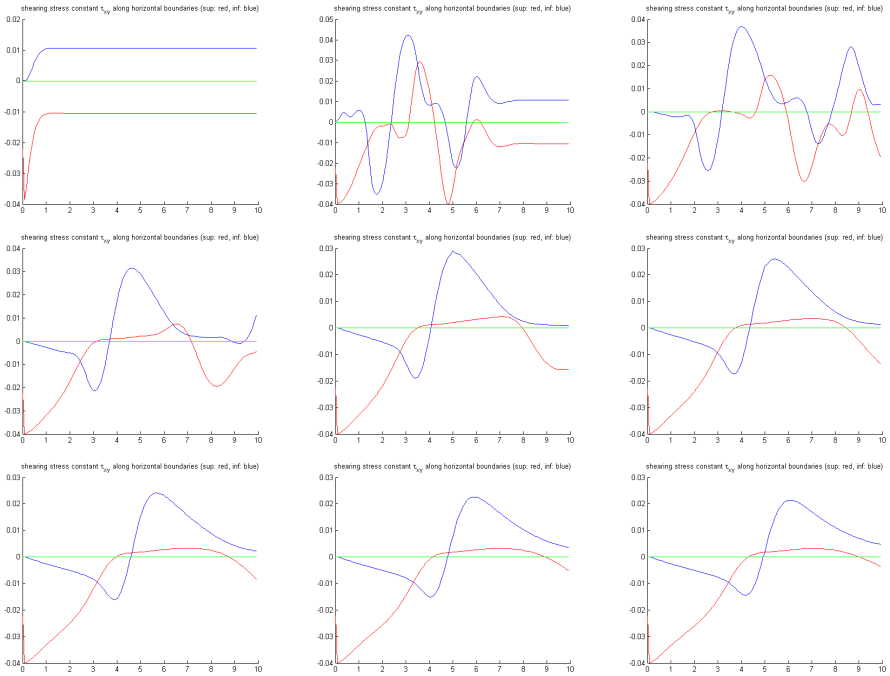

Figure 4.5: *$Re = 400$ evolution of τ corresponding to figure 4.4 at different time instances $[Dt : 5 + Dt : 40 + Dt]$.*

lift C_l, coefficients defined respectively as

$$C_d = \int_{\partial\mathcal{O}} pn_x - \frac{1}{Re}\frac{\partial u_1}{\partial x}n_x - \frac{1}{Re}\frac{\partial u_1}{\partial y}n_y dl,$$

$$C_l = \int_{\partial\mathcal{O}} pn_y - \frac{1}{Re}\frac{\partial u_2}{\partial x}n_x - \frac{1}{Re}\frac{\partial u_2}{\partial y}n_y dl.$$

In figure 4.7 it can be seen the dynamic until the periodic solution is reached, obtained using a semi-implicit discretization, whereas in figures 4.8 and 4.9 the dynamic of the periodic regime is plotted. The period is $[0, 5.2]$ and it is discretized using an explicit treatment of the nonlinear term with time step $Dt = 0.002$.

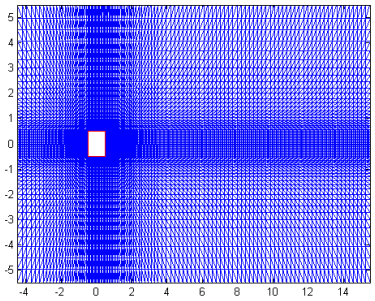

Figure 4.6: *Grid used to discretize the obstacle problem:* 116 *horizontal segments vs.* 60 *vertical segments).*

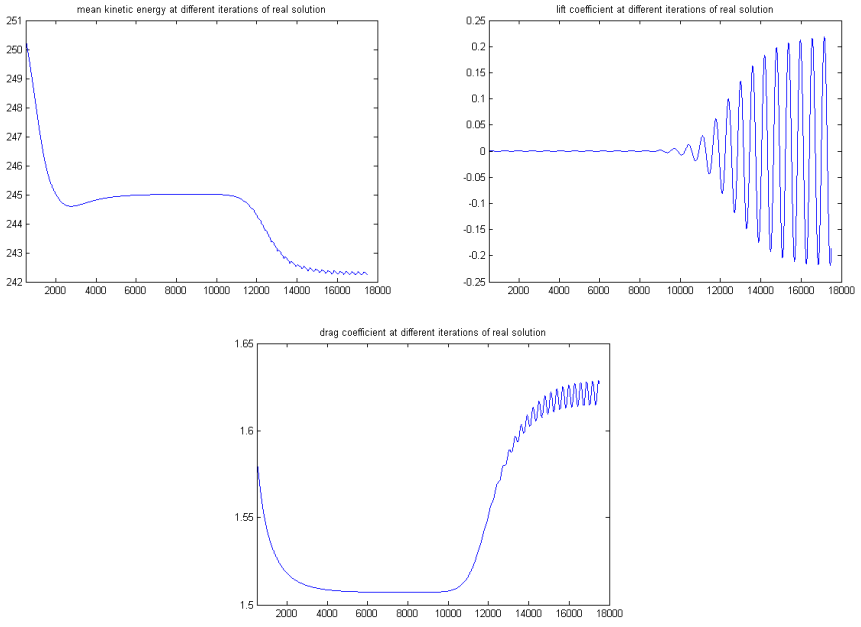

Figure 4.7: $Re = 100$ *Mean kinetic energy, lift and drag before the periodic regime.*

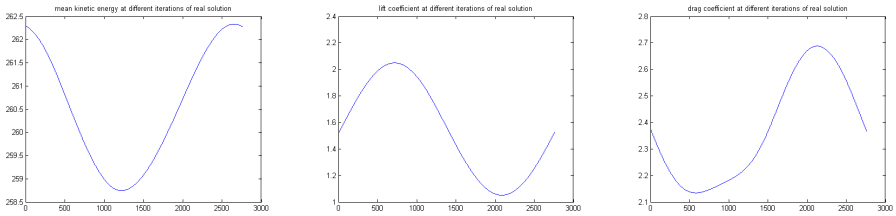

Figure 4.8: *$Re = 100$ Mean kinetic energy, lift and drag before the period* $[0, 5.52]$.

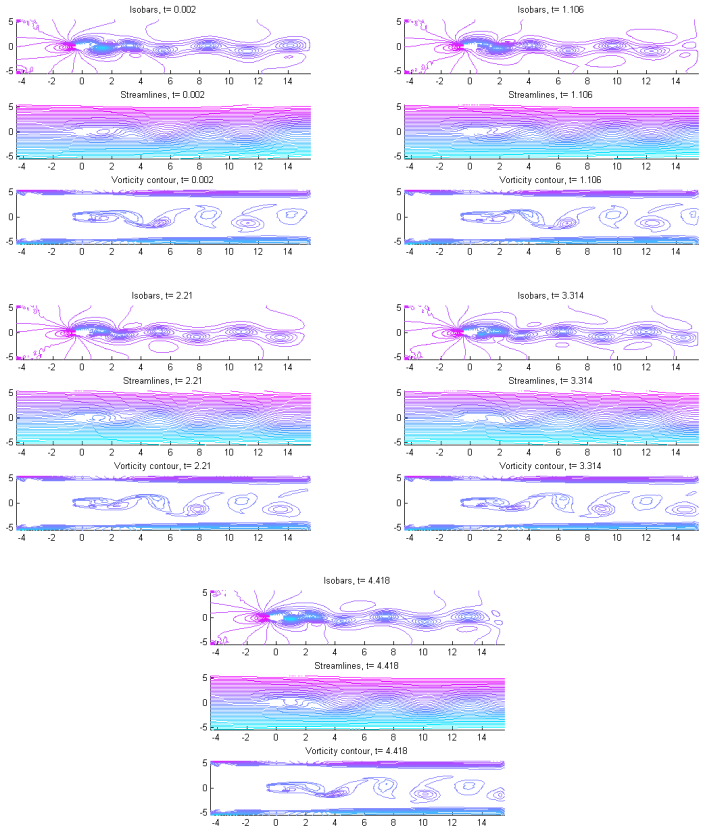

Figure 4.9: *$Re = 100$ streamlines, pressure and vorticity isolines at different time instances in the period* $[0, 5.52]$.

Part II

Reduced Order Modeling (ROM)

How can it be that mathematics, being after all a product of human thought independent of experience, is so admirably adapted to the objects of reality?

(A. Einstein)

In this part *Model Order Reduction* (MOR) techniques are introduced, fundamental tools for solving realistic problems. In particular we will describe the *Proper Orthogonal Decomposition (POD)* method and we will consider fluid dynamic problems, focusing on the reduction of Navier-Stokes equation.

5

Model Order Reduction (MOR): a general overview

5.1 Introduction

Computational models are useful primarily for two reasons: for *simulation* and *control* (74). However any realistic model will have high complexity, i.e. it will require many state variables to be adequately described: thus a *simplification* or *model order reduction* will be needed in order to perform a simulation in an amount of time which is acceptable or for the design of a low order controller. As mentioned in (144), for realistic simulations, many thousands or even millions of degrees of freedom are often required to obtain useful approximations. Thus, if one needs to do multiple simulations or to do a simulation in real time, the use of traditional discretization methods, e.g., finite element, finite volume, or spectral methods, may not be feasible.

Moreover important issues with large-scale systems are *storage*, *computational speed*, *accuracy* and preservation of system's properties (76). Figure 5.1 summarize briefly the genesis of a reduced model.

Consider a *continuous linear dynamical system* in state space form, $t \in \mathbb{R}^+$:

$$\Sigma : \begin{cases} \frac{d}{dt}\mathbf{x}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t), \\ y(t) &= C\mathbf{x}(t) + D\mathbf{u}(t), \end{cases} \tag{5.1}$$

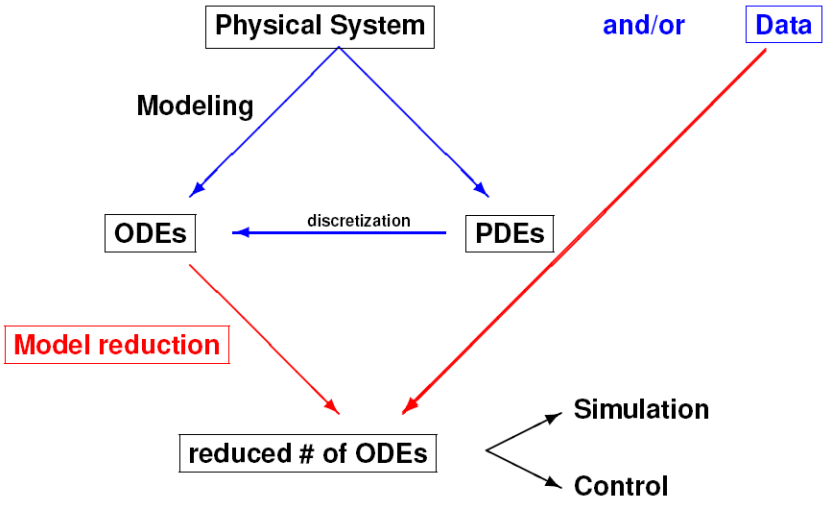

Figure 5.1: *This picture is taken from (76).*

$A \in \mathbb{R}^{n\times n}$ (*state space* matrix), $B \in \mathbb{R}^{n\times p}$ (*input* map), $C \in \mathbb{R}^{p\times n}$ (*output* map), $D \in \mathbb{R}^{p\times m}$ (*direct transmission* map), $\mathbf{u}(t) \in \mathbb{R}^m$ is called *input* or *control*, $\mathbf{y}(t) \in \mathbb{R}^p$ is the *output*, $\mathbf{x}(t) \in \mathbb{R}^n$ is the *state* vector and n is the *order* or *complexity* of the system. We will denote it briefly $\Sigma = (A, B, C, D)$. If $p, m > 1$, then Σ is a *multiple input-multiple output (MIMO)* system, whereas if $p = 1 = m$, it is called *single input-single output (SISO)*.

The corresponding discrete problem has the following expression, $t \in \mathbb{N}$

$$\Sigma : \begin{cases} \mathbf{x}(k+1) &= A_c\mathbf{x}(k) + B_c\mathbf{u}(k), \\ \mathbf{y}(k) &= C_c\mathbf{x}(k) + D_c\mathbf{u}(k). \end{cases} \tag{5.2}$$

and can be obtained discretizing in time (5.1).

5.2 Reduction of linear dynamical systems

The model reduction problem consists in approximating Σ defined in (5.1) with $\hat{\Sigma} = (\hat{A}, \hat{B}, \hat{C}, \hat{D})$ $\hat{A} \in \mathbb{R}^{k\times n}$, $\hat{B} \in \mathbb{R}^{k\times p}$, $\hat{C} \in \mathbb{R}^{p\times k}$, $\hat{D} \in \mathbb{R}^{p\times m}$,

$$k << n$$

such that (76)

1. the approximation error is *small*;

2. system properties, like stability and passivity, are preserved;

3. the algorithm is computationally stable and efficient.

The unifying feature of model reduction methods is that they are obtained choosing a suitable *Petrov-Galerkin projection*. Let $\pi = VW^*$ be a projection, then the corresponding reduced model of (5.1) is obtained as follows

$$\begin{cases} \frac{d}{dt}\hat{\boldsymbol{x}} = (W^*AV)\hat{\boldsymbol{x}} + (W^*B)\mathbf{u}, \\ \hat{\boldsymbol{y}} = (CV)\hat{\boldsymbol{x}} + D\mathbf{u}. \end{cases} \tag{5.3}$$

Observe that the input $\mathbf{u}(t) \in \mathbb{R}^m$ is not touched, while the output and the state are now denoted respectively by $\hat{\boldsymbol{y}}(t) \in \mathbb{R}^p$ and $\hat{\boldsymbol{x}}(t) \in \mathbb{R}^k$.

There are basically two sets of methods (74)

1. SVD based methods (Balanced Model Reduction, Hankel Norm Approximation, Singular Perturbation Approximation),

2. moment matching based (or Krylov) methods.

SVD-based (or *full space*) methods (Balanced truncation (107), Hankel norm approximation (90), Singular Perturbation Approximation (98)) have their roots in the Singular Value Decomposition and in the *lower-rank* approximation: they preserve stability and provide error upper bounds. The limitation of this approaches is that they involve the solution of two Lyapunov equations, which requires dense computation (the cost is $O(n^3)$ irrespective of the sparsity): hence *they are only applicable to moderately sized problems*. Moreover there could be problems with ill-conditioned problems, i.e. those ones with at least one eigenvalue of A (or equivalently one pole of $H(s)$) close to imaginary axis.

Instead Krylov methods (like *Lanczos* and *Arnoldi*) are *iterative* ones, thus they can be applied to high order systems (the cost is $O(kn^2)$) and they are based on moment matching of the transfer function $H(i\omega) = C(i\omega I - A)^{-1}B$ of the system Σ. These methods are applied both in iterative eigenvalue computations (117) and in model order reduction (*rational interpolation*, *realization*), but the resulting reduced systems have no guaranteed error bound and stability is not necessarily preserved. More recent algorithms are *Padè via Lanczos (PVL) (78)* and *multipoint rational interpolation (92)*, which requires that the selection of interpolation points is done by the user. However all these methods are local in nature, thus it is difficult to establish global error bounds. The methods differ in the way the bases are computed.

5.3 Reduction of nonlinear dynamical systems

The linear problem can be generalized in the following way:

$$\Sigma : \begin{cases} \frac{d}{dt}\mathbf{x}(t) &= f(\mathbf{x}(t), \mathbf{u}(t)), \\ \mathbf{y}(t) &= g(\mathbf{x}(t), \mathbf{u}(t)), \end{cases} \tag{5.4}$$

where $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ describes the *dynamics* of Σ and $g : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^p$ describes the way that the observations are deduced from the state and the input. Observe that the linear system (5.1) can be seen as a particular case of (5.4). Thus a first way to solve nonlinear problems is to *linearize* them, and apply techniques presented above. Neverless there exist methods, like e.g. *Proper Orthogonal Decomposition (POD)* and *Reduced Basis (RB)* methods, which are suitable for nonlinear systems.

In this context we can extend the concept of *Petrov-Galerkin projection*, introduced before for linear dynamical systems. Using projection-based MOR techniques we define a system of order k, considering a projection matrix V, whose columns are a basis of the reduced state space, such that $V\hat{x} \approx x$. Moreover to construct equations for the reduced system, we consider the residual $\mathbf{r} = -\frac{d}{dt}V\hat{\boldsymbol{x}}(t) + f(V\hat{\boldsymbol{x}}(t), u(t))$ and require that it is orthogonal to a reduced order space, spanned by the columns of a matrix W: $W^T\mathbf{r} = 0$. Thus we consider the projection $\pi = VW^*$, such that $W^*V = I_k$, $V, W \in \mathbb{R}^{n\times k}$. The reduced model can be defined in the following way

$$\hat{\Sigma} : \begin{cases} \frac{d}{dt}\hat{\boldsymbol{x}}(t) &= W^* f(V\hat{\boldsymbol{x}}(t), \mathbf{u}(t)), \\ \mathbf{y}(t) &= g(V\hat{\boldsymbol{x}}(t), \mathbf{u}(t)), \end{cases} \tag{5.5}$$

whose trajectories $\hat{\boldsymbol{x}} = W^*\mathbf{x}$ evolve in a k-dimensional subspace. For *linear* reduced systems (5.3), the cost of building the reduced model is independent on N: it depends only on the reduced size k. Thus, by applying MOR we immediately reduce the numerical cost of simulating a given system. In the nonlinear case the situation is more involved (116): although the number of equations is reduced to k, and the unknown vector is of dimension k, this does *not* imply that the reduced system is inexpensive to simulate. In fact the cost of evaluating $W^* f(V\hat{\boldsymbol{x}}(t), \mathbf{u}(t))$ and $g(V\hat{\boldsymbol{x}}(t), \mathbf{u}(t))$ must be considered: firstly we must compute $\mathbf{x} = V\hat{\boldsymbol{x}}$, then we must evaluate $f(\mathbf{x})$ and $g(\mathbf{x})$, and finally we multiply the first by W^T. If the evaluation of f and g costs $O(n^\alpha)$ operations, $\alpha \geq 1$, then the previous procedure costs $O(n^\alpha + 2nk)$. Moreover if we solve the reduced system using backward Euler, we also need to apply the Newton's method, and then we must compute the Jacobian of f and g. Thus in the nonlinear case reduction of the order of a system at hand does *not* automatically imply reduction of numerical cost associated with simulating the reduced order model: this is one of the main differences between linear and nonlinear MOR.

In this thesis, starting from next chapter, we will focus on POD as a reduction strategy; some other methods for solving nonlinear problems can be found in the literature e.g. Reduced Basis (RB) (91, 111, 113, 118), Trajectory Piecewise Linear (TPWL) (116), Volterra Series Representations and Harmonic Balance (104), Empirical Gramians (88, 101).

6

Proper Orthogonal Decomposition (POD) method

6.1 Introduction

Proper Orthogonal Decomposition (POD) method (or *Empirical Eigenfunctions Method*, or *Karhunen-Loève decomposition*) was introduced in (103) for the study of weather prediction: it is a strategy of finding *compact representations* of an ensemble of data in the form of a set of countable, orthonormal basis functions. One of the central issues of POD is the reduction of data expressing their *essential information* by means of a few basis vectors (76, 94, 124).

As mentioned in (115), POD is a model reduction technique used mainly for complex non-linear problems. It was first proposed by Karhunen (97) and Loeve (102), independently, thus is sometimes called the Karhunen-Loève expansion. Subsequently, it has been applied in various applications. In (120) an important progress was made and the *method of snapshots* was incorporated into the POD framework. In (105), the method was first called POD and it was used to study turbulent flows. In general POD was successfully used in a variety of fields (94): signal analysis and pattern recognition (87), fluidynamics and coherent structures (95, 108, 120, 121), control theory (73) and inverse problems (79).

As underlined in (77), the idea of expanding physical quantities with respect to an orthonormal basis, it dates back to the work of Joseph Fourier in his memoir *On the Propagation of Heat in Solid Bodies* written in 1805, where he proposed to expands an arbitrary function in a series of trigonometric basis functions: the Fourier expansion. When discretizing non-linear PDE's with finite volume, finite difference, finite element or spectral methods, one uses basis functions that have very little connection with the problem or with the underlying PDE's: instead, POD uses basis functions that are generated from the numerical solutions of the system or from the experimental measurements. Thus POD bases are better in approximating a set of data because they are derived directly from the data, while the other orthonormal bases are defined without any relation with the data. This property brings advantages and disadvantages of POD basis. Indeed, the POD basis can better approximate data from which they are generated than other orthonormal bases. The basis functions therefore reflect the relevant dynamics of the data, provided that these dynamics are captured in the data. However, the validity of POD's approximations is limited to how well the first k basis functions represent the dynamics of the system. Substantially the main idea of POD is to select a certain number of patterns such that the effects of the neglected patterns are minimized (77).

In the literature, the POD method is presented both as a way to construct directly

an approximated solution of a PDE problem in an Hilbert space, using *continuous in space* POD functions, in the *Reduced Order Modeling (ROM)* context (94), and as a *Model Order Reduction (MOR)* technique applied to a *discrete in space* state-space system ($\mathbb{X} = \mathbb{R}^n$) (77). In the following we will focus firstly on the MOR formulation: the continuous in space formulation is summarized in section 6.4, while their equivalence is analyzed in appendix B.

6.2 POD in the finite dimensional context

Consider the nonlinear state space system (5.4): for a fixed input $\mathbf{u}$ compute the state $\mathbf{x}(t)$ in N time-instances $0 < t_1 < t_2 < \ldots < t_N$ and define

$$\mathcal{X} = [\mathbf{x}(t_1), \ldots, \mathbf{x}(t_N)] \in \mathbb{R}^{n \times N}.$$

This collection of data is called *matrix of snapshots* of the state.

6.2.1 Computation of the POD basis

Consider the following optimization problem: let $\mathbf{u}_1$ be the solution of

$$\max_{\mathbf{u}_1 \in \mathbb{R}^n} \sum_{j=1}^{N} |\mathbf{x}(t_j) \cdot \mathbf{u}_1|^2, \qquad s.t. \qquad \|\mathbf{u}_1\|_{\mathbb{R}^n} = 1,$$

where we are using the euclidian scalar product. The corresponding Lagrangian function is $\mathcal{L}(\mathbf{u}_1, \lambda) = \sum_{j=1}^{N} |\mathbf{x}(t_j) \cdot \mathbf{u}_1|^2 + \lambda(1 - \|\mathbf{u}_1\|_{\mathbb{R}^n})$, and applying the first order necessary conditions for optimality: $\nabla \mathcal{L}(\mathbf{u}_1, \lambda) = \mathbf{0}$, which are also sufficient, it can be proved (124) that the solution of the optimization problem $\mathbf{u}_1$ must satisfy

$$\mathcal{X}\mathcal{X}^T \mathbf{u}_1 = \lambda \mathbf{u}_1, \qquad \|\mathbf{u}_1\|_{\mathbb{R}^n} = 1,$$

i.e. the solution of the optimization problem is the first eigenfunction.

Moreover,

$$\max_{\mathbf{u}_1 \in \mathbb{R}^n} \sum_{j=1}^{N} |\mathbf{x}(t_j) \cdot \mathbf{u}_1|^2 = \lambda_1,$$

the largest eigwnvalue of $\mathcal{X}\mathcal{X}^T$.

Iterating this procedure the following Theorem can be proved (124).

Theorem 6.2.1 *Let $\mathcal{X} \in \mathbb{R}^{n\times N}$ be a given matrix of rank d and let $\mathcal{X} = USV^*$ be its SVD, $U = [\boldsymbol{u}_1, \dots, \boldsymbol{u}_n]$, $S = diag(\sigma_1, \dots, \sigma_n)$. Then for any $k \leq d$ the solution of*

$$\max_{\boldsymbol{\xi}_1,\dots,\boldsymbol{\xi}_k \in \mathbb{R}^n} \sum_{i=1}^{k}\sum_{j=1}^{N} |\boldsymbol{x}(t_j)\cdot\boldsymbol{\xi}_i|^2, \qquad s.t. \qquad \boldsymbol{\xi}_i\cdot\boldsymbol{\xi}_j = \delta_{ij}, \qquad 1 \leq i,j \leq k \tag{6.1}$$

is given by the left singular vectors $\{\boldsymbol{u}_i\}_{i=1,\dots,k}$. Moreover

$$\sum_{i=1}^{k}\sum_{j=1}^{N} |\boldsymbol{x}(t_j)\cdot\boldsymbol{u}_i|^2 = \sum_i \sigma_i^2.$$

As a consequence the computation of the first k POD eigenfunctions $\{\mathbf{u}_1, \dots, \mathbf{u}_k\}$ is equivalent to find the eigenvalue decomposition of $\mathcal{X}\mathcal{X}^T$, or, analogously, to compute the SVD of $\mathcal{X}$ and considering its first k left singular vectors, i.e.

$$\mathcal{X} = USV^* \approx U_k S_k V_k^*, \ k << n,$$

defining $U_k = [u_1, \dots, u_k]$. This approximation corresponds to approximate

$$\mathbf{x}(t_i) \approx \hat{\boldsymbol{x}}(t_i) = \sum_{j=1}^{k} a_{ij}\mathbf{u}_j, \ i = 1, \dots, N$$

for suitable coefficients a_{ij}.

Remark 6.2.1 *Observe that the maximization problem (6.1) is equivalent to the following minimization one*

$$\min_{\boldsymbol{\xi}_1,\dots,\boldsymbol{\xi}_k \in \mathbb{R}^n} \sum_{j=1}^{N} \left| \boldsymbol{x}(t_j) - \sum_{i=1}^{k} (\boldsymbol{x}(t_j)\cdot\boldsymbol{\xi}_i)\boldsymbol{\xi}_i \right|^2, \qquad s.t. \qquad \boldsymbol{\xi}_i\cdot\boldsymbol{\xi}_j = \delta_{ij}, \qquad 1 \leq i,j \leq k, \tag{6.2}$$

which corresponds to find a proper k-dimensional subspace of $\mathbb{R}^n$ of minimal distance from the snapshots. Thus POD consists in choosing the orthonormal basis such that for every $k \leq d$ the mean square error between the elements $\boldsymbol{x}(t_j)$ and the corresponding $k-th$ partial sum of $\sum_{i=1}^{k}(\boldsymbol{x}(t_j)\cdot\boldsymbol{\xi}_i)\boldsymbol{\xi}_i$ is minimized on average.

For completeness we collect some properties of the POD basis (124).

Theorem 6.2.2 *(Optimality of the POD basis) Let all the assumptions of Theorem 6.2.1 be satisfied. Suppose that $Z \in \mathbb{R}^{n\times d}$ denotes a matrix with pairwise orthonormal vectors z_i, and define $C_{ij} = z_i \cdot \boldsymbol{x}(t_j)$, $i = 1, \dots, d$, $j = 1, \dots, N$, thus $\mathcal{X} = ZC$.*

Then for every $k = 1, \ldots, d$

$$\|\mathcal{X} - U_k \mathcal{A}_k\|_F \leq \|\mathcal{X} - Z_k C_k\|_F ,$$

denoting with Z_k *and* C_k *the first* k *columns of* Z *and* C *respectively* ($\|A\|_F = \sqrt{trace(A^T A)}$, $A \in \mathbb{R}^{n \times N}$).

Thus the POD basis of rank k is optimal in the sense of representing in the mean the columns of $\mathcal{X}$ as a linear combination of k orthonormal basis vectors (124) :

$$\sum_{i=1}^{k} \sum_{j=1}^{N} |\mathbf{x}(t_j) \cdot \mathbf{u}_i|^2 = \sum_i \sigma_i^2 \geq \sum_{i=1}^{k} \sum_{j=1}^{N} |\mathbf{x}(t_j) \cdot \mathbf{z}_i|^2 ,$$

for any set of orthonormal vectors $\{z_i\}$.

Interpreting $\mathcal{X}_{ij}$ as the velocity of a fluid at location $\mathbf{x}_i$ and at time t_j, this property means that the first k POD-basis functions capture more energy on average than the first k functions of any other basis.

Another important property is the following:

Corollary 6.2.1 *(Uncorrelated POD coefficients) Let all the hypothesis of Theorem 6.2.1 hold. Then*

$$\sum_{j=1}^{N} (\boldsymbol{x}(t_j) \cdot \boldsymbol{u}_i)(\boldsymbol{x}(t_j) \cdot \boldsymbol{u}_k) = \sigma_i^2 \delta_{ik},$$

(t-average of coefficients).

Observe that the error depends on the basis choice. A possible way to choose an optimal k is to consider the smallest k s.t.

$$\frac{\sum_i^k \sigma_i}{\sum_i^n \sigma_i} < \ tolerance,$$

where σ_i are $\mathcal{X}$'s singular values (77, 124). Another strategy consists in using a threshold on singular values.

Remark 6.2.2 *If* $N < n$*, an alternative way to compute the basis* U *is by solving*

$$\begin{aligned} \mathcal{X}^T \mathcal{X} \boldsymbol{v}_i &= \lambda_i \boldsymbol{v}_i, \qquad i = 1, \ldots, k \\ \boldsymbol{u}_i &= \tfrac{1}{\sqrt{\lambda_i}} \mathcal{X} \boldsymbol{v}_i. \end{aligned} \tag{6.3}$$

For historical reasons (120) this method is known as ***method of snapshots****. Both it and the problem*

$$\mathcal{X} \mathcal{X}^T \boldsymbol{u}_i = \lambda_i \boldsymbol{u}_i, \qquad i = 1, \ldots, k$$

are equivalent to compute the truncated SVD of $\mathcal{X}$.

Although it permits to work with an $N \times N$ *matrix, the method of snapshots is not stable as an algorithm to compute singular values (123). Thus in the following we prefer to use the SVD for the computation of singular vectors and values. The infinite dimensional counter-party is described in Remark 6.4.1.*

6.2.1.1 Generalization using a weighted inner product in $\mathbb{R}^n$

It is possible to extend previous results, considering a weighted inner product in $\mathbb{R}^n$ (124): $(\mathbf{u}, \mathbf{v})_W := \mathbf{v}^T W \mathbf{u}$, where W is a symmetric positive definite matrix. This formulation can be interpreted for example as a quadrature rule, used to approximate an integral $\int_\Omega \phi_v \phi_u dx$, $\phi_u, \phi_v \in \mathcal{L}^2(\Omega)$. In this framework W is a diagonal matrix whose elements are the weights and the vectors $\mathbf{v}$ and $\mathbf{u}$ are thought as nodal values of the continuous functions ϕ_u and ϕ_v.

Consider the optimization problem

$$\max_{\mathbf{u}_1 \in \mathbb{R}^n} \sum_{j=1}^{N} |(\mathbf{x}(t_j), \mathbf{u}_1)_W|^2, \qquad s.t. \qquad \|\mathbf{u}_1\|_W = 1,$$

using a Lagrangian formulation (124) it can be proved that it is equivalent to solve

$$(W\mathcal{X})(W\mathcal{X})^T \mathbf{u}_1 = \lambda W \mathbf{u}_1.$$

Setting $\bar{\boldsymbol{u}}_1 = W^{1/2}\mathbf{u}_1$ and $\bar{\mathcal{X}} = W^{1/2}\mathcal{X}$, we obtain the eigenvalue problem

$$\bar{\mathcal{X}}\bar{\mathcal{X}}^T \bar{\boldsymbol{u}}_1 = \lambda \bar{\boldsymbol{u}}_1,$$

which can be solved also computing the SVD of $\bar{\mathcal{X}} = \bar{U}\bar{S}\bar{V}^*$. It can be shown that the solution is

$$\mathbf{u}_1 = W^{-1/2}\bar{\boldsymbol{u}}_1.$$

Thus Theorem 6.2.1 can be restated in the following way

Corollary 6.2.2 *Let* $\mathcal{X} \in \mathbb{R}^{n \times N}$ *be a given matrix of rank* d *and let* $\bar{\mathcal{X}} = \bar{U}\bar{S}\bar{V}^*$ *be its SVD. Then for any* $k \leq d$ *the solution of*

$$\max_{\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_k \in \mathbb{R}^n} \sum_{i=1}^{k} \sum_{j=1}^{N} |(\boldsymbol{x}(t_j), \boldsymbol{\xi}_i)_W|^2, \qquad s.t. \qquad (\boldsymbol{\xi}_i, \boldsymbol{\xi}_j)_W = \delta_{ij}, \qquad 1 \leq i, j \leq k$$

is given by $\boldsymbol{u}_i = W^{-1/2}\bar{\boldsymbol{u}}_i$, $i = 1, \dots, k$. *Moreover*

$$\sum_{i=1}^{k} \sum_{j=1}^{N} |(\boldsymbol{x}(t_j), \boldsymbol{u}_i)_W|^2 = \sum_i \sigma_i^2.$$

Remark 6.2.3 *If $N < n$ the **method of snapshots** can be used, solving*

$$\begin{aligned} \bar{\mathcal{X}}^T\bar{\mathcal{X}}\bar{\boldsymbol{v}}_i &= \mathcal{X}^T W\mathcal{X}\bar{\boldsymbol{v}}_i = \lambda_i W\bar{\boldsymbol{v}}_i, \qquad i = 1,\dots,k \\ \boldsymbol{u}_i &= W^{-1/2}\bar{\boldsymbol{u}}_i = \tfrac{1}{\sqrt{\lambda_i}}\mathcal{X}\bar{\boldsymbol{v}}_i. \end{aligned} \tag{6.4}$$

More generally consider the following optimization problem (cfr. Remark 6.2.1.1):

$$\min_{\boldsymbol{\xi}_1,\dots,\boldsymbol{\xi}_k\in\mathbb{R}^n} \sum_{j=1}^{N} \alpha_j \left\| \mathbf{x}(t_j) - \sum_{i=1}^{k} (\mathbf{x}(t_j),\boldsymbol{\xi}_i)_W \boldsymbol{\xi}_i \right\|_W^2, \qquad s.t. \qquad (\boldsymbol{\xi}_i,\boldsymbol{\xi}_j)_W = \delta_{ij}, \qquad 1 \le i,j \le k. \tag{6.5}$$

Using the Lagrangian conditions, and defining $D = diag(\alpha_1,\dots,\alpha_N)$, (6.5) is equivalent to solve (124)

$$\mathcal{X}D\mathcal{X}^T W\mathbf{u}_i = \lambda_i\mathbf{u}_i.$$

Setting $\bar{\boldsymbol{u}} = W^{1/2}\mathbf{u}$ and $\bar{\mathcal{X}} = W^{1/2}\mathcal{X}D^{1/2}$, we obtain the eigenvalue problem

$$\begin{aligned} \bar{\mathcal{X}}\bar{\mathcal{X}}^T\bar{\boldsymbol{u}}_i &= \lambda_i\bar{\boldsymbol{u}}_i, \\ (\bar{\boldsymbol{u}},\bar{\boldsymbol{u}}_j)_W &= \delta_{ij} \qquad i,j = 1,\dots,k. \end{aligned} \tag{6.6}$$

Consider $\mathcal{R}^N : \mathbb{R}^n \to \mathbb{R}^n$, a linear, bounded non-negative and self adjoint operator defined as

$$\mathcal{R}^N\mathbf{u} := \sum_{j=1}^{N} \alpha_j(\mathbf{x}(t_j),\mathbf{u})_W\mathbf{x}(t_j) = \mathcal{X}D\mathcal{X}^T W\mathbf{u} \tag{6.7}$$

Thus (6.6) can be written equivalently

$$\mathcal{R}^N\mathbf{u}_i = \lambda_i\mathbf{u}_i, \qquad i = 1,\dots,n.$$

6.2.2 Using POD as a MOR technique

This formulation of POD is given for example in (76, 77).

Consider the general nonlinear state-space model (5.4). Having defined $U_k \in \mathbb{R}^{n\times k}$, consider the following transformation

$$\mathbf{x}(t) \approx U_k\mathbf{a}(t), \qquad \mathbf{a}(t) := (a_1,\dots,a_k)^T \in \mathbb{R}^k,$$

which implies the *reduced order state equation*

$$\Sigma : \begin{cases} \frac{d}{dt}\mathbf{a}(t) &= U_k^* f(U_k\mathbf{a}(t),\mathbf{u}(t)), \\ \mathbf{y}(t) &= g(U_k\mathbf{a}(t),\mathbf{u}(t)), \end{cases} \tag{6.8}$$

a particular case of (5.5): observe in fact that we can interpret POD as a projection, choosing $V = W = U_k$. Thus $\mathbf{a}(t)$ evolves on a *lower-dimensional* space, which is

spanned by the k leading columns of U, i.e. by the leading left singular vectors of $\mathcal{X}$ (125). It is important to note that the reduced model strictly depends on the data $\mathcal{X}$, thus they must reflect the typical operating condition of the system. For example if the state space model (5.4) is obtained discretizing in space a PDE (thus the order of the system n corresponds to the number of degrees of freedom of x), the reduced model can approximate properly only the states $\mathbf{x}(t)$ corresponding to boundary or initial conditions considered collecting data $\mathcal{X}$ (77).

6.2.3 Error estimation of POD technique

As a particular case of (5.4), consider the following dynamical system

$$\begin{aligned} \tfrac{d}{dt}\mathbf{x}(t) &= f(\mathbf{x}(t), \mathbf{u}(t)), \\ \mathbf{x}(0) &= \mathbf{x}_0. \end{aligned} \tag{6.9}$$

Suppose now that $\mathbf{x} \in \mathcal{C}([0,T];\mathbb{R}^n) \cap \mathcal{C}^1([0,T];\mathbb{R}^n)$ is the unique solution of it and $\{\mathbf{u}_i\}_{i=1,\dots,k}$ is the unique POD basis obtained solving

$$\min_{\boldsymbol{\xi}_1,\dots,\boldsymbol{\xi}_k \in \mathbb{R}^n} \int_0^T \left\| \mathbf{x}(t) - \sum_{i=1}^{k} (\mathbf{x}(t), \boldsymbol{\xi}_i)_W \boldsymbol{\xi}_i \right\|_W^2 dt, \qquad s.t. \qquad (\boldsymbol{\xi}_i, \boldsymbol{\xi}_j)_W = \delta_{ij}, \qquad 1 \le i,j \le k. \tag{6.10}$$

Observe that (6.5) can be obtained using a quadrature rule to approximate the time integral, with weights α_j and nodes t_j: *thus α_j must be chosen s.t. the quadrature rule converges to the integral.* For example using the trapezoidal rule we obtain (124) $W = Dx\ diag(\frac{1}{2}, 1, \dots, 1, \frac{1}{2})$, where Dx denotes a uniform step in the space discretization.

The corresponding reduced model thus is

$$\begin{aligned} \tfrac{d}{dt}\mathbf{a}(t) &= U_k^* f(U_k \mathbf{a}(t), \mathbf{u}(t)), \\ \mathbf{a}(0) &= U_k^* \mathbf{x}_0, \end{aligned} \tag{6.11}$$

a particular case of (6.8).

Now we are interested in *estimating the error*

$$\int_0^T \|\mathbf{x}(t) - \hat{\boldsymbol{x}}(t)\|_W^2 \, dt,$$

where $\hat{\boldsymbol{x}}(t) := U_k \mathbf{a}(t)$ (124).

Theorem 6.2.3 *With the above assumptions, it holds*

$$\int_0^T \|\boldsymbol{x}(t) - \hat{\boldsymbol{x}}(t)\|_W^2 \, dt \le C \sum_{i=k+1}^{n} (\lambda_i + \int_0^T |(\dot{\hat{\boldsymbol{x}}}(t), \boldsymbol{u}_i)_W|^2 \, dt)$$

for a constant $C > 0$.

Observe now that from a practical point of view we do not have the information on the whole trajectory in $[0,T]$, i.e. in general we cannot solve the reduced order model analytically. Thus suppose to discretize it in time, using e.g. backward Euler, with a uniform time step Δt denoting with $\hat{\boldsymbol{x}}$ the vector corresponding to the solution of the discretized reduced system, i.e. $\hat{\boldsymbol{x}}_i = \hat{\boldsymbol{x}}(t_i)$, $t_i = i\Delta t$.

We are interested in estimating

$$\sum_{j=1}^{N} \alpha_j \left\| \mathbf{x}(t_j) - \hat{\boldsymbol{x}}_j \right\|_W^2 .$$

Theorem 6.2.4 *With the above assumptions, suppose moreover that $\ddot{\boldsymbol{x}} \in L^2(0,T;\mathbb{R}^n)$ and that $\left\{\boldsymbol{u}_i^N\right\}_{i=1,\dots,k}$ is a POD basis solving (6.5). Then it holds*

$$\sum_{j=1}^{N} \alpha_j \left\| \boldsymbol{x}(t_j) - \hat{\boldsymbol{x}}_j \right\|_W^2 \leq C \left((\Delta t)^2 + \sum_{i=k+1}^{n} (\lambda_i^N + \sum_{j=1}^{N} \alpha_j \left| (\dot{\boldsymbol{x}}(t_j), \boldsymbol{u}_i^N)_W \right|^2) \right),$$

for a constant $C > 0$, if Δt is sufficiently small and f is Lipschitz-continuous with respect to the second argument.

One strength of POD is that models can be efficiently tuned to capture physics in a high-fidelity manner, using samples. However at the same time it is needed to compute samples and there could be a possible lack of model robustness to change in parameters (104). In the following sections we apply POD to two simple 1D problems, to better understand how the choice of the number of modes k is important and depends on the underlying dynamic.

6.3 Examples of application of POD

6.3.1 One dimensional linear heat equation

Consider $x \in \Omega = [0,L]$, $L = 10$, $t \in [0,T]$, $T = 1$, and the one dimensional heat equation

$$\begin{array}{rcll} \frac{\partial T}{\partial t} + c\frac{\partial^2 T}{\partial x^2} & = & 0, & in \qquad [0,T] \times \Omega \\ T(0,x) & = & g_0(x) = 20, & \end{array} \tag{6.12}$$

with different types of boundary conditions:

$$\begin{array}{rll} Neumann-Neumann: & \frac{\partial T}{\partial x}(t,0)=f_0(t)=100; & \frac{\partial T}{\partial x}(t,L)=0; \\ Dirichlet-Dirichlet: & T(t,0)=0; & T(t,0)=0; \\ Dirichlet-Dirichlet: & T(t,0)=10; & T(t,0)=5; \\ Neumann-Dirichlet: & \frac{\partial T}{\partial x}(t,0)=100; & T(t,0)=5; \\ Dirichlet-Neumann: & T(t,0)=10; & \frac{\partial T}{\partial x}(t,L)=0. \end{array} \tag{6.13}$$

Suppose moreover that $c = 10$. Results are plotted in figure (6.2).

If we discretize the heat equation in space, using the finite difference method, with a uniform step $\Delta x = 0.01$ in $[0, L]$, and $\Delta t = 0.01$ in $[0, T]$ we obtain the following linear state space system (5.1):

$$\begin{array}{rcll} \frac{d\mathbf{T}}{dt} & = & cA + b(t), & in \qquad [0,T]\times\Omega \\ \mathbf{T}(0) & = & (g_0(x_i)), & i = 1\ldots, N, \end{array} \tag{6.14}$$

where the matrix A and the vector $b(t)$ represent the discretization of the laplacian and the right-hand side term respectively and $\mathbf{T}(t) = (T(x_i, t))_i$.

Thus we can solve this system of ODE's using e.g. backward euler, and obtain a model, which is useful to construct the matrix of snapshots $\chi = (\mathbf{T}(t_j))$, $j = 1, \ldots, 100$:

$$\begin{array}{rcl} (\mathbb{I}_N - c\Delta x A)\mathbf{T}^{k+1} & = & \mathbf{T}^k + \Delta x b(k+1), \\ \mathbf{T}_0 & = & (g_0(x_i)),\ i = 1\ldots, N, \end{array} \tag{6.15}$$

Then we compute $\chi = USV^* \approx U_k S_k V_k^*$, $U_k = U(:, 1 : k)$, $S_k = S(1 : k, 1 : k)$, $V_k = V(:, 1 : k)$, for a suitable $k << N$, and finally we solve

$$A_r a^{n+1} = A_{0,r} a^n + B_r u^n, \qquad a^{n+1} \in \mathbb{R}^k$$

$A_r = U_k^*(\mathbb{I}_N - c\Delta x A)U_k \in \mathbb{R}^{k\times k}$, $A_{0,r} = U_k^* \mathbb{I}_N U_k = \mathbb{I}_k \in \mathbb{R}^{k\times k}$ and $B_r(k+1) = U_k^* \Delta x b(k+1)$.

$$\mathbf{T} \approx \hat{\boldsymbol{T}} = U_k \mathbf{a}.$$

Using $k = 20$ (2% of N), is sufficient to reconstruct well the dynamic (cfr. figure 6.2).

6.3.2 One dimensional nonlinear heat equation

Consider now the more interesting *nonlinear* reformulation of the heat problem:

$$\begin{array}{rcll} \frac{\partial T}{\partial t} + c(T)\frac{\partial^2 T}{\partial x^2} & = & 0,\ in \qquad [0,T]\times\Omega, \\ c(T) & = & e^{-10T} + 0.01\cdot T^3. \end{array} \tag{6.16}$$

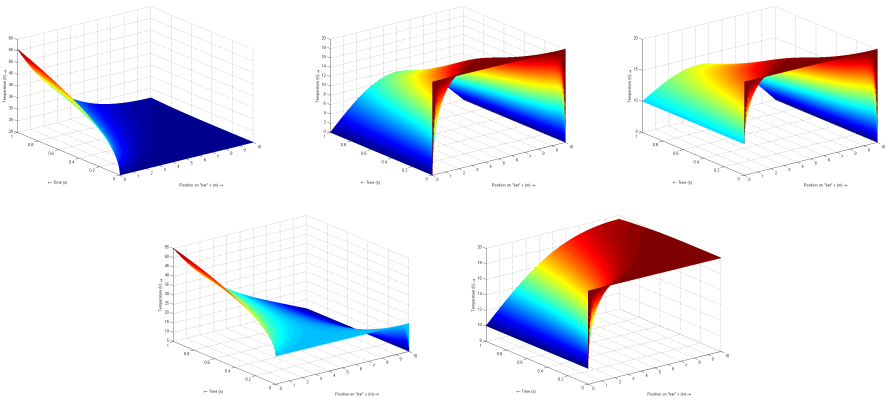

Figure 6.1: *Solution of the heat equation: Neumann-Neumann, Dirichlet-Dirichlet homogeneous; Dirichlet-Dirichlet nonhomogeneous; Neumann-Dirichlet; Dirichlet-Neumann.*

The main difference is that the discretized-in-space model is now

$$\begin{array}{rcll} \frac{d\mathbf{T}}{dt} & = & A(T)T + b(t), & in \qquad [0,T]\times\Omega \\ \mathbf{T}(0) & = & (g_0(x_i)), & i = 1\ldots,N, \end{array} \tag{6.17}$$

and using backward euler we obtain the following *nonlinear* system

$$\begin{array}{rcll} (\mathbb{I}_N - \Delta x A(T^{k+1}))\mathbf{T}^{k+1} & = & \mathbf{T}^k + \Delta x b(k+1), & in \qquad [0,T]\times\Omega \\ \mathbf{T}_0 & = & (g_0(x_i)), & i = 1\ldots,N. \end{array} \tag{6.18}$$

We decided to modify it, solving a semi-implicit problem

$$\begin{array}{rcll} (\mathbb{I}_N - \Delta x A(T^{k}))\mathbf{T}^{k+1} & = & \mathbf{T}^k + \Delta x b(k+1), & in \qquad [0,T]\times\Omega \\ \mathbf{T}_0 & = & (g_0(x_i)), & i = 1\ldots,N. \end{array} \tag{6.19}$$

This is important because we avoid the application of Newton's method, but it is accurate enough for our problem. However at each time step, we must compute the matrix $A(T^k)$. Computing once the SVD of χ, U_k is determined: $A_{0,r} = U_k^*\mathbb{I}_N U_k = \mathbb{I}_k \in \mathbb{R}^{k\times k}$, $A_r(T^k) = U_k^* A(T^k)U_k \in \mathbb{R}^{k\times k}$ and $B_r(k+1) = U_k^*\Delta x b(k+1)$. Neverless, this procedure can be time consuming too, since it is necessary first to build $A(T^k)$ for the full system and then to compute $A_r(T_k)$ for *every* time step. In figure 6.3 we solve the reduced problem with $k = 20$.

A problem related to POD is that it is not robust to changes e.g. in boundary conditions: this means that it is very important to construct a suitable matrix of snapshots χ, which contains useful information.

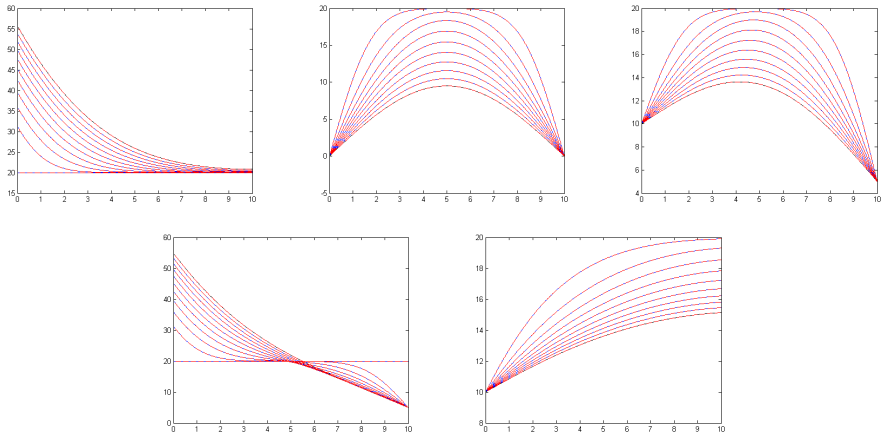

Figure 6.2: *From left to right, from the top to the bottom: Neumann-Neumann, Dirichlet-Dirichlet homogeneous; Dirichlet-Dirichlet nonhomogeneous; Neumann-Dirichlet; Dirichlet-Neumann. Real dynamic (blue), POD (red).*

Suppose for example that we want to solve the nonlinear heat equation presented above, for homogeneous boundary conditions, using $k = 20$, but using the matrix χ_1 obtained solving the same problem with Neumann boundary conditions. The result is very inaccurate (cfr. figure 6.4). A way to solve the problem is to consider more information. For example consider also the nonlinear heat problem with nonhomogeneous boundary conditions, and collect χ_2. Then define χ as the union of these subsets and apply the POD method. Using $k = 20$ the result now is more accurate (cfr. figure 6.4).

Observe that this is a weak nonlinear problem: lets consider now strong nonlinearities (e.g. shocks).

6.3.3 One dimensional Burgers' equation

This example is taken from (116). Our aim is to consider an example of shock movement in a fluid, which is a strongly nonlinear phenomenon. Consider the one dimensional Burgers' equation: $f: \mathbb{R} \to \mathbb{R}$, $f(u) = \frac{u^2}{2}$,

$$\begin{aligned} \frac{\partial u(t,x)}{\partial t} + \frac{\partial f(u(t,x))}{\partial x} &= g(x), \\ u(0,x) &= u_0(x), \\ u(t,0) &= \phi(t), \end{aligned} \tag{6.20}$$

where $u : I \times \Omega \to \mathbb{R}$, $I = [0,T]$, $\Omega = [0,L]$ is the unknown conserved quantity (e.g. mass, density, heat), $u_0(x)$ is the initial condition and $u(t,0) = \phi(t)$ is the inflow one.

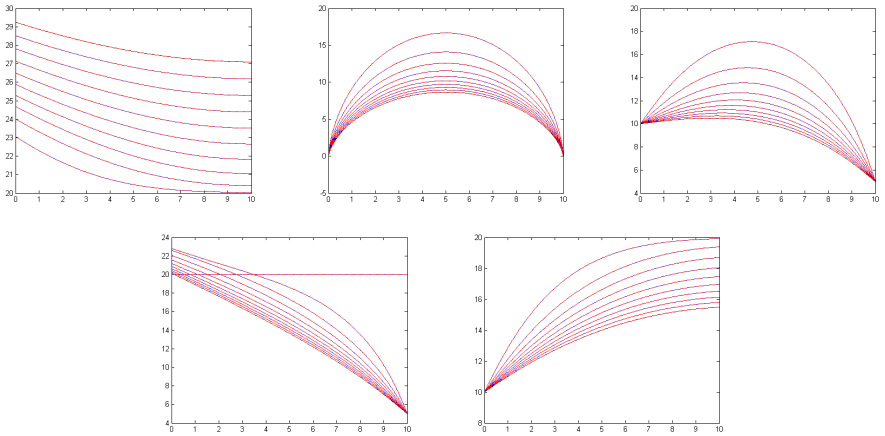

Figure 6.3: *From left to right, from the top to the bottom: Neumann-Neumann, Dirichlet-Dirichlet homogeneous; Dirichlet-Dirichlet nonhomogeneous; Neumann-Dirichlet; Dirichlet-Neumann. Real dynamic (blue), POD (red).*

Using this model, we present two different examples, the first is taken from (116), the second is the motion of the Heaviside function.

