CENTRO INTERNACIONAL DE MÉTODOS NUMÉRICOS EN INGENIERÍA

Stabilized Pressure Segregation Methods and their Application to Fluid-Structure Interaction Problems

S. Badia
R. Codina

Monograph CIMNE Nº-96, May 2006

Stabilized Pressure Segregation Methods and their Application to Fluid-Structure Interaction Problems

S. Badia
R. Codina

Monograph CIMNE Nº-96, May 2006

INTERNATIONAL CENTER FOR NUMERICAL METHODS IN ENGINEERING
Gran Capitán s/n, 08034 Barcelona, Spain

INTERNACIONAL CENTER FOR NUMERICAL METHODS IN ENGINEERING
Edificio C1, Campus Norte UPC
Gran Capitán s/n
08034 Barcelona, Spain
www.cimne.upc.es

First edition: May 2006

Stabilized Pressure Segregation Methods and their Application to Fluid-Structure Interaction
Monograph CIMNE M96

ISBN: 84-95999-93-5

Depósito legal: B-26075-2006

Abstract

In this work we design and analyze pressure segregation methods in order to approximate the Navier-Stokes equations. Pressure correction methods are widely used because they allow the decoupling of velocity and pressure computation, decreasing the computational cost. We have analyzed some of these schemes, obtaining inherent pressure stability. However, for second order accurate methods (in time) this inherent stability is too weak, requiring the introduction of a stabilized finite element methodology for the space discretization. Moreover, we have carried out a complete convergence analysis of a first order pressure segregation method.

We have used a stabilization technique justified from a multiscale approach that allows the use of equal velocity-pressure interpolation spaces and convection dominated flows.

A new kind of methods has been motivated from an alternative version of the monolithic fluid solver where the continuity equation is replaced by a discrete pressure Poisson equation. These methods belong to the family of velocity correction schemes, where it is the velocity instead of the pressure the extrapolated unknown. Some stability bounds have been proved, revealing that their inherent pressure stability is too weak. Further, predictor corrector schemes easily arise from the new monolithic system. Numerical experimentation shows the good behavior of these methods.

We have introduced the ALE framework in order for the fluid governing equations to be formulated on moving domains. Taking as the model equation the convection-diffusion equation, we have analyzed the blend of the ALE framework and a stabilized finite element method.

We suggest a coupling procedure for the fluid-structure problem taking benefit from the ingredients previously introduced: pressure segregation methods, a stabilized finite element formulation and the ALE framework. The final algorithm, using one loop, tends to the monolithic (fluid-structure) system.

This method has been applied to the simulation of bridge aerodynamics, obtaining a good convergence behavior.

We end with the simulation of wind turbines. The fact that we have a rotary body surrounded by the fluid (air) has motivated the introduction of a remeshing strategy. We consider a selective remeshing procedure that only affects a tiny portion of the domain, with little impact on the overall CPU time.

Contents

Introduction

The description of incompressible flow problems is determined by the Navier-Stokes equations. The numerical approach to this system of partial difference equations is a difficult task that has focused an active research in the last decades.

The incompressibility constraint couples the velocity and pressure calculation, being its numerical solution expensive. Furthermore, the velocity and pressure interpolation spaces must satisfy a compatibility condition for standard discretizations, as the Galerkin method.

Another unrelated complication is the instability that arises from the advective term, whose nature is completely different from the previous one. Oscillations appear for convection dominated flows and disappear under a specific mesh size, which in most cases is not affordable.

The instabilities related to the convective term motivated the appearance of stabilized numerical methods in the late 1970's. Later, it was observed that these techniques (with appropriate modifications) could avoid the compatibility requirement over the interpolation spaces.

Both instabilities can be avoided using a stabilized finite element method. This is the technique used here in order to obtain appropriate numerical methods. The stabilizing technique used in this work has been motivated by the decomposition of the continuous solution into a coarse component (finite element solution) and fine (subgrid) component.

On the other hand, in order to reduce the computational cost, we have explored pressure segregation methods (indistinctly called fractional step methods herein). Until its appearance in the late 1960's, with the pioneering works of Chorin and Temam, these methods have enjoyed a widespread popularity. Its common feature is the decoupling of velocity and pressure interpolation, yielding an important reduction of CPU time. This decoupling can be based on the splitting of the differential operator, as the classical projection method. In other cases it is due to the prediction (and possibly correction) of the velocity or the pressure. Furthermore, these methods are provided with an inherent stability, allowing in many situations the use of space interpolations which do not satisfy the compatibility condition. However, several aspects, as the pressure boundary layer, still deserve further studies.

We classify the pressure segregation methods presented in this work in two families: pressure correction methods and velocity correction methods.

The pressure correction methods include the most classical schemes proposed in the literature. We situate into this group the Chorin-Temam projection method and the Van Kan method. In the present work we focus on implicit pressure segregation schemes based on Backward Differencing (BDF) and θ-methods for the time integration.

We consider pressure correction methods split at the discrete level. First and second order accurate (in time) schemes have been proposed. The numerical analysis puts lights on the mechanism that introduces stability in this kind of methods. Moreover, this analysis

reveals the fact that the inherent pressure stability decreases whit the splitting error. For second order methods this pressure bound is too weak. We show how this bound is enhanced by the introduction of a stabilized finite element method for the pressure. After that, we introduce an *a priori* new kind of methods recently proposed in the literature. These methods turn out to be typical pressure correction schemes with an unusual pressure extrapolation.

Some analytical analyses of the continuous (in space) pressure segregation methods can be found in the literature. In this work we analyze a fully discrete pressure segregation method. We consider two different situations: velocity and pressure interpolation spaces holding the inf-sup condition, and secondly, a stabilized pressure correction method when a pressure Poisson equation is used.

Then we suggest a new format for the fully discrete monolithic system absolutely equivalent to the classical one. In this new format the continuity equation for the velocity is replaced by a discrete pressure Poisson equation (DPPE), which is not plagued by the problems found when the pressure Poisson equation is obtained at the continuous level and then discretized. From this algebraic system a new kind of pressure segregation methods naturally arise. But in this case the extrapolated variable is the velocity instead of the pressure. The schemes belonging to this family are consequently called velocity correction methods. A set of stability results for these schemes is obtained. These bounds show that these methods only have an inherent pressure stability under an approximation of the discrete pressure Poisson equation. However, the pressure stability bounds are too weak, being required the introduction of a stabilizing technique. Numerical tests show the good behavior of stabilized versions of this new kind of methods.

Predictor corrector methods are also proposed. Some of them are motivated from pressure correction methods. The rest are obtained directly from the DPPE monolithic system. Numerical experimentation is used to show the behavior of these schemes, specially its convergence towards the monolithic solution. Surprisingly, predictor-corrector methods obtained directly from the DPPE monolithic system have good convergence properties, while those from the classical monolithic system are not appropriate.

We introduce the ALE framework that allows to write the Navier-Stokes equations on moving domains. With this methodology we can easily extend all the pressure segregation methods to moving domains. We consider two ALE schemes that are first and second order accurate in time. Moreover, using as model test the convection-diffusion equation we analyze the blend of the ALE formulation and a stabilized finite element method for the space discretization. Optimal stability bounds and error estimates have been proved.

All these ingredients allow the simulation of flows on moving domains, as it is needed when treating fluid-structure interaction problems. The next aspect that we consider in order to simulate this heterogeneous system is how to couple the fluid and structure problems.

We introduce the fluid-structure problem in a setting influenced by domain decomposition methods. Our contribution to this topic is the use of pressure segregation methods. We suggest the use of a fixed point algorithm for the coupling together with a predictor corrector fluid solver. The final algorithm tends to the monolithic system using one external loop dealing with non-linearity of the fluid, coupling and the predictor corrector iterations for the fluid solver.

This method has been applied to a real engineering problem. We analyze the aeroelastic behavior of bridges, obtaining a good convergence ratio for the coupling procedure. We attain to assess flexural and torsional frequencies for a given inflow velocity. Moreover, increasing the inflow velocity we reach the flutter limit of the structure.

We end this work with the simulation of wind turbines. This problems consists of a rotary body that is surrounded by a fluid. In order to simulate this problem a domain decomposition method or a remeshing strategy must be used. We have considered a selective remeshing procedure. This process increases the overall CPU time and has to be done with care. We divide the finite element partition of the domain *a priori* in three different parts: the fixed mesh, the rotary mesh (that is moving with the body) and the transmission mesh (where the remeshing strategy is used). We have evaluated the forces and moments exerted by the air over the wind turbine using this methodology.

This dissertation is structured into nine chapters. We start the work introducing the Navier-Stokes equations for incompressible flows in their strong and weak forms. Further, Chapter 1 also includes their numerical approximation using finite element methods (in space) and finite differences (in time). The chapter is completed with some mathematical concepts that will be used throughout this monograph. In Chapter 2 we present the stabilized numerical method used along the work for the spatial discretization. Chapter 3 is devoted to pressure correction methods. Different pressure correction methods and predictor corrector versions are suggested and their inherent pressure stability studied. A complete convergence analysis of a first order pressure correction method has been elaborated in Chapter 4. We present a new kind of pressure segregation methods in Chapter 5, belonging to the family of velocity correction methods. These methods arise from an alternative version of the monolithic system at the discrete level. This new system also motivates predictor corrector schemes. The inherent pressure stability of velocity correction methods is also analyzed. Chapter 6 is devoted to the introduction of the ALE framework that allows the use of moving domains. Moreover, the numerical analysis of the transient convection diffusion equation written in an ALE framework and discretized using the stabilizing technique presented in Chapter 2 is carried out. In chapter 7 we propose some coupling procedures designed for the use of pressure segregation methods. The application of the numerical methods developed in the previous chapters to real engineering problems is presented in Chapters 8 and 9. Chapter 8 is devoted to the simulation of bridge aerodynamics. Finally, the numerical simulation of wind turbines is introduced in Chapter 9. This problem involves fixed and rotating objects. It has justified the introduction of a selective remeshing strategy.

Regarding the notation employed, most chapters are almost self-contained. This simplifies the reading at the expense of repeating some specific notation (general concepts are introduced in Chapter 1).

The present work has a strongly heterogeneous taste. Even though it is clearly inside the frame of computational mechanics, there is a wide distance between the motivation of the different chapters. Chapters 3, 5 and 7 are devoted to the *design of numerical methods* for the simulation of fluid flow problems and the coupling of fluids and structures. Further, some numerical analyses are carried out in order to understand these methods. Chapters 4 and 6 are clearly focused on the *numerical analysis of finite element approximations*. In these chapters we prove a complete set of convergence and stability results for a fully discrete pressure correction method and a stabilized ALE-FEM approximation of the convection-diffusion equation. Finally, different numerical methods suggested along this work have been exploited in Chapters 8 and 9 in order to simulate real problems. Thus, the last two chapters of this dissertation are devoted to the *application of numerical methods*.

Chapter 1

Preliminaries

It is mandatory to start the present work introducing the Navier-Stokes equations for incompressible flows. In this context, we first recall the basic theory of fluid mechanics. After that we introduce spaces and norms needed for the analysis of the equations. Concepts about existence, uniqueness and regularity of the solutions are pointed out, and some basic assumptions are stated. We define the forms associated with different terms of the set of equations in their weak form, and some continuity properties are listed. We also recall some regularity results of the Stokes operator. Later on, we introduce the basic theory of approximation used with the Finite Element Method, and added emphasis is placed on the compatibility condition required to be satisfied in standard numerical approximations. The last section is devoted to the time integration schemes which will be used hereafter.

1.1 The Incompressible Navier-Stokes equation

Let us start with the basic equations of fluid mechanics for incompressible flows. These equations are obtained from principles of conservation under the basic assumptions assumed for the continuum mechanics theory.

The evolutionary equations for a fluid moving in a domain Ω of $\mathbb{R}^d$ ($d = 2$ or 3 being the space dimension) in a time interval $[0, T]$ consists of finding a velocity $\boldsymbol{u}$ and a pressure p such that

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \frac{1}{\rho} \nabla \cdot \boldsymbol{\sigma} = \boldsymbol{f} \quad \text{in } \Omega \times (0, T), \tag{1.1a}$$

$$\nabla \cdot \boldsymbol{u} = 0 \quad \text{in } \Omega \times (0, T), \tag{1.1b}$$

where $\boldsymbol{f}$ is the force vector, $\boldsymbol{\sigma}$ the stress tensor and ρ is the density. For Newtonian and isotropic fluids the stress tensor is determined by the following constitutive equation:

$$\boldsymbol{\sigma} = -\hat{p}\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) \tag{1.2}$$

where μ is the viscosity, $\boldsymbol{I}$ is the identity tensor, $\hat{p}$ is the pressure field, and $\boldsymbol{\varepsilon}(\boldsymbol{u})$ the strain rate tensor, defined as

$$\boldsymbol{\varepsilon}(\boldsymbol{u}) = \frac{1}{2}[(\nabla \boldsymbol{u}) + (\nabla \boldsymbol{u})^t]. \tag{1.3}$$

Under these hypothesis the momentum equation (1.1a) can be rewritten as:

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} - 2\nu \nabla \cdot \boldsymbol{\varepsilon}(\boldsymbol{u}) + \nabla p = \boldsymbol{f} \quad \text{in } \Omega \times (0, T) \tag{1.4}$$

where $\nu = \mu/\rho$ is the kynematic viscosity and p the kynematic pressure defined as

$$p = \hat{p}/\rho. \tag{1.5}$$

Due to the incompressibility constraint (1.1b), for a constant μ it is easily seen that

$$2\nabla \cdot (\mu \boldsymbol{\varepsilon}(\boldsymbol{u})) = \mu \Delta \boldsymbol{u}, \tag{1.6}$$

arriving at the most usual form of the evolutionary incompressible Navier-Stokes equations,

$$\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla p = \boldsymbol{f} \quad \text{in } \Omega \times (0, T). \tag{1.7}$$

The equations set (1.7)-(1.1b) has to be supplemented with appropriate boundary and initial conditions in order to have a well-posed system. The boundary $\Gamma \equiv \partial\Omega$ is a $(d-1)$-dimensional manifold assumed locally Lipschitz (i.e., smooth enough). It is assumed that Γ can be partitioned into two non-overlapping subsets Γ_D and Γ_N, such that $\Gamma_D \cap \Gamma_N = \emptyset$ and $\Gamma_D \cup \Gamma_N = \Gamma$. Then, we have in the simplest case

$$\boldsymbol{u} = \boldsymbol{u}_g \qquad \text{on } \Gamma_D \times (0, T), \tag{1.8a}$$

$$\boldsymbol{n} \cdot \boldsymbol{\sigma} = \boldsymbol{t}_N \qquad \text{on } \Gamma_N \times (0, T), \tag{1.8b}$$

$$\boldsymbol{u}(\boldsymbol{x}, 0) = \boldsymbol{u}_0 \qquad \text{in } \Omega \times \{0\}, \tag{1.8c}$$

where $\boldsymbol{n}$ is the unit outward normal vector to Γ. For purely Dirichlet type boundary conditions, i.e. $\Gamma_N = \emptyset$, the following compatibility condition has to be satisfied

$$\int_\Gamma \boldsymbol{n} \cdot \boldsymbol{u}_g \, \mathrm{d}\Gamma = 0 \quad \forall t \in (0, T) \tag{1.9}$$

in order for the problem to be well posed. Otherwise, the incompressibility constraint (1.1b) cannot be verified.

For the sake of simplicity we have assumed the homogeneous boundary condition,

$$\boldsymbol{u}_g = 0 \quad \text{on } \Gamma \times (0, T) \tag{1.10}$$

for the following stability and convergence analyses.

The adimensionalized version of the Navier-Stokes equation is of great interest. Being U_0 and L a characteristic velocity and length of the problem, respectively, we can adimensionalize the independent variables using $\boldsymbol{x}' := \boldsymbol{x}/L$ and $t' := U_0 t/L$ and the dependent variables velocity and kynematic pressure as $\boldsymbol{u}'(\boldsymbol{x}, t) = \boldsymbol{u}(\boldsymbol{x}, t)/U_0$ and $p'(\boldsymbol{x}, t) = p(\boldsymbol{x}, t)/U_0^2$, respectively. Inserting these expressions in (1.7) we get

$$\partial_t \boldsymbol{u}' - \frac{1}{Re} \Delta \boldsymbol{u}' + \boldsymbol{u}' \cdot \nabla \boldsymbol{u}' + \nabla p' = \boldsymbol{f}' \quad \text{in } \Omega \times (0, T),$$

where $Re := U_0 L/\nu$ is the *Reynolds number* that informs about the relative weight of the convective and diffusive terms. The flux is convection dominated for high Re and dominated by the viscosity for low Re.

When the stationary case is considered, the time derivative disappears, resulting in the steady, incompressible Navier-Stokes (momentum) equation,

$$-\nu \Delta \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla p = \boldsymbol{f} \quad \text{in } \Omega. \tag{1.11}$$

Furthermore, for slow motion or low Reynolds number flows, the convective term can be neglected, resulting in the evolutionary Stokes system,

$$\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \nabla p = \boldsymbol{f} \quad \text{in } \Omega, \tag{1.12}$$

or in its steady state case,

$$-\nu \Delta \boldsymbol{u} + \nabla p = \boldsymbol{f} \quad \text{in } \Omega. \tag{1.13}$$

1.2 Some function spaces

Let us introduce some notation that will be used hereafter. For a wide exposition of the mathematical concepts mentioned here we refer to any standard functional analysis text (for instance, [131]). More specific aspects of functional analysis when applied to the Navier-Stokes equations can be found in [81] and [159].

Let $\Omega \subset \mathbb{R}^d$, $d = 2$ or 3, be a bounded domain. We define as $\mathcal{C}_0^\infty(\Omega)$ the set of infinitely differentiable real functions with compact support on Ω. Moreover, we denote by $L^p(\Omega)$, $1 \leq p < \infty$, the space of real functions defined on Ω with the p-th power absolutely integrable with respect to the Lebesgue measure. It is a Banach space with the associated norm

$$\|u\|_{L^p(\Omega)} := \left(\int_\Omega |u(x)|^p \, \mathrm{d}\Omega \right)^{1/p}.$$

For $1 < p < \infty$, $L^p(\Omega)$ is a reflexive space and its dual space is $L^q(\Omega)$, with q such that $1/p + 1/q = 1$. One has for $1 < s < r < \infty$, $L^r(\Omega) \subset L^s(\Omega)$.

The case $p = 2$ is of special interest; $L^2(\Omega)$ is a Hilbert space endowed with the scalar product

$$(u, v)_\Omega := \int_\Omega u(x)v(x) \, \mathrm{d}\Omega$$

and its induced norm

$$\|u\|_{L^p(\Omega)} := (u, u)_\Omega^{1/2}$$

Furthermore, $L^2(\Omega)$ is identified with its dual space. In the following, we will likely omit the subscript referred to the domain for which we consider the scalar product when this is the problem domain, that is to say, Ω. Moreover, we will write the $L^2(\Omega)$ norm simply as $\|\cdot\|_0$.

The space $L^\infty(\Omega)$ (for $p = \infty$) consists of essentially bounded functions in Ω. It is a Banach space equipped with the norm

$$\|u\|_{L^\infty(\Omega)} := \operatorname*{ess\,sup}_\Omega |u(x)|.$$

Its dual space is $L^1(\Omega)$. Furthermore, $L^\infty(\Omega) \subset L^p(\Omega)$ for all $p \in [1, \infty)$.

The Sobolev space $W^{m,p}(\Omega)$ is the space of functions in $L^p(\Omega)$ whose weak derivatives of order less than or equal to m belong to $L^p(\Omega)$, being m an integer and $1 \leq p \leq \infty$. It is a Banach space, denoting its norm as $\|\cdot\|_{W^{m,p}(\Omega)}$, or in a more compact manner $\|\cdot\|_{m,p}$. When $p = 2$, the space $W^{m,2}(\Omega) = H^m(\Omega)$ is a Hilbert space endowed with an scalar product and its associated norm $\|\cdot\|_{H^m(\Omega)}$ (also denoted $\|\cdot\|_m$). For instance, when $m = 1$, we have the scalar product

$$((u, v))_\Omega := (u, v)_\Omega + \sum_{i=1}^{d} (\partial_i u, \partial_i v)_\Omega$$

and norm

$$\|u\|_{H^1(\Omega)} := ((u, u))_\Omega^{1/2}.$$

The space $H_0^1(\Omega)$ consists of functions of $H^1(\Omega)$ with zero trace on the boundary. As Ω is a bounded domain the Poincaré inequality holds, and then,

$$\|u\|_0 \leq C_\Omega \|\nabla u\|_0 \quad \forall u \in H_0^1(\Omega). \tag{1.14}$$

Therefore the norm $\|\nabla u\|_0$ is equivalent to $\|u\|_1$ on $H_0^1(\Omega)$. Furthermore, we denote by $H^{-k}(\Omega)$ the dual space of $H_0^k(\Omega)$.

The bilinear form defined over $H^{-1}(\Omega) \times H_0^1(\Omega)$,

$$\langle v, u \rangle_\Omega := \int_\Omega v(x)u(x)\,\mathrm{d}\Omega$$

is the duality pairing in $H^1(\Omega)$. Again, we can omit the domain subscript when Ω is the problem domain.

We shall often consider d-dimensional vector functions with components in one of these spaces. We shall indicate them by boldface letters, for instance $\boldsymbol{H}^m(\Omega) = (H^m(\Omega))^d$. In the next, we will not distinguish between scalar products or norms for scalar or vector-valued functions.

Let us introduce some convenient spaces for the treatment of the incompressibility constraint. The first one is

$$\boldsymbol{H}(\mathrm{div}, \Omega) := \{\boldsymbol{u} \in \boldsymbol{L}^2(\Omega) \mid \nabla \cdot \boldsymbol{u} \in L^2(\Omega)\}, \tag{1.15}$$

which is a Hilbert space with the norm $\|\boldsymbol{u}\|_{\mathrm{div}} = \|\boldsymbol{u}\|_0 + \|\nabla \cdot \boldsymbol{u}\|_0$. We also introduce

$$\boldsymbol{H}_0(\mathrm{div}, \Omega) := \{\boldsymbol{u} \in \boldsymbol{H}(\mathrm{div}, \Omega) \mid \boldsymbol{n} \cdot \boldsymbol{u}|_\Gamma = 0\}.$$

We define the spaces whose functions are *weakly divergence-free*,

$$\boldsymbol{J}_0 := \{\boldsymbol{u} \in \boldsymbol{L}^2(\Omega) \mid \nabla \cdot \boldsymbol{u} = 0,\ \boldsymbol{n} \cdot \boldsymbol{u}|_\Gamma = 0\} \tag{1.16}$$

and,

$$\boldsymbol{J}_1 := \{\boldsymbol{u} \in \boldsymbol{H}_0^1(\Omega) \mid \nabla \cdot \boldsymbol{u} = 0\}. \tag{1.17}$$

Since $\boldsymbol{J}_0$ is a closed subspace of $\boldsymbol{L}^2(\Omega)$, it can be decomposed as $\boldsymbol{L}^2(\Omega) = \boldsymbol{J}_0 \oplus \boldsymbol{J}_0^\perp$, where

$$\boldsymbol{J}_0^\perp := \{\boldsymbol{u} \in \boldsymbol{L}^2(\Omega) \mid \boldsymbol{u} = \nabla p,\ p \in \boldsymbol{H}^1(\Omega)\}. \tag{1.18}$$

We define $P_{\boldsymbol{J}_0}$ as the orthogonal $L^2(\Omega)$-projector onto $\boldsymbol{J}_0$. This operator is of main importance, and the basis of the original projection methods (see [40] and [157]). It is obviously continuous in $\boldsymbol{L}^2(\Omega)$ by its definition. In fact, for Ω being an open bounded set of class $\mathcal{C}^2$, the operator $P_{\boldsymbol{J}_0}$ also maps $\boldsymbol{H}^1(\Omega)$ into itself and is continuous with respect to $\|\cdot\|_1$ (see [159]), that is to say, there exists a constant $C > 0$ such that:

$$\|P_{\boldsymbol{J}_0} f\|_1 \leq C\|f\|_1 \qquad \forall f \in \boldsymbol{H}^1(\Omega). \tag{1.19}$$

We need to introduce two forms associated to the different terms of the Navier-Stokes equations in their weak form, defined on the appropriate spaces. Let us start with the form related to the viscous term,

$$a(\boldsymbol{u}, \boldsymbol{v}) := \nu(\nabla \boldsymbol{u}, \nabla \boldsymbol{v}), \quad \forall \boldsymbol{u}, \boldsymbol{v} \in \boldsymbol{H}^1(\Omega). \tag{1.20}$$

This is a bilinear continuous form on $\boldsymbol{H}_0^1(\Omega)$ which is coercive with respect to $\|\cdot\|_1$.

The second form to be introduced is required for the pressure gradient and the incompressibility constraint,

$$b(\boldsymbol{v}, q) := -(q, \nabla \cdot \boldsymbol{v}), \quad \forall \boldsymbol{v} \in \boldsymbol{H}_0^1(\Omega),\ \forall q \in L^2(\Omega) \tag{1.21}$$

which is also continuous with respect to the norms $\|q\|_0$ and $\|\boldsymbol{v}\|_1$.

Finally, the trilinear form associated to the convective term in its standard form is

$$c(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}) = (\boldsymbol{u} \cdot \nabla \boldsymbol{v}, \boldsymbol{w}), \quad \forall \boldsymbol{u}, \boldsymbol{v} \in \boldsymbol{H}^1(\Omega),\ \forall \boldsymbol{w} \in \boldsymbol{H}_0^1(\Omega) \tag{1.22}$$

which is also continuous and well defined on these spaces. If $\boldsymbol{u} \in \boldsymbol{J}_0$, the form is skew-symmetric in its two last arguments and thus,

$$c(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{v}) = 0, \quad \forall \boldsymbol{u} \in \boldsymbol{J}_0,\ \forall \boldsymbol{v} \in \boldsymbol{H}_0^1(\Omega). \tag{1.23}$$

Let us introduce some continuity properties that will be used hereafter (see [58]):

$$c(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}) \leq \begin{cases} C\|\boldsymbol{u}\|_1 \|\boldsymbol{v}\|_1 \|\boldsymbol{w}\|_1, \\ C\|\boldsymbol{u}\|_0 \|\boldsymbol{v}\|_2 \|\boldsymbol{w}\|_1, \\ C\|\boldsymbol{u}\|_2 \|\boldsymbol{v}\|_1 \|\boldsymbol{w}\|_0, \\ C\|\boldsymbol{u}\|_0 \|\boldsymbol{v}\|_1 \|\boldsymbol{w}\|_2, \\ C\|\boldsymbol{u}\|_2 \|\boldsymbol{v}\|_0 \|\boldsymbol{w}\|_1. \end{cases} \tag{1.24}$$

By using the Hölder inequality together with some Sobolev inequalities, the following bound is obtained in [92],

$$\max\{c(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}), c(\boldsymbol{v}, \boldsymbol{u}, \boldsymbol{w})\} \leq C(\|\boldsymbol{u}\|_{0,\infty} + \|\boldsymbol{u}\|_{1,3}) \|\boldsymbol{v}\|_0 \|\boldsymbol{w}\|_1. \tag{1.25}$$

Its skew-symmetric form, introduced by Temam in [159],

$$\tilde{c}(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}) = \frac{1}{2}(c(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}) - c(\boldsymbol{u}, \boldsymbol{w}, \boldsymbol{v})), \quad \forall \boldsymbol{u} \in \boldsymbol{H}^1(\Omega),\ \boldsymbol{v}, \boldsymbol{w} \in \boldsymbol{H}_0^1(\Omega) \tag{1.26}$$

is also used in the following. The previous boundedness properties are also inherited by this skew-symmetric form. In any case the superscript $\tilde{}$ will be omitted.

For the treatment of evolutionary problems, we require the following notation. Given $T > 0$ and $1 < p < \infty$, and X a Banach space with norm $\|\cdot\|_X$, let $L^p(0, T; X)$ be the space of functions $f : (0, T) \to X$ such that $\|\boldsymbol{u}\|_{L^p(0,T;X)} = (\int_0^T \|\boldsymbol{u}(s)\|_X^p ds)^{1/p} < \infty$. In the case of $p = \infty$, we demand the property $\sup_{0 \leq s \leq T} \|\boldsymbol{u}(s)\|_X < \infty$. These spaces will be often employed.

In addition we will define the Stokes operator

$$A\boldsymbol{u} = -P_{\boldsymbol{J}_0} \Delta \boldsymbol{u} \qquad \forall \boldsymbol{u} \in \boldsymbol{J}_1 \cap \boldsymbol{H}^2(\Omega) \tag{1.27}$$

which is an unbounded positive self-adjoint closed operator in $\boldsymbol{J}_0$. The operator A^{-1} is the inverse of the Stokes operator A and is compact in $\boldsymbol{J}_0$. Given $\boldsymbol{u} \in \boldsymbol{J}_0$, by definition of A, $\boldsymbol{z} = A^{-1}\boldsymbol{u}$ is the solution of the following Stokes problem:

$$\begin{aligned} -\Delta \boldsymbol{z} + \nabla \xi &= \boldsymbol{u} \quad \text{in } \Omega, \\ \nabla \cdot \boldsymbol{z} &= 0 \quad \text{in } \Omega, \\ \boldsymbol{z} &= 0 \quad \text{on } \partial\Omega. \end{aligned} \tag{1.28}$$

When Ω is of class $\mathcal{C}^2$, or is a convex polygon or polyhedron (see [97]), there exists a constant $C > 0$ such that

$$\|A^{-1}\boldsymbol{u}\|_s \leq C\|\boldsymbol{u}\|_{s-2} \qquad \text{for} \;\; s = 1, 2. \tag{1.29}$$

Moreover, from (1.28) we know that $(A^{-1}\boldsymbol{u}, \boldsymbol{u}) = \|A^{-1}\boldsymbol{u}\|_1^2$, and then it is easily seen that

$$\|\boldsymbol{u}\|_{\boldsymbol{J}_1'}^2 = (A^{-1}\boldsymbol{u}, \boldsymbol{u}), \tag{1.30}$$

for all $\boldsymbol{u} \in \boldsymbol{J}_0$.

1.3 Finite element approximation

In this section we approximate the previous spaces with finite-dimensional subspaces that can be handled numerically. We start summarizing the definition of a finite element, used for constructing the approximation of the previous functional spaces. Then, some approximation properties and an inverse inequality are stated without any proof. The reader is referred to the complete texts of Ciarlet [43], Strang and Fix [153] or Brenner and Scott [23] for a deep study of the finite element approximation.

Let Θ_h be a finite element partition of the domain Ω in a family of elements $\{K_e\}_{e=1}^{n_{el}}$, that is to say, a sequence of elements K_e for $e = 1, ..., n_{el}$, being n_{el} the number of elements. We denote the diameter of the sphere that circumscribes the element K by h_K and the diameter of the sphere inscribed in K by ϱ_K. We also call $h = \max_{K\in\Theta_h}(h_K)$ and $\varrho = \min_{K\in\Theta_h}(\varrho_K)$. We assume that all the element domains $K \in \Theta_h$ are the image of a reference element $\tilde{K}$ through polynomial mappings F_k, affine for simplicial elements, bilinear for quadrilaterals and trilinear for hexahedra. On $\tilde{K}$ we define the polynomial spaces $R_k(\tilde{K})$ where R_k is, for simplicial elements, the set of polynomials in $x_1, ..., x_d$ of degree less than or equal to k, called P_k. For quadrilaterals and hexahedra R_k consists of polynomials in $x_1, ..., x_d$ of degree less than or equal to k in each variable, called Q_k. The finite element spaces we will use in the following are:

$$\mathcal{Q}_h = \{q_h \in \mathcal{C}^0(\Omega) \cap \boldsymbol{L}^2(\Omega)/\mathbb{R} \mid q_h|_K = \tilde{q} \circ F_K^{-1},\ \tilde{q} \in R_{k_\mathrm{q}}(\tilde{K}),\ K \in \Theta_h\}, \tag{1.31}$$

$$\mathcal{V}_h = \{\boldsymbol{v}_h \in (\mathcal{C}^0(\Omega))^d \mid \boldsymbol{v}_h|_K = \tilde{\boldsymbol{v}} \circ F_K^{-1},\ \tilde{\boldsymbol{v}} \in (R_{k_\mathrm{v}}(\tilde{K}))^d,\ K \in \Theta_h\}, \tag{1.32}$$

$$\mathcal{V}_{h,0} = \{\boldsymbol{v}_h \in \mathcal{V}_h \mid \boldsymbol{v}_h|_\Gamma = \boldsymbol{0}\}, \tag{1.33}$$

$$\mathcal{Y}_{h,0} = \{\boldsymbol{v}_h \in \mathcal{V}_h \mid \boldsymbol{n} \cdot \boldsymbol{v}_h|_\Gamma = 0\}, \tag{1.34}$$

which are finite dimensional spaces approximating $L^2(\Omega)$, $\boldsymbol{H}^1(\Omega)$, $\boldsymbol{H}_0^1(\Omega)$ and $\boldsymbol{J}_0$ respectively. In the following, finite element functions will be identified with a subscript h. The space $\mathcal{Q}_h$ will be associated with the pressure (k_q being the degree of the approximation for the pressure) and $\mathcal{V}_{h,0}$ (of degree k_v) with the velocity field. Both spaces are referred to the same partition and both are made up with continuous functions. In some cases, different combinations of $\mathcal{Q}_h$ and $\mathcal{V}_{h,0}$ will be used, for instance accomplishing the inf-sup condition (see next section). We point out these particular cases where used.

A family $\{\Theta_h\}_{h>0}$ of finite element partitions is called regular if there exists a $\varrho_1 > 0$ independent of h such that $\frac{\varrho_K}{h_K} \geq \varrho_1 > 0$, for all $K \in \Theta_h$ and $h > 0$. If $\{\Theta_h\}_{h>0}$ is regular, the following approximation property is satisfied (see [153]): given $\boldsymbol{v} \in \boldsymbol{H}^r(\Omega)$, $r \geq 2$, there exists a finite element interpolation $\pi_h(\boldsymbol{v})$ and constant C_h independent of the mesh size h such that,

$$\|\boldsymbol{v} - \pi_h(\boldsymbol{v})\|_m \leq C_h h^s \|\boldsymbol{v}\|_r, \tag{1.35}$$

for $0 \leq m \leq r$ and $s = \min\{k_v + 1 - m, r - m\}$, where $\pi_h(\boldsymbol{v})$ is the projection operator onto $\mathcal{V}_h$ defined by

$$(\pi_h(\boldsymbol{v}), \boldsymbol{v}_h) = (\boldsymbol{v}, \boldsymbol{v}_h), \ \forall \boldsymbol{v}_h \in \mathcal{V}_h. \tag{1.36}$$

Moreover, $\{\Theta_h\}_{h>0}$ is called *uniformly regular*, or *quasi-uniform*, if there exists a $\varrho_2 > 0$ independent of h such that $\frac{\varrho}{h} \geq \varrho_2 > 0$, as h tends to zero. If $\{\Theta_h\}_{h>0}$ is uniformly regular, the following inverse inequality holds (see [23]): given a $\boldsymbol{v}_h \in \mathcal{V}_h$, there exists a constant C independent of the mesh size h such that,

$$\|\boldsymbol{v}_h\|_1 \leq \frac{C}{h}\|\boldsymbol{v}_h\|_0. \tag{1.37}$$

1.4 The variational formulation of the Navier-Stokes equations

1.4.1 The continuous level

We obtain in this section the variational (or weak) form of the stationary Navier-Stokes equations for incompressible flows. However, all the concepts exposed in this section are easily extendable to the transient case, as well as to other mixed methods. The study of mixed methods was pioneered by Babuŝka in [3], whose ideas where applied to the specific case of the Stokes equations in [24] by Brezzi. A more recent complete text on this field is the book of Brezzi and Fortin [28].

Let $\mathcal{V} \equiv \boldsymbol{H}^1(\Omega)$, $\mathcal{V}_0 \equiv \boldsymbol{H}^1_0(\Omega)$ and $\mathcal{Q} \equiv \boldsymbol{L}^2(\Omega)/\mathbb{R}$ denote the real Hilbert spaces for velocity and pressure, respectively, with associated norms $\|\boldsymbol{v}\|_{\mathcal{V}}$ and $\|q\|_{\mathcal{Q}}$, and $\boldsymbol{f} \in \mathcal{V}'$. Then, the weak form of (1.11)-(1.1b) consist of finding $\boldsymbol{u} \in \mathcal{V}$ and $p \in \mathcal{Q}$ such that:

$$a(\boldsymbol{u}, \boldsymbol{v}) + b(\boldsymbol{v}, p) + c(\boldsymbol{u}, \boldsymbol{u}, \boldsymbol{v}) = \langle \boldsymbol{f}, \boldsymbol{v} \rangle, \qquad \forall \boldsymbol{v} \in \mathcal{V} \tag{1.38a}$$

$$b(\boldsymbol{u}, q) = 0, \qquad \forall q \in \mathcal{Q} \tag{1.38b}$$

where we have used the forms defined above. The study of this problem leads to introduce the operator defined as

$$B : \mathcal{V}_0 \to \mathcal{Q}' \quad | \quad \langle B(\boldsymbol{v}), q \rangle_{\mathcal{Q}' \times \mathcal{Q}} = b(\boldsymbol{v}, q) \quad \forall \boldsymbol{v} \in \mathcal{V}_0, \ \forall q \in \mathcal{Q} \tag{1.39}$$

and B^t its adjoint. Then, the condition in order for (1.38) to be well posed reads as follows: there exists a constant β such that,

$$\inf_{q \in \mathcal{Q}} \sup_{v \in \mathcal{V}_0} \frac{b(\boldsymbol{v}, q)}{\|\boldsymbol{v}\|_{\mathcal{V}} \|q\|_{\mathcal{Q}/\ker B^t}} \geq \beta > 0, \tag{1.40}$$

in which case existence of the solution can be proved. This condition, referred to as inf-sup condition along this work, is usually called the LBB condition honoring the works of Ladyzhenskaya [120], Babuŝka [3] and Brezzi [24].