6.3.3.1 First example

Consider $g(x) = \xi e^{\xi x}$, $\xi = 0.02$, $u_0(x) \equiv 1$, $\phi(t) \equiv \sqrt{5}$. Moreover we use $\Delta x = 0.1$ and $\Delta t = 0.01$ as space and time discretization steps respectively. Define $x_i = i\Delta x$, $i = 1, \ldots, N$, $N = 1000$ and $\mathbf{U} = (U_i)_i$, $U_i = U(x_i)$. Applying the Godunov's scheme for approximating the space derivative, we obtain the following equation at node $i > 0$:

$$\frac{dU_i}{dx} = -\frac{F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}}}{\Delta x} + g(x_i),$$

where

$$F_{i+\frac{1}{2}} = \begin{cases} \min_{U\in[U_i,U_{i+1}]} f(U), & if \quad U_i < U_{i+1} \\ \max_{U\in[U_i,U_{i+1}]} f(U), & if \quad U_i > U_{i+1} \end{cases}$$

Since $f(U) = \frac{U^2}{2}$, if all $U_i \geq 0$, then $F_{i+\frac{1}{2}} = \frac{U_i^2}{2}$. Discretizing in space Burgers equation and incorporating also boundary conditions, we obtain the following dynamical system:

$$\frac{d\mathbf{U}}{dt} = F(\mathbf{U}) + G + B\phi^2, \tag{6.21}$$

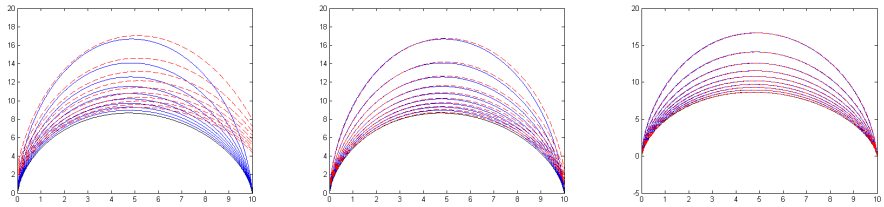

Figure 6.4: *POD reconstruction of Dirichlet-Dirichlet homogeneous boundary conditions using $k = 20$ and χ obtained solving only Neumann-Neumann problem (left), only Dirichlet-Dirichlet nonhomogeneous problem (right), Dirichlet-Dirichlet nonhomogeneous and Neumann-Neumann problems (center). Real dynamic (blue), POD (red).*

$G = (g(xi))_i$, $B = [\frac{1}{2\Delta x}, 0, \ldots, 0]^T \in \mathbb{R}^N$ and

$$F(\mathbf{U}) = \frac{1}{\Delta x} \begin{pmatrix} \frac{-U_1^2}{2} \\ \frac{U_1^2 - U_2^2}{2} \\ \vdots \\ \frac{U_{N-1}^2 - U_N^2}{2} \end{pmatrix}.$$

This *nonlinear* system can be solved e.g. using a ϑ-method:

$$\frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} = \theta F(\mathbf{U}^{n+1}) + (1-\theta) F(\mathbf{U}^n) + G + B\phi^2.$$

We used both forward Euler ($\theta = 0$) and backward Euler ($\theta = 1$): in the last case we must solve a nonlinear system of equations at every iteration n, e.g. applying Newton's method. If

$$\Phi(\mathbf{U}^{n+1}) := \mathbf{U}^{n+1} + \Delta t(F(\mathbf{U}^{n+1}) + G + B\phi^2) - \mathbf{U}^n,$$

$\Phi(\mathbf{U}^{n+1}) \approx \Phi(\mathbf{U}^n) + J\Phi(\mathbf{U}^n)(\mathbf{U}^{n+1} - \mathbf{U}^n)$, thus, to obtain $\mathbf{U}^{n+1}$ we must solve iteratively (until a convergence criterion is satisfied)

$$J\Phi(\mathbf{s}^k)\mathbf{d}^{k+1} = -\Phi(\mathbf{s}^k)$$

$$\mathbf{s}^{k+1} = \mathbf{s}^k + \mathbf{d}^{k+1}$$

starting from $\mathbf{s}^0 = \mathbf{U}^n$, with

$$J\Phi(\mathbf{U}) = \mathbb{I}_N - \frac{\Delta t}{\Delta x} \begin{pmatrix} -U_1 & U_1 & & & \\ & -U_2 & U_2 & & \\ & & \ddots & \ddots & \\ & & & & -U_N \end{pmatrix}.$$

After convergence, $U^{n+1} = \mathbf{s}^{k+1}$.

The solution of a nonlinear system is very expensive, since it involves evaluations both of f, to compute $\Phi(\mathbf{s}^k)$, and of its Jacobian, to compute $J\Phi(\mathbf{s}^k)$. Observe that for nonlinear problems, the reduced system still depends on N. A possible solution is to evaluate $J\Phi(\mathbf{s}^k)$ not at every iteration, but this strategy does not reduced the cost of the algorithm appreciably.

Observe that an alternative way to solve (6.21) is to linearize it *once* around a *fixed* point $\mathbf{U}_0 \in \mathbb{R}^N$ (cfr. figure 6.5), obtaining the following *linear* system

$$\frac{d\mathbf{U}}{dt} = F(\mathbf{U}) + G + B\phi^2 \approx F(\mathbf{U}_0) + JF(\mathbf{U}_0)(\mathbf{U} - \mathbf{U}_0) + G + B\phi^2,$$

and the corresponding time discretization

$$\frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} = F(\mathbf{U}_0) + JF(\mathbf{U}_0)(\mathbf{U}^n - \mathbf{U}_0) + G + B\phi^2.$$

The problem is that in general linearized dynamics are suitable only if nonlinearities

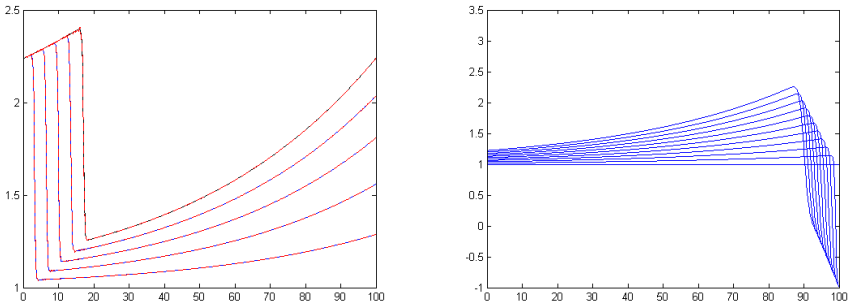

Figure 6.5: *Left: evolution of the solution of the first example of Burgers' equation (blue) and POD reconstruction (red), $k = 50$. Right: linearization of the problem around the initial point. It does not represent the dynamic.*

are stationary (104) and is *unable* to reproduce the nonlinear dynamic, e.g. the time propagation of the shock in Burgers' equation. In fact the previous Taylor approximation is accurate only if $\mathbf{U} - \mathbf{U}_0$ is small enough: suppose that at time n $\mathbf{U}^n - \mathbf{U}_0 < \epsilon$, where $\epsilon > 0$ is a suitable threshold. Nobody could guarantee us that $\mathbf{U}^l - \mathbf{U}_0 < \epsilon$ for every $l > n$, because the nonlinearity is rapidly propagating in time, and $\mathbf{U}^l$ could be distant from $\mathbf{U}^n$, even if $l - n$ is small. When the nonlinearities are moving a possible solution is to consider $\mathbf{U}_0$ a time function, $\mathbf{U}_0 : \ I \to \mathbb{R}^N$, $\mathbf{U}_0 = \mathbf{U}_0(t)$. Thus we deal with a constant re-linearization:

$$\frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} = F(\mathbf{U}_0^n) + JF(\mathbf{U}_0^n)(\mathbf{U}^n - \mathbf{U}_0^n) + G + B\phi^2,$$

s.t. $\mathbf{U}^l - \mathbf{U}_0^l < \epsilon$, for every $l \geq 0$ but these techniques are impractical, due to computational costs.

In figure (6.6) are represented solutions using forward Euler ($\theta = 0$), for the time discretization: although it is considerably less expensive, it needs a greater k: $k = 80$ instead of $k = 50$.

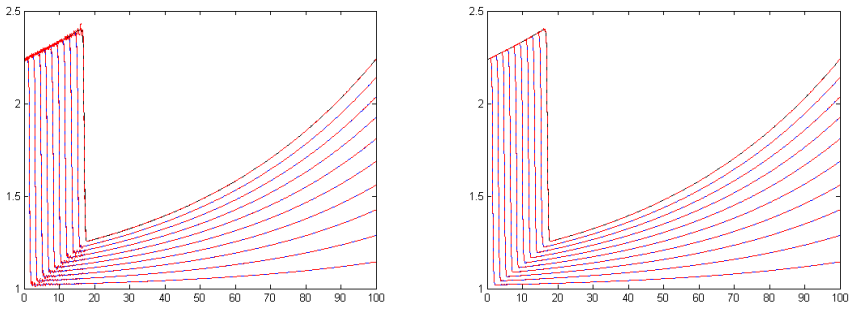

Figure 6.6: *Left: evolution of the solution of the first example of Burgers' equation (blue) and POD reconstruction (red) using forward euler, $k = 50$. Right: $k = 80$.*

Due to time efficiency, we can try to approximate the model increasing T, $T = 5000$ (cfr. figure 6.7), selecting time instants to define χ (e.g. every 5 time istants).

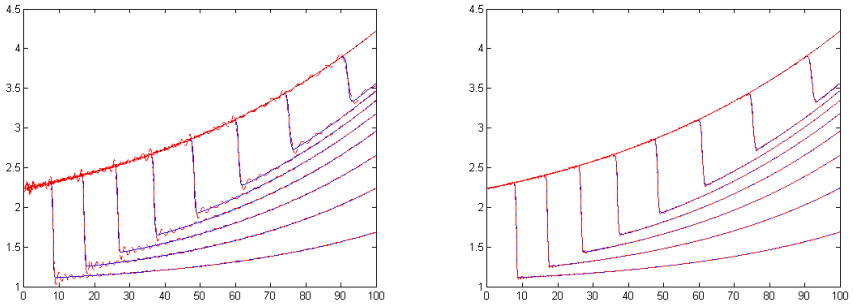

Figure 6.7: *Dynamic for $T = 5000$: $k = 80$ (left), $k = 160$ (right) (116).*

Observe that in general in presence of time varying shocks it is important to consider a sufficiently rich data set, to capture all shocks dynamics: this corresponds to the need of an higher number k of eigenfunctions.

6.3.3.2 Second example

The procedure is analogue to the previous example, but we consider different data: $g(x) \equiv 0$, $u_0(x) = \begin{cases} 1, & if \quad x \leq j \\ 0, & if \quad x > j \end{cases}$, $\phi(t) \equiv \sqrt{5}$. Moreover we use $\Delta x = 0.1$ and $\Delta t = 0.1$ as discretization space and time steps respectively.

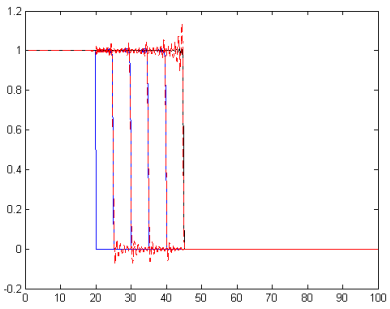

Figure 6.8: *Evolution of the solution of the second example of Burgers' equation (blue) and POD reconstruction (red),* $k = 50$.

In figure (6.9) are represented solutions using forward Euler ($\theta = 0$), for the time discretization. Although k is considerably greater, it is convenient to use forward euler, because much less time consuming.

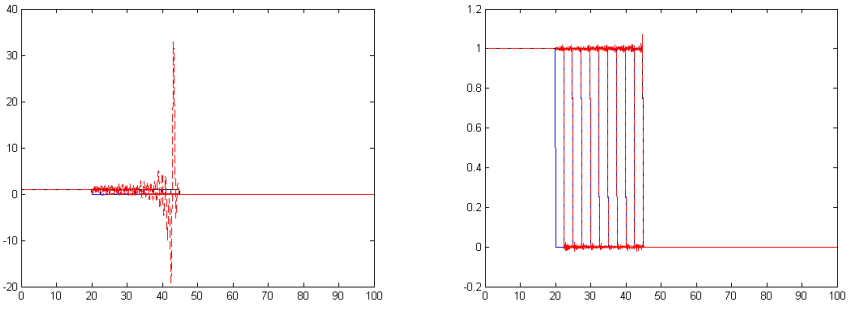

Figure 6.9: *Left: evolution of the solution of the second example of Burgers' equation (blue) and POD reconstruction (red) using forward euler,* $k = 50$. *Right:* $k = 200$.

6.4 POD in an infinite dimensional context

In this section we present the main ideas of the POD formulation in an infinite dimensional context: the corresponding algebraic reduced system is equivalent to the one introduced in section 6.2 (cfr. appendix B): for more details we refer to (124).

Let V be a real separable Hilbert space: consider $a: V \times V \to \mathbb{R}$ a bilinear, symmetric bounded and coercive form; $b: V \times V \to V'$ bilinear and continuous and $R: V \to V'$ linear and continuous. For given $f \in \mathcal{C}([0,T];V)$, $y_0 \in V$ we state the nonlinear evolution problem: find $y(t) \in V$ s.t. $\forall \phi \in V$

$$\begin{array}{rcll} \frac{d}{dt}(y(t),\phi)_V + a(y(t),\phi) + \langle b(y(t),y(t)) + Ry(t),\phi\rangle_{V',V} & = & (f(t),\phi)_V, & for\ a.e.\ t \in (0,T] \\ y(0) & = & y_0 & in\ V. \end{array} \tag{6.22}$$

Suppose that b and R satisfy suitable assumptions (94, 124) such that for every $f \in \mathcal{C}([0,T];V)$ and $y_0 \in V$, (6.22) has a unique solution y satisfying $y \in \mathcal{C}([0,T];V) \cap L^2([0,T];V) \cap H^1([0,T];V)$.

For given $N \in \mathbb{N}$, let $0 = t_0 < t_1 < \ldots < t_N \leq T$ denote a grid in $[0,T]$ and denote by Δt and δt the maximum and minimum step respectively. Suppose to know the snapshots $y(t_j)$, taken in t_j, $j = 0, \ldots, N$. Define

$$\mathcal{V} = span\,\{y(t_0), \ldots, y(t_N)\} \subset V$$

the ensamble of snapshots.

Let $\{\psi_i\}_{i=1}^d$ denote an *orthonormal basis* of $\mathcal{V}$, $d = dim\mathcal{V}$: thus

$$y = \sum_{i=1}^{\infty} (y, \psi_i)_V \psi_i.$$

For every $k \leq d$, the POD method consists in choosing an orthonormal basis of $\mathcal{V}$ solution of the following optimization problem:

$$\begin{array}{ll} \min_{\psi_1 \ldots, \psi_k \in V} & \sum_{j=0}^{N} \alpha_j \left\| y(t_j) - \sum_{i=1}^{k} (y(t_j), \psi_i)_V \psi_i \right\|_V^2, \\ & (\psi_i, \psi_j)_V = \delta_{ij}, \qquad 1 \leq i, j \leq k, \end{array} \tag{6.23}$$

for suitable weights α_j: e.g. they can be thought as weights of a quadrature rule for the approximation of the integral $\int_0^T \left\| y(t) - \sum_{i=1}^{k} (y(t), \psi_i)_V \psi_i \right\|_V^2 dt$, using $\{t_0, \ldots, t_N\}$ as temporal nodes.

$\{\psi_1 \ldots, \psi_k\}$ is called *POD basis of rank k*.

6. PROPER ORTHOGONAL DECOMPOSITION (POD) METHOD

Consider now the following bounded linear operator

$$\mathcal{Y}_N : \mathbb{R}^{N+1} \to V, \qquad \mathcal{Y}_N v := \sum_{j=0}^{N} \alpha_j v_j y(t_j).$$

The corresponding adjoint $\mathcal{Y}_N^* : V \to \mathbb{R}^{N+1}$ is given by

$$\mathcal{Y}_N^* z = \left(\langle z, y(t_0)\rangle_V, \ldots, \langle z, y(t_n)\rangle_V\right)^T.$$

Thus the bounded and linear operator on V

$$\mathcal{R}_N = \mathcal{Y}_N \mathcal{Y}_N^*,$$

and the matrix

$$\mathcal{K}_N = \mathcal{Y}_N^* \mathcal{Y}_N \in \mathbb{R}^{(N+1)\times(N+1)}$$

are given by

$$\mathcal{R}_N z = \sum_{j=0}^{N} \alpha_j (z, y(t_j))_V y(t_j), \qquad z \in V$$

and

$$(\mathcal{K}_N)_{ij} = \langle y(t_j), y(t_i)\rangle_V.$$

Since $\mathcal{R}_N$ is bounded, self-adjoint and non-negative, with finite dimensional image, it is also compact (124). Thus, by Hilbert-Schmidt theory, there exists an orthonormal basis $\{\psi_i\}_{i\in\mathbb{N}}$ of V and a non-negative sequence of decreasing real numbers $\{\lambda_i\}_{i\in\mathbb{N}}$ such that

$$\begin{aligned} \mathcal{R}_N \psi_i &= \lambda_i \psi_i, \\ \lambda_i &= 0, \ i > d, \end{aligned} \tag{6.24}$$

moreover $\mathcal{V} = span\{\psi_i\}_{i\le d}$. Using the Lagrangian framework (124) it follows that $V^k = span\{\psi_i\}_{i\le k}$ is the optimal solution of (6.23).

Proposition 6.4.1 *Under the above assumptions, $\{\psi_i\}_{i\le k}$, chosen such that (6.24) holds, is a POD basis of rank $k \le d$ and*

$$\sum_{j=0}^{n} \alpha_j \left\| y(t_j) - \sum_{i=1}^{k} (y(t_j), \psi_i)_V \psi_i \right\|_V^2 = \sum_{i=l+1}^{d} \lambda_i.$$

Remark 6.4.1 *Setting*

$$v_i = \frac{1}{\sqrt{\lambda_i}} \mathcal{Y}_N^* \psi_i, \ i = 1, \ldots, d$$

we find $\mathcal{K}_N v_i = \lambda_i v_i$ *and* $\langle v_i, v_j \rangle_{\mathbb{R}^{N+1}} = \delta_{ij}$, *i.e.* $\{v_i\}_{i=1}^d$ *are an orthonormal basis of eigenvectors of* $\mathcal{K}_N$.

Conversely, given $\{v_i\}_{i=1}^d$, *then*

$$\psi_i = \frac{1}{\sqrt{\lambda_i}} \mathcal{Y}_N v_i, \ i = 1, \dots, d.$$

It is important to note that the last equation is a generalization of the ***method of snapshots*** *(6.3) introduced in the finite dimensional context.*

More details can be found in (124), e.g. what happens if $\Delta t \to 0$.

Thus, given $f \in \mathcal{C}([0,T]; V)$ and $y_0 \in V$, the *POD-Galerkin reduction* of (6.22) consists in finding $\hat{y}(t) \in V^k$ s.t. $\forall \psi \in V^k$

$$\begin{aligned} \tfrac{d}{dt}(\hat{y}(t), \psi)_V + a(\hat{y}(t), \psi) + \langle b(\hat{y}(t), \hat{y}(t)) + R\hat{y}(t), \psi \rangle_{V',V} &= (f(t), \psi)_V, \qquad \text{a.e. } t \in (0,T] \\ (\hat{y}(0), \psi)_V &= \hat{y}_0. \end{aligned} \tag{6.25}$$

Remark 6.4.2 *Assuming that* $y = \sum_{i=1}^k \alpha_i \psi_i$ *(POD-Galerkin ansatz), substituting it in (6.22), the POD-Galerkin system (6.25) is equivalent to find* $\boldsymbol{\alpha} \in \mathbb{R}^k$ *s.t.* $\forall \psi_i \in V^k$

$$\frac{d}{dt}\alpha_i + \sum_j \alpha_j a(\psi_j, \psi_i) + \sum_j \sum_s \alpha_j \alpha_s \langle b(\psi_j, \psi_s), \psi_i \rangle + \sum_j \alpha_j \langle R\psi_j, \psi_i \rangle = (f(\tau_l), \psi_i)_V, \qquad \forall \psi_i \in V^k. \tag{6.26}$$

Suppose now that $y = y_m + \sum_{i=1}^k \beta_i \tilde{\psi}_i$, *where* $\left\{\tilde{\psi}_i\right\}$ *is the POD basis of the transformed snapshots matrix* $\{y(t_j) - y_m\}$, *for a fixed* $y_m \in V$. *Thus* $y(t) = (y(t) - y_m) + y_m =: \tilde{y} + y_m$. *Suppose that* y_m *is chosen such that* $\tilde{\psi}_i$ *are all solenoidal and* $y(t) - y_m$ *verifies homogeneous Dirichlet boundary conditions: e.g. consider* $y_m = \frac{1}{N}\sum_{j=1}^N y(t_j)$, *independent on time. Thus the POD-Galerkin problem consists in finding* $\boldsymbol{\beta} \in \mathbb{R}^k$ *s.t.* $\forall \tilde{\psi}_i$

$$\begin{aligned} \tfrac{d}{dt}\beta_i &+ \textstyle\sum_j \beta_j \left(a(\tilde{\psi}_j, \tilde{\psi}_i) + \underline{\left\langle b(\tilde{\psi}_j, y_m) + b(y_m, \tilde{\psi}_j), \tilde{\psi}_i \right\rangle} + \left\langle R\tilde{\psi}_j, \tilde{\psi}_i \right\rangle \right) + \\ &+ \textstyle\sum_{js} \beta_j \beta_s \left\langle b(\tilde{\psi}_j, \tilde{\psi}_s), \tilde{\psi}_i \right\rangle = \\ &= (f(\tau_l), \tilde{\psi}_i)_V - \underline{a(y_m, \tilde{\psi}_i) - \left\langle Ry_m, \tilde{\psi}_i \right\rangle - \left\langle b(y_m, y_m), \tilde{\psi}_i \right\rangle}, \end{aligned} \tag{6.27}$$

then $y(t) = \tilde{y}(t) + y_m$. *The advantage is that this system has homogeneous Dirichlet boundary conditions, thus it is less costly to solve.*

In (93) the more general case of time varying parameter dependent boundary conditions is treated.

To discretize (6.25) in time, consider another grid in $[0,T]$ $0 = \tau_0 < \tau_1 < \ldots < \tau_m = T$ and denote by $\Delta\tau$ and $\delta\tau$ the maximum and minimum step respectively. Using the backward euler method we solve the following POD-Galerkin problem:
find a sequence $\{Y_l\}_{l=0}^m \in V^k$ s.t.

$$\begin{aligned} (\tfrac{Y_l - Y_{l-1}}{\delta\tau_l}, \psi)_V + a(Y_l, \psi) + \langle b(Y_l, Y_l) + RY_l, \psi\rangle_{V',V} &= (f(\tau_l), \psi)_V, \quad \forall \psi \in V^k \\ (Y_0, \psi)_V &= (y_0, \psi)_V, \qquad \forall\ \psi \in V^k. \end{aligned} \tag{6.28}$$

If $\Delta\tau$ is sufficiently small, the sequence $\{Y_l\}_{l=0}^m$ is uniquely determined.

Theorem 6.4.1 *(Error estimation) If y is regular enough (124) and if $\Delta\tau$ is sufficiently small, there exists a constant C depending only on T, s.t.*

$$\begin{aligned} \textstyle\sum_{j=0}^m \beta_j \left\| y(\tau_j) - Y_j \right\|_H^2 &\leq C \textstyle\sum_{i=k+1}^d \left(|(\psi_i, y_0)_V|^2 + \frac{\sigma_n}{\delta t} \left(\frac{1}{\delta\tau} + \Delta\tau \right) \lambda_i \right) \\ &+ C\sigma_n \Delta\tau \Delta t \left\| y_t \right\|_{L^2(0,T;V)}^2 \\ &+ C\sigma_n (1 + c_P^2) \Delta\tau \left(\Delta t \left\| y_t \right\|_{L^2(0,T;H)}^2 + (\Delta t + \Delta\tau) \left\| y_{tt} \right\|_{L^2(0,T;H)}^2 \right) \end{aligned}$$

where σ_n is related to the two temporal grids, c_P is such that $\left\| \mathcal{P}^k \right\|_{L(V)} \leq c_P$ and β_j are suitable weights.

Observe that the estimate depends (through ψ_i, d) on the way in which the snapshots are taken, on the number k of basis elements and on the relative location of the snapshots and the time discretization (through σ_n). The linear and semi-nonlinear case is treated in (99), while the nonlinear case is treated in (100).

6.5 POD applied to Computational Fluid Dynamics (CFD) problems

POD is largely used to solve inverse or optimal control *fluid problems*: in (163) it is observed that the control of fluid flows is an important area of technological and scientific research having important applications in industrial processes, such as the viscous drag reduction to minimize the drag force on a submerged body or the control of mixing patterns in chemical reactors to enhance the reactor performance by using several control mechanisms. Also in (114, 115) is presented the application of POD as a reduced order technique useful to solve an optimal control problem.

Following e.g. (94), the 2D incompressible Navier Stokes equations can be seen as a particular case of (6.22), as proved in (122): if boundary data are sufficiently smooth, there exists a unique solution. In (94) an optimal control problem of a flow around a

cylinder is taken as example. Given the velocity trajectories $\{\mathbf{u}_1, \ldots, \mathbf{u}_N\}$ the POD basis $\{\psi_1, \ldots, \psi_N\}$ is computed and the ansatz $\mathbf{u}_i = \bar{u} + \sum_{i=1}^{N} a_i \psi_i$ is considered, where $\bar{u}$ is the mean of the snapshots (cfr. Remark 6.4.2). Thus the reduced model has the following form:

$$\langle \mathbf{u}_t, \psi_j \rangle_V + \nu \langle \nabla \mathbf{u}, \nabla \psi_j \rangle_V + \langle (\mathbf{u} \cdot \nabla)\mathbf{u}, \psi_j \rangle_V = \langle f, \psi_j \rangle_V ,$$

for a suitable f. Observe that in (94) pressure is not taken into account, because the authors assume that ψ_j is solenoidal, as a linear combination of velocities, thus $\langle \nabla p, \psi_j \rangle_V = - \langle p, div \psi_j \rangle_V = 0$. We underline that from a numerical point of view velocities are not solenoidal, due to approximation errors, thus not considering the pressure could be a possible source of instabilities: in the following we will considered also pressure in the reduced model. In fact in (109) it is demonstrated how the accuracy of empirical Galerkin models for shear flows can be significantly improved by introducing an appropriate pressure-term representation. Also in (121) it is underlined the need of considering the pressure in the model, to correct velocity predictions. In (106) the approach of (100) and (94) has been extended: both velocity and pressure fields are simultaneously approximated, and the pressure appears in the reduced system:

$$\begin{aligned} \left\langle \mathbf{u}_t, \psi_j^u \right\rangle_{V_u} + \nu \left\langle \nabla \mathbf{u}, \nabla \psi_j^u \right\rangle_{V_u} + \left\langle (\mathbf{u} \cdot \nabla)\mathbf{u}, \psi_j^u \right\rangle_{V_u} + \left\langle -\nabla p, \nabla \psi_j^u \right\rangle_{V_u} &= \left\langle f, \psi_j^u \right\rangle_{V_u} , \\ \left\langle div \mathbf{u}, \nabla \psi_j^p \right\rangle_{V_p} &= 0; \end{aligned}$$

where $(\mathbf{u}, p) \in V = V_u \times V_p$.

The standard POD formulation has been analyzed and modified by different authors, expecially when inverse or optimal control problems are considered: in the following we mention some proposed strategies. In (114) the idea is to improve the POD base through an adaptive procedure, beginning with an ensemble and deriving a POD base which is then used to compute a control. A new ensemble is next generated by applying the control to the original flow model. The new ensemble is used to replace the current and a new POD base is computed. This process is repeated until convergence is achieved.

In (80) optimal control theory is used to minimize the total mean drag for a circular cylinder wake flow in the laminar regime ($Re = 200$). To adapt the POD basis to changes in physics when the flow is altered by a control, a first *a priori* approach consists in distributing uniformly in the control parameter space the snapshot ensemble to be used for POD. However, in this case, a lot of runs of the high-dimensional code would be necessary to generate the snapshots (81). The second *a posteriori* approach consists in an adaptive method in which new snapshots are regularly determined during the

optimization process when the effectiveness of the existing POD to represent accurately the controlled flow is considered to be insufficient (80).

In (110) a viscous flow in a two-dimensional grooved cavity is considered, when the lid velocity varies.

POD method has been largely used also to study *turbulence* (121): POD may be applied to obtain a basis that captures more of the kinetic energy of the system on average, since POD modes are optimal in the sense of capturing, on average, the greatest possible fraction of total energy for a projection onto a given number of modes. Thus relatively high ($O(1000)$) dimensional projections onto POD modes can capture observed modal energy budgets and provide acceptable short-term tracking of individual solutions for turbolent flows. In applications, however, one often wants to consider much lower ($O(10)$) dimensional projections: with suitable modeling of the neglected modes, very low dimensional models are able to capture many aspects of turbolent flows (121).

Weighted POD is a variation of POD which gives more weight to some members of the snapshot set and can be accomplished, e.g., by including multiple copies of an "important" snapshot in the snapshot set. In (86) it is observed that, while POD gives a good description of the structure for a fixed dynamical system, problems occur when *parameter variations* are included (for example considering dynamical systems depending on the parameter Re). Different modes may be important at different parameter values or events may be short in time.

POD with derivatives get more information into the snapshot set in order to get a better POD basis, adding time derivatives of simulated states to the snapshot set.

H^1 *POD* change the error measure for POD using H^1 norms and inner products (instead of L^2) in the definition and construction of POD bases (89).

Constrained POD impose a constraint (e.g. symmetry) on the POD basis.

Adaptive POD change the POD basis when it no longer seems to be working, requires detection of failure of the POD basis, the determination of new snapshot vectors, and the computation of the SVD for the new snapshot matrix determined from the new snapshot vectors.

In (108) the Galerkin-POD reduction of the circular cylinder test is carefully examined and the *shift-mode* is included to significantly improve the resolution of the transient dynamics from the onset of vortex shedding to the periodic von Karman vortex street.

In (77) it is proposed the *Missing Point Estimation* (MPE) strategy: the Galerkin projection is conducted only on equations describing the dynamics of several points in

the spatial domain instead of the equations of all grid points.

To recover the effects of the truncated modes, that is generally of the small scales, in (112) *eddy viscosities* are used, i.e. the viscous terms of the POD-Galerkin system are perturbed. An alternative is *calibrated reduced-order POD-Galerkin* method presented in (85) and tested on two examples (2D square obstacle, Re=100 and 3D backward facing step, Re 7432, considering *periodic regimes*). To correct the behavior of a low-order POD-Galerkin system, the polynomial coefficients which define the POD-Galerkin system are adjusted by solving a minimization problem.

Finally in (83) and (84) the *POD-Discrete Empirical Interpolation Method* (DEIM) is presented, to reduce the POD's complexity for computing a projected nonlinear term, which still depends on the dimension of the original full-order system.

6.6 POD applied to the Navier Stokes problem

In this section we describe with more details the POD reduction of Navier Stokes equations that we have adopted.

First of all collect snapshots: let $\{t_1, \dots, t_N\}$ be a subdivision of the time interval $[0, T]$ and $(\mathbf{u}_1(t_i), \mathbf{u}_2(t_i)) \in \mathbb{R}^{Nu\times 2}$, $\mathbf{p}(t_i) \in \mathbb{R}^{Np}$ be respectively the velocity and the pressure FE solutions at time t_i. Consider

$$\chi_1 = (\mathbf{u}_1(t_j))_{j=1,\dots,T} \in \mathbb{R}^{Nu\times T}, \qquad \chi_2 = (\mathbf{u}_2(t_j))_{j=1,\dots,T} \in \mathbb{R}^{Nu\times T} \qquad \chi_p = (\mathbf{p}(t_j))_{j=1,\dots,T} \in \mathbb{R}^{Np\times T}$$

the matrices of snapshots of the first component of velocity, of the second one and of the pressure, respectively.

To obtain the POD basis we compute their SVD's: denoting with U_1, U_2 and U_p the corresponding matrices of left singular values, we truncate them, defining the projection spaces

$$U_{r1} = U_1(:, 1 : k_1),\ U_{r2} = U_2(:, 1 : k_2),\ U_{rp} = U_p(:, 1 : m),$$

choosing appropriate thresholds k_1, k_2 and k_p, depending on the magnitude of the corresponding singular values. We present now two possible reductions, corresponding to the two treatments of the nonlinear convection term presented in section 4.3.

6.6.1 Explicit treatment of the nonlinear term: reduced system

Consider system (4.32): compute the reduced matrices

$$\mathcal{A}_{k_1,1} = U_{r1}^T \mathcal{A} U_{r1},\ \mathcal{A}_{k_2,2} = U_{r2}^T \mathcal{A} U_{r2},\ B_{k_1,1} = U_{rp}^T B_1 U_{r1},\ B_{k_2,2} = U_{rp}^T B_2 U_{r2}.$$

Thus the reduced system is

$$\begin{pmatrix} \mathcal{A}_{k_1,1} & 0 & -B_{k_1,1}^T \\ 0 & \mathcal{A}_{k_2,2} & -B_{k_2,2}^T \\ -B_{k_1,1} & -B_{k_2,2} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_p \end{pmatrix} = \begin{pmatrix} U_{r1}^T \mathbf{F}_1 \\ U_{r2}^T \mathbf{F}_2 \\ U_{rp}^T \mathbf{F}_p \end{pmatrix}. \tag{6.29}$$

Finally

$$\mathbf{u}_1 \approx U_{r1}\mathbf{a}_1,\ \mathbf{u}_2 \approx U_{r2}\mathbf{a}_2,\ \mathbf{p} \approx U_{rp}\mathbf{a}_p.$$

Observe that the reduced system can be solved directly (e.g. LU-factorization) or using the *Pressure-matrix method* as the unreduced one. Moreover observe that, if $\mathcal{A}$, B_1 and B_2 does not depend on t the numerical method could be optimized, computing the reduced matrices only once at the beginning.

6.6.2 Semi-implicit treatment of the nonlinear term: reduced system

Consider now the system (4.34): now at each iteration the system matrix must be reduced, computing

$$\mathcal{A}_{k_1,1}^{(n)} = U_{r1}^T \mathcal{A}^{(n)} U_{r1},\ \mathcal{A}_{k_2,2}^{(n)} = U_{r2}^T \mathcal{A}^{(n)} U_{r2}.$$

Thus the reduced system is

$$\begin{pmatrix} \mathcal{A}_{k_1,1}^{(n)} & 0 & -B_{k_1,1}^T \\ 0 & \mathcal{A}_{k_2,2}^{(n)} & -B_{k_2,2}^T \\ -B_{k_1,1} & -B_{k_2,2} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_p \end{pmatrix} = \begin{pmatrix} U_{r1}^T \mathbf{F}_1^n \\ U_{r2}^T \mathbf{F}_2^n \\ U_{rp}^T \mathbf{F}_p^n \end{pmatrix}. \tag{6.30}$$

Finally

$$\mathbf{u}_1 \approx U_{r1}\mathbf{a}_1,\ \mathbf{u}_2 \approx U_{r2}\mathbf{a}_2,\ \mathbf{p} \approx U_{rp}\mathbf{a}_p.$$

Here we need to compute at every iteration the system matrices.

Although the semi-implicit method could be useful when dealing with long intervals, the explicit treatment of the convection term, in combination with the deviation from the mean velocity field (cfr. Remark 6.4.2), is much less expensive as a reduction technique, as in the unreduced case. Thus a possible strategy should be the following: if we are interested in analyzing the dynamic in a reference interval $[t_i, T]$, $t_i > 0$ (e.g. where the period regime is achieved), we start applying the semi-implicit method in $[0, t_i]$, with a bigger time step, and then continue applying the explicit one in $[t_i, T]$, using a smaller step.

6.6.3 Application of POD to the backward facing step problem

In this section we see how this reduction works, using as a model the POD reduction of the backward facing step problem, presented in section 4.4.1, considering $t \in [0, 75]$.

6.6.3.1 Modes computation

In figure 6.11 the first 8 singular vectors (*POD modes*) ($U1(:, 1:8)$ and $U2(:, 1:8)$) of the matrix of all snapshots, are plotted. As can be seen, the horizontal and vertical components are close to each other. Observe that the bigger the index of the singular vector in the basis is, the smaller the corresponding structures are: these modes represent the main dynamics in the data set. In figure 6.10 are plotted the singular values of the matrices of snapshots for both components of velocity and pressure. Some values are given in table 6.1.

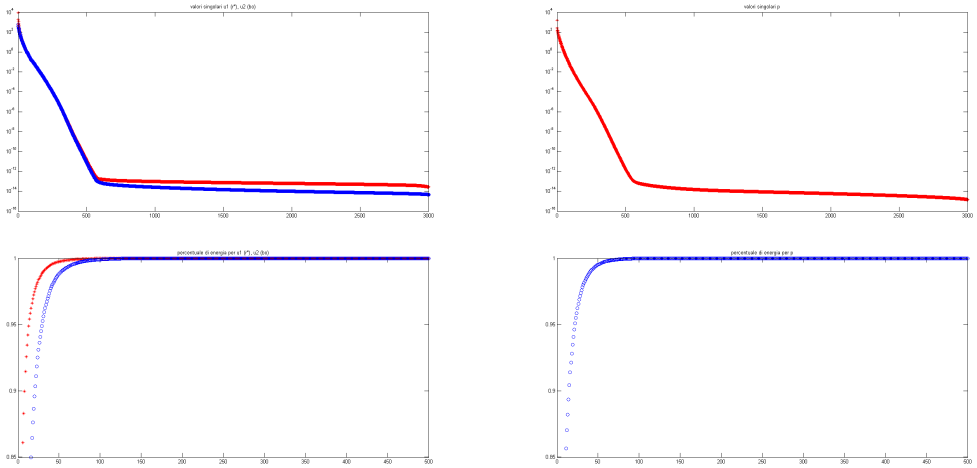

Figure 6.10: *$Re = 400$: First row: Singular values, second row: percentage of energy ($e(k) := \frac{\sum_{j=1}^{k} \sigma_j}{\sum_{j=1}^{n} \sigma_j}$), Left: velocity; Right: pressure*

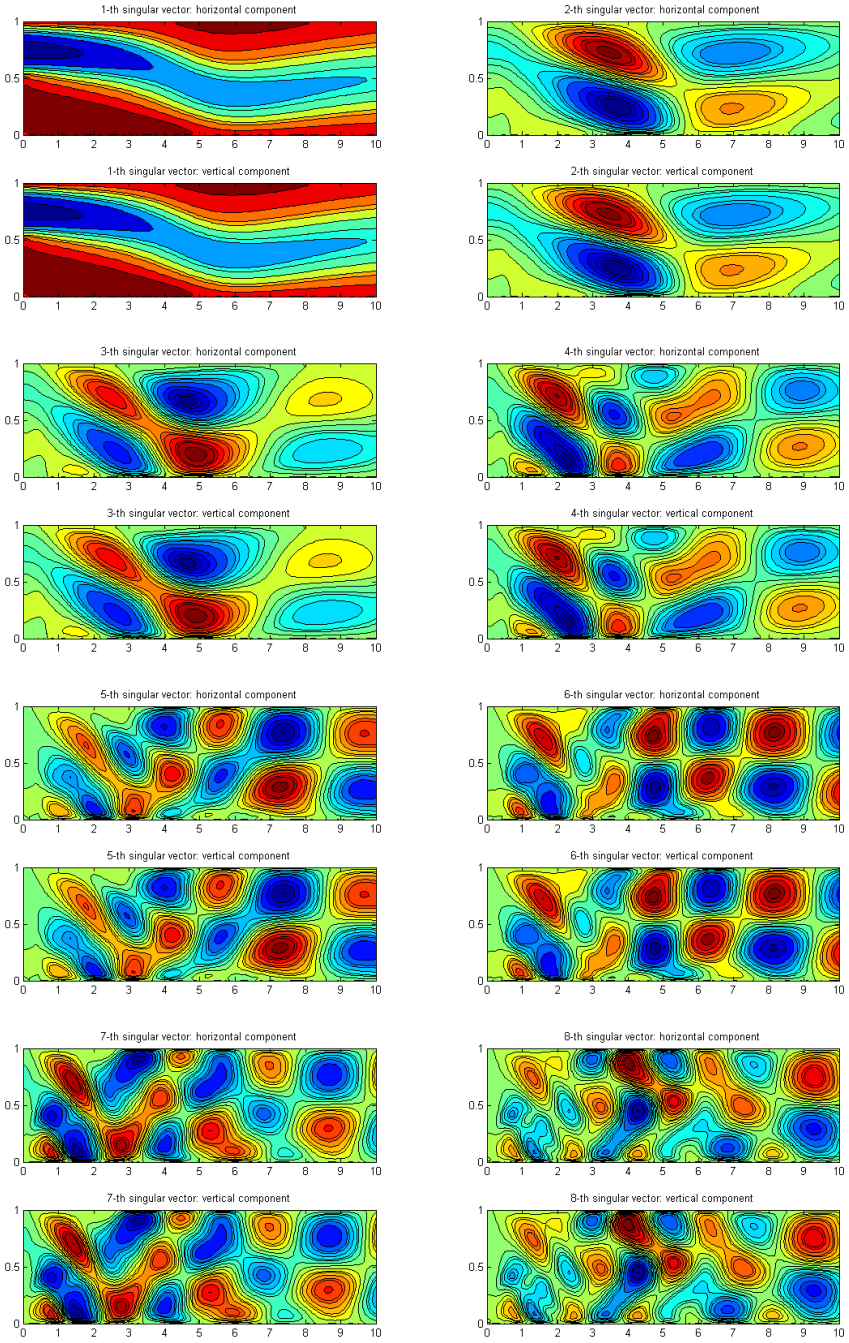

Figure 6.11: *$Re = 400$: First eight singular (POD) vectors computed using two different SVD's for the velocity ($\mathcal{X}_1$ and $\mathcal{X}_2$).*

$k_1,\ k_2,\ k_p$	Percentage of energy of u_1	Percentage of energy of u_2	Percentage of energy of p
1	0.558390252354875	0.166134151527396	0.510498357703101
10	0.925686479743807	0.723035393116410	0.840069110689584
50	0.997457932701127	0.988205126770044	0.995215497553616
100	0.999792472546980	0.999054839808156	0.999826401003764
150	0.999981270336716	0.999905978307868	0.999989253036818
200	0.999998351854391	0.999992115103012	0.999999162219908
250	0.999999881137085	0.999999470928219	0.999999935672432
300	0.999999993949290	0.999999974323508	0.999999996448935
350	0.999999999790786	0.999999999225040	0.999999999852688
400	0.999999999992858	0.999999999976311	0.999999999994602
450	0.999999999999674	0.999999999999071	0.999999999999784
500	0.999999999999986	0.999999999999962	0.999999999999990

Table 6.1: *Percentage of energy for various k_1, k_2 and k_p.*

6.6.3.2 Criteria to evaluate POD's performance

First of all we apply POD without truncation to estimate the maximum level of accuracy reachable: the result is depicted in figure 6.12 and is approximately 10^{-11} in $\mathbb{R}^N$.

Given the matrices of snapshots, to evaluate POD's performance truncating at k_1, k_2 and k_p, the first strategy consists in computing the L^2-error in $\mathbb{R}^n$, i.e. computing directly $\left\|\mathbf{u}^e - \mathbf{u}^{pod}\right\|_2$, and $\left\|p^e - p^{pod}\right\|_2$, where the apex e stands for *exact solution* while *pod* stands for *POD approximation*. Consider the backward facing step test case in $[0, 75]$: it can be seen that the dynamic is still well represented, although the L^2 error is not very small in the last part, as depicted in figure 6.13. Thus the error could not be a good indicator.

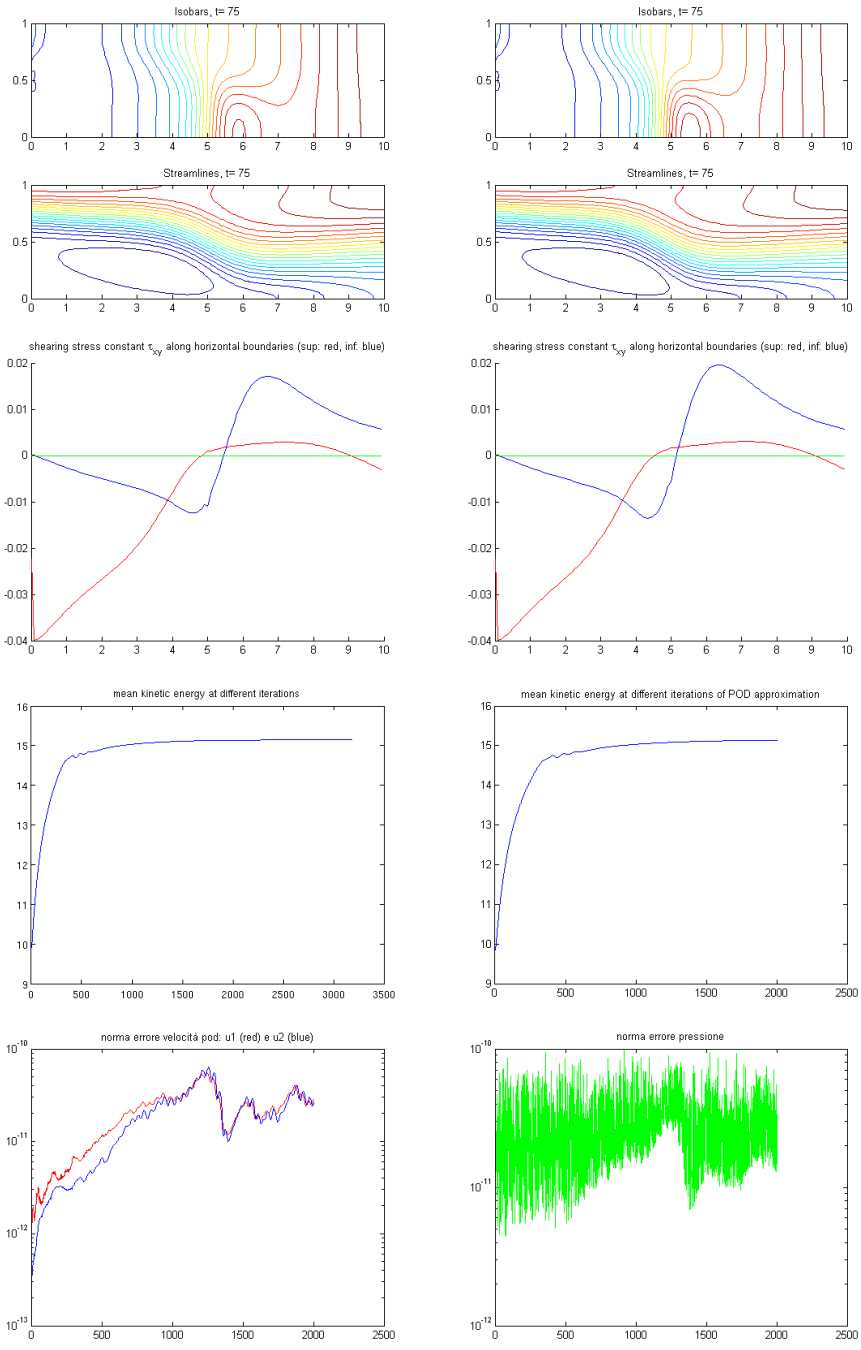

Figure 6.12: *$Re = 400$: Application of POD without truncation. First row Left: "Real streamlines and pressure contour plots of the semi-implicit NS stationary solution". First row Right: streamlines and pressure contour plots of POD NS stationary solution. Second row Left: real τ, Second row Right: τ of the POD solution. Third row Left: real mean kinetic energy, Third row Right: POD mean kinetic energy. Fourth row Left: velocity error at each iteration, Fourth row Right: pressure error at each iteration.*

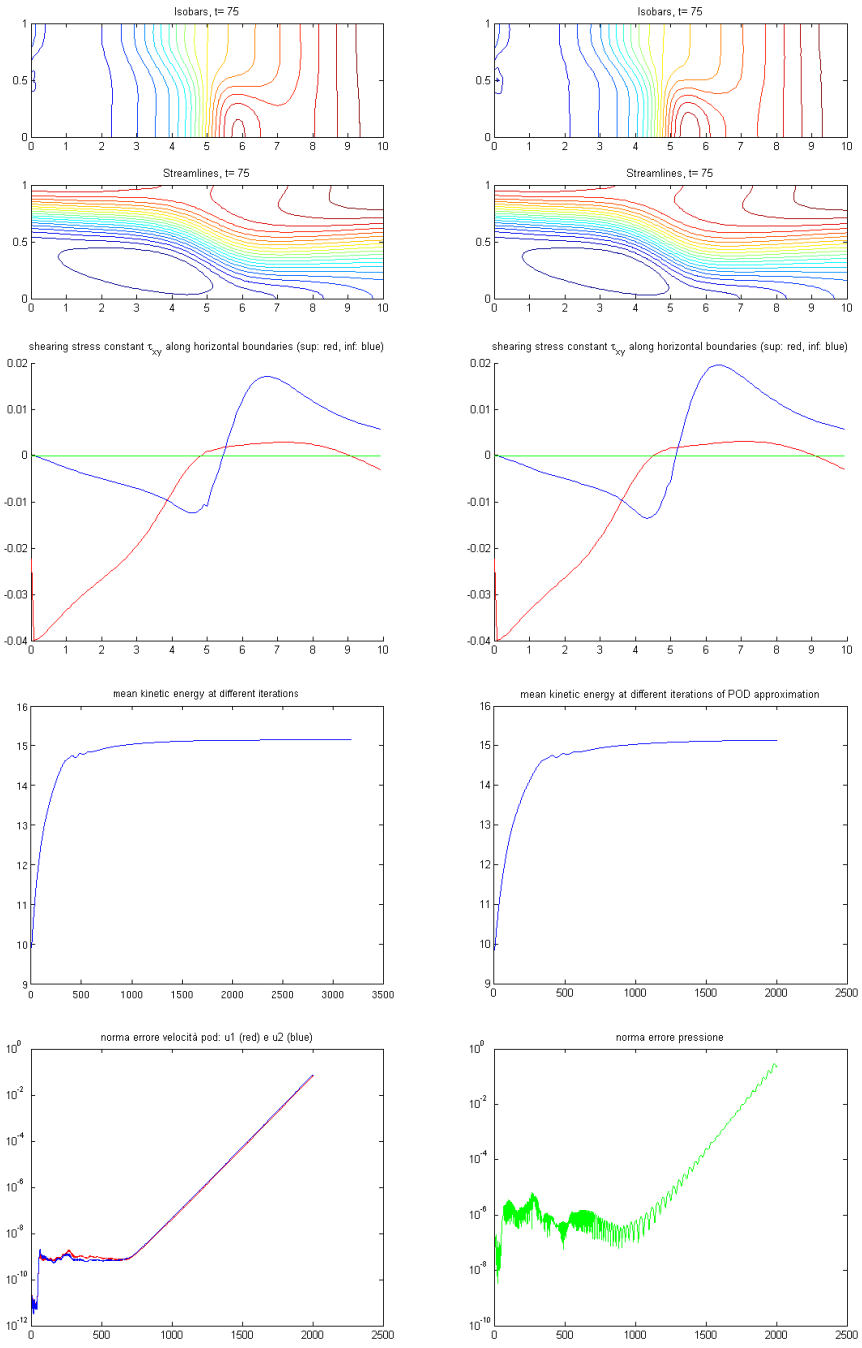

Figure 6.13: *$Re = 400$: $k_1 = k_2 = k_p = 400$. First row Left: "Real streamlines and pressure contour plots of the semi-implicit NS stationary solution". First row Right: streamlines and pressure contour plots of POD NS stationary solution. Second row Left: real τ, Second row Right: τ of the POD solution. Third row Left: real mean kinetic energy, Third row Right: POD mean kinetic energy. Fourth row Left: velocity error at each iteration, Fourth row Right: pressure error at each iteration.*

Observe that, given the POD basis for the velocity $\{\psi_{u_{1i}}, \psi_{u_{2j}}\}$, $i = 1, \ldots, k_1$, $j = 1, \ldots, k_2$ and the POD basis for the pressure $\{\psi_{p_l}\}$, $l = 1, \ldots, k_p$

$$\mathbf{u}_1^{pod} = \sum_{i=1}^{k_1} \mathbf{a}_{u_{1,i}}^{pod} \psi_{u_{1i}}, \qquad \mathbf{u}_2^{pod} = \sum_{i=1}^{k_2} \mathbf{a}_{u_{2,i}}^{pod} \psi_{u_{2i}}$$

$$p^{pod} = \sum_{i=1}^{k_p} a_{p_i}^{pod} \psi_{p_i},$$

the coefficients $\mathbf{a}_{u_{1,i}}^{pod}$,$\mathbf{a}_{u_{2,i}}^{pod}$ and $a_{p_i}^{pod}$ are found numerically, solving a $k_1 + k_2 + k_p$ dimensional ODE (equations 6.29 or 6.30).

Otherwise the exact solution is

$$u_1^e = \sum_{i=1}^{N_u} \mathbf{a}_{u_{1,i}}^e \psi_{u_{1i}}, \qquad u_2^e = \sum_{i=1}^{N_u} \mathbf{a}_{u_{2,i}}^e \psi_{u_{2i}}$$

$$p^e = \sum_{i=1}^{N_p} a_{p_i}^e \psi_{p_i}$$

where $\mathbf{a}_{u_{l,i}}^e := (\mathbf{u}^e, \psi_{u_{li}})$, $l = 1, 2$, and $a_{p_i}^e := (p^e, \psi_{p_i})$. Thus another indicator of POD performance corresponds to compute at each iteration for each mode $\left\| \mathbf{a}_{u_l}^e(1 : k_l) - \mathbf{a}_{u_l}^{pod} \right\|_2$, $l = 1, 2$, and $\left\| \mathbf{a}_p^e(1 : k_p) - \mathbf{a}_p^{pod} \right\|_2$.

As a consequence the quality of the approximation can be tested considering the temporal evolution of the coefficients of POD modes, comparing the exact ones (obtained projecting the matrix of snapshots) and the approximated ones (obtained projected the POD approximation trajectories matrix): cfr. figure 6.14. Observe that when the streamlines are well approximated, the corresponding coefficients of the POD modes are well approximated too. Thus this could be a way to quantify how good the reconstruction of the dynamic is.

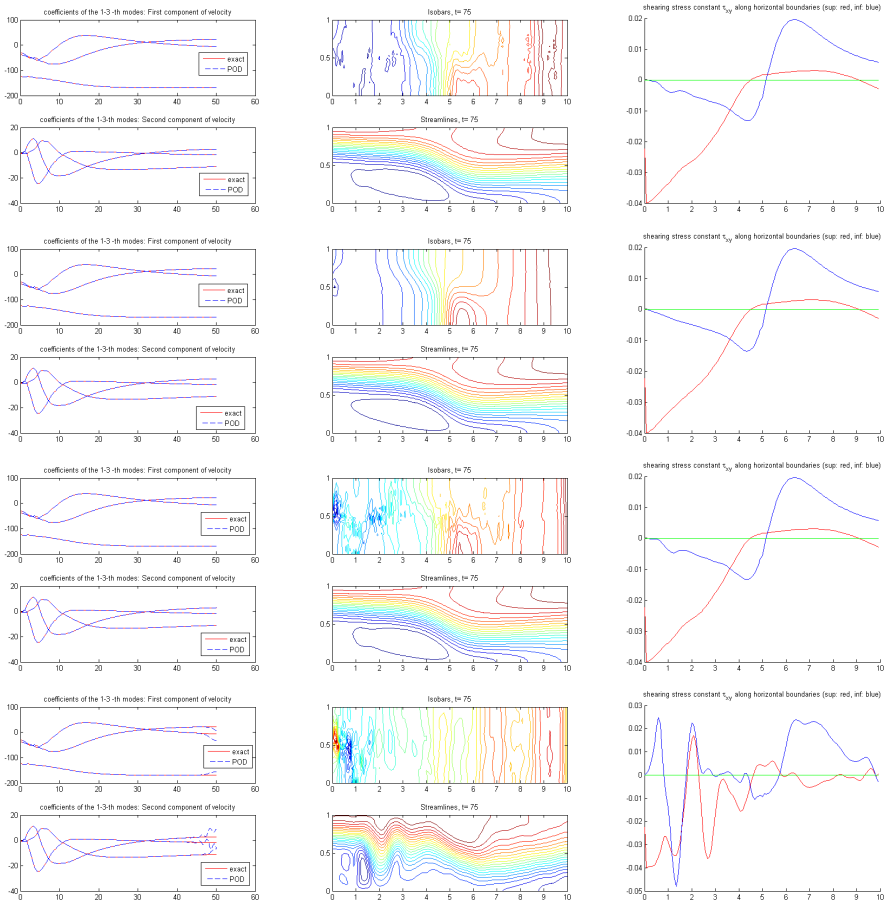

Figure 6.14: *$Re = 400$. First column: Comparison between real temporal evolution of coefficients of the first 3 POD modes (red) and approximated ones (blue) for the velocity field. Second column: Approximated streamlines and isobars. Third column: Approximated τ. First line: $k_1 = k_2 = k_p = 500$. Second line: $k_1 = k_2 = k_p = 400$. Third line: $k_1 = k_2 = k_p = 300$. Fourth line: $k_1 = k_2 = k_p = 250$.*

Observe moreover that the dynamic is lost in the final part of the interval: this is due to the *truncation of smaller modes*, in fact it grows while k_1, k_2 and k_p decrease: the more energetic modes correspond to the first part of the kinetic energy, where there is a higher energy variation. Thus truncating the smaller modes corresponds to neglect the information about the stationary part of the solution. Thus we restrict to the subinterval $[0, 25]$, in which the transitional dynamic occurs.

Observe that the behavior of the coefficients explains why the error is increasing in the last part of the time interval. In fact, the coefficients differ in this subinterval

and in the error the effects of all coefficients of all modes is summed up, and thus it is amplified.

6.6.3.3 Application of POD to the backward facing step problem in the interval $[0, 25]$ of transitional dynamic

For $[0, 25]$ (1000 snapshots), in figures 6.15 and 6.16 it is depicted the reduced dynamic using $k_1 = k_2 = k_p = 200$ and $k_1 = k_2 = k_p = 150$ respectively. For larger truncation thresholds the approximation is better. In figure 6.17 it is shown the analysis of the dynamic of the coefficients of the 3 dominant modes: the approximation is more accurate when the reduced system has higher dimension.

6.6.3.4 Application of POD to the backward facing step problem in the interval $[25, 75]$ of transitional dynamic

For $[25, 75]$ (2000 snapshots), in figures 6.19 and 6.20 it is depicted the reduced dynamic using $k_1 = k_2 = k_p = 50$ and $k_1 = k_2 = k_p = 10$ respectively, while in figure 6.18 it is shown the analysis of the dynamic of the coefficients of the dominant modes. It is evident how in the stationary regime a lower number of modes can describe the dynamic.

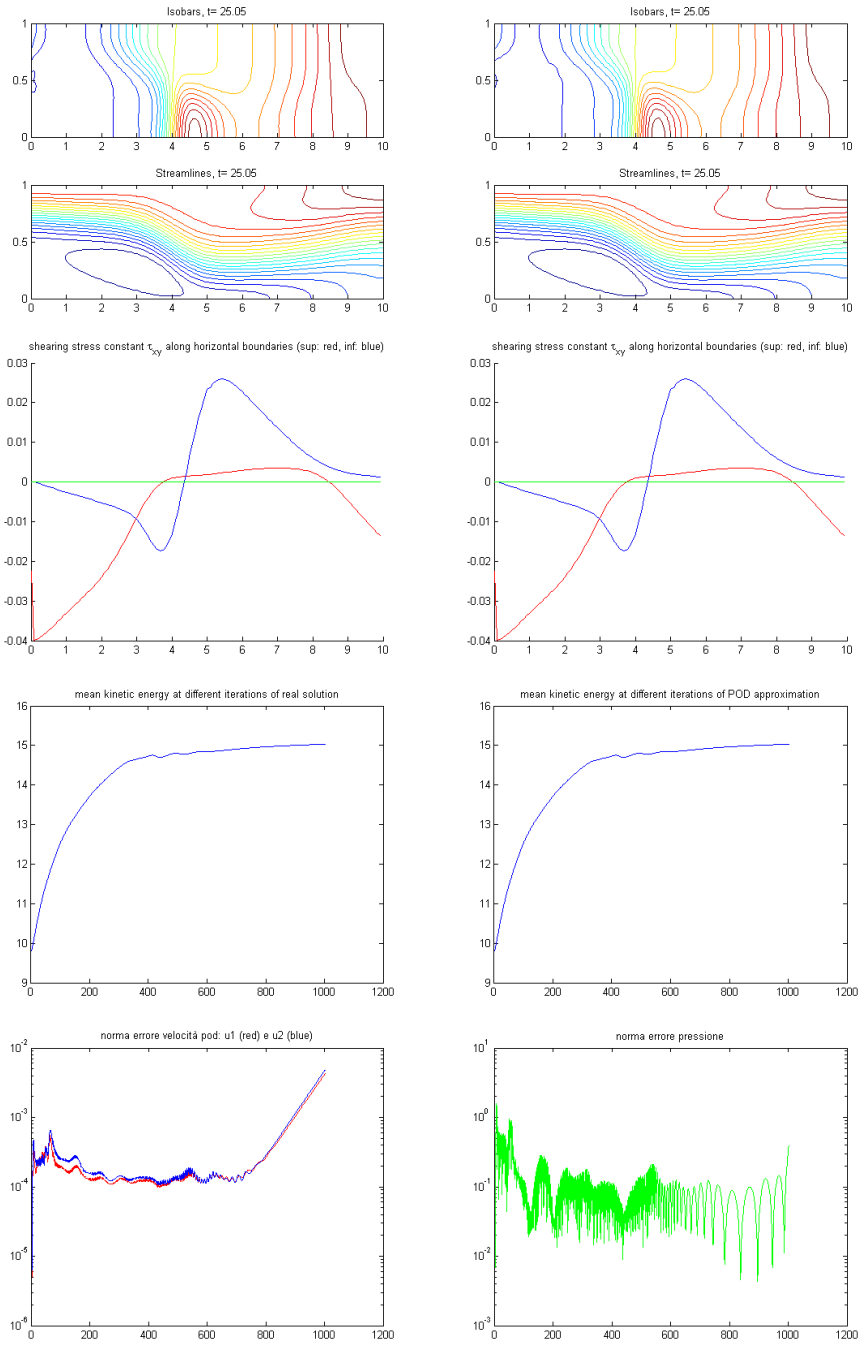

Figure 6.15: *$Re = 400$: $[0, 25]$, $k_1 = k_2 = k_p = 200$. First row Left: "Real streamlines and pressure contour plots of the semi-implicit NS stationary solution". First row Right: streamlines and pressure contour plots of POD NS stationary solution. Second row Left: real τ, Second row Right: τ of the POD solution. Third row Left: real mean kinetic energy, Third row Right: POD mean kinetic energy. Fourth row Left: velocity error at each iteration, Fourth row Right: pressure error at each iteration.*

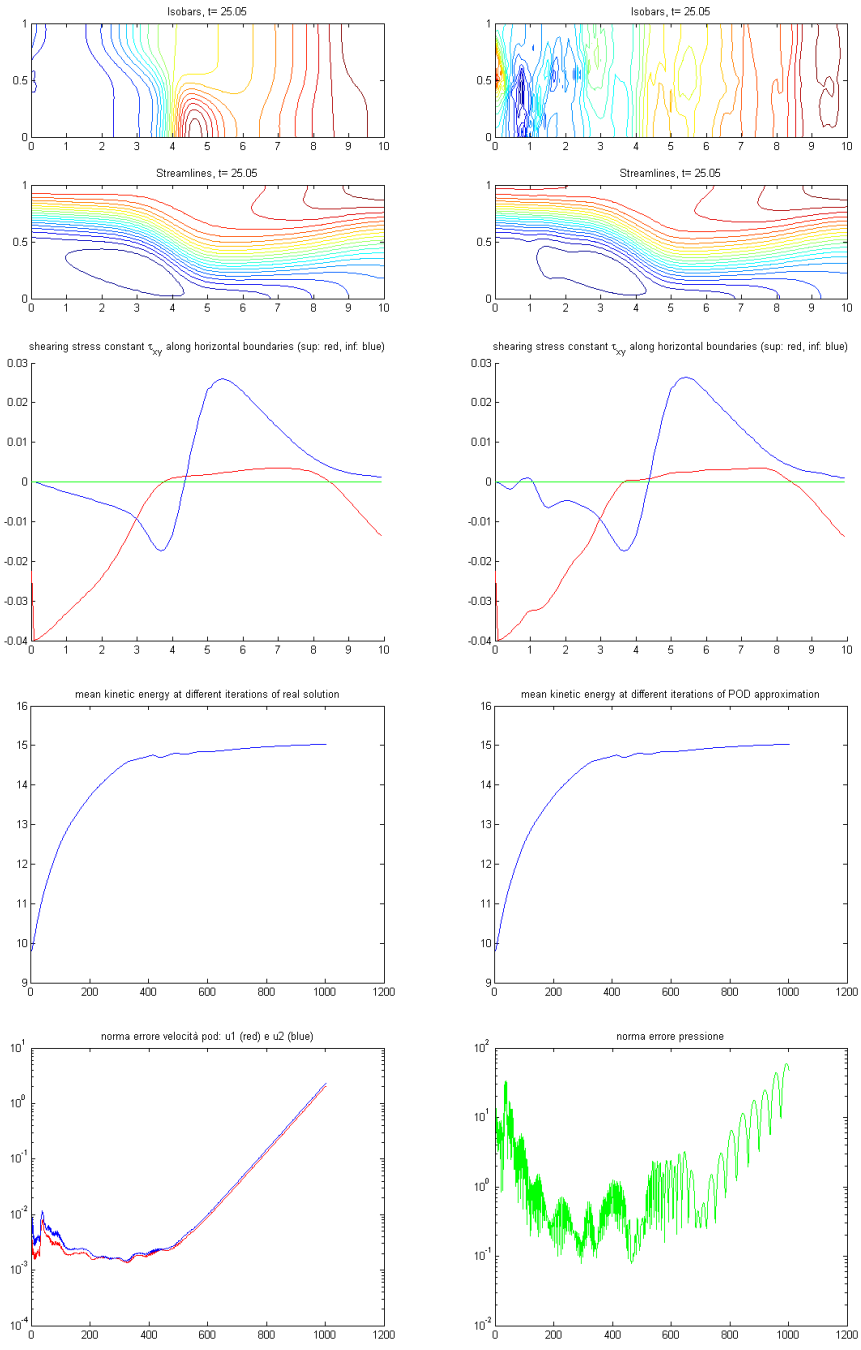

Figure 6.16: *$Re = 400$: $[0, 25]$, $k_1 = k_2 = k_p = 150$. First row Left: "Real streamlines and pressure contour plots of the semi-implicit NS stationary solution". First row Right: streamlines and pressure contour plots of POD NS stationary solution. Second row Left: real τ, Second row Right: τ of the POD solution. Third row Left: real mean kinetic energy, Third row Right: POD mean kinetic energy. Fourth row Left: velocity error at each iteration, Fourth row Right: pressure error at each iteration.*

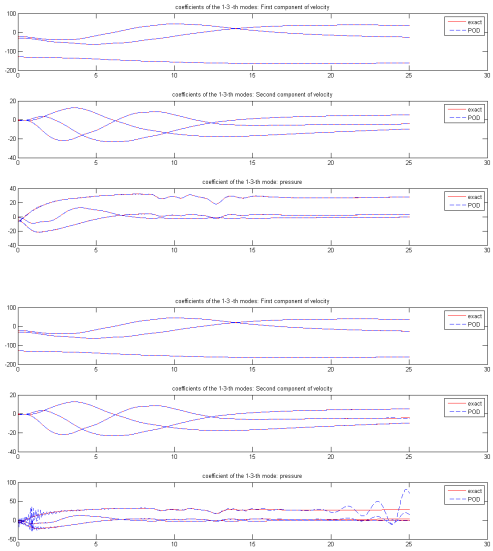

Figure 6.17: *$Re = 400$: $[0, 25]$. Comparison between real temporal evolution of coefficients of the first 3 POD modes (red) and approximated ones (blue). Left: $k_1 = k_2 = k_p = 200$. Right: $k_1 = k_2 = k_p = 150$.*

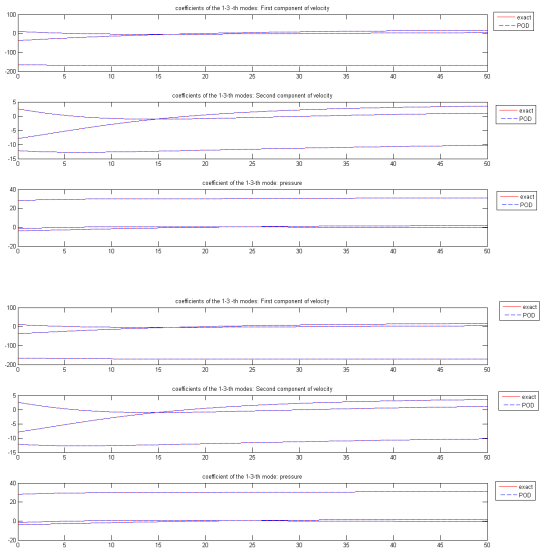

Figure 6.18: *$Re = 400$: $[25, 75]$. Comparison between real temporal evolution of coefficients of the first 3 POD modes (red) and approximated ones (blue). First line: $k_1 = k_2 = k_p = 50$. Second line: $k_1 = k_2 = k_p = 10$. .*

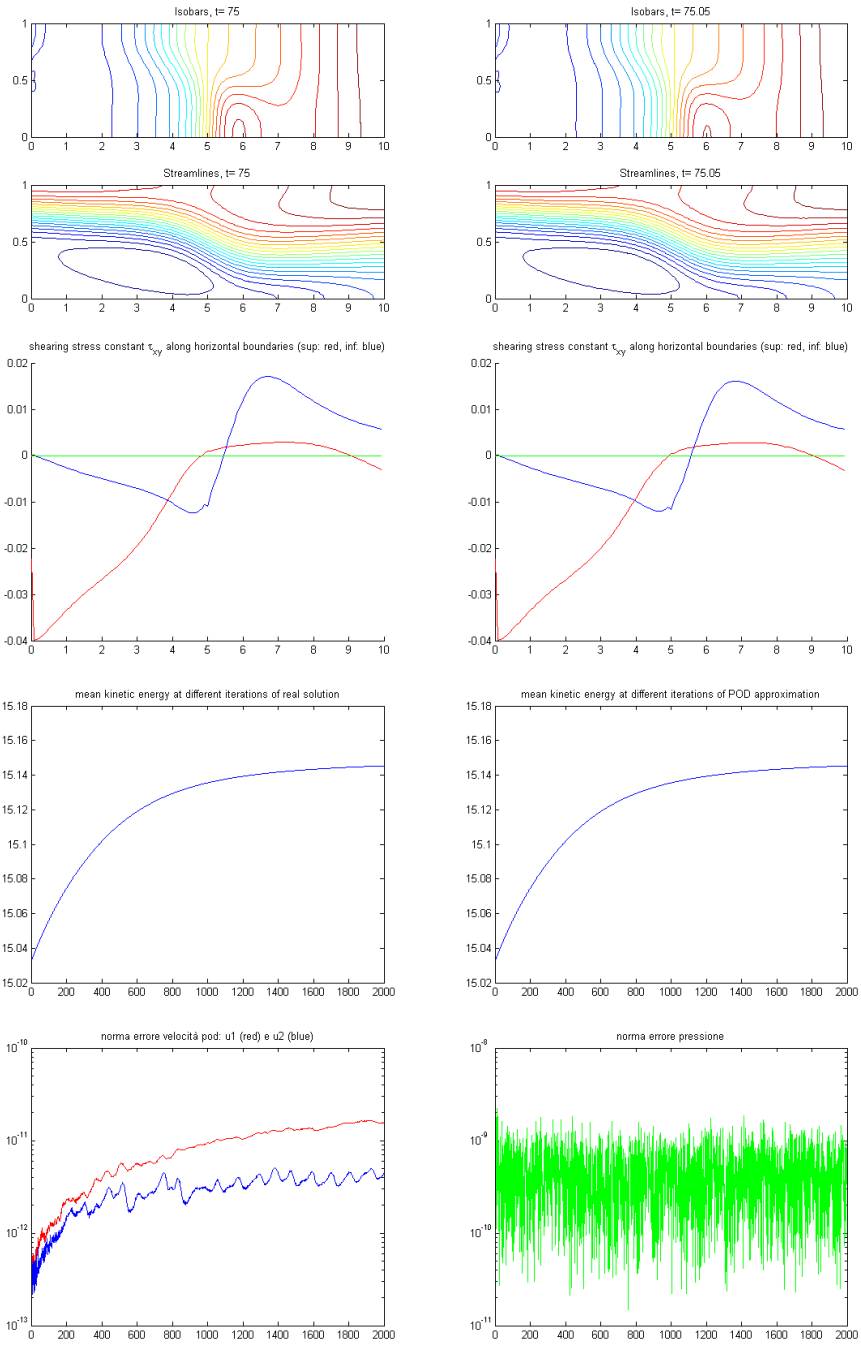

Figure 6.19: *$Re = 400$: $[25, 75]$, $k_1 = k_2 = k_p = 50$. First row Left: "Real streamlines and pressure contour plots of the semi-implicit NS stationary solution". First row Right: streamlines and pressure contour plots of POD NS stationary solution. Second row Left: real τ, Second row Right: τ of the POD solution. Third row Left: real mean kinetic energy, Third row Right: POD mean kinetic energy. Fourth row Left: velocity error at each iteration, Fourth row Right: pressure error at each iteration.*

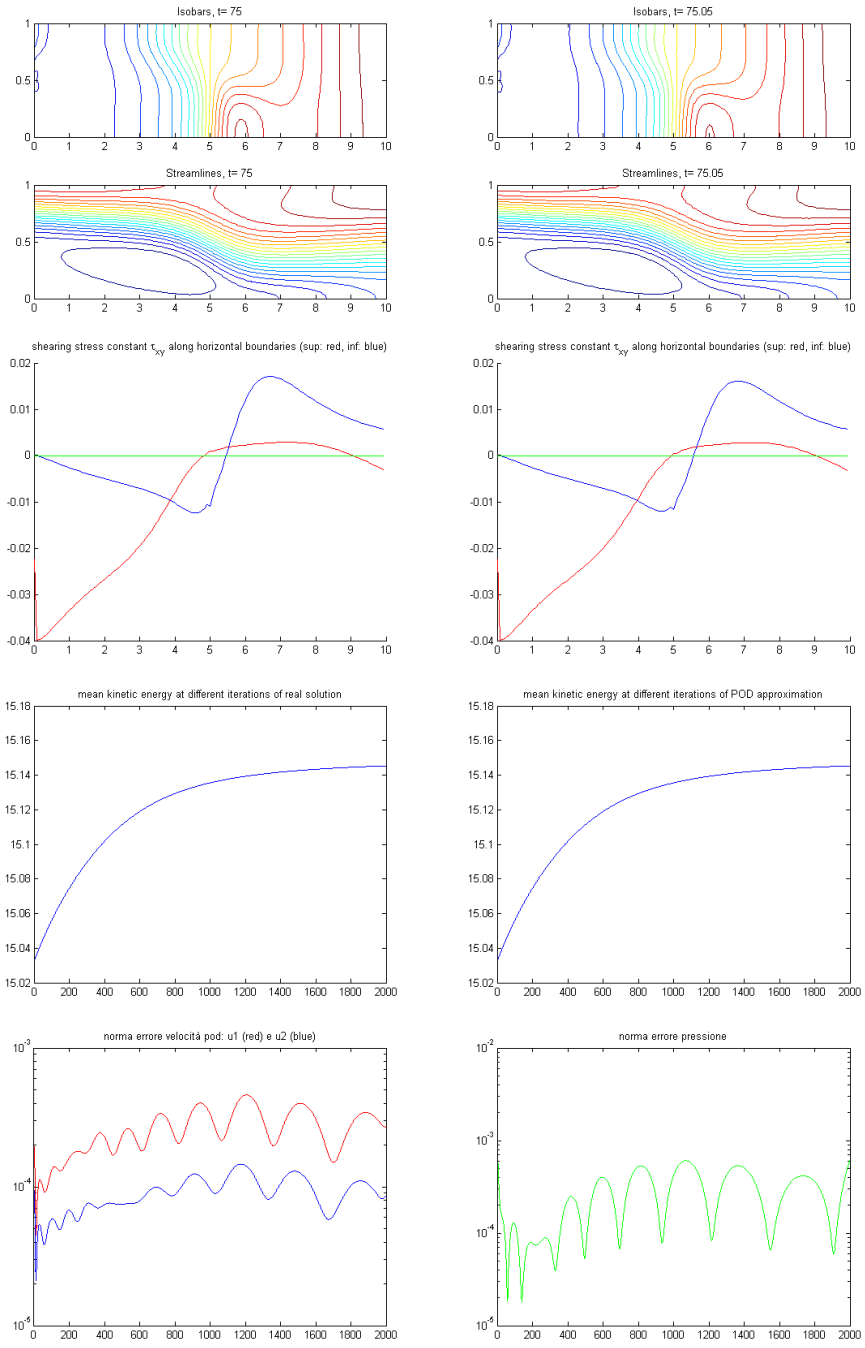

Figure 6.20: *$Re = 400$: $[25, 75]$, $k_1 = k_2 = k_p = 10$. First row Left: "Real streamlines and pressure contour plots of the semi-implicit NS stationary solution". First row Right: streamlines and pressure contour plots of POD NS stationary solution. Second row Left: real τ, Second row Right: τ of the POD solution. Third row Left: real mean kinetic energy, Third row Right: POD mean kinetic energy. Fourth row Left: velocity error at each iteration, Fourth row Right: pressure error at each iteration.*

6.7 Corrected POD

As mentioned in section 6.5, dealing with complicate dynamics, like transitional phenomena, POD performs well if the projection space is large enough, or equivalently, if a sufficiently high number k of POD's modes are taken into account. The larger is k, the bigger is the reduced system. Thus the idea is to keep k as low as possible, trying to model in some way the truncated dynamics.

In this section we present an idea to take into account truncated dynamics. Numerical tests show us that it is not easy to reduce the dimension of the POD reduced system, dealing with complicate dynamics.

As observed in Remark 6.4.2, assuming that $y = \sum_{i=1}^{k} \alpha_i \psi_i$ (POD-Galerkin ansatz), the POD-Galerkin system (6.28) consists in finding $\boldsymbol{\alpha} \in \mathbb{R}^k$ s.t. $\forall \psi_i \in V^k$

$$\frac{d}{dt}\alpha_i + \sum_{j=1}^{k} \alpha_j a(\psi_j, \psi_i) + \sum_{j=1}^{k}\sum_{s=1}^{k} \alpha_j \alpha_s \langle b(\psi_j, \psi_s), \psi_i \rangle + \sum_{j=1}^{k} \alpha_j \langle R\psi_j, \psi_i \rangle = (f(\tau_l), \psi_i)_V. \tag{6.31}$$

This is a system of k ODE's, in k unknowns $\alpha_1, \ldots, \alpha_k$.

Consider now $\bar{k} < k$, thus 6.31 can be written equivalently $\forall i = 1, \ldots, \bar{k}, \ldots, k$ solve

$$\frac{d}{dt}\alpha_i + \sum_{j=1}^{\bar{k}} \alpha_j a(\psi_j, \psi_i) + \sum_{j=1}^{\bar{k}}\sum_{s=1}^{\bar{k}} \alpha_j \alpha_s \langle b(\psi_j, \psi_s), \psi_i \rangle + \sum_{j=1}^{\bar{k}} \alpha_j \langle R\psi_j, \psi_i \rangle = (f(\tau_l), \psi_i)_V - r(\alpha_1, \ldots, \alpha_k), \tag{6.32}$$

where

$$\begin{aligned} r(\alpha_1, \ldots, \alpha_k) &= \sum_{j=\bar{k}+1}^{k} \alpha_j \left(a(\psi_j, \psi_i) + \langle R\psi_j, \psi_i \rangle\right) + \sum_{j=1}^{\bar{k}} \sum_{s=\bar{k}+1}^{k} \alpha_j \alpha_s \langle b(\psi_j, \psi_s), \psi_i \rangle \\ &+ \sum_{j=\bar{k}+1}^{k} \sum_{s=1}^{\bar{k}} \alpha_j \alpha_s \langle b(\psi_j, \psi_s), \psi_i \rangle + \sum_{j=1+\bar{k}}^{k} \sum_{s=\bar{k}+1}^{k} \alpha_j \alpha_s \langle b(\psi_j, \psi_s), \psi_i \rangle . \end{aligned}$$

The idea now is to consider the first $\bar{k}$ equations of (6.32), $\underline{\forall i = 1, \ldots, \bar{k}}$ solve

$$\frac{d}{dt}\alpha_i + \sum_{j=1}^{\bar{k}} \alpha_j a(\psi_j, \psi_i) + \sum_{j=1}^{\bar{k}}\sum_{s=1}^{\bar{k}} \alpha_j \alpha_s \langle b(\psi_j, \psi_s), \psi_i \rangle + \sum_{j=1}^{\bar{k}} \alpha_j \langle R\psi_j, \psi_i \rangle = (f(\tau_l), \psi_i)_V - r(\alpha_1, \ldots, \alpha_k), \tag{6.33}$$

obtaining an underdeterminate system of $\bar{k}$ equations in k unknowns. How to fix the remaining $k - \bar{k}$ unknowns? At some time instances, when a suitable criterium is satisfied, as will be presented in the following, compute $\alpha_{\bar{k}+1}, \ldots, \alpha_k$, and then use these values in (6.32) to *correct* the right-hand side of the ODE's system corresponding to the first $\bar{k}$ equations (using $r(\alpha_1, \ldots, \alpha_k)$).

The idea is sketched in algorithm 1.

Algorithm 1 Corrected POD:

1: Given the matrix of snapshots $\chi \in \mathbb{R}^{n\times N}$, and its left singular vectors $U \in \mathbb{R}^{n\times n}$, compute the thresholds $\bar{k} < k$.
2: Initialize $\alpha_{\bar{k}+1} = \alpha_k = 0$;
3: **for all** time steps **do**
4: solve the system (6.33)
5: **if** A suitable criterium is satisfied **then** { %% update $\alpha_{\bar{k}+1}, \ldots, \alpha_k$}
6: Solve the bigger system (6.31), obtaining a new estimate also for $\alpha_{\bar{k}+1}, \ldots, \alpha_k$
7: **end if**
8: **end for**

Some criteria used to determine when it is necessary to correct will be presented in the following section, where some numerical tests are considered.

6.7.1 Algebraic formulation

To better understand the method, now we present its albebraic formulation. Consider the reduction of Navier Stokes equation presented in section 6.6. Assume that $k_1 = k_2 = k$ and consider the $2k + k_p$ dimensional POD reduced algebraic system, obtained after discretizing in time (6.31) using an explicit or semi-implicit treatment of the nonlinear term,

$$\begin{pmatrix} \mathcal{A}_{k,1} & 0 & -B_{k,1}^T \\ 0 & \mathcal{A}_{k,2} & -B_{k,2}^T \\ -B_{k,1} & -B_{k,2} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_p \end{pmatrix} = \begin{pmatrix} U_{r1}^T \mathbf{F}_1 \\ U_{r2}^T \mathbf{F}_2 \\ U_{rp}^T \mathbf{F}_p \end{pmatrix}, \tag{6.34}$$

where $\alpha(t_i) = (\mathbf{a}_1^{(i)}, \mathbf{a}_2^{(i)}, \mathbf{a}_p^{(i)})^T$, coefficients corresponding to the first and second components of velocity and to the pressure respectively.

Define $U_j^{(c)} := U_j(:, 1 : \bar{k})$ and $U_j^{(f)} := U_j(:, \bar{k} + 1 : k)$. Since $U_{rj} = [U_j^{(c)}, U_j^{(f)}] \supset U_{\bar{k},j} = U_j^{(c)}$, $j = 1, 2$, the computation on the k-th dimensional space could be seen as a correction of the $\bar{k}$-dimensional one. Observe that

$$\begin{aligned} \mathcal{A}_{k,1} &= \begin{pmatrix} (U_1^{(c)})^T \mathcal{A} U_1^{(c)} & (U_1^{(c)})^T \mathcal{A} U_1^{(f)} \\ (U_1^{(f)})^T \mathcal{A} U_1^{(c)} & (U_1^{(f)})^T \mathcal{A} U_1^{(f)} \end{pmatrix}, \\ \mathcal{A}_{k,2} &= \begin{pmatrix} (U_2^{(c)})^T \mathcal{A} U_2^{(c)} & (U_2^{(c)})^T \mathcal{A} U_2^{(f)} \\ (U_2^{(f)})^T \mathcal{A} U_2^{(c)} & (U_2^{(f)})^T \mathcal{A} U_2^{(f)} \end{pmatrix}, \\ \mathcal{B}_{k,1} &= \begin{pmatrix} U_p^T B_1 U_1^{(c)} & U_p^T B_1 U_1^{(f)} \end{pmatrix}, \\ \mathcal{B}_{k,2} &= \begin{pmatrix} U_p^T B_2 U_2^{(c)} & U_p^T B_2 U_2^{(f)} \end{pmatrix}. \end{aligned}$$

In particular given the solution of the system of order $2\bar{k}+k_p$, this could be corrected to obtain an approximation of the solution of the $2k + k_p$-th dimensional system. More

precisely, at iteration n, given $(\mathbf{a}_{1,old}, \mathbf{a}_{2,old}, \mathbf{a}_{p,old})^T \in \mathbb{R}^{2\bar{k}+k_p}$, instead of solving the POD $2\bar{k}+k_p$-dimensional system

$$\begin{pmatrix} \mathcal{A}_{\bar{k},1} & 0 & -B_{\bar{k},1}^T \\ 0 & \mathcal{A}_{\bar{k},2} & -B_{\bar{k},2}^T \\ -B_{\bar{k},1} & -B_{\bar{k},2} & 0 \end{pmatrix} \begin{pmatrix} \bar{\boldsymbol{a}}_1 \\ \bar{\boldsymbol{a}}_2 \\ \bar{\boldsymbol{a}}_p \end{pmatrix} = \begin{pmatrix} (U_1^{(c)})^T \mathbf{F}_1 \\ (U_2^{(c)})^T \mathbf{F}_2 \\ U_{rp}^T \mathbf{F}_p \end{pmatrix}, \tag{6.35}$$

solve

$$\begin{pmatrix} \mathcal{A}_{\bar{k},1} & 0 & -B_{\bar{k},1}^T \\ 0 & \mathcal{A}_{\bar{k},2} & -B_{\bar{k},2}^T \\ -B_{\bar{k},1} & -B_{\bar{k},2} & 0 \end{pmatrix} \begin{pmatrix} \bar{\boldsymbol{a}}_1 \\ \bar{\boldsymbol{a}}_2 \\ \bar{\boldsymbol{a}}_p \end{pmatrix} = \begin{pmatrix} (U_1^{(c)})^T \mathbf{F}_1 - (U_1^{(c)})^T \mathcal{A} U_1^{(f)} \mathbf{a}_{1,old}(:, \bar{k}+1:k) \\ (U_2^{(c)})^T \mathbf{F}_2 - (U_2^{(c)})^T \mathcal{A} U_2^{(f)} \mathbf{a}_{2,old}(:, \bar{k}+1:k) \\ U_{rp}^T \mathbf{F}_p \end{pmatrix}, \tag{6.36}$$

To update the $2k+k_p$ dimensional vector of coefficients, the original $2k+k_p$-dimensional POD system is solved only at properly chosen instances (*correction*). In the following section some ideas to correct the approximation will be introduced.

Remark 6.7.1 *Previous ideas can be extended considering also a correction of pressure ($\bar{k}_p$), and distinguishing the two components of velocities (i.e. using two different thresholds k_1 and k_2).*

6.7.2 Numerical simulations

Consider two different thresholds k_1 and k_2 for $\mathbf{u}_1$ and $\mathbf{u}_2$ respectively.

As will be shown in this section, although this strategy allow one to deal with a smaller system, the quality of the approximation is good only if the correction is done a high number of times.

This can be seen for example in figure 6.21, where corrected POD is applied, correcting (i.e. solving the higher $k_1+k_2+k_p$ dimensional system) when the errors of the POD modes $\left\|\mathbf{a}_{u_1}^e(1:k_1) - \mathbf{a}_{u_1}^{pod}\right\|_2$, $\left\|\mathbf{a}_{u_2}^e(1:k_2) - \mathbf{a}_{u_2}^{pod}\right\|_2$ and $\left\|\mathbf{a}_p^e - \mathbf{a}_p^{pod}\right\|_2$ are greater than tolerance $Tol = 0.001$. We consider what happens using different dimensions of the reduced systems.

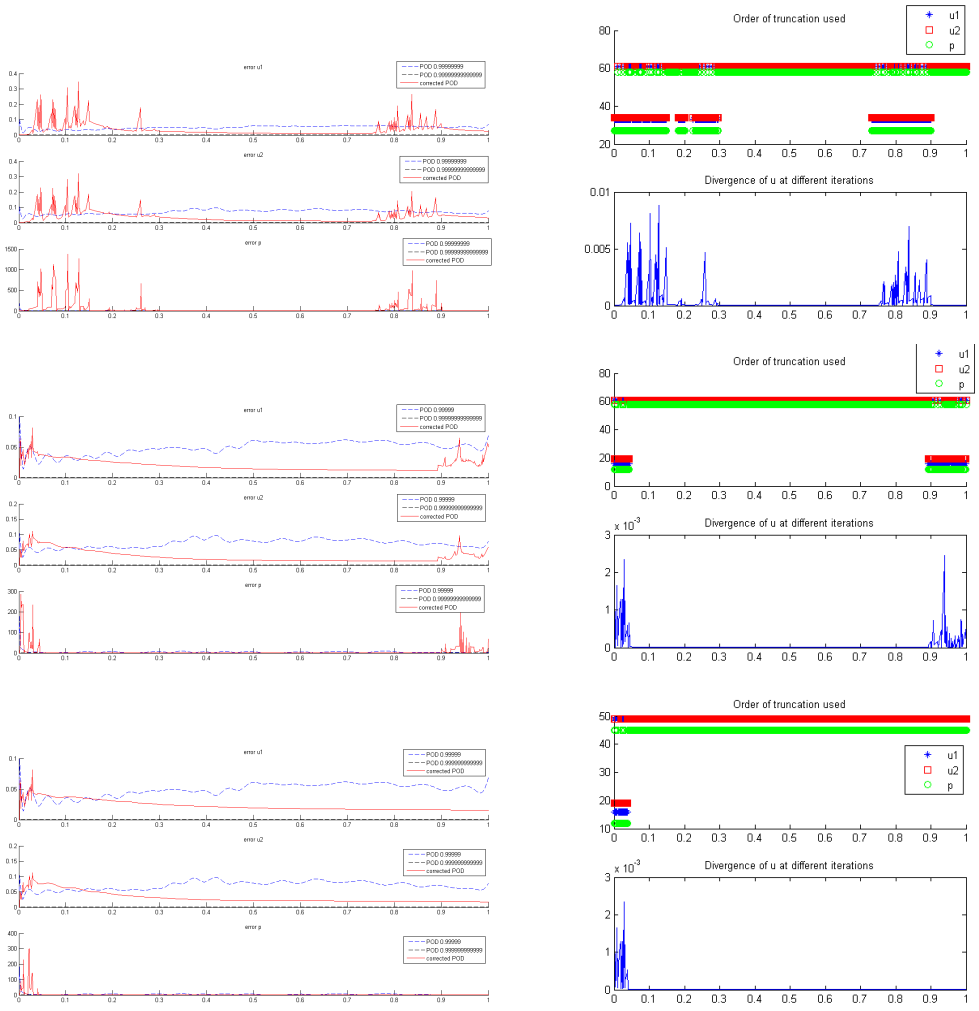

Figure 6.21: *Tol* $= 0.001$. *Using an explicit method in* $[0,1]$. *First row:* $\bar{k}_1 = 33$, $\bar{k}_2 = 34$, $\bar{m} = 27$ *and* $k_1 = k_2 = 61$, $k_p = 58$. *Second row: Levels* $\bar{k}_1 = 16$, $\bar{k}_2 = 19$, $\bar{m} = 12$ *and* $k_1 = k_2 = 61$, $k_p = 58$. *Third row:* $\bar{k}_1 = 16$, $\bar{k}_2 = 19$, $\bar{m} = 12$ *and* $k_1 = k_2 = 49$, $k_p = 45$. *Left: error, Right: truncation levels and divergence.*

Since we observe that there are peaks both in the velocity and in the divergence curves, instead of considering the coefficients error, we impose a constraint on the divergence: the method solves the higher dimensional POD system when the divergence of the solution becomes greater than a fixed threshold (cfr. figure 6.22). As before, the correction is done a moderate number of times and POD approximation on the smallest space is not improved substantialy.

Consider now a periodic regime, using as a test case the obstacle problem, presented

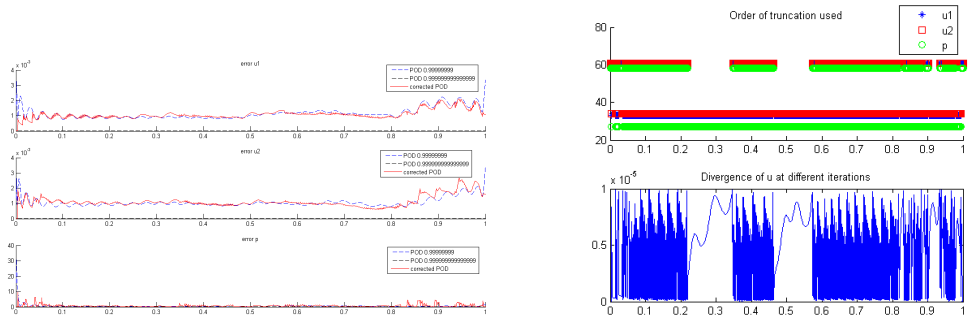

Figure 6.22: $\bar{k}_1 = 33$, $\bar{k}_2 = 34$, $\bar{m} = 27$ *and* $k_1 = k_2 = 61$, $m = 58$. *Criterium on the divergence: threshold* $1e-5$. *Using an explicit method in* $[0,1]$.

in section 4.4.2, restricted to the interval $[0, 1.4]$ (700 time steps). Impose to correct using a theshold on the divergence and also every fixed number of iterations (e.g. 10 iterations): cfr. figure 6.23.

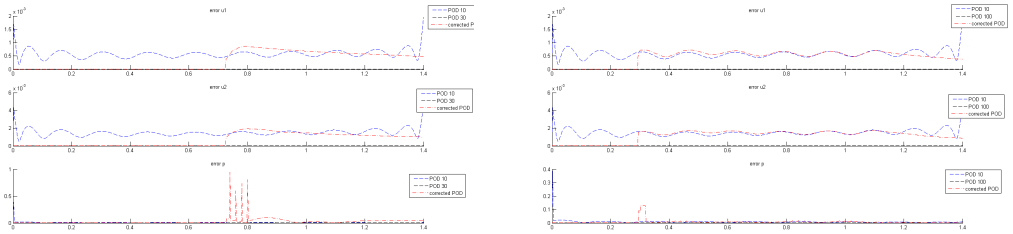

Figure 6.23: *Obstacle. Correcting also every* 10 *iterations. Left:* $\bar{k} = 10$ *and* $k = 30$.*Right:* $\bar{k} = 10$ *and* $k = 100$. $tol_{div} = 5e-5$.

The approximation is good in the first part of the interval, where correction is done often, but gets worse in the second one, where it corrects only every 10 iterations.

All these tests show that dealing with complicated dynamics means a high variability in POD coefficients, which could not be considered constant for a too high number of iterations. Thus standard POD performance on the smaller space is only slightly improved by this method.

Part III

Parabolic inverse problems

A lack of information cannot be remedied by any mathematical trickery!

(Lanczos)

In this last part two *Inverse Problems* are solved, starting from numerically simulated experimental data and some *a priori* knowledge.

In particular first of all we will consider a geometric conduction inverse problem of corrosion estimation, based upon the heat equation: it is solved with a novel strategy, the so called *Predictor-Corrector* method, originally presented in (155) and here substantially improved (138). This strategy deals with an adaptive parametrization.

In the last chapter concepts introduced previously will be summarized to solve a boundary convection inverse problem of pollution rate estimation. The motion of the pollutant is described by the convection-diffusion-reaction equation and can be generalized considering also Navier-Stokes equation. The inverse problem is solved with a novel strategy considering both an adaptive parametrization and time localization (139); moreover to reduce its computational cost, POD reduction is studied.

7

Inverse problems

7.1 Introduction

As presented in (150), inverse problems are largely used in applications, for solving, among the others, medical (e.g. in tomographical methods), industrial (e.g. monitoring oil pipelines), image analysis and mine detection problems (e.g. ground penetration radar and electromagnetic induction).

In this preliminary chapter the general problem will be presented, describing briefly classical solution strategies and focusing at the end on *least-squares problems*, a class of models that will be treated deeply in the following chapters.

The first point is to understand what *inverse problem* means. As presented in (149), an exhaustive description can be found in the following quotation, taken by *A Study in Scarlet*, of Arthur Conan Doyle:

> "Most people if you describe a train of events to them, will tell you what the result would be. They can put those events together in their minds, and argue from them that something will come to pass. There are few people, however, who, if you tell them a result, would be able to evolve from their own inner consciousness what the steps were which led up that result. This power is what I mean when I talk of *reasoning backward*."

Thus *inverse problems* could be described as problems where the answer is known, but not the question, or where the results, or consequences are known, but not the cause.

Following (151) to characterize them mathematically, we present some examples.

Example 7.1.1 *Find a polynomial p of degree n with given zeros $x_1, \ldots, x_n$. Inverse problem solution is simply $p(x) = c(x - x_1) \ldots (x - x_n)$, $c \in \mathbb{R}$. The corresponding direct problem reads: find the zeros $x_1, \ldots, x_n$ of a given polynomial p.*

Example 7.1.2 (Inverse scattering problem) *Find the shape of a scattering object, given the intensity (and phase) of sound or electromagnetic waves scattered by this object. The corresponding direct problem is that of calculating the scattered wave for a given object. The rigorous mathematical description could be found in (151).*

Example 7.1.3 (Computer tomography) *Consider a fixed plane through a human body: let $\rho(x, y)$ denote the change of density at the point (x, y). Suppose that we direct a thin beam of X-rays into the body along a line L of the plane, and measure how much the intensity is attenuated by going through the body. The inverse problem consists in determining the density ρ given the attenuation of the intensity along all line integrals (Radon transform of ρ). The rigorous mathematical description could be found in (151).*

Example 7.1.4 (Sturm-Liouville eigenvalue problem) *Let a string of length L and mass density $\rho = \rho(x) > 0$, $0 \leq x \leq L$, be fixed at the endpoints $x = 0$ and $x = L$. Plucking the string produces tones due to vibrations. Let $v(x, t)$ $t > 0$ be the displacement at x and time t. Consider a pure tone, i.e. a displacement of the form $v(x, t) = w(x)(a \cos \omega t + b \sin \omega t)$. In the inverse problem one tries to determine the mass density ρ from a number of measured frequencies ω.*

Example 7.1.5 (Backward heat equation) *Consider the one-dimensional heat equation*

$$\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2}$$

with boundary conditions

$$u(0, t) = u(\pi, t) = 0, \ t \geq 0$$

and initial condition

$$u(x, 0) = u_0(x), \ 0 \leq x \leq \pi.$$

In the inverse problem one measures the final temperature distribution $u(\cdot, T)$ and tries to determine the initial temperature $u(\cdot, 0)$.

Example 7.1.6 (Diffusion in inhomogeneous medium) *The equation of diffusion in an inhomogeneous medium is*

$$\frac{\partial u(x,t)}{\partial t} = \frac{1}{c} div(k \nabla u(x,t)),\ x \in D,\ t > 0,$$

where c is a constant and $k = k(x)$ is a parameter describing the medium. In the inverse problem one measures u and the flux $\frac{\partial u}{\partial n}$ on the boundary ∂D and tries to determine the unknown function k in D.

This is an example of parameter identification for a partial differential equation.

Further examples could be found e.g. in (149).

As noted in (151), given two normed space X and Y, an operator $\mathcal{K} :\ X \to Y$ and a measurement y, in all of these examples we can formulate the inverse problem as the solution of the equation

$$\mathcal{K}(x) = y. \tag{7.1}$$

In order to formulate an inverse problem, the definition of the operator $\mathcal{K}$, including its domain and range, has to be given. In general the evaluation of $\mathcal{K}(x)$ means *solving a boundary value problem for a differential equation* or *evaluating an integral*.

Usually inverse problems are *ill-posed* or *improperly posed* in the sense of Hadamard (145), while the corresponding direct problem is *well-posed*.

Definition 7.1.1 *Let X and Y be normed spaces, $\mathcal{K} :\ X \to Y$ a (linear or nonlinear) mapping. The equation $\mathcal{K}(x) = y$ is called* ***properly posed*** *or* ***well-posed*** *if the following holds:*

1. *Existence: for every $y \in Y$ there is at least one $x \in X$ s.t. $\mathcal{K}(x) = y$.*

2. *Uniqueness: for every $y \in Y$ there is at most one $x \in X$ with $\mathcal{K}(x) = y$.*

3. *Stability: the solution x depends continuously on y, i.e. for every sequence (x_n) with $\mathcal{K}(x_n) \to \mathcal{K}(x)$ as $n \to \infty$, it follows that $x_n \to x$.*

Equations for which at least one of these properties does not hold are called ***improperly posed*** *or* ***ill-posed***.

As mentioned in (151), observe that mathematically the existence of a solution can be enforced by enlarging the solution space. Moreover if a problem has more

than one solution, then information about the model is missing. The requirement of stability is the most important one: if a problem lacks stability, then its solution is difficult to compute because any measurement or numerical computation is polluted by unavoidable errors.

Usually inverse problems are characterized by an intrinsic *loss of information* (131), due to the smoothing properties of the operator $\mathcal{K}$: also from a discrete point of view, when we discretize the infinite dimensional operator $\mathcal{K}$, we deal with a sequence of linear algebraic system of the form $A_k x = y$, which inherits the intrinsic ill-posedness in the form of *ill-conditioning* of the system matrix A_k, for every iteration step k. Thus the direct inversion is not a good strategy to reconstruct x, due to the amplification of data's noise: we must search an approximate solution, satisfying *additional constraints* coming from the physics of the problem (131). To compensate the loss of information we use some additional *a priori knowledge* about the problem, i.e. we adopt a *regularization method*.

Remark 7.1.1 *In this thesis we are particularly interested in differential models. In a direct well-posed problem, it is required to find a solution that satisfies a given partial differential equation and some initial and boundary conditions. In inverse problems, the PDE and/or initial conditions and/or boundary conditions are not fully specified but, instead, some additional information is available. So separating out inverse mathematical physics problems (166), we can speak of coefficient inverse problems (in which the equation is not specified completely as some equation coefficients and/or right-hand side are unknown), boundary inverse problems (in which boundary conditions are unknown), geometric inverse problems (in which the domain is unknown) and evolutionary inverse problems (in which initial conditions are unknown).*

In the following chapters a geometric conduction inverse problem of corrosion estimation and a boundary convection inverse problem of pollution rate estimation will be presented.

7.2 Solution strategies

As explained above, since noise in measurement data may lead to significant misinterpretations of the solution, the ill-posedness must be handled either by incorporating a priori information via the use of transformations, which stabilizes the problem, or by using appropriate numerical methods, called *regularization techniques* (150).

More precisely to solve an inverse problem two different approaches can be adopted:

1. *first optimize than discretize* strategy: given (7.1), first an optimization problem is defined, and then it is discretized;

2. *first discretize than optimize* strategy: first (7.1) is discretized and then a discrete optimization problem is solved.

7.2.1 First optimize than discretize strategy

Following (150), we denote the measured perturbed data by $y^{(\delta)}$ and assume that these noisy data satisfy

$$\left\| y^{(\delta)} - y \right\| \leq \delta,$$

$\delta > 0$.

7.2.1.1 Tikhonov regularization

The most well-known method for solving ill-posed problems is *Tikhonov regularization*: it consists in approximating a solution of (7.1) by a minimizer $x_\alpha^{(\delta)}$ of

$$J(x) := \|\mathcal{K}(x) - y\|^2 + \alpha \|x - x_0\|^2, \tag{7.2}$$

where $x_0 \in X$ typically unifies all available a priori information on the solution and $\alpha > 0$ is the *regularization parameter*.

As summarized in (150), under mild assumptions on the operator K it can be shown that, for $\alpha > 0$ fixed, the minimizers $x_\alpha^{(\delta)}$ of (7.2) are stable with respect to perturbations of the data y. Moreover, if (7.1) is solvable and if the regularization parameter $\alpha = \alpha(\delta)$ satisfies that $\alpha \to 0$ and $\frac{\delta^2}{\alpha} \to 0$ as $\delta \to 0$, then $x_\alpha^{(\delta)}$ converges to a solution of (7.1). In general, this convergence can be arbitrarily slow (150).

7.2.1.2 Iterative regularization methods

A detailed treatment of regularizing techniques for linear problems could be found in (151), here we follow (150), describing the more general nonlinear case.

While for linear ill-posed problems iterative regularization methods are an alternative to Tikhonov regularization, also the minimization of (7.1) for nonlinear ill-posed problems is usually realized via iterative methods, i.e. methods s.t.

$$x_{k+1}^{(\delta)} = x_k^{(\delta)} + G_k(x_k^{(\delta)}, y^{(\delta)}),\ k \in \mathbb{N}$$

for various choices of G_k. In general for iterative methods the regularization parameter is the number of iterations itself.

Here we briefly present classical methods, referring to (150) for more details.

Assuming that $\mathcal{K}$ has a continuous Fréchet derivative $\mathcal{K}^{'}$, the *nonlinear Landweber iteration* is defined via

$$x_{k+1}^{(\delta)} = x_k^{(\delta)} + \mathcal{K}^{'}(x_k^{(\delta)})^*(y^{(\delta)} - \mathcal{K}(x_k^{(\delta)})),\ k \in \mathbb{N}, \tag{7.3}$$

starting from an initial guess $x_0^{(\delta)} = x_0$.

In case of noisy data, the iteration procedure has to be combined with a stopping rule in order to act as a regularization method. The *discrepancy principle* consists in stopping the iteration after $k_* = k_*(\delta, y^{(\delta)})$ steps s.t.

$$\left\|y^{(\delta)} - \mathcal{K}(x_{k_*}^{(\delta)})\right\| \leq \tau\delta \leq \left\|y^{(\delta)} - \mathcal{K}(x_k^{(\delta)})\right\|,\ 0 \leq k < k_*, \tag{7.4}$$

where τ is an appropriately chosen positive number.

Usually this method converges slowly to the real solution: details could be found in (150). Better rates may be obtained either for solutions that satisfy stronger smoothness conditions if the iteration is performed in a subspace of X with a stronger norm or by adding an additional penalty term to the iteration scheme. An example of the last method is the *iteratively regularized Landweber* iteration (150)

$$x_{k+1}^{(\delta)} = x_k^{(\delta)} + \mathcal{K}^{'}(x_k^{(\delta)})^*(y^{(\delta)} - \mathcal{K}(x_k^{(\delta)})) + \beta_k(x_0 - x_k^{(\delta)}),\ 0 < \beta_k < \frac{1}{2}. \tag{7.5}$$

Another largely used method is the *steepest descent*

$$x_{k+1}^{(\delta)} = x_k^{(\delta)} + w_k^{(\delta)}\mathcal{K}^{'}(x_k^{(\delta)})^*(y^{(\delta)} - \mathcal{K}(x_k^{(\delta)})),\ k \in \mathbb{N}, \tag{7.6}$$

where $w_k^{(\delta)} := \dfrac{\left\|K^{'}(x_k^{(\delta)})^*(y^{(\delta)} - K(x_k^{(\delta)}))\right\|^2}{\left\|K^{'}(x_k^{(\delta)})K^{'}(x_k^{(\delta)})^*(y^{(\delta)} - K(x_k^{(\delta)}))\right\|^2}$.

Faster methods are *Newton type* algorithms: the key idea consists in repeatedly linearize the operator equation (7.1) around an approximate solution $x_k^{(\delta)}$. However, usually these linearized problems are also ill-posed if the nonlinear problem is ill-posed and, therefore, *they have to be regularized*. If we apply Tikhonov regularization to the linearized problem, we end up with the *Levenberg-Marquardt method*:

$$x_{k+1}^{(\delta)} = x_k^{(\delta)} + (\mathcal{K}^{'}(x_k^{(\delta)})^*\mathcal{K}^{'}(x_k^{(\delta)}) + \alpha_k\mathbb{I})^{-1}(\mathcal{K}^{'}(x_k^{(\delta)})^*(y^{(\delta)} - \mathcal{K}(x_k^{(\delta)}))),\ k \in \mathbb{N}, \tag{7.7}$$

where α_k is such that

$$\left\|y^{(\delta)} - \mathcal{K}(x_k^{(\delta)}) - \mathcal{K}^{'}(x_k^{(\delta)})(x_{k+1}^{(\delta)}(\alpha_k) - x_k^{(\delta)})\right\| = q\left\|y^{(\delta)} - \mathcal{K}(x_k^{(\delta)})\right\|,$$

for some fixed $q \in (0,1)$ (*discrepancy principle*).