Ladyzhenskaya proved condition (1.40) for the particular case of the Stokes problem (for which existence and uniqueness are known) in [120].

For the non-linear equation, non-uniqueness of the solution is the general situation. Uniqueness only occurs under very restrictive requirements on the data. Let us state the uniqueness theorem (see [81] or [159]):

Theorem 1.1. *If ν is sufficiently large or $\boldsymbol{f}$ sufficiently small so that*

$$\nu^2 > C\|\boldsymbol{f}\|_{\boldsymbol{J}_1'}$$

then there exists a unique solution of (1.38).

The stationary Navier-Stokes equations have in general more than one solution, unless the data fulfills the restriction stated in the previous theorem. However, in many practical situations these solutions can be isolated, that is to say, there exists a neighborhood in which the solution is unique. Further, the solution depends on the Reynolds number. Thus, as the Reynolds number varies along an interval, each solution of the Navier-Stokes equations describes an isolated branch. This means that the bifurcation phenomena *is rare*. This situation is expressed mathematically by the notion of *branches of nonsingular solutions*. We refer to [81] for the description of this abstract framework.

For the evolutionary Navier-Stokes equations the spaces $(\mathcal{V}_0)_t \equiv L^2(0,T;\boldsymbol{H}_0^1(\Omega))$ and $\mathcal{Q}_t \equiv L^1(0,T;\boldsymbol{L}^2(\Omega)/\mathbb{R})$ denote the real Hilbert spaces for velocity and pressure respectively. Then, the weak form of (1.7)-(1.1b) consists of finding e.g., $\boldsymbol{u} \in (\mathcal{V}_0)_t$ and $p \in \mathcal{Q}_t$ such that:

$$(\partial_t \boldsymbol{u}, \boldsymbol{v}) + a(\boldsymbol{u}, \boldsymbol{v}) + b(\boldsymbol{v}, p) + c(\boldsymbol{u}, \boldsymbol{u}, \boldsymbol{v}) = \langle \boldsymbol{f}, \boldsymbol{v} \rangle, \qquad \forall \boldsymbol{v} \in (\mathcal{V}_0), \tag{1.41a}$$

$$b(\boldsymbol{u}, q) = 0, \qquad \forall q \in \mathcal{Q}. \tag{1.41b}$$

For $2d$ problems, the existence and uniqueness theory is fairly complete. The solution is as regular as allowed by the data and we have continuous dependence of the data in the corresponding function spaces (see [159]).

In the three-dimensional case the $2d$ result can not be extended due to the lack of information concerning the regularity of the weak solution. Only partial results have been proved (see [159]). Regarding *strong* (or classical) solutions, existence and uniqueness have been proved on some time interval depending on the data. It is known since the work of Leray [122] that, provided the data regular enough, there is locally in time a unique smooth solution:

Theorem 1.2 (Local existence and uniqueness of classical solutions)**.** *Being the space dimension $d = 3$, for any $\boldsymbol{u}_0 \in \boldsymbol{J}_1$, $\boldsymbol{f} \in \boldsymbol{J}_0$ and $T > 0$, there is a T_* ($0 < T_* < T$) depending on the data ($\boldsymbol{u}_0$, $\boldsymbol{f}$, ν, T and Ω) such that there exists a unique solution of problem (1.41) in the interval $(0, T_*)$ that fulfills $\boldsymbol{u}$, $\partial_t \boldsymbol{u}$, $\nabla \boldsymbol{u}$, $\nabla \cdot (\nabla \boldsymbol{u}) \in \boldsymbol{L}^2(\Omega \times (0, T_*))$ and $\boldsymbol{u} \in \mathcal{C}^0(0, T_*; \boldsymbol{J}_1)$.*

We do not even know if this solution remains smooth for all time. This theorem proves uniqueness (on some interval $(0, T_*)$) for a class of solutions (the so called *strong* solutions) for which existence is not proved in general. In the $3d$ case existence is known only for *weak* solutions, as stated in the next theorem, due to Hopf [99]:

Theorem 1.3 (Existence of weak solutions)**.** *For any $\boldsymbol{u}_0 \in \boldsymbol{J}_0$, $\boldsymbol{f} \in L^2(0,T;\boldsymbol{H}^{-1}(\Omega))$ and $T > 0$, there exists at least one solution $\boldsymbol{u}$ of problem (1.41) such that $\boldsymbol{u}$, $\nabla \boldsymbol{u} \in L^2(\Omega \times (0,T))$ and $\boldsymbol{u}$ is weakly continous (that is, $\forall \boldsymbol{v} \in \boldsymbol{J}_0$, $t \mapsto (\boldsymbol{u}(\boldsymbol{x}, t), \boldsymbol{v}(\boldsymbol{x}))$ is a continuous scalar function). Furthermore, $\boldsymbol{u}$ satisfies the following energy inequality (*Leray inequality*):*

$$\frac{1}{2}\|\boldsymbol{u}(\boldsymbol{x},t)\|_0^2 + \nu \int_0^t \|\nabla \boldsymbol{u}(\boldsymbol{x},s)\|_0^2 \, \mathrm{d}s \leq \frac{1}{2}\|\boldsymbol{u}(\boldsymbol{x},0)\|_0^2 + \int_0^t \langle \boldsymbol{f}(\boldsymbol{x},s), \boldsymbol{u}(\boldsymbol{x},s) \rangle \, \mathrm{d}s, \tag{1.42}$$

which implies $\boldsymbol{u} \in L^2(0,T;\boldsymbol{J}_1) \cap L^\infty(0,T;\boldsymbol{J}_0)$.

Unfortunately, uniqueness is not proved for the *weak* solutions for which existence is known. There is a gap between the class of solutions for which existence is proved and the smaller class for which uniqueness is proved. Further, it has been proved that any *weak* solution of (1.41) belongs to $L^{8/3}(0,T;\boldsymbol{L}^4(\Omega))$ and any *strong* solution of (1.41) belongs to $L^8(0,T;\boldsymbol{L}^4(\Omega))$. The distance between these two classes of functions seems to be the distance that separate us from the global result of existence and uniqueness.

According to this description of the state-of-the-art there are two fundamental problems that still remain open: uniqueness of weak solutions and existence of strong solutions for any time value. The key importance of these questions have motivated their consideration as one of the seven "millennium problems" proposed by the Clay Mathematics Institute [44] with a reward of 1000000$.

Let us end with an alternative point of view that is summarized with the following question: How should we modify the Navier-Stokes equations in order for weak solutions to be unique?. Lots of works have been devoted to this question. It has motivated the appearance of modified Navier-Stokes equations for which some kind of uniqueness of weak solutions can be proved. Among them, some well-known examples are the mollification model of Leray [122], the hyperviscosity model of Lions [124, 125] or the non-linear viscosity model of Ladyzenskaya and Kaniel [119, 120]. A recent interpretation of these methods as pre-LES (*large eddy simulation*) models can be found in [89, 90].

1.4.2 The discrete approach

We now consider the approximation of the weak form (1.38) using the finite element approximation theory. Let $\mathcal{V}_{h,0}$ and $\mathcal{Q}_h$ be the finite dimensional subspaces approximating $\mathcal{V}_0$ and $\mathcal{Q}$ respectively, where the index h refers to the mesh size. The discrete version of system (1.38) consists of finding $\boldsymbol{u}_h \in \mathcal{V}_{h,0}$ and $p_h \in \mathcal{Q}_h$ such that

$$a(\boldsymbol{u}_h,\boldsymbol{v}_h)+b(\boldsymbol{v}_h,p_h)+c(\boldsymbol{u}_h,\boldsymbol{u}_h,\boldsymbol{v}_h)=\langle \boldsymbol{f},\boldsymbol{v}_h\rangle, \qquad \forall \boldsymbol{v}\in\mathcal{V}_{h,0} \tag{1.43a}$$

$$b(\boldsymbol{u}_h,q_h)=0, \qquad \forall q\in\mathcal{Q}_h. \tag{1.43b}$$

We denote by B_h the equivalent to B at the discrete level, that is

$$B_h:\mathcal{V}_{h,0}\to\mathcal{Q}_h' \quad | \quad \langle B_h(\boldsymbol{v}_h),q_h\rangle_{\mathcal{Q}_h'\times\mathcal{Q}_h}=b(\boldsymbol{v}_h,q_h) \quad \forall \boldsymbol{v}_h\in\mathcal{V}_{h,0},\ \forall q_h\in\mathcal{Q}_h. \tag{1.44}$$

The equivalent version of the inf-sup condition at the discrete level holds if there exists a constant β_d, independent of the mesh size h, such that

$$\inf_{q_h\in\mathcal{Q}_h}\sup_{v_h\in\mathcal{V}_{h,0}}\frac{b(\boldsymbol{v}_h,q_h)}{\|\boldsymbol{v}_h\|_{\mathcal{V}}\|q_h\|_{\mathcal{Q}/\ker B_h^t}}\geq\beta_d>0, \tag{1.45}$$

in which case existence can be proved. Again, the non-uniqueness of the continuous equation is inherited by the discretized version (1.43) using the finite element method. The analysis of the approximation of branches of non-singular solutions is due to Brezzi, Rappaz and Raviart [30, 31, 32].

Unfortunately, condition (1.45) does not hold for simple cases, as equal order velocity-pressure interpolation. The pairs which are stable, that is, accomplish the discrete inf-sup condition (1.45), are called *div-stable* in the terminology of [20]. We refer to [25] for a different approach to the inf-sup condition at the fully discrete level.

The space discretization of the evolutionary system (1.41) consists of finding $\boldsymbol{u}_h \in (\mathcal{V}_{h,0})_t$ and $p_h\in(\mathcal{Q}_h)_t$ such that:

$$(\partial_t\boldsymbol{u}_h,\boldsymbol{v}_h)+a(\boldsymbol{u}_h,\boldsymbol{v}_h)+b(\boldsymbol{v}_h,p_h)+c(\boldsymbol{u}_h,\boldsymbol{u}_h,\boldsymbol{v}_h)=\langle \boldsymbol{f},\boldsymbol{v}_h\rangle, \qquad \forall \boldsymbol{v}_h\in\mathcal{V}_{h,0} \tag{1.46a}$$

$$b(\boldsymbol{u}_h,q_h)=0, \qquad \forall q_h\in\mathcal{Q}_h. \tag{1.46b}$$

being $(\mathcal{V}_{h,0})_t \equiv L^2(0,T;\mathcal{V}_h)$ and $(\mathcal{Q}_h)_t \equiv L^2(0,T;Q_h)$. Existence and uniqueness of the semi-discrete system (1.46) are known.

1.5 Time discretization

Let us introduce some notation that we will use throughout the work. Consider a uniform partition of the time interval of size δt, and let us denote by f^n the approximation of a time dependent function f at time level $t^n = n\delta t$. For a parameter $\theta \in [0,1]$, we will denote

$$\begin{aligned} f^{n+\theta} &= \theta f^{n+1} + (1-\theta) f^n, \\ \delta f^{n+1} &\equiv \delta^{(1)} f^{n+1} = f^{n+1} - f^n, \\ \delta^{(i+1)} f^{n+1} &= \delta^{(i)} f^{n+1} - \delta^{(i)} f^n, \ i = 1,2,3,... \end{aligned}$$

Let us also define

$$D_t(\cdot) = \frac{\delta(\cdot)}{\delta t}. \tag{1.47}$$

The discrete operators $\delta^{(i+1)}$ are centered. We will also use the backward difference operators

$$\begin{aligned} D_k f^{n+1} &= \frac{1}{\gamma_k}(f^{n+1} - \sum_{i=0}^{k-1} \alpha_k^i f^{n-i}), \\ D_1 f^{n+1} &= \delta f^{n+1} = f^{n+1} - f^n, \\ D_2 f^{n+1} &= \frac{3}{2}(f^{n+1} - \frac{4}{3} f^n + \frac{1}{3} f^{n-1}), \\ D_3 f^{n+1} &= \frac{11}{6}(f^{n+1} - \frac{18}{11} f^n + \frac{9}{11} f^{n-1} - \frac{2}{11} f^{n-2}), \end{aligned}$$

as well as the backward extrapolation operators

$$\begin{aligned} \widetilde{f}_i^{n+1} &= f^{n+1} - \delta^{(i)} f^{n+1} = f^{n+1} + \mathcal{O}(\delta t^i), \\ \widetilde{f}_1^{n+1} &= f^n, \\ \widetilde{f}_2^{n+1} &= 2f^n - f^{n-1}, \\ \widetilde{f}_3^{n+1} &= 3f^n - 3f^{n-1} + f^{n-2}. \end{aligned}$$

For the time integration of problem (1.46) we consider two sorts of finite difference approximations. The first is the generalized trapezoidal rule, which consists of solving the following problem: from known $\boldsymbol{u}^n$, find $\boldsymbol{u}^{n+1} \in \mathcal{V}_0$ and $p^{n+1} \in \mathcal{Q}$ such that

$$(\frac{1}{\delta t} D_1 \boldsymbol{u}^{n+1} + \boldsymbol{u}^{n+\theta} \cdot \nabla \boldsymbol{u}^{n+\theta}, \boldsymbol{v}) + \nu(\nabla \boldsymbol{u}^{n+\theta}, \nabla \boldsymbol{v}) - (p^{n+\theta}, \nabla \cdot \boldsymbol{v}) = \langle \bar{\boldsymbol{f}}^{n+\theta}, \boldsymbol{v} \rangle, \tag{1.48a}$$

$$(q, \nabla \cdot \boldsymbol{u}^{n+\theta}) = 0, \tag{1.48b}$$

for all $(\boldsymbol{v}, q) \in \mathcal{V}_0 \times \mathcal{Q}$. The force term $\bar{\boldsymbol{f}}^{n+\theta}$ in (1.48a) and below has to be understood as the time average of the force in the interval $[t^n, t^{n+1}]$, even though we use a superscript $n+\theta$ to characterize it. The pressure value computed here has been identified as the pressure evaluated at $t^{n+\theta}$, although this is irrelevant for the velocity approximation. The

values of interest of θ are $\theta = 1/2$, corresponding to the second order Crank-Nicolson scheme, and $\theta = 1$, which corresponds to the backward Euler method.

Backward differencing (BDF) time integration schemes will be suggested for the methods proposed hereafter. The first order one (BDF1) coincides with Backward Euler method. BDF1 and the second order scheme BDF2 are $\mathcal{A}$-stable. $\mathcal{A}$-stability is based on the *Dahlquist test equation*:

$$\frac{\mathrm{d}y}{\mathrm{d}t} = \lambda y.$$

This property implies that $|y(t)| \leq |y(0)|$ for $t \geq 0$ if $\lambda \in \mathbb{C}^-$, being

$$\mathbb{C}^- := \{z \in \mathbb{C} \,|\, \mathrm{Re}(z) \leq 0\}, \tag{1.49}$$

where $\mathrm{Re}(z)$ stands for the real part of the complex number z. For BDF time integration schemes, given the test equation

$$\frac{D_k y^{n+1}}{\delta t} = \lambda y^{n+1} \tag{1.50}$$

with $\lambda \in \mathbb{C}^-$, $\mathcal{A}$-stability implies that $|y^{n+1}| \leq |y^n|$.

Higher order methods do not keep this interesting property anymore, limitation known as the *second Dahlquist barrier*. Nevertheless, BDF3 holds a less demanding $\mathcal{A}(\alpha)$-stability, with $\alpha = 86°$, that makes this method appropriate. $\mathcal{A}(\alpha)$-stability means that $|y^{n+1}| \leq |y^n|$ for any $\lambda \in \mathbb{C}^-$ such that:

$$\left|\arctan\left(\frac{\mathrm{Re}(\lambda)}{\mathrm{Im}(\lambda)}\right)\right| \leq \alpha,$$

where $\mathrm{Im}(\lambda)$ stands for the imaginary part of λ. See [112] for a complete exposition of BDF methods and their stability properties.

For the second order scheme BDF2, $\boldsymbol{u}^1$ can be computed using the backward Euler method, whereas for $n \geq 1$ the unknowns $\boldsymbol{u}^{n+1} \in \mathcal{V}_0$ and $p^{n+1} \in \mathcal{Q}$ are found by solving the problem

$$(\frac{1}{\delta t} D_2 \boldsymbol{u}^{n+1} + \boldsymbol{u}^{n+1} \cdot \nabla \boldsymbol{u}^{n+1}, \boldsymbol{v}) + \nu(\nabla \boldsymbol{u}^{n+1}, \nabla \boldsymbol{v}) - (p^{n+1}, \nabla \cdot \boldsymbol{v}) = \langle \bar{\boldsymbol{f}}^{n+1}, \boldsymbol{v} \rangle, \tag{1.51a}$$

$$(q, \nabla \cdot \boldsymbol{u}^{n+1}) = 0, \tag{1.51b}$$

for all $(\boldsymbol{v}, q) \in \mathcal{V}_0 \times \mathcal{Q}$.

The BDF3, which requires also $\boldsymbol{u}^2$ for being initialized, has been analyzed in [9] for the time discretization of the incompressible Navier-Stokes problem.

For a kth order BDF method, the unknowns $\boldsymbol{u}^{n+1} \in \mathcal{V}_0$ and $p^{n+1} \in \mathcal{Q}$, with $n+1 \geq k$, are found by solving the problem

$$(\frac{1}{\delta t} D_k \boldsymbol{u}^{n+1} + \boldsymbol{u}^{n+1} \cdot \nabla \boldsymbol{u}^{n+1}, \boldsymbol{v}) + \nu(\nabla \boldsymbol{u}^{n+1}, \nabla \boldsymbol{v}) - (p^{n+1}, \nabla \cdot \boldsymbol{v}) = \langle \boldsymbol{f}^{n+1}, \boldsymbol{v} \rangle, \tag{1.52a}$$

$$(q, \nabla \cdot \boldsymbol{u}^{n+1}) = 0, \tag{1.52b}$$

for all $(\boldsymbol{v}, q) \in \mathcal{V}_0 \times \mathcal{Q}$.

Chapter 2

Stabilization with Orthogonal Subscales

In this chapter we present the Orthogonal Subgrid Scale *stabilized finite element method that will be used and analyzed along this work. This section is a summary of Codina's works devoted to this method, specially [49].*

2.1 Introduction

The numerical approximation of the Navier-Stokes equations seems straightforward using the Galerkin approximation. Unfortunately, the standard Galerkin approach fails for two reasons: the effect of the convective term when this one is dominant and the compatibility required for the velocity and pressure finite element spaces posed by the discrete inf-sup condition (1.45). The finite elements satisfying this condition are complicated and computationally expensive in practice (see [28]). These two kinds of instabilities have motivated lots of works on the design of numerical methods stabilizing the convective term and allowing equal velocity-pressure interpolation pairs. Both can be overcome by the use of *stabilized* formulations, which will be used in this work. These methods are based on the modification of the weak form obtained by the Galerkin approach by the addition of mesh-dependent terms weighted by the residuals of the differential equations.

Stabilized finite element methods were initially developed in the context of convective flows. It was already known that the numerical oscillations that appeared for convection dominated cases could be avoided introducing numerical diffusion (see [163] in the context of finite differences). The next step was the introduction of this artificial diffusion only along the streamlines in the original works [103] and [117], obtaining less overdiffusive schemes. The widely known SUPG method was introduced in [33], where the streamline diffusion idea was put in the context of weighted residual methods.

Later, Hughes *et al.* [105] observed for the Stokes problem that if the pressure gradient is treated as a convective term and the SUPG technique is used, the discrete inf-sup condition can be avoided. This fact motivated the consideration of the whole Stokes operator applied to the test function as stabilization term in [106] and [76]. This ideas led to the popular Galerkin/Least Squares (GLS) method suggested by Hughes *et al.* in [107]. Other examples of similar stabilized methods are the Characteristic Galerkin method (see [65]), least square methods for first-order systems as those in [18] and the Taylor-Galerkin method (see [63]).

In [102] Hughes introduces a procedure for developing methods capable of dealing with

multiscale phenomena, later called the *variational multiscale method* in [104]. The key idea is *to approximate the effect of the scales that can not be resolved by the finite element mesh on the discrete finite element solution.* It has helped to elucidate the origins of stabilized finite element methods and provide a variational framework for subgrid scale models. This setting led to the subgrid scale method (SGS) that differs slightly from the previous GLS method.

In the context of this work we use and analyze the *orthogonal subgrid scales method* (OSS). This method is based on the subgrid scale concept introduced by Hughes. The key idea of this methodology is to assume the *subgrid component to be L^2-orthogonal to the finite element space.*

The seed of this method was the study of some fractional step schemes that use a Poisson equation to compute the pressure (see Chapter 3). In [53] it was designed a stabilized method for the Stokes problem that inherited the inherent stability mechanism of first order projection methods (see [47]). This method was extended to the non-linear case in [54]. In [46] this stabilization technique is also applied to the convection-diffusion equation.

The OSS approach was first introduced by Codina in [48] as an extension of the stabilized method introduced for the Stokes problem and the convection-diffusion equation. The stabilization of both convection dominated flows and pressure together with the goal of dealing with transient problems was elaborated in [49].

Most of the existing stabilization techniques cannot be extended to transient problems without the framework of discontinuous Galerkin or space-time finite element formulations. However, the OSS method is easily extended to transient problems and the effect of the time discretization over the stabilization method is clearly explained (see [49]).

When using the *subgrid* concept, some *modeling* assumptions are needed in order to solve the problem for the subscales. The *modeling assumption* used for the OSS approach by Codina in [49] is based on a Fourier analysis of the subscales equation. Different options have been proposed in the literature; for instance, the approximation of the subgrid solution using a finer finite element space and the introduction of numerical diffusion for the subgrid system, as suggested in [86, 108]. Likewise, when using bubble functions for the approximation of the subscales, as in [8, 26, 29, 36, 77, 142, 134], solving for them is what can be considered the modeling step. A recent alternative method is the *multiscale discontinuous Galerkin* approach introduced in [111], where the coarse scale is considered continuous and the subgrid scale discontinuous.

The OSS method has been widely analyzed in different situations by Codina, Blasco and Badia. In [54] the pressure stabilization introduced by this method was fully analyzed for the stationary Navier-Stokes equations. Its extension to the transient case was developed in [15]. Further, the approximation of a first order pressure segregation method using OSS has been recently analyzed in [5] (see Chapter 4). In [55], the finite element approximation of the transient convection-diffusion-reaction equation using orthogonal subscales is analyzed for the BDF1 time integration scheme. More recently, in [4] we analyze the stabilized approximation of the transient convection-diffusion equation in an ALE framework for first and second order time integration schemes (see also Chapter 6). Therein we also obtain convergence and stability results of the transient convection-diffusion equation using BDF2 as time integrator.

A similar approach is the *nonlinear Galerkin* method (see [1, 34, 35, 73, 114, 126]). The key idea of this method is to decompose the solution into large and small scales in space and use a different treatment for them in time, generally with a coarser time approximation for the small scales. Again, the goal is to obtain a numerical model whose

computational cost is the computation of the large scales, but in which the subgrid scales are accounted for using some *model*.

For a complete exposition of stabilization techniques (within the framework of the diffusion-convection-reaction equation) we refer to [45].

Let us organize the present chapter. In Section 2.2 we introduce the OSS stabilization technique for the stationary and linearized form of the Navier-Stokes equations, that is, replacing the momentum equation by,

$$-\nu\Delta\boldsymbol{u} + \boldsymbol{a}\cdot\nabla\boldsymbol{u} + \nabla p = \boldsymbol{f} \quad \text{in } \Omega, \tag{2.1}$$

(and of course dropping the initial condition), where $\boldsymbol{a}$ is a given solenoidal velocity field.

In Section 2.3 these concepts are applied to a fully discrete transient problem. Again, the linearized version of the momentum equation is considered,

$$\partial_t\boldsymbol{u} - \nu\Delta\boldsymbol{u} + \boldsymbol{a}\cdot\nabla\boldsymbol{u} + \nabla p = \boldsymbol{f} \quad \text{in } \Omega,\ t\in(0,T). \tag{2.2}$$

Finally, the stabilization technique is applied for the transient (nonlinear) Navier-Stokes equations in Section 2.4. This can be easily done after the linearization of the problem.

2.2 Stationary Oseen equations

2.2.1 Problem statement

In this section we consider the linear and stationary problem (2.1)-(1.1b), supplied with the homogeneous Dirichlet condition for the velocity field.

Let us introduce the additional spaces $\mathcal{W}_0 := \mathcal{V}_0 \times \mathcal{Q}$, and $\mathcal{W} := \mathcal{V} \times \mathcal{Q}$.

Let $\boldsymbol{U} \equiv [\boldsymbol{u}, p] \in \mathcal{W}_0$. The equations to be solved can be written as

$$\mathcal{L}(\boldsymbol{U}) := \begin{bmatrix} -\nu\Delta\boldsymbol{u} + \boldsymbol{a}\cdot\nabla\boldsymbol{u} + \nabla p \\ \nabla\cdot\boldsymbol{u} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ 0 \end{bmatrix} =: \boldsymbol{F} \tag{2.3}$$

in the domain Ω and $\boldsymbol{u} = \boldsymbol{0}$ on Γ. Let $\boldsymbol{V} \equiv [\boldsymbol{v}, q] \in \mathcal{W}_0$. The variational statement for problem (2.3) can be written in terms of the bilinear form defined on $\mathcal{W}_0 \times \mathcal{W}_0$ as

$$B(\boldsymbol{U}, \boldsymbol{V}) := \nu(\nabla\boldsymbol{u}, \nabla\boldsymbol{v}) + (\boldsymbol{a}\cdot\nabla\boldsymbol{u}, \boldsymbol{v}) - (p, \nabla\cdot\boldsymbol{v}) + (q, \nabla\cdot\boldsymbol{u}) \tag{2.4}$$

and the linear form $L(\boldsymbol{V}) := \langle \boldsymbol{f}, \boldsymbol{v}\rangle$. Problem (2.3) with the homogeneous Dirichlet condition consists then in finding $\boldsymbol{U} \in \mathcal{W}_0$ such that

$$B(\boldsymbol{U}, \boldsymbol{V}) = L(\boldsymbol{V}), \quad \forall\ \boldsymbol{V} \in \mathcal{W}_0. \tag{2.5}$$

The standard Galerkin approximation of this abstract variational problem is now straightforward. Using the finite element spaces introduced in Section 1.3 and the additional finite element space $\mathcal{W}_h = \mathcal{V}_h \times \mathcal{Q}_h$ and $\mathcal{W}_{h,0} = \mathcal{V}_{h,0} \times \mathcal{Q}_h$, the discrete version of problem (2.5) is: find $\boldsymbol{U}_h \in \mathcal{W}_{h,0}$ such that

$$B(\boldsymbol{U}_h, \boldsymbol{V}_h) = L(\boldsymbol{V}_h), \quad \forall\ \boldsymbol{V}_h \in \mathcal{W}_{h,0}. \tag{2.6}$$

The well posedness of this problem relies on the ellipticity of the viscous term and the inf–sup or Babuŝka–Brezzi condition (see [28] and Section 1.4), which can be shown to hold for the continuous problem. The first property is automatically inherited by its discrete

counterpart. However, the inf–sup condition needs to be explicitly required. This leads to the need of using mixed interpolations, that is, different for $\boldsymbol{u}$ and p, and verifying

$$\inf_{q_h \in \mathcal{Q}_h} \sup_{\boldsymbol{v}_h \in \mathcal{V}_{h,0}} \frac{(q_h, \nabla \cdot \boldsymbol{v}_h)}{\|q_h\|_{\mathcal{Q}} \, \|\boldsymbol{v}_h\|_{\mathcal{V}}} \geq \beta_d > 0, \tag{2.7}$$

for a constant β_d independent of h.

Convenient velocity-pressure interpolations, such as *equal* interpolation, turn out to violate condition (2.7). This is why many of the so called *stabilized* formulations have been proposed to approximate problem (2.5). The idea is to replace (2.6) by another discrete variational problem in which the bilinear form B is replaced by a possibly mesh dependent bilinear form B_h with enhanced stability properties. Likewise, it has already been mentioned that instability problems may arise when the convective term dominates the viscous one. Both this and the need to satisfy (2.7) can be overcome by using the finite element formulation described next.

2.2.2 The subgrid scale approach

Let $\mathcal{W} = \mathcal{W}_h \oplus \tilde{\mathcal{W}}$, where $\tilde{\mathcal{W}}$ is any space to complete $\mathcal{W}_h$ in $\mathcal{W}$. Obviously, $\tilde{\mathcal{W}}$ is infinite-dimensional, but once the final method will be formulated, it will be approximated by a finite-dimensional space (cf. Remark 2.1 below), although we will keep the same symbol $\tilde{\mathcal{W}}$ for it. The elements of this space are denoted by $\tilde{\boldsymbol{V}} = [\tilde{\boldsymbol{v}}, \tilde{q}]$. Likewise, let $\mathcal{W}_0 = \mathcal{W}_{h,0} \oplus \tilde{\mathcal{W}}_0$, with $\tilde{\mathcal{W}}_0$ any complement of $\mathcal{W}_{h,0}$ in $\mathcal{W}_0$. The space $\tilde{\mathcal{W}}_0$ will be called the space of *subgrid scales* or *subscales*.

The continuous problem is equivalent to find $\boldsymbol{U}_h \in \mathcal{W}_{h,0}$ and $\tilde{\boldsymbol{U}} \in \tilde{\mathcal{W}}_0$ such that

$$B(\boldsymbol{U}_h, \boldsymbol{V}_h) + B(\tilde{\boldsymbol{U}}, \boldsymbol{V}_h) = L(\boldsymbol{V}_h) \quad \forall \ \boldsymbol{V}_h \in \mathcal{W}_{h,0}, \tag{2.8}$$

$$B(\boldsymbol{U}_h, \tilde{\boldsymbol{V}}) + B(\tilde{\boldsymbol{U}}, \tilde{\boldsymbol{V}}) = L(\tilde{\boldsymbol{V}}) \quad \forall \ \tilde{\boldsymbol{V}} \in \tilde{\mathcal{W}}_0. \tag{2.9}$$

Integrating by parts within each element in (2.8)-(2.9), it is found that these two equations can be written as

$$B(\boldsymbol{U}_h, \boldsymbol{V}_h) + \sum_K \int_K \tilde{\boldsymbol{U}} \cdot \mathcal{L}^*(\boldsymbol{V}_h) \, \mathrm{d}\Omega + \sum_K \int_{\partial K} \tilde{\boldsymbol{u}} \cdot (\nu \boldsymbol{n} \cdot \nabla \boldsymbol{v}_h) \, \mathrm{d}\Gamma = L(\boldsymbol{V}_h), \tag{2.10}$$

$$\sum_K \int_{\partial K} \tilde{\boldsymbol{v}} \cdot (p\boldsymbol{n} + \nu \boldsymbol{n} \cdot \nabla \boldsymbol{u}) \, \mathrm{d}\Gamma + \sum_K \int_K \tilde{\boldsymbol{V}} \cdot \mathcal{L}(\tilde{\boldsymbol{U}}) \, \mathrm{d}\Omega = \sum_K \int_K \tilde{\boldsymbol{V}} \cdot [\boldsymbol{F} - \mathcal{L}(\boldsymbol{U}_h)] \, \mathrm{d}\Omega, \tag{2.11}$$

where $\sum_K$ stands for the summation over all $K \in \Theta_h$, $\boldsymbol{n}$ is the unit normal exterior to the integration domain, and $\mathcal{L}^*$ is the formal adjoint of $\mathcal{L}$, given by

$$\mathcal{L}^*(\boldsymbol{V}_h) = \begin{bmatrix} -\nu \Delta \boldsymbol{v}_h - \boldsymbol{a} \cdot \nabla \boldsymbol{v}_h - \nabla q_h \\ -\nabla \cdot \boldsymbol{v}_h \end{bmatrix}. \tag{2.12}$$

Assuming that the exact tractions are continuous across inter-element boundaries, the first term of (2.11) vanishes. This equation is then equivalent to:

$$\mathcal{L}(\tilde{\boldsymbol{U}}) = \boldsymbol{R} := \boldsymbol{F} - \mathcal{L}(\boldsymbol{U}_h) + \boldsymbol{V}_{h,\mathrm{ort}} \ \text{ in } K \in \Theta_h, \ \ \boldsymbol{V}_{h,\mathrm{ort}} \in \tilde{\mathcal{W}}_0^{\perp}, \tag{2.13}$$

which must be satisfied together with boundary conditions on ∂K that are unknown, but who must ensure in particular the continuity of the diffusive fluxes across interior boundaries.

The element $\boldsymbol{V}_{h,\text{ort}}$ appearing in (2.13) is responsible to guarantee that $\mathcal{L}(\tilde{\boldsymbol{U}}) - [\boldsymbol{F} - \mathcal{L}(\boldsymbol{U}_h)]$ belongs to $\tilde{\mathcal{W}}_0^{\perp}$, that is what (2.11) (without the first term) implies. Its expression depends on the choice of the space $\tilde{\mathcal{W}}_0$, and will be determined for the particular option that we will use later on. Here and below, orthogonality is understood with respect to the L^2-inner product, unless otherwise specified.

The idea now is to approximate the solution of (2.13) with the appropriate boundary conditions by

$$\tilde{\boldsymbol{U}} \approx \boldsymbol{\tau}_K \boldsymbol{R} \quad \text{in } K \in \Theta_h, \tag{2.14}$$

where $\boldsymbol{\tau}_K$ is a matrix of algorithmic parameters depending on K and the coefficients of the operator $\mathcal{L}$. This approximation for $\tilde{\boldsymbol{U}}$ is intended to mimic the effect of the exact subscales in the volume integral of (2.10), whereas the integral over the element faces will be neglected. Observe that *the pointwise values of* $\tilde{\boldsymbol{U}}$ *are not needed*, and thus (2.14) needs not to be understood pointwise. If the coefficients of $\mathcal{L}$ are constant, only the moments of $\tilde{\boldsymbol{U}}$ appear in (2.10) ($\mathcal{L}^*(\boldsymbol{V}_h)$ is a polynomial).

Matrix $\boldsymbol{\tau}_K$ in (2.14) will be called the *matrix of stabilization parameters*. Its design is one of the cornerstones in the development of stabilized finite element methods, many of which can be formulated in the previous framework [102, 45]. An expression of the stabilization parameters for the convection-diffusion-reaction equation is suggested by Codina in [49] using *heuristic* arguments. In particular, he analyzes the behavior of these parameters from a Fourier analysis of the problem for the subscales. If the equation to be solved is

$$-\nu\Delta\boldsymbol{u} + \boldsymbol{a}\cdot\nabla\boldsymbol{u} + \sigma\boldsymbol{u} = \boldsymbol{f} \tag{2.15}$$

with $\sigma \geq 0$ the reaction coefficient, Codina concludes therein that *if we take*

$$\tau = \left[\left(c_1\frac{\nu}{h^2}\right)^2 + \sigma^2 + \left(c_2\frac{|\boldsymbol{a}|}{h}\right)^2\right]^{-1/2} \tag{2.16}$$

then, there exist values of c_1 *and* c_2 *independent of* h *for which Eq. (2.14) holds, in the sense that both* $\tilde{\boldsymbol{U}}$ *and* $\boldsymbol{\tau}_K\boldsymbol{R}$ *have (approximately) the same* L^2*-norm over element* K. Moreover, c_1 is independent of the coefficients ν, σ and $\boldsymbol{a}$, whereas c_2 depends only on the direction of $\boldsymbol{a}$, but not on its magnitude.

The stabilization parameter in Eq. (2.16) behaves asymptotically in h, ν, σ and $|\boldsymbol{a}|$ as

$$\tau \sim \left[c_1\frac{\nu}{h^2} + \sigma + c_2\frac{|\boldsymbol{a}|}{h}\right]^{-1}.$$

This expression was previously proposed in [45] using a completely different reasoning.

The previous expressions are straightforwardly applicable to any linear system that can be understood as a convection-diffusion-reaction equation. Particularly, we can apply these ideas to the Oseen equation we treat in this section. The adjoint of $\mathcal{L}$ is now given by (2.12) and the matrix of stabilization parameters by

$$\boldsymbol{\tau}_K = \text{diag}(\boldsymbol{\tau}_{1,K}, \tau_{2,K}), \quad \boldsymbol{\tau}_{1,K} = \tau_{1,K}\boldsymbol{I}_d, \tag{2.17}$$

where $\boldsymbol{I}_d$ is the $d \times d$ identity matrix and with $\tau_{1,K}$ and $\tau_{2,K}$ computed elementwise as

$$\tau_{1,K} = \left[c_1 \frac{\nu}{h^2} + c_2 \frac{|\boldsymbol{a}|_{\infty,K}}{h} \right]^{-1}, \tag{2.18a}$$

$$\tau_2 = \frac{h^2}{c_1 \tau_1}. \tag{2.18b}$$

This relationship between τ_1 and τ_2 was also found in [48] *based only on the convergence analysis* of the Oseen problem and using a stabilization technique similar to the Galerkin/least-squares method, found by dropping $\boldsymbol{V}_{h,\text{ort}}$ in (2.13).

Matrix $\boldsymbol{\tau}_K$ defined in (2.17) is symmetric and positive–definite, a requirement needed for (2.24) to be an inner product.