Adding a penalty term to the linearized problem yields the regularizing algorithm

$$x_{k+1}^{(\delta)} = x_k^{(\delta)} + (\mathcal{K}'(x_k^{(\delta)})^* \mathcal{K}'(x_k^{(\delta)}) + \alpha_k \mathbb{I})^{-1} (\mathcal{K}'(x_k^{(\delta)})^* (y^{(\delta)} - \mathcal{K}(x_k^{(\delta)}))) + \alpha_k (x_0 - x_k^{(\delta)}), \quad (7.8)$$

From a more general point of view, given $\alpha > 0$, define the *regularizing operator* $R_\alpha(\mathcal{K}'(x)) \approx \mathcal{K}'(x)^*$, i.e. an operator s.t.

$$\begin{gathered} R_\alpha(\mathcal{K}) y \to \mathcal{K}^* y \quad as\ \alpha \to 0,\ \forall\ y \in \mathcal{K}(X), \\ \|R_\alpha(\mathcal{K})\| \leq \Phi(\alpha), \quad \|R_\alpha(\mathcal{K})\mathcal{K}\| \leq c_K, \quad \forall\ \mathcal{K} \in \mathcal{L}(X,Y),\ with\ \|\mathcal{K}\| \leq c_s \end{gathered} \quad (7.9)$$

for some positive function $\Phi(\alpha)$ and some positive constants c_K and c_s.

Within this class many well-known regularization methods can be found, such as Tikhonov regularization and Landweber iteration. For example the special choice

$$R_{\alpha_k}(F'(x)) = (\mathcal{K}'(x_k^{(\delta)})^* \mathcal{K}'(x_k^{(\delta)}) + \alpha_k \mathbb{I})^{-1} (\mathcal{K}'(x_k^{(\delta)})^*$$

corresponds to the Levenberg-Marquardt method. However, this slightly more general concept additionally includes regularization by discretization (150). This last approach is motivated by the fact that for the numerical treatment of such equations one has to discretize the continuous problem and reduce it to a finite system of linear or nonlinear equations. As explained in (151), discretization schemes themselves are regularization strategies, e.g. Galerkin methods: in this case the regularization parameter coincides e.g. with the mesh size.

7.2.2 First discretize than optimize strategy

This methodology will be adopted in the following chapters of this thesis.

First of all, we assume that the unknown $x \in X$, solution of (7.1), can be described by a vector $\boldsymbol{\vartheta} \in \mathbb{R}^{n_\theta}$ of non negative parameters. The way this assumption is imposed is problem dependent: we are going to see how this can be done in the following chapters.

Thus we can restate the continuous problem (7.1) as a discrete one: given the operator $\tilde{\mathcal{K}}:\ \mathbb{R}^{n_\theta} \to \mathbb{R}^{N n_y}$ and a measurement $\mathbf{y} \in \mathbb{R}^{N n_y}$, find $\boldsymbol{\vartheta} \in \mathbb{R}^{n_\theta}$ such that

$$\tilde{\mathcal{K}}(\boldsymbol{\vartheta}) = \mathbf{y}. \quad (7.10)$$

7.2.2.1 Least-squares approach

To solve (7.10), we minimize the following functional

$$\tilde{J}(\boldsymbol{\vartheta}) := \frac{1}{N} \|\mathbf{e}_\theta\|_2^2, \quad (7.11)$$

where the *residual* or *prediction error* $\mathbf{e}_\theta : \mathbb{R}^{n_\theta} \to \mathbb{R}^{Nn_y}$, $\mathbf{e}_\theta := \tilde{\mathcal{K}}(\boldsymbol{\vartheta}) - \mathbf{y}$. This methodology is called *least squares*.

Observe that in this context *regularization* is done using an *adaptive parametrization*, as will be explained in the following chapters.

As summarized in (162), least-squares problems have been a fruitful area of study for over 30 years, mainly because of their applicability to many practical problems (chemical, physical, financial, or economic) to measure the discrepancy between the model and the output of the system at various observation points. By minimizing this function, they select values for the parameters that best match the model to the data.

The computation of $\tilde{J}'$ can be done in different ways. A first strategy consists in adopting the adjoint model (170): this choice is preferred especially when the quantity to be estimated corresponds to a large number of parameters. When parameters are not too many, or when a sparse parametrization is used, we consider instead the *sensitivity matrix*:

$$\psi_\vartheta := \left(\frac{\partial e_\theta(j)}{\partial \vartheta_i}\right)_{j=1,\dots,Nn_y;\ i=1,\dots,n_\theta}.$$

Thus

$$\begin{aligned} \nabla\tilde{J}(\boldsymbol{\vartheta}) &= \textstyle\sum_{j=1}^{Nn_y} e_\theta(j)\nabla e_\theta(j) = \psi_\vartheta^T \mathbf{e}_\theta, \\ \nabla^2\tilde{J}(\boldsymbol{\vartheta}) &= \textstyle\sum_{j=1}^{Nn_y} \nabla e_\theta \nabla e_\theta^T + \sum_{j=1}^{Nn_y} e_\theta(j)\nabla^2 e_\theta(j) \\ &= \psi_\vartheta^T\psi_\vartheta + \textstyle\sum_{j=1}^{Nn_y} e_\theta(j)\nabla^2 e_\theta(j). \end{aligned} \tag{7.12}$$

The distinctive feature of least-squares problems is that by knowing the Jacobian we can compute the first part of the Hessian $\nabla^2\tilde{J}$ for free. Moreover, this term $\psi_\vartheta^T\psi_\vartheta$ is often more important than the second summation term, either because of near-linearity of the model near the solution (that is $\nabla^2 e_\theta(j)$ small) or because of small residuals (that is, $e_\theta(j)$ small).

To solve the least-squares minimization problem, in the sequel we will adopt a line search method: starting from $\boldsymbol{\vartheta}_0$, we compute a sequence

$$\boldsymbol{\vartheta}_{k+1} = \boldsymbol{\vartheta}_k + \alpha_k \mathbf{s}_k.$$

Instead of generating the search direction $\mathbf{s}_k$ by solving the standard Newton equations

$$\nabla^2\tilde{J}(\boldsymbol{\vartheta}_k)\mathbf{s}_k = -\nabla\tilde{J}(\boldsymbol{\vartheta}_k),$$

we exclude the second-order term form $\nabla^2\tilde{J}$ obtaining the *Gauss Newton* equation

$$\psi_{\vartheta_k}^T\psi_{\vartheta_k}\mathbf{s}_k = -\psi_{\vartheta_k}^T\mathbf{e}_{\vartheta_k}. \tag{7.13}$$

As underlined in (162), this approximation gives some advantages over the plain Newton's method. First, it is not necessary to compute the individual Hessians $\nabla^2 e_\theta(j)$ of

the residuals. Second, in practice there are many interesting situations in which $\psi_\vartheta^T \psi_\vartheta$ is much more significant than $\sum_{j=1}^{Nn_y} e_\theta(j) \nabla^2 e_\theta(j)$, so that the Gauss-Newton method gives performance quite similar to that of Newton's method. A sufficient condition for the dominance of $\psi_\vartheta^T \psi_\vartheta$ is that the size of each second order term $\left\| e_\theta(j) \nabla^2 e_\theta(j) \right\|$ be smaller than the eigenvalues of $\psi_\vartheta^T \psi_\vartheta$. This happens, for instance, when the residuals are small, or when each $e_\theta(j)$ is nearly a linear function. Also the speed of convergence of Gauss-Newton near a solution $\boldsymbol{\vartheta}^*$ depends on how much the leading term $\psi_\vartheta^T \psi_\vartheta$ dominates the second-order term in the Hessian. A third advantage of Gauss-Newton is that whenever ψ_{ϑ_k} has full rank and $\nabla \tilde{J}$ is nonzero, then $\mathbf{s}_k$ is a *descent direction* for $\tilde{J}$:

$$\mathbf{s}_k^T \nabla \tilde{J}(\boldsymbol{\vartheta}_k) = \mathbf{s}_k^T \psi_{\vartheta_k}^T \mathbf{e}_{\vartheta_k} = -\mathbf{s}_k^T \psi_{\vartheta_k}^T \psi_{\vartheta_k} \mathbf{s}_k = -\left\| \psi_{\vartheta_k} \mathbf{s}_k \right\|_2^2 < 0.$$

Finally observe that (7.13) at each iteration are *normal equations*, thus the problem can be solved using e.g. QR or SVD decompositions of ψ_{ϑ_k}. More details can be found in (162).

8

Inverse heat conduction problem

8.1 Introduction

In this chapter we solve numerically an inverse geometric conduction problem of corrosion detection in an unobservable surface of a metal slab, whose thickness and thermo physical properties are known: this study can be found in (138).

Since the corrosion is not directly measurable, we estimate it using a *nondestructive infrared thermographic inspection*. The a priori knowledge about the material object allows us to use a physical-mathematical model to support the estimate. Given a suitable discretization of the corrosion geometric profile, the mathematical problem

consists in estimating the corroded model domain, in particular the corrosion depth at each interval of the discrete profile, starting from the reference (sound) one.

Pulsed infrared thermography becomes practical in detecting hidden corrosion when induced temperature signals are high enough, even if they exist for short time intervals. The 1D approach models only the depth dimension and it therefore assumes that transient thermal events occur simultaneously in sound and corroded areas of the surface: the defects have to be very large so that the boundary heat diffusion effect can be neglected in their center. In such a case an analytical approach is possible. However, when dealing with small defects, the lateral heat diffusion is no longer negligible and must be taken into account (2D and 3D cases) (157). This chapter is focused on the 2D problem: a Finite Element (FE) model is used in an optimization loop to solve the inverse heat transfer problem.

In the framework of the two-dimensional (2D) approach, it has been shown that pulse heating is capable of producing high temperature contrasts but absolute temperature signals might be low due to the insufficient amount of total energy injected into the sample. Oppositely, long heating can significantly warm up the tested object but provides lower contrasts over defects (171). In the literature, different aspects and solution methods for this kind of problems have been studied. In (132, 148) the authors consider the time-harmonic case. Uniqueness and stability have been studied in (134, 135, 158).

In the numerical model adopted in this chapter, the corrosion profile is approximated by a general piecewise-constant function. Since the corrosion profile can have high gradients in unknown positions, the simplest strategy consists in unsing a uniform small subdivision step. However, this corresponds to a large number of parameters to be estimated, increasing the computational complexity of the estimation problem. Considering also its ill-conditioning, it may ask for prohibitive computing times for a real-time diagnostic instrument. In principle less parameters could be sufficient to have a good approximation of the profile, for example where the profile possesses small gradients. In identification theory using a model of complexity not higher than necessary is a guideline (153). In the problem at hand, this can be accomplished by using an *adaptive* subdivision of the profile, based on a posteriori indicators, obtained after iterative comparisons between the experimental measurements and the predictions given by a reference adaptive FE model. In (155), two different algorithms were presented to solve the corrosion estimation problem from the experimental data produced by infrared thermography. While the first one (*inner-outer loop algorithm*) estimates the values of parameters using two nested loops, in the second one (*predictor-corrector*

algorithm), to reduce computational costs, the adaptation of the parametrization is done by a linear predictor step, while parameter estimation is done in the nonlinear corrector step. Following (138), in this chapter a novel formulation of the prediction step is presented.

In section 8.2, the mathematical problem is presented in the general and 2D cases. In section 8.3 a suitable parametrization is chosen for the discrete inverse problem at hand. The numerical strategy is described in section 8.4 and tested in section 8.5.

8.2 Problem formulation

Figure 8.1: *Infrared thermographic inspection: in the time interval* $[0, t_f]$, $t_f > 0$, S *is heated with a thermal flash* $\boldsymbol{q}(t)$ *and experimental temperatures are collected.*

Suppose to deal with a metal slab, $D_c^{(0)}$, whose thickness and thermo physical properties are known, and to interact only with one face S, which is provided with n_y temperature sensors. A nondestructive test is used, consisting of an infrared thermographic inspection: in the time interval $[0, t_f]$, $t_f > 0$, S is heated with a thermal flash $\mathbf{q}(t)$ and experimental temperatures are collected (cfr. figure 8.1). Suppose that the material surface, excluding S, is adiabatic: there is no heat exchange with the outside environment (cfr. Remark 8.2.1).

The underlying mathematical model is based on solving the heat equation on the corroded domain.

More precisely, let $D_c^{(0)} = [0, 1] \times [0, L] \times [-z_0, z_0]$ be the *reference uncorroded (sound) domain* (Figure 8.2 left), $S := \{(x, 0, z),\ x \in [0, 1],\ z \in [-z_0, z_0]\}$ and solve

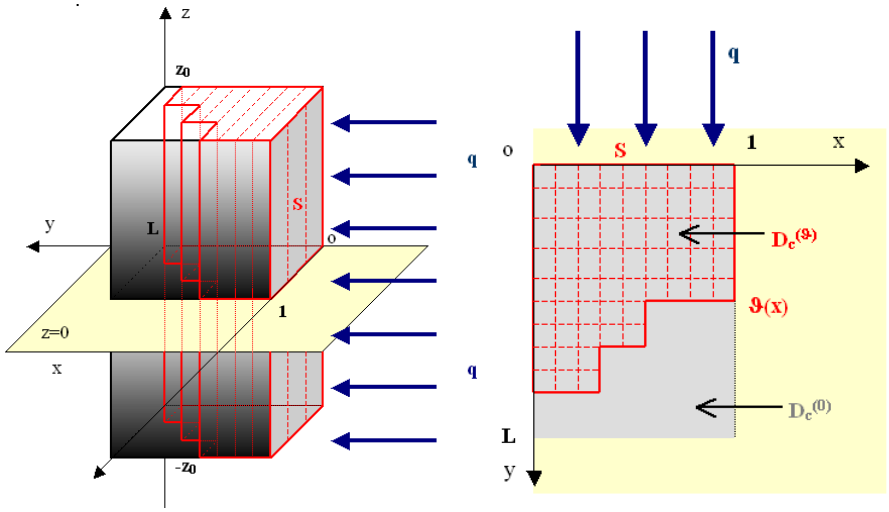

Figure 8.2: *3D problem: corroded piece of material (red), absorbs the heat flux* $\boldsymbol{q}$ *(left); 2D reduction, dealing with its section over* $z = 0$ *(right).*

the following linear heat conduction problem

$$\begin{cases} \rho C \, \frac{\partial}{\partial t} T^{(0)} &= k \, \Delta T^{(0)}, & \text{in } D_c^{(0)} \times [0, t_f] \\ k \, \nabla T^{(0)} \cdot \mathbf{n}_S &= q(t), & \text{on } S \times [0, t_f] \\ k \, \nabla T^{(0)} \cdot \mathbf{n} &= 0, & \text{on } \delta D_c^{(0)} / S \times [0, t_f] \\ T^{(0)}(0, \cdot) &= T_0(\cdot), & \text{in } D_c^{(0)}. \end{cases} \tag{8.1}$$

ρC is the heat capacity of the material, k is its thermal conductivity, and $\mathbf{n}_S$ and $\mathbf{n}$ are respectively the outward normal to S and $\delta D_c^{(0)}/S$. Suppose to know ρC, k and the heat flux $\mathbf{q}(t) = -q(t)\mathbf{n}_S$, which is assumed to be approximately a Dirac impulse in time, centered in $t = 0$, and constant over S. In section 8.5 the heat flux is modelled by $q(t) = \frac{Wt}{\sigma_q^2} e^{-\frac{\sqrt{t}}{\sigma_q}}$, with $\sigma_q > 0$ sufficiently small to have a narrow pulse and $W > 0$. The initial condition $T_0(\cdot)$ is simply set as a constant temperature over the spatial domain. Observe that (8.1) is the *reference sound model*. Consider a temporal discretization of $[0, t_f]$, $\{t_0, \ldots, t_{N-1}\}$, $t_0 = 0$, $t_{N-1} = t_f$. The experimental data of the sound model are denoted by $T_{uc}^s \in \mathbb{R}^{n_y \times N}$, such that $(T_{uc}^s)_{ij}$ represents the temperature in the i-th sensor at time t_{j-1}. The FE solution of (8.1) in S' n_y nodes is denoted by $T_h^{(0)} \in \mathbb{R}^{n_y \times N}$. The quantity

$$\sigma := \left\| T_{uc}^s - T_h^{(0)} \right\|_2$$

is a measure of the goodness of the model.

Consider now the real *corroded domain* $D_c^{(\vartheta)}$ (cfr. the dashed domain in Figure

8.2), described by a scalar function $\vartheta \in \mathcal{L}^2(S)$. The corresponding PDE over $D_c^{(\vartheta)}$ is the following

$$\begin{cases} \rho C \, \frac{\partial}{\partial t} T^{(\vartheta)} &= k \, \Delta T^{(\vartheta)}, & in \; D_c^{(\vartheta)} \times [0, t_f] \\ k \, \nabla T^{(\vartheta)} \cdot \mathbf{n}_S &= q(t), & on \; S \times [0, t_f] \\ k \, \nabla T^{(\vartheta)} \cdot \mathbf{n} &= 0, & on \; \delta D_c^{(\vartheta)} / S \times [0, t_f] \\ T^{(\vartheta)}(0, \cdot) &= T_0(\cdot), & in \; D_c^{(\vartheta)}. \end{cases} \tag{8.2}$$

Assume that the corrosion does not modify the boundary conditions, but only the geometry of the domain. The experimental data of the corroded model are denoted by $T_c^s \in \mathbb{R}^{n_y \times N}$, such that $(T_c^s)_{ij}$ represents the temperature in the i-th sensor at time t_{j-1}.

Supposing that the temperatures T_c^s are known, the *inverse problem* consists in finding a suitable approximation of the real corrosion profile, using a non-destructive approach. This strategy is physically motivated by the fact that, in presence of corrosion, the heat supplied at the surface accessible from the source, S, has less material to diffuse within and the superficial temperature in S remains locally higher for a non-trivial time-interval $[0, t_f]$: a mathematical proof of this property is given in Lemma 8.4.1.

In the following it is assumed that, if we are able to accurately describe the profile of the corrosion, we can describe the thermal response of the corroded system at the same level of accuracy that we do with the uncorroded one, measured by σ.

Remark 8.2.1 *The adiabatic hypothesis is usually invoked in thermal Non-Destructive Evaluation when dealing with thin metal slab. Indeed the key parameter is the Biot number which is defined as the ratio of the heat transfer resistances inside of and at the surface of a body. The smaller the Biot number is the better the approximation of the real thermal process with an adiabatic one is. In figure 8.3, the surface temperature evolutions (analytically computed) for a* 5 *mm thick slab heated by a Dirac energy pulse in adiabatic and non-adiabatic conditions are shown, for two different materials: metal and plastic. In case of plastic, the Biot number is more than 600 times larger than metal. In this example, the assumption of adiabatic thermal process causes errors in temperature less than 0.1% in case of metal and less than 10% for plastic.*

8.2.1 Reduction to a 2D problem

In the following, we assume that the corrosion does not vary along the z-axis, such that (8.1) and (8.2) can be restated as 2D problems, considering $S = [0, 1]$ and $D_c^{(0)} =$

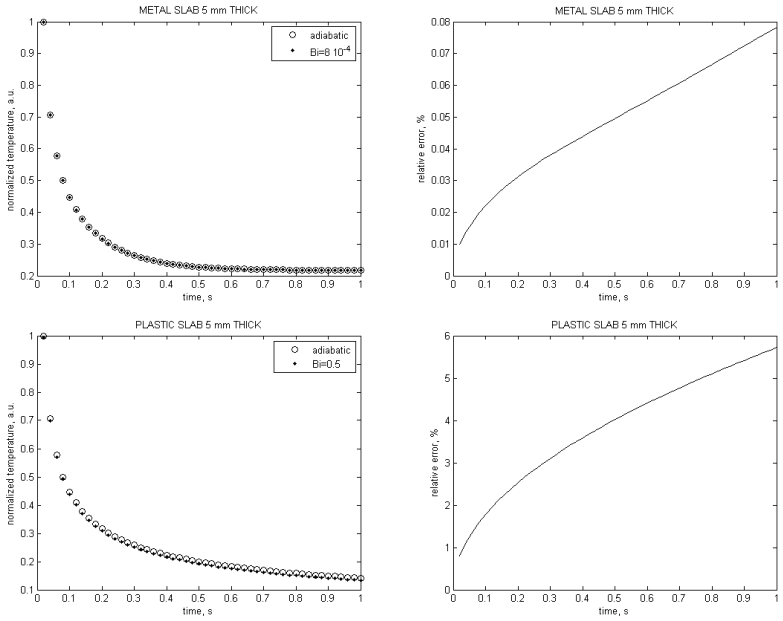

Figure 8.3: *First row: metal slab. Left: comparison between surface temperature evolution of a 5 mm thick metal slab in adiabatic and non-adiabatic conditions. Right: Relative error caused by modeling the non-adiabatic process with an adiabatic one. Second row: plastic slab. Left: comparison between surface temperature evolution of a 5 mm thick plastic slab in adiabatic and non-adiabatic conditions. Right: Relative error caused by modeling the non-adiabatic process with an adiabatic one.*

$[0,1] \times [0,L]$ (Figure 8.2 right). Thus we can describe analytically the corroded region in the following way:

$$D_c^{(\vartheta)} := \{(x,y) \; s.t. \; x \in [0,1], \; 0 \leq y \leq L - \vartheta(x)\},$$

where $\vartheta(x) : \; [0,1] \to [0,L]$ is a suitable smooth non negative function, such that $\vartheta(0) = 0 = \vartheta(1)$, which represents the *corrosion profile*.

8.2.2 Choice of a numerical solution strategy

The idea now is to restate (8.2), defining it on $D_c^{(0)} \times [0,t_f]$, i.e. on the sound domain, modifying properly the PDE coefficients. This is important because in (8.2) it is intuitive that the shape of corrosion influences the temperature profile, but it is not evident how it enters in the PDE, since it characterizes only the geometrical domain. To obtain an equivalent analytical problem defined in $D_c^{(0)} \times [0,t_f]$, we use some ideas introduced in (132).

8. INVERSE HEAT CONDUCTION PROBLEM

Let $F:\ D_c^{(\vartheta)} \to D_c^{(0)}$, $(x,y) \mapsto (\xi,\zeta)$ be a smooth change of coordinates such that

$$\begin{aligned} \xi &= F_1(x,y) = x, \\ \zeta &= F_2(x,y) = y + \vartheta(x)\psi(y), \end{aligned}$$

where $\psi(y):\ [0,L] \to \mathbb{R}$ is a suitable smooth non decreasing function, such that $\psi(0)=0$, $\psi(L)=1$. Define now

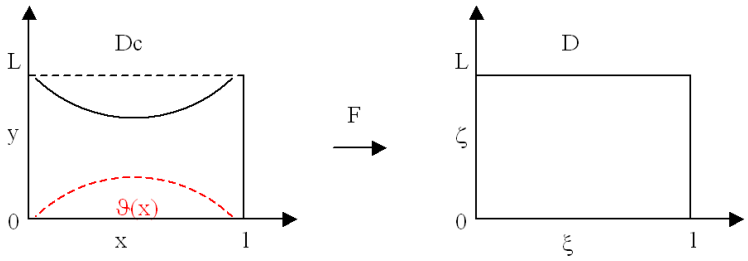

Figure 8.4: *Bidimensional study: corroded and uncorroded domains.*

$$v := T \circ F^{-1}:\ D_c^{(0)} \to \mathbb{R},$$

$v(\xi,\zeta) = T(x,y)\,|_{(x,y)=F^{-1}(\xi,\zeta)}$.

It can be shown (132) that such v satisfies the following heat equation

$$\begin{cases} \frac{\rho C}{|JF|}\frac{\partial}{\partial t}v &= \nabla\cdot(k\frac{JF\ JF^T}{|JF|})\nabla v, & in\ D_c^{(0)}\times[0,t_f] \\ k\,\nabla v\cdot\mathbf{n}_S &= q(t), & on\ S\times[0,t_f] \\ k\,\nabla v\cdot\mathbf{n} &= 0, & on\ \delta D_c^{(0)}/S\times[0,t_f] \\ v(t_0,\xi,\zeta) &= T_0\circ F^{-1}(\xi,\zeta), & in\ D_c^{(0)}, \end{cases} \tag{8.3}$$

using the hypothesis on ϑ and ψ, and where JF is the Jacobian matrix of F:

$$JF(x,y)\,|_{(x,y)=F^{-1}(\xi,\zeta)} = \begin{pmatrix} 1 & \vartheta'(x)\psi(y) \\ 0 & 1+\vartheta(x)\psi'(y) \end{pmatrix}_{|(x,y)=F^{-1}(\xi,\zeta)}.$$

In (132), under suitable hypothesis, the corrosion estimation problem has been solved analytically: assuming a sinusoidal impulse $q(t)$, using a change of coordinates, (8.2) is rewritten as a heat equation over the sound domain $D_c^{(0)}\times[0,t_f]$, with its PDE coefficients depending on ϑ. However, we assume that the heating flux $q(t)$ is approximately a *Dirac pulse heating*: this choice is motivated by higher contrast signal and shorter test duration. Indeed pulse thermography, that is a transient technique, does not require the sample to reach the stationary periodic regime, as in case of an harmonic heating. Since the analytical solution of the corresponding heat equation becomes very difficult, a numerical approach has been adopted.

8.3 The discrete inverse problem

First of all a particular approximation of the real corrosion profile $\vartheta(x)$ is introduced, choosing a *piecewise constant function* (cfr. figure 8.5). This approach characterize

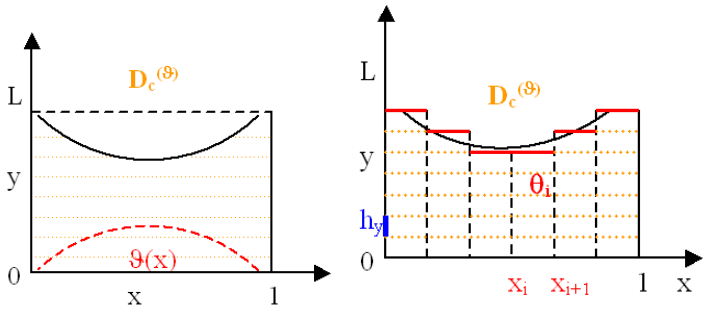

Figure 8.5: *2D study: corroded domain (left) and a piecewise constant approximation (right).*

the inverse problem from a geometrical point of view and it can be seen as additional a priori information about the problem. As explained below, under this hypothesis, instead of estimating a continous unknown $\vartheta(x)$, we hand up with a vectorial parameter estimation problem, and thus with a *discrete inverse problem.*

To understand how this can be done, consider a subdivision of $[0,1]$, coincident with a subset of the n_y temperature sensors' locations, with distinct spatial nodes $\{x_i\}_{i=1,\dots,n_\theta}$, $n_\theta \leq n_y$, $x_0 = 0$, $x_{n_\theta} = 1$, and a uniform subdivision of $[0, L]$, with step h_y, $\{y_i\}_{i=0,\dots,n_L}$, $y_0 = 0$, $y_{n_L} = L$. Define

$$\theta_j := \frac{1}{h_c(j)} \int_{x_j}^{x_{j+1}} \vartheta(x)dx \approx L - y_k,$$

for a suitable $k \in \{0, \dots, n_L\}$, $h_c(j) := |x_{j+1} - x_j|$, $j = 1, \dots, n_\theta - 1$.

Consider now the set of functions

$$\mathcal{P} = \left\{ \tilde{\vartheta} \; s.t. \; \tilde{\vartheta} : \; [0,1] \longrightarrow [0,L], \; \tilde{\vartheta}(x) = \sum_{j=1}^{n_\theta - 1} \theta_j \chi_{[x_i, x_{i+1})}(x) \right\},$$

where $\chi_{[x_i - x_{i+1})}(x) = \begin{cases} 1, \; x \in [x_i - x_{i+1}) \\ 0, \; \text{elsewhere} \end{cases}$ is the characteristic function of $[x_i, x_{i+1})$.
The approximated corroded domain is defined as follows

$$D_c^{(\tilde{\theta})} := D_c^{(0)} \setminus \int_0^1 \tilde{\vartheta}(x)dx.$$

Thus $D_c^{(\tilde{\theta})}$ is identified by the vector of parameters $\boldsymbol{\theta} \in \mathbb{R}^{n_\theta - 1}$.

Define now the *matrix of prediction errors* $E_\theta := T_c^s - T_h^{(\theta)} \in \mathbb{R}^{n_y \times N}$ where $T_h^{(\theta)} \in \mathbb{R}^{n_y \times N}$ denotes the FE solution at every time discretization point in S' n_y nodes, solving (8.2) on the approximated corroded domain $D_c^{(\tilde{\theta})}$.

Consider the real valued function $\tilde{J}:\ \mathbb{R}^{n_\theta - 1} \to \mathbb{R}$,

$$\tilde{J}(\boldsymbol{\theta}) := \frac{1}{N} \sum_{n=1}^{N} \|E_\theta(\cdot, n)\|_2^2 \,. \tag{8.4}$$

It corresponds to find the optimal $\tilde{\vartheta}^* \in \mathcal{P}$, or equivalently the optimal parameters θ_j^*, $j = 1, \ldots, n_\theta - 1$ such that

$$\boldsymbol{\theta}^* = \arg \min_{\boldsymbol{\theta} \in \mathbb{R}^{n_\theta - 1}} \tilde{J}(\boldsymbol{\theta}). \tag{8.5}$$

Reshaping the matrices, define $\mathbf{e}_\theta, \mathbf{y}, \hat{\boldsymbol{y}}_\theta \in \mathbb{R}^{n_y N}$ such that

$$\begin{aligned} \mathbf{e}_\theta((n-1)n_y + 1 : nn_y) &= E_\theta(:, n), \\ \mathbf{y}((n-1)n_y + 1 : nn_y) &= T_c^s(:, n), \\ \hat{\boldsymbol{y}}_\theta((n-1)n_y + 1 : nn_y) &= T_\theta^h(:, n) \end{aligned}$$

$n = 1, \ldots, N$. The *sensitivity matrix* $\psi \in \mathbb{R}^{n_y N \times n_\theta}$ is such that $\psi_\theta(:, i) := \frac{\partial}{\partial \theta_i} \hat{\boldsymbol{y}}_\theta$, for all $i = 1, \ldots, n_\theta$.

Thus

$$\tilde{J}(\boldsymbol{\theta}) = \frac{1}{N} \sum_{n=1}^{N} \|\mathbf{e}_\theta((n-1)n_y + 1 : nn_y)\|_2^2 = \frac{1}{N} \sum_{n=1}^{N} \|\mathbf{y} - \hat{\boldsymbol{y}}_\theta((n-1)n_y + 1 : nn_y)\|_2^2 \,.$$

Observe that

$$\begin{aligned} \tilde{J}(\boldsymbol{\theta}) &= \tfrac{1}{N} \textstyle\sum_{n=1}^{N} \|\mathbf{y} - \hat{\boldsymbol{y}}_\theta((n-1)n_y + 1 : nn_y)\|_2^2 = \tfrac{1}{N} \textstyle\sum_{n=1}^{N} \sum_{l=1}^{n_y} (y(l) - \hat{y}_\theta(l))^2 \\ &= \tfrac{1}{N} \textstyle\sum_{k=1}^{n_y N} (y(k) - \hat{y}_\theta(k))^2 = \tfrac{1}{N} \|\mathbf{y} - \hat{\boldsymbol{y}}\|_2^2 \,. \end{aligned}$$

Thus our functional coincides with the minimization of the square of the 2-norm in $\mathbb{R}^{n_y N}$ of the prediction error. Observe that this is a *least-squares problem* of the form (7.11).

Since a piecewise constant approximation of the corrosion profile $\vartheta(x)$ is chosen, it must be assumed that every parameter corresponds to a well-defined piece of the real corrosion profile, whose length strictly depends on the local behavior of $\vartheta(x)$. Moreover it is assumed to deal with non overlapping parameters.

In the following the approximated corroded profile is identified with the real corroded one, which is thus assumed to be piecewise constant. Also numerical experimental data are collected assuming a profile belonging to $\mathcal{P}$.

8.4 Adopted numerical approach

Assuming that the corrosion profile is a parametric piecewise constant function, the aim of the numerical algorithm is to estimate the real shape of the domain. It solves the discrete inverse problem (8.5), using (8.2) as the underlying direct model, solved in the approximated corroded domain $D_c^{(\tilde{\theta})}$.

8.4.1 Key assumption

Let $\bar{i} \in [1, n_\theta - 1]$, a key assumption in the development of the algorithm is that when at iteration $k+1$, $k \geq 0$, the estimation algorithm changes the $\bar{i}$-th component of the estimate $\hat{\theta}^{(k)}(\bar{i})$ to a value $\hat{\theta}^{(k+1)}(\bar{i})$ closer to the real one, leaving unchanged the others, the cost function diminishes monotonically. This property is an immediate consequence of the following physical principle: in $[0, t_f]$, under the same initial and boundary conditions, if $D_{c,1} \subset D_{c,2}$, then temperatures corresponding to the smallest domain $D_{c,1}$ are higher supposing that we are dealing with initial constant temperatures and a thermal flux q which is independent on the space variable. In fact, assuming that $D_c^{(\theta)^*} \subset D_c^{(\hat{\theta}^{(k+1)})} \subset D_c^{(\hat{\theta}^{(k)})}$, it can be deduced that $T_s^c > T_h^{(\hat{\theta}^{(k+1)})} > T_h^{(\hat{\theta}^{(k)})}$ and thus $\tilde{J}(\hat{\boldsymbol{\theta}}^{(k+1)}) < \tilde{J}(\hat{\boldsymbol{\theta}}^{(k)})$. A rigorous proof of this property is given in the following Lemma.

Lemma 8.4.1 *Consider the heat problems represented in Figure 8.6, solving the heat*

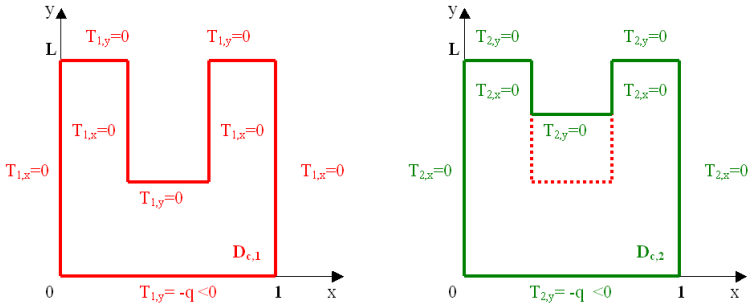

Figure 8.6:

equation model

$$\begin{cases} \rho C \frac{\partial}{\partial t} T_i &= k \, \Delta T_i, & in \ D_{c,i} \times [0, t_f] \\ k T_{i,y} &= -q(t), & on \ S \times [0, t_f] \\ k \, \nabla T_i \cdot \boldsymbol{n} &= 0, & on \ \delta D_{c,i} \setminus S \times [0, t_f] \\ T_i(0, x, y) &= T_0(x, y), & in \ D_{c,i}. \end{cases} \tag{8.6}$$

where $i = 1, 2$ *and* $S = [0, 1]$. *Moreover define* $\frac{\partial T_i}{\partial x} = T_{i,x}$ *and* $\frac{\partial T_i}{\partial y} = T_{i,y}$. *Suppose that temperatures at* $t = 0$ *are constant in space,* $T_0(x, y) = T_0 \in \mathbb{R}$, *and that* $q(t, x) = q(t) > 0$. *Then* $T_1(x, y) > T_2(x, y)$ *for every* $(x, y) \in D_{c,1}$.

Proof. The proof is a consequence of the *maximum principle for parabolic operators* (141). To compare temperatures T_1 and T_2 in $D_{c,1}$, it is necessary to collect information about the values taken by T_2 in $D_{c,1}$. To do this, define $v := T_{2,x}$: v satisfies the heat equation $\frac{\partial v}{\partial t} = \Delta v$, under the boundary conditions of Figure 8.7 (up left), where we have used $v_y = T_{2,xy} = (T_{2,y})_{,x}$. Thus, using the maximum principle for parabolic operators, we know that strictly maximum and minimum values are taken at the boundary, where Dirichlet boundary conditions are applied, or at $t = 0$. Since we assume that T_0 is constant, $v = 0$ at $t = 0$. It follows that

$$T_{2,x}(t, x, y) = v(t, x, y) = 0$$

for every $(t, x, y) \in [0, t_f] \times D_{c,2}$. Moreover define $z := T_{2,y}$: z is a solution of the heat

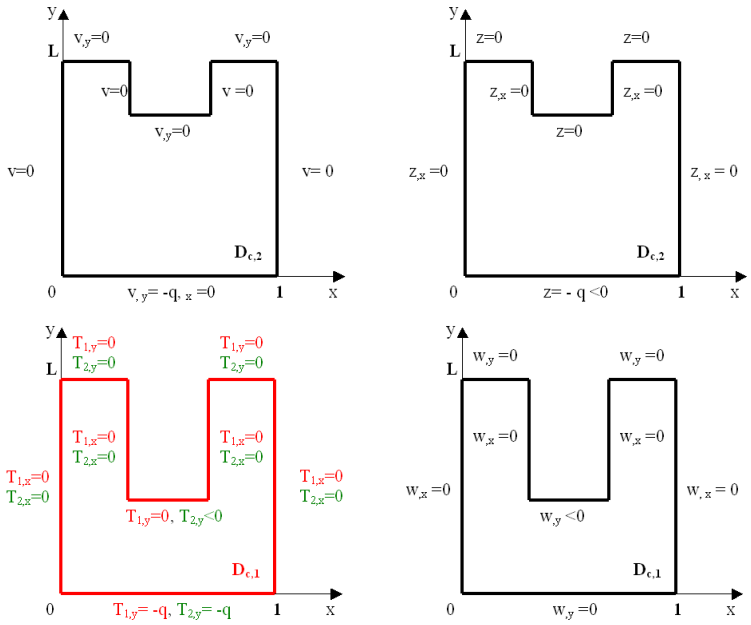

Figure 8.7: $v := T_{2,x}$, $z := T_{2,y}$, $w := T_2 - T_1$.

equation $\frac{\partial z}{\partial t} = \Delta z$, under the boundary conditions of Figure 8.7 (up right), where we have used $z_x = T_{2,yx} = (T_{2,x})_{,y}$. Using again the maximum principle we conclude that

$$-q(t) \leq T_{2,y}(t, x, y) = z(t, x, y) \leq 0,$$

for every $t \in [0, t_f]$ and (x, y) in the interior of $D_{c,2}$. All this information is summarized in Figure 8.7 (bottom left). Now we can compare directly T_1 and T_2 over $D_{c,1} \subset D_{c,2}$: define $w := T_2 - T_1$, which solves the heat equation $\frac{\partial w}{\partial t} = \Delta w$, under the boundary conditions of Figure 8.7 (bottom right). Now we can again apply the maximum principle. Since we suppose that at $t = 0$ $T_2 = T_1$, $w = 0$ in $t = 0$. Moreover if a maximum is taken, then it must be placed on the boundary, where $\frac{\partial w}{\partial n} > 0$ (141). Since is always $\frac{\partial w}{\partial n} \leq 0$, then a maximum does not exists, thus $w < 0$ over $D_{c,1}$.

□

8.4.2 Projected damped Gauss-Newton iterations

Given a subdivision of the interval S, the inner loop consists in solving the discrete inverse problem (8.5) using a *projected damped Gauss-Newton method*: at every iteration k, the inner loop finds an estimate $\hat{\boldsymbol{\theta}}^k$, $k \geq 0$ of $\boldsymbol{\theta}^*$.

The Damped Newton method for this problem is sketched in algorithm 2. Now

Algorithm 2 Damped Newton:

$\hat{\boldsymbol{\theta}}^0 = \mathbf{0}$, $\mu^0 = 1$;
for $k = 0 : n_{max}$ **do**
  solve $\tilde{J}''(\hat{\boldsymbol{\theta}}^k)\mathbf{s}^k = -\tilde{J}'(\hat{\boldsymbol{\theta}}^k)$, $\tilde{J}''(\hat{\boldsymbol{\theta}}^k) \in \mathbb{R}^{n_\theta - 1 \times n_\theta - 1}$, $\tilde{J}'(\hat{\boldsymbol{\theta}}^k) \in \mathbb{R}^{n_\theta - 1}$;
  $\hat{\boldsymbol{\theta}}^{k+1} = \hat{\boldsymbol{\theta}}^k + \mu^k \mathbf{s}^k$
  compute $\tilde{J}(\hat{\boldsymbol{\theta}}^{k+1})$
  if $\tilde{J}(\hat{\boldsymbol{\theta}}^{k+1}) < \tilde{J}(\hat{\boldsymbol{\theta}}^k)$ **then**
    $\mu^{k+1} = \mu^k$
  else
    $l = 0$;
    $\mu^{k,l} = \frac{\mu^k}{2}$
    while $\tilde{J}(\hat{\boldsymbol{\theta}}^{k+1}) < \tilde{J}(\hat{\boldsymbol{\theta}}^k)$ **do**
      $\hat{\boldsymbol{\theta}}^{k+1} = \hat{\boldsymbol{\theta}}^k + \mu^{k,l}\mathbf{s}^k$
      $l = l + 1$;
      $\mu^{k,l} = \frac{\mu^{k,l}}{2}$
    end while
    $\mu^{k+1} = \mu^{k,l}$
  end if
end for

some computations are done to simplify numerically the algorithm, computing $\tilde{J}''(\boldsymbol{\theta})$ and $\tilde{J}'(\boldsymbol{\theta})$, Hessian and Jacobian of $\tilde{J}(\boldsymbol{\theta})$ respectively, as explained in section 7.2.2.1. Suppose that $i, j = 1, \ldots, n_\theta$.

$$\begin{aligned} \frac{\partial \tilde{J}}{\partial \theta_i} &= \frac{1}{N}\sum_{n=1}^{N} \frac{\partial}{\partial \theta_i}\left[(\mathbf{y} - \hat{\boldsymbol{y}}_\theta((n-1)n_y + 1 : nn_y)) \cdot (\mathbf{y} - \hat{\boldsymbol{y}}_\theta((n-1)n_y + 1 : nn_y))\right] \\ &= -\frac{2}{N}\sum_{n=1}^{N} \mathbf{e}_\theta((n-1)n_y + 1 : nn_y) \cdot \frac{\partial}{\partial \theta_i}\left(\hat{\boldsymbol{y}}_\theta((n-1)n_y + 1 : nn_y)\right). \end{aligned}$$

Define now the sensitivity matrix

$$\psi_\theta \in \mathbb{R}^{n_y N \times n_\theta},\ \psi_\theta(:,i) := \frac{\partial}{\partial \theta_i}\hat{\boldsymbol{y}}_\theta.$$

$$\begin{aligned}\frac{\partial \tilde{J}}{\partial \theta_i} &= -\frac{2}{N}\sum_{n=1}^{N}\mathbf{e}_\theta((n-1)n_y+1:nn_y)\cdot\psi_\theta((n-1)n_y+1:nn_y,i)\\ &= -\frac{2}{N}\sum_{k=1}^{n_y N} e_\theta(k)\psi_\theta(k,i).\end{aligned}$$

Finally

$$\tilde{J}'(\boldsymbol{\theta}) = -\frac{2}{N}\psi_\theta^T\mathbf{e}_\theta.$$

$$\begin{aligned}\frac{\partial^2 \tilde{J}}{\partial \theta_i \theta_j} &= -\frac{2}{N}\sum_{k=1}^{n_y N}\frac{\partial}{\partial \theta_j}e_\theta(k)\psi_\theta(k,i) - \frac{2}{N}\sum_{k=1}^{n_y N} e_\theta(k)\frac{\partial}{\partial \theta_j}\psi_\theta(k,i)\\ &= \frac{2}{N}\sum_{k=1}^{n_y N}\psi_\theta(k,j)\psi_\theta(k,i) - \frac{2}{N}\sum_{k=1}^{n_y N} e_\theta(k)\frac{\partial}{\partial \theta_j}\psi_\theta(k,i).\end{aligned}$$

Thus

$$\tilde{J}''(\boldsymbol{\theta}) \approx \frac{2}{N}\psi_\theta^T\psi_\theta :$$

this approximation corresponds to use a *Gauss-Newton method* (cfr. equation (7.13)).

As explained in the previous chapter, given the damping parameter μ^k and $\hat{\boldsymbol{\theta}}^k$, the $k+1$-th iteration $\hat{\boldsymbol{\theta}}^{k+1} = \hat{\boldsymbol{\theta}}^k + \mu^k\mathbf{s}^k$, is obtained substituting the standard Newton step

$$\tilde{J}''(\hat{\boldsymbol{\theta}}^k)\mathbf{s}^k = -\tilde{J}'(\hat{\boldsymbol{\theta}}^k)$$

by the Gauss-Newton approximation. The last one corresponds to solve $\frac{2}{N}\psi_{\hat{\theta}^k}^T\psi_{\hat{\theta}^k}\mathbf{s}^k = \frac{2}{N}\psi_{\hat{\theta}^k}^T\mathbf{e}_{\hat{\theta}^k}$, i.e. the following overdetermined system

$$\psi_{\hat{\theta}^k}\mathbf{s}^k = \mathbf{e}_{\hat{\theta}^k}$$

in a least square sense (cfr. algorithm 3).

To compute numerically the sensitivity matrix a centered finite difference scheme is needed: making the dependence of $\hat{\boldsymbol{y}} \in \mathbb{R}^{n_y N}$ on $\boldsymbol{\theta}$ explicit, $\hat{\boldsymbol{y}} = \hat{\boldsymbol{y}}(\boldsymbol{\theta}) = \hat{\boldsymbol{y}}(\theta_1,\ldots,\theta_{n_\theta-1})$ we wrote

$$\psi_\theta(:,i) = -\frac{\partial}{\partial \theta_i}\hat{\boldsymbol{y}} = \frac{1}{\delta\theta}[\hat{\boldsymbol{y}}(\theta_1,\ldots,\theta_i+\frac{\delta\vartheta}{2},\ldots,\theta_{n_\theta-1}) - \hat{\boldsymbol{y}}(\theta_1,\ldots,\theta_i-\frac{\delta\vartheta}{2},\ldots,\theta_{n_\theta-1})].$$

Computing ψ_θ is expensive, since two different predicted temperatures, corresponding to the perturbations of the i-th parameter, must be computed in order to estimate one single column: this is computationally expensive, since, in order to estimate one single column, two different temperatures predictions must be computed. It can be used if the number of parameters to be estimated is sufficiently small.

The perturbation of different parameters may produce quite similar responses in the simulation data, generating couples of columns in the matrix ψ_θ which are close to linear dependence. In our problem, this is related to the length of the corrosion

Algorithm 3 Projected Gauss-Newton method (INNER LOOP):

Given a fixed subdivision of $[0,1]$, $\{x_1, \dots, x_{n_\theta}\}$:
$\hat{\theta}^0 = \mathbf{0}$, $\mu^0 = 1$;
for $k = 1 : n_{max}$ **do**
  solve $\psi_{\hat{\theta}^k} \mathbf{s}^k = \mathbf{e}_{\hat{\theta}^k}$;
  $\hat{\boldsymbol{\theta}}^{k+1} = \hat{\boldsymbol{\theta}}^k + \mu^k \mathbf{s}^k$
  projection: for every $j \in [0, n_\theta - 1]$ s.t. $\hat{\theta}^{k+1}(j) < 0$, impose $\hat{\theta}^{k+1}(j) = 0$; for every $m \in [0, n_\theta - 1]$ s.t. $\hat{\theta}^{k+1}(m) > L$, impose $\hat{\theta}^{k+1}(m) = L$
  compute $\tilde{J}(\hat{\boldsymbol{\theta}}^{k+1})$
  if $\tilde{J}(\hat{\boldsymbol{\theta}}^{k+1}) < \tilde{J}(\hat{\boldsymbol{\theta}}^k)$ **then**
    $\mu^{k+1} = \mu^k$
  else
    $l = 0$;
    $\mu^{k,l} = \frac{\mu^k}{2}$
    while $\tilde{J}(\hat{\boldsymbol{\theta}}^{k+1}) < \tilde{J}(\hat{\boldsymbol{\theta}}^k)$ **do**
      $\hat{\boldsymbol{\theta}}^{k+1} = \hat{\boldsymbol{\theta}}^k + \mu^{k,l} \mathbf{s}^k$
      $l = l + 1$;
      $\mu^{k,l} = \frac{\mu^{k,l}}{2}$
    end while
    $\mu^{k+1} = \mu^{k,l}$
  end if
end for
$\hat{\boldsymbol{\theta}} \in \mathbb{R}^{n_\theta - 1}$ is the optimal parameter estimation on $\{x_1, \dots, x_{n_\theta}\}$

profile segment corresponding to each parameter. Thus, *the presence of short segments* $h_c(i)$, $i = 1, \ldots, n_\theta - 1$, produces, in general, an ill-conditioned matrix ψ_θ. Therefore, the search for a better accuracy in the determination of the corrosion profile, which means to reduce the size of a few parameters, brings to higher numerical problems, as usually happens solving inverse problems (131). Thus a regularization technique is needed (155): we choose to adopt an adaptive parametrization, using also the *Truncated Singular Value Decomposition* (TSVD).

8.4.3 Convergence properties of the projected damped Gauss-Newton method

In this section it will be proved that, if the finer parametrization is chosen and the sites of corrosion are known, then the inverse problem of corrosion estimation does not admit local minima.

Suppose to use the finer parametrization and to know exactly the sites of corrosion $\{\bar{i}_1, \ldots, \bar{i}_l\}$ of the real corroded profile $\boldsymbol{\theta}^*$, $\bar{i}_j \in [1, n_y]$. Thus $\tilde{J}(\boldsymbol{\theta^*}) < \tilde{J}(\boldsymbol{\theta})$ for every $\boldsymbol{\theta} \in \Psi$, where Ψ denotes the set of profiles with corrosion sites $\{\bar{i}_1, \ldots, \bar{i}_l\}$:

$$\Psi = \left\{ \boldsymbol{\theta} \in \mathbb{R}^{n_\theta - 1} \; s.t. \; n_\theta = n_y, \; \theta(j) = 0, \; \forall j \notin \{\bar{i}_1, \ldots, \bar{i}_l\} \right\}.$$

A local minima $\bar{\boldsymbol{\theta}}$, is a corrosion profiles such that $\tilde{J}(\bar{\boldsymbol{\theta}}) < \tilde{J}(\boldsymbol{\theta})$ for every $\boldsymbol{\theta} \in \Psi_{\bar{\theta}}$, where $\Psi_{\bar{\theta}}$ is the set of Ψ's profiles perturbed of a quantity δ:

$$\Psi_{\bar{\theta}} = \left\{ \boldsymbol{\theta} \in \Psi, \; \theta(j) = \bar{\theta}(j) + \delta_j, \; \forall j \in \{\bar{i}_1, \ldots, \bar{i}_l\} \right\}, \; \delta_j \in \{0, h_y, -h_y\}, \; \boldsymbol{\delta} \neq \mathbf{0}.$$

The following Proposition is equivalent to prove that there are no local minima.

Proposition 8.4.1 *For every $\bar{\boldsymbol{\theta}} \in \Psi$, $\bar{\boldsymbol{\theta}} \neq \boldsymbol{\theta^*}$, there exists at least a sequence of profiles $\{\boldsymbol{\theta}\}_n$, $\boldsymbol{\theta}_0 = \bar{\boldsymbol{\theta}}$, $\boldsymbol{\theta}_{n+1} \in \Psi_{\theta_n}$, converging decreasing in $\mathcal{L}^2(\mathbb{R}^{n_\theta - 1})$ to the real profile $\boldsymbol{\theta}^*$, such that $\tilde{J}(\boldsymbol{\theta}_n) \downarrow \tilde{J}(\boldsymbol{\theta}^*)$.*

First of all we demonstrate the following Lemma.

Lemma 8.4.2 *Given $\{\bar{i}_1, \ldots, \bar{i}_l\}$, for every $\bar{\boldsymbol{\theta}} \in \Psi$, $\bar{\boldsymbol{\theta}} \neq \boldsymbol{\theta}^*$, and for every $k \in [1, l]$ such that $\theta^*(i_k) \neq \bar{\theta}(i_k)$, define*

$$\theta^{*,k}(j) := \begin{cases} \bar{\theta}(j), & j \neq \bar{i}_k \\ \theta^*(j), & j = \bar{i}_k \end{cases}, \tag{8.7}$$

$j = 1, \ldots, n_\theta - 1$. Thus for every $\bar{\boldsymbol{\theta}} \in \Psi$ there exists at least a sequence of profiles $\{\boldsymbol{\theta}_n\}_n$, $\boldsymbol{\theta}_0 = \bar{\boldsymbol{\theta}}$, $\boldsymbol{\theta}_{n+1} \in \Psi_{\theta_n}$, converging decreasing in $\mathcal{L}^2(\mathbb{R}^{n_\theta - 1})$ to $\boldsymbol{\theta}^{*,k}$: $\tilde{J}(\boldsymbol{\theta}_n) \downarrow \tilde{J}(\boldsymbol{\theta}^{*,k})$.

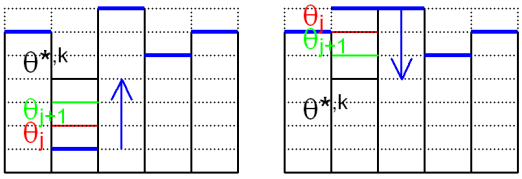

Figure 8.8: *Converging sequence (8.8) (left) and (8.9) (right): in blue $\bar{\theta}(\bar{i}_k)$, in black the optimal profile $\theta^{*,k}(\bar{i}_k)$, in red $\theta_j(\bar{i}_k)$ and in green $\theta_{j+1}(\bar{i}_k)$.*

Proof.(Lemma 8.4.2)
We indicate with $T_c^{s,k}$ temperatures corresponding to $\boldsymbol{\theta}^{*,k}$.
Let $r \in \mathbb{Z}$ such that $\bar{\theta}(\bar{i}_k) - \theta^{*,k}(\bar{i}_k) = rh_y \neq 0$ by hypothesis.
Suppose that $r > 0$: consider the following converging sequence

$$\theta_n(j) := \begin{cases} \bar{\theta}(j), & j \neq \bar{i}_k \\ \bar{\theta}(j) - nh_y, & j = \bar{i}_k \end{cases}, \tag{8.8}$$

for $n = 0, \ldots, r$, $\theta_r = \theta^{*,k}$ by construction, and $\theta_s := \theta_r$, $s \geq n$. An example is sketched in Figure 8.8 (left). By definition $\theta_0(\bar{i}_k) > \ldots > \theta_j(\bar{i}_k) > \theta_{j+1}(\bar{i}_k) > \ldots > \theta^{*,k}(\bar{i}_k)$, or equivalently $D_c^{(\theta_0)} \subset \ldots \subset D_c^{(\theta_j)} \subset \ldots \subset D_c^{(\theta^{*,k})}$. Thus the underlying heat equation operator tells us that temperatures corresponding to θ_j are greater than those corresponding to θ_{j+1}, and they are both greater than $T_c^{s,k}$, applying Lemma 8.4.1. Thus $\tilde{J}(\boldsymbol{\theta}_0) > \ldots > \tilde{J}(\boldsymbol{\theta}^{*,k})$.

Finally observe that if $r < 0$, the proof is analogous considering

$$\theta_n(j) := \begin{cases} \bar{\theta}(j), & j \neq \bar{i}_k \\ \bar{\theta}(j) + nh_y, & j = \bar{i}_k \end{cases}, \tag{8.9}$$

for $n = 0, \ldots, |r|$ instead of (8.8) (Figure 8.8 (right)).

□

Proof. (Proposition 8.4.1).

We use the above Lemma 8.4.2. In fact, suppose that the corrosion sites are $\{\bar{i}_1, \ldots, \bar{i}_l\}$. Thus we can construct the sequence $\{\boldsymbol{\theta}_n\}_n$ in the following way. Suppose that $k_1 \in [1, l]$ is the first index such that $\bar{\theta}(\bar{i}_{k_1}) - \theta^*(\bar{i}_{k_1}) = r_1 h_y$, $r_1 \in \mathbb{Z} \setminus \{0\}$. Thus

$$\theta_n(j) := \begin{cases} \bar{\theta}(j), & j \in [1, n_\theta - 1] \setminus \{\bar{i}_{k_1}\} \\ \bar{\theta}(j) + nh_y, & j = \bar{i}_{k_1},\ r_1 < 0 \\ \bar{\theta}(j) - nh_y, & j = \bar{i}_{k_1},\ r_1 > 0 \end{cases}, \tag{8.10}$$

for $n = 0, \ldots, |r_1|$. Using Lemma 8.4.2 we know that $\tilde{J}(\boldsymbol{\theta}_0) > \tilde{J}(\boldsymbol{\theta}_1) > \ldots > \tilde{J}(\boldsymbol{\theta}_{|r_1|})$ and by construction

$$\theta_{|r_1|}(j) = \begin{cases} \bar{\theta}(j), & j \in [1, n_\theta - 1] \setminus \{\bar{i}_{k_1}\} \\ \theta^*(j), & j = \bar{i}_{k_1} \end{cases} .$$

Now we choose the second index $k_2 \in [1, l]$ such that $\bar{\theta}(\bar{i}_{k_2}) - \theta^*(\bar{i}_{k_2}) = r_2 h_y,\ r_2 \in \mathbb{Z} \setminus \{0\}$ and define $\boldsymbol{\theta}_{|r_1|+1}, \ldots, \boldsymbol{\theta}_{|r_1|+|r_2|}$, using (9.20), replacing r_1 with r_2 and k_1 with k_2. Using Lemma 8.4.2 we know that $\tilde{J}(\boldsymbol{\theta}_{|r_1|}) > \tilde{J}(\boldsymbol{\theta}_{|r_1|+1}) > \ldots > \tilde{J}(\boldsymbol{\theta}_{|r_2|})$ and by construction

$$\theta_{|r_2|}(j) = \begin{cases} \bar{\theta}(j), & j \in [1, n_\theta - 1] \setminus \{\bar{i}_{k_1}, \bar{i}_{k_2}\} \\ \theta^*(j), & j \in \{\bar{i}_{k_1}, \bar{i}_{k_2}\} \end{cases} .$$

This idea can be repeated for every $k \in [1, l]$ such that $\bar{\theta}(\bar{i}_k) - \theta^*(\bar{i}_k) \neq 0$. Finally we obtain the desired decreasing sequence $\{\boldsymbol{\theta}_n\}_n$, converging in $\mathcal{L}^2(\mathbb{R}^{n_\theta - 1})$ to the real profile $\boldsymbol{\theta}^*$, such that $\tilde{J}(\boldsymbol{\theta}_n) \downarrow \tilde{J}(\boldsymbol{\theta}^*)$.

□

8.4.4 Predictor-Corrector algorithm

As demonstrated in the previous section, the problem has no local minima, if the finer discretization is used and the sites of corrosion are known. Unfortunately the last ones are unknown and using the finer discretization is very expensive and ill-conditioned.

To deal with the intrinsic ill-posedness of the inverse problem of corrosion detection, an *adaptive formulation* is adopted, to reduce the computational cost. The adaptive parametrization is determined starting from an initial subdivision of the corrosion profile with a quite large $h_c(i)$, $i = 1, \ldots, n_\theta - 1$. According to a suitable *a posteriori indicator*, the algorithm decides where eventually to refine locally the subdivision of the corrosion profile. The refinement operation corresponds to a bisection of the indicated segments, with a consequent increase in the number of segments and, therefore, of parameters of the model. This is iteratively made until the comparison between the actual value of the cost function and the reference value, previously obtained for the sound (uncorroded) system, shows that the model describes the experimental data in an optimal way. As in (155), the *a posteriori indicator* is based upon parameter estimates, obtained at previous iterations. In general these values are accurate only when the parameterization is good, thus they are not always reliable estimates of the corrosion depth; otherwise they are reliable indicators of the *regions* where the corrosion exists. Note that the accuracy of this localization is disturbed by the strong diffusive character of the heat conduction process.

It is assumed to use as initial point the null profile over a chosen coarse subdivision of S, $\left\{x_1^0, \ldots, x_{n_\theta^0}^0\right\}$, $x_1^0 = 0$, $x_{n_\theta^0}^0 = 1$. Observe that this assumption is motivated by the physical problem: first of all it is important to understand if the material is uncorroded. If not, it is meaningful to adopt a proper research strategy.

8.4.4.1 Inner-Outer loop algorithm

The inner-outer loop strategy is the simplest one and it is sketch in algorithm 4. Given two estimates of $\boldsymbol{\theta}^*$, $\hat{\boldsymbol{\theta}}^{l-1} \in \mathbb{R}^{n_\theta^{l-1}}$ and $\hat{\boldsymbol{\theta}}^{l} \in \mathbb{R}^{n_\theta^{l-1}}$, if there exists at least one $j \in [1, n_\theta^l - 1]$ such that $\theta^l(j) > \theta^{l-1}(j)$, Θ is defined as the set of all indices j satisfying this property. Otherwise, Θ is the set of j such that $\theta^l(j) > 0$. The new iteration $\hat{\boldsymbol{\theta}}^{l+1}$ is obtained bisecting every segment $[x_j^l, x_{j+1}^l]$ of the l-th subdivision of S, such that $j \in \Theta$. Then, in the inner loop, the projected damped Gauss-Newton method is applied with respect to the refined subdivision. This method in general converges, but it is slow, due to the computational cost of two nested loops. In fact while the outer loop adapts the parametrization, the inner one estimates model parameters' values for the current refinement level of $[0, 1]$. Moreover this strategy tends to over-refine S.

Algorithm 4 Outer Loop:

1: Fix a uniform step in $[0, 1]$, $\Delta^0 x$. Consider the coarse subdivision $\left\{x_1^0, \ldots, x_{n_\theta^0}^0\right\}$, $x_1^0 = 0$, $x_i^0 = (i-1)\Delta^0 x$, $x_{n_\theta^0}^0 = 1$, $h_c^0(i) = \Delta^0 x$, $i = 1, \ldots, n_\theta^0 - 1$, $l = 0$;
2: $\hat{\boldsymbol{\theta}}^0 = \mathbf{0}_{n_\theta^0 - 1} \in \mathbb{R}^{n_\theta^0 - 1}$;
3: **while** $\tilde{J}(\hat{\boldsymbol{\theta}}^l) \approx \sigma$ **do** {σ is the reference value of the cost function obtained for the sound model }
4: $\left\{x_1^{l+1}, \ldots, x_{n_\theta^{l+1}}^{l+1}\right\} = \left\{x_1^l, \ldots, x_{n_\theta^l}^l\right\}$
5: $h_c^{l+1}(i) = h_c^l(i)$, $i = 1, \ldots, n_\theta^l - 1$
6: $\Delta^{l+1} x = \frac{\Delta^l x}{2}$
7: **for all** $\hat{\theta}^l(i) > 0$, $i = 1, \ldots, n_\theta^l - 1$ **do**
8: $n_\theta^{l+1} = n_\theta^{l+1} + 1$
9: $\left\{x_1^{l+1}, \ldots, x_{n_\theta^{l+1}}^{l+1}, \frac{x_{i+1}^{l+1} - x_i^{l+1}}{2}\right\}$, $h_c^{l+1}(i) = h_c^{l+1}(i+1) = \Delta^{l+1} x$, $h_c^{l+1}(i+2 : end) = h_c^l(i+1 : end)$.
10: **end for**
11: solve the INNER LOOP, obtaining $\hat{\boldsymbol{\theta}}^{l+1} \in \mathbb{R}^{n_\theta^{l+1} - 1}$
12: $l = l + 1$;
13: **end while**

8.4.4.2 Formulation of the predictor-corrector algorithm

The high computational cost of the inner-outer loop algorithm has motivated the research of a smarter algorithm: the idea is to reorganize it in a predictor-corrector form

(155). Observe that, since it is assumed to start from the null corrosion profile $\hat{\boldsymbol{\theta}}^0$ it is known that, if the material is corroded, we are underestimating its corrosion profile. The idea is to try to build a sequence of estimated corroded domains, $\left\{\hat{\boldsymbol{\theta}}^l\right\}$, $l \geq 0$, avoiding huge overestimations of $\hat{\boldsymbol{\theta}}^*$. In fact in practice small overestimations are usually allowed and preferred to underestimations. To limit progressive refinements and to obtain a better conditioned matrix ψ, two estimators are used: the $\mathcal{L}^2$ norm and the mean of the prediction error $\mathbf{e}_\theta$ respectively. While the norm is a measure of the distance between $\hat{\boldsymbol{\theta}}^l$ and $\hat{\boldsymbol{\theta}}^*$, the mean permits to understand if $\hat{\boldsymbol{\theta}}^l$ is a big overestimate of the profile. In fact, using Lemma 8.4.1, it is known that a local big overestimate corresponds to local negative values of the prediction error, whose absolute values are big too. More precisely, the predictor step works as follows: given $\Lambda = \emptyset$, which represents the set of parameters to be estimated in the corrector step, given a fixed scalar perturbation $\delta > 0$ and two suitable thresholds $\alpha_\eta,\ \alpha_\nu > 0$, substitute the outer loop with the *linear predictor step*. Given $\hat{\boldsymbol{\theta}}^l \in \mathbb{R}^{n_\theta^l - 1}$ and the l-th subdivision of S, $\left\{x_1^l, \ldots, x_{n_\theta^l}^l\right\}$, for every $i \in [1, n_\theta^l - 1]$, consider the perturbed parameter

$$\hat{\boldsymbol{\theta}}_{\delta,i}^l := \begin{cases} \hat{\theta}^l(k) + \delta, & if\ k = i \\ \hat{\theta}^l(k), & elsewhere \end{cases} \tag{8.11}$$

and compute the corresponding prediction error $\mathbf{E}_{\hat{\theta}_{\delta,i}^l} \in \mathbb{R}^{n_\theta^l \times N}$. Then for all $j \in [1, n_\theta^l]$, $\eta_\delta^l(i,j)$ and $\nu_\delta^l(i,j)$ are computed, the norm and the temporal mean of $\mathbf{E}_{\hat{\theta}_{\delta,i}^l}(j,:)$ respectively.

Given $\eta_\delta^l,\ \nu_\delta^l \in \mathbb{R}^{n_\theta^l - 1 \times n_\theta^l}$, the algorithm proceeds as follows: initialize $\left\{x_1^{l+1}, \ldots, x_{n_\theta^{l+1}}^{l+1}\right\} = \left\{x_1^l, \ldots, x_{n_\theta^l}^l\right\}$. Given $I := [1, n_\theta^l - 1]$, for all $i \in I$

- if the perturbation $\hat{\boldsymbol{\theta}}_{\delta,i}^l$ improves significatively the cost function, or equivalently if $\eta_\delta^l(i,i)$ and $\eta_\delta^l(i,i+1)$ are both less than α_η and it does not correspond to a big overestimate, or likewise if $\nu_\delta^l(i,i)$ and $\nu_\delta^l(i,i+1)$ are both greater than $-\alpha_\nu$, then $\Lambda = \Lambda \cup \{i\}$;

- otherwise, if there is a small improvement in the cost function in at least one node using $\hat{\boldsymbol{\theta}}_{\delta,i}^l$, or equivalently the minimum of $\eta_\delta^l(i,:)$ is less than α_η and it does not correspond to a big overestimate, or likewise if $\nu_\delta^l(i,i)$ and $\nu_\delta^l(i,i+1)$ are both greater than $-\alpha_\nu$, or if there is a change of sign between $\nu_\delta^l(i,i)$ and $\nu_\delta^l(i,i+1)$, bisect the segment $[x_i^{l+1}, x_{i+1}^{l+1}]$: $\left\{x_1^{l+1}, \ldots, x_{n_\theta^{l+1}}^{l+1}\right\} \cup \frac{x_i^{l+1} + x_{i+1}^{l+1}}{2}$. Consider $I = I + 1$

and compute $\hat{\boldsymbol{\theta}}^l_{\delta,s}$, where s represents the indexes of parameters corresponding to the new two subsegments. Then η^l_δ and ν^l_δ are updated considering also those values.

Observe that only $n_\theta - 1$ matrix-vector products are needed in this phase. The choice of the thresholds α_η and α_ν characterized the parametrization: in fact if α_η is chosen too big the algorithm will refine the parameterization less than necessary, whether if it is too small the algorithm will over-refine the parameterization. Instead α_ν is a limit for the allowed overestimation permitted. The optimal choice depends obviously on the specific application, but its tuning is not a problem. Instead, a general auto-tuning strategy is not easy to formulate.

In the corrector step, the projected damped Gauss-Newton method (algorithm 5) is applied only to those parameters whose indexes belongs to Λ. Observe that this strategy reduce the ill-conditioning of ψ_θ, since we choose not to optimize parameters which do not improve the value of the cost function, or equivalently parameters whose perturbations do not change significatively the predicted temperatures.

The detailed description of the predictor-corrector is given in algorithm 5.

Remark 8.4.1 *Observe that to obtain more reliable estimates, in the predictor step it is better to consider two different perturbation parameters δ_1, δ_2, and then to consider for every node the minimum value between $\eta^l_{\delta_1}$ and $\eta^l_{\delta_2}$ and between $\nu^l_{\delta_1}$ and $\nu^l_{\delta_2}$ respectively.*

8.5 Numerical results

In this section some numerical experiments are described, to validate the algorithms presented. In particular in (8.1) and (8.2) the following values of constants are used: $t_f = 1.51\ s$, $L = 0.1\ m$; $\rho C = 3.2 \cdot 10^6\ \frac{J}{m^3\,^\circ C}$, $k = 3.77 \cdot 10^3\ \frac{W}{m^\circ C}$, and

$$q(t) = \frac{Wt}{\sigma_q^2} e^{-\frac{\sqrt{t}}{\sigma_q}},\ t \in (0, t_f] \tag{8.12}$$

where $\sigma_q = 0.0106$, $W = 2.9511 \cdot 10^{17}\ J$. The initial condition is set to $T_0(\cdot) = 20^\circ C$. In this section the backward Euler method is adopted for the time discretization, using a temporal step $\Delta t = 0.0005$ in $(0, 0.1]$ and $\Delta t = 0.05$ in $(0.1, t_f]$. A $P1$-FE method is used for space discretization, on a variable mesh, whose step length along y is $h_y = 0.01\ m$ or $h_y = 0.005\ m$, and a variable step along x, depending on the adaptive parametrization. The sensors are supposed to be $n_y = 11$ or $n_y = 21$, distributed with

8. INVERSE HEAT CONDUCTION PROBLEM

Algorithm 5 Predictor-Corrector algorithm:

Fix a uniform step in $[0,1]$, $\Delta^0 x$. Consider the coarse subdivision $\left\{x_1^0,\dots,x_{n_\theta^0}^0\right\}$, $x_1^0=0$, $x_i^0=(i-1)\Delta^0 x$, $x_{n_\theta^0}^0=1$, $h_c^0(i)=\Delta^0 x$, $i=1,\dots,n_\theta^0-1$;
fix α_η,α_ν, $l=0$, a small parameter perturbation δ;
$\hat{\boldsymbol{\theta}}^0=\mathbf{0}_{n_\theta^0-1}\in\mathbb{R}^{n_\theta^0-1}$;
while $\tilde{J}(\hat{\boldsymbol{\theta}}^l)\approx\sigma$ **do** {% σ is the reference value of the cost function obtained for the sound model }
  $\left\{x_1^{l+1},\dots,x_{n_\theta^{l+1}}^{l+1}\right\}=\left\{x_1^l,\dots,x_{n_\theta^l}^l\right\}$, $h_c^{l+1}(i)=h_c^l(i)$, $i=1,\dots,n_\theta^l-1$
  $\Delta^{l+1}x=\frac{\Delta^l x}{2}$, $n_\theta^{l+1}=n_\theta^l$, $I=n_\theta^{l+1}-1$
  $\Lambda=\emptyset$, set of indexes of parameters to be optimized
  for all $i\in[1,I]$ **do**
    compute $\hat{\boldsymbol{\theta}}_{\delta,i}^l$, $\eta_\delta^l(i,:)$ and $\nu_\delta^l(i,:)$
  end for
  for all $i\in[1,I]$ **do**
    if $\max\left\{\eta_\delta^l(i,i),\eta_\delta^l(i,i+1)\right\}<\alpha_\eta$ and $\min\left\{\nu_\delta^l(i,i),\nu_\delta^l(i,i+1)\right\}>-\alpha_\nu$ **then** {substantial decrease of the cost function, without overestimating }
      $\Lambda=\Lambda\cup\{i\}$ % this is a parameter to be optimized
      $i=i+1$;
    end if
    if $\min_j\left\{\eta_\delta^l(i,j)\right\}<\alpha_\eta$ and $\min\left\{\nu_\delta^l(i,i),\nu_\delta^l(i,i+1)\right\}>-\alpha_\nu$, or $\nu_\delta^l(i,i)\cdot\nu_\delta^l(i,i+1)<0$ **then** { moderate decrease of the cost function, without overestimating or change of sign in ν }
      $n_\theta^{l+1}=n_\theta^{l+1}+1$, $I=I+1$; % bisect the corresponding segment
      $\left\{x_1^{l+1},\dots,x_{n_\theta^{l+1}}^{l+1}\right\}\cup\frac{x_{i+1}^{l+1}-x_i^{l+1}}{2}$,
      $h_c^{l+1}(i)=h_c^{l+1}(i+1)=\Delta^{l+1}x$, $h_c^{l+1}(i+2:end)=h_c^l(i+1:end)$,
      $\eta_\delta^l(i+2:end+1,:)=\eta_\delta^l(i+1:end,:)$, $\nu_\delta^l(i+2:end+1,:)=\nu_\delta^l(i+1:end,:)$.
      for all $i\in[i,i+1]$ **do**
        compute $\hat{\boldsymbol{\theta}}_{\delta,i}^l$, $\eta_\delta^l(i,:)$ and $\nu_\delta^l(i,:)$
      end for
    end if
  end for
  given the subdivision $\left\{x_1^{l+1},\dots,x_{n_\theta^{l+1}}^{l+1}\right\}$
  apply the projected damped Gauss-Newton method, optimizing **only** parameters whose indexes belong to Λ obtaining $\hat{\boldsymbol{\theta}}^{l+1}\in\mathbb{R}^{n_\theta^{l+1}-1}$
  $l=l+1$;
end while

uniform distance $h_x = 0.1\ m$ or $h_x = 0.05\ m$ respectively. Numerical experiments are carried out using MATLAB.

In all the examples presented, experimental temperatures are simulated numerically. The first step is to validate the numerical model: dealing with pure experimental data, this step is fundamental also to decide the optimal values of the coefficients of the model. In our simulated context, it is still important to estimate the reference minimal value of the cost function σ. To obtain a significative threshold σ, the validation is done using the initial coarse grid used in the estimation of the corroded one. Thus the predictor-corrector strategy reveals uncorroded domains, comparing their cost functions with σ.

In this section the predictor step is applied considering two distinct perturbation parameters, $\delta_1 = 2h_y$ and $\delta_2 = 0.03$ (cfr. Remark 8.4.1), while the inner-outer strategy builds the sensitivity matrix using a perturbation $\delta = 0.02$.

In Figure 8.11 the real corroded profiles (left) are compared to the inner-outer (center) and predictor-corrector (right) estimates respectively. As can be seen the predictor strategy tends to refine less and it is also less computationally expensive, due to its linear predictor step. It is important to note that it is formulated such that small overestimates are preferred to underestimates: as a consequence usually the estimated corroded domain is contained in the optimal one, but the distance is small enough. In contrast, although inner-outer algorithm is a simpler strategy, it is more expensive and also tends to over-refine the profile, bisecting also segments which corresponds to null corrosion in the real profile.

In Figure 8.9, given the real corrosion profile represented in the up-left picture, some iterations of the predictor-corrector algorithm are collected. The algorithm refines properly the segment S: thus ψ_θ in the Newton method is computed only for a small subset of parameters, improving its ill-conditioning. Observe that predictor-corrector overestimates only just outside the corrosion front, due to the diffusive nature of the underlying heat equation. In Figure 8.10 both the $\mathcal{L}^2$-norm of the error (left) and the $O(1)$ estimate of the order of convergence are presented. Thus the adaptive refinement strategy, although it diminishes the computational cost, it causes a slow down in the convergence of a Newton-type algorithm, which is usually $O(2)$, starting near enough to the optimum. Finally observe that, in contrast to the inner-outer algorithm, the predictor-corrector convergence is approximately monotonic, since the refinement of S is entirely done before the local optimization of parameters.

Figure 8.12 (left) shows a real profile difficult to estimate, due to the presence of two deep corrosion fronts, close to each other. The predictor-corrector strategy

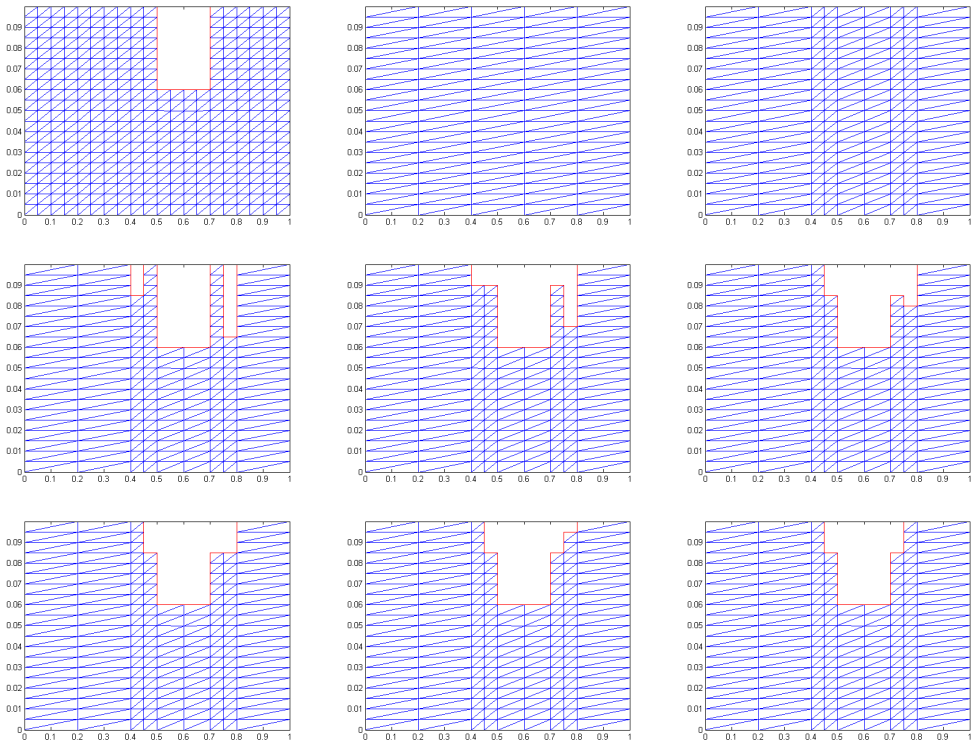

Figure 8.9: *Real profile (first row left) and some iterations of the predictor-corrector method.*

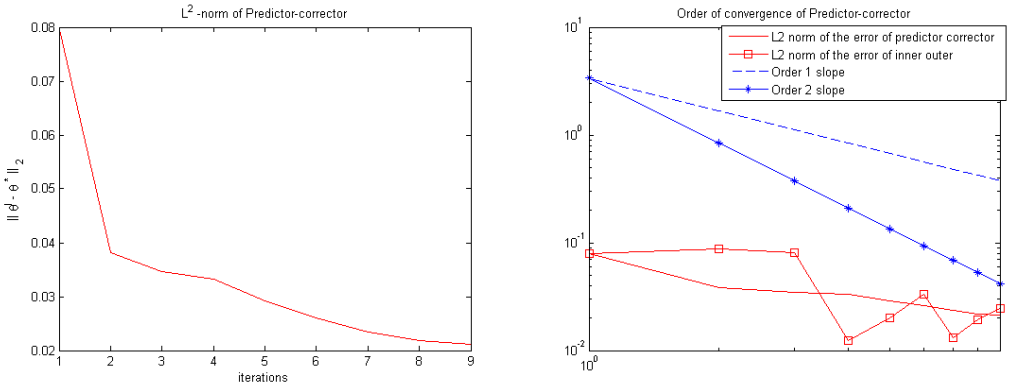

Figure 8.10: $\mathcal{L}^2$*-norm of the error and estimates of the order of convergence.*

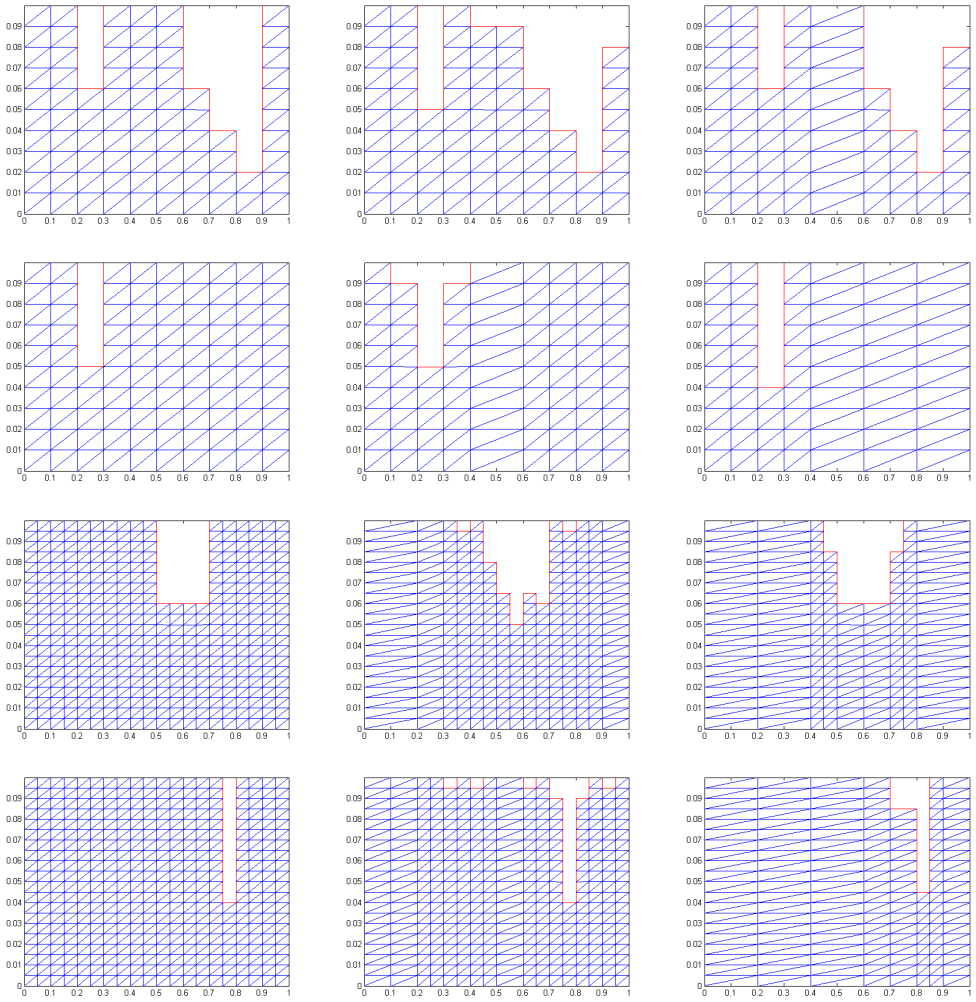

Figure 8.11: *Real corroded profiles (left), inner-outer (center) and predictor-corrector (right) estimates.*

converges to a local minimum (Figure 8.12 (right)). In fact, as mentioned in section 8.4.4, the adaptive strategy could introduce local minima in the problem. Observe that inner-outer algorithm could be more robust (Figure 8.12, center), although it is more computationally expensive. However the estimated predictor-corrector's minimum is a satisfying one, because it reveals both the local position of corrosion and its shape.

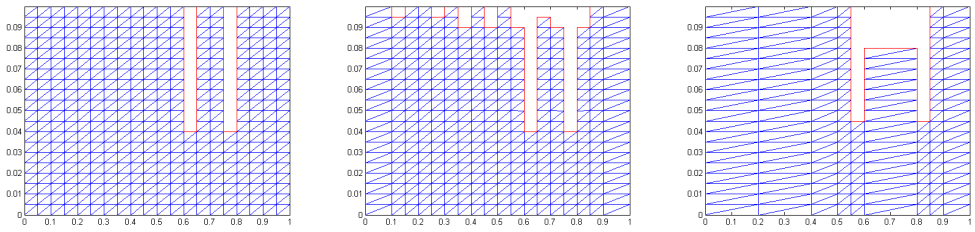

Figure 8.12: *Real profile (left), inner-outer (center) and predictor-corrector (right) estimates.*

9

Inverse convection problem

9.1 Introduction

Inverse heat or mass convection problems, classically deal with the estimation of wall heat flux densities or intensities of source terms (143, 160, 161, 163). As presented in chapter 7, inverse problems are usually mathematically ill-posed and regularization methods have been developed to ensure stable solutions (149, 151, 172). Classical methods are penalization such as Tikhonov's regularization (169), or Bayesian methods using prior information (129), iterative regularization (128) and regularization using singular value decomposition followed by truncation of the singular values spectrum (167).

This chapter follows (139): we are interested in solving an inverse convection problem, whose direct model coincides with a parabolic convection diffusion reaction equation on a fixed domain. To deal with its ill-posedness we adopt a regularization algorithm based upon Truncated Singular Value Decomposition (TSVD) and diagonal scaling (162); moreover an adaptive parametrization with time localization is formulated.

Convection-diffusion-reaction equation can be used to model a variety of physical problems. For example in (146), this equation is used to predict water quality in rivers, by measuring the quantity of organic matter contained. The importance of these pollution is estimated by the measures of the so-called BOD (Biologic Oxygen Demand) and COD (Chemical Oxygen Demand). In (146) the problem of identifying the location and the magnitude (intensity) of pollution point sources from the measurements of BOD on a part of the river is considered: the problem of source term identification is solved using an algorithm based on the minimization of a cost function of Kohn and Vogelius type. Also in (165) water pollution is considered: knowing the *origin* of the source of contamination is probably the most important aspect when attempting to understand, and therefore to control, the pollution transport process. Thus, a challenging issue in environmental problems is the *identification of sources* of pollution in waters. (165) deals with source identification problem, using Boundary Element Method (BEM). In (142), the same problem of source estimation is considered to estimate the time-varying emission rates of pollutant sources in a ventilated enclosure, assuming that the velocity field is stationary: in fact in the frame of occupational risk prevention, the knowledge of both space and time distributions of contaminant concentration is a crucial issue to evaluate the workers exposure. Although air pollution is considered, instead of

water, the underlying model is still a convection-diffusion-reaction equation, with a different convective velocity field. In (142) source's location is supposed to be known. Possible applications of this study are concerned with cartography of pollutants in buildings, estimation of contaminant emission rates inside manufactures, leak detection, environment and process control through 'intelligent sensors' (controlled ventilation with closed-loop function of pollution threshold). A similar problem is considered in (133). Finally in (137), a convection inverse problem is solved to determine an estimate of the source term as a function of the altitude and the temporal of iodine-131, caesium-134 and caesium-137 in the Chernobyl disaster.

In general, in inverse convection problems, either distributed control (142, 165), or boundary control (173) or both (154) are considered. In the present chapter we are interested in estimating location and intensity of pollution, and we assume to deal with boundary control, i.e. we suppose that the sources are located along domain's boundary. Thus, as in (173), we deal with an inverse problem in which one is looking for the unknown conditions in a part of the boundary, while overspecified boundary conditions are supplied in another part of the boundary (here the outflow region). As mentioned above, this type of problem can model both water and air pollution.

As mentioned e.g. in (144), in *inverse problems* or *optimal control* or *optimization settings*, one is faced with the need to do multiple state solves during an iterative process that determines the optimal solution. If one approximates the state in the reduced, k-dimensional space and if k is small, then the cost of each iteration of the optimizer would be very small with respect to that using full, high-fidelity state approximations. Thus *Proper Orthogonal Decomposition (POD)* will be studied in this paper as a model reduction technique, to bring our study closer to a real time problem.

In section 9.2 the direct problem is described, introducing also POD reduction. In sections 9.3 and 9.4.2 the continuous and the discrete inverse problems are formulated, respectively. Section 9.5 deals with the problem of known source location, while in section 9.6 also source position is estimated. Finally in section 9.7 the importance of stabilizing the problem is underlined.

9.2 Description of the direct problem

Let $[0, t_f) \subset \mathbb{R}$ and Ω be an open, limited and Lipschitz continuous boundary subset $\Omega \subset \mathbb{R}^2$, sufficiently regular. We denote with $\partial\Omega$ the boundary of Ω. Let $c : [0, t_f) \times \Omega \to \mathbb{R}$, $c = c(t, \mathbf{x})$ be the solution of the following (direct) parabolic convection-diffusion-

reaction equation:

$$\begin{cases} \frac{\partial c}{\partial t} - \mu\Delta c + \nabla\cdot(\mathbf{u}c) + \sigma c &= 0, & in \quad (0,t_f)\times\Omega \\ c &= c_0, & on \quad \{0\}\times\Omega \\ c &= c_{in}, & on \quad (0,t_f)\times\Gamma_{in} \\ c &= c_{up}, & on \quad (0,t_f)\times\Gamma_{up} \\ \mu\frac{\partial c}{\partial n} &= 0, & on \quad (0,t_f)\times\Gamma_{down} \\ c &= 0, & on \quad (0,t_f)\times\Gamma_{r} \end{cases} \tag{9.1}$$

where Γ_{in}, Γ_{up}, Γ_{down} and Γ_r are given disjoint sets such that $\partial\Omega = \Gamma_{in}\cup\Gamma_{up}\cup\Gamma_{down}\cup\Gamma_r$.

Suppose that $c_{in} \in H^{\frac{1}{2}}(\Gamma_{in})$, $c_{up} \in H^{\frac{1}{2}}(\Gamma_{up})$, the initial condition $c_0 \in L^2(\Omega)$ and the coefficients are independent on time, moreover $\mu \in L^\infty(\Omega)$, $\mu(\mathbf{x}) \geq \mu_0 > 0$ for all $\mathbf{x} \in \Omega$, $\sigma \in L^\infty(\Omega)$, $\sigma(x) \geq 0$ a.e. in Ω, $\mathbf{u} \in [L^\infty(\Omega)]^2$, $div(\mathbf{u}) \in L^2(\Omega)$ are known. The *direct problem* consists in finding the concentration c over Ω at time t_f. A model example for the stationary problem is given e.g. in (140).

As in (142), we assume that the physical properties of the fluid are constant and that the transported contaminant is considered as a passive scalar, which means that it does not affect the velocity field. Thus we suppose to know $\mathbf{u}$.

An example of the 2D domain Ω is illustrated in figure 9.1.

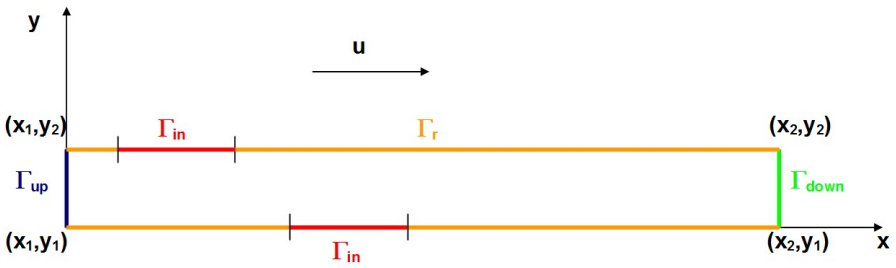

Figure 9.1: *Example of problem's domain* Ω.

9.2.1 Wellposedness of the direct problem and finite element discretization

As analyzed in chapter 9, let $H^1_{\Gamma_r\cup\Gamma_{up}\cup\Gamma_{in}}(\Omega)$ be the set of $v \in H^1(\Omega)$ such that $v_{|_{\Gamma_r\cup\Gamma_{up}\cup\Gamma_{in}}} = 0$. Given $V \subset H^1_{\Gamma_r\cup\Gamma_{up}\cup\Gamma_{in}}(\Omega)$, the weak formulation of (9.1) consists in finding $c \in L^2(0,t_f;H^1(\Omega)) \cap \mathcal{C}^0([0,t_f);L^2(\Omega))$ such that

$$\begin{aligned} \tfrac{d}{dt}(c(t),v) + a(\mathbf{u}(t);c(t),v) &= 0, \quad \forall v \in V, \\ c(0) &= c_0, \quad in\ \Omega, \end{aligned} \tag{9.2}$$

where $a(\mathbf{u};\cdot,\cdot)$ is a bilinear form defined as

$$a(\mathbf{u};w,v) := \int_\Omega k\nabla w\nabla v d\omega + \int_\Omega \mathbf{u}\cdot\nabla w v d\omega + \int_\Omega \sigma w v d\omega.$$

Consider now two families of subspaces $\{W_h,\ h>0\}$ and $\{V_h,\ h>0\}$ of $H^1(\Omega)$ and V respectively, and let $c_{0,h}\in W_h$ be a suitable approximation of c_0. Then the Finite Element (FE) discretization of (9.2) consists in finding $c_h\in W_h$ such that

$$\begin{aligned} \tfrac{d}{dt}(c_h(t),v_h) + a(\mathbf{u}(t);c_h(t),v_h) &= 0, \qquad \forall v_h\in V_h, \\ c_h(0) &= c_{0,h}, \quad in\ \Omega. \end{aligned} \tag{9.3}$$

Given a basis of W_h, $\{\phi_i\}$, $i=1,\dots,N_h$, where N_h denotes the number of nodes in Ω, the FE discretization is equivalent to the solution of the following system of ODE's:

$$\begin{aligned} M\dot{\mathbf{C}}(t) + A(\mathbf{u}(t))\mathbf{C}(t) &= \mathbf{F}(c_{in}), \\ \mathbf{C}(0) &= \mathbf{C}_0. \end{aligned} \tag{9.4}$$

where $M_{ij}=(\phi_i,\phi_j)$, $A(\mathbf{u})_{ij}=a(\mathbf{u};\phi_i,\phi_j)$ and $\mathbf{F}(c_{in})$ involves boundary conditions, in particular c_{in}.

Given a time step Δt, consider a uniform subdivision of $[0,t_f)$ $\{t_j\}$, $j=0,\dots,N-1$ such that $(N-1)\Delta t = t_f$. Discretizing (9.4) in time, using e.g. the backward euler method, we obtain

$$\begin{aligned} \left(M+\Delta t A(\mathbf{u}(k+1))\right)\mathbf{C}(k+1) &= M\mathbf{C}(k) + \Delta t\mathbf{F}(c_{in}), \\ \mathbf{C}(0) &= \mathbf{C}_0. \end{aligned} \tag{9.5}$$

9.2.2 Proper Orthogonal Decomposition (POD) reduction

To obtain a faster solution algorithm, a reduction technique can be used. As described in chapter 5.1, a complete overview of all classical methods can be found e.g. in (127, 168). Since system matrices in (9.4) vary with iterations, techniques largely used for linear constant matrices problems, like e.g. Balanced Truncation (BT), becomes too costly to be used. Thus we choose to adopt the Proper Orthogonal Decomposition (POD) method, presented in chapter 6: although its basis is strictly related to local dynamics, it is less costly to compute.

As described in chapter 6, given a time step $\Delta\tau>0$ (which could be different from Δt), consider $t_m\in(0,t_f)$ and $\bar{N}$ such that $\bar{N}\Delta\tau = t_m$: first the unreduced model (9.4) is solved in $[0,t_m]$, collecting the snapshots $\mathcal{X}=(\mathbf{C}_j)$, where $\mathbf{C}_j\in\mathbb{R}^{N_h}$ is the nodal vector of the FE discretization at time $t_j=j\Delta\tau$, $j=0,\dots,\bar{N}$. After computing the

Singular Value Decomposition (SVD) of $\mathcal{X}$, $\mathcal{X} = USV^t$, a suitable threshold k is chosen. A largely used strategy is to choose k such that

$$\frac{\sum_{i=1}^{k} S(i,i)^2}{\sum_{i=1}^{min(N_h,\bar{N})} S(i,i)^2}$$

is greater than a fixed tolerance. Another possibility is to impose that the first k singular values are greater than a fixed tolerance $\tau_\sigma > 0$.

Finally (9.4) is projected on the space generated by the first k POD basis vectors, i.e. we solve the reduced system

$$\begin{aligned} U_k^t M U_k \dot{\mathbf{a}}(t) + U_k^t A(\mathbf{u}) U_k \mathbf{a}(t) &= U_k^t \mathbf{F}(c_{in}), \\ \mathbf{a}(0) &= U_k^t \mathbf{C}_0. \end{aligned} \tag{9.6}$$

in (t_m, t_f), where $U_k := U(:, 1 : k)$, i.e. the system is projected on the subspace generated by the first k columns of U. We denote with $\tilde{\boldsymbol{C}}(t) := U_k \mathbf{a}(t)$ the estimate of $\mathbf{C}(t)$, $t \in (t_m, t_f)$ computed using POD.

9.3 Inverse problem formulation

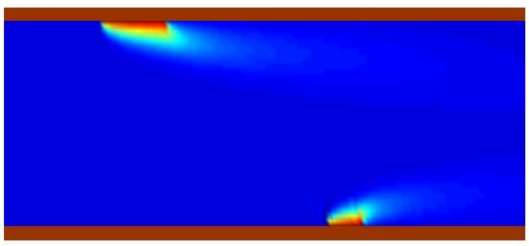

Figure 9.2: *Given the concentration C_s on Γ_{down}, localize Γ_{in} and quantify the concentration released in C_{in}.*

We are interested in solving the following inverse problem (cfr. figure 9.2): given the additional *a priori* information

$$c = c_s, \qquad on \quad [0, t_f) \times \Gamma_{down}, \tag{9.7}$$

where $c_s \in L^2([0, t_f] \times \Gamma_{down})$ is a known scalar function, determine $c_{in}^* \in H^{\frac{1}{2}}(\Gamma_{in})$ such that

$$c_{in}^* = \arg \min_{c_{in} \in H^{\frac{1}{2}}(\Gamma_{in})} J(c_{in}), \tag{9.8}$$

where the *cost function* is

$$J(c_{in}) := \|c(c_{in};t,\mathbf{x}) - c_s(t,\mathbf{x})\|^2_{L^2([0,t_f]\times\Gamma_{down})} = \int_0^{t_f}\int_{\Gamma_{down}} (c(c_{in};t,\mathbf{x}) - c_s(t,\mathbf{x}))^2 d\gamma dt$$

and we have made explicit the dependence of c, solution of (9.1), on c_{in}: $c(c_{in};t,\mathbf{x}) := c(t,\mathbf{x})$ such that $c(t,\mathbf{x}) = c_{in}(\mathbf{x})$, if $\mathbf{x} \in \Gamma_{in}$.

As mentioned in (173), one may consider c_s to be a desired one. In that case, the present inverse problem is a *design problem* where the boundary flux c_{in} is controlled such that a desired concentration is achieved on the boundary Γ_{down}. c_s can also be considered to represent a continuous approximation of a set of discrete experimental *temperature measurements* obtained at a finite number of locations in the boundary Γ_{down} and at discrete time instances within the interval $[0,t_f)$. In this chapter we refer to this second case. Observe that this class of inverse problems are of significant experimental interest for situations where the direct measurement of the heat flux c_{in} is not possible.

9.4 Solution strategies

As indicated in (173), the main difficulty with the minimization problem (9.8) is the calculation of the gradient of J. Mainly two different approaches could be used: the *first discretize than optimize* or vice versa the *first optimize than discretize*. A solution strategy belonging to the last class is based e.g. upon the formulation of the continuous adjoint problem (cfr. section 9.4.1). In this chapter we mainly focus on the first strategy: in particular we adopt a discrete approximation of $J'(c_{in})$, combined with a Gauss-Newton approach, as explained starting from section 9.4.2.

9.4.1 First optimize than discretize strategy: main ideas

In this section we will present the main ideas of the *first optimize than discretize* strategy, based upon the formulation of the adjoint model.

As indicated in (173), we introduce the directional derivative

$$\begin{aligned} D_{\Delta c_{in}} J(c_{in}) &:= (J'(c_{in}), \Delta c_{in})_{L^2([0,t_f]\times\Gamma_{in})} \\ &= (c(t,\mathbf{x};c_{in}) - c_s(t,\mathbf{x}), \Theta(t,\mathbf{x};c_{in},\Delta c_{in}))_{L^2([0,t_f]\times\Gamma_{down})} \end{aligned} \tag{9.9}$$

where the *sensitivity* concentration field $\Theta(t,\mathbf{x};c_{in},\Delta c_{in}) := D_{\Delta c_{in}} c(t,\mathbf{x};c_{in})$ is such that

$$c(t,\mathbf{x};c_{in}+\Delta c_{in}) = c(t,\mathbf{x};c_{in}) + \Theta(t,\mathbf{x};c_{in},\Delta c_{in}) + O(\|\Delta c_{in}\|^2_{L^2([0,t_f]\times\Gamma_{down})}).$$

9. INVERSE CONVECTION PROBLEM

Taking the directional derivative of (9.1) in the direction Δc_{in} and calculated at $c(t,\mathbf{x};c_{in})$, gives the following *linear sensitivity natural convection problem*:

$$\left\{\begin{array}{rcll} \frac{\partial\Theta}{\partial t}-\mu\Delta\Theta+\nabla\cdot(\mathbf{u}\Theta)+\sigma\Theta &=& 0, & in \quad \Omega \\ \Theta &=& 0, & on \quad \{0\}\times\partial\Omega \\ \Theta &=& \Delta c_{in}, & on \quad [0,t_f]\times\Gamma_{in} \\ \Theta &=& 0, & on \quad [0,t_f]\times\Gamma_{up} \\ \mu\frac{\partial\Theta}{\partial n} &=& 0, & on \quad [0,t_f]\times\Gamma_{down} \\ \Theta &=& 0, & on \quad [0,t_f]\times\Gamma_{r} \end{array}\right. \tag{9.10}$$

which is useful to define the *adjoint operator* $\mathcal{L}^*$, i.e. an operator such that it satisfies the *Lagrange identity*

$$(\mathcal{L}^*\psi,\Theta)_{L^2([0,t_f]\times\Omega)}=(\psi,\mathcal{L}\Theta)_{L^2([0,t_f]\times\Omega)}\equiv 0,$$

where $\mathcal{L}$ denotes the differential operator associated to (9.1).

$$\begin{array}{rcl} \int_{[0,t_f]\times\Omega}(\frac{\partial\Theta}{\partial t}-\mu\Delta\Theta+\nabla\cdot(\mathbf{u}\Theta)+\sigma\Theta)\psi dtd\mathbf{x} &=& \int_{[0,t_f]\times\Omega}(-\frac{\partial\psi}{\partial t}-\mu\Delta\psi-\mathbf{u}\nabla\psi+\sigma\psi)\Theta dtd\mathbf{x} \\ &+& \psi(t_f)\Theta(t_f) \\ &+& \int_{[0,t_f]\times\partial\Omega}\Theta(\mu\nabla\psi+\psi\mathbf{u})\cdot\mathbf{n}-\mu\psi\nabla\Theta\cdot\mathbf{n}dtd\gamma \end{array}$$

where we have applied the Divergence Theorem and we have used initial and boundary conditions of Θ and $div\mathbf{u}=0$.

Define the adjoint problem in the following manner:

$$\left\{\begin{array}{rcll} -\frac{\partial\psi}{\partial t}-\mu\Delta\psi-\mathbf{u}\cdot\nabla\psi+\sigma\psi &=& 0, & in \quad \Omega \\ \psi &=& 0, & on \quad \{t_f\}\times\partial\Omega \\ \mu\frac{\partial\psi}{\partial n}+\psi\mathbf{u}\cdot\mathbf{n} &=& -(c-c_s), & on \quad [0,t_f]\times\Gamma_{down} \\ \psi &=& 0, & on \quad [0,t_f]\times\Gamma_{up} \\ \psi &=& 0, & on \quad [0,t_f]\times\Gamma_{in} \\ \psi &=& 0, & on \quad [0,t_f]\times\Gamma_{r} \end{array}\right. \tag{9.11}$$

then

$$\begin{array}{l} \int_{[0,t_f]\times\Omega}(\frac{\partial\Theta}{\partial t}-\mu\Delta\Theta+\nabla\cdot(\mathbf{u}\Theta)+\sigma\Theta)\psi dtd\mathbf{x}=\int_{[0,t_f]\times\Omega}(-\frac{\partial\psi}{\partial t}-\mu\Delta\psi-\mathbf{u}\nabla\psi+\sigma\psi)\Theta dtd\mathbf{x} \\ \qquad -\int_{[0,t_f]\times\Gamma_{down}}\Theta(c-c_s)dtd\gamma+\int_{[0,t_f]\times\Gamma_{in}}\mu\frac{\partial\psi}{\partial n}\Delta c_{in}dtd\gamma. \end{array}$$

Observe that

$$\int_{[0,t_f]\times\Gamma_{down}}\Theta(c-c_s)dtd\gamma=\int_{[0,t_f]\times\Gamma_{in}}\mu\frac{\partial\psi}{\partial n}\Delta c_{in}dtd\gamma$$

iff (using (9.9))

$$J^{'}(c_{in}) = \mu \frac{\partial \psi}{\partial n},$$

on $[0, t_f] \times \Gamma_{in}$.

The minimization strategy of the cost functional is presented in (173) and is sketched in algorithm 6.

Algorithm 6 k-th iteration of the algorithm which solves the inverse problem using adjoint variables:

while $c_{in}^{(k)} < tol$, $k \geq 0$ **do**
  given $c_{in}^{(k)}$, solve the direct problem for $c(t, \mathbf{x}; c_{in}^{(k)})$;
  compute the prediction error $c_s(t, \mathbf{x}) - c(t, \mathbf{x}; c_{in}^{(k)})$ on $[0, t_f] \times \Gamma_{down}$;
  solve the adjoint problem backward in time for $\psi(t, \mathbf{x}; c_{in}^{(k)})$;
  set $J^{'}(c_{in}^{(k)}) = \mu \frac{\partial \psi}{\partial n}$, on $[0, t_f] \times \Gamma_{in}$;
  if $k = 0$ **then**
    set $\gamma^{(k)} = 0$
  else
    $\gamma^{(k)} = \frac{(J^{'}(c_{in}^{(k)}), J^{'}(c_{in}^{(k)}) - J^{'}(c_{in}^{(k-1)}))_{L^2([0,t_f]\times\Gamma_{in})}}{\left\| J^{'}(c_{in}^{(k-1)}) \right\|_{L^2([0,t_f]\times\Gamma_{in})}}$;
  end if
  if $k = 0$ **then**
    define $p^{(k)} = -J^{'}(c_{in}^{(k)})$
  else
    $p^{(k)} = -J^{'}(c_{in}^{(k)}) + \gamma^{(k)} p^{(k-1)}$;
  end if
  to calculate the optimal step $\alpha^{(k)}$ use the bisection method;
  $c_{in}^{(k+1)} = c_{in}^{(k)} + \alpha^{(k)} p^{(k)}$
end while

The finite element discretization of (9.11) is equivalent to the solution of the following system of ODE's:

$$\begin{aligned} -M\dot{\mathbf{p}} + \tilde{A}(\mathbf{u}(t))\mathbf{C} &= \mathbf{G}, \\ \mathbf{p}(t_f) &= \mathbf{0}, \end{aligned} \tag{9.12}$$

where $\tilde{A}(\mathbf{u})_{ij} = \tilde{a}(\mathbf{u}; \phi_i, \phi_j)$, $\tilde{a}(\mathbf{u}; w, v) := \int_\Omega k \nabla w \nabla v d\omega + \int_\Omega \mathbf{u} \cdot \nabla v w d\omega + \int_\Omega \sigma w v d\omega$ and $\mathbf{G}$ is such that $G_i = -\int_{\Gamma_{down}} \phi_i (c - c_s) d\gamma$.