2.2.3 Orthogonal subscales

The starting point of the developments in [49] have been the decompositions $\mathcal{W} = \mathcal{W}_h \oplus \tilde{\mathcal{W}}$ and $\mathcal{W}_0 = \mathcal{W}_{h,0} \oplus \tilde{\mathcal{W}}_0$. If $\cong$ denotes an isomorphism between two vector spaces, we have that $\tilde{\mathcal{W}} \cong \mathcal{W}_h^\perp \cap \mathcal{W}$ and $\tilde{\mathcal{W}}_0 \cong \mathcal{W}_{h,0}^\perp \cap \mathcal{W}_0$. Nevertheless, there are many possibilities to choose $\tilde{\mathcal{W}}$ and $\tilde{\mathcal{W}}_0$. *The particular one adopted in [49] is to take precisely*

$$\tilde{\mathcal{W}} = \mathcal{W}_h^\perp \cap \mathcal{W}. \tag{2.19}$$

Note that $\mathcal{W}_h^\perp$ is not closed in $\mathcal{W}$, but it will be a closed subspace of the final approximating space.

To obtain a feasible numerical method we need to introduce some approximations. The first concerns the choice for $\tilde{\mathcal{W}}_0$. First, we assume that functions in $\tilde{\mathcal{W}}$ already vanish on $\partial\Omega$, and thus $\tilde{\mathcal{W}}_0 \approx \tilde{\mathcal{W}}$. Additionally we assume that $\tilde{\mathcal{W}} \approx \mathcal{W}_h^\perp$, which can be thought of as a non-conforming approximation for the subscales. Altogether, this amounts to saying that

$$\tilde{\mathcal{W}}_0 \approx \tilde{\mathcal{W}} \approx \mathcal{W}_h^\perp. \tag{2.20}$$

With this approximation, it follows from (2.13) that

$$\boldsymbol{V}_{h,\text{ort}} \in \tilde{\mathcal{W}}_0^\perp \approx \mathcal{W}_h, \tag{2.21}$$

$$\tilde{\boldsymbol{U}} \in \tilde{\mathcal{W}}_0 \approx \mathcal{W}_h^\perp, \tag{2.22}$$

which means that $\boldsymbol{V}_{h,\text{ort}}$ is a finite element function and therefore *numerically computable.* We refer to this particular choice for the space of $\tilde{\boldsymbol{U}}$, motivated by the election (2.19) and the approximation (2.20), as the *space of orthogonal subscales.*

Imposing condition (2.22) in expression (2.14) for $\tilde{\boldsymbol{U}}$ we have that

$$\begin{aligned}(\tilde{\boldsymbol{U}}, \boldsymbol{V}_h) &= \sum_K (\boldsymbol{\tau}_K [\boldsymbol{F} - \mathcal{L}(\boldsymbol{U}_h)], \boldsymbol{V}_h) + \sum_K (\boldsymbol{\tau}_K \boldsymbol{V}_{h,\text{ort}}, \boldsymbol{V}_h) \\ &= 0 \quad \forall\ \boldsymbol{V}_h \in \mathcal{W}_h. \end{aligned} \tag{2.23}$$

Let us assume that *matrices $\boldsymbol{\tau}_K$ are all symmetric and positive–definite.* From (2.23) it follows that $\boldsymbol{V}_{h,\text{ort}}$ *is the projection of the residual $\mathcal{L}(\boldsymbol{U}_h) - \boldsymbol{F}$ onto the finite element space with respect to the L^2 inner product weighted element by element by the matrices*

of algorithmic parameters $\boldsymbol{\tau}_K$. We denote this weighted inner product and its associated norm by

$$(\boldsymbol{X},\boldsymbol{Y})_\tau := \sum_K (\boldsymbol{\tau}_K \boldsymbol{X},\boldsymbol{Y})_K = \sum_K (\boldsymbol{X},\boldsymbol{\tau}_K \boldsymbol{Y})_K, \tag{2.24}$$

$$\|\boldsymbol{Y}\|_\tau := \sqrt{(\boldsymbol{Y},\boldsymbol{Y})_\tau}. \tag{2.25}$$

In these expressions, the functions $\boldsymbol{X}$ are $\boldsymbol{Y}$ need not being continuous for the local L^2-products to make sense.

Equation (2.23) now becomes

$$(\boldsymbol{F} - \mathcal{L}(\boldsymbol{U}_h), \boldsymbol{V}_h)_\tau + (\boldsymbol{V}_{h,\mathrm{ort}}, \boldsymbol{V}_h)_\tau = 0, \quad \forall\ \boldsymbol{V}_h \in \mathcal{W}_h. \tag{2.26}$$

If we call Π_τ the projection onto $\mathcal{W}_h$ associated to the inner product (2.24), hereafter referred to as *τ-projection*, we see that

$$\boldsymbol{V}_{h,\mathrm{ort}} = -\Pi_\tau[\boldsymbol{F} - \mathcal{L}(\boldsymbol{U}_h)], \tag{2.27}$$

Likewise, we will denote by $\Pi_{\tau,0}$ the τ-projection onto $\mathcal{W}_{h,0}$ and $\Pi_\tau^\perp := I - \Pi_\tau$, where I is the identity in $\boldsymbol{L}^2(\Omega)$.

From (2.14) and (2.27) it follows that

$$\tilde{\boldsymbol{U}} = \boldsymbol{\tau}_K \Pi_\tau^\perp[\boldsymbol{F} - \mathcal{L}(\boldsymbol{U}_h)] \quad \text{in } K \in \Theta_h. \tag{2.28}$$

If this expression is now introduced in (2.10) and, as already mentioned, the integrals over the interelement boundaries are neglected, we finally obtain the modified discrete problem: find $\boldsymbol{U}_h \in \mathcal{W}_{h,0}$ such that

$$B_h(\boldsymbol{U}_h, \boldsymbol{V}_h) = \langle \boldsymbol{F}, \boldsymbol{V}_h\rangle - (\Pi_\tau^\perp(\boldsymbol{F}), \mathcal{L}^*(\boldsymbol{V}_h))_\tau, \quad \forall\ \boldsymbol{V}_h \in \mathcal{W}_{h,0}, \tag{2.29}$$

where the stabilized bilinear form B_h is

$$B_h(\boldsymbol{U}_h, \boldsymbol{V}_h) = B(\boldsymbol{U}_h, \boldsymbol{V}_h) - (\Pi_\tau^\perp[\mathcal{L}(\boldsymbol{U}_h)], \mathcal{L}^*(\boldsymbol{V}_h))_\tau. \tag{2.30}$$

We introduce further simplifying assumptions that lead to a method easy to implement and with good stability properties. These are:

- The weighted projection Π_τ associated to the inner product defined in (2.24) will be approximated by the L^2-projection, denoted by Π. Likewise, $\Pi_\tau^\perp$ will be approximated by $\Pi^\perp = I - \Pi$. The difference between Π_τ and Π depends on the variation of the stabilization parameters from element to element. From the computational point of view, it is very convenient to use Π, since L^2 projections can be computed very efficiently.

- $\tau_{1,K}\Pi^\perp(\boldsymbol{f}) = \boldsymbol{0}$, which means that the force vector belongs to the finite element space $\mathcal{W}_h$ or it is approximated by an element of this space. In any case, the term $\tau_{1,K}\Pi^\perp(\boldsymbol{f})$ is of the same order as the optimal error that can be expected. Taking for example $\boldsymbol{a} = \boldsymbol{0}$, for $\boldsymbol{f} \in \boldsymbol{H}^m(\Omega)$, $m = -1, 0, ...$, we may expect $\boldsymbol{u} \in \boldsymbol{H}^{m+2}(\Omega)$, $p \in H^{m+1}(\Omega)$ and an L^2 velocity error of order $\mathcal{O}(h^r)$, with $r = \min\{m+2, p+1\}$ and p the order of the finite element interpolation, and this is precisely the order of $\tau_{1,K}\Pi^\perp(\boldsymbol{f})$.

- Second order derivatives of finite element functions within element interiors will be neglected. They are exactly zero for linear elements and for higher order interpolations disregarding them leads to a method which is still consistent (in a sense explained later; cf. Remark 2.2).

Under these conditions, the second term in the RHS of (2.29) vanishes and the stabilized bilinear form (2.30) reduces to

$$B_I(\boldsymbol{U}_h,\boldsymbol{V}_h) = B(\boldsymbol{U}_h,\boldsymbol{V}_h) + (\Pi^{\perp}(\boldsymbol{a}\cdot\nabla\boldsymbol{u}_h+\nabla p_h),\boldsymbol{a}\cdot\nabla\boldsymbol{v}_h+\nabla q_h)_{\tau_1} + (\Pi^{\perp}(\nabla\cdot\boldsymbol{u}_h),\nabla\cdot\boldsymbol{v}_h)_{\tau_2}, \tag{2.31}$$

where B is defined in (2.4).

Once arrived to (2.31) it is observed that what the present method provides with respect to the standard Galerkin method is a *least–squares control on the component of the terms* $\boldsymbol{a}\cdot\nabla\boldsymbol{u}_h+\nabla p_h$ *and* $\nabla\cdot\boldsymbol{u}_h$ *orthogonal to the corresponding finite element spaces.*

There is a simple modification of the bilinear form (2.31) which leads to another stabilized method with slightly better stability properties. The idea is to control separately the components of $\boldsymbol{a}\cdot\nabla\boldsymbol{u}_h$ and ∇p_h orthogonal to $\mathcal{V}_h$. The bilinear form associated to this method is

$$B_{II}(\boldsymbol{U}_h,\boldsymbol{V}_h) = B(\boldsymbol{U}_h,\boldsymbol{V}_h) + (\Pi^{\perp}(\boldsymbol{a}\cdot\nabla\boldsymbol{u}_h),\boldsymbol{a}\cdot\nabla\boldsymbol{v}_h)_{\tau_1} + (\Pi^{\perp}(\nabla p_h),\nabla q_h)_{\tau_1} + (\Pi^{\perp}(\nabla\cdot\boldsymbol{u}_h),\nabla\cdot\boldsymbol{v}_h)_{\tau_2}. \tag{2.32}$$

Remark 2.1. Equation (2.28), together with (2.21) and (2.22), indirectly determine the finite-dimensional space $\tilde{\mathcal{W}}$ for the subscales. The set

$$\left\{\boldsymbol{\tau}_t\Pi^{\perp}(\mathcal{L}(\boldsymbol{F}))|_K\ ,\ \boldsymbol{\tau}_t\Pi^{\perp}(\mathcal{L}(\boldsymbol{V}_h))|_K\ ,\ \boldsymbol{V}_h\in\mathcal{W}_h,\ K\in\Theta_h\right\}$$

is not a basis for the space of subscales, but, according to (2.28), it spans this space, that consists of piecewise discontinuous functions generated by functions in $\mathcal{W}_h$.

Remark 2.2. There is a way to formulate the present method in a manner that it can be viewed as *consistent*. Indeed, if we introduce

$$B_I^{\star}([\boldsymbol{u}_h,p_h,\boldsymbol{\xi}_h,\delta_h],[\boldsymbol{v}_h,q_h,\boldsymbol{\eta}_h,\gamma_h]) := B([\boldsymbol{u}_h,p_h],[\boldsymbol{v}_h,q_h]) + (\boldsymbol{a}\cdot\nabla\boldsymbol{u}_h+\nabla p_h-\boldsymbol{\xi}_h,\boldsymbol{a}\cdot\nabla\boldsymbol{v}_h+\nabla q_h-\boldsymbol{\eta}_h)_{\tau_1} + (\nabla\cdot\boldsymbol{u}_h-\delta_h,\nabla\cdot\boldsymbol{v}_h-\gamma_h)_{\tau_2},$$

the discrete problem is equivalent to find $[\boldsymbol{u}_h,p_h,\boldsymbol{\xi}_h,\delta_h]\in\mathcal{V}_{h,0}\times\mathcal{Q}_h\times\mathcal{V}_h\times\mathcal{Q}_h$ such that $B_I^{\star}([\boldsymbol{u}_h,p_h,\boldsymbol{\xi}_h,\delta_h],[\boldsymbol{v}_h,q_h,\boldsymbol{\eta}_h,\gamma_h]) = \langle\boldsymbol{f},\boldsymbol{v}_h\rangle$ for all $[\boldsymbol{v}_h,q_h,\boldsymbol{\eta}_h,\gamma_h]\in\mathcal{V}_{h,0}\times\mathcal{Q}_h\times\mathcal{V}_h\times\mathcal{Q}_h$. This problem is consistent in the sense that, for smooth enough solutions $[\boldsymbol{u},p]$ of the continuous problem, $B_I^{\star}([\boldsymbol{u},p,\boldsymbol{a}\cdot\nabla\boldsymbol{u}+\nabla p,\nabla\cdot\boldsymbol{u}],[\boldsymbol{v}_h,q_h,\boldsymbol{\eta}_h,\gamma_h]) = \langle\boldsymbol{f},\boldsymbol{v}_h\rangle$.

2.3 Transient Oseen equations

2.3.1 Discretization in time

Let us consider now the transient Oseen problem, that is,

$$\partial_t\boldsymbol{u}-\nu\Delta\boldsymbol{u}+\boldsymbol{a}\cdot\nabla\boldsymbol{u}+\nabla p=\boldsymbol{f}\quad\text{in }\Omega,\ t\in(0,T), \tag{2.33}$$

$$\nabla\cdot\boldsymbol{u}=0\quad\text{in }\Omega,\ t\in(0,T), \tag{2.34}$$

supplied with an initial condition and the homogeneous Dirichlet condition for the velocity. In order to be able to use the same notation as in the previous section, we need to introduce the matrix

$$\boldsymbol{M} = \mathrm{diag}(\boldsymbol{I}_d, 0),$$

which allows us to write Eqs. (2.33)-(2.34) as

$$\boldsymbol{M}\partial_t \boldsymbol{U} + \mathcal{L}(\boldsymbol{U}) = \boldsymbol{F}, \tag{2.35}$$

where the notation involved is the same as before. For the sake of simplicity, we will consider throughout that $\boldsymbol{F}$ is time-independent.

Problem (2.35) needs to be approximated both in space and in time. For the time discretization we will consider here the simple trapezoidal rule, although the ideas to be developed can be equally applied to any other finite difference time integration scheme. If time is also discretized using finite elements, the methodology would be the straightforward extension of what has been developed for the stationary problem. However, the goal is to analyze how does the time discretization affect the stabilization method when using finite differences.

Let us consider a uniform partition of the time interval of analysis $[0, T]$ with time step size δt. We will denote by a superscript the time step level at which the algorithmic solution is computed. If $\theta \in (0, 1]$ and $\boldsymbol{U}^n$ is known, the trapezoidal rule applied to the variational form of Eq. (2.35) consists of finding $\boldsymbol{U}^{n+1}$ as the solution of the problem

$$(\boldsymbol{M} D_t \boldsymbol{U}^n, \boldsymbol{V}) + B(\boldsymbol{U}^{n+\theta}, \boldsymbol{V}) = L(\boldsymbol{V}) \quad \forall\, \boldsymbol{V} \in \mathcal{W}_0. \tag{2.36}$$

2.3.2 Subgrid scale decomposition and modeling of the subscales

From the semidiscrete problem (2.36) we can now obtain the fully discrete formulation applying the same ideas as for the stationary case. We start by considering the same decompositions $\mathcal{W} = \mathcal{W}_h \oplus \tilde{\mathcal{W}}$ and $\mathcal{W}_0 = \mathcal{W}_{h,0} \oplus \tilde{\mathcal{W}}_0$, which allow us to split (2.36) into two equations, the first of which is

$$\begin{aligned}(\boldsymbol{M} D_t \boldsymbol{U}_h^{n+1}, \boldsymbol{V}_h) + (\boldsymbol{M} D_t \tilde{\boldsymbol{U}}^{n+1}, \boldsymbol{V}_h) + B(\boldsymbol{U}_h^{n+\theta}, \boldsymbol{V}_h) \\ + \sum_K \int_K \tilde{\boldsymbol{U}}^{n+\theta} \cdot \mathcal{L}^*(\boldsymbol{V}_h)\, \mathrm{d}\Omega = L(\boldsymbol{V}_h) \quad \forall\, \boldsymbol{V}_h \in \mathcal{W}_{h,0},\end{aligned} \tag{2.37}$$

which corresponds to (2.10) of the stationary problem. Observe that we have already neglected the contribution from the integrals over the element boundaries. Note also that there is a contribution from the transient evolution of the subscales. These are solution of

$$\begin{aligned}&\boldsymbol{M} D_t \tilde{\boldsymbol{U}}^{n+1} + \mathcal{L}(\tilde{\boldsymbol{U}}^{n+\theta}) = \boldsymbol{R}_t^{n+\theta} \quad \text{in } K \in \Theta_h, \\ &\boldsymbol{R}_t^{n+\theta} := \boldsymbol{F}^{n+\theta} - \left[\boldsymbol{M} D_t \boldsymbol{U}_h^{n+1} + \mathcal{L}(\boldsymbol{U}_h^{n+\theta})\right] + \boldsymbol{V}_{h,\mathrm{ort}}\end{aligned} \tag{2.38}$$

which is the counterpart of (2.13) for the stationary case. Again, $\boldsymbol{V}_{h,\mathrm{ort}}$ is any arbitrary element in $\tilde{\mathcal{W}}_0^{\perp}$. In what follows, it is understood that the subscales are computed within each element K of the finite element partition Θ_h.

We can equivalently write (2.38) as

$$\left(\boldsymbol{M}\frac{1}{\theta \delta t} + \mathcal{L}\right) \tilde{\boldsymbol{U}}^{n+\theta} = \boldsymbol{M}\frac{1}{\theta \delta t}\tilde{\boldsymbol{U}}^{n} + \boldsymbol{R}_t^{n+\theta}, \tag{2.39}$$

from where a closed-form expression for $\tilde{\boldsymbol{U}}$ has to be proposed. It is the same *modeling* step as for the stationary case. There, the operator $\mathcal{L}$ was replaced by matrix $\boldsymbol{\tau}^{-1}$, which was designed on the grounds that both the exact and the modeled subscales had approximately the same L^2-norm over each element. To be consistent with the approximations made in the previous section, let us introduce the matrix

$$\boldsymbol{\tau}_t := \left(\boldsymbol{M}\frac{1}{\theta\delta t} + \boldsymbol{\tau}^{-1}\right)^{-1} = \mathrm{diag}(\tau_{1,t}\boldsymbol{I}_d, \tau_2), \tag{2.40}$$
$$\tau_{1,t} := \left(\frac{1}{\theta\delta t} + \frac{1}{\tau_1}\right)^{-1}.$$

The modeling of (2.39) proposed is

$$\tilde{\boldsymbol{U}}^{n+\theta} = \boldsymbol{\tau}_t\boldsymbol{M}\frac{1}{\theta\delta t}\tilde{\boldsymbol{U}}^{n} + \boldsymbol{\tau}_t\boldsymbol{R}_t^{n+\theta}. \tag{2.41}$$

Once $\tilde{\boldsymbol{U}}^{n+\theta}$ is computed we can obtain $\tilde{\boldsymbol{U}}^{n+1}$.

At this point we can impose that the subscales be orthogonal to the finite element space. This determines the function $\boldsymbol{V}_{h,\mathrm{ort}}$ in $\tilde{\mathcal{W}}_0^{\perp}$. The counterparts of Eqs. (2.27) and (2.28) are

$$\boldsymbol{V}_{h,\mathrm{ort}} = -\Pi\left[\boldsymbol{F}^{n+\theta} - \left(\boldsymbol{M}D_t\boldsymbol{U}_h^{n+1} + \mathcal{L}(\boldsymbol{U}_h^{n+\theta})\right)\right], \tag{2.42}$$
$$\tilde{\boldsymbol{U}}^{n+\theta} = \boldsymbol{\tau}_t\boldsymbol{M}\frac{1}{\theta\delta t}\tilde{\boldsymbol{U}}^{n} + \boldsymbol{\tau}_t\Pi^{\perp}\left[\boldsymbol{F}^{n+\theta} - \left(\boldsymbol{M}D_t\boldsymbol{U}_h^{n+1} + \mathcal{L}(\boldsymbol{U}_h^{n+\theta})\right)\right]. \tag{2.43}$$

Expression (2.43) is what we were looking for.

2.3.3 Stabilized finite element problem

The previous development is general and applicable to any system of convection-diffusion-reaction equations. Let us specialize it to the transient Oseen problem using the same approximations as for the stationary case. First, observe that

$$\Pi^{\perp}(\boldsymbol{F}^{n+\theta}) = \boldsymbol{0}, \quad \text{same approximation as for the stationary case,} \tag{2.44}$$
$$\Pi^{\perp}(\boldsymbol{M}D_t\boldsymbol{U}_h^{n+1}) = \boldsymbol{0}, \quad \text{since } \boldsymbol{M}D_t\boldsymbol{U}_h^{n+1} \text{ is a finite element function,} \tag{2.45}$$
$$(\boldsymbol{M}D_t\tilde{\boldsymbol{U}}^{n}, \boldsymbol{V}_h) = 0, \quad \text{since } \boldsymbol{M}D_t\tilde{\boldsymbol{U}}^{n} \text{ is orthogonal to } \mathcal{W}_{h,0}. \tag{2.46}$$

Using (2.44) and (2.45) in (2.43), taking into account expression (2.40) for $\boldsymbol{\tau}_t$ and neglecting the orthogonal projection of second derivatives as we did for the stationary case, we obtain the expression for the velocity and pressure subscales

$$\tilde{\boldsymbol{u}}^{n+\theta} = \tau_{1,t}\frac{1}{\theta\delta t}\tilde{\boldsymbol{u}}^{n} - \tau_{1,t}\Pi^{\perp}(\boldsymbol{a}\cdot\nabla\boldsymbol{u}_h^{n+\theta} + \nabla p_h^{n+\theta}), \tag{2.47}$$
$$\tilde{p}^{n+\theta} = -\tau_2\Pi^{\perp}(\nabla\cdot\boldsymbol{u}_h^{n+\theta}), \tag{2.48}$$

which inserted into the equation for the finite element solution (2.37) and noting (2.46) yields

$$\begin{aligned}
&(D_t\boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h) + B(\boldsymbol{U}_h^{n+\theta}, \boldsymbol{V}_h) \\
&\quad + (\Pi^{\perp}(\boldsymbol{a}\cdot\nabla\boldsymbol{u}_h^{n+\theta} + \nabla p_h^{n+\theta}), \boldsymbol{a}\cdot\nabla\boldsymbol{v}_h + \nabla q_h)_{\tau_{1,t}} \\
&\quad + (\Pi^{\perp}(\nabla\cdot\boldsymbol{u}_h^{n+\theta}), \nabla\cdot\boldsymbol{v}_h)_{\tau_2} \\
&\quad = L(\boldsymbol{V}_h) + \frac{1}{\theta\delta t}\left(\tilde{\boldsymbol{u}}^{n}, \boldsymbol{a}\cdot\nabla\boldsymbol{v}_h + \nabla q_h\right)_{\tau_{1,t}}.
\end{aligned} \tag{2.49}$$

This is the transient version of problem defined by the bilinear form B_I defined in (2.31). The main differences of the stabilizing terms of this transient problem with respect to the stationary one are

- The stabilization parameter τ_1 is replaced by $\tau_{1,t}$ (see (2.40)).
- There is a RHS contribution that comes from the fact that subscales need to be tracked in time.

This fact gives an answer to the question of how does the finite difference time integration affect the stabilization. First, we see that the stability parameter $\tau_{1,t}$ is certainly affected by δt. Expressions similar to (2.40) have been proposed for example in [145]. However, if no RHS modification is introduced, the steady-state solution would depend on the magnitude of δt and the method would lack stability for $\delta t \to 0$. On the other hand, if the present approach is used, it is clear that the stabilization terms tend to those of the stationary problem as the steady solution is reached. This can be seen for example from (2.41): If $\tilde{\boldsymbol{U}}^{n+\theta} = \tilde{\boldsymbol{U}}^{n}$ it is easily checked that $\tilde{\boldsymbol{U}}^{n+\theta} = \boldsymbol{\tau}\boldsymbol{R}_t^{n+\theta}$. In fact, using this assumption is a possible alternative. We could assume that the subscales do not change in time, and thus that $\tilde{\boldsymbol{U}}^{n+\theta} = \tilde{\boldsymbol{U}}^{n}$ always. This would lead to the same stabilization terms as for the stationary problem. Since the basic assumption is that the temporal variation of the subscales is negligible, they are call *quasi-static subscales*. In [49] it is shown how the *tracking* can be done when this assumption is not used.

2.4 Extension to the Navier-Stokes problem

2.4.1 Temporal discretization and linearization

Before discretizing in space the incompressible Navier-Stokes equations (1.7)-(1.1b), let us consider the time discretization using the generalized trapezoidal rule, as for the Oseen equations in Section 2.3. Using the same notation as above, at each time step the problem to be solved is

$$D_t \boldsymbol{u}^{n+1} + \boldsymbol{u}^{n+\theta} \cdot \nabla \boldsymbol{u}^{n+\theta} - \nu \Delta \boldsymbol{u}^{n+\theta} + \nabla p^{n+1} = \boldsymbol{f}, \tag{2.50}$$

$$\nabla \cdot \boldsymbol{u}^{n+\theta} = 0. \tag{2.51}$$

Observe that the pressure computed in (2.50) has been considered evaluated at time level $n+1$.

This problem is nonlinear. Before going to the finite element discretization we can linearize it. Again, several options are possible, but now we will restrict ourselves to the simple fixed point (or Picard) algorithm, which leads to an Oseen problem within each iteration step. Denoting by $f^{n,i}$ the i-th iteration of the unknown f at time level n, the linearized form of problem (2.50)-(2.51) is

$$D_t \boldsymbol{u}^{n+1,i} + \boldsymbol{u}^{n+\theta,i-1} \cdot \nabla \boldsymbol{u}^{n+\theta,i} - \nu \Delta \boldsymbol{u}^{n+\theta,i} + \nabla p^{n+1,i} = \boldsymbol{f}, \tag{2.52}$$

$$\nabla \cdot \boldsymbol{u}^{n+\theta,i} = 0. \tag{2.53}$$

This is the linear system to which we apply the previous developments.

2.4.2 Final algorithm

Problem (2.52)-(2.53) is clearly an Oseen problem for the velocity $\boldsymbol{u}^{n+\theta,i}$, being the advection velocity given by $\boldsymbol{a} \equiv \boldsymbol{u}^{n+\theta,i-1}$. Hence, we can apply the formulation developed in the previous section in a straightforward manner. However, there is an important remark to be made. When the unknown velocity is split into its finite element component and the subscale, this decomposition also affects the advection velocity $\boldsymbol{a}$, that is to say, we will have

$$\boldsymbol{a} \equiv \boldsymbol{u}_h^{n+\theta,i-1} + \tilde{\boldsymbol{u}}^{n+\theta,i-1}.$$

This implies that the velocity subscale not only needs to be tracked in time, but also along the iterative process. However, along this work some simplifications are assumed:

- We do not consider the tracking of the subscales in time (*quasi-static subscales*).
- We replace the advective velocity $\boldsymbol{u}_h + \tilde{\boldsymbol{u}}_h$ by $\boldsymbol{u}_h$. That is, we neglect the effect of the subscales in the transport of momentum.
- We do not consider the subgrid pressure, that is, we take $\tau_2 = 0$.

The final *simplified* algorithm is written in Box 2.1.

Remark 2.3. In Box 2.1 the projections onto the finite element space have been treated iteratively using the same iterative loop as for the fixed point linearization of the convective term. It is also possible to treat these projections explicitly, or even with a nested iteration within each linearization step. In any case, it has been observed in [49] from numerical experiments that it is important to treat the advective velocity implicitly.

Remark 2.4. The influence of the subscale in the advection velocity is a point that deserves further research. This influence is the key of turbulence modeling. How does this *numerical method* behaves as *physical model* is a point that deserves further research.

Box 2.1. Algorithm for solving the Navier-Stokes equations

- Read (or compute) $\boldsymbol{u}_h^0$ and set $p_h^0 = 0$.

FOR $n = 0, ..., N-1$ DO:

- Set $i = 0$
- Set $\boldsymbol{u}_h^{n+\theta,0} = \boldsymbol{u}_h^n,\ \ p_h^{n+1,0} = p_h^n$.

WHILE (not converged) DO:

- $i \leftarrow i + 1$
- Set $\boldsymbol{a} = \boldsymbol{u}_h^{n+\theta,i-1}$
- Compute τ from (2.18a)
- Compute the projections

$$\boldsymbol{\xi}_h = \Pi(\boldsymbol{a} \cdot \nabla \boldsymbol{u}_h^{n+\theta,i-1} + \nabla p_h^{n+1,i-1})$$

- Compute $\boldsymbol{u}_h^{n+\theta,i}$ and $p_h^{n+1,i}$ by solving

$$\begin{aligned}
&(D_t \boldsymbol{u}_h^{n+1,i}, \boldsymbol{v}_h) + (\boldsymbol{a} \cdot \nabla \boldsymbol{u}_h^{n+\theta,i}, \boldsymbol{v}_h) + \nu(\nabla \boldsymbol{u}_h^{n+\theta,i}, \nabla \boldsymbol{v}_h) \\
&\quad - (p_h^{n+1,i}, \nabla \cdot \boldsymbol{v}_h) + (q_h, \nabla \cdot \boldsymbol{u}_h^{n+\theta,i}) \\
&\quad + (\boldsymbol{a} \cdot \nabla \boldsymbol{u}_h^{n+\theta,i} + \nabla p_h^{n+1,i}, \boldsymbol{a} \cdot \nabla \boldsymbol{v}_h + \nabla q_h)_{\tau_1} \\
&\quad = \langle \boldsymbol{f}^{n+\theta}, \boldsymbol{v}_h \rangle + (\boldsymbol{\xi}_h, \boldsymbol{a} \cdot \nabla \boldsymbol{v}_h + \nabla q_h)_{\tau_1}
\end{aligned}$$

- Check convergence

END

- Set up converged values

$$\boldsymbol{u}_h^{n+\theta} = \boldsymbol{u}_h^{n+\theta,i},\ \ p_h^{n+1} = p_h^{n+1,i}$$

END

Chapter 3

Pressure Correction methods

This chapter is devoted to the popular pressure correction methods. We start this work with a wide overview of different methods suggested in the literature. Then we propose some pressure correction methods obtained at the fully discrete level. Several of them are analyzed, obtaining stability bounds. Finally, numerical experimentation exhibits the behavior of these methods. This work has been already published in [52, 50, 51].

3.1 Overview

The basic idea of the pressure segregation methods is to split the problem in two or more steps. They were initially developed as schemes for the time discretization. All of them have a common feature, the ability for decoupling the pressure and velocity calculation. They have become widely popular because of their low computational cost. Let us make an overview of these methods. For simplicity, we shall assume homogeneous Dirichlet conditions (1.10) in the following exposition.

3.1.1 Classical projection methods

The pressure segregation methods were originated independently by Chorin (see [40, 39, 42, 41]) and Temam (see [156, 155, 157, 158]). These methods are based on the orthogonal decomposition $\boldsymbol{L}^2(\Omega) = \boldsymbol{J}_0 \oplus \boldsymbol{J}_0^{\perp}$ (see (1.18) and (1.19)) which derives from a theorem due to Ladyzenskaya (see [120]). This theorem is based on the classical Helmholtz decomposition of a vector field into the sum of a solenoidal field and a gradient of a scalar function and a more general theorem proved by De Rham (see [59]).

Hence, in the projection method an intermediate velocity obtained from the momentum equation without the pressure term is decomposed into a solenoidal field, the velocity, and the gradient of a scalar field, the pressure. We present this method in its semidiscrete version in time.

This procedure involves two basic steps. The first step consists of finding an intermediate velocity $\hat{\boldsymbol{u}}^{n+1}$ such that:

$$\frac{\hat{\boldsymbol{u}}^{n+1} - \boldsymbol{u}^n}{\delta t} - \nu \Delta \hat{\boldsymbol{u}}^{n+1} + \hat{\boldsymbol{u}}^{n+1} \cdot \nabla \hat{\boldsymbol{u}}^{n+1} = \boldsymbol{f}^{n+1}, \tag{3.1a}$$

$$\hat{\boldsymbol{u}}^{n+1}|_{\Gamma} = 0. \tag{3.1b}$$

In (3.1a) the convective term is treated implicitly. Alternatively, semi-implicit and explicit approaches can be used. Note that the full Dirichlet boundary conditions are imposed on

$\hat{\boldsymbol{u}}^{n+1}$. The weak form of this equation is required for the numerical approximation. Its weak formulation consists of: find a $\hat{\boldsymbol{u}}^{n+1} \in \mathcal{V}_0$ such that

$$(\frac{\hat{\boldsymbol{u}}^{n+1} - \boldsymbol{u}^n}{\delta t}, \boldsymbol{v}) + a(\hat{\boldsymbol{u}}^{n+1}, \boldsymbol{v}) + c(\hat{\boldsymbol{u}}^{n+1}, \hat{\boldsymbol{u}}^{n+1}, \boldsymbol{v}) = \langle \boldsymbol{f}^{n+1}, \boldsymbol{v} \rangle, \qquad \forall \boldsymbol{v} \in \mathcal{V}_0. \tag{3.2}$$

Temam in [159] proposed the use of the skew-symmetric form of the convective term (see (3.2)). This form is usually assumed in order to obtain stability bounds for the solution of implicit projection methods.

The second step of this method consists of finding an end-of-step velocity $\boldsymbol{u}^{n+1} \in \boldsymbol{J}_0$ and a pressure $p^{n+1} \in \mathcal{Q}$ such that

$$\frac{\boldsymbol{u}^{n+1} - \hat{\boldsymbol{u}}^{n+1}}{\delta t} + \nabla p^{n+1} = 0, \tag{3.3a}$$

$$\nabla \cdot \boldsymbol{u}^{n+1} = 0, \tag{3.3b}$$

$$\boldsymbol{n} \cdot \boldsymbol{u}^{n+1}|_\Gamma = 0, \tag{3.3c}$$

which is equivalent to find the projection of $\hat{\boldsymbol{u}}^{n+1}$ onto the space $\boldsymbol{J}_0$ (see (1.16)),

$$\boldsymbol{u}^{n+1} = P_{\boldsymbol{J}_0}(\hat{\boldsymbol{u}}^{n+1}). \tag{3.4}$$

This is the reason why these methods are popularly known as *projection methods*. We point out that at this step only the normal component of the velocity is prescribed, not satisfying the end-of-step velocity $\boldsymbol{u}^{n+1}$ the proper boundary conditions. Hence, we infer easily from (3.1b) and (3.3c) that the pressure is satisfying on the Dirichlet boundary an *artificial Neumann condition*,

$$\boldsymbol{n} \cdot \nabla p^{n+1}|_\Gamma = 0. \tag{3.5}$$

This is one of the most controversial points of the classical projection method. There has been much speculation about whether the pressure p^{n+1} is a good approximation to the exact pressure $p(t^{n+1})$ (see [160]). It is conjectured by Rannacher in [140] and by Gresho in [83] that the non-physical boundary condition lives only in a narrow boundary layer of width $\mathcal{O}(\sqrt{\nu \delta t})$.

In 1996 Timmermans, Minev and Van de Vosse proposed in [161] a modified version of the projection method that leads to improved pressure approximations. Once solved the classical projection method, the modification consists of a correction of the obtained pressure p^{n+1} as follows:

$$\hat{p}^{n+1} = p^{n+1} - \nu \nabla \cdot \hat{\boldsymbol{u}}^{n+1}. \tag{3.6}$$

It is evident that this update will not cause a severe additional numerical effort. Furthermore, it has been observed in [95] that with this method, unlike (3.5), the following consistent pressure boundary condition is imposed,

$$\boldsymbol{n} \cdot \nabla p^{n+1}|_\Gamma = (\boldsymbol{f}(t^{n+1}) - \nabla \times \nabla \times \boldsymbol{u}^{n+1}). \tag{3.7}$$

Due to the fact that the operator $\nabla \times \nabla \times$ plays a key role, the pressure correction methods using this improvement are called *rotational* pressure correction methods, as proposed in [95]. A similar *enhanced* boundary condition for the pressure was previously proposed in [132] and [116].

The main interest of the projection method is the possible uncoupling of the pressure from the velocity in its numerical approximation. It is achieved by taking the divergence of (3.3a), obtaining a Pressure Poisson Equation (PPE from here onwards) for the pressure,

$$\Delta p^{n+1} = \frac{1}{\delta t}\nabla \cdot \hat{\boldsymbol{u}}^{n+1}, \tag{3.8a}$$

$$\boldsymbol{n} \cdot \nabla p^{n+1}|_{\Gamma} = 0. \tag{3.8b}$$

Once the pressure p^{n+1} is calculated, the end-of-step velocity is recovered from (3.3a).

Another interesting issue of this method is its inherent stability. It can be seen by numerical experimentation that equal order velocity and pressure space approximations lead to stabilized systems, so that the discrete inf-sup condition (1.45) can be violated. However, the stability is decreasing with the time step size. This effect has been hidden by the convergence analysis, where it is required a time step size *small enough*. A complete study of the machinery that makes the fully discretized projection method stable was developed by Codina in [47] and extended in [52].

The improvement of the error estimates for the pressure taking the *rotational form* has been studied in [136] and [93].

The long term behavior, dissipativity and unconditional stability of the projection scheme was analyzed in [151] and [2] by Simó and Armero. They also showed that this method exhibits an absorving set, as does the continuum flow generated by the Navier-Stokes equations.