9.4.1.1 POD reduction of the adjoint model

It works similarly to the primal problem: first we solve the unreduced model (9.12) in $[t_f - t_m, t_f]$, collecting the matrix of snapshots $\mathcal{X}_a = (\mathbf{p}_j)$, where $\mathbf{p}_j$ is the nodal vector of the finite element discretization at $t_j = (N - 1 - j)\Delta t$, $j = 0, \ldots, M$. After

computing the SVD of $\mathcal{X}_a$, $\mathcal{X}_a = U_a S_a V_a^t$, and choosing a suitable threshold k_a, we solve the reduced system

$$\begin{aligned} -U_{a,k}^t M U_{a,k}\dot{\mathbf{q}} + U_{a,k}^t \tilde{A}(\mathbf{u}(t)) U_{a,k}\dot{\mathbf{q}} &= U_{a,k}^t \mathbf{G}, \\ \mathbf{q}(t_f) &= \mathbf{0}_{k,a}, \end{aligned} \tag{9.13}$$

where the projection space is $U_{a,k} := U_a(:, 1 : k_a)$. The estimate is $\tilde{\boldsymbol{p}} := U_{a,k}\mathbf{q}$.

9.4.1.2 Numerical results

Consider now $\Omega = [0, 8] \times [0, 1]$, $\Gamma_h = [0, 8] \times \{1\} \cup [0, 8] \times \{0\}$; the velocity field $\mathbf{u}$ is modeled as a Poiseuille flow i.e.

$$\mathbf{u}(x_1, x_2) = \begin{pmatrix} -4\nu x_2^2 + 4\nu x_2 \\ 0 \end{pmatrix}.$$

We assume that $\nu = 50$, $\mu = 0.1$, $\sigma = 0.1$, $c_{up} = 0.1$ and $\Gamma_{in} = [4, 4.5] \times \{1\}$, $\vartheta = 100$. In figure 9.3 the cost function of the first optimize then discretize method is depicted.

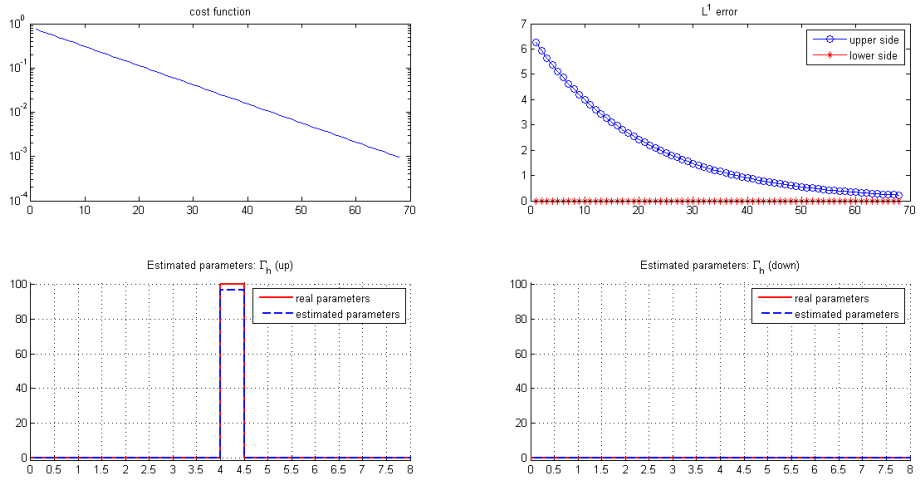

Figure 9.3: *Convergence results applying the adjoint method, supposing that* Γ_{in} *is known.*

The adjoint method is useful mainly if the number of parameters to be estimated in the discretized problem is high. Since we expect that pollution sources are concentrated only in a small part of the boundary, in the following we will suppose that the number of parameters to be estimated when Γ_{in} is known is low. Thus we will adopt a Gauss Newton approach, as explained starting from next section.

9.4.2 First discretize than optimize strategy

Consider the set of time instants

$$\{t_j\}, \quad j = 0, \dots, N-1. \tag{9.14}$$

In the *first discretize than optimize* context, we assume that $c_s(t, \mathbf{x})$ is known only in the n_y nodes of Γ_{down}, for every discrete time t_j. Let $\mathbf{C}_s(t_j) \in \mathbb{R}^{n_y}$ be the vector of measured concentration at $t = t_j$. For simplicity we suppose that

$$\Gamma_{in} = \bigcup_{l=1}^{n_\theta} \Gamma_{in}^{(l)},$$

being $\Gamma_{in}^{(l)}$ disjoint sets, such that c_{in} is constant on each $\Gamma_{in}^{(l)}$, for all $l = 1, \dots, n_\theta$. Thus we have to estimate a vector $\boldsymbol{\vartheta}$ of n_θ non negative parameters: equivalently we assume that the function $c_{in} \in H^{\frac{1}{2}}(\Gamma_{in})$ is a piecewise constant function such that

$$c_{in}(\mathbf{x}) = \vartheta(l), \qquad \mathbf{x} \in \Gamma_{in}^{(l)}.$$

In this context the nodal vector solution is $\mathbf{C}(t_j) = \mathbf{C}(\boldsymbol{\vartheta}; t_j)$, where we have made explicit its dependence on $\boldsymbol{\vartheta}$.

Let $c_{down}(c_{in}; t, \mathbf{x}) := c(c_{in}; t, \mathbf{x})_{|\Gamma_{down}}$, be the *predicted concentration* on Γ_{down}, obtained by solving (9.1) imposing c_{in} on Γ_{in}, and $\boldsymbol{C_{down}}(\boldsymbol{\vartheta}; t_j)$ the corresponding nodal vector, computed at time $t = t_j$. In a space-time discrete setting, (9.8) could be restated as the following *discrete inverse problem*

$$\hat{\boldsymbol{\vartheta}} = \arg \min_{\boldsymbol{\vartheta} \in \mathbb{R}_+^{n_\theta}} \tilde{J}(\boldsymbol{\vartheta}), \tag{9.15}$$

where the *discrete cost function* is defined as

$$\tilde{J}(\boldsymbol{\vartheta}) := \frac{1}{N} \sum_{j=1}^{N} \|\boldsymbol{C_{down}}(\boldsymbol{\vartheta}; t_j) - \mathbf{C}_s(t_j)\|_2^2 . \tag{9.16}$$

Observe that this is a *least squares problem* (cfr. section 7.2.2.1).

9.4.2.1 POD reduction

Using model order reduction techniques to solve (9.8), consists in replacing the cost function (9.16) in (9.15) with the following one

$$\tilde{J}(\boldsymbol{\vartheta}) := \frac{1}{N} \sum_{j=1}^{N} \left\| \tilde{\boldsymbol{C}}_{\boldsymbol{down}}(\boldsymbol{\vartheta}; t_j) - \mathbf{C}_s(t_j) \right\|_2^2 \tag{9.17}$$

where $\tilde{\boldsymbol{C}}$ is the solution of (9.6). An example of application of POD to solve optimal control problems can be found e.g. in (147).

Since the POD basis depends on the collected snapshots, it is necessary to update the projection space as the estimated control C_{in} varies. Let $\bar{n}$ a small positive integer: at every iteration i in this chapter we adopt the following index

$$\mathcal{I}^{(i)} := \frac{1}{\bar{n}} \left\| \sum_{j=1}^{\bar{n}} \tilde{\boldsymbol{C}}(\boldsymbol{\vartheta}^{(i)}; t_j) - \mathbf{C}(\boldsymbol{\vartheta}^{(i)}; t_j) \right\|_2^2 ,$$

i.e. we compare the first iterations of the unreduced system with those obtained projecting on the old POD basis used at iteration $i-1$. Only if $\mathcal{I}^{(i)}$ is greater than a fixed threshold, the i-th basis is updated, computing new snapshots, as described in section 9.2.2. Two strategies can be used (147): old snapshots can be discarded or not. In practice this consists in adding POD modes computed in the $i-1$-th iteration to the new snapshots ensemble: in this case the projection space is more robust to control variations but usually is slightly bigger. For our experimental tests we prefer to discard old snapshots. We obverse that in (147) a new basis is computed at every iteration, without considering an index $\mathcal{I}$.

In the following sections the *first discretize than optimize* strategy is described, starting from the simpler case of known source location, and then extending it to the unknown case.

9.5 Known source location Γ_{in}

As a first step toward the solution strategy, we consider a simpler problem, assuming that the source location Γ_{in} is known.

9.5.1 Solution uniqueness

In this section we demonstrate that if Γ_{in} is known, then the discrete inverse problem admits a unique solution, since there are no local minima. Moreover changes in c_{in} corresponds to changes in the registered concentration.

First of all we prove the following Lemma, which justifies mathematically the physical principle that, as c_{in} increases on Γ_{in}, the concentration on Γ_{down} increases too.

Lemma 9.5.1 *Consider the two problems*

$$\begin{cases} \frac{\partial c_i}{\partial t} - \mu\Delta c_i + \nabla\cdot(\boldsymbol{u}c_i) + \sigma c_i &= 0, & in \quad (0,t_f)\times\Omega \\ c_i &= c_0, & on \quad \{0\}\times\Omega \\ c_i &= c_{in}^{(i)}, & on \quad (0,t_f)\times\Gamma_{in} \\ c_i &= c_{up}, & on \quad (0,t_f)\times\Gamma_{up} \\ \mu\frac{\partial C_i}{\partial n} &= 0, & on \quad (0,t_f)\times\Gamma_{down} \\ c_i &= 0, & on \quad (0,t_f)\times\Gamma_r \end{cases} \tag{9.18}$$

represented in Figure 9.4 (up), where $i = 1, 2$. Suppose that $c_{in}^{(2)}(\boldsymbol{x}) > c_{in}^{(1)}(\boldsymbol{x})$, for every $\boldsymbol{x} \in \Gamma_{in}$. Then $c_2(t,\boldsymbol{x}) > c_1(t,\boldsymbol{x})$ for every $t \in (0,t_f)$ and $\boldsymbol{x} \in \Gamma_{down}$.

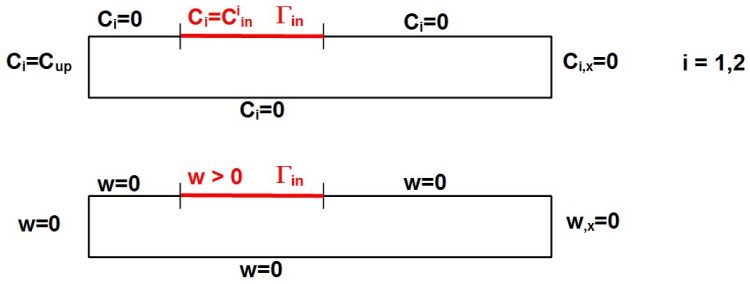

Figure 9.4:

Proof.

Define $w := c_2 - c_1$, which solves

$$\begin{cases} \frac{\partial w}{\partial t} - \mu\Delta w + \nabla\cdot(\mathbf{u}w) + \sigma w &= 0, & in \quad (0,t_f)\times\Omega \\ w &= 0, & on \quad \{0\}\times\Omega \\ w &= c_{in}^{(2)} - c_{in}^{(1)}, & on \quad (0,t_f)\times\Gamma_{in} \\ w &= 0, & on \quad (0,t_f)\times\Gamma_{up} \\ \mu\frac{\partial w}{\partial n} &= 0, & on \quad (0,t_f)\times\Gamma_{down} \\ w &= 0, & on \quad (0,t_f)\times\Gamma_r \end{cases} \tag{9.19}$$

as illustrated in figure 9.4 (down). Observe that w is smooth only inside the domain, but it is not continuous near the boundary, where it admits discontinuities of the first kind: thus generalized solutions must be considered. The strong minimum principle for parabolic operators can be extended for generalized solutions (141, 152): thus we know that the minimum is assumed at the boundary. Moreover, for every open neighbourhood U of Γ_{down}, such that w is regular inside $U \cap \Omega$, $\frac{\partial w}{\partial n} = 0$ on Γ_{down} implies that the

maximum and the minimum of w over $U \cap \Omega$ cannot belong to Γ_{down} (cfr. (141)). As a consequence, $w \geq 0$ in $(0, t_f) \times \Omega$ and, since the minimum is not attained on Γ_{down}, $w = c_2 - c_1 > 0$ on Γ_{down}, for all $t \in (0, t_f)$ i.e. the thesis holds true.

□

The following Proposition is equivalent to prove that there are no local minima.

Proposition 9.5.1 *For every $\bar{\boldsymbol{\vartheta}} \in \mathbb{R}_+^{n_\theta}$, $\bar{\boldsymbol{\vartheta}} \neq \boldsymbol{\vartheta}^*$, there exists at least a sequence of profiles $\{\boldsymbol{\vartheta}\}_n$, $\boldsymbol{\vartheta}_0 = \bar{\boldsymbol{\vartheta}}$, converging in $\mathcal{L}^2(\mathbb{R}^{n_\theta})$ to the real profile $\boldsymbol{\vartheta}^*$, such that $\tilde{J}(\boldsymbol{\vartheta}_n) \downarrow \tilde{J}(\boldsymbol{\vartheta}^*)$.*

Proof. We can construct the sequence $\{\boldsymbol{\vartheta}_n\}_n$ in the following way. For every $k = 1, \ldots, n_\theta$

$$\vartheta_k(j) := \begin{cases} \vartheta_{k-1}(j), & j \neq k \\ \bar{\vartheta}(j) - (\bar{\vartheta}(j) - \vartheta^*(j)), & j = k \end{cases} . \tag{9.20}$$

Thus $\boldsymbol{\vartheta}_{n_\theta} = \boldsymbol{\vartheta}^*$ by construction. Moreover the corresponding sequence of cost functions is *decreasing*: $\tilde{J}(\boldsymbol{\vartheta}_1) > \tilde{J}(\boldsymbol{\vartheta}_2) > \ldots > \tilde{J}(\boldsymbol{\vartheta}^*)$. This fact is a direct consequence of the application of Lemma 9.5.1: suppose that $\vartheta_{k-1}(k) < \vartheta^*(k)$. Then $\vartheta_k(k) > \vartheta_{k-1}(k)$ by construction and thus $\boldsymbol{C_{down}}(\boldsymbol{\vartheta}_k; t)$ will be higher than $\boldsymbol{C_{down}}(\boldsymbol{\vartheta}_{k-1}; t)$ for every $t \in (0, t_f)$ (Lemma 9.5.1) and thus closer to $\boldsymbol{C_{down}}(\boldsymbol{\vartheta}^*; t)$. Analogously if $\vartheta_{k-1}(k) > \vartheta^*(k)$, applying Lemma 9.5.1, $\boldsymbol{C_{down}}(\boldsymbol{\vartheta}_k; t)$ will be lower than $\boldsymbol{C_{down}}(\boldsymbol{\vartheta}_{k-1}; t)$ for all t and thus closer to $\boldsymbol{C_{down}}(\boldsymbol{\vartheta}^*; t)$.

□

9.5.2 Numerical solution strategy

As explained in section 7.2.2.1, starting from an initial guess $\hat{\boldsymbol{\vartheta}}^{(0)}$, line search algorithms find the $k+1$-iteration starting from the k-th one in the following way:

$$\hat{\boldsymbol{\vartheta}}^{(k+1)} = \hat{\boldsymbol{\vartheta}}^{(k)} + \alpha^{(k)} \mathbf{s}^{(k)},$$

where the *damping parameter* $\alpha^{(k)}$ is obtained using a bisection procedure.

Let $\mathcal{R} : \mathbb{R}^{n_y \times N} \to \mathbb{R}^{n_y N}$ be the reshape map such that starting from an $n_y \times N$ matrix $B = [\mathbf{b}_1, \ldots, \mathbf{b}_N]$, it gives $\mathcal{R}(B) := \begin{pmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_N \end{pmatrix}$.

The Gauss-Newton approximation (cfr. section 7.2.2.1 (162)), consists in solving at each iteration

$$\Psi_{\hat{\boldsymbol{\vartheta}}^{(k)}} \mathbf{s}^{(k)} = \mathbf{e}_{\hat{\boldsymbol{\vartheta}}^{(k)}}, \tag{9.21}$$

where the *sensitivity matrix* $\Psi_{\hat{\boldsymbol{\vartheta}}^{(k)}} \in \mathbb{R}^{n_y N \times n_\theta}$ is such that

$$\Psi_{\hat{\boldsymbol{\vartheta}}^{(k)}}(:, i) := \frac{\partial}{\partial \hat{\boldsymbol{\vartheta}}^{(k)}(i)} \mathcal{R}(\boldsymbol{C_{down}}(\hat{\boldsymbol{\vartheta}}^{(k)}; \cdot)), \tag{9.22}$$

for all $i = 1, \ldots, n_\theta$ and the *prediction error* is defined as follows:

$$\mathbf{e}_{\hat{\boldsymbol{\vartheta}}^{(k)}} := \mathcal{R}(\mathbf{C}_s(\cdot)) - \mathcal{R}(\boldsymbol{C_{down}}(\hat{\boldsymbol{\vartheta}}^{(k)}; \cdot)). \tag{9.23}$$

System (9.21) is solved using TSVD.

To compute numerically the sensitivity matrix a finite difference scheme is needed:

$$\Psi_{\hat{\boldsymbol{\vartheta}}^{(k)}}(:, j) \approx \frac{1}{\delta} \left[\mathcal{R}(\boldsymbol{C_{down}}(\hat{\boldsymbol{\vartheta}}^{(k)}(1), \ldots, \hat{\boldsymbol{\vartheta}}^{(k)}(j) + \delta, \ldots, \hat{\boldsymbol{\vartheta}}^{(k)}(n_\theta); \cdot)) - \mathcal{R}(\boldsymbol{C_{down}}(\hat{\boldsymbol{\vartheta}}^{(k)}; \cdot)) \right],$$

where $\delta > 0$ is a small perturbation parameter.

Observe that in general this approximation is computationally expensive, since, it requires the computation of the concentration also for the perturbed input. When Γ_{in} is known, only very few parameters are considered, thus this approximation is effective. The problem becomes more involving when Γ_{in} is unknown, since the number of parameters is higher: in section 9.6 we will explain how the adaptive parametrization and time localization can reduce the computational cost.

If $\delta > 0$ is too small, the finite difference estimate could be inaccurate, since at the numerator we are considering the difference between two quantities which has approximately the same absolute value, and this is divided by a very small denominator, which amplifies the error. A possible solution e.g. is to adopt the Complex-Step Derivative Approximation (159), in which an imaginary increment $i\delta$ is used, approximating

$$\Psi_{\hat{\boldsymbol{\vartheta}}^{(k)}}(:, j) \approx \frac{1}{\delta} Im \left(\mathcal{R}(\boldsymbol{C_{down}}(\hat{\boldsymbol{\vartheta}}^{(k)}(1), \ldots, \hat{\boldsymbol{\vartheta}}^{(k)}(j) + i\delta, \ldots, \hat{\boldsymbol{\vartheta}}^{(k)}(n_\theta); \cdot)) \right).$$

Finally observe that we are assuming that the pollutant is put into the domain, thus

$$C_{in} \geq 0 :$$

as a consequence we need also a *projection* step onto $[0, +\infty)$ of each component of $\hat{\boldsymbol{\vartheta}}^{(k)}$, after its computation.

9.5.3 Numerical results

In this section the projected damped Gauss Newton (PDGN) is compared to other classical solution strategies. Experimental data are simulated numerically, on $\Omega = [0,8] \times [0,1]$, $\Gamma_h = [0,8] \times \{1\} \cup [0,8] \times \{0\}$. Moreover the velocity field $\mathbf{u}$ is modeled as a Poiseuille flow i.e.

$$\mathbf{u}(x_1, x_2) = \begin{pmatrix} -4\nu x_2^2 + 4\nu x_2 \\ 0 \end{pmatrix}.$$

We assume that $\nu = 50$, $\mu = 0.1$, $\sigma = 0.1$ and $c_{up} = 0.1$. Moreover in this section a Gaussian error of variance 0.05 and mean zero is added.

Classical solution strategies cited in this section are well described e.g. in (150). As a regularization parameter, when needed, we use $\alpha = 0.01$, moreover we choose a maximum number of iterations $max_{it} = 20$. Consider the following two examples:

1. $\Gamma_{in} = [4, 4.5] \times \{1\}$, $\vartheta = 100$;

2. $\Gamma_{in} = [4.5, 5] \times \{1\} \cup [1.5, 2] \times \{0\}$, $\boldsymbol{\vartheta} = (100, 80)$;

and see how different techniques approximate them. First of all we consider the example 1. Performances of different methods are depicted in figure 9.5. In the second example, two parameters have to be estimated: results are plotted in figure 9.6.

Observe that in both cases the projected damped Gauss Newton algorithm performs well, converging faster to the optimal solution. It should be noted that, in contrast to Tikhonov and Levenberg-Marquardt it does not need a regularization parameter.

9.5.4 Reduce the order of the system using POD

In this section we analyze the POD reduction introduced in section 9.2.2 on a test case. Consider example 2 introduced in the previous section; in POD reduction two parameters play a central role: t_m, which characterizes the interval $[t_0, t_m]$ when snapshots are collected, and the threshold τ_σ on the singular values of the snapshots matrix. As can be seen in table 9.1, increasing t_m corresponds to a better approximation, since more snapshots are collected. To obtain higher accuracy decreasing t_m, it is necessary to increase τ_σ, considering a higher number of left singular vectors, corresponding to bigger reduced model.

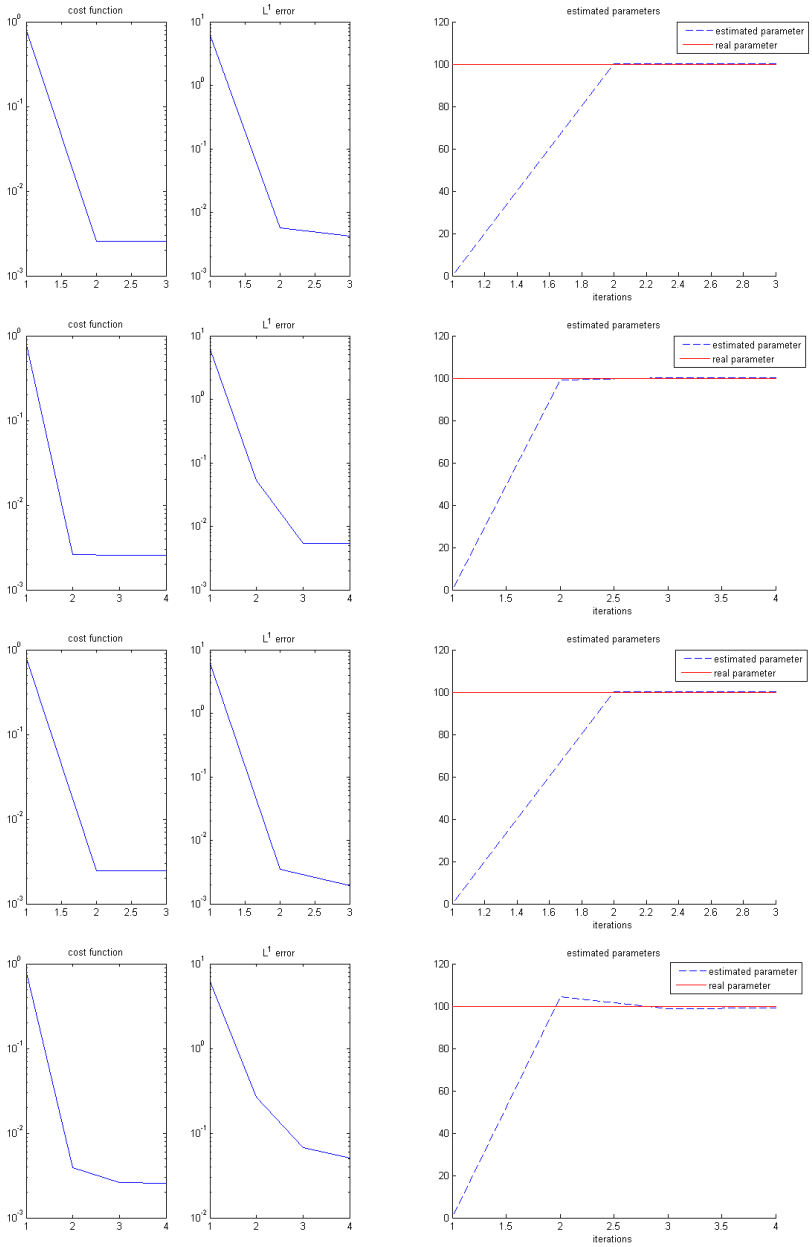

Figure 9.5: *First example: different strategies. Left: cost function and error, right: convergence. First row: projected damped Gauss Newton, second row: Levenberg Marquardt, third row: steepest descent, fourth row: Tikhonov method.*

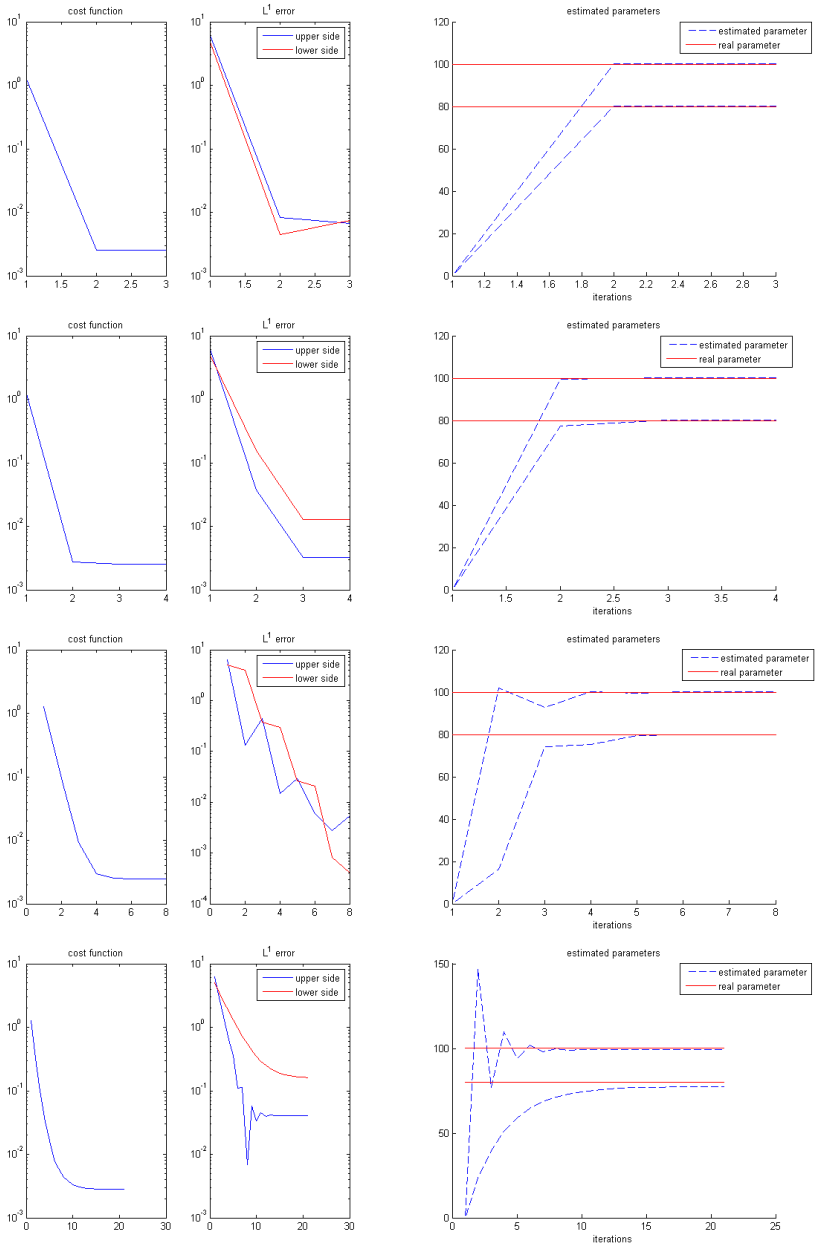

Figure 9.6: *Second example: different strategies. Left: cost function and error, right: convergence. First row: projected damped Gauss Newton, second row: Levenberg Marquardt, third row: steepest descent, fourth row: Tikhonov method.*

		L^1 error: up	L^1 error: down	$\tilde{J}(\boldsymbol{\vartheta})$	Dim. model	num. it.
Unreduced model		0	0	10^{-20}	1071	2
Reduced models: t_m	τ_σ					
2.5	0.01	0.117	4.8	0.113	23	5
2.5	10^{-4}	0.083	1.24	0.089	32	4
3.75	0.01	0.08	0.08	$6 \cdot 10^{-4}$	25	3
3.75	10^{-4}	0.02	0.02	$3 \cdot 10^{-5}$	39	4
5	0.01	0.0015	0.0015	10^{-6}	29	3

Table 9.1: *Example 2 of section 9.5.3, choosing different intervals $[t_0, t_m]$ to collect snapshots and different thresholds τ_σ on singular values of the snapshots matrix.*

It is important to note that the reduction is significative with respect to the unreduced model, which has dimension 1071. However, as described in section 9.2.2, it should be noted that it is necessary to update the POD basis: in all these examples the basis is updated at every new iteration, imposing 0.1 as a threshold on $\mathcal{I}^{(i)}$.

9.5.5 Using Navier Stokes equation: generalization to a time varying velocity field

More generally, problem (9.1) can be completed adding a model for the velocity field $\mathbf{u}$: for an incompressible fluid flow, the incompressible Navier Stokes model can be used:

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} - \nu\Delta\mathbf{u} + (\mathbf{u}\cdot\nabla)(\mathbf{u}) + \nabla p = \mathbf{0}, & in\ (t_0,t_f)\times\Omega, \\ div\mathbf{u} = 0, & in\ (t_0,t_f)\times\Omega, \\ \mathbf{u} = \mathbf{u}_0, & on \quad \{0\}\times\partial\Omega \\ \mathbf{u} = \mathbf{u}_{up}, & on \quad [0,t_f]\times\Gamma_{up} \\ \nu\frac{\partial \mathbf{u}}{\partial n} = 0, & on \quad [0,t_f]\times\Gamma_{down}\cup\Gamma_r\cup\Gamma_{in} \end{cases} \tag{9.24}$$

where p denotes the pressure of the flow field.

In this context, using a reduced order technique is important to limit the computational cost. POD has been adopted to reduced both (9.1), as explained in section 9.2.2 and (9.24), as explained in chapter 6. Consider example 1 described in section 9.5.3: suppose that $\mathbf{u}_{up} = 10$, $Re = 100$, $t_f = 14$ and $t_m = 7$. Navier Stokes reduced system has dimensions $k_1 = 1326$, $k_2 = 1898$, and $k_p = 3501$, respectively for the two components of the velocity and the pressure. The unreduced finite element discretization uses 19521 nodes for each component of the velocity and 4961 for the pressure. The reduced system for (9.1) has dimension 33, instead of 4961, for $\tau_\sigma = 0.01$. Convergence results

for the inverse problem are shown in figure 9.7. The estimated control is 99.9902, thus the error is of order 0.01, and the cost function has order 10^{-9}.

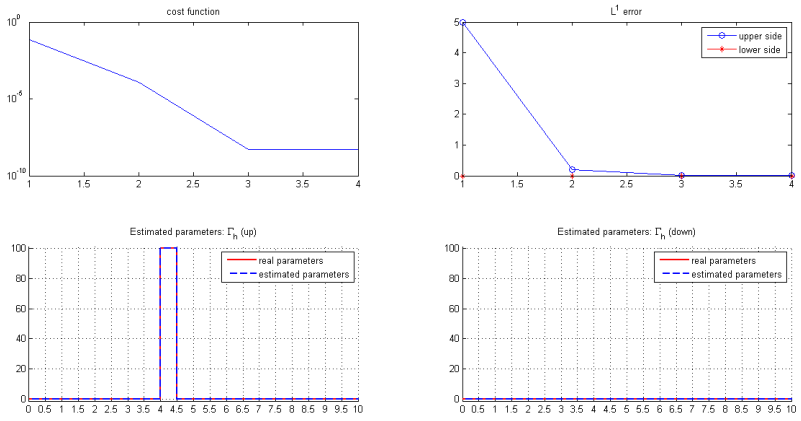

Figure 9.7: *Convergence results for example 1 of section 9.6.4, supposing that* Γ_{in} *is known.*

In the following we will model the velocity field as a Poiseuille flow, since it is more convenient from a computational point of view. However all presented strategies can be applied also to more general velocity fields, e.g. numerical solutions of Navier Stokes equations.

A more involving problem is considered in section 9.6, where it is assumed that also the source location Γ_{in} is unknown. In general in that case projected damped Gauss Newton could not be sufficient and it is too costly, thus it is necessary to adopt a suitable solution strategy based upon an adaptive parametrization and time localization.

9.6 Unknown source location Γ_{in}

Suppose now that the *location* Γ_{in} is unknown.

9.6.1 Introduction: ill-posedness of the problem

To study analytically what happens when Γ_{in} is unknown, we consider a simplified model problem: let $c = c(x)$, $x \in [x_1, x_2] \subset \mathbb{R}$, $x_2 > x_1$ be the solution of the following

one dimensional ODE:

$$\begin{cases} -\mu c''(x) + uc'(x) &= f(x), \qquad in\ (x_1, x_2), \\ c(x_1) &= c_{up}, \\ c'(x_2) &= 0, \end{cases} \tag{9.25}$$

where $f(x) = \begin{cases} M, & |x - x_m| \leq h \\ 0, & elsewhere\ in\ (x_1, x_2) \end{cases}$, $M > 0$, $x_m \in (x_1, x_2)$, $h \in (0, 1)$ s.t. $x_m \pm h \in (x_1, x_2)$. Observe that (9.25) can be viewed as the one dimensional stationary counterpart of (9.1) when $\sigma = 0$ and considering only the x-axis in figure 9.1: the unknown immision boundary Γ_{in} can be represented by an unknown forcing term f, applied in $[x_m - h, x_m + h]$, of intensity M. In this context the inverse problem (9.8) is equivalent to determine the source position (h and x_m) and intensity (M) given the measured concentration $C_s \in \mathbb{R}$ in $x_2 = 1$ (cfr. figure 9.8).

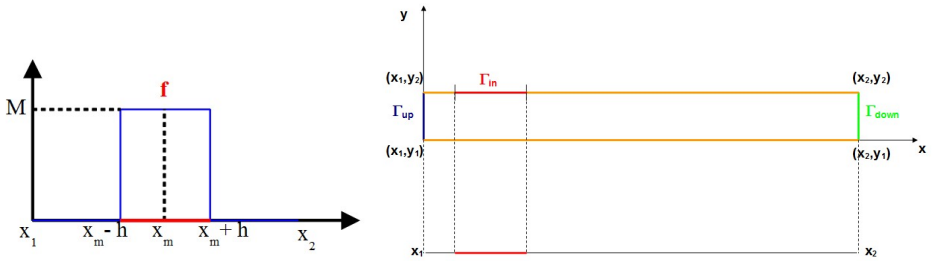

Figure 9.8: *Reduction of the 2D problem to a 1D one with forcing term f.*

The anaytical solution of (9.25) is

$$c(x) = \begin{cases} d_1 + d_2 e^{\frac{u}{\mu}x}, & x < x_m - h, \\ d_3 + \frac{M}{u}x + d_4 e^{\frac{u}{\mu}x}, & |x - x_m| \leq h, \\ d_5 + d_6 e^{\frac{u}{\mu}x}, & x > x_m + h, \end{cases}$$

where $d_1, \ldots, d_6$ are suitable real coefficients obtained imposing boundary conditions and continuity of u and u' in $x_m \pm h$. In particular we are interested in estimating the concentration at the measurement point $x = x_2$. For simplicity we assume that $x_1 = 0$ and $x_2 = 1$. It can be derived that

$$d_6 = 0, \qquad d_5 = \frac{1}{u^2} \exp \frac{-u(x_m + h)}{\mu} \left(2uhM \exp \frac{u(x_m + h)}{\mu} + \mu M \left(1 - \exp \frac{2uh}{\mu} \right) \right).$$

Thus $c(x)$ is constantly equal to d_5 in $[x_m + h, 1]$. We can now study how $c(1)$ depends on M, h, L. We consider $\mu = 0.5$ and $u = 10$ (Peclet number $Pe = \frac{u}{2\mu} = 10$, quantity that

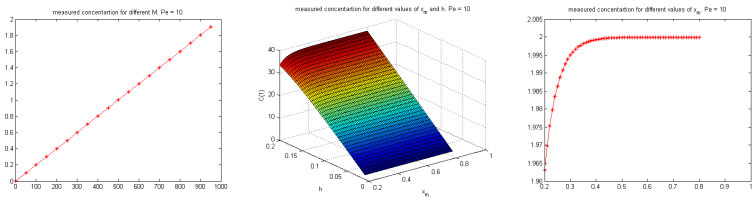

Figure 9.9: *Solution of (9.25) at the measurement point $x_2 = 1$ for different values of M (left), h and x_m (center), x_m (right).*

characterize convection diffusion problems). As can be seen in figure 9.9, varying only M, fixing h and x_m (i.e. knowing the source location), corresponds to a linear striclty increasing $c(1)$ (cfr. figure 9.9 (left)). On the contrary fixing M but varying h and x_m corresponds to the surface plotted in figure 9.9 (center): fixing h for different values of x_m we obtain almost the same $c(1)$ (cfr. figure 9.9 (right)). Thus measuring $c(1)$, *the problem of determining the source is ill-posed in the stationary regime.* Increasing the Peclet number this phenomenum is stressed. Even for this simplified 1D stationary problem, in general unknown source position gives rise to an ill-conditioned problem.

9.6.2 Numerical solution of the discrete inverse problem

The problem consists both in *localizing* Γ_{in} in the horizontal segments $\Gamma_h := \Gamma_r \cup \Gamma_{in}$ and in estimating the *intensity* c_{in}.

9.6.2.1 Algorithm 1: working on the finest subdivision

First of all we consider $\left\{ x_1, \ldots, x_{\frac{n_\theta^f}{2}+1} \right\}$ a *reference uniform finest subdivision* of Γ_h of step length Δx, which represents the minimum width of estimated source emissions. The simplest strategy consists in applying the Gauss Newton method directly on the finest subdivision, i.e. in estimating n_θ^f parameters (cfr. algorithm 7). This problem is particularly demanding for its high computational cost, due to the large number of parameters to be estimated at each Newton's iteration.

9.6.2.2 Algorithm 2: working on the finest subdivision with time localization

As explained in section 9.6.1, in the stationary regime the problem is illposed: time localization corresponds to a better conditioned problem, since it consists in selecting

Algorithm 7 Sketch of the algorithm working on the finest subdivision:

Given the finest subdivision of Γ_h, $\hat{\boldsymbol{\theta}}^0 = \mathbf{0}$, $\mu^0 = 1$;
while $\tilde{J}(\hat{\boldsymbol{\vartheta}}^l) < tol$ **do**
  solve $\psi_{\hat{\theta}^k}\mathbf{s}^k = \mathbf{e}_{\hat{\theta}^k}$;
  $\hat{\boldsymbol{\theta}}^{k+1} = \hat{\boldsymbol{\theta}}^k + \mu^k\mathbf{s}^k$
  projection: for every $j \in [0, n_\theta - 1]$ s.t. $\hat{\theta}^{k+1}(j) < 0$, impose $\hat{\theta}^{k+1}(j) = 0$
  compute $\tilde{J}(\hat{\boldsymbol{\theta}}^{k+1})$
  if $\tilde{J}(\hat{\boldsymbol{\theta}}^{k+1}) > \tilde{J}(\hat{\boldsymbol{\theta}}^k)$ **then**
    $l = 0$;
    $\mu^{k,l} = \frac{\mu^k}{2}$
    while $\tilde{J}(\hat{\boldsymbol{\theta}}^{k+1}) < \tilde{J}(\hat{\boldsymbol{\theta}}^k)$ **do**
      $\hat{\boldsymbol{\theta}}^{k+1} = \hat{\boldsymbol{\theta}}^k + \mu^{k,l}\mathbf{s}^k$
      $l = l + 1$;
      $\mu^{k,l} = \frac{\mu^{k,l}}{2}$
    end while
  end if
end while

only those rows of the sensitivity matrix which are significative, i.e. corresponding to the transitional dynamics.

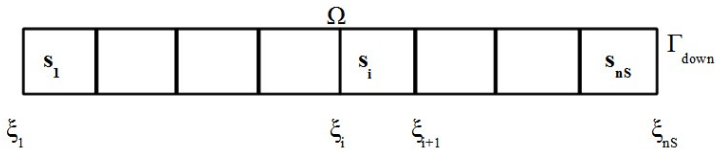

Figure 9.10: *Example of partition of* Ω *in sections.*

More precisely, the idea is to partition the domain Ω in a suitable number $n_s > 1$ of *sections* $\mathcal{U} = \{s_j\}$, $j = 1, \ldots, n_s$ (cfr. e.g. figure 9.10). Referring to figure 9.1, we suppose that $s_j := [\xi_j, \xi_{j+1}] \times [y_1, y_2]$, $\xi_1 = x_1$, $\xi_{n_s+1} = x_2$. In particular in algorithm 2 we assume that $\{\xi_1, \ldots, \xi_{n_s+1}\} = \left\{x_1, \ldots, x_{\frac{n_\theta^f}{2}+1}\right\}$, i.e. it coincides with the finest subdivision. Denote with $I^{(j)}$, the set of parameters belonging to s_j. The parameters belonging to $I^{(j)}$ are estimated using the PDGN method with a TSVD regularization: Starting from s_{n_s}, first it computes the sensitivity matrix only of those parameters belonging to $I^{(n_s)}$ and only in the time interval $[t_0^{(n_s)}, t_f^{(n_s)}]$, $t_0^{(n_s)} \geq t_0$, $t_f^{(n_s)} \leq t_f$; below it is explained how to choose the interval. Then, it considers sections $s_{n_s-1}, s_{n_s-2}, \ldots, s_1$. If we denote with $\mathcal{O}^{(j)}$, $j = 1, \ldots, n_s - 1$, the set of parameters

estimated in section s_{j+1} greater than a threshold $\epsilon_3 > 0$, then in section s_j all the $n_\theta^{(j)}$ parameters belonging to $\mathcal{O}^{(j)} \cup I^{(j)}$ will be estimated, only in the time interval $[t_0^{(j)}, t_f^{(j)}]$. In algorithm 8 previous ideas are summarized.

Algorithm 8 Sketch of the algorithm working on the finest subdivision with time localization:

Given $\{\xi_1, \dots, \xi_{n_s+1}\}$ coincident with the finest subdivision of Γ_h and the threshold $\epsilon_3 > 0$;
while $\tilde{J}(\hat{\boldsymbol{\vartheta}}^k) < tol$ **do**
  $i = n_s$; $\mathcal{O}^{(n_s)} = \emptyset$
  while $i > 0$ **do**
    Let $I^{(i)}$ be the set of parameters of $\hat{\boldsymbol{\vartheta}}^k$ that belongs to section i;
    in $[t_0^{(i)}, t_f^{(i)}]$ apply the regularized projected damped Gauss Newton method to optimize parameters whose indices belong to $I^{(i)} \cup \mathcal{O}^{(i)}$;
    update the positions $I^{(i)} \cup \mathcal{O}^{(i)}$ of $\hat{\boldsymbol{\vartheta}}^k$;
    define $\mathcal{O}^{(i-1)}$ as the set of indices of parameters greater than ϵ_3;
    $i = i - 1$;
  end while
end while

9.6.2.3 Algorithm 3: using an adaptive parametrization

A different improvement is to use an *adaptive parametrization*, i.e. to adaptively update the subdivision of Γ_h used in the current iteration k of the Newton method. This strategy is important since usually the immision occurs only in a local part of Γ_h: using a uniform subdivision would bring to a sparse vector of parameters and would hence require a more computational demanding regularization (e.g. l_1-optimization). Instead, this algorithm tries to localize Γ_{in} in Γ_h and refines the parametrization only around that point. This limits the computational cost, reducing the number of columns of the sensitivity matrix. A similar strategy has been presented in (138, 155), to solve an inverse conduction problem of corrosion estimation (cfr. chapter 8).

The algorithm works as follows: starting from an initial *coarse* subdivision of Γ_h, $\mathcal{S}^{(1)}$, at the k-th iteration the algorithm first computes a Gauss-Newton iteration $\hat{\boldsymbol{\vartheta}}^{(k)} \in \mathbb{R}^{n_\theta^{(k)}}$. For every element of $\hat{\boldsymbol{\vartheta}}^{(k)} \in \mathbb{R}^{n_\theta^{(k)}}$ greater than a fix threshold $\epsilon_1 > 0$, the segment of $\mathcal{S}^{(k)}$ corresponding to that parameter is bisected: thus a new subdivision $\mathcal{S}^{(k+1)}$ is defined adding to $\mathcal{S}^{(k)}$ all the computed middle points. Finally only those parameters which are greater than a fixed threshold $\epsilon_2 > 0$ are selected: we indicate with $\Lambda^{(k)}$ this ensemble. The other parameters remain constant in the following iteration. The main ideas of the adaptive algorithm are sketched in algorithm 9.

To solve the system (9.21) a *diagonal scaling* (162) is used. Here it means that at iteration k, given the subdivision $\mathcal{S}^{(k)}$, for every $i = 1, \dots, n_\theta^{(k)}$, $\Psi_{\hat{\boldsymbol{\vartheta}}^{(k)}}(:, i)$ is multiplied

Algorithm 9 Sketch of the adaptive algorithm:

Given the finest subdivision of Γ_h of step length Δx, the tolerance $tol > 0$ and thresholds $\epsilon_1, \epsilon_2 > 0$, consider the coarse subdivision $\mathcal{S}^{(1)} = \left\{ x_1^1, \dots, x^1_{\frac{n_\theta^1}{2}+1} \right\}$, of Γ_h;
$\hat{\boldsymbol{\vartheta}}^1 = \mathbf{0}_{n_\theta^1} \in \mathbb{R}^{n_\theta^1}$;
$k = 1$, $\Lambda^{(1)} = [1, \dots, n_\theta^1]$, set of indexes of parameters to be optimized
while $\tilde{J}(\hat{\boldsymbol{\vartheta}}^k) < tol$ **do**
  apply the PDGN method, optimizing **only** parameters whose indexes belong to $\Lambda^{(k)}$, obtaining $\hat{\boldsymbol{\vartheta}}^k \in \mathbb{R}^{n_\theta^k}$
  $\mathcal{S}^{(k+1)} := \left\{ x_1^{k+1}, \dots, x^{k+1}_{\frac{n_\theta^{k+1}}{2}+1} \right\} = \mathcal{S}^{(k)}$, $n_\theta^{k+1} = n_\theta^k$, $I = n_\theta^{k+1}$;
  for all $i \in [1, I]$ **do**
    if $\hat{\theta}^k(i) > \epsilon_1\%$ bisect the corresponding segment **then**
      $n_\theta^{k+1} = n_\theta^{k+1} + 1$, $I = I + 1$;
      let $[x^{k+1}(\hat{\theta}^k(i)), x^{k+1}(\hat{\theta}^k(i))]$ be the segment corresponding to parameter $\hat{\theta}^k(i)$;
      $\mathcal{S}^{(k+1)} = \mathcal{S}^{(k+1)} \cup \frac{x^{k+1}(\hat{\theta}^k(i)) - x^{k+1}(\hat{\theta}^k(i))}{2}$,
    end if
  end for
  $\Lambda^{(k+1)} = \emptyset$;
  for all $i \in [1, I]$ **do**
    if $\hat{\vartheta}^k(i) > \epsilon_2$ **then**
      $\Lambda^{(k+1)} = \Lambda^{(k+1)} \cup i$;
    end if
  end for
  $k = k + 1$;
end while

by a weight d_i, equal to the length of the maximal segment of the current subdivision, divided by the length of the segment corresponding to the i-th column. Thus diagonal scaling corresponds to solve

$$\begin{aligned} \Psi_{\hat{\boldsymbol{\vartheta}}^{(k)}} D^{(k)} \tilde{\boldsymbol{s}}^{(k)} &= \mathbf{e}_{\hat{\boldsymbol{\vartheta}}^{(k)}}, \qquad D^{(k)} = diag(d_i^{(k)}), \qquad d_i^{(k)} = \frac{\max_{x_{j+1}^k, x_j^k \in \mathcal{S}^{(k)}} x_{j+1}^k - x_j^k}{x_{i+1}^k - x_i^k}, \\ \mathbf{s}^{(k)} &= D^{(k)} \tilde{\boldsymbol{s}}^{(k)}, \end{aligned} \tag{9.26}$$

instead of (9.21).

9.6.2.4 Algorithm 4: using an adaptive parametrization and time localization

As in algorithm 2, the domain Ω is partitioned in $n_s > 1$ *sections* $\mathcal{U} = \{s_j\}$, $j = 1, \ldots, n_s$, however in algorithm 4 we assume that $\{\xi_1, \ldots, \xi_{n_s+1}\} = \mathcal{S}^{(1)}$, i.e. it coincides with the coarse initial subdivision applied in the adaptive strategy. In section s_j, considering the time interval $[t_0^{(j)}, t_f^{(j)}]$, all parameters belonging to $\mathcal{O}^{(j)} \cup I^{(j)}$ will be estimated, and the adaptive procedure will be applied until a minimum is reached. Observe that this coincides with an internal loop: this trategy is sketched in algorithm 10.

9.6.2.5 Time localization: how to choose time intervals $[t_0^{(i)}, t_f^{(i)}]$

A key point is the choice of the local time intervals $[t_0^{(i)}, t_f^{(i)}]$, for every section s_i, $i = 1, \ldots, n_s$, $t_0^{(i)} \geq t_0$ and $t_f^{(i)} \leq t_f$. The i-th interval must be chosen such that it

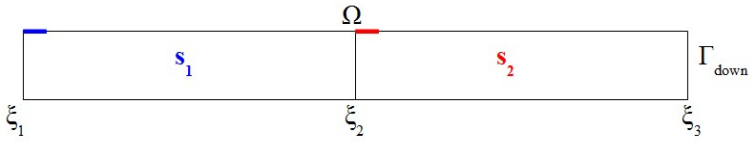

Figure 9.11: *Partition of Ω in 2 sections to draw the curves of figure 9.12: to obtain the red (blue) curve of figure 9.12, it is considered the mean concentration on Γ_{down}, obtained imposing a control different from zero only in the most left segment of the finest subdivision of the upper horizontal segment of section s_2 (s_1), indicated in red (blue).*

contains the transitional dynamics of section s_i but not that of sections s_j, $j < i$. To describe more clearly this idea, consider the model problem introduced in section 9.5.3: moreover suppose for simplicity that $n_s = 2$, $\{\xi_1, \xi_2, \xi_3\} = \{0, 4, 8\}$, as depicted

Algorithm 10 Sketch of the adaptive algorithm with space-time localization:

Given the partition of Ω $\{\xi_1, \dots, \xi_{n_s+1}\} = \mathcal{S}^{(1)}$, the thresholds $\epsilon_1, \epsilon_2, \epsilon_3 > 0$, $\hat{\boldsymbol{\vartheta}}^0 = \mathbf{0}$, $k = 1$;
$\mathcal{O}^{(k,n_s)} = \emptyset$;
while $\tilde{J}(\hat{\boldsymbol{\vartheta}}^k) < tol$ **do**
  $i = n_s$; $\hat{\boldsymbol{\vartheta}}^{k,i} = \hat{\boldsymbol{\vartheta}}^k$, $n_\theta^{k,i} = n_\theta^k$, $\mathcal{S}^{(k,i)} = \mathcal{S}^{(k)}$
  while $i > 0$ **do**
    $l = 1$, $\hat{\boldsymbol{\vartheta}}^{k,i,l} = \hat{\boldsymbol{\vartheta}}^{k,i}$, $n_\theta^{k,i,l} = n_\theta^{k,i}$, $\mathcal{S}^{(k,i,l)} = \mathcal{S}^{(k,i)}$, $\mathcal{O}^{(k,i,l)} = \mathcal{O}^{(k,i)}$;
    $\Lambda^{(k,i,l)} = [1, \dots, n_\theta^{k,i,l}]$, set of indices of parameters to be optimized
    while a minimum is reached% apply the adaptive strategy **do**
      Let $I^{(k,i,l)}$ be the set of parameters of $\hat{\boldsymbol{\vartheta}}^{k,i,l}$ that belongs to section i;
      in $[t_0^{(i)}, t_f^{(i)}]$ apply the PDGN method to optimize parameters whose indices belong to $P^{(k,i,l)} := (I^{(k,i,l)} \cup \mathcal{O}^{(k,i,l)}) \cap \Lambda^{(k,i,l)}$;
      update the positions $P^{(k,i,l)}$ of $\hat{\boldsymbol{\vartheta}}^{k,i,l}$;
      $\mathcal{S}^{(k,i,l+1)} = \mathcal{S}^{(k,i,l)}$;
      for all $j \in [1, n_\theta^{k,i,l}]$ **do**
        if $\hat{\theta}^{k,i,l}(j) > \epsilon_1$ **then**
          update $\mathcal{S}^{(k,i,l+1)}$, bisecting the segment corresponding to $\hat{\theta}^{k,i,l}(j)$;
        end if
      end for
      $\Lambda^{(k,i,l+1)} = \emptyset$;
      for all $j \in [1, n_\theta^{k,i,l+1}]$ **do**
        if $\hat{\vartheta}^{k,i,l}(j) > \epsilon_2$ **then**
          $\Lambda^{(k,i,l+1)} = \Lambda^{(k,i,l+1)} \cup j$;
        end if
      end for
      $\hat{\boldsymbol{\vartheta}}^{k,i,l+1}$ corresponds to $\hat{\boldsymbol{\vartheta}}^{k,i,l}$ values on the finer subdivision $\mathcal{S}^{(k,i,l+1)}$;
      $\mathcal{O}^{(k,i,l+1)}$ corresponds to $\mathcal{O}^{(k,i,l)}$ values on the finer subdivision $\mathcal{S}^{(k,i,l+1)}$;
      $l = l + 1$;
    end while
    $\mathcal{S}^{(k,i)} = \mathcal{S}^{(k,i,l)}$; $\hat{\boldsymbol{\vartheta}}^{k,i} = \hat{\boldsymbol{\vartheta}}^{k,i,l}$
    define $\mathcal{O}^{(k,i-1)}$ as the set of indices of parameters greater than ϵ_3;
    $i = i - 1$;
  end while
  $\hat{\boldsymbol{\vartheta}}^k = \hat{\boldsymbol{\vartheta}}^{k,i}$
  $k = k + 1$;
end while

in figure 9.11, and consider as the finest subdivision a uniform one of step length 0.5. Consider figure 9.12: the j-th curve ζ_j, $j = 1, 2$, represents the mean concentration (left) and its derivative (right) at the outflow when the boundary control is different from zero only in the most left position of s_j with respect to the finest subdivision. The interval corresponding to s_2 can be $[t_0^{(2)}, t_f^{(2)}] = [180, 260]$, when the red dotted

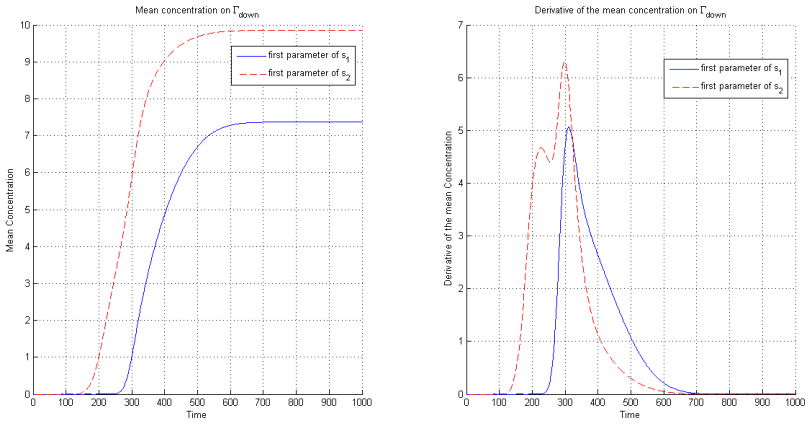

Figure 9.12: *Time evolution of the mean concentrations at Γ_{down}, ζ_1 and ζ_2, (left) and their derivative (right) for different boundary controls: the boundary control is different from zero only in the most left position of the finest subdivision of s_1 (blue) and s_2 (red).*

curve corresponding to s_2, ζ_2 , is increasing (transitional regime) and the blue curve corresponding to s_1, ζ_1, is flat, i.e. when only the pollutant released into Ω in s_2 could reach Γ_{down}. While in s_1 the choice can be $[t_0^{(1)}, t_f^{(1)}] = [240, 400]$, since in this interval the transitional regime of s_1 occurs, as showed by ζ_1. This intervals are used in section 9.6.4, to test algorithm 4.

The previous idea can be extended more rigorously to a general number of sections: let ζ_i, $i = 1, \ldots, n_s$, be the mean concentration at the outflow Γ_{down} when the boundary control is different from zero only in the most left position of s_i, with respect to the finest subdivision. Consider a small threshold $\epsilon_4 > 0$, and two positive parameters $d, D > 0$. Given

$$\begin{aligned} t_0^{(n_s)} &= \min_{t\in[t_0,t_f]} \left\{ \zeta'_{n_s}(t) > \epsilon_4 \text{ and } \zeta'_{n_s-1}(t) < \epsilon_4 \right\}, \\ t_f^{(n_s)} &= \max_{t\in[t_0,t_f]} \left\{ \zeta'_{n_s}(t) > \epsilon_4 \text{ and } \zeta'_{n_s-1}(t) < \epsilon_4 \right\}, \end{aligned}$$

then for $i = 1, \dots, n_s - 1$

$$
\begin{aligned}
t_0^{(i)} &= t_f^{(i+1)} - d, \\
t_f^{(i)} &= \begin{cases} \max_{t\in[t_0,t_f]} \left\{ \zeta_i^{'}(t) > \epsilon_4 \text{ and } \zeta_{i-1}^{'}(t) < \epsilon_4 \right\}, & i > 1 \\ \min\left\{ t_f^{(i+1)} + D, \max_{t\in[t_0,t_f]} \left\{ \zeta_i^{'}(t) > \epsilon_4 \text{ and } \zeta_{i-1}^{'}(t) < \epsilon_4 \right\} \right\}, & i = 1. \end{cases}
\end{aligned}
$$

The parameter d allows a small overlapping between local time intervals, while D could limit the length of the inteval $[t_0^{(1)}, t_f^{(1)}]$: in the example presented above $n_s = 2$, $d = 0.2$ and $D = 1.6$.

Observe that the definition of the intervals $[t_0^{(i)}, t_f^{(i)}]$ depends on the shape of the domain, on the velocity field and on the coefficients of the PDE (9.1): each time one of them is changed, also the intervals should be estimated, observing the transitional dynamics of each section, as explained above.

9.6.3 Comparing computational costs

In this section we compare the computational costs of the four algorithms. Note that each algorithm require a certain number of direct problem solutions, whose cost amounts to NN_h^β each, where β depends on the numerical method used: tipically $\beta = 2$ for direct methods and down to $\beta = 1.5$ for preconditioned iterative methods. At each iteration, computing the new prediction error (9.23) costs NN_h^β.

The first algorithm consists in using the finest subdivision, with the projected damped Gauss Newton strategy. The computational cost of each iteration is pretty high: computing the sensitivity matrix $\Psi_{\boldsymbol{\vartheta}} \in \mathbb{R}^{n_y N \times n_\theta^{(f)}}$ has cost $n_\theta^{(f)} N N_h^\beta$, where n_θ^f is the number of parameter of the finest subdivision, which is maximal. Moreover computing the SVD to obtain the new iteration has cost $4n_y^2 N^2 n_\theta^f + 8N n_y (n_\theta^f)^2 + 9(n_\theta^f)^3$.

To decrease the cost, the idea is to consider a sensitivity matrix of lower dimensions. The second algorithm consists in combining the finest subdivision with localization in time. The number of sections in this case coincides with one half of the number of parameters of the finest subdivision n_θ^f. At each iteration k, for every section $i = 1, \dots, n_s$, $n_s = \frac{n_\theta^{(f)}}{2}$, computing $\Psi_{\boldsymbol{\vartheta}}^{(i)} \in \mathbb{R}^{n_y \frac{t_f^{(i)} - t_0^{(i)}}{Dt} \times n_\theta^{(k,i)}}$ costs $n_\theta^{(k,i)} (\frac{t_f^{(i)} - t_0^{(i)}}{Dt}) N_h^\beta$, where $n_\theta^{(k,i)}$ denotes the cardinality of $I^{(i)} \cup \mathcal{O}^{(i)}$. Moreover computing the SVD to obtain the new iteration has cost $4n_y^2 \left(\frac{t_f^{(i)} - t_0^{(i)}}{Dt} \right)^2 n_\theta^{(k,i)} + 8 \frac{t_f^{(i)} - t_0^{(i)}}{Dt} n_y (n_\theta^{(k,i)})^2 + (n_\theta^{(k,i)})^3$. Although a higher number of systems must be solved, the algorithm is less costly since the sensitivity matrix has much lower dimensions.

Another possibility to decrease the cost of algorithm one, is to use the third algorithm, which consists in adopting an adaptive parametrization. At the $k-$th iteration

computing the sensitivity matrix $\Psi_{\boldsymbol{\vartheta}} \in \mathbb{R}^{n_y N \times n_\theta^{(k)}}$ has cost $n_\theta^{(k)} N N_h^\beta$, where the number of parameters $n_\theta^{(k)}$ varies during the iterations and $n_\theta^{(k)} < n_\theta^f$. The gain with respect to the first strategy is evident if $n_\theta^{(k)} << n_\theta^{(f)}$.

The fourth algorithm combines both time localization and the adaptive parametrization. The number of sections in this case coincides with one half the number of parameters of the initial coarse subdivision $\mathcal{S}^{(1)}$. The difference with respect to the second algorithm is that the number of sections n_s is lower, because it is no more related to the finest subdivision: in fact the adaptive parametrization guides the choice of parameters to be estimated at each iteration. However the introduction of the adaptive parametrization introduces an inner loop. In detail, at each iteration k, for every section $i = 1, \ldots, n_s$, applying the adaptive procedure until a minimum is reached (index l), computing $\Psi_{\boldsymbol{\vartheta}}^{(k,i,l)} \in \mathbb{R}^{n_y \frac{t_f^{(i)} - t_0^{(i)}}{Dt} \times n_\theta^{(k,i,l)}}$ costs $n_\theta^{(k,i,l)} (\frac{t_f^{(i)} - t_0^{(i)}}{Dt}) N_h^\beta$.

Computational costs of the four algorithms are summarized in table 9.2, averaging results of tests presented in section 9.6.4.

	Computation of Ψ at k-th iteration	SVD of Ψ at k-th iteration	**Total computational cost**
Finest subdivision	$n_\theta^f N N_h^\beta$	$4n_y^2 N^2 n_\theta^f + 8N n_y (n_\theta^f)^2 + 9(n_\theta^f)^3$	$1.3 \cdot 10^{12}$
Finest subdivision + space-time localization	$n_\theta^{(k,i)} (\frac{t_f^{(i)} - t_0^{(i)}}{Dt}) N_h^\beta$ $i = 1, \ldots, n_s$	$4n_y^2 \left(\frac{t_f^{(i)} - t_0^{(i)}}{Dt} \right)^2 n_\theta^{(k,i)} + 8 \frac{t_f^{(i)} - t_0^{(i)}}{Dt} n_y (n_\theta^{(k,i)})^2 + (n_\theta^{(k,i)})^3$ $i = 1, \ldots, n_s$	$9 \cdot 10^{10}$
Adaptive subdivision	$n_\theta^{(k)} N N_h^\beta$	(not required)	$9 \cdot 10^9$
Adaptive subdivision + space-time localization	$n_\theta^{(k,i,l)} (\frac{t_f^{(i)} - t_0^{(i)}}{Dt}) N_h^\beta$ $i = 1, \ldots, n_s$	(not required)	$8 \cdot 10^8$

Table 9.2: *Estimated computational cost of the four algorithms previously described.*

9.6.4 Numerical results

In this section we present some numerical tests to verify the effectiveness of the algorithms. As in section 9.5.3, experimental data are simulated numerically, on $\Omega = [0, 8] \times [0, 1]$, $\Gamma_h = [0, 8] \times \{1\} \cup [0, 8] \times \{0\}$. Moreover the velocity field $\mathbf{u}$ is modeled

as a Poiseuille flow i.e.

$$\mathbf{u}(x_1, x_2) = \begin{pmatrix} -4\nu x_2^2 + 4\nu x_2 \\ 0 \end{pmatrix}.$$

We assume that $\mu = 0.1$, $\sigma = 0.1$ and $c_{up} = 0.1$. Moreover we consider the finest subdivision with step length $\Delta x = 0.5$. In algorithm 2 we consider $\{\xi_1, \ldots, \xi_{n_s+1}\}$ coincident with the finest subdivision, while in algorithm 4 $n_s = 2$ and $\{\xi_1, \ldots, \xi_{n_s+1}\} = \{0, 4, 8\}$. Define *optimal subdivision* the one which describes the real profile with the minimum number of parameters using the bisection criterium. With *distance from the optimal subdivision* we indicate the number of points added (sign +) or subtracted (sign -) to the optimal sundivision. We consider the 9 test cases described in Table 9.3: each one is characterized by two vectors $\boldsymbol{\theta}_{up} \in \mathbb{R}^{16}$ and $\boldsymbol{\theta}_{down} \in \mathbb{R}^{16}$ which represent the concentration of pollutant released in each subsegment of the finest subdivision of Γ_h; only the elements different from zero are indicated. $\boldsymbol{\theta}_{up}$ represents the upper horizontal segment, whereas $\boldsymbol{\theta}_{down}$ the bottom one.

Test	$\boldsymbol{\theta}_{up}$	$\boldsymbol{\theta}_{down}$
1	$\theta_{up}(2) = 100$	
2		$\theta_{down}(2) = 100$
3	$\theta_{up}(12) = 100$	
4	$\theta_{up}(2) = 100$, $\theta_{up}(3) = 80$	
5	$\theta_{up}(2) = 100$, $\theta_{up}(4) = 80$	
6	$\theta_{up}(2) = 100$, $\theta_{up}(12) = 80$	
7	$\theta_{up}(2) = 100$	$\theta_{down}(2) = 80$
8	$\theta_{up}(2) = 100$	$\theta_{down}(12) = 80$
9	$\theta_{up}(2) = 100$, $\theta_{up}(3) = 80$, $\theta_{up}(12) = 60$	

Table 9.3: *Test cases: only the elements different from zero are indicated.*

In table 9.4, the four algorithms are compared: using the finest subdivision and the projected damped Gauss Newton method, using the finest subdivision and the localization in time, using the adaptive parametrization and using the adaptive parametrization and time localization.

9. INVERSE CONVECTION PROBLEM

Test		Finest subdivision			Finest subdivision + time localization			Adaptive subdivision			Adaptive subdivision + time localization		
		up	*down*		*up*	*down*		*up*	*down*		*up*	*down*	
1	L^1-*err*	10^{-12}	10^{-12}		0.442	0.442		7.69	0.12		1.15	0.168	
	opt. sub.	+11	+14		+11	+14		+1	0		+1	0	
	num. it.			2			18			4			7
	$\tilde{J}(\boldsymbol{\vartheta})$			10^{-20}			10^{-6}			10^{-5}			10^{-4}
2	L^1-*err*	10^{-12}	10^{-12}		0.02	0.02		0.12	7.69		0.168	1.15	
	opt. sub.	+14	+11		+14	+11		0	+1		0	+1	
	num. it.			2			20			4			7
	$\tilde{J}(\boldsymbol{\vartheta})$			10^{-20}			10^{-6}			10^{-5}			10^{-4}
3	L^1-*err*	2.72	0.3974		0	0		1.33	0.02		0.18	10^{-3}	
	opt. sub.	+11	+14		+11	+14		0	+1		+2	0	
	num. it.			6			17			9			9
	$\tilde{J}(\boldsymbol{\vartheta})$			0.028			10^{-20}			0.0012			10^{-5}
4	L^1-*err*	8.911	0.1102		1.021	10^{-3}		11.25	0.16		2.247	0.07	
	opt. sub.	+10	+14		+10	+14		0	0		0	+1	
	num. it.			5			13			4			8
	$\tilde{J}(\boldsymbol{\vartheta})$			10^{-3}			10^{-3}			10^{-3}			10^{-5}
5	L^1-*err*	5.576	0.047		1.611	10^{-4}		12	0.14		2.224	0.03	
	opt. sub.	+10	+14		+10	+14		-1	0		+1	+1	
	num. it.			5			11			3			7
	$\tilde{J}(\boldsymbol{\vartheta})$			10^{-3}			10^{-4}			10^{-4}			10^{-5}
6	L^1-*err*	2.653	0.2871		8.591	10^{-13}		2.33	0.01		2.48	0.01	
	opt. sub.	+8	+14		+8	+14		+1	0		+3	0	
	num. it.			5			17			10			15
	$\tilde{J}(\boldsymbol{\vartheta})$			10^{-2}			0.068			10^{-4}			10^{-4}
7	L^1-*err*	10^{-13}	10^{-13}		0.36	0.36		7.63	6.12		1.267	1.019	
	opt. sub.	+9	+9		+9	+9		0	0		+1	+1	
	num. it.			2			17			4			9
	$\tilde{J}(\boldsymbol{\vartheta})$			10^{-20}			10^{-6}			10^{-5}			10^{-6}
8	L^1-*err*	1.969	1.002		6.25	9.17		14.9	8.9		0.95	0.95	
	opt. sub.	+9	+9		+9	+9		+2	0		+1	+2	
	num. it.			5			17			5			13
	$\tilde{J}(\boldsymbol{\vartheta})$			10^{-2}			0.13			0.19			10^{-5}
9	L^1-*err*	14.22	0.1818		14.34	10^{-12}		2.65	0.9		2.01	0.004	
	opt. sub.	+7	+14		+7	+14		+3	0		+2	0	
	num. it.			11			19			16			21
	$\tilde{J}(\boldsymbol{\vartheta})$			10^{-2}			0.2			0.001			10^{-4}

Table 9.4: *Comparison between four algorithms: L^1-error in the upper and lower horizontal segments, number of points added to the optimal subdivision in the upper and lower horizontal segments, number of iterations and final cost function.*

First of all observe that the number of iterations of algorithms 2 and 4 is higher since also sub-iterations to reach the minimum inside each section are counted (inner

loop). In tests 1, 2 and 7, also working on the finest subdivision performs well, but it is much more costly. When the condition number of the sensitivity matrix $\Psi_{\boldsymbol{\vartheta}}$ increases, the accuracy is low. In particular in tests 3, 4, 5 it is evident how time localization improves convergence results both in algorithms 2 and 4, selecting only some rows of $\Psi_{\boldsymbol{\vartheta}}$. However adopting only space-time localization is not sufficient in tests 6,7,9. In algorithm 3, instead, the number of points added to the optimal subdivision is very low, but in general the estimates are not accurate enough and thus the error is high. The best strategy consists in combining both adaptive parametrization and time localization (algorithm 4): this is a good compromise between good estimates and reasonable computational cost. Its effectiveness is evident e.g. in tests 8 and 9. Moreover it only adds few points to the optimal subdivision.

In figures 9.13 and 9.14 different iterations of algorithm 4 are shown for test 8: it is evident how the algorithm firstly optimize parameters of section $s_2 = [4, 8] \times [0, 1]$ (figure 9.14), and then that of $s_1 = [0, 4] \times [0, 1]$ (in figure 9.13 the first 7 iteration are identical to the first one, thus only iterations 1 and 8 are plotted). The estimated subdivision is sketched in figure 9.15: it is evident how algorithm 4 slightly over-refine the optimal subdivision.

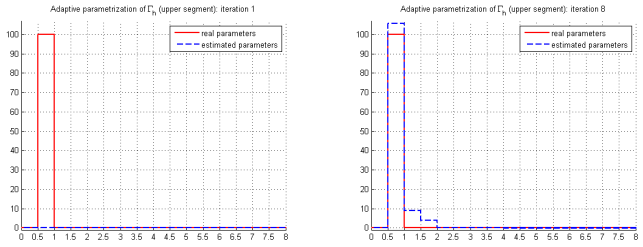

Figure 9.13: *Test 8. Adaptive parametrization and time localization. Evolution of the approximation (blu dotted line), real control (red line). Upper horizontal segment.*

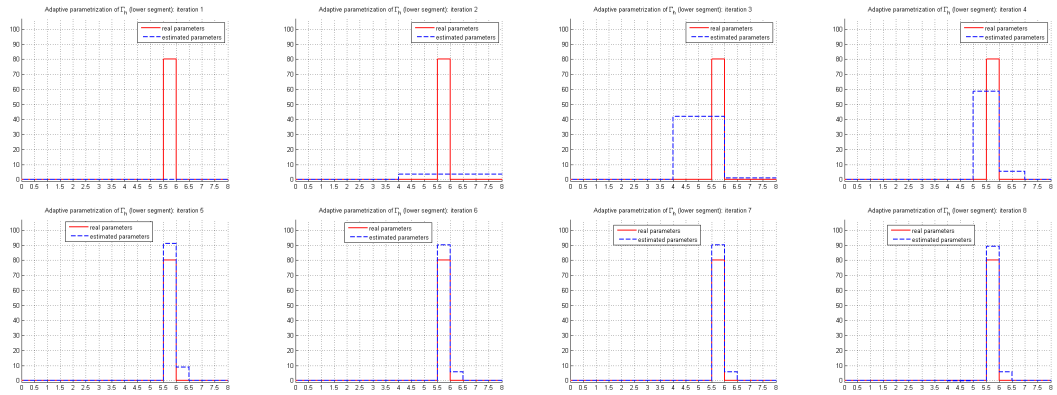

Figure 9.14: *Test 8. Adaptive parametrization and time localization. Evolution of the approximation (blu dotted line), real control (red line). Bottom horizontal segment.*

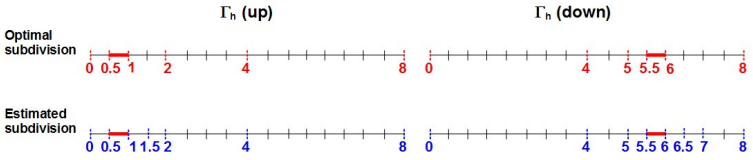

Figure 9.15: *Test 8. First row: optimal subdivision that could be obtained using a bisection strategy. Second row: estimated subdivision.*

Results for the adaptive strategy with localization in time for all tests are shown in figure 9.16.

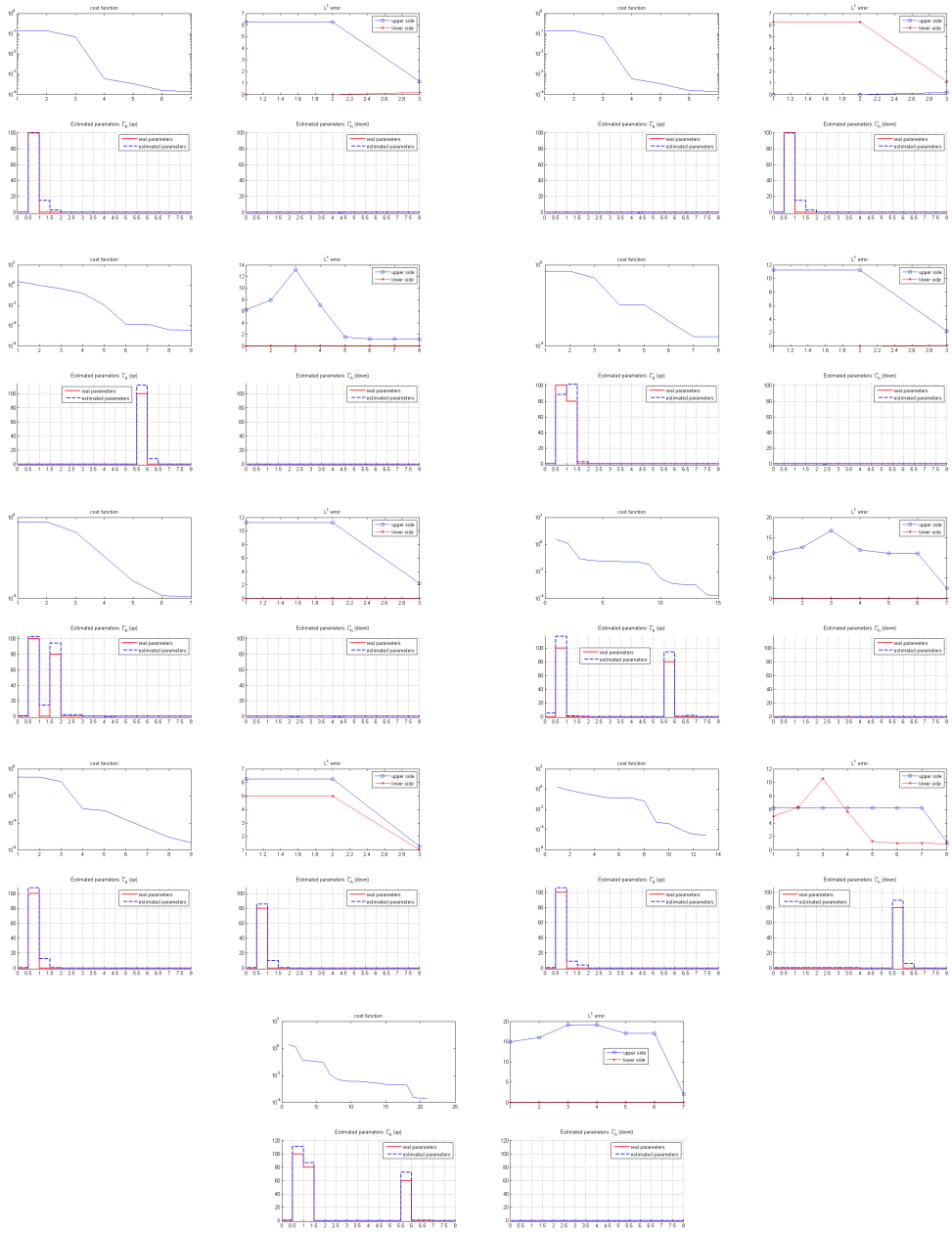

Figure 9.16: *Nine test cases: results of the adaptive strategy with time localization: computed estimate (blu dotted line), real control (red line). For each figure: cost function (first row, left), L^1 error (first row, right), approximation of the upper horizontal segment (second row, left), approximation of the bottom horizontal segment (second row, right).*

9.6.5 Conditioning of the problem

The ill-conditioning of the system matrix $\Psi_{\hat{\boldsymbol{\vartheta}}}$ could increase when smaller segments are considered in Γ_h: in fact in this case consecutive columns tend to be close to linear dependence, due to the small distance (Δx) of the corresponding nodes in Γ_h. This can be demonstrated numerically: consider in fact the example presented in section 9.5.3 and generalize it considering the following parametric problem

$$\Gamma_{in} = [5-h, 5] \times \{1\} \cup [2-h, 2] \times \{0\}\,, \ \boldsymbol{\vartheta} = (100, 80), \qquad 0 < h \leq 2.$$

Even supposing to know source location Γ_{in}, solving the problem for different values of $h = \{0.0625, 0.125, 0.25, 0.5, 1, 2\}$ and computing the condition number of the sensitivity matrix, it can be seen that as h decreases, the condition number increases (cfr. figure 9.17). Since the condition number of the sensitivity matrix could become higher when

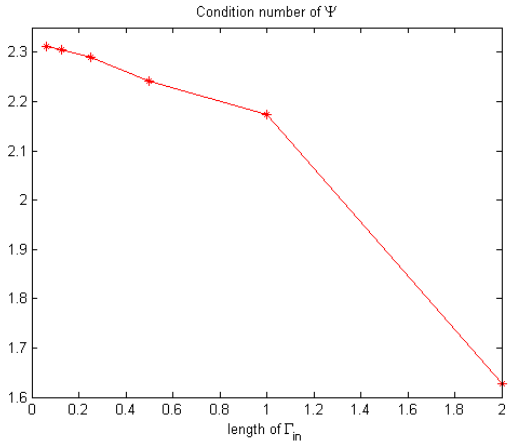

Figure 9.17: *Example 2, with* $\Gamma_{in} = [5-h, 5] \times \{1\} \cup [2-h, 2] \times \{0\}$. *Condition number of* $\Psi_{\hat{\boldsymbol{\vartheta}}}$ *for different values of* $h = \{0.0625, 0.125, 0.25, 0.5, 1, 2\}$.

smaller segments are considered, working on the finest subdivision could not be effective to reduce the ill-conditioning of the problem and an adaptive parametrization should be preferred. Observe moreover that in adaptive algorithms the Gauss Newton method is applied only to those parameters belonging to $\Lambda^{(k)}$: avoiding parameters less than the threshold ϵ_2 is useful to reduce columns linear dependence.

Moreover, as analyzed in section 9.6.1, at the stationary regime, the problem becomes ill-conditioned: thus, considering only the transitional regime, space-time localization could limit the ill-conditioning of the problem. In the 9 tests, on the finest subdivision, the mean condition number is $O(10^5)$ or $O(10^3)$ respectively using or not

space-time localization. Using an adaptive parametrization, it reduces from $O(10^3)$ to $O(10^2)$.

9.6.6 Sensitivity of the fourth algorithm to thresholds variations

It is interesting to analyze what happens when thresholds used in the fourth algorithm are changed. ϵ_1 decides when a the segment corresponding to a parameter should be refined: it is important to keep it not too low, to avoid over-refinements. ϵ_2 is such that parameters less than it are not considered to build the sensitivity matrix: avoiding small parameters reduces computational cost and the ill-conditioning of the problem, since we expect that they are not effective in output variations.

Previous observations are summarized in table 9.5, where test 1 is considered to understand how convergence results varies when thresholds are slightly changed: when ϵ_1 is decreased the over-refinement increases, while when ϵ_2 is lower both the computational cost (number of iterations) and the condition number increase. When both ϵ_1 and ϵ_2 decrease both the distance from the optimal subdivision and the computational cost and the condition number increase. Thus in general to reduce the cost is it better to increase ϵ_1, while to obtain more accurate results it could be useful to adopt smaller ϵ_1 and ϵ_2.

ϵ_1	ϵ_2	L^1 error:		opt. sub.:		$\tilde{J}(\boldsymbol{\vartheta})$	num. it.	mean condition number of Ψ
		up	down	up	down			
0.4	0.4	1.15	0.168	+1	0	10^{-4}	7	79.9513
0.3	0.4	1.15	0.168	+1	+1	10^{-4}	7	79.9513
0.01	0.4	1.15	0.168	+1	+7	10^{-5}	7	79.9513
0.4	0.3	1.192	0.02	+1	0	10^{-5}	8	173.2498
0.4	0.01	1.207	0.05	+1	0	10^{-6}	9	252.7891
0.01	0.01	1.259	0.01	+1	+3	10^{-6}	9	210.4405

Table 9.5: *Test 1: results for different values of ϵ_1 and ϵ_2.*

9.7 The importance of stabilizing the problem

Dealing with convection dominated problems ($\|\mathbf{u}\| >> \mu$) could be problematic, due to spurious oscillations caused by the standard FE method. The simplest way to stabilize the problem is to refine the mesh, i.e. to consider a higher number of degrees of

freedom; otherwise on a coarse mesh a stabilization method such as SUPG, DW or GLS, to mention only some of them, should be used (cfr. chapter 3). To simplify the problem in the following we apply the simplest strategy, i.e. we refine the mesh. However a stabilization method could be included in the model, modifying the weak FE formulation. Stabilization techniques are used e.g. in (130, 136); also a time dependent extension of the BAWR method, introduced in section 3.6, could be used to stabilize the problem.

In this section we want to point out that the problem must be stabilized to obtain a correct estimate. In fact consider $\Omega = [0,8] \times [0,1]$, $\Gamma_h = [0,8] \times \{1\} \cup [0,8] \times \{0\}$, the velocity field $\mathbf{u}$ is modeled as a Poiseuille flow i.e.

$$\mathbf{u}(x_1, x_2) = \begin{pmatrix} -4\nu x_2^2 + 4\nu x_2 \\ 0 \end{pmatrix},$$

assume moreover that $\mu = 0.1$, $\sigma = 0.1$, $c_{up} = 0.1$ and $\Gamma_{in} = [0.5, 1] \times \{1\}$, $\vartheta = 100$. Apply to it the adaptive strategy with time localization, on different meshes. Results are depicted in figure 9.18. As it can be seen, when the mesh is too coarse, the presence of spurious oscillations compromise the convergence of the algorithm to the real profile, whereas adopting a fine mesh eliminates them and gives a good estimate of the boundary control.

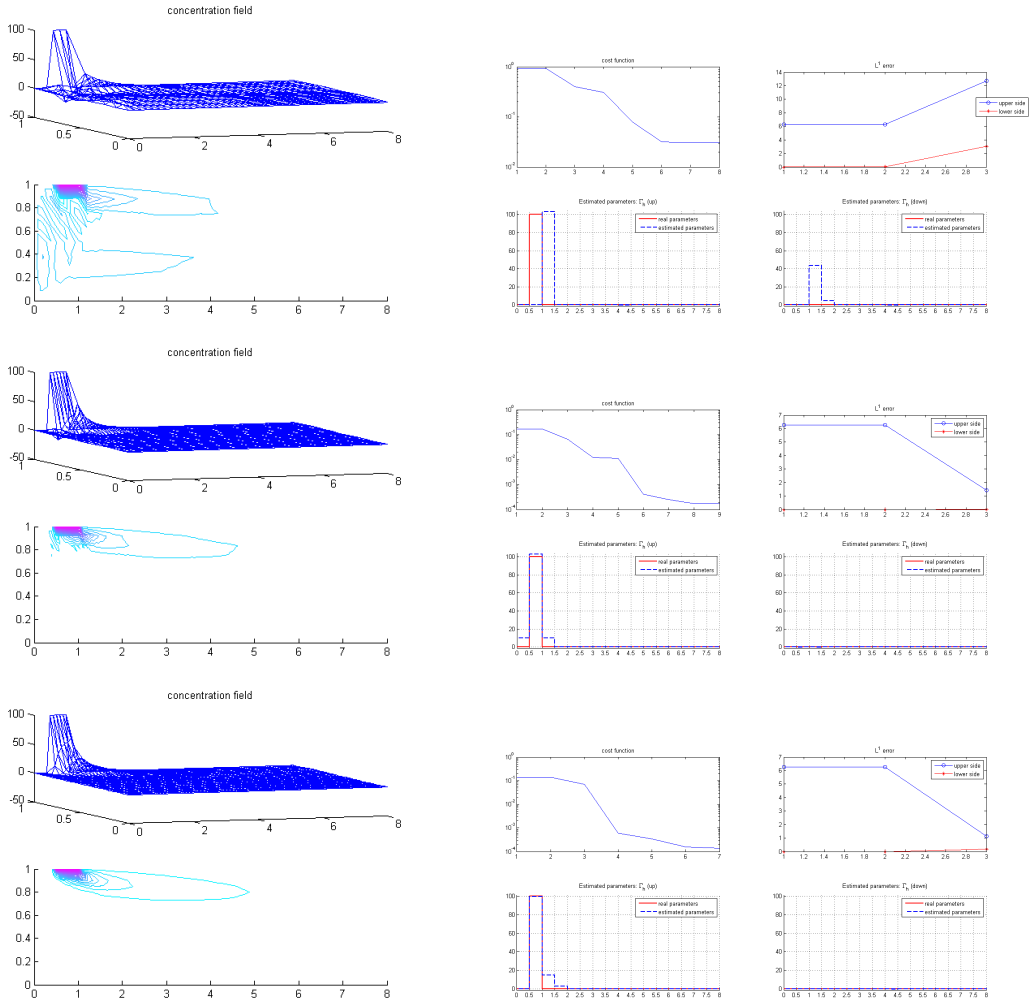

Figure 9.18: *Importance of using stabilization: concentration field (left), estimated profile (right). First row: using* 41 *nodes along x-axis and* 9 *along y-axis. Second row: using* 81 *nodes along x-axis and* 13 *along y-axis. Third row: using* 81 *nodes along x-axis and* 21 *along y-axis.*

10

Conclusions

The first part of the thesis describes the parabolic models considered in the following parts. In particular in chapter 3 convection-dominated problems are analyzed and the *Best Approximation Weighted Residuals (BAWR)* method is presented, which is an analytical, parameter-free, Petrov-Galerkin method that gives stable solutions of convection dominated boundary-value problems. The method has an analytic foundation (Theorem 3.6.1) and it is computationally efficient thanks to an approximation, made to the optimal weighting functions. The estimated order of convergence is asyntotically one and two for H^1 and L^2 norms respectively: it has been studied in section 3.6.2 both from an analytical and a numerical point of view. Numerical tests and benchmarks for convection-dominated problems have been presented in section 3.6.4 (cfr. e.g. (14, 27, 35). They show the effectiveness of the BAWR method, compared with the Galerkin method. The BAWR solution can be improved by a post-processing technique (60), thanks to its least-squares character, that makes more meaningful a direct comparison with other stabilization methods. Possible future perspectives are going in this direction (see (60) for preliminary results).

The third part is about *parabolic inverse problems*. The corrosion estimation problem is described in chapter 8. The underlying dynamic is described by the heat equation and the adopted numerical approach is based upon an adaptive FE discretization over a variable domain. The inverse problem consists in estimating the vector of parameters that best describes the depth of the real corroded profile. Two algorithms have been presented: Inner-Outer Loop algorithm and Predictor-Corrector. While the first one is more simple, usually it over-refine S, it is computationally more expensive and corresponds to a worse conditioned problem. Instead the predictor-corrector strategy uses a linear strategy to substitute the outer loop and it is able to limit the local refine-

ment procedure to proper parts of S, using the norm and the mean of the prediction error. This strategy allows the presence of small overestimates, penalizing huge ones. Due to its linear predictor step and the application of the corrector step only to some selected parameters, it is both less computational expensive and better conditioned. Conducted numerical experiments reveals its ability to refine only where it is necessary and its tendency to obtain small overestimates of the corroded profile.

The problem of pollution rate estimation is introduced in chapter 9: it extends some ideas presented in (138, 156). Both liquid (e.g. water) and gas (e.g. air) pollution problems could be considered: when source location is known, we have demonstrated that the problem can be solved e.g. using the Projected Damped Gauss Newton method. When Γ_{in} is unknown, we have compared four solution strategies: working on the finest subdivision or adopting an adaptive parametrization, and considering for both of them also time localization. It has been proved that adaptive parametrization with time localization (algorithm 4 introduced in chapter 9) is an effective strategy to estimate a vector of parameters representig the pollutant released in the fluid. It is interesting to note that it could be introduced also an unrefinement strategy, trying to get closer to the optimal subdivision. For example consider figure 10.1: the optimal strategy would estimate only one parameter in $[1,2] \times \{1\}$, and it would not bisect the segment $[1,2]$. Instead algorithm 4 bisects $[1,2]$: the problem here is that the direction of the convective field $\mathbf{u}$ produces an overestimate of the right hand side parameter of $[1,2]$ and an underestimate of the left hand side one. Another interesting aspect could be the gen-

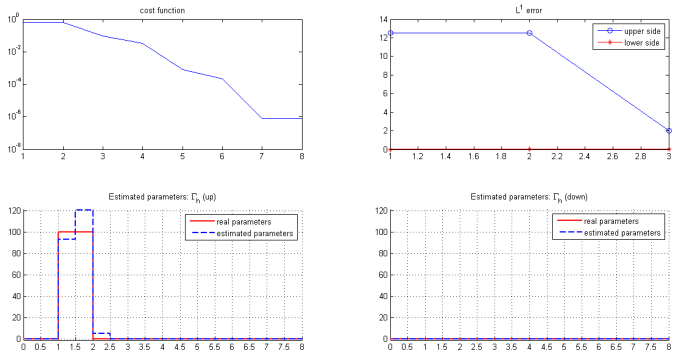

Figure 10.1: *Need of an under-refinement strategy.*

eralization of the problem to time varying boundary conditions on Γ_{in} and to analyze more deeply the problem when space-time varying velocity fields are considered.

Appendix A

Some classical results

A.1 Introduction

The aim of this appendix is to summarize some classical results concerning classical methods used for discretizing boundary value problems, focusing on the Galerkin method and some generalizations of it. We will follow the very well written presentation of (65): we only mention the most important Theorems about a priori error analysis: the interested reader could find more about these arguments e.g. in (4, 15, 64, 65, 71).

A.2 Definition of the continuous problem

Let Ω be an open, limited and lipschitz continuous boundary subset $\Omega \subset \mathbb{R}^n$, sufficiently regular. We denote with $\partial\Omega$ its boundary and with $\partial\Omega^*$ a subset of $\partial\Omega$. Consider the boundary value problem

$$\begin{cases} \mathcal{L}\Theta = f & in \quad \Omega \\ B\Theta = 0 & on \quad \partial\Omega \end{cases} \tag{A.1}$$

where f is a given function, $\mathcal{L}$ is a linear differential operator, often unbounded in $L^2(\Omega)$, B is an affine boundary operator and $\Theta \in X \subset L^2(\Omega)$ is the unknown. The space X is defined such that $\mathcal{L}$ and B have meaning for functions belonging to it.

Problem (A.1) can be reformulated in a weak (variational) form, which allows to search *weak* solutions in an Hilbert space $V \supseteq X$ of *admissible* solutions, which don't necessarily satisfy (A.1) in a pointwise manner. This is possible choosing an Hilbert *weighting functions space* W, multiplying $\mathcal{L}\Theta = f$ by an arbitrary test function, integrating on Ω and applying boundary conditions $B\Theta = 0$ after using the *Green's Lemma*. The choice of V and W depends on $\mathcal{L}$ and B.

The corresponding variational problem can be stated in the following way:

$$\text{find } \Theta \in V \text{ s.t. } a(\Theta, w) = F(w), \qquad \forall w \in W, \tag{A.2}$$

where $a(\cdot,\cdot)$ is a bilinear form $a : V \times W \to \mathbb{R}$ corresponding to $\mathcal{L}$ and $F(\cdot)$ is a linear operator $F : \ W \to \mathbb{R}$, which accounts for the right hand side f and for the possible non-homogeneous Neumann boundary terms. It is important to underline that both the choice of the spaces V and W and the forms a and F striclty depend upon the differential operators $\mathcal{L}$ and $F(\cdot)$. Moreover in general Dirichlet boundary conditions are *essential*, i.e. they are imposed explicitly (strongly) in the choice of functional spaces V and W, while Neumann boundary conditions are *natural*, because they are imposed by the weak formulation itself, choosing a proper F (24, 65).

Following (24), we give the following

Definition A.2.1 *(Well-posedness) Problem (A.2) is said to be well-posed it admits one and only one solution and if the following a priori estimate holds:*

$$\exists c > 0 \ s.t. \ \forall F \in W^{'}, \ \|\Theta\|_V \leq c \|\Theta\|_{W'} ,$$

where Θ is the solution of the corresponding (A.2).

Thus it is important to understand which properties must be satisfied by a and F such that the variational problem (A.2) is *well-posed*.

If $W = V$ the following Theorem gives sufficient conditions for well-posedness:

Theorem A.2.1 ***(Lax-Milgram lemma)**. Let V be a (real) Hilbert space, endowed with the norm $\|\cdot\|$, $a : V \times V \to \mathbb{R}$ a bilinear form and $F : \ V \to \mathbb{R}$ a linear continuous operator. Assume moreover that a is continuous, i.e.*

$$\exists \gamma > 0 : \ |a(\Theta, w)| \leq \gamma \|\Theta\| \|w\| , \ \forall \ \Theta, w \in V,$$

and (strongly) coercive (or V-elliptic), i.e.,

$$\exists C > 0 : \ |a(\Theta, \Theta)| \geq C \|\Theta\|^2 , \ \forall \ \Theta \in V.$$

Then there exists a unique $\Theta \in V$ solution of (A.2) and $\|\Theta\| \leq \frac{1}{C} \|F\|_{V'}$, where $V^{'}$ denotes the dual space of V.

The proof is based on the Riesz representation theorem and we remaind e.g. to (65) for it.

Under the hypotesis of the previous Theorem if a is simmetric, i.e.

$$a(\Theta, w) = a(w, \Theta),\ \forall \Theta, w \in V,$$

then it defines a scalar product on V and (A.2) is equivalent to the following *minimization problem*:

$$\text{find } \Theta \in V \text{ s.t. } J(\Theta) \leq J(w), \qquad \forall w \in V,$$

where $J(\Theta) := \frac{1}{2}a(\Theta, \Theta) - F(\Theta)$.

In the more general case in which $W \neq V$ the following Theorem gives necessary and sufficient conditions for well-posedness (*Nečas, 1962*):

Theorem A.2.2 *Let V and W be two (real) Hilbert spaces, endowed with norms $\|\cdot\|_V$ and $\|\cdot\|_W$ respectively. Assume that there exist two positive constants $\gamma > 0$ and $C > 0$ s.t. the bilinear form $a : V \times V \to \mathbb{R}$ satisfies*

$$|a(\Theta, w)| \leq \gamma \|\Theta\|_V \|w\|_W,\ \forall\ \Theta \in V,\ w \in W,\ \textit{(a is continuous)},$$

$$\left.\begin{array}{l} \sup_{w \in W, w \neq 0} \frac{a(\Theta, w)}{\|w\|_W} \geq C \|\Theta\|_V,\ \forall\ \Theta \in V, \\ \sup_{\Theta \in V} a(\Theta, w) > 0,\ \forall w \in W,\ w \neq 0. \end{array}\right\} \textit{(a is weakly coercive)}.$$

Then, for any $F \in W'$, there exists a unique $\Theta \in V$ solution of (A.2) and $\|\Theta\|_V \leq \frac{1}{C}\|F\|_{V'}$.

The proof is similar to that of Lax-Milgram lemma and uses the Riesz representation theorem. We remaind e.g. to (65) for it.

A.3 Discretization methods

Let $h > 0$ identify the mesh size of Ω_h, discetizazion of Ω, and consider the families of subspaces of V $\{V_h\}_{h>0}$ and of W $\{W_h\}_{h>0}$. Assume that $\forall v \in V, \inf_{v_h \in V_h} \|v - v_h\| \to 0$, as $h \to 0$. This is possible e.g. if $V_h = X_h^r$, i.e. using a finite elements space, where $X_h^r := \left\{v_h \in C^0(\bar{\Omega}) :\ v_h\ |_{K_j} \in \mathbb{P}_r,\ j = 1, \ldots, N_e l\right\}$ and $\mathbb{P}_r$ denotes the set of polynomials of degree $r \geq 1$ (see e.g. (64)).

A.3.1 Galerkin Method

The (standard) Galerkin approximation to (A.2) is:

$$\text{given } F \in V', \text{ find } \Theta_h \in V_h \text{ s.t. } a(\Theta_h, w_h) = F(w_h), \qquad \forall w_h \in V_h. \tag{A.3}$$

If $\{\phi_j\}_{j=1,\dots,N_h}$, $N_h = dim(V_h)$, is a basis of V_h, then, writing $\Theta_h(\mathbf{x}) = \sum_{i=1}^{N_h} \Theta_i \phi_i(\mathbf{x})$, (A.3) is equivalent to the following N_h-dimensional linear system:

$$A\boldsymbol{\Theta} = \mathbf{F},$$

with $\boldsymbol{\Theta} = (\Theta_j)_j$, $\mathbf{F} = (F(\phi_j))_j$ and the *stifness matrix* $A_{ij} = a(\phi_j, \phi_i)$, $i, j = 1, \dots, N_h$.

Theorem A.3.1 *Under the assumption of Theorem A.2.1 there exists a unique* $\Theta_h \in V_h$ *solution of (A.3) and* $\|\Theta_h\|_V \leq \frac{1}{C} \|F\|_{V'}$ *(stability).*

Moreover, if Θ *is the solution of the countinuos variational problem (A.2), then*

$$\|\Theta - \Theta_h\|_V \leq \frac{\gamma}{C} \inf_{v_h \in V_h} \|\Theta - v_h\|_V \ \ (Cea\ Lemma).$$

This implies the convergence of Θ_h *to* Θ *as* $h \to 0$.

The proof is simple and uses Lax-Milgram lemma. We refer e.g. to (65) for it. Observe that it states that consistency and stability implies convergence. It is important to note that the convergence depends upon the approximation properties of the family $\{V_h\}_h$, and $\inf_{v_h \in V_h} \|\Theta - v_h\|_V$ is usually estimated with the polinomial interpolation error. This leads to the following estimate (*a priori error analysis*)

$$\|\Theta - \Theta_h\|_V \leq \tilde{C} h^{l+1} |\Theta|_{H^{l+1}(\Omega)},$$

$$\|\Theta - \Theta_h\|_V \leq \tilde{C} h^{l} \|\Theta\|_{H^{l+1}(\Omega)},$$

$l = min(r, s-1)$, where r denotes the degree of the interpolating polynomials of the finite element space V_h and s depends on the regularity of Θ ($\Theta \in H^s(\Omega)$), i.e. on the choice of the space V (64). These convergence results are *optimal* in the H^1-norm, i.e. they provide the highest possible rate of convergence in the H^1-norm allowed by the polynomial degree r. The term $\tilde{C}$ plays a central role, because it is a measure of Galerkin method's precision: if it is large, the corresponding solution Θ_h could be inaccurate: in diffusion-convection-reaction problem, when convection or reaction are dominant with respect to diffusion, the Galerkin solution presents spurious oscillations, also when the analytical solution is monothonic.

A.3.2 Petrov-Galerkin (or non-Standard Galerkin) Method

The Petrov-Galerkin approximation to (A.2) is:

$$\text{find } \Theta_h \in V_h \text{ s.t. } a_h(\Theta_h, w_h) = F_h(w_h), \qquad \forall w_h \in W_h, \tag{A.4}$$

where $\{V_h\}_h$ and $\{W_h\}_h$ are two families of finite dimensional spaces s.t. $W_h \neq V_h$ and $dim(W_h) = dim(V_h) = N_h,\ \forall h > 0$, with norms $\|\cdot\|_{V_h}$ and $\|\cdot\|_{W_h}$ respectively (if $V_h \subset V$ and $W_h \subset W$ then $\|\cdot\|_{V_h} = \|\cdot\|_V$ and $\|\cdot\|_{W_h} = \|\cdot\|_W$). Define $V(h) = V + V_h$ and assume that (24) that there exists a norm $|\cdot|_{V(h)}$ s.t. $|\Theta_h|_{V(h)} = |\Theta_h|_{V_h}$ for $\Theta_h \in V_h$ and $|\Theta|_{V(h)} \leq c\,|\Theta_h|_V$ for all $\Theta \in V$.

If $a_h : V_h \times W_h \to \mathbb{R}$ and $F_h :\ W_h \to \mathbb{R}$ are apprximations to a and F respectively (possibly coinciding with them), then (A.4) can be seen as an approximation of (A.2). Observe that W and V need not be necessarily different.

The analysis of stability and convergence is a consequence of the following Theorem, due to Babuška (4).

Theorem A.3.2 *Under the assumption of Theorem A.2.2, suppose further that $F_h : W_h \to \mathbb{R}$ is a linear map and that $a_h : V_h \times W_h \to \mathbb{R}$ is a bilinear form s.t. there exists a constant $C_h > 0$ s.t.*

$$\sup_{w_h \in W_h, w_h \neq 0} \frac{a_h(\Theta_h, w_h)}{\|w_h\|_{W_h}} \geq C_h \|\Theta_h\|_{V_h}\ ,\ \forall\ \Theta_h \in V_h,$$

$$\sup_{\Theta_h \in V_h} a_h(\Theta_h, w_h) > 0,\ \forall w_h \in W_h,\ w_h \neq 0.$$

Then there exists a unique $\Theta_h \in V_h$ solution of (A.4) and $\|\Theta_h\|_{V_h} \leq \frac{1}{C_h} \sup_{w_h \in W_h, w_h \neq 0} \frac{F_h(w_h)}{\|w_h\|_{W_h}}$ (stability).

Moreover, if Θ is the solution of the countinuos variational problem (A.2), and $V_h \subset V$ and $W_h \subset W$ then

$$|\Theta - \Theta_h|_V \leq \inf_{v_h \in V_h} \left[\left(1 + \frac{\gamma}{C_h}\right) |\Theta - v_h|_V + \frac{1}{C_h} \sup_{w_h \in W_h, w_h \neq 0} \frac{|a(v_h, w_h) - a_h(v_h, w_h)|}{\|w_h\|_W} \right] + \frac{1}{C_h} \sup_{w_h \in W_h, w_h \neq 0} \frac{|F(w_h) - F_h(w_h)|}{\|w_h\|_W}, \tag{A.5}$$

For the proof see e.g. (65), (24).

Now we give an important definition.

Definition A.3.1 *(Consistency) Let Θ be the solution of (A.2); define the truncation error of a Petrov Galerkin method as*

$$\tau_h(\Theta) = \sup_{w_h \in W_h, w_h \neq 0} \frac{|a_h(\Theta, w_h) - F_h(w_h)|}{\|w_h\|}.$$

Then a method is consistent if $\lim_{h \to 0} \tau_h(\Theta) = 0$. Moreover it is strongly consistent if $\tau_h(\Theta) \equiv 0,\ \forall h > 0$.

It is possible to state an algebraic problem equivalent to (A.4): let $\{\phi_i\}_{i=1,\dots,N_h}$ and $\{\psi_j\}_{j=1,\dots,N_h}$ be basis of V_h and W_h respectively, then, writing $\Theta_h(\mathbf{x}) = \sum_{i=1}^{N_h} \Theta_i \phi_i(\mathbf{x})$, we obtain

$$A\boldsymbol{\Theta} = \mathbf{F},$$

with $\boldsymbol{\Theta} = (\Theta_i)_i$, $\mathbf{F} = (F_h(\psi_j))_j$ and the *stifness matrix* $A_{ji} = a_h(\phi_i, \psi_j)$, $i, j = 1, \dots, N_h$.

Conditions of Theorem A.3.2 on a_h can be interpreted as conditions on A, i.e., $ker(A) = \{0\}$ and $rank(A) = dim W_h$.

A.3.3 Generalized (or Standard) Galerkin Method

The Generalized Galerkin approximation to (A.2) is particular case of the Petrov Galerkin one (A.4):

$$\text{find } \Theta_h \in V_h \text{ s.t. } a_h(\Theta_h, w_h) = F_h(w_h), \qquad \forall w_h \in V_h, \tag{A.6}$$

where $\{V_h\}_h$ is a family of finite dimensional subspaces of V, $\forall h > 0$. If the bilinear form $a_h : V_h \times V_h \to \mathbb{R}$ and the linear operator $F_h : V_h \to \mathbb{R}$ are apprximations to a and F respectively, then the following results hold (for their proofs cfr. e.g. (65)).

Theorem A.3.3 *((First) Strang Lemma) Under the assumption of Theorem A.2.1, suppose further that F_h is a linear map and that a_h is uniformly coercive over $V_h \times V_h$, i.e. there exists a constant $C^* > 0$ s.t. $\forall h > 0$*

$$a_h(w_h, w_h) \geq C^* \|w_h\|_V^2, \ \forall\ w_h \in V_h.$$

Then there exists a unique $\Theta_h \in V_h$ solution of (A.6) and $\|\Theta_h\|_V \leq \frac{1}{C^*} \sup_{w_h \in V_h, w_h \neq 0} \frac{F_h(w_h)}{\|w_h\|}$ (stability).