A modified scheme was proposed by Shen in [146]. It consists of introducing in the momentum equation an approximation of the pressure, more specifically, the term ∇p^n. This method is usually called *incremental projection method*. The original scheme proposed by Shen was composed by a first step where the intermediate velocity $\hat{\boldsymbol{u}}^{n+1}$ is obtained,

$$\frac{1}{\delta t}(\hat{\boldsymbol{u}}^{n+1} - \hat{\boldsymbol{u}}^{n}) - \nu\Delta\hat{\boldsymbol{u}}^{n+1} + \boldsymbol{u}^{n} \cdot \nabla\hat{\boldsymbol{u}}^{n+1} + \nabla p^{n} = \boldsymbol{f}^{n+1}, \tag{3.9a}$$

$$\hat{\boldsymbol{u}}^{n+1}|_{\Gamma} = 0, \tag{3.9b}$$

and a second step for the end-of-step velocity $\boldsymbol{u}^{n+1}$ and pressure p^{n+1},

$$\frac{1}{\delta t}(\boldsymbol{u}^{n+1} - \hat{\boldsymbol{u}}^{n+1}) + \nabla(p^{n+1} - p^{n}) = \boldsymbol{0}, \tag{3.10a}$$

$$\nabla \cdot \boldsymbol{u}^{n+1} = 0, \tag{3.10b}$$

$$\boldsymbol{n} \cdot \boldsymbol{u}^{n+1}|_{\Gamma} = 0. \tag{3.10c}$$

Alternativelly, fully implicit or fully explicit convective terms can be used. As before, the pressure can be obtained from a Poisson equation. However, the variable obtained in this case is a *correction* of the pressure δp^{n+1} instead of the total pressure p^{n+1},

$$\Delta(p^{n+1} - p^{n}) = \frac{1}{\delta t}\nabla \cdot \hat{\boldsymbol{u}}^{n+1}, \tag{3.11a}$$

$$\boldsymbol{n} \cdot \nabla(p^{n+1} - p^{n})|_{\Gamma} = 0. \tag{3.11b}$$

This scheme is also called *pressure correction* method. This is the nomenclature we choose in this work. We note that the classical projection method can also be considered as a pressure correction method. The only difference is that in the classical method a zero order approximation of the pressure is used in the momentum equation whereas in (3.9a) a first order approximation is chosen.

So far, the approach adopted is first the discretization in time, then the splitting. We have a semidiscrete problem which can be approximated by its discretization in space. Hence, the controversy arises on which boundary conditions must be imposed in order to have well-posed problems at the continuous level.

A different approach to the projection method in a discrete setting (in space and time) was initially proposed by Perot in [133] in the framework of finite volumes. He suggested a pressure correction method for the solution of the discrete linearized system, obtained from the Navier-Stokes equations, based on an *incomplete block LU factorization*. In [137] this idea was generalized to different methods. The key point of this discrete approach is that the discussion about the correct boundary conditions can be skipped. This is the approach we will follow.

At the fully discrete level, the PPE introduces a perturbation term in the continuity equation which consists of a difference between two different Laplacian approximations. As it is showed in [47] this term is the origin of the inherent stability of the discretized form of the projection method. The characterization of the *stabilizer* motivated the Orthogonal Subscales methods suggested by Codina [49] and introduced in Chapter 2.

We refer to [88] for a recent overview of fractional step methods.

3.1.2 Higher order methods

Second order methods

The pressure correction methods showed above are first order schemes in the time step size. Several alternatives have been suggested to achieve second order methods. The first one was proposed by Kim and Moin in [118]. Therein they introduce a fractional step method where the convective term is evaluated by an explicit second order Adams-Bashfort method and the viscous term by a second order implicit Crank-Nicholson method. The main novelty of this scheme are the boundary condition imposed on the intermediate velocity.

Van Kan proposed in [162] a second order accurate pressure correction method. This method is equivalent to Shen's method but replacing the Backward Euler time integration scheme by the second order Crank-Nicholson. It is proved in [162] that the velocity and pressure solution of this method differ from the solution of the monolithic problem by $\mathcal{O}(\delta t^2)$.

In 1989 Bell Colella and Glaz introduced the first predictor corrector method. They proposed in [12] a second order iterative scheme which converges to the Crank-Nicholson monolithic system. For every iteration, first the intermediate velocity is obtained, treating the convection term explicitly and the diffusion term implicitly. Then, the end-of-step velocity for the current iteration and pressure are calculated coupled. The convective term is differentiated using a second order Godunov procedure.

Identically, second order accuracy in time can be reached using a BDF2 time integration method instead of Backward Euler in Shen's scheme (3.9)-(3.10). This second order pressure correction method is the most used in the present work.

In different papers second order pressure correction schemes have been treated analitically. E and Liu studied in [66] Kim and Moin's scheme. Shen analized the Van Kan method in [146] and [150], obtaining optimal error estimates assuming a sufficiently accurate initial data for the velocity field and pressure field and some regularity assumptions. In [136] optimal error estimates are recovered with weaker regularity requirements.

Third Order Methods

The first author who proposed a third order pressure correction method was Gresho in [83]. However, numerical experimentation shows that this method is unstable in time. In [148] Shen tried to explain why the solutions obtained with third order pressure correction methods can not be bounded uniformly in time. In fact, he obtained that *all the pressure correction methods of order higher than two are unstable.*

But according to the author's analysis, the explanation of Shen is not appropriate because some terms are neglected. In fact Shen pointed out that the *equivalent* continuous system he analyzes is obtained *by intuition.*

The reason why higher order pressure correction methods are unstable remains an open question that has not been fully explained yet.

3.2 Some pressure correction methods

The objective of this chapter is to study several aspects of some pressure correction methods for the *transient* incompressible Navier-Stokes equations using a finite element approximation for the space discretization. Our reference will be the solution of the *monolithic* problem, that is, the coupled calculation of the velocity and the pressure. Obviously, the fully discrete and linearized monolithic scheme leads to an algebraic system the structure of which can be exploited so as to solve independently for the velocity and pressure degrees of freedom. However, we consider this a particular algebraic treatment of the final linear system that we will not study in this work (even though it might have a counterpart at the space-continuous level).

It is *not* our purpose to discuss how to deal with the pressure interpolation. All our discussion will be based on a Galerkin finite element interpolation for the velocity and the pressure. If the velocity-pressure pairs satisfy the classical inf-sup condition, that will yield a stable pressure approximation. Otherwise, if for example equal velocity-pressure interpolation is used, the Galerkin formulation can be modified to a stabilized finite element method for which all the discussion that follows is easily adapted. Likewise, no stabilization methods for the convective term when it dominates the viscous one will be taken into account, although these could also be easily incorporated to what follows.

Referring to the time integration, we will concentrate on first and second order implicit finite difference schemes.

After describing the problem and its time and space discretization, the methods we wish to consider are presented. The first family are the classical fractional step methods. Our approach here is to present the splitting at the pure algebraic level, as in [133, 137], rather than at the space continuous level which is still the most common approach. The algebraic viewpoint obviates the discussion on the pressure boundary conditions (but nevertheless it is obviously there).

The second family of methods is the one based on a particular pressure Poisson equation presented in [93]. The momentum equation can be solved treating the pressure explicitly, and then updating the pressure. Contrary to classical fractional step methods, in this case there is no intermediate velocity to deal with. However, the velocity obtained with this scheme is not (weakly) divergence free for the discrete problem. Our contribution here is to compare methods based on this approach with the pressure correction methods discussed previously. We explicitly show the equivalence between this *momentum-pressure Poisson equation* approach and classical splitting methods.

The third and last family of methods is of predictor corrector type in the spirit of [16]

(see also references therein). Starting from pressure correction methods, we propose an iterative scheme the goal of which is to converge to the solution of the monolithic problem. The robustness of this scheme relies on the presence of a term that is motivated precisely by the starting splitting method.

3.2.1 Monolithic time discretization

The backward Euler method will be used for first order pressure correction methods, whereas for second order methods we will consider both the Crank-Nicolson and backward differencing (or Gear) schemes. These methods and convenient notation have been introduced in Chapter 1.

The time integration of the discretizated (in space) Navier–Stokes equations (1.43) using the generalized trapezoidal rule leads to problem (1.48). In this section the values of θ that we will take are $\theta = 1/2$, corresponding to the second order Crank-Nicolson scheme, and $\theta = 1$, which corresponds to the backward Euler method (or BDF1).

The BDF2 time integration scheme will be also adopted for second order pressure correction schemes. In this case we obtain the fully discrete system (1.51).

3.2.2 Space discretization

Let us recall the notation introduced in Section 1.3. Let Θ_h denote a finite element partition of the domain Ω of diameter h, from which we construct the finite element spaces $\mathcal{Q}_h$ and $\mathcal{V}_{h,0}$, approximations to $\mathcal{Q}$ and $\mathcal{V}_0$, respectively. The former is made up with continuous functions of degree k_{q} and the other with continuous vector functions of degree k_{v} verifying the homogeneous Dirichlet boundary conditions. In the following, finite element functions will be identified with a subscript h.

The discrete problem is obtained by approximating $\boldsymbol{u}$ and p. We assume that $\boldsymbol{u}_h^n$ and p_h^n are constructed using the standard finite element interpolation from the nodal values. From problem (1.48), these are solution of the nonlinear algebraic system

$$\mathsf{M}\frac{1}{\delta t}D_1\mathsf{U}^{n+1} + \mathsf{K}(\mathsf{U}^{n+\theta})\mathsf{U}^{n+\theta} + \mathsf{G}\mathsf{P}^{n+\theta} = \mathsf{F}^{n+\theta}, \tag{3.12a}$$

$$\mathsf{D}\mathsf{U}^{n+\theta} = \mathsf{0}, \tag{3.12b}$$

where U and P are the arrays of nodal unknowns for $\boldsymbol{u}$ and p, respectively. If we denote the node indexes with superscripts a, b, the space indexes with subscripts i, j, and the standard shape function of node a by N^a, the components of the arrays involved in these equations are:

$$\mathsf{M}_{ij}^{ab} = (N^a, N^b)\delta_{ij} \quad (\delta_{ij} \text{ is the Kronecker } \delta),$$
$$\mathsf{K}(\mathsf{U}^{n+\theta})_{ij}^{ab} = (N^a, \boldsymbol{u}_h^{n+\theta}\cdot\nabla N^b)\delta_{ij} + \frac{1}{2}\left(N^a, (\nabla\cdot\boldsymbol{u}_h^{n+\theta})N^b\right)\delta_{ij} + \nu(\nabla N^a, \nabla N^b)\delta_{ij},$$
$$\mathsf{G}_i^{ab} = (N^a, \partial_i N^b),$$
$$\mathsf{D}_j^{ab} = (N^a, \partial_j N^b),$$
$$\mathsf{F}_i^a = \langle N^a, f_i\rangle.$$

It is understood that all the arrays are matrices (except F, which is a vector) whose components are obtained by grouping together the left indexes in the previous expressions (a and possibly i) and the right indexes (b and possibly j). Likewise, (3.12) need to be modified to account for the Dirichlet boundary conditions (matrix G can be replaced by

$-\mathsf{D}^{\mathrm{t}}$ when this is done). Observe also that we have used the skew-symmetric form of the convective term, which yields the convective contribution to matrix $\mathsf{K}(\mathsf{U}^{n+\theta})$ skew-symmetric.

For problem (1.51) the resulting algebraic system is analogous to (3.12), simply replacing $D_1\mathsf{U}$ by $D_2\mathsf{U}$ and evaluating the rest of the terms at $n+1$ instead of $n+\theta$.

3.2.3 Classical pressure correction methods

The pressure correction methods can be introduced at this point, applied to the fully discrete problem (3.12). This is exactly equivalent to

$$\mathsf{M}\frac{1}{\delta t}(\tilde{\mathsf{U}}^{n+1}-\mathsf{U}^{n})+\mathsf{K}(\mathsf{U}^{n+\theta})\mathsf{U}^{n+\theta}+\gamma\mathsf{G}\mathsf{P}^{n}=\mathsf{F}^{n+\theta}, \tag{3.13a}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1}-\tilde{\mathsf{U}}^{n+1})+\mathsf{G}(\mathsf{P}^{n+1}-\gamma\mathsf{P}^{n})=\mathsf{0}, \tag{3.13b}$$

$$\mathsf{D}\mathsf{U}^{n+1}=\mathsf{0}, \tag{3.13c}$$

where $\tilde{\mathsf{U}}^{n+1}$ is an auxiliary variable and γ is a numerical parameter, whose values of interest are 0 and 1. At this point we can make the essential approximation

$$\mathsf{K}(\mathsf{U}^{n+\theta})\mathsf{U}^{n+\theta}\approx\mathsf{K}(\tilde{\mathsf{U}}^{n+\theta})\tilde{\mathsf{U}}^{n+\theta}, \tag{3.14}$$

where $\tilde{\mathsf{U}}^{n+\theta}:=\theta\tilde{\mathsf{U}}^{n+1}+(1-\theta)\mathsf{U}^{n}$. Expressing U^{n+1} in terms of $\tilde{\mathsf{U}}^{n+1}$ using (3.13b) and inserting the result in (3.13c), the set of equations to be solved is

$$\mathsf{M}\frac{1}{\delta t}(\tilde{\mathsf{U}}^{n+1}-\mathsf{U}^{n})+\mathsf{K}(\tilde{\mathsf{U}}^{n+\theta})\tilde{\mathsf{U}}^{n+\theta}+\gamma\mathsf{G}\mathsf{P}^{n}=\mathsf{F}^{n+\theta}, \tag{3.15a}$$

$$\delta t\mathsf{D}\mathsf{M}^{-1}\mathsf{G}(\mathsf{P}^{n+1}-\gamma\mathsf{P}^{n})=\mathsf{D}\tilde{\mathsf{U}}^{n+1}, \tag{3.15b}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1}-\tilde{\mathsf{U}}^{n+1})+\mathsf{G}(\mathsf{P}^{n+1}-\gamma\mathsf{P}^{n})=\mathsf{0}, \tag{3.15c}$$

which have been ordered according to the sequence of solution, for $\tilde{\mathsf{U}}^{n+1}$, P^{n+1} and U^{n+1}. This uncoupling of variables has been made possible by virtue of (3.14).

Even though problem (3.15) can be implemented as such, it is very convenient to make a further approximation. Observe that $\mathsf{D}\mathsf{M}^{-1}\mathsf{G}$ represents an approximation to the Laplacian operator. In order to avoid dealing with this matrix (which is computationally feasible only if M is approximated by a diagonal matrix), we can approximate

$$\mathsf{D}\mathsf{M}^{-1}\mathsf{G}\approx\mathsf{L},\quad\text{with components }\mathsf{L}^{ab}=-(\nabla N^{a},\nabla N^{b}). \tag{3.16}$$

Matrix L is the standard approximation to the Laplacian operator. Clearly, this approximation is only possible when continuous pressure interpolations are employed. Likewise, it introduces implicitly the same wrong pressure boundary condition as when the splitting is performed at the continuous level (see [82] for a discussion on boundary conditions for the pressure Poisson equation). In [84], the use of approximation (3.16) is referred to as "approximate projection".

After using (3.14) and (3.16) the problem to be solved is:

$$\mathsf{M}\frac{1}{\delta t}(\tilde{\mathsf{U}}^{n+1}-\mathsf{U}^{n})+\mathsf{K}(\tilde{\mathsf{U}}^{n+\theta})\tilde{\mathsf{U}}^{n+\theta}+\gamma\mathsf{G}\mathsf{P}^{n}=\mathsf{F}^{n+\theta}, \tag{3.17a}$$

$$\delta t\mathsf{L}(\mathsf{P}^{n+1}-\gamma\mathsf{P}^{n})=\mathsf{D}\tilde{\mathsf{U}}^{n+1}, \tag{3.17b}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1}-\tilde{\mathsf{U}}^{n+1})+\mathsf{G}(\mathsf{P}^{n+1}-\gamma\mathsf{P}^{n})=\mathsf{0}. \tag{3.17c}$$

We will consider three possibilities depending on the choice of θ and γ. Formally, it is easy to see that the perturbation term introduced by approximation (3.14), is of order $\mathcal{O}(\delta t)$ when $\gamma = 0$ and of order $\mathcal{O}(\delta t^2)$ when $\gamma = 1$ (observe from (3.13b) that $\mathcal{O}(\|\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}\|) = \delta t\mathcal{O}(\|\mathsf{P}^{n+1} - \gamma\mathsf{P}^n\|)$ in any norm $\|\cdot\|$). Thus, we will refer to the case $\gamma = 0$ as the case with splitting error of order 1, called SE1 in the following, and the case $\gamma = 1$ as the case with splitting error of order 2, called SE2. The three possibilities mentioned are:

- $\theta = 1$, $\gamma = 0$. Method BDF1-SE1.
- $\theta = 1$, $\gamma = 1$. Method BDF1-SE2.
- $\theta = 1/2$, $\gamma = 1$. Method CN-SE2.

Method BDF1-SE2 will obviously be first order, and thus the second order splitting error unnecessary. However, this method has some interesting properties that will be discussed below.

So far, we have considered the trapezoidal rule for the time integration. If, instead, we use BDF2 with a second order splitting error, the final algebraic system will be

$$\mathsf{M}\frac{1}{2\delta t}(3\tilde{\mathsf{U}}^{n+1} - 4\mathsf{U}^n + \mathsf{U}^{n-1}) + \mathsf{K}(\tilde{\mathsf{U}}^{n+1})\tilde{\mathsf{U}}^{n+1} + \mathsf{G}\mathsf{P}^n = \mathsf{F}^{n+1}, \tag{3.18a}$$

$$\frac{2}{3}\delta t\mathsf{L}(\delta\mathsf{P}^{n+1}) = \mathsf{D}\tilde{\mathsf{U}}^{n+1}, \tag{3.18b}$$

$$\mathsf{M}\frac{1}{2\delta t}(3\mathsf{U}^{n+1} - 3\tilde{\mathsf{U}}^{n+1}) + \mathsf{G}(\delta\mathsf{P}^{n+1}) = \mathsf{0}. \tag{3.18c}$$

We will call this method BDF2-SE2.

Remark 3.1. The system matrix $\mathsf{D}\mathsf{M}^{-1}\mathsf{G}$ can be used without any approximation when an iterative solver that only involves matrix-vector products is used (for instance, GMRES). Moreover, the computational cost can be reduced integrating the mass matrix M^{-1} with a closed rule. In this situation, *no artificial boundary conditions are introduced*. Recently, the use of the rotational form of pressure correction methods (see [161, 95]) has improved slightly the artificial boundary conditions when using the usual discrete Laplacian L. However, from our point of view, the use of $\mathsf{D}\mathsf{M}^{-1}\mathsf{G}$ is the way to choose if we want to avoid non-physical boundary conditions over the pressure. This could be the case of fluid-structure interaction, where artificial boundary conditions over the pressure are being imposed on the interface when the pressure over the solid boundary is of main interest. It could also be justified when using domain decomposition methods due to the fact that Dirichlet transmission conditions have to be imposed in an *interest zone*.

Equivalent Stabilized Monolithic Formulation

At this point we will compare the pressure correction method with the monolithic system as it was done in [47]. Being $\widetilde{\mathsf{P}}_p^{n+1}$ an extrapolation of P^{n+1} of order p obtained from previous known values, we can write an *approximated* pressure correction method as follows:

$$\mathsf{M}\frac{1}{\delta t}(\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\tilde{\mathsf{U}}^{n+1})\tilde{\mathsf{U}}^{n+1} + \mathsf{G}\widetilde{\mathsf{P}}_p^{n+1} = \mathsf{F}^{n+1}, \tag{3.19}$$

$$\delta t\mathsf{L}(\mathsf{P}^{n+1} - \widetilde{\mathsf{P}}_p^{n+1}) = \mathsf{D}\tilde{\mathsf{U}}^{n+1}, \tag{3.20}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}) + \mathsf{G}(\mathsf{P}^{n+1} - \widetilde{\mathsf{P}}_p^{n+1}) = \mathsf{0}. \tag{3.21}$$

We assume BDF1 for the sake of simplicity. We could rewrite (3.19)-(3.21) as an equivalent stabilized monolithic formulation (see [47]) in order to identify the perturbation terms introduced by the splitting, getting

$$\mathsf{M}\frac{1}{\delta t}D_k\mathsf{U}^{n+1} + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{E}(\mathsf{U}^{n+1}) + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{3.22a}$$

$$\mathsf{D}\mathsf{U}^{n+\theta} - \delta t(\mathsf{D}\mathsf{M}^{-1}\mathsf{G} - \mathsf{L})(\mathsf{P}^{n+1} - \widetilde{\mathsf{P}}_p^{n+1}) = \mathsf{0}. \tag{3.22b}$$

The term $\mathsf{E}(\mathsf{U}^{n+1})$ is given by

$$\begin{aligned}\mathsf{E}(\mathsf{U}^{n+1}) &:= \mathsf{K}(\mathsf{S}^{n+1})\mathsf{U}^{n+1} + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{S}^{n+1},\\ \mathsf{S}^{n+1} &:= \delta t\mathsf{M}^{-1}\mathsf{G}(\mathsf{P}^{n+1} - \widetilde{\mathsf{P}}_p^{n+1}).\end{aligned} \tag{3.23}$$

Remark 3.2. The perturbation of the momentum equation $\mathsf{E}(\mathsf{U}^{n+1})$ is formally of order $\mathcal{O}(\delta t^{p+1})$ where p is the approximation of the $\widetilde{\mathsf{P}}_p^{n+1}$ extrapolation. The continuity equation is perturbed with the term $\delta t\mathsf{B}(\mathsf{P}^{n+1} - \widetilde{\mathsf{P}}_p^{n+1})$ which is formally of order $\mathcal{O}(\delta t^{p+1})$.

Remark 3.3. This method is unconditionally stable for $p \leq 1$. Higher order methods seem to be conditionally stable.

3.2.4 Momentum-pressure Poisson equation methods

In this section we discuss a family of methods recently proposed in [93] that allow us to segregate the pressure calculation. The idea is to start with a formulation of the continuous problem equivalent to (1.7)-(1.1b) obtained by replacing the continuity equation by a pressure Poisson equation. The system of equations to be solved is thus

$$\partial_t \boldsymbol{u} + \boldsymbol{u}\cdot\nabla\boldsymbol{u} - \nu\Delta\boldsymbol{u} + \nabla p = \boldsymbol{f}, \tag{3.24a}$$

$$\Delta p = \nabla\cdot(\boldsymbol{f} + \nu\Delta\boldsymbol{u} - \boldsymbol{u}\cdot\nabla\boldsymbol{u}). \tag{3.24b}$$

The pressure boundary condition is obtained by imposing that the normal component of the pressure gradient be equal to the normal component of the term within parenthesis in the right-hand-side of (3.24b). The viscous term in this equation could be deleted by assuming that the divergence and the Laplacian operator commute, but this would lead to a non-physical pressure boundary condition.

The key point is the way the pressure appearing in (3.24) is treated in the time discretization. In principle, to guarantee that the incompressibility condition holds, both pressures should be the same. However, another possibility is to *use an explicit treatment of the pressure* in (3.24a). This implies that the incompressibility constraint will be relaxed, but allows to uncouple the velocity and pressure calculation. Using a BDF time integration method of order k, $k = 1, 2$, the equations to be solved are

$$\frac{1}{\delta t}D_k\boldsymbol{u}^{n+1} + \boldsymbol{u}^{n+1}\cdot\nabla\boldsymbol{u}^{n+1} - \nu\Delta\boldsymbol{u}^{n+1} + \nabla p_k^{*,n+1} = \boldsymbol{f}^{n+1}, \tag{3.25a}$$

$$\Delta p^{n+1} = \nabla\cdot(\boldsymbol{f}^{n+1} - \boldsymbol{u}^{n+1}\cdot\nabla\boldsymbol{u}^{n+1} + \nu\Delta\boldsymbol{u}^{n+1}), \tag{3.25b}$$

where $p_k^{*,n+1}$ is an explicit approximation to p^{n+1} of order k. The right-hand-side of (3.25b) is cumbersome to evaluate numerically. Making use of (3.25a) in (3.25b) we can alternatively solve

$$\frac{1}{\delta t}D_k\boldsymbol{u}^{n+1} + \boldsymbol{u}^{n+1}\cdot\nabla\boldsymbol{u}^{n+1} - \nu\Delta\boldsymbol{u}^{n+1} + \nabla p_k^{*,n+1} = \boldsymbol{f}^{n+1}, \tag{3.26a}$$

$$\delta t\Delta(p^{n+1} - p_k^{*,n+1}) = \nabla\cdot(D_k\boldsymbol{u}^{n+1}). \tag{3.26b}$$

Let us compare now these *momentum-pressure Poisson equation* methods with the pressure correction methods of the previous subsection when the space discretization is carried out. For $k = 1$, problem (3.26) leads to

$$\mathsf{M}\frac{1}{\delta t}(D_1\mathsf{U}^{n+1}) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}_1^{*,n+1} = \mathsf{F}^{n+1}, \tag{3.27a}$$

$$\delta t\mathsf{L}(\mathsf{P}^{n+1} - \mathsf{P}_1^{*,n+1}) = \mathsf{D}(D_1\mathsf{U}^{n+1}), \tag{3.27b}$$

whereas the algebraic equations of the BDF1-SE1 of the previous subsection can be re-arranged to yield

$$\mathsf{M}\frac{1}{\delta t}(D_1\tilde{\mathsf{U}}^{n+1}) + \mathsf{K}(\tilde{\mathsf{U}}^{n+1})\tilde{\mathsf{U}}^{n+1} + \mathsf{G}\mathsf{P}^{n} = \mathsf{F}^{n+1}, \tag{3.28a}$$

$$\delta t\mathsf{L}(\mathsf{P}^{n+1} - \mathsf{P}^{n}) = \mathsf{D}(D_1\tilde{\mathsf{U}}^{n+1}). \tag{3.28b}$$

It is observed that *problems (3.27) and (3.28) are identical*, provided the intermediate velocity of the fractional step method is identified with the velocity to be computed at each time step and the following two conditions hold: an initial pressure (which is unnecessary in fractional step methods) is obtained from the equation $\delta t\mathsf{L}\mathsf{P}^0 = \mathsf{D}\mathsf{U}^0$ and the explicit first order approximation to the pressure is taken as $\mathsf{P}_1^{*,n+1} = \mathsf{P}^n$.

In the case $k = 2$, problem (3.26) leads to

$$\mathsf{M}\frac{1}{\delta t}(D_2\mathsf{U}^{n+1}) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}(\mathsf{P}_2^{*,n+1}) = \mathsf{F}^{n+1}, \tag{3.29a}$$

$$\delta t\mathsf{L}(\mathsf{P}^{n+1} - \mathsf{P}_2^{*,n+1}) = \mathsf{D}(D_2\mathsf{U}^{n+1}), \tag{3.29b}$$

whereas the algebraic equations of the BDF2-SE2 of the previous subsection, with an appropriate choice for the initial conditions, can be written as

$$\mathsf{M}\frac{1}{\delta t}(D_2\tilde{\mathsf{U}}^{n+1}) + \mathsf{K}(\tilde{\mathsf{U}}^{n+1})\tilde{\mathsf{U}}^{n+1} + \mathsf{G}(\widetilde{\mathsf{P}}_2^{n+1} - \frac{1}{3}\delta^2\mathsf{P}^n) = \mathsf{F}^{n+1}, \tag{3.30a}$$

$$\delta t\mathsf{L}(\mathsf{P}^{n+1} - (\widetilde{\mathsf{P}}_2^{n+1} - \frac{1}{3}\delta^2\mathsf{P}^n)) = \mathsf{D}(D_2\tilde{\mathsf{U}}^{n+1}). \tag{3.30b}$$

Problems (3.29) and (3.30) are again *identical*, also identifying the intermediate velocity of the fractional step method with the velocity to be computed at each time step and taking $\mathsf{P}_2^{*,n+1} = \widetilde{\mathsf{P}}_2^{n+1} - \frac{1}{3}\delta^2\mathsf{P}^n$ as explicit second order approximation to the pressure at t^{n+1}. Therefore, scheme BDF2-SE2 can be considered a particular case of (3.29).

3.2.5 Predictor corrector schemes

Starting from the fractional step method (3.17), a predictor corrector scheme is proposed in [57] whose goal is to converge to the monolithic time discretized problem. We will omit the details of the motivation. Denoting by a superscript i the ith iteration of the scheme, the resulting linearized system is

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1,i+1} - \mathsf{U}^{n}) + \mathsf{K}(\mathsf{U}^{n+\theta,i})\mathsf{U}^{n+\theta,i+1} + \mathsf{G}\mathsf{P}^{n+\theta,i} = \mathsf{F}^{n+\theta}, \tag{3.31a}$$

$$\delta t\mathsf{L}(\mathsf{P}^{n+\theta,i+1} - \mathsf{P}^{n+\theta,i}) = \mathsf{D}\mathsf{U}^{n+\theta,i+1}. \tag{3.31b}$$

Apparently, this is a straightforward iteration procedure for solving the original *monolithic* problem (3.12) freezing the pressure gradient in the momentum equation. However, there is a term whose presence would be hardly motivated by looking only at this system,

namely, the term $\delta t\mathsf{L}(\mathsf{P}^{n+\theta,i+1}-\mathsf{P}^{n+\theta,i})$. The motivation to introduce it comes from the inspection of what happens in the fractional step scheme.

If instead of starting from the generalized trapezoidal rule the second order BDF scheme is employed, the iterative scheme we propose is

$$\mathsf{M}\frac{1}{2\delta t}(3\mathsf{U}^{n+1,i+1}-4\mathsf{U}^{n}+\mathsf{U}^{n-1})+\mathsf{K}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i+1}+\mathsf{G}\mathsf{P}^{n+1,i}=\mathsf{F}^{n+1}, \tag{3.32a}$$

$$\frac{2}{3}\delta t\mathsf{L}(\mathsf{P}^{n+1,i+1}-\mathsf{P}^{n+1,i})=\mathsf{D}\mathsf{U}^{n+1,i+1} \tag{3.32b}$$

with

$$\mathsf{P}^{n+1,0}=2\mathsf{P}^{n}-\mathsf{P}^{n-1}, \tag{3.33a}$$

$$\mathsf{U}^{n+1,0}=2\mathsf{U}^{n}-\mathsf{U}^{n-1}. \tag{3.33b}$$

Both (3.31) and (3.32) are iterative schemes in which the pressure calculation is uncoupled from the velocity. This is why we have included them in this section about pressure segregation methods. Their numerical performance will be discussed below.

3.3 Stability of pressure correction methods

This section is devoted to the *stability* analysis of several pressure correction methods presented above. Four methods are considered. In the first two, a first order time integration is used, both with a first and a second order splitting error. For the third and fourth methods the splitting error is of second order, the same as the time integration error. In one case the time integration is performed using the Crank-Nicolson method, whereas the BDF2 scheme is used in the last case. Two of these four stability results were already presented in [47], whereas the other two are new and recently published in [52]. The important issue is that we do not rely on the pressure interpolation. It leads to poor stability estimates for the pressure. They could be improved either by making use of the inf-sup condition or by resorting to stabilized finite element methods. Nevertheless, we consider important to point out that *some* pressure stability is obtained even if none of these possibilities is used. We end the discussion obtaining improved stability results for the BDF2 scheme stabilized with orthogonal subscales.

Let us first introduce some additional notation. If X, Y are arrays, $\{\mathsf{X}^n\}_{n=0}^N$ is a sequence of arrays of $N+1$ terms and A a symmetric positive semi-definite matrix, we define

$$\begin{aligned}
(\mathsf{X},\mathsf{Y})_{\mathsf{A}} &:= \mathsf{X}\cdot\mathsf{A}\mathsf{Y},\\
\|\mathsf{X}\|_{\mathsf{A}} &:= (\mathsf{X}\cdot\mathsf{A}\mathsf{X})^{1/2},\\
\|\mathsf{Y}\|_{-\mathsf{A}} &:= \sup_{\mathsf{x}\neq 0}\frac{\mathsf{Y}\cdot\mathsf{X}}{\|\mathsf{X}\|_{\mathsf{A}}} \quad \text{(here } \mathsf{A} \text{ is assumed to be positive definite)},\\
\{\mathsf{X}^n\}\in\ell^\infty(\mathsf{A}) &\iff \|\mathsf{X}^n\|_{\mathsf{A}}\le C<\infty \quad \forall\, n=0,1,...,N,\\
\{\mathsf{X}^n\}\in\ell^p(\mathsf{A}) &\iff \sum_{n=0}^{N}\delta t\|\mathsf{X}^n\|_{\mathsf{A}}^p\le C<\infty, \quad 1\le p<\infty.
\end{aligned}$$

Here and in the following, C denotes a positive constant, not necessarily the same at different appearances. Moreover, we denote by $N=[T/\delta t]$.

A remark is needed when $\mathsf{A} = \mathsf{K}$. This matrix is not symmetric, but it has the contribution from the convective term, which is skew-symmetric, and the contribution from the viscous term, $\mathsf{K}_{\rm visc}$, which is symmetric and positive definite. We will simply write $\mathsf{U} \cdot \mathsf{K}(\mathsf{U})\mathsf{U} = \mathsf{U} \cdot \mathsf{K}_{\rm visc}\mathsf{U} \equiv \|\mathsf{U}\|^2_{\mathsf{K}}$.

We will make use also of $\mathsf{L}^+ := -\mathsf{L}$, which is the positive semi-definite matrix corresponding to the discretization of $-\Delta$.

These definitions will allow us to express our stability results in a compact manner.

3.3.1 Stability results for non-stabilized schemes

For obtaining stability bounds in this section the basic assumption in all the cases will be that

Assumption 3.1. *The force vector* $\{\mathsf{F}^n\}_{n=0}^N$ *accomplishes*

$$\sum_{n=0}^{N} \delta t \|\mathsf{F}^n\|^2_{-\mathsf{K}} \leq C < \infty, \tag{3.34}$$

for all $\delta t > 0$.

This is the matrix version of the classical condition required for the problem to be well posed. Apart from this, *no other regularity assumptions will be required*. Thus, the following estimates hold for the minimum velocity-pressure regularity.

The first stability result we present was proved already in [47]. For method BDF1-SE1, we have:

Stability of BDF1-SE1:

$$\{\mathsf{U}^n\} \in \ell^\infty(\mathsf{M}), \quad \{\tilde{\mathsf{U}}^n\} \in \ell^\infty(\mathsf{M}) \cap \ell^2(\mathsf{K}), \quad \{\sqrt{\delta t}\,\mathsf{P}^n\} \in \ell^2(\mathsf{L}^+)$$

The stability estimate for the pressure shows that the pressure gradient *multiplied by* δt is ℓ^2 bounded. When δt is of order $\mathcal{O}(h^2)$ this is optimal [76, 27, 140]. For the velocity, the stability estimates are optimal.

Method BDF1-SE1 is first order because of the order of both the time integration and the splitting error. However, if we consider method BDF1-SE2, with a second order splitting error, we obtain the same estimates for the velocity but much weaker estimates for the pressure. The result is:

Stability of BDF1-SE2:

$$\{\mathsf{U}^n\} \in \ell^\infty(\mathsf{M}), \quad \{\tilde{\mathsf{U}}^n\} \in \ell^\infty(\mathsf{M}) \cap \ell^2(\mathsf{K}), \quad \{\delta t \mathsf{P}^n\} \in \ell^\infty(\mathsf{L}^+)$$

The stability estimate for the pressure is now multiplied by δt instead of $\sqrt{\delta t}$ as in the previous case, which makes it weaker (even though the temporal norm is stronger). The way to improve it is by making use of the inf-sup condition, if it holds for the velocity-pressure interpolation employed, or by using stabilization techniques.

The stability bounds for the BDF1-SE2 method are summarized in the next theorem.

Theorem 3.1. *Under Assumption 3.1 the following stability estimates hold for the BDF1-SE2 method*

$$\max_{0\leq n\leq N} \{\|\mathsf{U}^n\|^2_{\mathsf{M}} + \|\tilde{\mathsf{U}}^n\|^2_{\mathsf{M}} + \|\delta t \mathsf{P}^n\|^2_{\mathsf{L}^+}\} + \sum_{n=1}^{N} \delta t \|\tilde{\mathsf{U}}^n\|^2_{\mathsf{K}} \leq C,$$

for all $\delta t > 0$.

Proof. Let us prove this result. First, let us write the scheme as

$$\mathsf{M}\frac{1}{\delta t}(\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\tilde{\mathsf{U}}^{n+1})\tilde{\mathsf{U}}^{n+1} + \mathsf{G}\mathsf{P}^n = \mathsf{F}^{n+1}, \tag{3.35}$$

$$\delta t \mathsf{L}(\delta \mathsf{P}^{n+1}) = \mathsf{D}\tilde{\mathsf{U}}^{n+1}, \tag{3.36}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}) + \mathsf{G}(\delta \mathsf{P}^{n+1}) = 0. \tag{3.37}$$

Taking the inner product of (3.35) with $2\delta t\tilde{\mathsf{U}}^{n+1}$ and using the identity

$$(2a, a-b) := a^2 - b^2 + (a-b)^2,$$

we get

$$\begin{aligned}\|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{M}} - \|\mathsf{U}^n\|^2_{\mathsf{M}} + \|\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n\|^2_{\mathsf{M}} + 2\delta t\|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{K}} + 2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{G}\mathsf{P}^n \\ = 2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{F}^{n+1} \le \delta t\|\mathsf{F}^{n+1}\|^2_{-\mathsf{K}} + \delta t\|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{K}}.\end{aligned} \tag{3.38}$$

Multiplying (3.37) by $2\delta t\mathsf{U}^{n+1}$ we obtain

$$\|\mathsf{U}^{n+1}\|^2_{\mathsf{M}} - \|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{M}} + \|\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{M}} + 2\delta t\mathsf{U}^{n+1}\cdot\mathsf{G}\delta\mathsf{P}^{n+1} = 0. \tag{3.39}$$

Adding up (3.38) and (3.39) it is found that

$$\begin{aligned}\|\mathsf{U}^{n+1}\|^2_{\mathsf{M}} - \|\mathsf{U}^n\|^2_{\mathsf{M}} + \|\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{M}} + \|\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n\|^2_{\mathsf{M}} + \delta t\|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{K}} \\ + 2\delta t\mathsf{U}^{n+1}\cdot\mathsf{G}\delta\mathsf{P}^{n+1} + 2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{G}\mathsf{P}^n \le \delta t\|\mathsf{F}^{n+1}\|^2_{-\mathsf{K}}.\end{aligned} \tag{3.40}$$

Since (3.37) implies

$$\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1} = -\delta t\mathsf{M}^{-1}\mathsf{G}(\delta\mathsf{P}^{n+1}),$$

and $\mathsf{G}^{\mathrm{t}} = -\mathsf{D}$ with the boundary conditions considered, we have that

$$\begin{aligned}\|\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{M}} &= \delta t^2\mathsf{M}^{-1}\mathsf{G}(\delta\mathsf{P}^{n+1})\cdot\mathsf{G}(\delta\mathsf{P}^{n+1}) \\ &= -\delta t^2(\delta\mathsf{P}^{n+1})\cdot\mathsf{D}\mathsf{M}^{-1}\mathsf{G}(\delta\mathsf{P}^{n+1}) = \delta t^2\|\delta\mathsf{P}^{n+1}\|^2_{\mathsf{L}^+_D},\end{aligned} \tag{3.41}$$

being $\mathsf{L}^+_D := -\mathsf{D}\mathsf{M}^{-1}\mathsf{G}$ positive semi-definite.

From (3.36) we now obtain

$$\begin{aligned}2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{G}\mathsf{P}^n &= -2\delta t\mathsf{P}^n\cdot\mathsf{D}\tilde{\mathsf{U}}^{n+1} = 2\delta t^2\mathsf{P}^n\cdot\mathsf{L}^+\delta\mathsf{P}^{n+1} \\ &= \delta t^2(\|\mathsf{P}^{n+1}\|^2_{\mathsf{L}^+} - \|\mathsf{P}^n\|^2_{\mathsf{L}^+} - \|\delta\mathsf{P}^{n+1}\|^2_{\mathsf{L}^+}).\end{aligned} \tag{3.42}$$

On the other hand, using again (3.37) and (3.36) we get

$$\begin{aligned}2\delta t\mathsf{U}^{n+1}\cdot\mathsf{G}(\delta\mathsf{P}^{n+1}) &= -2\delta t(\delta\mathsf{P}^{n+1})\cdot\mathsf{D}(\tilde{\mathsf{U}}^{n+1} - \delta t\mathsf{M}^{-1}\mathsf{G}\delta\mathsf{P}^{n+1}) \\ &= 2\delta t^2(\delta\mathsf{P}^{n+1})\cdot\mathsf{L}^+(\delta\mathsf{P}^{n+1}) + 2\delta t^2(\delta\mathsf{P}^{n+1})\cdot\mathsf{D}\mathsf{M}^{-1}\mathsf{G}(\delta\mathsf{P}^{n+1}) \\ &= \delta t^2\|\delta\mathsf{P}^{n+1}\|^2_{\mathsf{L}^+} + \delta t^2\|\delta\mathsf{P}^{n+1}\|^2_{\mathsf{B}} - \delta t^2\|\delta\mathsf{P}^{n+1}\|^2_{\mathsf{L}^+_D},\end{aligned} \tag{3.43}$$

being $\mathsf{B} := \mathsf{D}\mathsf{M}^{-1}\mathsf{G} - \mathsf{L} = \mathsf{L}^+ - \mathsf{L}^+_D$ positive semi-definite (see [47]).

Using (3.41), (3.42) and (3.43) in (3.40), we find that

$$\|\mathsf{U}^{n+1}\|^2_{\mathsf{M}} - \|\mathsf{U}^n\|^2_{\mathsf{M}} + \delta t\|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{K}} + \delta t^2\|\mathsf{P}^{n+1}\|^2_{\mathsf{L}^+} - \delta t^2\|\mathsf{P}^n\|^2_{\mathsf{L}^+} \le \delta t\|\mathsf{F}^{n+1}\|^2_{-\mathsf{K}},$$

and summing from $n = 1$ to $n = N$, an arbitrary time level, we obtain

$$\|\mathsf{U}^N\|^2_{\mathsf{M}} + \sum_{n=1}^{N} \delta t \|\tilde{\mathsf{U}}^n\|^2_{\mathsf{K}} + \delta t^2 \|\mathsf{P}^N\|^2_{\mathsf{L}^+} \leq C, \tag{3.44}$$

where C includes the norm of the force vector and the initial condition. This inequality (3.44) proves the desired stability estimate for method BDF1-SE2, except for the $\ell^\infty(\mathsf{M})$ estimate for $\{\tilde{\mathsf{U}}^n\}$, which is easily obtained from (3.37), the $\ell^\infty(\mathsf{M})$ estimate for $\{\mathsf{U}^n\}$ and noting that the right-hand-side of (3.43) is non-negative.□

For the CN-SE2 the stability estimate was already obtained in [47]. The result is the following:

Stability of CN-SE2:

$$\{\mathsf{U}^n\} \in \ell^\infty(\mathsf{M}), \quad \{\tilde{\mathsf{U}}^n\} \in \ell^\infty(\mathsf{M}), \quad \{\tilde{\mathsf{U}}^{n+1/2}\} \in \ell^2(\mathsf{K}),$$
$$\{\delta t \mathsf{P}^n\} \in \ell^\infty(\mathsf{L}^+), \quad \{\sqrt{\delta t}\, \delta \mathsf{P}^n\} \in \ell^2(\mathsf{L}^+)$$

The same remarks as those made concerning the stability of method BDF1-SE2 apply now. We therefore conclude that the pressure stability depends on how the splitting is done rather than on the time integration scheme. This is also confirmed by the stability estimate for method BDF2-SE2, which we present now:

Stability of BDF2-SE2:

$$\{\mathsf{U}^n\} \in \ell^\infty(\mathsf{M}), \quad \{\tilde{\mathsf{U}}^n\} \in \ell^2(\mathsf{K}),$$
$$\{\delta t \mathsf{P}^n\} \in \ell^\infty(\mathsf{L}^+), \quad \{\sqrt{\delta t}\, \delta \mathsf{P}^n\} \in \ell^2(\mathsf{L}^+)$$

We conclude this section by proving these results, grouped in the next theorem.

Theorem 3.2. *Under Assumption 3.1 the following stability estimates hold for the BDF2-SE2 method*

$$\max_{0\leq n\leq N} \{\|\mathsf{U}^N\|^2_{\mathsf{M}} + \|\delta t \mathsf{P}^N\|^2_{\mathsf{L}^+}\} + \sum_{n=1}^{N} \delta t \{\|\tilde{\mathsf{U}}^n\|^2_{\mathsf{K}} + \|\sqrt{\delta t}\delta \mathsf{P}^n\|^2_{\mathsf{L}^+}\} \leq C,$$

for all $\delta t > 0$.

Proof. Let us start by the method obtained *without using approximation (3.16)*. This method reads

$$\mathsf{M}\frac{1}{2\delta t}(3\tilde{\mathsf{U}}^{n+1} - 4\mathsf{U}^n + \mathsf{U}^{n-1}) + \mathsf{K}(\tilde{\mathsf{U}}^{n+1})\tilde{\mathsf{U}}^{n+1} + \mathsf{G}\mathsf{P}^n = \mathsf{F}^{n+1}, \tag{3.45}$$

$$\mathsf{D}\mathsf{U}^{n+1} = 0, \tag{3.46}$$

$$\mathsf{M}\frac{1}{2\delta t}(3\mathsf{U}^{n+1} - 3\tilde{\mathsf{U}}^{n+1}) + \mathsf{G}(\delta \mathsf{P}^{n+1}) = 0. \tag{3.47}$$

In this case, the velocity at the end of the step is divergence free (in the discrete weak sense). Alternatively, (3.46) could be replaced by

$$\frac{2}{3}\delta t \mathsf{D}\mathsf{M}^{-1}\mathsf{G}(\delta \mathsf{P}^{n+1}) = \mathsf{D}\tilde{\mathsf{U}}^{n+1}, \tag{3.48}$$

in contrast with the pressure Poisson equation

$$\frac{2}{3}\delta t \mathsf{L}(\delta \mathsf{P}^{n+1}) = \mathsf{D}\tilde{\mathsf{U}}^{n+1}, \tag{3.49}$$

that would be obtained making use of approximation (3.16).

To obtain the stability of problem (3.45)-(3.47), let us start by multiplying (3.45) by $4\delta t\tilde{\mathsf{U}}^{n+1}$, getting

$$\begin{aligned}(2\tilde{\mathsf{U}}^{n+1}, 3\tilde{\mathsf{U}}^{n+1} - 4\mathsf{U}^n + \mathsf{U}^{n-1})_{\mathsf{M}} + 4\delta t\|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{K}} + 4\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{G}\mathsf{P}^n \\ = 4\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{F}^{n+1} \le 2\delta t\|\mathsf{F}^{n+1}\|^2_{-\mathsf{K}} + 2\delta t\|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{K}}\end{aligned} \tag{3.50}$$

Expanding the first term of (3.50) we get

$$\begin{aligned}&(2\tilde{\mathsf{U}}^{n+1}, 3\tilde{\mathsf{U}}^{n+1} - 4\mathsf{U}^n + \mathsf{U}^{n-1})_{\mathsf{M}} \\ &= (2\mathsf{U}^{n+1} + 2\tilde{\mathsf{U}}^{n+1} - 2\mathsf{U}^{n+1}, 3\mathsf{U}^{n+1} - 4\mathsf{U}^n + \mathsf{U}^{n-1} + 3\tilde{\mathsf{U}}^{n+1} - 3\mathsf{U}^{n+1})_{\mathsf{M}} \\ &= (2\mathsf{U}^{n+1}, 3\mathsf{U}^{n+1} - 4\mathsf{U}^n + \mathsf{U}^{n-1})_{\mathsf{M}} + (2\tilde{\mathsf{U}}^{n+1} - 2\mathsf{U}^{n+1}, 3\mathsf{U}^{n+1} - 4\mathsf{U}^n + \mathsf{U}^{n-1})_{\mathsf{M}} \\ &+ (2\tilde{\mathsf{U}}^{n+1}, 3\tilde{\mathsf{U}}^{n+1} - 3\mathsf{U}^{n+1})_{\mathsf{M}}.\end{aligned} \tag{3.51}$$

Using the identity

$$(2a, 3a - 4b + c) := a^2 - b^2 + (2a - b)^2 - (2b - c)^2 + (a - 2b + c)^2,$$

we can manipulate the first term in the right-hand-side of (3.51) as follows:

$$\begin{aligned}&(2\mathsf{U}^{n+1}, 3\mathsf{U}^{n+1} - 4\mathsf{U}^n + \mathsf{U}^{n-1})_{\mathsf{M}} \\ &\quad =\|\mathsf{U}^{n+1}\|^2_{\mathsf{M}} - \|\mathsf{U}^n\|^2_{\mathsf{M}} + \|2\mathsf{U}^{n+1} - \mathsf{U}^n\|^2_{\mathsf{M}} - \|2\mathsf{U}^n - \mathsf{U}^{n-1}\|^2_{\mathsf{M}} + \|\delta^2\mathsf{U}^{n+1}\|^2_{\mathsf{M}}.\end{aligned} \tag{3.52}$$

From (3.47) it follows that

$$\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^{n+1} = \frac{2}{3}\delta t\mathsf{M}^{-1}\mathsf{G}(\delta\mathsf{P}^{n+1}),$$

which can be used to express the second term in the right-hand-side of (3.51) as

$$(2\tilde{\mathsf{U}}^{n+1} - 2\mathsf{U}^{n+1}, 3\mathsf{U}^{n+1} - 4\mathsf{U}^n + \mathsf{U}^{n-1})_{\mathsf{M}} = -\frac{4}{3}\delta t(\delta\mathsf{P}^{n+1})\cdot\mathsf{D}(3\mathsf{U}^{n+1} - 4\mathsf{U}^n + \mathsf{U}^{n-1}) = 0, \tag{3.53}$$

which is zero because of (3.46).

For the last term in (3.51) we have

$$3(2\tilde{\mathsf{U}}^{n+1}, \tilde{\mathsf{U}}^{n+1} - \mathsf{U}^{n+1})_{\mathsf{M}} = 3\|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{M}} - 3\|\mathsf{U}^{n+1}\|^2_{\mathsf{M}} + 3\|\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^{n+1}\|^2_{\mathsf{M}}. \tag{3.54}$$

Using (3.52), (3.53) and (3.54) in (3.51), and applying the result in (3.50) we obtain

$$\begin{aligned}&3\|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{M}} - 3\|\mathsf{U}^{n+1}\|^2_{\mathsf{M}} + 3\|\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^{n+1}\|^2_{\mathsf{M}} + \|\mathsf{U}^{n+1}\|^2_{\mathsf{M}} - \|\mathsf{U}^n\|^2_{\mathsf{M}} \\ &\quad + \|2\mathsf{U}^{n+1} - \mathsf{U}^n\|^2_{\mathsf{M}} - \|2\mathsf{U}^n - \mathsf{U}^{n-1}\|^2_{\mathsf{M}} + \|\delta^2\mathsf{U}^{n+1}\|^2_{\mathsf{M}} + 2\delta t\|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{K}} \\ &\quad + 4\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{G}\mathsf{P}^n \le 2\delta t\|\mathsf{F}^{n+1}\|^2_{-\mathsf{K}}.\end{aligned} \tag{3.55}$$

On the other hand, (3.47) can be reordered to get

$$\frac{3}{2\delta t}\mathsf{U}^{n+1} + \mathsf{M}^{-1}\mathsf{G}\mathsf{P}^{n+1} = \frac{3}{2\delta t}\tilde{\mathsf{U}}^{n+1} + \mathsf{M}^{-1}\mathsf{G}\mathsf{P}^n.$$

Squaring both terms of this equation with the inner product $(\cdot,\cdot)_{\mathsf{M}}$ we obtain

$$\begin{aligned}&(\frac{3}{2\delta t}\mathsf{U}^{n+1} + \mathsf{M}^{-1}\mathsf{G}\mathsf{P}^{n+1})\cdot(\mathsf{M}\frac{3}{2\delta t}\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1}) \\ &\qquad = (\frac{3}{2\delta t}\tilde{\mathsf{U}}^{n+1} + \mathsf{M}^{-1}\mathsf{G}\mathsf{P}^n)\cdot(\mathsf{M}\frac{3}{2\delta t}\tilde{\mathsf{U}}^{n+1} + \mathsf{G}\mathsf{P}^n).\end{aligned}$$

After expanding the terms of this equality and using the fact that the velocity at the end of step is divergence free, it is found

$$\frac{9}{4\delta t^2}\|\mathsf{U}^{n+1}\|^2_{\mathsf{M}} + \|\mathsf{P}^{n+1}\|^2_{\mathsf{L}_D^+} = \frac{9}{4\delta t^2}\|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{M}} + \frac{6}{2\delta t}\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{G}\mathsf{P}^n + \|\mathsf{P}^n\|^2_{\mathsf{L}_D^+}. \tag{3.56}$$

Recall that $\mathsf{L}_D^+ := -\mathsf{D}\mathsf{M}^{-1}\mathsf{G}$ is positive semi-definite. Multiplying this equation by $\frac{4}{3}\delta t^2$ and adding it to (3.55) we obtain

$$\begin{aligned}&\|\mathsf{U}^{n+1}\|^2_{\mathsf{M}} - \|\mathsf{U}^{n}\|^2_{\mathsf{M}} + 3\|\tilde{\mathsf{U}}^{n+1}-\mathsf{U}^{n+1}\|^2_{\mathsf{M}} + \|2\mathsf{U}^{n+1}-\mathsf{U}^{n}\|^2_{\mathsf{M}} - \|2\mathsf{U}^{n}-\mathsf{U}^{n-1}\|^2_{\mathsf{M}}\\ &\quad + \|\delta^2\mathsf{U}^{n+1}\|^2_{\mathsf{M}} + \frac{4}{3}\delta t^2\|\mathsf{P}^{n+1}\|^2_{\mathsf{L}_D^+} - \frac{4}{3}\delta t^2\|\mathsf{P}^{n}\|^2_{\mathsf{L}_D^+} + 2\delta t\|\tilde{\mathsf{U}}^{n+1}\|^2_{\mathsf{K}} + \leq 2\delta t\|\mathsf{F}^{n+1}\|^2_{-\mathsf{K}}.\end{aligned} \tag{3.57}$$

Using (3.47) we can express $\|\tilde{\mathsf{U}}^{n+1}-\mathsf{U}^{n+1}\|_{\mathsf{M}}$ as

$$\begin{aligned}3\|\tilde{\mathsf{U}}^{n+1}-\mathsf{U}^{n+1}\|^2_{\mathsf{M}} &= \frac{4}{3}\delta t^2\mathsf{M}^{-1}\mathsf{G}(\mathsf{P}^{n+1}-\mathsf{P}^n)\cdot\mathsf{M}\mathsf{M}^{-1}\mathsf{G}(\mathsf{P}^{n+1}-\mathsf{P}^n)\\ &= -\frac{4}{3}\delta t^2(\mathsf{P}^{n+1}-\mathsf{P}^n)\cdot\mathsf{D}\mathsf{M}^{-1}\mathsf{G}(\mathsf{P}^{n+1}-\mathsf{P}^n) = \frac{4}{3}\delta t\|\sqrt{\delta t}(\mathsf{P}^{n+1}-\mathsf{P}^n)\|^2_{\mathsf{L}_D^+}.\end{aligned}$$

Using this in (3.57), adding the result up from $n=1$ to an arbitrary time level N and neglecting some positive terms we get

$$\|\mathsf{U}^N\|^2_{\mathsf{M}} + \frac{4}{3}\sum_{n=1}^{N}\delta t\|\sqrt{\delta t}\delta\mathsf{P}^n\|^2_{\mathsf{L}_D^+} + \frac{4}{3}\|\delta t\mathsf{P}^N\|^2_{\mathsf{L}_D^+} + 2\sum_{n=1}^{N}\delta t\|\tilde{\mathsf{U}}^n\|^2_{\mathsf{K}} \leq C, \tag{3.58}$$

where C involves the norm of the force vector and the initial condition. Therefore, the stability results obtained are:

$$\{\mathsf{U}^n\}\in\ell^\infty(\mathsf{M}),\ \{\tilde{\mathsf{U}}^n\}\in\ell^2(\mathsf{K}),\ \{\sqrt{\delta t}\delta\mathsf{P}^n\}\in\ell^2(\mathsf{L}_D^+),\ \{\delta t\mathsf{P}^n\}\in\ell^\infty(\mathsf{L}_D^+).$$

Let us consider now scheme BDF2-SE2 using approximation $\mathsf{D}\mathsf{M}^{-1}\mathsf{G}\cong\mathsf{L}$. In this case, the velocity at the end of the step is not divergence free. Making use of (3.18c) in (3.18b) the scheme can be written as follows:

$$\mathsf{M}\frac{1}{2\delta t}(3\tilde{\mathsf{U}}^{n+1}-4\mathsf{U}^n+\mathsf{U}^{n-1}) + \mathsf{K}(\tilde{\mathsf{U}}^{n+1})\tilde{\mathsf{U}}^{n+1} + \mathsf{G}\mathsf{P}^n = \mathsf{F}^{n+1}, \tag{3.59}$$

$$\mathsf{D}\mathsf{U}^{n+1} + \frac{2}{3}\delta t\mathsf{B}(\delta\mathsf{P}^{n+1}) = 0, \tag{3.60}$$

$$\mathsf{M}\frac{1}{2\delta t}(3\mathsf{U}^{n+1}-3\tilde{\mathsf{U}}^{n+1}) + \mathsf{G}(\delta\mathsf{P}^{n+1}) = 0. \tag{3.61}$$

Recall that $\mathsf{B} := \mathsf{D}\mathsf{M}^{-1}\mathsf{G}-\mathsf{L}$ is positive semi-definite.

The places of the previous stability analysis where there will be differences are those where the divergence free property of the end-of-step velocity has been used. This assumption has just been made in equations (3.53) and (3.56). Using (3.60), expression (3.53) in this case is:

$$\begin{aligned}&-\frac{4}{3}\delta t(\delta\mathsf{P}^{n+1})\cdot\mathsf{D}(3\mathsf{U}^{n+1}-4\mathsf{U}^n+\mathsf{U}^{n-1}) = \frac{4}{9}\delta t^2(2\delta\mathsf{P}^{n+1})\cdot(3\mathsf{B}\delta\mathsf{P}^{n+1}-4\mathsf{B}\delta\mathsf{P}^n+\mathsf{B}\delta\mathsf{P}^{n-1})\\ &\quad = \frac{4}{9}\delta t^2\cdot(\|\delta\mathsf{P}^{n+1}\|^2_{\mathsf{B}} - \|\delta\mathsf{P}^n\|^2_{\mathsf{B}} + \|2\delta\mathsf{P}^{n+1}-\delta\mathsf{P}^n\|^2_{\mathsf{B}} - \|2\delta\mathsf{P}^n-\delta\mathsf{P}^{n-1}\|^2_{\mathsf{B}} + \|\delta^3\mathsf{P}^n\|_{\mathsf{B}}).\end{aligned} \tag{3.62}$$

On the other hand, the following term has to be added to the left-hand-side of (3.56):

$$\frac{6}{2\delta t^2}\mathsf{U}^{n+1}\cdot\mathsf{G}\mathsf{P}^{n+1} = -\frac{6}{2\delta t^2}\mathsf{P}^{n+1}\cdot\mathsf{D}\mathsf{U}^{n+1}$$
$$= 2\mathsf{P}^{n+1}\cdot\mathsf{B}\delta\mathsf{P}^{n+1} = \|\mathsf{P}^{n+1}\|_{\mathsf{B}}^2 - \|\mathsf{P}^{n}\|_{\mathsf{B}}^2 + \|\delta\mathsf{P}^{n+1}\|_{\mathsf{B}}^2. \tag{3.63}$$

With the change (3.62) in (3.53) and (3.63) in (3.56), inequality (3.57) has to be replaced by

$$\begin{aligned}
&\|\mathsf{U}^{n+1}\|_{\mathsf{M}}^2 - \|\mathsf{U}^{n}\|_{\mathsf{M}}^2 + \frac{4}{3}\delta t\|\sqrt{\delta t}\delta\mathsf{P}^{n+1}\|_{\mathsf{L}_D^+}^2 + \frac{4}{3}\|\delta t\mathsf{P}^{n+1}\|_{\mathsf{L}_D^+}^2 - \frac{4}{3}\|\delta t\mathsf{P}^{n}\|_{\mathsf{L}_D^+}^2\\
&\quad + \frac{4}{3}\|\delta t\mathsf{P}^{n+1}\|_{\mathsf{B}}^2 - \frac{4}{3}\|\delta t\mathsf{P}^{n}\|_{\mathsf{B}}^2 + \frac{4}{3}\delta t\|\sqrt{\delta t}\delta\mathsf{P}^{n+1}\|_{\mathsf{B}}^2\\
&\quad + 2\delta t\|\tilde{\mathsf{U}}^{n+1}\|_{\mathsf{K}}^2 + \|2\mathsf{U}^{n+1} - \mathsf{U}^{n}\|_{\mathsf{M}}^2 - \|2\mathsf{U}^{n} - \mathsf{U}^{n-1}\|_{\mathsf{M}}^2 + \|\delta^2\mathsf{U}^{n+1}\|_{\mathsf{M}}^2\\
&\quad + \frac{4}{9}\delta t^2\cdot(\|\delta\mathsf{P}^{n+1}\|_{\mathsf{B}}^2 - \|\delta\mathsf{P}^{n}\|_{\mathsf{B}}^2 + \|2\delta\mathsf{P}^{n+1} - \delta\mathsf{P}^{n}\|_{\mathsf{B}}^2 - \|2\delta\mathsf{P}^{n} - \delta\mathsf{P}^{n-1}\|_{\mathsf{B}}^2 + \|\delta^3\mathsf{P}^{n}\|_{\mathsf{B}})\\
&\leq 2\delta t\|\mathsf{F}^{n+1}\|_{-\mathsf{K}}^2.
\end{aligned}$$

Adding up from $n = 1$ to an arbitrary time level N and neglecting some positive terms we find that

$$\begin{aligned}
&\|\mathsf{U}^{N}\|_{\mathsf{M}}^2 + \frac{4}{3}\sum_{n=1}^{N}\delta t\|\sqrt{\delta t}\delta\mathsf{P}^{n}\|_{\mathsf{L}_D^+}^2 + \frac{4}{3}\|\delta t\mathsf{P}^{N}\|_{\mathsf{L}_D^+}^2 + 2\sum_{n=1}^{N}\delta t\|\tilde{\mathsf{U}}^{n}\|_{\mathsf{K}}^2\\
&\quad + \frac{4}{3}\sum_{n=1}^{N}\delta t\|\sqrt{\delta t}\delta\mathsf{P}^{n}\|_{\mathsf{B}}^2 + \frac{4}{3}\|\delta t\mathsf{P}^{N}\|_{\mathsf{B}}^2 \leq C.
\end{aligned}\tag{3.64}$$

which, noting that $\mathsf{B} + \mathsf{L}_D^+ = \mathsf{L}^+$, yields the stability result we wished to prove.□

Remark 3.4. In this section we have obtained bounds over $\|\sqrt{\delta t}\nabla p_h^n\|_0$ (for first order schemes) or over $\|\sqrt{\delta t}\nabla(\delta p_h^n)\|_0$ (for second order schemes). However, as it will be seen below, more stability over the finite element projection of ∇p_h^n can be obtained by exploiting the momentum equation. It consists (approximately) of a bound over $\|h\Pi_h(\nabla p_h^n)\|_0$, h being the mesh size.

3.3.2 Improved stability results for a stabilized scheme

In this section we study again the BDF2-SE2 scheme analyzed above, obtaining pressure stability without relying on a velocity-pressure pair satisfying the inf-sup condition. These results are now improved by the orthogonal subscales stabilization method. We point out that in the following analysis we only consider the pressure stabilization terms. Even though we do not consider the stabilization of the convective term, it can be easily introduced, as commented in Remark 3.5.

We consider the variational form of the BDF2-SE2 method adding the stabilization terms required. The problem consists of finding $\tilde{\boldsymbol{u}}_h^{n+1} \in \mathcal{V}_{h,0}$, $\boldsymbol{u}_h^{n+1} \in \mathcal{V}_{h,0}$, $p_h^{n+1} \in \mathcal{Q}_h$ and

$\boldsymbol{\pi}_h^{n+1} \in \mathcal{V}_h$ such that

$$\frac{1}{2\delta t}(3\tilde{\boldsymbol{u}}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}, \boldsymbol{v}_h) + (\tilde{\boldsymbol{u}}_h^{n+1} \cdot \nabla \tilde{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h) + \nu(\nabla \tilde{\boldsymbol{u}}_h^{n+1}, \nabla \boldsymbol{v}_h)$$
$$+ (\nabla p_h^n, \boldsymbol{v}_h) = \langle \bar{\boldsymbol{f}}^{n+1}, \boldsymbol{v}_h \rangle \qquad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}, \tag{3.65a}$$
$$(q_h, \nabla \cdot \tilde{\boldsymbol{u}}_h^{n+1}) = -\frac{2\delta t}{3}(\nabla(p_h^{n+1} - p_h^n), \nabla q_h)$$
$$- \tau(\nabla p_h^{n+1} - \boldsymbol{\pi}_h^{n+\beta}, \nabla q_h) \qquad \forall q_h \in \mathcal{Q}_h, \tag{3.65b}$$
$$\frac{1}{2\delta t}(3\boldsymbol{u}_h^{n+1} - 3\tilde{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h) + (\nabla(p_h^{n+1} - p_h^n), \boldsymbol{v}_h) = 0 \qquad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}, \tag{3.65c}$$
$$(\boldsymbol{\pi}_h^{n+1}, \eta_h) = (\nabla p_h^{n+1}, \eta_h) \qquad \forall \eta_h \in \mathcal{V}_h, \tag{3.65d}$$

where $\mathcal{V}_h$ is the space $\mathcal{V}_{h,0}$ enlarged with the test functions associated to the boundary nodes and β can be either $\beta = 0$ or $\beta = 1$. Here $\boldsymbol{\pi}_h^{n+1}$ is the projection of ∇p_h^{n+1} onto $\mathcal{V}_h$ as it could be easily observed from (3.65d). Furthermore, we will introduce other projections of ∇p_h^{n+1} with respect to the L^2-inner product

$\boldsymbol{\pi}_1$: projection onto $\mathcal{V}_{h,0}$
$\boldsymbol{\pi}_2$: projection onto $(\mathcal{V}_{h,0})^\perp \cap \mathcal{V}_h$
$\boldsymbol{\pi}_3$: projection onto $(\mathcal{V}_h)^\perp$

and we will denote $\boldsymbol{\pi}_{ij} := \boldsymbol{\pi}_i + \boldsymbol{\pi}_j$ and $\boldsymbol{\pi}_h = \boldsymbol{\pi}_{12}$. This notation is the same as that used in [47] for a similar analysis to the one presented here but using the Crank-Nicholson scheme for the time discretization. As in the above reference, we will need the following assumption:

Assumption 3.2. *There is a positive constant C independent of h such that*

$$\|\nabla p_h\| < c(\|\boldsymbol{\pi}_1\| + \|\boldsymbol{\pi}_3\|)$$

for any $p_h \in \mathcal{Q}_h$.

This condition is studied in [53], and holds for equal velocity-pressure interpolation. Once again we need regularity assumptions for the data.

Assumption 3.3. *The sequence $\{\bar{\boldsymbol{f}}^n\}_{n=0,1,\ldots,N}$ is such that*

$$\sum_{n=0}^{N} \delta t \|\bar{\boldsymbol{f}}^n\|_{-1}^2 \leq C < \infty,$$

for all $\delta t > 0$.

Furthermore, some assumptions over the stabilization parameter are needed.

Assumption 3.4. *There exist two positive constants C_1 and C_2, possibly depending on the viscosity ν, such that*

$$\tau = C_1 h^2$$

and

$$\tau \leq C_2 \delta t.$$

for all $\delta t > 0$.

Under this assumptions, the stability results to be proved are:

Stability of the stabilized BDF2-SE2:
$$\{\boldsymbol{u}_h^n\} \in \ell^\infty(L^2), \quad \{\tilde{\boldsymbol{u}}_h^n\} \in \ell^2(H^1),$$
$$\{\delta t \nabla p_h^n\} \in \ell^\infty(L^2), \quad \{\sqrt{\delta t}\,\nabla \delta p_h^n\} \in \ell^2(L^2), \quad \{\sqrt{\tau}\nabla p_h^n\} \in \ell^1(L^2)$$

The remaining of this section is devoted to prove this result. As before, C will denote a positive constant, possibly with different values at different appearances. It may also depend on the viscosity ν, since our intention *is not* to treat convection dominated flows.

Theorem 3.3. *Under Assumptions 3.2, 3.3 and 3.4 the following stability estimates hold for the method (3.65)*

$$\max_{0\le n\le N}\{\|\boldsymbol{u}_h^n\|_0^2 + \|\delta t\nabla p_h^n\|_0^2\} + \sum_{n=1}^{N} \delta t\{\|\tilde{\boldsymbol{u}}_h^n\|_1^2 + \|\sqrt{\delta t}\nabla\delta p_h^n\|_0^2 + \|\sqrt{\tau}\nabla p_h^n\|_0\} \le C,$$

for all $\delta t > 0$.

Proof. Taking $\boldsymbol{v}_h = 4\delta t\tilde{\boldsymbol{u}}_h^{n+1}$ we obtain

$$\begin{aligned}(3\tilde{\boldsymbol{u}}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}, 2\tilde{\boldsymbol{u}}_h^{n+1}) + 4\delta t\nu\|\tilde{\boldsymbol{u}}_h^{n+1}\|_1^2 + 4\delta t(\nabla p_h^n, \tilde{\boldsymbol{u}}_h^{n+1}) = \\ 4\delta t(\bar{\boldsymbol{f}}^{n+1}, \tilde{\boldsymbol{u}}_h^{n+1}) \le 2\delta tC\|\bar{\boldsymbol{f}}^{n+1}\|_{-1}^2 + 2\delta t\nu\|\tilde{\boldsymbol{u}}_h^{n+1}\|_1^2. \end{aligned} \tag{3.66}$$

Expanding the first term of (3.66) we get

$$\begin{aligned}&(3\tilde{\boldsymbol{u}}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}, 2\tilde{\boldsymbol{u}}_h^{n+1}) \\ &\quad= (3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1} + 3\tilde{\boldsymbol{u}}_h^{n+1} - 3\boldsymbol{u}_h^{n+1}, 2\boldsymbol{u}_h^{n+1} + 2\tilde{\boldsymbol{u}}_h^{n+1} - 2\boldsymbol{u}_h^{n+1}) \\ &\quad= (3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}, 2\boldsymbol{u}_h^{n+1}) + (3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}, 2\tilde{\boldsymbol{u}}_h^{n+1} - 2\boldsymbol{u}_h^{n+1}) \\ &\qquad+ (3\tilde{\boldsymbol{u}}_h^{n+1} - 3\boldsymbol{u}_h^{n+1}, 2\tilde{\boldsymbol{u}}_h^{n+1}). \end{aligned} \tag{3.67}$$

Using the identity

$$(2a, 3a - 4b + c) = a^2 - b^2 + (2a - b)^2 - (2b - c)^2 + (a - 2b + c)^2,$$

we can manipulate the first term in the right hand side of (3.67) as follows:

$$\begin{aligned}&(3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}, 2\boldsymbol{u}_h^{n+1}) \\ &\quad= \|\boldsymbol{u}_h^{n+1}\|_0^2 - \|\boldsymbol{u}_h^n\|_0^2 + \|2\boldsymbol{u}_h^{n+1} - \boldsymbol{u}_h^n\|_0^2 - \|2\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}\|_0^2 + \|\delta^2\boldsymbol{u}_h^{n+1}\|_0^2. \end{aligned} \tag{3.68}$$

Using (3.65c) we can express the second term of the right hand side of (3.67) as

$$\begin{aligned}(3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}, 2\tilde{\boldsymbol{u}}_h^{n+1} - 2\boldsymbol{u}_h^{n+1}) &= \frac{4\delta t}{3}(3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}, \nabla(\delta p_h^{n+1})) \\ &= -\frac{4\delta t}{3}(\nabla\cdot(3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}), \delta p_h^{n+1}). \end{aligned} \tag{3.69}$$

Manipulating (3.65c) we get

$$\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^{n+1} = \frac{2\delta t}{3}(\boldsymbol{\pi}_1^{n+1} - \boldsymbol{\pi}_1^n), \tag{3.70}$$

and then we can obtain, from (3.65b),

$$(q_h, \nabla\cdot \boldsymbol{u}_h^{n+1}) = -\frac{2\delta t}{3}(\boldsymbol{\pi}_{23}^{n+1} - \boldsymbol{\pi}_{23}^{n}, \nabla q_h) - \tau(\nabla p_h^{n+1} - \boldsymbol{\pi}_h^{n+\beta}, \nabla q_h)$$
$$= -\frac{2\delta t}{3}(\delta\boldsymbol{\pi}_{23}^{n+1}, \nabla q_h) - \tau(\boldsymbol{\pi}_3^{n+1}, \nabla q_h) - \tau\bar{\beta}(\delta\boldsymbol{\pi}_h^{n+1}, \nabla q_h), \tag{3.71}$$

being $\bar{\beta} = 1-\beta$. Using (3.71) in (3.69) we have

$$(3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}, 2\tilde{\boldsymbol{u}}_h^{n+1} - 2\boldsymbol{u}_h^{n+1}) = \frac{4\delta t^2}{9}(3\delta\boldsymbol{\pi}_{23}^{n+1} - 4\delta\boldsymbol{\pi}_{23}^{n} + \delta\boldsymbol{\pi}_{23}^{n-1}, 2\delta\boldsymbol{\pi}_{23}^{n+1})$$
$$+ \frac{2\delta t\tau}{3}(3\boldsymbol{\pi}_3^{n+1} - 4\boldsymbol{\pi}_3^{n} + \boldsymbol{\pi}_3^{n-1}, 2\delta\boldsymbol{\pi}_3^{n+1}) + \frac{2\delta t\tau\bar{\beta}}{3}(3\delta\boldsymbol{\pi}_h^{n+1} - 4\delta\boldsymbol{\pi}_h^{n} + \delta\boldsymbol{\pi}_h^{n}, 2\delta\boldsymbol{\pi}_h^{n+1})$$
$$= \frac{4\delta t^2}{9}(\|\delta\boldsymbol{\pi}_{23}^{n+1}\|_0^2 - \|\delta\boldsymbol{\pi}_{23}^{n}\|_0^2 + \|2\delta\boldsymbol{\pi}_{23}^{n+1} - \delta\boldsymbol{\pi}_{23}^{n}\|_0^2 - \|2\delta\boldsymbol{\pi}_{23}^{n} - \delta\boldsymbol{\pi}_{23}^{n-1}\|_0^2 + \|\delta^3\boldsymbol{\pi}_{23}^{n+1}\|_0^2)$$
$$+ \frac{2\delta t\tau\bar{\beta}}{3}(\|\delta\boldsymbol{\pi}_h^{n+1}\|_0^2 - \|\delta\boldsymbol{\pi}_h^{n}\|_0^2 + \|2\delta\boldsymbol{\pi}_h^{n+1} - \delta\boldsymbol{\pi}_h^{n}\|_0^2 - \|2\delta\boldsymbol{\pi}_h^{n} - \delta\boldsymbol{\pi}_h^{n-1}\|_0^2 + \|\delta^3\boldsymbol{\pi}_h^{n+1}\|_0^2)$$
$$+ \frac{2\delta t\tau}{3}(5\|\delta\boldsymbol{\pi}_3^{n+1}\|_0^2 - \|\delta\boldsymbol{\pi}_3^{n}\|_0^2 + \|\delta^2\boldsymbol{\pi}_3^{n+1}\|_0^2). \tag{3.72}$$