Moreover, if Θ is the solution of the countinuos variational problem (A.2), then

$$|\Theta - \Theta_h|_V \leq \inf_{v_h \in V_h} \left[\left(1 + \tfrac{\gamma}{C^*}\right) |\Theta - v_h|_V + \tfrac{1}{C^*} \sup_{w_h \in V_h, w_h \neq 0} \tfrac{|a(v_h, w_h) - a_h(v_h, w_h)|}{\|w_h\|} \right] + \tfrac{1}{C^*} \sup_{w_h \in W_h, w_h \neq 0} \tfrac{|F(w_h) - F_h(w_h)|}{\|w_h\|}. \tag{A.7}$$

Proposition A.3.1 *Under the assumptions of Theorem A.3.3, suppose further that a_h is defined at (Θ, v_h), where Θ is the solution of (A.2) and $v_h \in V_h$ and the exists a constant $\gamma^* > 0$ s.t. a_h satisfies*

$$|a_h(\Theta - w_h, v_h)| \leq \gamma^* \|\Theta - w_h\|_V \|v_h\|_W, \ \forall\ w_h, v_h \in V_h,$$

uniformly with respect to $h > 0$. Then the following convergence estimate holds:

$$|\Theta - \Theta_h|_V \leq \left(1 + \frac{\gamma^*}{C^*}\right) \inf_{w_h \in V_h} |\Theta - w_h|_V + \frac{1}{C^*} \sup_{v_h \in V_h, v_h \neq 0} \frac{|a_h(\Theta, v_h) - F_h(v_h)|}{\|v_h\|}. \tag{A.8}$$

A.4 Mixed (or constained) variational problems

A.4.1 Infinite dimensional variational problem

For particular kind of problems (e.g. Stokes problem), it could be convenient to consider a *mixed formulation*, which is an alternative to Theorems A.2.2 and A.3.2. In this section we present only main results concerning constained problems. For more details cfr. e.g. (24, 65). Let X and M be two real Hilbert spaces with norms $\|\cdot\|_X$ and $\|\cdot\|_M$ respectively and dual spaces $X^{'}$ and $M^{'}$. Consider two bilinear forms

$$a: \ X \times X \to \mathbb{R}, \ b: \ X \times M \to \mathbb{R},$$

such that

$$\begin{aligned} |a(w,v)| &\leq \gamma \|w\|_X \|v\|_X , \\ |b(w,\mu)| &\leq \delta \|w\|_X \|\mu\|_M . \end{aligned} \tag{A.9}$$

Consider the following problem:

$$\begin{cases} \text{find } (u,\eta) \in X \times M \text{ s.t.} \\ a(u,v) + b(v,\eta) = \langle l, v \rangle , \qquad \forall v \in X, \\ b(u,\mu) = \langle \sigma, \mu \rangle , \qquad \forall \mu \in M, \end{cases} \tag{A.10}$$

where $l \in X^{'}$, $\sigma \in M^{'}$ and $\langle \cdot, \cdot \rangle$ denotes the duality pairing between $X^{'}$ and X or $M^{'}$ and M.

Consider two linear continuous operators $\mathcal{A}: \ X \to X^{'}$ and $\mathcal{B}: \ X \to M^{'}$ s.t. $\langle \mathcal{A}w, v \rangle = a(w,v)$, $\forall w, v \in X$ and $\langle \mathcal{B}v, \mu \rangle = b(v,\mu)$, $\forall v \in X, \mu \in M$; with this notation the adjoint of $\mathcal{B}$ is $\mathcal{B}^*: \ M \to X^{'}$ (i.e. s.t. $\langle \mathcal{B}^*\mu, v \rangle = \langle \mathcal{B}v, \mu \rangle$, $\forall v \in X, \mu \in M$). Then (A.10) is equivalent to

$$\begin{cases} \text{find } (u,\eta) \in X \times M \text{ s.t.} \\ \mathcal{A}u + \mathcal{B}^*\eta = l, \qquad in \quad X^{'}, \\ \mathcal{B}u = \sigma, \qquad in \quad M^{'}. \end{cases} \tag{A.11}$$

Let us now introduce the linear operator $\phi: \ X \times M \to X^{'} \times M^{'}$, $\phi(v,\mu) := (\mathcal{A}v + \mathcal{B}^*\mu, \mathcal{B}v)$. The problem (A.11) is *well-posed* if ϕ is an isomorphism. The aim is to find necessary and sufficient conditions (32).

Define the affine manifold

$$X^{\sigma} = \{ v \in X \ s.t. \ b(v,\mu) = \langle \sigma, \mu \rangle , \ \forall \mu \in M \} .$$

Then $X^0 = ker(\mathcal{B})$, which is a closed subset of X.

Associate now to problem (A.10) the following one

$$\text{find } u \in X^{\sigma} \text{ s.t. } a(u,v) = \langle l, v \rangle , \qquad \forall v \in X^0; \tag{A.12}$$

if (u, η) is a solution of (A.10), then it is a solution of (A.12). The aim is to introduce suitable conditions which garantees that the converse is also true, and that the solution to (A.12) does exist and is unique.

Theorem A.4.1 *Assume that the bilinear form a satisfies (A.9) and is coercive on X^0, i.e. there exists a constant $C > 0$ s.t.*

$$a(v, v) \geq C \|v\|_X^2 , \ \forall v \in X^0.$$

Moreover suppose that the bilinear form b satisfies (A.9) and the compatibility condition

$$\text{there exists } \beta^* > 0 \text{ s.t. } \forall \mu \in M \exists v \in X, v \neq 0 \text{: } b(v, \mu) \geq \beta^* \|v\|_X \|\mu\|_M . \tag{A.13}$$

Then for each $l \in X^{'}$, $\sigma \in M^{'}$ there exists a unique solution u of (A.12) and a unique $\eta \in M$ s.t. (u, η) is the unique solution of (A.10). Moreover the map $(l, \sigma) \to (u, \eta)$ is an isomorphism from $X^{'} \times M^{'}$ into $X \times M$, and

$$\|u\|_X \leq \frac{1}{C} \left(\|l\|_{X'} + \frac{C + \gamma}{\beta^*} \|\sigma\|_{M'} \right),$$

$$\|\eta\|_M \leq \frac{1}{\beta^*} \left(\left(1 + \frac{\gamma}{C} \right) \|l\|_{X'} + \frac{\gamma(C + \gamma)}{C\beta^*} \|\sigma\|_{M'} \right).$$

For a proof see e.g. (65).

A.4.2 Finite dimensional variational problem

Consider now an approximation of (A.10): let X_h and M_h be finite dimensional subspaces of X and M respectively. Consider the following discete problem:

$$\begin{cases} \text{find } (u_h, \eta_h) \in X_h \times M_h \text{ s.t.} \\ a(u_h, v_h) + b(v_h, \eta_h) = \langle l, v_h \rangle , \qquad \forall v_h \in X_h, \\ b(u_h, \mu_h) = \langle \sigma, \mu_h \rangle , \qquad \forall \mu_h \in M_h; \end{cases} \tag{A.14}$$

define moreover the space

$$X_h^\sigma = \{ v_h \in X_h \text{ s.t. } b(v_h, \mu_h) = \langle \sigma, \mu_h \rangle , \ \forall \mu_h \in M_h \} .$$

Observe that $M_h \subset M$ does not imply that X_h^σ is a subspace of X^σ.

The discretization of problem (A.12) is the following one

$$\text{find } u_h \in X_h^\sigma \text{ s.t. } a(u_h, v_h) = \langle l, v_h \rangle , \qquad \forall v_h \in X_h^0; \tag{A.15}$$

if (u_h, η_h) is a solution of (A.14), then it is a solution of (A.15). As in the continuous case, the aim is to introduce suitable conditions which garantees that the converse is true and an analysis of convergence and stability.

Theorem A.4.2 *(Stability) Assume that the bilinear form a satisfies (A.9) and is coercive on X^0, i.e. there exists a constant $C_h > 0$ s.t.*

$$a(v_h, v_h) \geq C_h \|v_h\|_X^2 , \ \forall v_h \in X_h^0.$$

Moreover suppose that the bilinear form b satisfies (A.9) and the compatibility condition

$$\text{there exists } \beta_h > 0 \text{ s.t. } \forall \mu_h \in M_h \exists v_h \in X_h, v_h \neq 0 \text{: } b(v_h, \mu_h) \geq \beta_h \|v_h\|_X \|\mu_h\|_M . \tag{A.16}$$

Then for each $l \in X^{'}$, $\sigma \in M^{'}$ there exists a unique solution (u_h, η_h) of (A.14) s.t.

$$\|u_h\|_X \leq \frac{1}{C_h} \left(\|l\|_{X'} + \frac{C_h + \gamma}{\beta_h} \|\sigma\|_{M'} \right),$$

$$\|\eta_h\|_M \leq \frac{1}{\beta_h} \left(\left(1 + \frac{\gamma}{C_h} \right) \|l\|_{X'} + \frac{\gamma(C_h + \gamma)}{C_h \beta_h} \|\sigma\|_{M'} \right),$$

which are stability results in those cases in which both C_h and β_h are indipendent of h.

Observe that the *discrete compatibility condition* (A.16) is also called *inf-sup* of *Ladyzhenskaya-Babuška-Brezzi (LBB)* condition and can be written equivalently (32)

$$\text{there exists } \beta_h > 0 \text{ s.t. } \forall \mu_h \in M_h \exists v_h \in X_h, v_h \neq 0 \text{: } b(v_h, \mu_h) = \|\mu_h\|_M^2 , \ \|v_h\|_X \leq \frac{1}{\beta_h} \|\mu_h\|_M . \tag{A.17}$$

This condition it is necessary to achieve uniqueness of η_h. Observe that it can be written as:

$$\text{if } \mu_h \in M_h \text{ and } b(v_h, \mu_h) = 0, \forall v_h \in X_h, \text{ then } \mu_h = 0.$$

Thus, if the compatibility condition is not satisfied, there exists a *spurious* (or *parasitic*) *mode* $\mu_h^* \in M_h, \ \mu_h^* \neq 0$ s.t.

$$b(v_h, \mu_h^*) = 0, \ \forall v_h \in X_h.$$

This means that if (u_h, η_h) solves (A.14), also $(u_h, \eta_h + \lambda \mu_h^*)$ is a solution for every $\lambda \in \mathbb{R}$: uniqueness is lost and instabilities may be generated for the numerical method, as we will see for the Stokes problem.

Theorem A.4.3 *(Convergence) Let the assumptions of Theorems A.4.1 and A.4.2 be satisfied. Then the solutions (u, η) and (u_h, η_h) to (A.10) and (A.14) respectively satisfy the following error estimates*

$$\begin{aligned} &\|u - u_h\|_X \leq \left(1 + \tfrac{\gamma}{C_h} \right) \inf_{v_h \in X_h^\sigma} \|u - v_h\|_X + \tfrac{\delta}{C_h} \inf_{\mu_h \in M_h} \|\eta - \mu_h\|_M \\ &\|\eta - \eta_h\|_M \leq \tfrac{\gamma}{\beta_h} \left(1 + \tfrac{\gamma}{C_h} \right) \inf_{v_h \in X_h^\sigma} \|u - v_h\|_X + \left(1 + \tfrac{\delta}{\beta_h} + \tfrac{\gamma \delta}{C_h \beta_h} \right) \inf_{\mu_h \in M_h} \|\eta - \mu_h\|_M . \end{aligned} \tag{A.18}$$

Moreover the following estimate holds:

$$\inf_{v_h \in X_h^\sigma} \|u - v_h\|_X \leq \left(1 + \frac{\delta}{\beta_h}\right) \inf_{v_h \in X_h} \|u - v_h\|_X \tag{A.19}$$

For proofs see e.g (65).

Observe that to derive optimal error bounds, inequality constants must be independent of h. For the *infsup* condition the following lemma holds:

Lemma A.4.1 *(Fortin) The infsup condition (A.16) holds with a constant $\beta_h = \beta > 0$ independent of h iff there exists a linear continuous operator $\Pi_h : \ X \to X_h$ s.t.*

$$b(v - \Pi_h v, \mu_h) = 0, \ \forall \mu_h \in M_h, \ \forall v \in X$$

and

$$\|\Pi_h v\|_X \leq C \|v\|_X , \ \forall v \in X,$$

with $C > 0$ independent of h.

For a proof cfr. (32).

Appendix B

POD: comparison between finite and infinite dimensional formulations

In the literature are presented substantially two different formulations of POD:

1. the first one, in a Model Order Reduction (MOR) context, first discretizes (6.22) in space, for example using the FE method. Thus a dynamical system of type (5.4) is obtained and the POD reduction is applied to it starting from $\mathcal{X}$, through a Galerkin projection, obtaining (6.8). In this context the POD basis is $\{\mathbf{u}_1, \dots, \mathbf{u}_k\}$, the first k left singular vectors of the matrix of snapshots $\mathcal{X}$, and $U_k = [\mathbf{u}_1, \dots, \mathbf{u}_k]$, $\mathbf{u}_i \in \mathbb{R}^n$. Equivalently, the basis could be computed applying the method of snapshots (cfr. equation (6.3) and (6.4)).

2. The second method reduces directly (6.22), applying the POD-Galerkin method (equation (6.25)). The POD basis $\{\psi_1, \dots, \psi_k\}$, $\psi_i \in V$ could be computed applying the method of snapshots:

$$\psi_i = \frac{1}{\sqrt{\lambda_i}} \mathcal{Y}_N \mathbf{v}_i = \frac{1}{\sqrt{\lambda_i}} \sum_{j=0}^{N} \alpha_j v_i(j) y(t_j), \ i = 1, \dots, d,$$

   where $\mathbf{v}_i$ is such that $\mathcal{K}_N \mathbf{v}_i = \lambda_i \mathbf{v}_i$ (Remark 6.4.1). Thus ψ_i is a linear combination of snapshots $y(t_j)$.

In this section it is shown that the associated algebraic systems are equivalent.

B. POD: COMPARISON BETWEEN FINITE AND INFINITE DIMENSIONAL FORMULATIONS

B.1 Computation of the POD modes

First of all, given the parabolic problem (6.22), consider the snapshots ensamble

$$\mathcal{V} = \{y(t_1), \dots, y(t_N)\}.$$

From a numerical point of view these trajectories are only known in n discretization points: thus we can associate to $\mathcal{V}$ the discretization matrix

$$\mathcal{X} = \{\mathbf{x}(t_1), \dots, \mathbf{x}(t_N)\} \in \mathbb{R}^{n \times N},$$

s.t. vector $\mathbf{x}(t_i)$ can be thought as a space discretization of the corresponding continuous in time snapshot $y(t_i)$.

From a numerical point of view trajectories are known only on space discretization points: thus, also the POD basis ψ_i is known only on these points. In practice

$$\psi_i \approx \boldsymbol{\Psi_i} = \frac{1}{\sqrt{\lambda_i}} \sum_{j=0}^{N} \alpha_j v_i(j) \mathbf{x}(t_j), \ i = 1, \dots, d.$$

Thus we obtain

$$\begin{aligned} \mathcal{X}^T W \mathcal{X}^T D \mathbf{v}_i &= \lambda_i \mathbf{v}_i, \\ \boldsymbol{\Psi_i} &= \tfrac{1}{\sqrt{\lambda_i}} \mathcal{X} W \mathbf{v}_i \end{aligned} \tag{B.1}$$

which is the discrete method of snapshots (6.4), where W is the diagonal matrix of space quadrature weights.

Thus from a numerical point of view, the computation of the discrete POD basis $\{\mathbf{u}_1, \dots, \mathbf{u}_k\}$ is equivalent to the approximation of $\{\psi_1, \dots, \psi_k\}$ over spacial nodes ($\mathbf{u}_i = \boldsymbol{\Psi_i}$ for all $i = 1, \dots, k$).

B.2 Reduced models

MOR approach

First of all we discretize (6.22) using the FE method. Thus let $V_h \subset \mathcal{V}$ be a FE space with basis $\{\phi_h^i\}$, $i = 1, \dots, n$, $n \geq k$. The FE discretization of (6.22) is the following: given $f \in \mathcal{C}([0,T]; V)$, $y_{0_h} \in V_h$ we consider the nonlinear evolution problem: find $y_h(t) \in V_h$ s.t. $\forall \phi_h \in V_h$, *a.e.* $t \in (0,T]$

$$\begin{aligned} \tfrac{d}{dt}(y_h(t), \phi_h)_V + a(y_h(t), \phi_h) + \langle b(y_h(t), y_h(t)) + R y_h(t), \phi_h \rangle_{V',V} &= (f(t), \phi_h)_V, \\ y_h(0) &= y_{0_h} \quad \text{in } V_h. \end{aligned} \tag{B.2}$$

Discretizing in time (B.2), using the backward Euler method, we should find a sequence $\{Z_l\}_{l=0}^m \in V^n$ s.t.

$$(\tfrac{Z_l - Z_{l-1}}{\delta\tau_l}, \phi_h)_V + a(Z_l, \phi_h) + \langle b(Z_l, Z_l) + RZ_l, \phi_h\rangle_{V',V} = (f(\tau_l), \phi_h)_V, \qquad \forall \phi_h \in V^n \tag{B.3}$$

Writing $Z_l = \sum_{i=1}^n z_l(i)\phi_h^i$, for all $l = 0, \dots, m$, $\mathbf{z}_l \in \mathbb{R}^n$, the finite dimensional system (B.3) is equivalent to the following algebric problem

$$\begin{aligned} M\tfrac{\mathbf{z}_l - \mathbf{z}_{l-1}}{\delta\tau_l} + K\mathbf{z}_l + B(\mathbf{z}_l)\mathbf{z}_l + R\mathbf{z}_l &= F(\tau_l) \\ \mathbf{z}_0 &= \mathbf{y}_0, \end{aligned} \tag{B.4}$$

where $M_{ij} := (\phi_h^i, \phi_h^j)_V$, $K_{ij} := a(\phi_h^j, \phi_h^i)_V$, $(B(\mathbf{z}_l))_{ij} = \left\langle \sum_{k=1}^n z_l^k b(\phi_h^i, \phi_h^k), \phi_h^j \right\rangle_{V',V}$, $R_{ij} = \left\langle R\phi_h^i, \phi_h^j \right\rangle_{V',V}$ and $F(\tau_l) = (f(\tau_l), \phi_h^j)_V$, $i, j = 1, \dots, n$.

Let $U_k = [\mathbf{u}_1, \dots, \mathbf{u}_k] \in \mathbb{R}^{n\times k}$ be the k-th POD basis (cfr. Corollary 6.2.2), for a fixed $k \leq n$. Applying POD, as descrived in section 6.2, corresponds to consider the following reduced system

$$\begin{aligned} M_k\tfrac{\mathbf{a}_l - \mathbf{a}_{l-1}}{\delta\tau_l} + K_k\mathbf{a}_l + B_k(\hat{\boldsymbol{z}}_l)\mathbf{a}_l + R_k\mathbf{a}_l &= F_k(\tau_l) \\ \mathbf{a}_0 &= U_k\mathbf{y}_0, \\ \hat{\boldsymbol{z}}_l &= U_k\mathbf{a}_l, \end{aligned} \tag{B.5}$$

where $\hat{\boldsymbol{z}}_l \approx \mathbf{z}_l$ and

$$\begin{aligned} M_k &= U_k^* M U_k, \\ K_k &= U_k^* K U_k, \\ B_k(\hat{\boldsymbol{z}}_l) &= U_k^* B(\hat{\boldsymbol{z}}_l) U_k, \\ R_k &= U_k^* R U_k, \\ F_k(\tau_l) &= U_k^* F(\tau_l). \end{aligned} \tag{B.6}$$

ROM approach

Observe that in (6.25), $\hat{y}(t) \in V^k$, thus it is a linear combination of $\{\psi_1, \dots, \psi_k\}$. Then (6.25) is equivalent to a system of k ODEs, and not n, like the one we would obtain discretizing (6.22) with the finite element method (equation (B.2)). Thus we are solving *directly* a reduced problem. This is a difference with respect to the MOR approach presented in section 6.2, which first compute the unreduced ODE system.

The time discretization of (6.25) is given by (6.28). Considering $Y_l = \sum_{i=1}^k y_l(k)\psi_k$, $\mathbf{y}_l \in \mathbb{R}^k$, the corresponding algebraic problem is the following

$$\tilde{M}_k\tfrac{\mathbf{y}_l - \mathbf{y}_{l-1}}{\delta\tau_l} + \tilde{K}_k\mathbf{y}_l + \tilde{B}_k(\mathbf{y}_l)\mathbf{y}_l + \tilde{R}_k\mathbf{y}_l = \tilde{F}_k(\tau_l) \tag{B.7}$$

where $\tilde{M}_{k_{ij}} := (\psi^i, \psi^j)_V$, $\tilde{K}_{k_{ij}} := a(\psi^j, \psi^i)_V$, $(\tilde{B}_k(\mathbf{y}_l))_{ij} = \left\langle \sum_{s=1}^k y_l^s b(\psi^i, \psi^s), \psi^j \right\rangle_{V',V}$, $\tilde{R}_{k_{ij}} = \left\langle R\psi^i, \psi^j \right\rangle_{V',V}$ and $\tilde{F}_{k_j}(\tau_l) = (f(\tau_l), \psi^j)_V$, $i, j = 1, \ldots, k$.

As noticed before, in general $\{\psi_i\}$ is only known on discretization points: considering e.g. the FE space V_h we suppose that $\psi_i = \sum_{l=1}^n \Psi_i^l \phi_h^l$, using the nodal values $\boldsymbol{\Psi_i} \in \mathbb{R}^n$ previously introduced. Then, defining defining $\Psi = [\boldsymbol{\Psi_1}, \ldots, \boldsymbol{\Psi_k}] \in \mathbb{R}^{n \times k}$, the matrices of system (B.7) can be computed in the following way:

$$\begin{aligned} \tilde{M}_k &= \Psi^* M \Psi; \\ \tilde{K}_k &= \Psi^* K \Psi; \\ \tilde{B}_k(y_l) &= \Psi^* B(\Psi \mathbf{y}_l) \Psi; \\ \tilde{R}_{k_{ij}} &= \Psi^* R \Psi; \\ \tilde{F}_{k_j}(\tau_l) &= \Psi^* F(\tau_l). \end{aligned} \tag{B.8}$$

In conclusion, applying both techniques from a numerical point of view we hand up with equivalent reduced albegraic system.

Bibliography

Part I: Parabolic Models

[1] D. Acheson, "'Elementary fluid dynamics"', Oxford University Press, 1990

[2] U.Albocher, A.A. Oberai, P. E. Barbone, I. Harari, "Adjoint-weighted equation for inverse problems of incompressible plane-stress elasticity", *Comp.Meth.Appl.Mech.Eng.* **198** (2009) 2412-2420.

[3] C. Baiocchi, F.Brezzi, L.P. Franca, "Virtual bubbles and Galerkin-least squares type methods"', *Comp.Meth.Appl.Mech.Eng.* **105** (1993) 125-141

[4] A.K. Aziz, I. Babuška, "The Mathematical Foundations of the Finite Element Method with Application to Partial Differential Equations", Academic Press, New York, 1972.

[5] G.Batchelor, "'An introduction to Fluid Dynamics"', Cambridge University Press, 2000

[6] P.E. Barbone, A.A. Oberai, I. Harari, "Adjoint-weighted variational formulation for direct solution of inverse heat conduction problem", *Invers. Probl.* **23** (2007) 2325-2342.

[7] G. Biswas, M. Breuer, F. Durst "Backward-Facing step flow for various expansion ratios at low and moderate Reynolds numbers", *Journal of Fluids Engineering*, **126** (2004), 362-375.

[8] P.B. Bochev, M.D. Gunzburger "Least-squares finite element methods", Springer, 2009

[9] H.Brezis, "Analisi funzionale", Liguori Editore, 1986

BIBLIOGRAPHY

[10] F.Brezzi, D.Marini, A.Russo, "'On the choice of a stabilizing subgrid for convection-diffusion problems"', *Comp.Meth.Appl.Mech.Eng.* **194** (2005) 127-148

[11] F.Brezzi, M.O.Bristeau, L.P. Franca, M.Mallet and G.Rogè, A.Russo, "A relationship between stabilized finite element methods and the Galerkin method with bubble functions"', *Comp.Meth.Appl.Mech.Eng.* **96** (1992) 117-129

[12] F.Brezzi, L.P. Franca, T.J.R. Hughes, A. Russo "$b = \int g$"', *Comp.Meth.Appl.Mech.Eng.* **145** (1997) 329-339

[13] F.Brezzi, J. Douglas "Stabilized Mixed methods for the Stokes problem"', *Numer. Math.* **53** (1988) 225-235

[14] F.Brezzi, L.P.Franca, A.Russo, "Further considerations on residual-free bubbles for advective-diffusive equations", *Comp.Meth.Appl.Mech.Eng.* **166** (1998) 25-33

[15] P.G.Ciarlet "The finite element method for elliptic problems", North-Holland, 1980

[16] R. Codina, "Comparison of some finite element methods for solving the diffusion-convection-reaction equation"', *Comp.Meth.Appl.Mech.Eng.* **156** (1998) 185-210

[17] R. Codina, "On stabilized finite element methods for linear systems of convection-diffusion-reaction equations"', *Comp.Meth.Appl.Mech.Eng.* **188** (2000) 61-82

[18] R. Codina, J.Blasco "A finite element formulation for the Stokes problem allowing equal velocity-pressure interpolation"', *Comp.Meth.Appl.Mech.Eng.* **143** (1997) 373-391

[19] P.Davis "Interpolation and Approximation", Dover Publications, 1963

[20] G. Deolmi, F. Marcuzzi, M. Morandi Cecchi "The Best-Approximation Weighted-Residuals method for the steady convection diffusion reaction problem", *Mathematics and Computers in Simulation* **82** (2011) 144-162.

[21] C. R. Dohrmann, P. B. Bochev, "A stabilized Finite element method for the Stokes problem based on polynomial pressure projections", *Int. J. Numer. Methods Fluids* **46** (2004) 183-201.

[22] J.Douglas and J.Wang "An absolutely stabilized finite element method for the Stokes problem", *Math.Comput.* **52** (1989) 495-508

[23] H. Elman, D.Silvester, A.Wathen, "'Finite Elements and Fast Iterative Solvers"', Oxford Science Publications, 2005

[24] A.Ern, J.L. Guermond "Theory and Practise of Finite Elements", Springer, 2004

[25] L.Formaggia, F.Saleri, A.Veneziani "Applicazioni ed esercizi di modellistica numerica per problemi differenziali", Springer, 2005

[26] L.P. Franca, T.E. Tezduyar and A. Masud (Eds.) "Finite Element Methods: 1970's and beyond (dedicated to the work of T.J.R. Hughes)", CIMNE, Barcelona, 2004

[27] L.P. Franca, G. Hauke, A. Masud "Revisiting stabilized finite element methods for the advective-diffusive equation"', *Comp.Meth.Appl.Mech.Eng.* **195** (2006) 1560-1572

[28] L.P. Franca, S.L. Frey, T.J.R. Hughes "Stabilized finite element methods: I. Application to the advective-diffusive model"', *Comp.Meth.Appl.Mech.Eng.* **95** (1992) 253-276

[29] L.P. Franca, A.Russo "Deriving Upwinding, Mass Lumping and Selective Reduced Integration by Residual-Free Bubbles"', *Appl.Math.Lett.* **9** (1996) 83-88

[30] L.P. Franca, C.Farhat "Bubble functions prompt unusual stabilized finite element methods"', *Comp.Meth.Appl.Mech.Eng.* **123** (1995) 299-308

[31] L.P.Franca, S.L.Frey and T.J.R. Hughes "Stabilized finite element methods I. Application to the advective-diffusive model", *Comp.Meth.Appl.Mech.Eng.* **95** (1992) 253-276

[32] V. Girault, P.A. Raviart "Finite Element Methods for Navier-Stokes Equations", Springer, 1986

[33] M.D. Greenberg "Application of Green's functions in Science and Engineering", Prentice-Hall, 1971

[34] J.L.Guermond "'Stabilization of Galerkin approximations of transport equations by subgrid modeling"', M2AN **33**, n.6, (1999) 1293-1316

[35] I.Harari, T.J.R. Hughes "Stabilized finite-element methods for steady advection-diffusion with production", *Comp.Meth.Appl.Mech.Eng.* **115** (1-2) (1994) 165-191

[36] P.Hansbo "Adaptivity and streamline diffusion procedures in the finite element method", PhD Thesis, Chalmers University of Tecnology, Göteborg, 1989

[37] G.Hauke, A.G.Olivares "Variational subgrid scale formulations for the advection-diffusion-reaction equation", *Comp.Meth.Appl.Mech.Eng.* **190** (2001) 6847-6865

BIBLIOGRAPHY

[38] E. Hopf, "Uber die Anfangswertaufgabe fiir die hydrodynamischen Grundgleichungen", *Math. Nachr.* **4** (1951) 213-231.

[39] T.J.R. Hughes "Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods", *Comp.Meth.Appl.Mech.Eng.* **127** (1995) 387-401

[40] T.J.R.Hughes, A.N.Brooks "A multidemensional upwind scheme with no crosswind diffusion", *Finite Elements Methods for Convection Dominated Flows*, T.J.R.Hughes ed., The American Society of Mechanical Engineers (1979) 19-35

[41] T.J.R.Hughes, "Recent progress in the development and understanding of SUPG methods with special reference to the compressible Euler and Navier-Stokes equations", *Int.J.Num.Meth. Fluids* **7** (1987) 1261-1275

[42] T.J.R.Hughes, M.Mallet and A.Mizukami "A new finite element formulation for computational fluid dynamics: II. Beyond SUPG", *Comp.Meth.Appl.Mech.Eng.* **54** (1986) 341-355

[43] O.A. Ladyzhenskaya, "Solution 'in the large' of the nonstationary boundary value problem for the Navier-Stokes system with two space variables", *Comm. Pure Appl. Math.* **12** (1959) 427-433.

[44] J. Leray, "Essai sur les mouvements plans d'un liquide visqueux que limitent des parois", *J. Math. Pures Appl.* **13** (1934) 331-418. equations de Navier-Stokes en dimension 2", *C.RA.S. Paris* **248** (1959) 3519-3521.

[45] T.J.R.Hughes, M.Mallet "A new finite element formulation for computational fluid dynamics: III. The generalized streamline operator for multidimensional advective-diffusive systems", *Comp.Meth.Appl.Mech.Eng.* **58** (1986) 305-328

[46] T.J.R.Hughes, M.Mallet "A new finite element formulation for computational fluid dynamics: IV. A discontinuity-capturing operator for multidimensional advective-diffusive systems", *Comp.Meth.Appl.Mech.Eng.* **58** (1986) 329-336

[47] T.J.R.Hughes, L.P.Franca and G.M. Hulbert "A new finite element formulation for computational fluid dynamics: VII. The Galerkin/Least-Squares method for advective-diffusive equations", *Comp.Meth.Appl.Mech.Eng.* **73** (1989) 173-189

[48] T.J.R.Hughes, A.N.Brooks "Streamline Upwind Petrov-Galerkin Formulations for Convection Dominated Flows with particular emphasis on the Incompressible Navier-Stokes Equations", *Comp.Meth.Appl.Mech.Eng.* **32** (1982) 199-259

[49] T.J.R.Hughes, A.N.Brooks "A theoretical framework Petrov-Galerkin methods with discontinuous weighting functions: application to the streamline-upwind procedure", *Finite Elements in Fluids, Vol. 4* R.H. Gallagher, D.H. Norrie, J.T. Oden and O.C. Zienkiewicz eds., (1982) 47-65

[50] T.J.R. Hughes, G.R. Feijóo, L. Mazzei, J.B. Quincy, "The variational multiscale method - a paradigm for computational mechanics", *Comp.Meth.Appl.Mech.Eng.* **166** (1998) 3-24

[51] T.J.R. Hughes, G.R. Franca, L. Balestra "A new finite element formulation for computational fluid dynamics: V. Circumventing the Babŭska-Brezzi condition: a stable Petrov-Galerkin formulation of the Stokes problem accomodating equal-order interpolations", *Comp.Meth.Appl.Mech.Eng.* **59** (1986) 85-99

[52] T.J.R. Hughes, G.R. Franca "A new finite element formulation for computational fluid dynamics: VII. The Stokes problem with various well-posed boundary conditions: symmetric formulations that converge for all velocity/pressure spaces", *Comp.Meth.Appl.Mech.Eng.* **65** (1987) 85-96

[53] C. Johnson, A.H. Schatz, L.B. Wahlbin "Crosswind Smear and Pointwise Errors in the Streamline Diffusion Finite Elements Methods", *Math.Comp.* **49** (1987) 25-38

[54] C. Johnson, U. Nävert, J.Pitkäranta "Finite Elements Methods for linear Hyperbolic Problems", *Comp.Meth.Appl.Mech.Eng.* **45** (1984) 285-312

[55] N.Kopteva "How accurate is the streamline-diffusion FEM inside characteristic (boundary and interior) layers?", *Comp.Meth.Appl.Mech.Eng.* **193** (2004) 4875-4889

[56] J.-L. Lions and G. Prodi, "Un theorems d'existence et unicite dans les equations de Navier-Stokes en dimension 2", *C.RA.S. Paris* **248** (1959) 3519-3521.

[57] G.I.Marchuk, V.I.Agoshkov, V.P.Shutyaev "Adjoint equations and perturbation algorithms", CRC Press, 1996

[58] P.A. Markowich, "'Applied Partial Differential Equations: A visual approach"', Springer, 2007

[59] F.Marcuzzi, "Adaptivity in the Finite Element method", PhD Thesis, University of Padova, 2000, available at http://www.math.unipd.it/ marcuzzi/

BIBLIOGRAPHY

[60] F.Marcuzzi, M.Morandi-Cecchi, "Least squares FEM approximation and subgrid extraction for convection dominated problems", Proceedings of the MASCOT07-IMACS/ISGG Workshop, IAC-CNR, Rome, Italy

[61] F.Marcuzzi, M.Morandi-Cecchi, "The Best-Approximation Weighted-Residuals Method for Finite Element Approximations", Proceedings of the MASCOT08-IMACS/ISGG Workshop, IAC-CNR, Rome, Italy

[62] A. Mizukami, T.J.R. Hughes, "A Petrov-Galerkin Finite Element method for Convection-Dominated Flows: An Accurate Upwinding Technique for Satisfying the Maximum Principle", *Comp.Meth.Appl.Mech.Eng.* **50** (1985) 181-193

[63] A.A. Oberai, P.E. Barbone, I. Harari, "The adjoint weighted equation for steady advection in a compressible fluid", *Int. J. Numer. Meth. Fluids* **54** (2007) 683-693.

[64] A.Quarteroni, "Modellistica numerica per problemi differenziali", Springer, 2006

[65] A.Quarteroni, A.Valli, "Numerical approximation of Partial Differential Equations", Springer, 1994

[66] A.Quarteroni, R.Sacco, F.Saleri "Matematica Numerica", Springer, 2008

[67] H.G.Roos, M.Stynes, L.Tobiska "Numerical Methods for Singularly Perturbed Differential Equations. Convection-Diffusion and Flow Problems", Springer, 1996

[68] H.G.Roos "Stabilized FEM for convection-diffusion problems on layer-adapted meshes", *J. Comput. Math.* **27** (2-3) (2009) 266-279.

[69] R.Temam "Infinite-dimensional Dynamical Systems in Mechanics and Physics", Springer, 1988

[70] S.Turek "Efficient solvers for incompressible flow problems: an algorithmic and computational approach", Springer, 1999

[71] O.Zienkiewicz, K.Morgan "Finite Element Approximations", Wiley, 1984

[72] G.Zhou "How accurate is the streamline diffusion finite element method?", *Math. Comp.* **66** (217) (1997) 31-44.

Part II: Reduced Order Modeling

[73] Afanasiev K., Hinze M., "Adaptive control of a wake flow using Proper Orthogonal Decomposition", *Lect. Notes Pure Appl. Math.*, **216** (2001), 317-332.

[74] A.C. Antoulas, D.C. Sorensen, "Approximation of large-scale dynamical systems: an overview", http://www-ece.rice.edu/ aca/Pub.html (2001)

[75] A.C. Antoulas, D.C. Sorensen, S. Gugercin "A survey of model reduction methods for large-scale systems", "Structured Matrices in Operator Theory, Numerical Analysis, Control, Signal and Image Processing", Contemporary Mathematics, AMS publications (2001)

[76] A.C. Antoulas, "Approximation of large-scale dynamical systems", Siam, 2005

[77] Astrid, P. "Reduction of process Simulation Models: A Proper Orthogonal Decomposition", Ph.D. Thesis, http://alexandria.tue.nl/extra2/200413220.pdf, 2004

[78] Z. Bai, P. Feldmann, R.W. Freund, "Stable and passive reduced-order models based on partial Padé approximation via the Lanczos process"', Numerical Analysis Manuscript 97-3-10, Bell Laboratories, Murray Hill (1997)

[79] Banks H.T., Joyner M.L., Winchesky B., Winfree W.P., "Nondestructive evaluation using a reduced order computational methodology", *Inverse Problems*, **16** (2000), 1-17.

[80] M. Bergmann, L. Cordier, "Optimal control of the cylinder wake in the laminar regime by trust-region methods and POD reduced-order models", *Journal of Computational Physics*, **227** (2008), 7813-7840.

[81] M. Bergmann, L. Cordier, J.P. Brancher "Optimal rotary control of the cylinder wake using proper orthogonal decomposition reduced-order model", *Physics of fluids*, **17** (2005), 7813-7840.

[82] G. Biswas, M. Breuer, F. Durst "Backward-Facing step flow for various expansion ratios at low and moderate Reynolds numbers", *Journal of Fluids Engineering*, **126** (2004), 362-375.

[83] S. Chaturantabut, D.C. Sorensen, "Discrete Empirical Interpolation for Nonlinear Model Reduction", TR09-05, CAAM, Rice University, 2009.

BIBLIOGRAPHY

[84] S. Chaturantabut, D.C. Sorensen, "Application of POD and DEIM to Dimension Reduction of Nonlinear Miscible Viscous Fingering in Porous Media", TR09-25, CAAM, Rice University, 2009.

[85] M. Couplet, C. Basdevant, P. Sagaut "Calibrated reduced-order POD-Galerkin system for fluid flow modelling", *Journal of Computational Physics* **207** (2005), 192-220.

[86] E.A. Christensen, M. Brons, J.N. Sorensen "Evaluation of Proper Orthogonal Decomposition-based decomposition techniques applied to parameter-dependent nonturbolent flows",*SIAM J. Sci. Comput.* **21** (2000), 1419-1434.

[87] K. Fukunaga, "Introduction to statistical Recognition.", Accademic Press, 1990.

[88] J. Hahn, T.F. Edgar, "An improved method for nonlinear model reduction using balancing of empirical gramians", *Comput. Chemical Engrg.*, **26** (2002), 1379-1397.

[89] A. Iollo, S. Lanteri, J. Dèsidèri "Stability properties of POD-Galerkin approximations for the compressible Navier Stokes equations",*Theoret. Comput. Fluid Dynamics* **13** (2000), 377-396.

[90] K. Glover, "All Optimal Hankel-norm Approximations of Linear Multivariate Systems and their L^{∞}-error bounds"', *Int.J.Control* **39** (1984) 1115-1193

[91] M. Grepl, "Reduced-Basis Approximation and A Posteriori Error Estimation for Parabolic Partial Differential Equations", Ph.D. Thesis, http://augustine.mit.edu/methodology/papers/grepl_thesis2005.pdf, 2005

[92] E.J. Grimme, "Krylov Projection Methods for Model Reduction", Ph.D. Thesis, http://web.mit.edu/mor/papers/grimme.ps, 1997

[93] M.D. Gunzburger, J.S. Peterson, J.N.Shadid "Reduced-order modeling of time-dependent PDEs with multiple parameters in the boundary data"', *Comput. Methods Appl. Mech. Engrg.* **196** (2007) 1030-1047

[94] M.Hinze, S.Volkwein, "Proper Orthogonal Decomposition Surrogate Models for Nonlinear Dynamical Systems : Error Estimates and Suboptimal Control", *Lecture Notes in Computational Science and Engineering*, **45**, 2005

[95] Holmes P., Lumley J.L., Berkooz G.,"Turbolence, Coderent Structures, Dynamical Systems and Symmetry", Cambridge Monographs on Mechanics, Cambridge University Press, 1996

[96] P.J. Holmes, J.L. Lumley, G. Berkooz, J.C. Mattingly, R.W. Wittenberg "Low-dimensional models of coherent structures in turbulence"', *Physics Reports* **287** (1997) 337-384

[97] K. Karhunen "Zur spektral theorie stochasticher prozesse"', *Ann Acad Sci Fennicae, Ser A1 Math Phys* **34** (1946) 1-7

[98] P.V. Kokotovic, R.E. O'Malley, P. Sannuti "Singular Perturbations and Order Reduction in Control Theory - an Overview"', *Automatica* **12** (1976) 123-132

[99] K. Kunisch, S. Volkwein "Galerkin proper orthogonal decomposition methods for parabolic problems"', *Numer.Math.* **90** (2001) 117-148

[100] K. Kunisch, S. Volkwein "Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics"', *SIAM J. Numer. Anal.* **40** (2002) 492-515

[101] S. Lall, J.E.Marsden, S. Glavaski "Empirical model reduction of controlled nonlinear systems". *Proceedings of the 14th IFAC World Congress*, (1999) pp. 473-478.

[102] M.Loeve "Functiona aleatoire de second ordre"', *Revue Science* **84** (1946) 195-206

[103] E.N. Lorentz, "Empirical orthogonal functions and statistical weather prediction"', Scientific Report 1, Statistical Forecasting Project, MIT, Cambridge, 1956

[104] D.J. Lucia, P.S. Beran, W.A.Silva "Reduced-order modeling: new approaches for computational physics", *Progress in Aerospace Sciences* **40** (2004), 51-117.

[105] J.L. Lumley "The structure of inhomogeneous turbulence", *Atmospheric Turbulence and Radio Wave Propagation* (1967), 166-178.

[106] Z. Luo, J. Chen, I.M.Navon, X. Yang, "Mixed finite element formulation and error estimates based on Proper Orthogonal Decomposition for the Nonstationary Navier-Stokes equations", *SIAM J. Numer. Anal.* **47** (2008), 1-19.

[107] B.C. Moore, "Principal Component Analysis in Linear System: Controllability, Observability and Model Reduction"', IEEE Transactions on Automatic Control, AC-26:17-32, 1981

[108] Noak B., Afanasiev K., Morzynsky M., Tadmor G., Thiele F. "A hierarchy of low dimensional models for the transient and post-transient cylinder wake"', *J. Fluid. Mech.*, **497** (2003), 335-363.

[109] B. Noak B., P. Papas, M. Monkewitz "The need for a pressure-term representation in empirical Galerkin models of incompressible shear flows"', *J. Fluid. Mech.*, **523** (2005), 339-365.

[110] H.M. Park, M.W. Lee, "An efficient method of solving the Navier-Stokes equations for flow control", *Int. J. Num. Meth. Engng* **41** (1998), 1113-1151.

[111] A.T. Patera and G. Rozza, "Reduced Basis Approximation and A Posteriori Error Estimation for Parametrized Partial Differential Equations", Version 1.0, Copyright MIT 2006-2007, to appear in (tentative rubric) MIT Pappalardo Graduate Monographs in Mechanical Engineering (http://augustine.mit.edu/methodology/methodology_book.htm).

[112] B.Podvin, "On the adequacy of the ten-dimensional model for the wall layer", *Phys.Fluids*, **13** (2001), 210-224.

[113] A. Quarteroni, G.Rozza, "Numerical Solution of Parametrized Navier-Stokes Equations by Reduced Basis Methods", *Numerical Methods for PDEs*, **23** No.4 (2007), 923-948.

[114] S.S. Ravindran, "Reduced-Order Adaptive Controllers for Fluid Flows Using POD", *Journal of Scientific Computing*, **15** No.4 (2000), 457-478.

[115] S.S. Ravindran, "A reduced-order approach for optimal control of fluids using proper orthogonal decomposition", *Int. J. Numer. Meth. Fluids*, **34** (2000), 425-448.

[116] Rewienski, M. J. "A Trajectory Piecewise-Linear Approach to Model Order Reduction of Nonlinear Dynamical Systems", Ph.D. Thesis, http://web.mit.edu/mor/papers/rewienski.pdf, 2003

[117] J. Rommes, "Methods for eigenvalue problems with applications in model order reduction", Ph.D. Thesis, http://igitur-archive.library.uu.nl/dissertations/2007-0626-202553/index.htm, 2007

[118] G. Rozza, "Shape design by optimal flow control and reduced basis techniques: applications to bypass configurations in haemodynamics", Ph.D. Thesis, http://biblion.epfl.ch/EPFL/theses/2005/3400/EPFL_TH3400.pdf, 2005

[119] W.H.A. Schilders, H.A. van der Vorst, J. Rommes, "Model Order Reduction: Theory, Research Aspects and Applications", Springer, 2008

[120] L. Sirovich, "Turbolence and the dynamics of coherent structures, parts I-III", *Quart.Appl.Math.* **XLV** (1987), 561-590.

[121] T.R. Smith, J. Moehlis, P. Holmes "Low-dimensional Modeling of Turbolence using the Proper Orthogonal Decomposition: a tutorial", *Nonlinear Dynamics* **41** (2005), 275-307.

[122] R.Temam "Infinite-dimensional Dynamical Systems in Mechanics and Physics", Springer, 1988

[123] L.N. Trefethen, D. Bau, "Numerical Linear Algebra", SIAM, 1997

[124] S. Volkwein, "Proper orthogonal decomposition: applications in optimization and control", lecture notes "CEA-EDF-INRIA Summer School 2008 - Model Reduction and Reduced Basis methods: application in optimization", 2008

[125] S. Volkwein, "Proper orthogonal decomposition and singular value decomposition", Technical Report SFB-153, Institut für Mathematik, Universität Graz, 1999

[126] D. Wee, T. Yi, A. Annaswamy, F. Ghoniem "Self-sustained oscillations and vortex shedding in backward-facing step flows: Simulation and linear instability analysis", *Physics of fluids* **16** (2004), 3361-3373.

Part III: Parabolic Inverse Problems

[127] A.C. Antoulas, "Approximation of large-scale dynamical systems", Siam, 2005

[128] O.M. Alifanov, E.A.Artyukhin, S.V. Rumyantsev, "Extreme Methods for Solving III-Posed Problems with Applications to Inverse Heat Conduction Problems", Begell House, 1995

[129] C.A. Aster, B.Borchers, C.H.Thurber, "Parameter Estimation and Inverse Problems", Elsevier, 2005

[130] R. Becher, B.Vexler, "Optimal control of the convection-diffusion equation using stabilized finite element methods", *Numerische Mathematik*, **106**, 2007, 349-367

[131] M.Bertero and P.Boccacci, "Introduction to Inverse problems in imaging", IOP Publishing Ltd, 1998

BIBLIOGRAPHY

[132] P.Bison and D.Fasino and G.Inglese, "Active infrared thermography in nondestructive evaluation of surface corrosion", Series on Advances in Mathematics for Applied Sciences - Proceedings of the 7th SIMAI Conference - Venice 2004, **69**, (2005),143–154

[133] R. Bracconier, F. Bonthoux, "A Numerical Method of Reconstructing the Pollutant Concentration Field in a Ventilated Room", *Ann.Occup.Hyg.*, **51**, 2007, 311-325

[134] K. Bryan and L. F.Caudill, "Uniqueness for a boundary identification problem in thermal imaging", *Electronic Journal of Differential Equations* **1** (1997) 23–39.

[135] K. Bryan and L. F.Caudill, "Reconstruction of an unknown boundary portion from cauchy data in n-dimensions", *Inverse Problems* **21** (2005) 239–256.

[136] S. S. Collis, M. Heinkenschloss "Analysis of the streamline upwind/petrov galerkin method applied to the solution of optimal control problems", CAAM TR02-01, 2002

[137] X. Davoine, M. Bocquet, "Inverse modelling-based reconstruction of the Chernobyl source term available for long-range transport", *Atmos.Chem.Phys.*, **7**, 2007, 1549-1564

[138] G.Deolmi, F.Marcuzzi, S.Marinetti, S.Poles "Numerical algorithms for an inverse problem of corrosion detection", *Communications in Applied and Industrial Mathematics*, **1**, 2010, 78-98

[139] G.Deolmi, F.Marcuzzi "Parabolic inverse convection-diffusion-reaction problem solved using an adaptive parametrization", *Preprint* http://arxiv.org/abs/1110.2376

[140] L.Formaggia, F.Saleri, A.Veneziani "Applicazioni ed esercizi di modellistica numerica per problemi differenziali", Springer, 2005

[141] A. Friedman, "Partial Differential Equations of Parabolic Type", Dover Publications, 2008

[142] M.Girault, D.Maillet, F.Bonthoux, B. Galland, P. Martin, R. Braconnier, J. R. Fontaine, "Estimation of time-varying pollutant emission rates in a ventilated enclosure: inversion of a reduced model obtained by experimental application of the modal identification method", *Inverse Problems*, **24**, 2008, 1-22

[143] M. Girault, D.Petit, "Resolution of linear inverse forced convection problems using model reduction by the modal identification method: application to turbulent flow in parallel-plate duct", *Int. J. Heat Mass Transfer*, **47**, 2004, 3909-3925

[144] M.D. Gunzburger, J.S. Peterson, J.N.Shadid "Reduced-order modeling of time-dependent PDEs with multiple parameters in the boundary data"', *Comput. Methods Appl. Mech. Engrg.* **196** (2007) 1030-1047

[145] J.Hadamard, "Lectures on the Cauchy Problem in Linear Partial Differential Equations", Yale University Press, New Haven, 1923

[146] A. Hamdi, "Identification of Point Sources in Two Dimensional Advection-Diffusion-Reaction Equation: Application to Pollution Sources in a River. Stationary Case", *Inverse Problems in Science and Engineering*, **00**, 2006, 1-20

[147] M.Hinze, S.Volkwein, "Proper Orthogonal Decomposition Surrogate Models for Nonlinear Dynamical Systems : Error Estimates and Suboptimal Control", *Lecture Notes in Computational Science and Engineering*, **45**, 2005

[148] T. Hohage and M.L. Rapun and F.J. Sajas, "Detecting corrosion using thermal measurements", *Inverse Problems* **23** (2007) 53–72.

[149] V.Isakov, "Inverse Problems for Partial Differential Equations", Springer, 2006

[150] B. Kaltenbacher, A. Neubauer, O. Scherzer "Iterative Regularization Methods for Nonlinear Ill-Posed Problems", Walter de Gruyter, 2008

[151] A.Kirsch, "An introduction to the mathematical theory of Inverse Problems", Springer, 1996

[152] O.A. Ladyzenskaja, V.A. Solonnikov, N.N. Uralceva "Linear and Quasilinear Equations of Parabolic type", American Mathematical Society, 1968

[153] L. Ljung, "System Identification: theory for the user", Prentice Hall, 1987

[154] G. Lube, B. Tews,"Distributed and boundary control of singularly perturbed advection-diffusion-reaction problems", *Lecture Notes in Computational Science and Engineering*, **69**, 2009, 205-215

[155] F. Marcuzzi and S. Marinetti, "Efficient reconstruction of corrosion profiles by infrared thermography", *Journal of Physics: Conference Series* **124** (2008).

BIBLIOGRAPHY

[156] F. Marcuzzi, "Space and time localization for the estimation of distributed parameters in a finite element model", *Computer Methods in Applied Mechanics and Engeneering*, **198**, 2009, 3020-3025

[157] S. Marinetti and P.G. Bison and E. Grinzato, "3D heat effects in the experimental evaluation of corrosion by IR thermography", Proceedings of the 6th International Conference on Quantitative Infrared Thermography QIRT 2002, Dubrovnik (Croatia), (2002), 92–98

[158] S.Marinetti and V.Vavilov, "IR thermographic detection and characterization of hidden corrosion in metals: General analysis", *Corrosion Science* **52** (2010) 865–872.

[159] J.R.R.A. Martins, P.Sturdza, J.J.Alonso "The Complex-Step Derivative Approximation", *ACM Transactions on Mathematical Software*, **29**, 2003, 245-262

[160] B.P. McGrail, "Inverse reactive transport simulator (INVERTS): an inverse model for contaminant transport with nonlinear adsorption and source terms", *Environ. Model. Softw.*, **16**, 2001, 711-723

[161] A.Moutsoglou, "An inverse convection problem", *J. Heat Transfer*, **111**, 1989, 37-43

[162] J.Nocedal, S.J. Wright, "Numerical optimization", Springer, 1999

[163] H.M.Park, J.Chung, "A sequential method of solving inverse natural convection problems", *Inverse Problems*, **18**, 2002, 529-546

[164] A.Quarteroni, A.Valli, "Numerical approximation of Partial Differential Equations", Springer, 1994

[165] A. Rap, L. Elliott, D.B.Ingham, D. Lesnic, X.Wen,"An inverse source problem for the convection-diffusion equation", *International Journal of Numerical Methods for Heat & Fluid Flow*, **16**,2006, 125-150

[166] A. Samarskii, P. Vabishchevich, "Numerical Methods for Solving Inverse Problems of Mathematical Physics", Walter de Gruyter, 2007

[167] J.R. Shenefelt, R. Luck, R.P. Taylor, J.T. Berry, "Solution to inverse heat conduction problems employing singular value decomposition and model-reduction", *Int. J. Heat Mass Transfer*, **45**,2002, 67-74

[168] W.H.A. Schilders, H.A. van der Vorst, J. Rommes, "Model Order Reduction: Theory, Research Aspects and Applications", Springer, 2008

[169] A.N. Tikhonov, V.Y. Arsenin, "Solutions of Ill-Posed Problems", V HWinston, 1977

[170] F. Tröltzsch, "Optimal control of Partial Differntial Equations", American Mathematical Society, 2010

[171] V.Vavilov and E.Grinzato and P.G. Bison and S.Marinetti and M.J. Bales, "Surface transient temperature inversion for hidden corrosion characterization: Theory and applications", *Int. Journal Heat and Mass Tranfer* **39** (1996) 355–371.

[172] K.A. Woodbury, "Inverse Engineering Handbook", CRC Press, 2002

[173] N. Zabaras, G. Z. Yang, "A functional optimization formulation and implementation of an inverse natural convection problem", *Comput. Methods Appl. Mech. Engrg.*, **144**, 1997, 245-274

BIBLIOGRAPHY