For the last term in (3.67) we get

$$(3\tilde{\boldsymbol{u}}_h^{n+1} - 3\boldsymbol{u}_h^{n+1}, 2\tilde{\boldsymbol{u}}_h^{n+1}) = 3\|\tilde{\boldsymbol{u}}_h^{n+1}\|_0^2 - 3\|\boldsymbol{u}_h^{n+1}\|_0^2 + 3\|\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^{n+1}\|_0^2. \tag{3.73}$$

Using (3.68), (3.72) and (3.73) in (3.67), and applying the result in (3.66) it is found that,

$$3\|\tilde{\boldsymbol{u}}_h^{n+1}\|_0^2 - 3\|\boldsymbol{u}_h^{n+1}\|_0^2 + 3\|\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^{n+1}\|_0^2 + \|\boldsymbol{u}_h^{n+1}\|_0^2 - \|\boldsymbol{u}_h^{n}\|_0^2$$
$$+ \|2\boldsymbol{u}_h^{n+1} - \boldsymbol{u}_h^{n}\|_0^2 - \|2\boldsymbol{u}_h^{n} - \boldsymbol{u}_h^{n-1}\|_0^2 + \|\delta^2\boldsymbol{u}_h^{n+1}\|_0^2 + 2\delta t\nu\|\tilde{\boldsymbol{u}}_h^{n+1}\|_1^2$$
$$+ \frac{4\delta t^2}{9}(\|\delta\boldsymbol{\pi}_{23}^{n+1}\|_0^2 - \|\delta\boldsymbol{\pi}_{23}^{n}\|_0^2 + \|2\delta\boldsymbol{\pi}_{23}^{n+1} - \delta\boldsymbol{\pi}_{23}^{n}\|_0^2 - \|2\delta\boldsymbol{\pi}_{23}^{n} - \delta\boldsymbol{\pi}_{23}^{n-1}\|_0^2 + \|\delta^3\boldsymbol{\pi}_{23}^{n+1}\|_0^2)$$
$$+ \frac{2\delta t\tau\bar{\beta}}{3}(\|\delta\boldsymbol{\pi}_h^{n+1}\|_0^2 - \|\delta\boldsymbol{\pi}_h^{n}\|_0^2 + \|2\delta\boldsymbol{\pi}_h^{n+1} - \delta\boldsymbol{\pi}_h^{n}\|_0^2 - \|2\delta\boldsymbol{\pi}_h^{n} - \delta\boldsymbol{\pi}_h^{n-1}\|_0^2 + \|\delta^3\boldsymbol{\pi}_h^{n+1}\|_0^2)$$
$$+ \frac{2\delta t\tau}{3}(5\|\delta\boldsymbol{\pi}_3^{n+1}\|_0^2 - \|\delta\boldsymbol{\pi}_3^{n}\|_0^2 + \|\delta^2\boldsymbol{\pi}_3^{n+1}\|_0^2) + 4\delta t(\nabla p_h^n, \tilde{\boldsymbol{u}}_h^{n+1})$$
$$\leq 2\delta tC\|\bar{\boldsymbol{f}}^{n+1}\|_{-1}^2. \tag{3.74}$$

The only term that is not bounded is the last one on the left hand side of (3.74). From (3.70) we can take $\boldsymbol{v}_h = 3\boldsymbol{u}_h^{n+1}/(2\delta t) + \boldsymbol{\pi}_1^{n+1} = 3\tilde{\boldsymbol{u}}_h^{n+1}/(2\delta t) + \boldsymbol{\pi}_1^n$ in (3.65c), yielding

$$(\frac{3}{2\delta t}\boldsymbol{u}_h^{n+1} + \nabla p_h^{n+1}, \frac{3}{2\delta t}\boldsymbol{u}_h^{n+1} + \boldsymbol{\pi}_1^{n+1}) = (\frac{3}{2\delta t}\tilde{\boldsymbol{u}}_h^{n+1} + \nabla p_h^{n}, \frac{3}{2\delta t}\tilde{\boldsymbol{u}}_h^{n+1} + \boldsymbol{\pi}_1^{n}). \tag{3.75}$$

After expanding this equality it is found that

$$\frac{9}{4\delta t^2}\|\boldsymbol{u}_h^{n+1}\|_0^2 + \frac{3}{\delta t}(\boldsymbol{u}_h^{n+1}, \nabla p_h^{n+1}) + \|\boldsymbol{\pi}_1^{n+1}\|_0^2$$
$$= \frac{9}{4\delta t^2}\|\tilde{\boldsymbol{u}}_h^{n+1}\|_0^2 + \frac{3}{\delta t}(\tilde{\boldsymbol{u}}_h^{n+1}, \nabla p_h^{n}) + \|\boldsymbol{\pi}_1^{n}\|_0^2. \tag{3.76}$$

Using (3.71) with $q_h = p_h^{n+1}$ we get for the second term in the left hand side of (3.76) that

$$\frac{3}{\delta t}(\boldsymbol{u}_h^{n+1}, \nabla p_h^{n+1}) = -\frac{3}{\delta t}(\nabla\cdot\boldsymbol{u}_h^{n+1}, p_h^{n+1})$$
$$= \|\boldsymbol{\pi}_{23}^{n+1}\|_0^2 - \|\boldsymbol{\pi}_{23}^{n}\|_0^2 + \|\delta\boldsymbol{\pi}_{23}^{n+1}\|_0^2 + \frac{3\tau}{\delta t}\|\boldsymbol{\pi}_3^{n+1}\|_0^2$$
$$+ \frac{3\tau\bar{\beta}}{2\delta t}(\|\boldsymbol{\pi}_h^{n+1}\|_0^2 - \|\boldsymbol{\pi}_h^{n}\|_0^2 + \|\delta\boldsymbol{\pi}_h^{n+1}\|_0^2). \tag{3.77}$$

Multiplying (3.77) by $\frac{4}{3}\delta t^2$ and adding (3.76) to (3.74) we obtain

$$\begin{aligned}
&\|\boldsymbol{u}_h^{n+1}\|_0^2 - \|\boldsymbol{u}_h^n\|_0^2 + 3\|\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^{n+1}\|_0^2 \\
&+ \|2\boldsymbol{u}_h^{n+1} - \boldsymbol{u}_h^n\|_0^2 - \|2\boldsymbol{u}_h^n - \boldsymbol{u}_h^{n-1}\|_0^2 + \|\delta^2\boldsymbol{u}_h^{n+1}\|_0^2 + 2\delta t\nu\|\tilde{\boldsymbol{u}}_h^{n+1}\|_1^2 \\
&+ \frac{4\delta t^2}{9}(\|\delta\boldsymbol{\pi}_{23}^{n+1}\|_0^2 - \|\delta\boldsymbol{\pi}_{23}^n\|_0^2 + \|2\delta\boldsymbol{\pi}_{23}^{n+1} - \delta\boldsymbol{\pi}_{23}^n\|_0^2 - \|2\delta\boldsymbol{\pi}_{23}^n - \delta\boldsymbol{\pi}_{23}^{n-1}\|_0^2 + \|\delta^3\boldsymbol{\pi}_{23}^{n+1}\|_0^2) \\
&+ \frac{2\delta t\tau\bar{\beta}}{3}(\|\delta\boldsymbol{\pi}_h^{n+1}\|_0^2 - \|\delta\boldsymbol{\pi}_h^n\|_0^2 + \|2\delta\boldsymbol{\pi}_h^{n+1} - \delta\boldsymbol{\pi}_h^n\|_0^2 - \|2\delta\boldsymbol{\pi}_h^n - \delta\boldsymbol{\pi}_h^{n-1}\|_0^2 + \|\delta^3\boldsymbol{\pi}_h^{n+1}\|_0^2) \\
&+ \frac{2\delta t\tau}{3}(5\|\delta\boldsymbol{\pi}_3^{n+1}\|_0^2 - \|\delta\boldsymbol{\pi}_3^n\|_0^2 + \|\delta^2\boldsymbol{\pi}_3^{n+1}\|_0^2) + \frac{4\delta t^2}{3}(\|\boldsymbol{\pi}_{23}^{n+1}\|_0^2 - \|\boldsymbol{\pi}_{23}^n\|_0^2 + \|\delta\boldsymbol{\pi}_{23}^{n+1}\|_0^2) \\
&+ 4\delta t\tau\|\boldsymbol{\pi}_3^{n+1}\|_0^2 + 4\delta t\tau\bar{\beta}(\|\boldsymbol{\pi}_h^{n+1}\|_0^2 - \|\boldsymbol{\pi}_h^n\|_0^2 + \|\delta\boldsymbol{\pi}_h^{n+1}\|_0^2) + \frac{4\delta t^2}{3}(\|\boldsymbol{\pi}_1^{n+1}\|_0^2 - \|\boldsymbol{\pi}_1^n\|_0^2) \\
&\leq 2\delta tC\|\bar{\boldsymbol{f}}^{n+1}\|_{-1}^2. \qquad (3.78)
\end{aligned}$$

From (3.70) we know that

$$\|\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^{n+1}\|_0^2 = \frac{4}{9}\delta t^2\|\delta\boldsymbol{\pi}_1^{n+1}\|_0^2, \qquad (3.79)$$

that used in (3.78), adding up the result from $n = 1$ to an arbitrary level set $N - 1$ and neglecting some positive terms yields

$$\begin{aligned}
&\|\boldsymbol{u}_h^N\|_0^2 + \sum_{n=1}^N \|\tilde{\boldsymbol{u}}_h^n - \boldsymbol{u}_h^n\|_0^2 + \sum_{n=1}^N \|\delta^2\boldsymbol{u}_h^n\|_0^2 + \nu\sum_{n=1}^N \delta t\|\tilde{\boldsymbol{u}}_h^n\|_1^2 + \|\delta t\nabla p_h^N\|_0^2 \\
&\quad + \sum_{n=1}^N \delta t\|\sqrt{\delta t}\nabla\delta p_h^n\|_0^2 + \sum_{n=1}^N \delta t\|\sqrt{\tau}\boldsymbol{\pi}_3^n\|_0^2 \leq C. \qquad (3.80)
\end{aligned}$$

The stability results we already have obtained from (3.80) are

$$\{\boldsymbol{u}_h^n\} \in \ell^\infty(L^2),\ \{\tilde{\boldsymbol{u}}_h^n\} \in \ell^2(H^1),\ \{\sqrt{\delta t}\nabla\delta p_h^n\} \in \ell^2(L^2),\ \{\delta t\nabla p_h^n\} \in \ell^\infty(L^2).$$

On the other hand, from (3.79) and the previous stability results it is obtained that

$$\|\tilde{\boldsymbol{u}}_h^{n+1}\|_0^2 \leq \|\boldsymbol{u}_h^{n+1}\|_0^2 + \frac{4}{9}\delta t^2\|\boldsymbol{\pi}_1^{n+1}\|_0^2 \leq C \qquad (3.81)$$

and then $\{\tilde{\boldsymbol{u}}_h^n\} \in \ell^\infty(L^2)$.

The proof of the stability results expected will conclude bounding $\{\boldsymbol{\pi}_1^n\}$ in $\ell^1(L^2)$. This stability bound will be extracted from the momentum equation actually solved, obtained after adding (3.65a) and (3.65c),

$$\begin{aligned}
&\frac{1}{2\delta t}(3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}, \boldsymbol{v}_h) + (\tilde{\boldsymbol{u}}_h^{n+1}\cdot\nabla\tilde{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h) + \nu(\nabla\tilde{\boldsymbol{u}}_h^{n+1}, \nabla\boldsymbol{v}_h) \\
&\qquad + (\nabla p_h^{n+1}, \boldsymbol{v}_h) = \langle\bar{\boldsymbol{f}}^{n+1}, \boldsymbol{v}_h\rangle \qquad \forall\boldsymbol{v}_h \in \boldsymbol{V}_h. \qquad (3.82)
\end{aligned}$$

Taking $\boldsymbol{v}_h = \boldsymbol{\pi}_1^{n+1}$ and using the standard inverse estimate

$$\|\boldsymbol{\pi}_1^n\|_1 \leq \frac{C_{\text{inv}}}{h}\|\boldsymbol{\pi}_1^n\|_0, \qquad (3.83)$$

valid for quasi-uniform finite element partitions, we have

$$\|\boldsymbol{\pi}_1^{n+1}\|_0^2 \leq \|\bar{\boldsymbol{f}}^{n+1}\|_{-1}\frac{C_{\text{inv}}}{h}\|\boldsymbol{\pi}_1^{n+1}\|_0 + \frac{1}{2\delta t}\|3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}\|_0\|\boldsymbol{\pi}_1^{n+1}\|_0 + (N_a\|\tilde{\boldsymbol{u}}_h^{n+1}\|_1 + N_c\|\tilde{\boldsymbol{u}}_h^{n+1}\|_1^2)\frac{C_{\text{inv}}}{h}\|\boldsymbol{\pi}_1^{n+1}\|_0, \tag{3.84}$$

where N_a and N_c are the norms of the viscous and convective terms, respectively. Dividing (3.84) by $\|\boldsymbol{\pi}_1^{n+1}\|$, summing up to N and noting that $\sqrt{\tau} \leq Ch$, we now have that

$$\sum_{n=1}^{N-1}\sqrt{\tau}\delta t\|\boldsymbol{\pi}_1^{n+1}\| \leq \sum_{n=1}^{N-1} C\|\bar{\boldsymbol{f}}^{n+1}\|_{-1}\delta t + \sum_{n=1}^{N-1}\frac{\sqrt{\tau}}{2}\|3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}\|_0 + \sum_{n=1}^{N-1} C\delta t(N_a\|\tilde{\boldsymbol{u}}_h^{n+1}\|_1 + N_c\|\tilde{\boldsymbol{u}}_h^{n+1}\|_1^2). \tag{3.85}$$

The only term that still has not been bounded is the second one on the right-hand side. We can bound this term as follows,

$$\sum_{n=1}^{N-1}\frac{\sqrt{\tau}}{2}\|3\boldsymbol{u}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}\|_0 \leq \sum_{n=1}^{N-1}\sqrt{\tau}\|\boldsymbol{u}_h^{n+1} - \tilde{\boldsymbol{u}}_h^{n+1}\|_0 + \sum_{n=1}^{N-1}\sqrt{\tau}\|\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n\|_0 + \sum_{n=1}^{N-1}\frac{\sqrt{\tau}}{2}\|\delta^2\boldsymbol{u}_h^{n+1}\|_0. \tag{3.86}$$

From (3.80) we have stability over the first and the third terms. For the second term we take $\boldsymbol{v}_h = \tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n$ in (3.65a), yielding

$$\frac{1}{2\delta t}(3\tilde{\boldsymbol{u}}_h^{n+1} - 4\boldsymbol{u}_h^n + \boldsymbol{u}_h^{n-1}, \tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n) + (\tilde{\boldsymbol{u}}_h^{n+1}\cdot\nabla\tilde{\boldsymbol{u}}_h^{n+1}, \tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n) + \nu(\nabla\tilde{\boldsymbol{u}}_h^{n+1}, \nabla\tilde{\boldsymbol{u}}_h^{n+1} - \nabla\boldsymbol{u}_h^n) + (\nabla p_h^n, \tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n) = \langle\bar{\boldsymbol{f}}^{n+1}, \tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n\rangle \tag{3.87}$$

from where it is easily obtained, adding up to N and using the inverse estimate (3.83),

$$\begin{aligned}
&\sum_{n=1}^{N}\sqrt{\tau}\|\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n\|_0^2 \leq \sum_{n=1}^{N}\delta t A_n\|\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n\|_0 + \sum_{n=1}^{N}\delta t\sqrt{\tau}B_n,\\
&\quad A_n := \frac{\sqrt{\tau}}{2\delta t}(\|\delta^2\boldsymbol{u}_h^{n+1}\|_0 + \|\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^{n+1}\|_0) +\\
&\qquad \sqrt{\tau}\frac{C_{\text{inv}}}{h}(N_a\|\tilde{\boldsymbol{u}}_h^{n+1}\|_1 + N_c\|\tilde{\boldsymbol{u}}_h^{n+1}\|_1^2 + \|\boldsymbol{f}^{n+1}\|_{-1}),\\
&\quad B_n := (p_h^n, \nabla\cdot\tilde{\boldsymbol{u}}_h^{n+1}) - (p_h^n, \nabla\cdot\boldsymbol{u}_h^n).
\end{aligned} \tag{3.88}$$

With the stability results already obtained we know that $\sum_{n=1}^{N}\delta t A_n \leq C$ and $\sup_n\|\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n\| \leq C$, which is enough to say that

$$\sum_{n=1}^{N}\delta t A_n\|\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n\|_0 \leq \left(\sum_{n=1}^{N}\delta t A_n\right)\sup_n\|\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n\|_0 \leq C < \infty. \tag{3.89}$$

It only remains to bound $\sum_{n=1}^{N} \delta t\sqrt{\tau} B_n$ and the proof will be finished. Taking $q_h = p_h^n$ in (3.65b) and adding up to $N-1$, we get

$$\sum_{n=1}^{N-1} \delta t(p_h^n, \nabla \cdot \tilde{\boldsymbol{u}}_h^{n+1}) = \sum_{n=1}^{N-1} \left[-\frac{2\delta t^2}{3}(\nabla \delta p_h^{n+1}, \nabla p_h^n) - \delta t\tau(\nabla p_h^{n+1} - \boldsymbol{\pi}_h^{n+\beta}, \nabla p_h^n) \right]$$
$$\leq C \sum_{n=1}^{N-1} \delta t^2 (\|\nabla \delta p_h^{n+1}\|_0^2 + \|\boldsymbol{\pi}_3^{n+1}\|_0^2 + \|\boldsymbol{\pi}_3^n\|_0^2 + \bar{\beta}\|\boldsymbol{\pi}_h^{n+1} - \boldsymbol{\pi}_h^n\|_0^2) < \infty, \quad (3.90)$$

and doing the same with (3.71) it follows that

$$-\sum_{n=1}^{N-1} \delta t(p_h^n, \nabla \cdot \boldsymbol{u}_h^n) = \sum_{n=1}^{N} \frac{2\delta t^2}{3}(\|\boldsymbol{\pi}_{23}^n\|_0^2 - \|\boldsymbol{\pi}_{23}^{n-1}\|_0^2 + \|\delta \boldsymbol{\pi}_{23}^n\|_0^2)$$
$$+ \sum_{n=1}^{N} \delta t\tau[\|\boldsymbol{\pi}_3^n\|_0^2 + \bar{\beta}(\|\boldsymbol{\pi}_h^n\|_0^2 - \|\boldsymbol{\pi}_h^{n-1}\|_0^2 + \|\delta \boldsymbol{\pi}_h^n\|_0^2)] < \infty. \quad (3.91)$$

With all these results the proof of stability has been finished.□

Remark 3.5. The stabilization of the convective term

$$\tau(\boldsymbol{u}_h^{n+1} \cdot \nabla \boldsymbol{u}_h^{n+1} - \Pi_h(\boldsymbol{u}_h^{n+\beta} \cdot \nabla \boldsymbol{u}_h^{n+\beta}), \boldsymbol{u}_h^{n+1} \cdot \nabla \boldsymbol{v}_h) \quad (3.92)$$

has not been considered in the previous analysis. Obviously, for $\beta = 1$, this bilinear form is positive semidefinite, and it can be easily added to K_{visc}. We can also do that using GLS or SUPG. Further, we have to exploit the expression of τ (2.18a) in (3.84) and (3.88). For $\beta = 0$ we have to proceed as for the pressure in Eq. (3.71).

3.4 Numerical tests

In this section we present some numerical results to test the time integration schemes described in this chapter. Even though all the exposition has been based on the Galerkin method for the spatial discretization, in the following examples we have used equal velocity-pressure interpolation and the OSS pressure stabilization technique. In particular, we have taken $k_{\text{q}} = k_{\text{v}} = 1$, with the notation of Section 1.3.

3.4.1 Convergence test

The first example we consider is a simple convergence test whose goal is to check numerically the rate of convergence in time for some of the numerical methods described.

The computational domain is the unit square, discretized using a uniform triangular mesh of 11×11 nodal points (200 triangles). The boundary and initial conditions and the force term are prescribed so that the analytic solution is $\boldsymbol{u} = (y, -x)\sin(\pi t/10)\exp(t/25)$ and $p = 0$. Note that the exact solution belongs to the finite element space, and thus the only source of numerical error is the time approximation. The nonlinear term of the Navier-Stokes equations is neglected (it is zero for the exact solution), that is, we consider the transient Stokes problem.

Results are shown in Figure 3.1. The error E is measured in the ℓ^2 norm of the sequence $\{\boldsymbol{u}^n - \boldsymbol{u}(t^n)\}$. It is seen that all the methods show the expected rate of convergence. This is particularly relevant for the predictor corrector schemes, whose error is affected by the convergence tolerance adopted in the iterative loop of each time step.

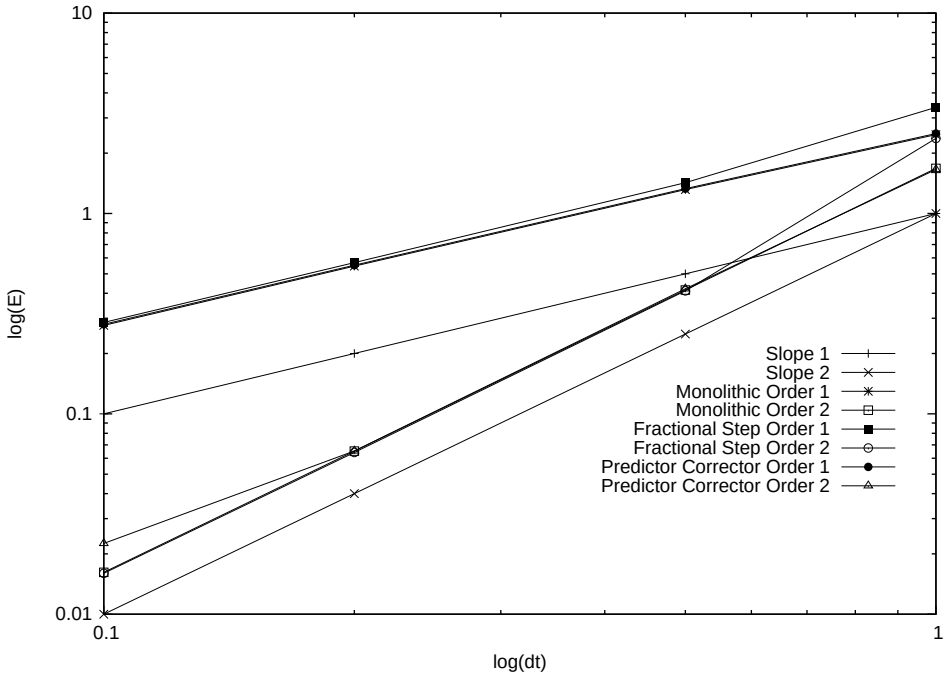


Figure 3.1: Convergence test

3.4.2 Flow in a cavity

In this second example we solve the classical cavity flow problem at a Reynolds number Re = 100. The computational domain is the unit square, discretized using a mesh of 21×21 nodal points (400 triangles). The velocity is fixed to zero everywhere except on the top boundary, where it is prescribed to (1,0).

Even though the solution in this simple example is stationary, we obtain it by stepping in time. The goal of this test is precisely to check the properties of the schemes proposed for the long-term time integration of stationary solutions (very often difficult to obtain in a stationary calculation) and, particularly, their numerical dissipation. The time step employed is $\delta t = 1$.

Figure 3.2 shows the evolution towards the steady state obtained. It is observed that the monolithic scheme is more dissipative than the fractional step method, particularly for the second order scheme (BDF2-SE2). In this particular example, BDF2 seems to be slightly more dissipative than BDF1 for the monolithic case.

When the predictor-corrector scheme is employed, the evolution towards the steady-state depends on the final error of the iterative scheme within each time step. For loose convergence requirements, it is expected that the predictor-corrector method will behave in a way similar to the fractional step method, whereas the behavior will approach that of the monolithic scheme as the iterative error per time step decreases. This is what is observed in Figure 3.3. With a small tolerance (10^{-6}) the error is directly given by the number of iterations allowed per time step. It is observed that for 20 iteration the behavior in time is similar to that of the monolithic scheme, whereas for 5 iterations it is much less dissipative.

3.4.3 Flow over a cylinder

The last example is also a classical benchmark, namely, the flow over a cylinder. The computational domain is $\bar{\Omega} = [0, 16] \times [0, 8] \setminus D$, with the cylinder D of diameter 1 and

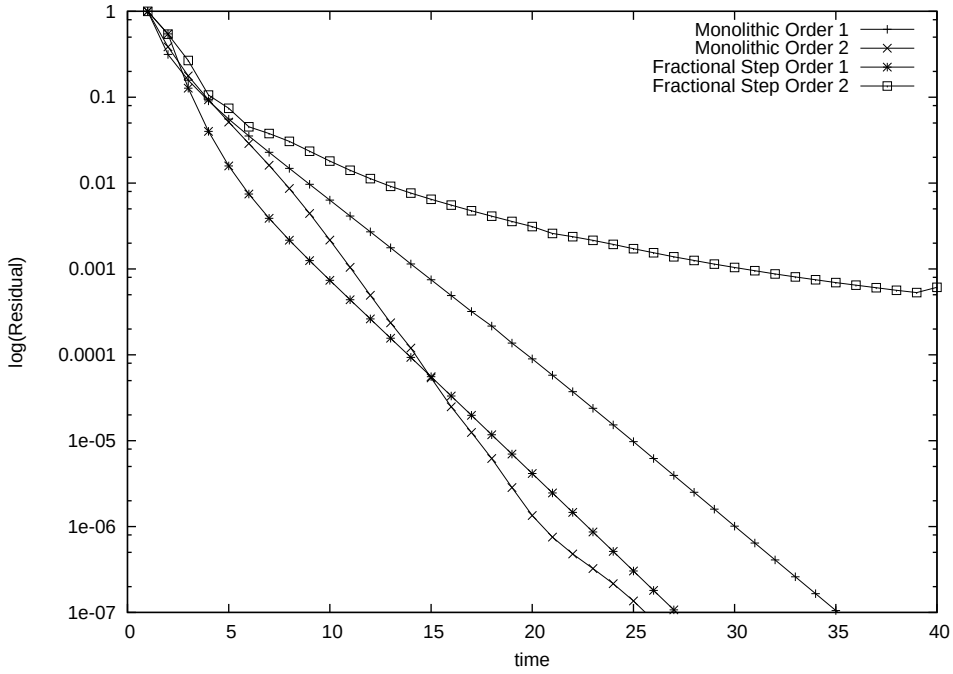


Figure 3.2: Evolution towards the steady-state for the cavity flow problem using different schemes

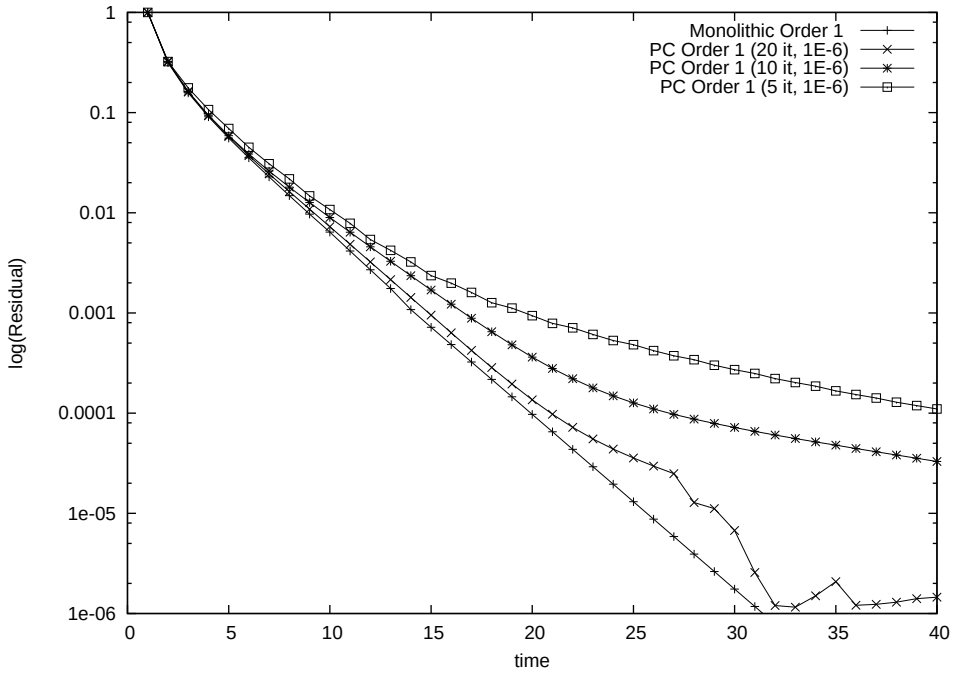


Figure 3.3: Time dissipation of the predictor-corrector scheme in terms of the time step iterations

centered at $(4, 4)$. The velocity at $x = 0$ is prescribed to $(1, 0)$, whereas at $y = 0$ and $y = 8$ the y-velocity component is prescribed to 0 and the x-component is left free. The outflow boundary (where both the x- and y-components are free) is $x = 16$. The Reynolds number is 100, based on the cylinder diameter and the prescribed inflow velocity. The finite element mesh employed consists of 3604 linear triangles, with 1902 nodal points. A snapshot of the contours of the velocity norm and pressure isolines is shown in Figures 3.4 and 3.5. The purpose of this example is not to compare the quality of these results with those presented in the literature, but rather to discuss the time behavior of the schemes proposed in this paper.

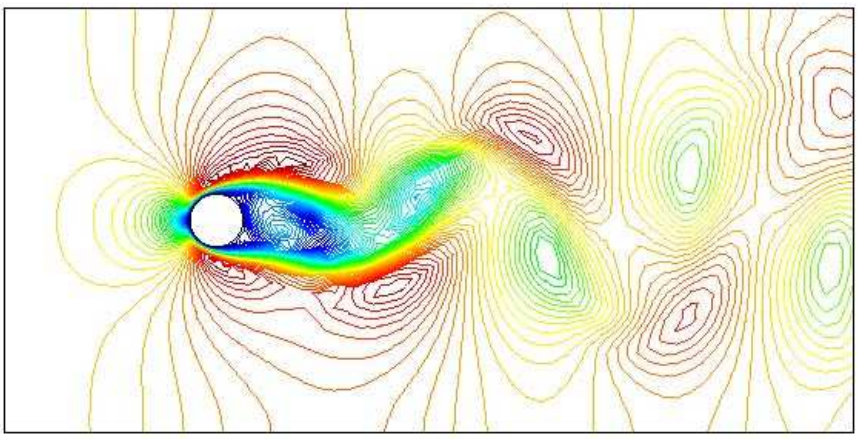


Figure 3.4: Contours of velocity norm for the flow over a cylinder

The evolution of the y-velocity component at the control point located at (6,4) is shown in Figure 3.6. The time step size used in all the cases is $\delta t = 0.05$. The tolerance of the iterative procedure of each time step has been set to 0.0001 (0.01%), although a maximum of only 5 iterations has been permitted. It is observed that the least dissipative scheme (with higher frequency and amplitude) is the second order monolithic method (using BDF2 for the time integration), and the most dissipative one (with the smaller frequency and smaller amplitude) is BDF1-SE1. This example serves to show that even though fractional step schemes are attractive for their low computational cost, they are usually *less* accurate for a given time step size than monolithic methods. Likewise, for this particular example predictor-corrector methods lie in between fractional-step and monolithic schemes (recall that only 5 iterations have been allowed per time step). The improvement with respect to fractional step methods is particularly significant for the first order method.

Finally, concerning the number of iterations performed (for a tolerance of 0.01%) it is interesting to study the effect of the initial guess. Independently of the time integration employed, it is always possible to use a second order extrapolation for the unknowns as given by (3.33a)-(3.33b) to start the iterative procedure. The following table shows

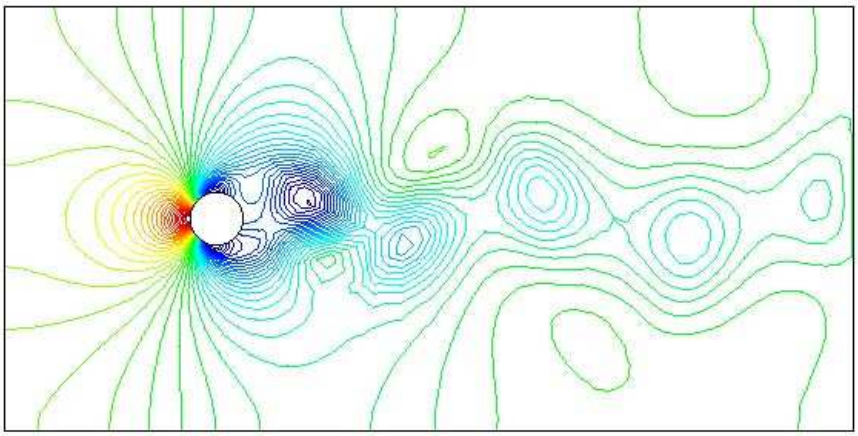


Figure 3.5: Contours of pressure for the flow over a cylinder

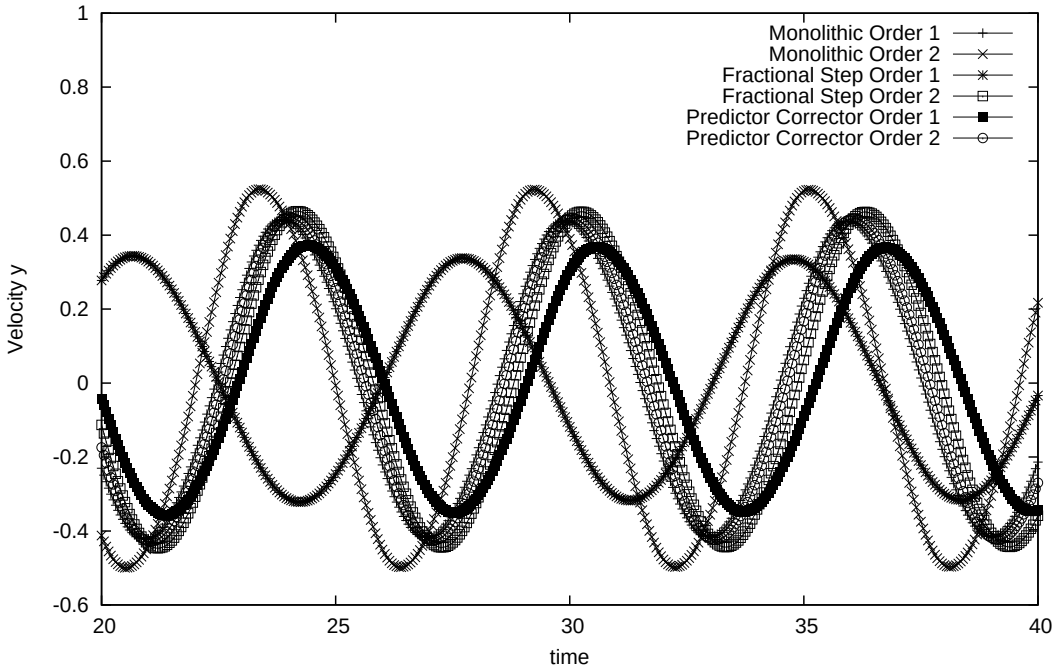


Figure 3.6: Temporal evolution of the y-velocity component at the control point

the influence of how the initial guess is taken. The numbers listed are the total number of iterations from $t = 0$ to $t = 10$ using two time steps sizes, predictor-corrector and monolithic schemes of second order and first or second order extrapolations to take the initial guess (X refers here to both velocity and pressure degrees of freedom). It is observed that the monolithic schemes need fewer iterations, which can be easily explained since the iterations for the predictor-corrector scheme need to deal not only with the nonlinearity of the problem but also with the velocity-pressure coupling to converge to the monolithic solution. Nevertheless, what is interesting is that an important gain in the total number of iterations is obtained using the second order extrapolation. For first order schemes, it implies that the unknowns at two previous time steps need to be stored (which is unnecessary) but it certainly pays off in view of the significant reduction in the calculation time.

	PC using BDF2		Monolithic using BDF2	
	$\delta t = 0.10$	$\delta t = 0.01$	$\delta t = 0.10$	$\delta t = 0.01$
$\mathsf{X}^{n+1,0} = \mathsf{X}^n$	414	2025	300	2000
$\mathsf{X}^{n+1,0} = 2\mathsf{X}^n - \mathsf{X}^{n-1}$	325	1046	204	1006

3.5 Conclusions

In this chapter we have discussed several ways to implement first and second order time integration schemes for the incompressible Navier-Stokes equations whose objective is to uncouple the calculation of the velocity and the pressure.

Taking as a reference the monolithic approach, the advantage of all these methods is that they are more effective from the computational point of view, although other aspects need to be taken into account.

Fractional steps methods based on a pressure Poisson equation offer some pressure stability independent of the space approximation. We have presented here stability results for four of these methods that show this. The inherent stability of first order methods is reduced when second order methods are used. Further, we have analyzed a second order fractional step method stabilized using OSS. As expected, the stability bound for the pressure can be made optimal by properly choosing the stabilization parameter τ.

Numerical evidence also shows that they are less dissipative in the evolution towards a steady solution, and thus less effective when a transient calculation is used to reach a steady state. Only one example showing this effect has been presented here, but we have observed this in many cases. Likewise, for a given time step they tend to be less accurate than monolithic methods, in spite of the fact that the rate is convergence be optimal. Again, an example of this behavior has been presented in this paper.

Another family of methods discussed is what we have called momentum-pressure Poisson equation with an explicit treatment of the pressure gradient in the momentum equation. Even though the derivation of these methods is different, we have shown that the bottom line is a method that can be considered a fractional step one.

Finally, predictor-corrector schemes can be considered as a way to avoid the shortcomings of fractional step methods while allowing to uncouple the velocity and pressure calculations. As it has been shown in the last numerical example presented, the price is a higher number of iterations to be done per time step. We believe that in some cases this can be worthy (see also [56]). Also in this last example and as a by-product, we have

observed that using as initial guess for the iterations within each time step a second order extrapolation of the unknowns leads to significantly fewer iterations. This also can be worthy in many cases and for different methods.

Chapter 4

Numerical Analysis of a first order Pressure Correction method

In this chapter we analyze a classical first order projection method in its fully discrete version. We consider two possible situations for the space discretization: the use of an stabilized finite element method or finite element interpolation spaces for velocity and pressure holding the inf-sup condition. Reference [5] contains part of the following analysis.

4.1 Introduction

This chapter is devoted to the analysis of the most classical pressure segregation method. Since its appearence in the late 1960's, with the pioneering works of Chorin [40] and Temam [157], this method has enjoyed a widespread popularity. Its key feature is the decoupling of the velocity and pressure calculation, yielding an important reduction of computational cost. We refer to the previous chapter for an introduction to this kind of methods.

The fractional step method has been studied analitically in order to prove convergence and obtain error estimates. For periodic boundary conditions and a centered finite difference space approximation, Chorin proved in [41] that the convergence was of first order in the time step size δt, provided $\delta t = \mathcal{O}(h^2)$, h being the diameter of the space discretization. A proof of convergence of the semidiscrete (space continuous) projection method to a continuous solution $\boldsymbol{u}(t)$ was given by Temam (see [157]) in a general setting.

The pioneering work of Shen [147] analyzed the semi-implicit version of the semidiscrete projection method, taking the convective term as $\boldsymbol{u}^n \cdot \nabla \hat{\boldsymbol{u}}^{n+1}$. He obtained the first error estimates for the velocity, but in a weak norm, $L^2(0,T;\boldsymbol{L}^2(\Omega))$. The proof of the error estimates in this work is plagued by a mistake when using dual norms which was originally observed by Guermond in [85]. In [149] these mistakes were corrected. Furthermore, for obtaining the pressure estimates, $p_t \in L^2(0,T;H^1(\Omega)/\mathbb{R})$ has to be assumed. This requirement is not appropriate, because of the lack of compatibility with the given data at time $t \to 0$ (see [97]). In this chapter, we use both $(\cdot)_t$ and $\partial_t(\cdot)$ to denote the partial time derivative.

Rannacher proposed an alternative version of the projection method in [140]. In this new format the end-of-step velocity disappears, and the problem can be understood as a

penalized method with a lag in the evaluation of the pressure in the momentum equation,

$$\frac{1}{\delta t}(\hat{\boldsymbol{u}}^{n+1} - \hat{\boldsymbol{u}}^{n}) - \nu\Delta\hat{\boldsymbol{u}}^{n+1} + \nabla\hat{\boldsymbol{u}}^{n+1} + \nabla p^{n} = \boldsymbol{f}^{n+1} \tag{4.1a}$$

$$\nabla\cdot\hat{\boldsymbol{u}}^{n+1} + \delta t\Delta p^{n+1} = 0. \tag{4.1b}$$

Based on this approach, Prohl obtained sharp error estimates for the semidiscrete in time projection method (see [136]). For instance, first order error estimates for the velocity were obtained with a stronger norm, $L^{\infty}(0,T;\boldsymbol{L}^{2}(\Omega))$.

The improvement of the error estimates for the pressure taking the *rotational form* of the Navier-Stokes equations has been studied in [136] and [93].

Shen obtained improved error estimates for method (3.9) in [146]. Both the intermediate $\hat{\boldsymbol{u}}^{n+1}$ and end-of-step $\boldsymbol{u}^{n+1}$ velocities are first order approximations to the continuous velocities in $L^{\infty}(0,T;\boldsymbol{L}^{2}(\Omega))$.

Error estimates for a fully discrete fractional step method were introduced in [92, 91]. The method analyzed therein is the incremental scheme proposed by Shen. Furthermore, the pressure is not calculated from a Poisson equation, but from a Stokes problem. This analysis requires velocity and pressure interpolations satisfying the inf-sup condition for obtaining error estimates of first order in the time step size and of optimal order in the mesh size.

In this chapter we obtain convergence results for the fully discrete projection method, relying on the inf-sup condition. After that, we study the fully discrete projection method using a Poisson equation for the pressure. But in this case the velocity-pressure interpolations do not need to satisfy the inf-sup condition. In both cases we get error estimates comparing these schemes with an auxiliary problem semidiscretized in time for which we have obtained optimal convergence results. The analysis of the auxiliary problem follows the analysis in [136], due to the similarity of this auxiliary problem with the total projection method studied therein. Then, we get some stability bounds of the auxiliary variables. As far as we are aware, this is the first convergence analysis for the fully discrete first order projection method in the original version proposed by Chorin and Temam.

The chapter is organized as follows: Assumptions for the continuous solution of the Navier-Stokes equations are stated and some regularity properties are presented in Section 4.2. The fractional step method to be analyzed is presented in Section 4.3. In Section 4.4 we introduce a semidiscrete in time auxiliary problem and some convergence and stability results are stated. Section 4.5 is devoted to the analysis of the fully discrete projection scheme satisfying the inf-sup condition. The fully discrete projection scheme using a Poisson equation for the pressure and not relying on the inf-sup condition is presented in Section 4.6. Optimal convergence results are obtained in time and space in both cases. Under the inf-sup condition we pursue an alternative technique (similar to that in [147] at the continuous level) for the proof of *a priori* error estimates.

4.2 Problem statement and preliminaries

Let us recall the evolution equations for an incompressible fluid moving in a domain Ω of $\mathbb{R}^{d}$ ($d = 2$ or 3) in a time interval $(0,T)$. They consist of finding a velocity $\boldsymbol{u}$ and a

pressure p such that

$$\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla p = \boldsymbol{f} \quad \text{in } \Omega \times (0,T), \tag{4.2a}$$

$$\nabla \cdot \boldsymbol{u} = 0 \quad \text{in } \Omega \times (0,T), \tag{4.2b}$$

$$\boldsymbol{u} = \boldsymbol{0} \quad \text{on } \Gamma \times (0,T), \tag{4.2c}$$

$$\boldsymbol{u}(\boldsymbol{x},0) = \boldsymbol{u}_0 \quad \text{in } \Omega \times \{0\}. \tag{4.2d}$$

where $\boldsymbol{f}$ is the force vector, ν is the kynematic viscosity, $\boldsymbol{u}_0$ the initial condition and $\Gamma = \partial\Omega$.

Hereafter, we analyze some specific time and space approximation of system (4.2). We refer to Chapter 1 for the introduction of the ingredients needed for the following analysis.

We point out that the seminorn $|u|_1 = \|\nabla u\|_0$ in $H^1(\Omega)$ is a norm in $H^1_0(\Omega)$. From now onwards in this chapter we will not distinguish between $\|u\|_1$ and $\|\nabla u\|_0$ for $u \in H^1_0(\Omega)$. The skew-symmetric form (1.26) is also used in the following.

Assuming $\boldsymbol{u}_0 \in \boldsymbol{H}^1_0$ and $\boldsymbol{f} \in L^2(0,T;\boldsymbol{H}^{-1}(\Omega))$, and if Ω is bounded and Lipschitz continuous, problem (4.2) has at least one solution $\boldsymbol{u} \in L^\infty(0,T;\boldsymbol{L}^2(\Omega)) \cap L^2(0,T;\boldsymbol{H}^1_0(\Omega))$ (see [159] and Chapter 1).

For the convergence analysis in this chapter we assume more regularity of the solution. We assume that $\boldsymbol{u}(t)$ and $p(t)$ satisfy

(R1) $\boldsymbol{u}(t) \in \mathcal{C}(0,T;\boldsymbol{J}_1) \cap L^\infty(0,T;\boldsymbol{H}^2(\Omega))$, $p \in L^\infty(0,T;H^1(\Omega)) \cap \mathcal{C}(0,T;L^2(\Omega)/\mathbb{R})$

(R2a) $\boldsymbol{u}_t \in L^\infty(0,T;\boldsymbol{L}^2(\Omega))$

(R2b) $\boldsymbol{u}_t \in L^2(0,T;\boldsymbol{H}^1_0(\Omega))$.

Assumptions (R1), (R2a) and (R2b) can be proven assuming for the data

$$\boldsymbol{u}_0 \in \boldsymbol{H}^2(\Omega) \cap \boldsymbol{J}_1, \qquad \boldsymbol{f}(t) \in L^\infty(0,T;\boldsymbol{L}^2(\Omega)) \cap L^2(0,T;\boldsymbol{H}^1(\Omega)),$$

and, if $d = 3$, we also need to assume

$$\boldsymbol{u}(t) \in \boldsymbol{L}^\infty(0,T;\boldsymbol{H}^1(\Omega)),$$

as it is proved in [97]. The last assumption can be replaced by a condition that is satisfied if ν is *large enough* or if $\boldsymbol{f}(t)$ and $\boldsymbol{u}_0$ are *small enough* for a Ω of class $\mathcal{C}^2$ (see [159, Theorems 3.7 and 3.8]). The next assumption will be needed in the convergence analysis.

Assumption 4.1. *Concerning the regularity of the data we assume that:*

$$\boldsymbol{f} \in \mathcal{C}^0(0,T;\boldsymbol{L}^2(\Omega)) \cap L^2(0,T;\boldsymbol{H}^1(\Omega)),$$
$$\boldsymbol{f}_t, \boldsymbol{f}_{tt} \in L^\infty(0,T;\boldsymbol{L}^2(\Omega)),$$
$$\boldsymbol{u}_0 \in \boldsymbol{H}^2(\Omega) \cap \boldsymbol{J}_1$$

and, if $d = 3$,

$$\boldsymbol{u}(t) \in \boldsymbol{L}^\infty(0,T;\boldsymbol{H}^1(\Omega)).$$

This implies the regularity properties (R1), (R2a) and (R2b). Furthermore, the following assumption concerning the regularity of the domain Ω is used,

Assumption 4.2. *The inverse of the Stokes operator* A^{-1} *verifies the regularity property (1.29), with* $s = 2$.

If Ω is of class $\mathcal{C}^2$ this assumption is satisfied.

In order to avoid any incompatible condition as $t \to 0$, let us introduce the weighting function $\lambda := \min\{t, t_0\}$, and its discrete counterpart $\lambda_n := \min\{t^n, t_0\}$, where $t_0 > 0$ is fixed. It allows us to obtain error estimates without taking care of what happens near $t = 0$. We summarize some stability results extracted from [97] in the following lemma.

Lemma 4.1. *Let $\boldsymbol{u}, p$ be the solution of (4.2). Under the regularity Assumptions 4.1, 4.2 and $\partial_t^m \boldsymbol{f} \in L^\infty(0,T;\boldsymbol{L}^2(\Omega))$ for $m \le r$ the following bounds hold,*

$$\sup_{(0,T]} \lambda^{r-1+i/2}\|\partial_t^r \boldsymbol{u}\|_i + \int_0^T \lambda^{2r-1+i}(s)\|\partial_t^{r+1}\boldsymbol{u}(s)\|_i^2 \,\mathrm{d}s \le C,$$

$$\sup_{(0,T]} \lambda^{r}\|\nabla\partial_t^r p\| + \int_0^T \lambda^{2r+1}(s)\|\nabla\partial_t^{r+1} p(s)\|^2 \,\mathrm{d}s \le C,$$

for $r \in \{1,2,3\}$ and $i \in \{0,1,2\}$ with $i + 2r \le 7$ and $r - 1 + \frac{i}{2} \ge 0$.

The previous lemma allows us to obtain stability results for discrete time derivatives. Let us define

$$D_t \boldsymbol{f}(t^{n+1}) = \frac{1}{\delta t}(\boldsymbol{f}(t^{n+1}) - \boldsymbol{f}(t^n)).$$

We will need the following lemma.

Lemma 4.2. *Given a function $g(t)$ regular enough, the following inequalities hold:*

$$\|D_t g(t^{n+1})\|_X \le \|\partial_t g(t^{n+1})\|_X + \frac{1}{\sqrt{2}}\left(\int_0^{\delta t} \xi\|\partial_t^2 g(t^n+\xi)\|_X^2 \,\mathrm{d}\xi\right)^{1/2}, \tag{4.3}$$

$$\|D_t^2 g(t^{n+1})\|_X \le \|\partial_t^2 g(t^{n+1})\|_X + \frac{1}{2}\left(\int_0^{2\delta t} \xi\|\partial_t^3 g(t^n+\xi)\|_X^2 \,\mathrm{d}\xi\right)^{1/2}. \tag{4.4}$$

Proof. These estimates are easily checked from Taylor expansions of $g(t)$ writing the residual in the integral form:

$$D_t g(t^{n+1}) = \partial_t g(t^{n+1}) - \frac{1}{\delta t}\int_0^{\delta t} \xi\partial_t^2 g(t^n+\xi)\,\mathrm{d}\xi, \tag{4.5}$$

$$D_t^2 g(t^{n+1}) = \partial_t^2 g(t^{n+1}) - \frac{1}{\delta t^2}\int_{t^n}^{t^{n+1}} (t^n - t)^2\partial_t^3 g(t)\,\mathrm{d}t + \frac{1}{\delta t^2}\int_{t^{n+1}}^{t^{n+2}} (t^{n+2} - t)^2\partial_t^3 g(t)\,\mathrm{d}t. \tag{4.6}$$

Taking the X-norm of the residual in the first case, we arrive to

$$\begin{aligned}\frac{1}{\delta t}\|\int_0^{\delta t} \xi\partial_t^2 g(t^n+\xi)\,\mathrm{d}\xi\|_X &\le \frac{1}{\delta t}\left(\int_0^{\delta t}\xi\,\mathrm{d}\xi\right)^{1/2}\cdot\left(\int_0^{\delta t}\xi\|\partial_t^2 g(t^n+\xi)\|_X^2\,\mathrm{d}\xi\right)^{1/2}\\ &\le \frac{1}{\sqrt{2}}\left(\int_0^{\delta t}\xi\|\partial_t^2 g(t^n+\xi)\|_X^2\,\mathrm{d}\xi\right)^{1/2}.\end{aligned}$$

Result (4.3) is easily obtained taking the X-norm of (4.5), using Young's inequality and inserting the last bound for the residual norm. The two residual terms of (4.6) are exploited using similar arguments,

$$\frac{1}{\delta t^2}\|\int_{t^n}^{t^n+1}(t-t^n)^2\partial_t^3 g(t)\,\mathrm{d}t\|_X \leq \frac{1}{2}\left(\int_0^{\delta} t\xi\|\partial_t^3 g(t^n+\xi)\|_X^2\,\mathrm{d}\xi\right)^{1/2},$$

$$\begin{aligned}\frac{1}{\delta t^2}\|\int_{t^{n+1}}^{t^{n+2}}(t^{n+2}-t)^2\partial_t^3 g(t)\,\mathrm{d}t\|_X &\leq \frac{1}{2}\left(\int_0^{\delta} t(\delta t-\xi)\|\partial_t^3 g(t^{n+1}+\xi)\|_X^2\,\mathrm{d}\xi\right)^{1/2}\\ &\leq \frac{1}{2}\left(\int_{\delta} t^{2\delta t}\xi\|\partial_t^3 g(t^n+\xi)\|_X^2\,\mathrm{d}\xi\right)^{1/2}.\end{aligned}$$

Again, taking the X-norm of (4.6), using Young's inequality and the previous results we get (4.4).□

Lemma 4.3. *Let $\boldsymbol{u}, p$ be the solution of (4.2). Under the regularity Assumptions 4.1 and 4.2 for $m \leq 2$ the following bounds hold,*

$$\begin{aligned}\max_{0\leq n\leq N}\{\|D_t\boldsymbol{u}(t^{n+1})\|_0 + \lambda_{n+1}\|\nabla D_t p(t^{n+1})\|_0 + \lambda_{n+1}\|D_t\boldsymbol{u}(t^{n+1})\|_2\} &\leq C,\\ \max_{0\leq n\leq N}\lambda_{n+1}\|D_t\boldsymbol{u}(t^{n+1})\|_1^2 &\leq C,\end{aligned}$$

for all $\delta t > 0$. Further, if $\partial_t^3\boldsymbol{f}(t) \in L^\infty(0,T;\boldsymbol{L}^2(\Omega))$ we have

$$\max_{1\leq n\leq N}\{\lambda_{n+1}\|D_t^2\boldsymbol{u}(t^{n+1})\|_0 + \lambda_{n+1}\|D_t(\partial_t\boldsymbol{u}(t^{n+1}))\|_0\} \leq C.$$

Proof. The proof of this lemma is based on the inequalities for the continuous unknowns of lemma 4.1 and the expressions obtained in Lemma 4.2 using Taylor series. For the first term, we take $X = \boldsymbol{L}^2(\Omega)$ in (4.3). Multiplying (4.3) by λ_n and taking $X = \boldsymbol{H}^2(\Omega)$ and $X = \boldsymbol{L}^2(\Omega)$ we get the bounds over the H^2-norm of the discrete time derivative and the last term respectively, being $n \geq 1$. Again (4.3), multiplied by $\lambda_n^{1/2}$ leads to the H^1-norm of the discrete time derivative ($n \geq 1$). For the discrete second time derivative we use (4.4) with $X = \boldsymbol{L}^2(\Omega)$ multiplied by λ_{n-1} with $n \geq 2$. Finally, using (4.3) for $p(t^{n+1})$ multiplied by λ_n we get the pressure estimate for $n \geq 1$. For obtaining the remaining bounds at $n = 0$ (or $n = 1$ for the second time derivative) we use the inequalities in Lemma 4.1 and exploit the definition of λ.□

The assumed regularity assumptions for the continuous velocity and pressure and the previous lemma allow the obtention of the bounds for the convective term summarized in the next corollary.

Corollary 4.1. *Under the conditions of Lemma 4.3 the following bounds hold,*

$$\max_{0\leq n\leq N}\{\|\boldsymbol{u}(t^{n+1})\cdot\nabla\boldsymbol{u}(t^{n+1})\|_0 + \lambda_{n+1}\|D_t(\boldsymbol{u}(t^{n+1})\cdot\nabla\boldsymbol{u}(t^{n+1}))\|_0\} \leq C.$$

To define the order of approximation of time integration schemes for time-dependent partial differential equation, the following terminology is used: Given a Banach space X equipped with $\|\cdot\|_X$, a continuous function $f : [0,T] \longrightarrow X$ and two real numbers $0 < p < \infty$ and $\alpha > 0$, for a time step size $\delta t > 0$, let $t^n = n\delta t$ for $n = 0, ..., N = [T/\delta t] - 1$.

A sequence $\{f^n\}_{n=1}^N$ is said to be an α-th order approximation of f in $\ell^p(X)$ if there exists a constant C independent of δt such that

$$\left(\sum_{n=1}^{N} \delta t \|f(t^{n+1}) - f^{n+1}\|_X^p\right)^{1/p} < C\delta t^\alpha.$$

Moreover, $\{f^n\}_{n=1}^N$ is said to be an α-th order approximation of f in $\ell^\infty(X)$ if there exists a constant C indendent of δt and n such that

$$\|f(t^{n+1}) - f^{n+1}\|_X < C\delta t^\alpha \qquad \forall n = 0, ..., N.$$

For the stability result we will use the skew-symmetric (in its last two arguments) part of the convective form c in the analysis of the fully discrete problem.

4.3 Classical projection method

Let us recall again the classical projection method we will consider in the following section. As explained in Chapter 3, this procedure involves two basic steps. The first step consists of finding an intermediate velocity $\hat{\boldsymbol{u}}^{n+1}$ such that:

$$\frac{\hat{\boldsymbol{u}}^{n+1} - \boldsymbol{u}^n}{\delta t} - \nu\Delta\hat{\boldsymbol{u}}^{n+1} + \hat{\boldsymbol{u}}^{n+1} \cdot \nabla\hat{\boldsymbol{u}}^{n+1} = \boldsymbol{f}^{n+1}, \tag{4.7a}$$

$$\hat{\boldsymbol{u}}^{n+1}|_\Gamma = \boldsymbol{0}. \tag{4.7b}$$

The second step of the method consists of finding an end-of-step velocity $\boldsymbol{u}^{n+1}$ and a pressure p^{n+1} such that

$$\frac{\boldsymbol{u}^{n+1} - \hat{\boldsymbol{u}}^{n+1}}{\delta t} + \nabla p^{n+1} = \boldsymbol{0}, \tag{4.8a}$$

$$\nabla \cdot \boldsymbol{u}^{n+1} = 0, \tag{4.8b}$$

$$\boldsymbol{n} \cdot \boldsymbol{u}^{n+1}|_\Gamma = 0, \tag{4.8c}$$

which is equivalent to find the projection of $\hat{\boldsymbol{u}}^{n+1}$ onto the space $\boldsymbol{J}_0$ (see (1.16)),

$$\boldsymbol{u}^{n+1} = P_{\boldsymbol{J}_0}(\hat{\boldsymbol{u}}^{n+1}).$$

We point out that at this step only the normal component of the velocity is prescribed.

The main interest of the projection method is the possible uncoupling of the pressure from the velocity in its numerical approximation. It is achieved by taking the divergence of (4.8a), obtaining a Pressure Poisson Equation for the pressure,

$$\Delta p^{n+1} = \frac{1}{\delta t}\nabla \cdot \hat{\boldsymbol{u}}^{n+1}, \tag{4.9a}$$

$$\boldsymbol{n} \cdot \nabla p^{n+1}|_\Gamma = 0. \tag{4.9b}$$

Once the pressure p^{n+1} is calculated, the end-of-step velocity is recovered from (4.8a).

4.4 Analysis of a semi-discrete projection-like problem

In this section we obtain optimal convergence results and stability results for an auxiliary problem that can be identified as a semi-discrete projection method evaluating exactly the convective term. The strategy we follow is divided in three stages. First, some results for the semi-discrete Navier-Stokes equations are stated. Then a penalized semi-discrete system obtained from the previous one perturbing the continuity equation is studied, getting some results of convergence and stability. In both cases the non-linear term is calculated with the continuous solution (in time and space). Finally, we get the results for the auxiliary projection-like problem.

4.4.1 The semi-discrete system

We obtain some convergence and stability results for the semi-discrete version in time of (4.2) using a Backward Euler scheme and evaluating exactly the convective term. This methods consists of finding $\bar{\boldsymbol{u}}^{n+1} \in \boldsymbol{H}^1(\Omega)$ and $\bar{p}^{n+1} \in \boldsymbol{L}^2(\Omega)$ such that,

$$D_t\bar{\boldsymbol{u}}^{n+1} - \nu\Delta\bar{\boldsymbol{u}}^{n+1} + \boldsymbol{u}(t^{n+1})\cdot\nabla\boldsymbol{u}(t^{n+1}) + \nabla\bar{p}^{n+1} = \boldsymbol{f}(t^{n+1}), \tag{4.10a}$$

$$\nabla\cdot\bar{\boldsymbol{u}}^{n+1} = 0, \tag{4.10b}$$

$$\bar{\boldsymbol{u}}^{n+1}|_\Gamma = 0. \tag{4.10c}$$

with $\bar{\boldsymbol{u}}^0 = \boldsymbol{u}_0$. For simplicity, we have assumed $\boldsymbol{f}$ continuous in time. Let us define also p_0 as the solution of the problem

$$(\nabla p_0, \nabla q) = \nu(\Delta\boldsymbol{u}_0, \nabla q) - (\boldsymbol{u}_0\cdot\nabla\boldsymbol{u}_0, \nabla q) + (\boldsymbol{f}(t^0), \nabla q) \quad \forall q \in H^1(\Omega)/\mathbb{R}$$

and $\bar{p}^0 = p_0$. We summarize in the next lemmas some stability and convergence results for the solution of this semi-discrete problem.

Lemma 4.4 (Stability I)**.** *Let $\bar{\boldsymbol{u}}^{n+1}, \bar{p}^{n+1}$ be the solution of (4.10). Under Assumptions 4.1 and 4.2 the following bounds hold:*

$$\max_{0\le n\le N}\{\|D_t\bar{\boldsymbol{u}}^{n+1}\|_0 + \|\bar{\boldsymbol{u}}^{n+1}\|_2 + \|\nabla\bar{p}^{n+1}\|_0\} \le C,$$

for all $\delta t > 0$.

Proof. For obtaining stability results over the velocity and pressure increments, let us start applying the $D_t(\cdot)$ operator to system (4.10) for $n \ge 1$. Then we test the *incremental* momentum equation with $D_t\bar{\boldsymbol{u}}^{n+1}$. Doing that, we have:

$$\begin{aligned}&\frac{1}{2}(\|D_t\bar{\boldsymbol{u}}^{n+1}\|_0^2 - \|D_t\bar{\boldsymbol{u}}^{n}\|_0^2 + \delta t^2\|D_t^2\bar{\boldsymbol{u}}^{n+1}\|_0^2) + \nu\delta t\|D_t\bar{\boldsymbol{u}}^{n+1}\|_1^2\\ &\quad= (\delta t D_t\bar{\boldsymbol{u}}^{n+1}, D_t\boldsymbol{f}(t^{n+1})) - c(\boldsymbol{u}(t^{n+1}), \boldsymbol{u}(t^{n+1}), D_t\bar{\boldsymbol{u}}^{n+1}) + c(\boldsymbol{u}(t^{n}), \boldsymbol{u}(t^{n}), D_t\bar{\boldsymbol{u}}^{n+1}).\end{aligned}$$

We can easily bound the right hand side term related to the force vector,

$$(D_t\bar{\boldsymbol{u}}^{n+1}, D_t\boldsymbol{f}(t^{n+1})) \le \frac{\nu}{4}\|D_t\bar{\boldsymbol{u}}^{n+1}\|_1^2 + C\|D_t\boldsymbol{f}(t^{n+1})\|_{-1}^2.$$

For the treatment of the convective term we exploit the identity

$$\begin{aligned}&- c(\boldsymbol{u}(t^{n+1}), \boldsymbol{u}(t^{n+1}), D_t\bar{\boldsymbol{u}}^{n+1}) + c(\boldsymbol{u}(t^{n}), \boldsymbol{u}(t^{n}), D_t\bar{\boldsymbol{u}}^{n+1})\\ &\quad= -\delta t c(\boldsymbol{u}(t^{n+1}), D_t\boldsymbol{u}(t^{n+1}), D_t\bar{\boldsymbol{u}}^{n+1}) - \delta t c(D_t\boldsymbol{u}(t^{n+1}), \boldsymbol{u}(t^{n}), D_t\bar{\boldsymbol{u}}^{n+1}).\end{aligned} \tag{4.11}$$

We can obtain an upper bound of (4.11) using some continuity properties of the convective form and the regularity property (R1). We have

$$\begin{aligned} -c(\boldsymbol{u}(t^{n+1}), D_t\boldsymbol{u}(t^{n+1}), D_t\bar{\boldsymbol{u}}^{n+1}) &\leq \|\boldsymbol{u}(t^{n+1})\|_2 \|D_t\boldsymbol{u}(t^{n+1})\|_0 \|D_t\bar{\boldsymbol{u}}^{n+1}\|_1 \\ &\leq C\|D_t\boldsymbol{u}(t^{n+1})\|_0^2 + \frac{\nu}{4}\|D_t\bar{\boldsymbol{u}}^{n+1}\|_1^2, \\ -c(D_t\boldsymbol{u}(t^{n+1}), \boldsymbol{u}(t^n), D_t\bar{\boldsymbol{u}}^n) &\leq C\|D_t\boldsymbol{u}(t^{n+1})\|_0^2 + \frac{\nu}{4}\|D_t\bar{\boldsymbol{u}}^{n+1}\|_1^2. \end{aligned}$$

Invoking the bounds for the right hand side and absorbing some terms by the left hand side we finally get, adding from $n = 1$ to $n = N$,

$$\begin{aligned} &\max_{1\leq n\leq N} \|D_t\bar{\boldsymbol{u}}^{n+1}\|_0^2 + \sum_{n=1}^{N} \frac{\nu\delta t}{4}\|D_t\bar{\boldsymbol{u}}^{n+1}\|_1^2 \\ &\qquad \leq C\sum_{n=1}^{N} \delta t\|D_t\boldsymbol{f}(t^{n+1})\|_{-1}^2 + C\sum_{n=1}^{N}\delta t\|D_t\boldsymbol{u}(t^{n+1})\|_0^2 + \|D_t\bar{\boldsymbol{u}}^1\|_0^2. \quad (4.12) \end{aligned}$$

The bound for the force vector is easily obtained from (4.3) and the assumed regularity properties. For the discrete time derivative of the continuous velocity we use the results of Lemma 4.3.

The discrete time derivative at the first time step is easily obtained from (4.10a) in the form:

$$D_t\bar{\boldsymbol{u}}^1 - \nu\Delta(\bar{\boldsymbol{u}}^1 - \bar{\boldsymbol{u}}^0) + \nabla\bar{p}^1 = \boldsymbol{f}(t^1) - \boldsymbol{u}(t^1)\cdot\nabla\boldsymbol{u}(t^1) + \nu\Delta\bar{\boldsymbol{u}}^0.$$

Then, testing this equation with $D_t\bar{\boldsymbol{u}}^1$ and using Young's inequality for the right hand side we arrive at

$$\begin{aligned} \|D_t\bar{\boldsymbol{u}}^1\|_0^2 + \nu\delta t\|D_t\bar{\boldsymbol{u}}^1\|_1^2 &\leq \frac{1}{2}\|D_t\bar{\boldsymbol{u}}^1\|_0^2 + C\|\boldsymbol{f}(t^1)\|_0^2 + C\|\boldsymbol{u}(t^1)\|_1^2\|\boldsymbol{u}(t^1)\|_2^2 + C\nu^2\|\boldsymbol{u}_0\|_2^2 \\ &\leq C, \quad (4.13) \end{aligned}$$

where we have used the regularity properties for the data and the continuous velocity and the fact that the initial velocity is weakly divergence-free (belongs to $\boldsymbol{J}_1$). Combining (4.12) and (4.13) we have

$$\max_{0\leq n\leq N} \|D_t\bar{\boldsymbol{u}}^{n+1}\|_0^2 + \sum_{n=0}^{N}\nu\delta t\|D_t\bar{\boldsymbol{u}}^{n+1}\|_1^2 \leq C,$$

proving the first result of the lemma. The remaining regularity results over $\bar{\boldsymbol{u}}^{n+1}$ and $\bar{p}^{n+1}$ can be easily obtained from the estimate (1.29) for the Stokes problem, the regularity assumption (R1) for the continuous velocity, the bound for the convective term of Corollary 4.1 and the stability result for the discrete time derivative already proved,

$$\|\bar{\boldsymbol{u}}^{n+1}\|_2 + \|\nabla\bar{p}^{n+1}\|_0 \leq \|D_t\bar{\boldsymbol{u}}^{n+1}\|_0 + \|\boldsymbol{f}(t^{n+1})\|_0 + \|\boldsymbol{u}(t^{n+1})\cdot\nabla\boldsymbol{u}(t^{n+1})\|_0 \leq C.$$

□

Lemma 4.5 (Stability II)**.** *Let $\bar{\boldsymbol{u}}^{n+1}, \bar{p}^{n+1}$ be the solution of (4.10). Under the assumptions of Lemma 4.4 and $\partial_t^3\boldsymbol{f}(t) \in L^\infty(0,T;\boldsymbol{L}^2(\Omega))$, the following bounds hold:*

$$\max_{1\leq n\leq N} \lambda_{n+1}\{\|D_t^2\bar{\boldsymbol{u}}^{n+1}\|_0 + \|D_t\bar{\boldsymbol{u}}^{n+1}\|_2 + \|\nabla D_t\bar{p}^{n+1}\|_0\} \leq C,$$

for all $\delta t > 0$.

Proof. Applying the discrete operator $D_t^2(\cdot)$ over (4.10) and taking $D_t^2\bar{\boldsymbol{u}}^{n+1}$ as a test function in the *momentum* equation we arrive to

$$\|D_t^2\bar{\boldsymbol{u}}^{n+1}\|_0^2 - \|D_t^2\bar{\boldsymbol{u}}^{n}\|_0^2 + \nu\delta t\|D_t^2\bar{\boldsymbol{u}}^{n+1}\|_1^2 = \delta t(D_t^2\bar{\boldsymbol{u}}^{n+1}, D_t^2\boldsymbol{f}(t^{n+1})) - \delta t\mathcal{NLT}$$

for $n \geq 2$, where $\mathcal{NLT}$ groups the convective terms,

$$\begin{aligned}\mathcal{NLT} =& c(D_t\boldsymbol{u}(t^{n+1}), D_t\boldsymbol{u}(t^{n+1}), D_t^2\bar{\boldsymbol{u}}^{n+1}) + c(D_t\boldsymbol{u}(t^{n}), D_t\boldsymbol{u}(t^{n}), D_t^2\bar{\boldsymbol{u}}^{n+1})\\ &+ c(\boldsymbol{u}(t^{n}), D_t^2\boldsymbol{u}(t^{n+1}), D_t^2\bar{\boldsymbol{u}}^{n+1}) + c(D_t^2\boldsymbol{u}(t^{n+1}), \boldsymbol{u}(t^{n}), D_t^2\bar{\boldsymbol{u}}^{n+1}).\end{aligned}$$

Using some continuity properties of the convective form and (R1) we can work over the right hand side, obtaining

$$\begin{aligned}\|D_t^2\bar{\boldsymbol{u}}^{n+1}\|_0^2 - \|D_t^2\bar{\boldsymbol{u}}^{n}\|_0^2 + \nu\delta t\|D_t^2\bar{\boldsymbol{u}}^{n+1}\|_1^2 \leq& C\delta t\|D_t^2\boldsymbol{f}\|_{-1}^2 + C\delta t\|D_t^2\boldsymbol{u}(t^{n+1})\|_0^2\\ &+ C\delta t\|D_t\boldsymbol{u}(t^{n+1})\|_1^4 + C\delta t\|D_t\boldsymbol{u}(t^{n})\|_1^4.\end{aligned}$$

Multiplying this inequality by λ_{n+1}^2, invoking the bounds of the Lemma 4.3 and adding up from $n = 2$ to $n = N$ we get

$$\begin{aligned}\lambda_{N+1}^2\|D_t^2\bar{\boldsymbol{u}}^{N+1}\|_0^2 \leq& C + \lambda_2^2\|D_t^2\bar{\boldsymbol{u}}^{2}\|_0^2\\ \leq& C + \frac{\lambda_2^2}{\delta t^2}(\|D_t\bar{\boldsymbol{u}}^{1}\|_0^2 + \|D_t\bar{\boldsymbol{u}}^{0}\|_0^2) \leq C.\end{aligned} \tag{4.14}$$

In the last inequality we have used the results of Lemma 4.4 and exploited the definition of the weighting function λ. Applying now D_t over (4.10) and multiplying by λ_{n+1} both equations we get the following Stokes-like problem,

$$\begin{aligned}-\lambda_{n+1}\Delta D_t\bar{\boldsymbol{u}}^{n+1} + \lambda_{n+1}\nabla D_t\bar{p}^{n+1} =& \lambda_{n+1}D_t\boldsymbol{f}(t^{n+1}) - \lambda_{n+1}D_t(\boldsymbol{u}(t^{n+1})\cdot\nabla(\boldsymbol{u}(t^{n+1})))\\ &- \lambda_{n+1}(D_t\boldsymbol{u}(t^{n+1}))\cdot\nabla\boldsymbol{u}(t^{n}) - \lambda_{n+1}D_t^2\bar{\boldsymbol{u}}^{n+1},\\ \lambda_{n+1}\nabla\cdot D_t\bar{\boldsymbol{u}}^{n+1} =& 0,\end{aligned}$$

for $n \geq 1$. Due to previous stability estimates we know that the force term belongs to $L^\infty(0,T;\boldsymbol{L}^2(\Omega))$. Then we can use the inequality (1.29) with $s = 2$, yielding

$$\lambda_{n+1}\|D_t\bar{\boldsymbol{u}}^{n+1}\|_2 + \lambda_{n+1}\|\nabla D_t\bar{p}^{n+1}\|_0 \leq C.$$

For $n = 1$ the bound is easily obtained making use of the stability results of Lemma 4.4, the definition of the weighting function and the regularity of p_0. The proof is finished.□

Lemma 4.6 (Convergence)**.** *Let $\bar{\boldsymbol{u}}^{n+1}, \bar{p}^{n+1}$ be the solution of (4.10). Under assumptions 4.1, 4.2 the following bounds hold,*

$$\begin{aligned}\max_{0\leq n\leq N}\{&\|\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}^{n+1}\|_0 + \sqrt{\delta t}\|\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}^{n+1}\|_1\\ &+ \sqrt{\lambda_{n+1}}\sqrt{\delta t}\|p(t^{n+1}) - \bar{p}^{n+1}\|_0\} \leq C\delta t,\end{aligned}$$

for all $\delta t > 0$.

For the proof of these convergence results we refer to [97].

4.4.2 The penalized semi-discrete system

Now we introduce the perturbed problem to analyze, that consists of: find $\boldsymbol{u}_{\delta t}^{n+1} \in \boldsymbol{H}^1(\Omega)$ and $p_{\delta t}^{n+1} \in H^1(\Omega)/\mathbb{R}$ such that

$$D_t \boldsymbol{u}_{\delta t}^{n+1} - \nu \Delta \boldsymbol{u}_{\delta t}^{n+1} + \boldsymbol{u}(t^{n+1}) \cdot \nabla \boldsymbol{u}(t^{n+1}) + \nabla p_{\delta t}^{n+1} = \boldsymbol{f}(t^{n+1}), \tag{4.15a}$$

$$\nabla \cdot \boldsymbol{u}_{\delta t}^{n+1} - \delta t \Delta p_{\delta t}^{n+1} = 0, \tag{4.15b}$$

$$\boldsymbol{u}_{\delta t}^{n+1}|_\Gamma = \boldsymbol{0}. \tag{4.15c}$$

with $\boldsymbol{u}_{\delta t}^0 = \boldsymbol{u}_0$ and $p_{\delta t}^0 = p_0$.We start the characterization of (4.15) with some stability results.

Lemma 4.7 (Stability)**.** *Let $\boldsymbol{u}_{\delta t}^{n+1}, p_{\delta t}^{n+1}$ be the solution of (4.15). Under Assumptions 4.1 and 4.2 the following bounds hold:*

$$\max_{0 \le n \le N} \{\|D_t \boldsymbol{u}_{\delta t}^{n+1}\|_0^2\} + \sum_{n=0}^{N} \delta t^2 \|\nabla D_t p_{\delta t}^{n+1}\|_0^2 \le C,$$

$$\max_{0 \le n \le N} \{\|\boldsymbol{u}_{\delta t}^{n+1}\|_2^2 + \|\nabla p_{\delta t}^{n+1}\|_0^2\} \le C.$$

for all $\delta t > 0$.

Proof. The proof of this lemma is very similar to that of Lemma 4.4. We apply the $D_t(\cdot)$ operator over the penalyzed system (4.15). Then we test the *incremental* momentum equation with $D_t \bar{\boldsymbol{u}}^{n+1}$ and the *incremental* continuity equation with $\delta t D_t \bar{p}^{n+1}$ taking into account that $\boldsymbol{u}(t^{n+1}) \in \boldsymbol{J}_1$. Doing that, we have:

$$\begin{aligned}
&\frac{1}{2}(\|D_t \boldsymbol{u}_{\delta t}^{n+1}\|_0^2 - \|D_t \boldsymbol{u}_{\delta t}^{n}\|_0^2 + \delta t^2 \|D_t^2 \boldsymbol{u}_{\delta t}^{n+1}\|_0^2) + \nu \delta t \|D_t \boldsymbol{u}_{\delta t}^{n+1}\|_1^2 + \delta t^2 \|\nabla D_t p_{\delta t}^{n+1}\|_0^2 \\
&\quad = (\delta t D_t \boldsymbol{u}_{\delta t}^{n+1}, D_t \boldsymbol{f}(t^{n+1})) - c(\boldsymbol{u}(t^{n+1}), \boldsymbol{u}(t^{n+1}), D_t \boldsymbol{u}_{\delta t}^{n+1}) + c(\boldsymbol{u}(t^{n}), \boldsymbol{u}(t^{n}), D_t \boldsymbol{u}_{\delta t}^{n+1}).
\end{aligned}$$

We bound the right hand side as that in Lemma 4.4. Again, the assumptions over the continuous velocity and the force vector lead to the first stability result of the lemma:

$$\max_{1 \le n \le N} \{\|D_t \boldsymbol{u}_{\delta t}^{n+1}\|_0^2\} + \sum_{n=1}^{N} \delta t^2 \|\nabla D_t p_{\delta t}^{n+1}\|_0^2 \le C + \|D_t \boldsymbol{u}_{\delta t}^{1}\|_0^2.$$

As in Lemma 4.4, we bound the right hand side testing the system (4.15) at $n = 0$ with $D_t \boldsymbol{u}_{\delta t}^1$ for the momentum equation and $p_{\delta t}^1$ for the continuity equation, getting

$$\|D_t \boldsymbol{u}_{\delta t}^{1}\|_0^2 + \|\nabla p_{\delta t}^{1}\|_0^2 + \|\nabla p_0\|_0^2 \le C.$$

The first part of the lemma is proved. Again, due to the regularity assumption of the Stokes operator, the previous stability result and (R1) the proof is easily completed.□

The aim of this section is to obtain optimal convergence results for the penalized system (4.15). Let us introduce the error functions

$$\begin{aligned}
\boldsymbol{e}_{\delta t}^{n+1} &= \bar{\boldsymbol{u}}^{n+1} - \boldsymbol{u}_{\delta t}^{n+1}, \\
q_{\delta t}^{n+1} &= \bar{p}^{n+1} - p_{\delta t}^{n+1},
\end{aligned}$$

that evaluate the difference between the solutions of systems (4.10) and (4.15). Subtracting these systems we get the problem that governs the error functions:

$$D_t \boldsymbol{e}_{\delta t}^{n+1} - \nu \Delta \boldsymbol{e}_{\delta t}^{n+1} + \nabla q_{\delta t}^{n+1} = \boldsymbol{0}, \tag{4.16a}$$

$$\nabla \cdot \boldsymbol{e}_{\delta t}^{n+1} - \delta t \Delta q_{\delta t}^{n+1} = -\delta t \Delta \bar{p}^{n+1}. \tag{4.16b}$$

For obtaining error estimates for $\boldsymbol{e}_{\delta t}^{n+1}$ and $q_{\delta t}^{n+1}$ we have decomposed them in two parts:

$$\boldsymbol{e}_{\delta t}^{n+1} = \hat{\boldsymbol{e}}^{n+1} + \boldsymbol{E}_{\delta t}^{n+1},$$
$$q_{\delta t}^{n+1} = \hat{q}^{n+1} + \Pi_{\delta t}^{n+1},$$

using the strategy proposed by Prohl in [136]. We define the new error functions as follows:

$$\hat{\boldsymbol{e}}^{n+1} = \bar{\boldsymbol{u}}^{n+1} - \boldsymbol{U}_{\delta t}^{n+1},$$
$$\hat{q}^{n+1} = \bar{p}^{n+1} - P_{\delta t}^{n+1},$$
$$\boldsymbol{E}_{\delta t}^{n+1} = \boldsymbol{U}_{\delta t}^{n+1} - \boldsymbol{u}_{\delta t}^{n+1},$$
$$\Pi_{\delta t}^{n+1} = P_{\delta t}^{n+1} - p_{\delta t}^{n+1},$$

being $\boldsymbol{U}_{\delta t}^{n+1}$ and $P_{\delta t}^{n+1}$ defined by this auxiliary problem:

$$D_t \bar{\boldsymbol{u}}^{n+1} - \nu \Delta \boldsymbol{U}_{\delta t}^{n+1} + \boldsymbol{u}(t^{n+1}) \cdot \nabla \boldsymbol{u}(t^{n+1}) + \nabla P_{\delta t}^{n+1} = \boldsymbol{f}(t^{n+1}), \tag{4.17a}$$

$$\nabla \cdot \boldsymbol{U}_{\delta t}^{n+1} - \delta t \Delta P_{\delta t}^{n+1} = 0, \tag{4.17b}$$

$$\boldsymbol{U}_{\delta t}^{n+1}|_{\Gamma} = \boldsymbol{0}. \tag{4.17c}$$

Thus, the problem that governs the pair $\{\hat{\boldsymbol{e}}^{n+1}, \hat{q}^{n+1}\}$ is obtained subtracting (4.17) from (4.10),

$$-\nu \Delta \hat{\boldsymbol{e}}^{n+1} + \nabla \hat{q}^{n+1} = \boldsymbol{0},$$
$$\nabla \cdot \hat{\boldsymbol{e}}^{n+1} - \delta t \Delta \hat{q}^{n+1} = -\delta t \Delta \bar{p}^{n+1}.$$

We note that this system is well-defined for $n = -1$. Some convergence results related to this scheme are stated in the next lemma. For the proof of the first two results we refer to [140]. The obtention of the third one is identical to that of the first one but working over the *incremental* system obtained applying D_t over (4.16a).

Lemma 4.8 (Intermediate Convergence I)**.** *Under Assumption 4.2 the following error estimates hold:*

$$\|\hat{\boldsymbol{e}}^{n+1}\|_0 + \|\hat{q}^{n+1}\|_{-1} \leq C\delta t \|\nabla \bar{p}^{n+1}\|_0,$$
$$\|\hat{\boldsymbol{e}}^{n+1}\|_1 + \|\hat{q}^{n+1}\|_0 \leq C\sqrt{\delta t} \|\nabla \bar{p}^{n+1}\|_0,$$
$$\|D_t \hat{\boldsymbol{e}}^{n+1}\|_0 \leq C\delta t \|\nabla D_t \bar{p}^{n+1}\|_0,$$

for all $\delta t > 0$.

The last stage is to compare the penalized problem with the auxiliary problem (4.17). The difference between these systems leads to:

$$D_t \boldsymbol{E}_{\delta t}^{n+1} - \nu \Delta \boldsymbol{E}_{\delta t}^{n+1} + \nabla \Pi_{\delta t}^{n+1} = D_t \hat{\boldsymbol{e}}^{n+1}, \tag{4.18a}$$

$$\nabla \cdot \boldsymbol{E}_{\delta t}^{n+1} - \delta t \Delta \Pi_{\delta t}^{n+1} = 0, \tag{4.18b}$$

$$\boldsymbol{E}_{\delta t}^{n+1}|_{\Gamma} = 0, \tag{4.18c}$$

defining $\boldsymbol{E}_{\delta t}^{0} = \hat{\boldsymbol{e}}^{0}$ and $\Pi_{\delta t}^{0} = \hat{q}^{0}$.

Lemma 4.9 (Intermediate Convergence II)**.** *Under Assumptions 4.1,4.2, and* $\partial_t^3 \boldsymbol{f}(t) \in L^\infty(0,T;\boldsymbol{L}^2(\Omega))$ *the following error estimates hold:*

$$\max_{0\leq n\leq N}\{\|\boldsymbol{E}_{\delta t}^{n+1}\|_0+\sqrt{\delta t}\|\boldsymbol{E}_{\delta t}^{n+1}\|_1+\lambda_{n+1}\sqrt{\delta t}\|\Pi_{\delta t}^{n+1}\|_0\}\leq C\delta t,$$

for all $\delta t>0$.

Proof. We take as a test function $\delta t\boldsymbol{E}_{\delta t}^{n+1}$ in (4.18a) and $\delta t\Pi_{\delta t}^{n+1}$ in (4.18b) and arrive to the following inequality:

$$\begin{aligned}\frac{1}{2}\left(\|\boldsymbol{E}_{\delta t}^{n+1}\|_0^2-\|\boldsymbol{E}_{\delta t}^{n}\|_0^2+\delta t\|D_t\boldsymbol{E}_{\delta t}^{n+1}\|_0^2\right)+\nu\delta t\|\boldsymbol{E}_{\delta t}^{n+1}\|_1^2+\delta t^2\|\nabla\Pi_{\delta t}^{n+1}\|_0^2\\ =\delta t(D_t\hat{\boldsymbol{e}}^{n+1},\boldsymbol{E}_{\delta t}^{n+1}).\end{aligned}\tag{4.19}$$

We multiply (4.19) by the weighting function λ_{n+1} and exploit the Cauchy-Schwarz inequality in order to arrive to

$$\begin{aligned}\lambda_{n+1}\|\boldsymbol{E}_{\delta t}^{n+1}\|_0^2-\lambda_{n+1}\|\boldsymbol{E}_{\delta t}^{n}\|_0^2+\lambda_{n+1}\nu\delta t\|\boldsymbol{E}_{\delta t}^{n+1}\|_1^2+\lambda_{n+1}\delta t^2\|\nabla\Pi_{\delta t}^{n+1}\|_0^2\\ \leq C\delta t^3\lambda_{n+1}^2\|\nabla D_t\bar{p}^{n+1}\|_0^2+C\delta t\|\boldsymbol{E}_{\delta t}^{n+1}\|_0^2,\end{aligned}\tag{4.20}$$

where we have used the last bound of Lemma 4.8. Adding (4.20) from $n=0$ to an arbitrary time step $n=M$ and using the stability results for $\bar{p}^{n+1}$ from Lemma 4.5, we arrive to:

$$\lambda_{M+1}\|\boldsymbol{E}_{\delta t}^{M+1}\|_0^2+\sum_{n=0}^{M}\lambda_{M+1}\nu\delta t\|\boldsymbol{E}_{\delta t}^{n+1}\|_1^2\leq C\delta t^2t^{M+1}+C\sum_{n=0}^{M}\delta t\|\boldsymbol{E}_{\delta t}^{n+1}\|_0^2.\tag{4.21}$$

The bound for the last term in the right hand side has been extracted from [136]. We include here the proof. Let us introduce the following evolutionary backward auxiliary problem: find $\boldsymbol{\alpha}_m\in\boldsymbol{H}_0^1(\Omega)\cap\boldsymbol{H}_2(\Omega)$ and $\beta_m\in H^1(\Omega)/\mathbb{R}$ such that,

$$D_t\boldsymbol{\alpha}_{m+1}+\nu\Delta\boldsymbol{\alpha}_m-\nabla\beta_m=\boldsymbol{e}_{\delta t}^{m+1},\tag{4.22a}$$

$$\nabla\cdot\boldsymbol{\alpha}_m=0,\tag{4.22b}$$

$$\boldsymbol{\alpha}_{M+1}=\boldsymbol{0},\tag{4.22c}$$

for the interval $0\leq m\leq M$, where M is an arbitrary time step such that $0\leq M\leq N$. We test the first equation by $\boldsymbol{e}_{\delta t}^{n+1}$ and get

$$\|\boldsymbol{e}_{\delta t}^{m+1}\|_0^2=(D_t\boldsymbol{\alpha}_{m+1},\boldsymbol{e}_{\delta t}^{m+1})-\nu(\nabla\boldsymbol{\alpha}_m,\nabla\boldsymbol{e}_{\delta t}^{m+1})-(\nabla\beta_m,\boldsymbol{e}_{\delta t}^{m+1}).\tag{4.23}$$

Furthermore, testing equation (4.16a) by $\boldsymbol{\alpha}_m$ we get

$$(D_t\boldsymbol{e}_{\delta t}^{m+1},\boldsymbol{\alpha}_m)+\nu(\nabla\boldsymbol{\alpha}_m,\nabla\boldsymbol{e}_{\delta t}^{m+1})=0.\tag{4.24}$$

Adding equalities (4.23) and (4.24) and invoking the identity

$$(D_t\boldsymbol{e}_{\delta t}^{m+1},\boldsymbol{\alpha}_m)+(D_t\boldsymbol{\alpha}_{m+1},\boldsymbol{e}_{\delta t}^{m+1})=D_t(\boldsymbol{e}_{\delta t}^{m+1},\boldsymbol{\alpha}_{m+1}),$$

we arrive to the equality

$$\|\boldsymbol{e}_{\delta t}^{m+1}\|_0^2=D_t(\boldsymbol{e}_{\delta t}^{m+1},\boldsymbol{\alpha}_{m+1})-(\nabla\beta_m,\boldsymbol{e}_{\delta t}^{m+1}).\tag{4.25}$$

We test (4.16b) with β_m and add the resulting equation to (4.25). Then, adding for all the time steps and taking into account that $\boldsymbol{\alpha}_{M+1} = \mathbf{0}$ we get

$$\sum_{m=0}^{M} \delta t \|\boldsymbol{e}_{\delta t}^{m+1}\|_0^2 = \sum_{m=0}^{M} \delta t (\nabla \beta_m, \boldsymbol{e}_{\delta t}^{m+1}). \tag{4.26}$$

At this point we recall an estimate for the non-stationary Stokes problem (4.22). We test (4.22a) with $\delta t D_t \boldsymbol{\alpha}_{m+1}$ and sum over the whole range $0 \leq m \leq M$, arriving to

$$\sum_{m=0}^{M} \delta t \|D_t \boldsymbol{\alpha}_{m+1}\|_0^2 \leq C \sum_{m=0}^{M} \delta t \|\boldsymbol{e}_{\delta t}^{m+1}\|_0^2 + C\|\boldsymbol{\alpha}_{M+1}\|_1^2 = C \sum_{m=0}^{M} \delta t \|\boldsymbol{e}_{\delta t}^{m+1}\|_0^2.$$

Thus, estimate (1.29) for the Stokes problem (4.22) allows the obtention of the inequality

$$\sum_{m=0}^{M} \delta t \|\nabla \beta_m\|_0^2 + \sum_{m=0}^{M} \delta t \|\boldsymbol{\alpha}_{m+1}\|_2^2 \leq \sum_{m=0}^{M} \delta t \|\boldsymbol{e}_{\delta t}^{m+1}\|_0^2. \tag{4.27}$$

On the other hand, multiplying (4.16a) by β_m we get

$$(\nabla \beta_m, \boldsymbol{e}_{\delta t}^{m+1}) = \delta t (\nabla \beta_m, \nabla q_{\delta t}^{m+1}) - \delta t (\nabla \beta_m, \nabla \bar{p}^{m+1}). \tag{4.28}$$

Finally, we exploit the right hand side of (4.26) using (4.28) and Young's inequality (with appropriate constants in order to absorb the term related to β_{m+1} by the left hand side making use of (4.27)). Then, we arrive to

$$\begin{aligned}\sum_{m=0}^{M} \delta t \|\boldsymbol{e}_{\delta t}^{m+1}\|_0^2 &\leq C \sum_{m=0}^{M} \delta t^3 (\|\nabla q_{\delta t}^{m+1}\|_0^2 + \|\nabla \bar{p}^{m+1}\|_0^2) \\ &\leq C \sum_{m=0}^{M} \delta t^3 (\|\nabla \bar{p}^{m+1}\|_0^2 + \|\nabla p_{\delta t}^{m+1}\|_0^2) \leq C t^{M+1} \delta t^2,\end{aligned}$$

due to the stability result of the pressure terms extracted from Lemmas 4.4 and 4.7. On the other hand, we know from Lemma 4.8 that

$$\sum_{m=0}^{M} \delta t \|\hat{\boldsymbol{e}}^{m+1}\|_0^2 \leq C t^{M+1} \delta t^2.$$

If we insert in (4.21) these results we get the first error estimate for the velocity. In order to verify the second convergence result we test (4.18a) with $D_t \boldsymbol{E}_{\delta t}^{n+1}$ and (4.18b) with $\Pi_{\delta t}^{n+1}$. Summing the two resulting equations and finally summing over all time steps we get,

$$\begin{aligned}\sum_{n=0}^{M} \delta t \|D_t \boldsymbol{E}_{\delta t}^{n+1}\|_0^2 + \frac{\nu}{2} \|\boldsymbol{E}_{\delta t}^{M+1}\|_1^2 + \sum_{n=0}^{M} \delta t^2 \|\nabla \Pi_{\delta t}^{n+1}\|_0^2 \\ \leq \frac{1}{2} \sum_{n=0}^{N} \delta t \|D_t \boldsymbol{E}_{\delta t}^{n+1}\|_0^2 + \frac{1}{2} \sum_{n=0}^{N} \delta t \|D_t \hat{\boldsymbol{e}}^{n+1}\|_0^2.\end{aligned}$$

Invoking the last bound of Lemma 4.8 and the stability result for $\bar{p}^{n+1}$ in Lemma 4.5 we obtain,

$$\sum_{n=0}^{N}\delta t\|D_t\hat{\boldsymbol{e}}^{n+1}\|_0^2 \leq C\sum_{n=0}^{N}\delta t^3\|\nabla D_t\bar{p}^{n+1}\|_0^2$$
$$\leq C\delta t\sum_{n=0}^{L}\frac{\lambda_{n+1}^2}{(n+1)^2}\|\nabla D_t\bar{p}^{n+1}\|_0^2 + C\delta t\sum_{n=L+1}^{N}\delta t^2\|\nabla D_t\bar{p}^{n+1}\|_0^2 \leq C\delta t,$$

assuming that $L < N$ is the first time step for which $\lambda_{L+1} = 1$. For $L \geq N$ the bound also holds. Thus, the second convergence result has been obtained. For the obtention of the pressure estimate we use the classical stability property of the divergence operator,

$$\|\cdot\|_0 \leq C\sup_{\boldsymbol{v}\in\boldsymbol{H}_0^1}\frac{(\nabla\cdot\boldsymbol{v},\cdot)}{\|\boldsymbol{v}\|_1} \tag{4.29}$$

that combined with (4.16a) leads to

$$\|q_{\delta t}^{n+1}\|_0 \leq C\|D_t\boldsymbol{e}_{\delta t}^{n+1}\|_{-1} + C\|\boldsymbol{e}_{\delta t}^{n+1}\|_1 \leq C\|D_t\boldsymbol{E}_{\delta t}^{n+1}\|_{-1} + C\sqrt{\delta t}. \tag{4.30}$$

We apply the operator $D_t(\cdot)$ over system (4.16a), multiply the first equation by the test function $\lambda_{n+1}^2 D_t\boldsymbol{e}_{\delta t}^{n+1}$ and invoke the second equation of the *incremental* system in order to arrive to

$$\lambda_{n+1}^2\|D_t\boldsymbol{e}_{\delta t}^{N+1}\|_0^2 \leq C\sum_{n=0}^{N}\lambda_{n+1}^2\delta t^2\|\nabla D_t\bar{p}^{n+1}\|_0^2 \leq C\delta t. \tag{4.31}$$

The convergence result for the pressure is straightforward from (4.30) and (4.31). □

The previous lemmas lead to a corollary with the convergence results we need for the velocity of the penalized system (4.15).

Corollary 4.2 (Convergence)**.** *Let $\boldsymbol{u}_{\delta t}^{n+1}, p_{\delta t}^{n+1}$ be the solution of (4.15). Under the assumptions of Lemma 4.9, the following error estimates hold:*

$$\max_{0\leq n\leq N}\{\|\boldsymbol{u}(t^{n+1}) - \boldsymbol{u}_{\delta t}^{n+1}\|_0 + \sqrt{\delta t}\|\boldsymbol{u}(t^{n+1}) - \boldsymbol{u}_{\delta t}^{n+1}\|_1$$
$$+ \lambda_{n+1}\sqrt{\delta t}\|p(t^{n+1}) - p_{\delta t}^{n+1}\|_0\} \leq C\delta t,$$

for all $\delta t > 0$.

4.4.3 The semi-discrete projection-like system

Let us start introducing the auxiliary problem at the continuous level which will help us to obtain error estimates for the fully discrete first order fractional step method. It consists of: find $\hat{\boldsymbol{w}}^{n+1} \in \boldsymbol{H}_0^1(\Omega)$, $\boldsymbol{w}^{n+1} \in \boldsymbol{H}(\mathrm{div},\Omega)$ and $\phi^{n+1} \in H_0^1(\Omega)/\mathbb{R}$ such that,

$$\frac{1}{\delta t}(\hat{\boldsymbol{w}}^{n+1} - \boldsymbol{w}^n) - \nu\Delta\hat{\boldsymbol{w}}^{n+1} + \boldsymbol{u}(t^{n+1})\cdot\nabla\boldsymbol{u}(t^{n+1}) = \boldsymbol{f}(t^{n+1}), \tag{4.32a}$$
$$\frac{1}{\delta t}(\boldsymbol{w}^{n+1} - \hat{\boldsymbol{w}}^{n+1}) + \nabla\phi^{n+1} = \boldsymbol{0}, \tag{4.32b}$$
$$\nabla\cdot\boldsymbol{w}^{n+1} = 0, \tag{4.32c}$$

with boundary conditions

$$\hat{\boldsymbol{w}}^{n+1}|_{\Gamma} = \boldsymbol{0}, \tag{4.32d}$$

$$\boldsymbol{n} \cdot \boldsymbol{w}^{n+1}|_{\Gamma} = 0. \tag{4.32e}$$

This method can be interpreted as a particular version of (4.7)-(4.8) evaluating exactly the nonlinear term. Furthermore, adding (4.32a) to (4.32b) and taking the divergence of (4.32b) we obtain an alternative version of this scheme,

$$D_t\hat{\boldsymbol{w}}^{n+1} - \nu\Delta\hat{\boldsymbol{w}}^{n+1} + \nabla\phi^n = \boldsymbol{f}(t^{n+1}) - \boldsymbol{u}(t^{n+1}) \cdot \nabla\boldsymbol{u}(t^{n+1}), \tag{4.33a}$$

$$\nabla \cdot \hat{\boldsymbol{w}}^{n+1} - \delta t\Delta\phi^{n+1} = 0, \tag{4.33b}$$

where $\boldsymbol{w}^{n+1}$ has disappeared. This method can be identified as the penalized problem (4.15) with a lag in the evaluation of the pressure in the momentum equation.

Once convergence and stability results for the penalized problem (4.15) have been obtained, let us compare this with the projection-like system in the format (4.33) for convenience. The only term that introduces a difference between the solution of these two systems is the pressure lag in the momentum equation. We collect the convergence results we have obtained for the auxiliary projection-like problem in the following lemma.

Lemma 4.10 (Convergence)**.** *Let $\{\hat{\boldsymbol{w}}^{n+1}, \phi^{n+1}\}$ be the solution of (4.33). Further, let $\{\boldsymbol{u}_{\delta t}^{n+1}, p_{\delta t}^{n+1}\}$ be the solution of (4.15). Under the assumptions of Lemma 4.7, the following error estimates are satisfied:*

$$\max_{0\le n\le N}\{\|\boldsymbol{u}_{\delta t}^{n+1} - \hat{\boldsymbol{w}}^{n+1}\|_0\} \le C\delta t,$$

$$\max_{0\le n\le N}\{\|\boldsymbol{u}_{\delta t}^{n+1} - \hat{\boldsymbol{w}}^{n+1}\|_1 + \lambda_{n+1}\|p_{\delta t}^{n+1} - \phi^{n+1}\|_0\} \le C\sqrt{\delta t},$$

for $\delta t > 0$ small enough.

Proof. Let us define the difference between the velocities and pressures of (4.15) and (4.33),

$$\boldsymbol{e}_{PR}^{n+1} = \boldsymbol{u}_{\delta t}^{n+1} - \hat{\boldsymbol{w}}^{n+1},$$
$$q_{PR}^{n+1} = p_{\delta t}^{n+1} - \phi^{n+1}.$$

Then after subtracting these schemes we obtain,

$$D_t\boldsymbol{e}_{PR}^{n+1} - \nu\Delta\boldsymbol{e}_{PR}^{n+1} + \nabla q_{PR}^n = -\delta t\nabla(D_t p_{\delta t}^{n+1}), \tag{4.34a}$$

$$\nabla \cdot \boldsymbol{e}_{PR}^{n+1} - \delta t\Delta q_{PR}^{n+1} = 0, \tag{4.34b}$$

with $\boldsymbol{e}_{PR}^0 = 0$ and $q_{PR}^0 = 0$. After testing (4.34a) with $\delta t\boldsymbol{e}_{PR}^{n+1}$, we find that

$$\frac{1}{2}\{\|\boldsymbol{e}_{PR}^{n+1}\|_0^2 - \|\boldsymbol{e}_{PR}^n\|_0^2 + \|\boldsymbol{e}_{PR}^{n+1} - \boldsymbol{e}_{PR}^n\|_0^2\} + \nu\delta t\|\boldsymbol{e}_{PR}^{n+1}\|_1^2 + \delta t(\nabla q_{PR}^n, \boldsymbol{e}_{PR}^{n+1})$$
$$= -\delta t^2(\nabla D_t p_{\delta t}^{n+1}, \boldsymbol{e}_{PR}^{n+1}). \tag{4.35}$$

We treat the last term in the left hand side using (4.34b) as follows,

$$\begin{aligned}(\nabla q_{PR}^n, \boldsymbol{e}_{PR}^{n+1}) = \delta t(\nabla q_{PR}^{n+1}, \nabla q_{PR}^n) &= \delta t\|\nabla q_{PR}^{n+1}\|_0^2 - \delta t(\nabla q_{PR}^{n+1}, \nabla(\delta q_{PR}^{n+1})) \\ &= \delta t\|\nabla q_{PR}^{n+1}\|_0^2 - (\delta\boldsymbol{e}_{PR}^{n+1}, \nabla q_{PR}^{n+1}) \\ &\ge \frac{1}{2}\delta t\|\nabla q_{PR}^{n+1}\|_0^2 - \frac{1}{2\delta t}\|\delta\boldsymbol{e}_{PR}^{n+1}\|_0^2,\end{aligned}$$

owing to (4.34b) and Young's inequality. Inserting this result in (4.35), using again (4.34b) for the right hand side of (4.35), adding up from $n = 0$ to an arbitrary time level N and using the discrete Gronwall Lemma, we get

$$\|\boldsymbol{e}_{PR}^{N}\|_0^2+\sum_{n=0}^{N}\nu\delta t\|\boldsymbol{e}_{PR}^{n+1}\|_1^2+\sum_{n=0}^{N}\delta t^2\|\nabla q_{PR}^{n+1}\|_0^2 \leq C\delta t^2\sum_{n=0}^{N}\delta t^2\|\nabla D_t p_{\delta t}^{n+1}\|_0^2 \leq C\delta t^2, \tag{4.36}$$

using the stability results for the pressure in Lemma 4.7. This proves the error estimates. Moreover, a stability bound for the pressure has been obtained. From (4.34a), invoking the stability bound (4.29) for the divergence operator, it is easily seen that

$$\lambda_n\|q_{PR}^{n}\|_0 \leq \lambda_{n+1}\|\boldsymbol{e}_{PR}^{n+1}\|_1+\lambda_{n+1}\delta t\|D_t p_{\delta t}^{n+1}\|_0+\lambda_{n+1}\|D_t\boldsymbol{e}_{PR}^{n+1}\|_{-1}.$$

The first term is easily bounded from (4.36). For the pressure term we use the error bound for the pressure in Corollary 4.2 and the stability result in Lemma 4.3,

$$\begin{aligned}\lambda_{n+1}\delta t\|D_t p_{\delta t}^{n+1}\|_0 &\leq \lambda_{n+1}\|p(t^{n+1})-p_{\delta t}^{n+1}\|_0+\lambda_{n+1}\|p(t^{n})-p_{\delta t}^{n}\|_0+\lambda_{n+1}\delta t\|D_t p(t^{n+1})\|_0\\ &\leq C\delta t^{1/2}.\end{aligned}$$

For the third term we apply $D_t^2(\cdot)$ over the system (4.34), multiply the first equation by $\lambda_{n+1}D_t\boldsymbol{e}_{PR}^{n+1}$ and invoke the second equation in order to get,

$$\lambda_{n+1}\|D_t\boldsymbol{e}_{PR}^{N+1}\|_1^2 \leq C\delta t\sum_{n=0}^{N}\delta t^3\lambda_{n+1}\|\nabla D_t^2 p_{\delta t}^{n+1}\|_0^2. \tag{4.37}$$

We end the proof obtaining a bound for the RHS of (4.37). Applying D_t^2 over (4.15), multiplying the *momentum* equation by $D_t^2\boldsymbol{u}_{\delta t}^{n+1}$ and exploiting the second equation of the system, we arrive to

$$\sum_{n=0}^{N}\lambda_{n+1}^2\delta t^2\|\nabla D_t^2 p_{\delta t}^{n+1}\|_0^2 \leq C,$$

using an analogous procedure to that in Lemma 4.5. □

The stability results of the penalized problem listed in Lemma 4.7 are maintained for the projection-like system. We prove these bounds in the next lemma.

Lemma 4.11 (Stability I)**.** *Under the assumptions of Lemma 4.7, the following stability results hold*

$$\max_{0\leq n\leq N}\{\|\hat{\boldsymbol{w}}^{n+1}\|_2^2+\|\nabla\phi^{n+1}\|_0^2\} \leq C,$$

for $\delta t > 0$ small enough.

Proof. Let us start the proof with the stability of the pressure gradient. It is easily obtained from the stability results of Lemma 4.7 and the bound for q_{PR}^{n+1} in (4.36),

$$\max_{0\leq n\leq N}\|\nabla\phi^{n+1}\|_0 \leq \max_{0\leq n\leq N}\{\|\nabla p_{\delta t}^{n+1}\|_0+\|\nabla q_{PR}^{n+1}\|_0\} \leq C.$$

The time derivative can also be bounded,

$$\|D_t\hat{\boldsymbol{w}}^{n+1}\|_0 \leq \|D_t\boldsymbol{u}_{\delta t}^{n+1}\|_0 + \frac{1}{\delta t}\|\boldsymbol{e}_{PR}^{n+1}\|_0 + \frac{1}{\delta t}\|\boldsymbol{e}_{PR}^{n}\|_0 \leq C,$$

where for the first term we have used Lemma 4.7 and for the rest the results of Lemma 4.10. Then, taking the $\boldsymbol{L}^2(\Omega)$-norm in (4.33a), we find that

$$\|\nu\Delta\hat{\boldsymbol{w}}^{n+1}\|_0 \leq \|D_t\hat{\boldsymbol{w}}^{n+1}\|_0 + \|\nabla\phi^n\|_0 + \|\boldsymbol{f}(t^{n+1})\|_0 + \|\boldsymbol{u}(t^{n+1})\cdot\nabla\boldsymbol{u}(t^{n+1})\|_0 \leq C,$$

from the previous results. The lemma is proved. □

Remark 4.1. This last result is a key feature of our auxiliary problem and the justification of its introduction. In the convergence analysis of the fully discrete method developed later these properties will be often exploited.

We complete the characterization of the projection-like problem with some estimates for the end-of-step variable $\boldsymbol{w}^{n+1}$ and the difference between $\boldsymbol{w}^{n+1}$ and $\hat{\boldsymbol{w}}^{n+1}$ derived from the previous results. But first we state a new assumption concerning the regularity of the domain.

Assumption 4.3. *Expression (1.19) holds.*

If Ω is of class $\mathcal{C}^2$ this assumption and Assumption 4.2 are satisfied.

Lemma 4.12 (Stability II)**.** *Under the assumptions of Lemma 4.10, the following stability result holds:*

$$\max_{0\leq n\leq N}\|\frac{1}{\delta t}(\boldsymbol{w}^{n+1}-\hat{\boldsymbol{w}}^{n+1})\|_0 \leq C,$$

for $\delta t > 0$. Furthermore, under Assumption 4.3

$$\max_{0\leq n\leq N}\|\boldsymbol{w}^{n+1}\|_1 \leq C.$$

Proof. Let us start with the second result. Thanks to the fact that (1.19) holds, we get

$$\max_{0\leq n\leq N}\|\boldsymbol{w}^{n+1}\|_1 = \max_{0\leq n\leq N}\|P_{\boldsymbol{J}_0}(\hat{\boldsymbol{w}}^{n+1})\|_1 \leq \max_{0\leq n\leq N}\|\hat{\boldsymbol{w}}^{n+1}\|_1 \leq C.$$

Furthermore, taking the L^2-norm in (4.32b), we obtain the following bound,

$$\max_{0\leq n\leq N}\|\frac{1}{\delta t}(\boldsymbol{w}^{n+1}-\hat{\boldsymbol{w}}^{n+1})\|_0 \leq \max_{0\leq n\leq N}\|\nabla\phi^{n+1}\|_0 \leq C,$$

completing the proof of the stability results. □

We end the section with a convergence result for the projection-like system. This result is obtained from Lemma 4.10, the convergence results of Corollary 4.2 for the penalized system and the triangle inequality.

Corollary 4.3 (Convergence)**.** *Under the assumptions of Corollary 4.2, the following convergence results hold*

$$\max_{0\leq n\leq N}\{\|\boldsymbol{u}(t^{n+1})-\hat{\boldsymbol{w}}^{n+1}\|_0 + \|\boldsymbol{u}(t^{n+1})-\boldsymbol{w}^{n+1}\|_0\} \leq C\delta t,$$
$$\max_{0\leq n\leq N}\{\|\boldsymbol{u}(t^{n+1})-\hat{\boldsymbol{w}}^{n+1}\|_1 + \lambda_{n+1}\|p(t^{n+1})-\phi^{n+1}\|_0\} \leq C\sqrt{\delta t},$$

for $\delta t > 0$ small enough.

4.5 Convergence results satisfying the inf-sup condition

In this section we obtain optimal convergence results for the projection method with a velocity-pressure interpolation satisfying the inf-sup condition. Furthermore, the end-of-step velocity and the pressure are solved in a coupled way. It implies that the end-of-step velocity is weakly divergence free. The classical first order fractional step method, also called projection method, in its fully discretized version reads as follows: find the solution $(\hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{u}_h^{n+1}, p_h^{n+1}) \in \mathcal{V}_{h,0} \times \mathcal{Y}_{h,0} \times \mathcal{Q}_h$ such that,

$$\begin{aligned}
&\frac{1}{\delta t}(\hat{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n, \boldsymbol{v}_h) + \nu(\nabla \hat{\boldsymbol{u}}_h^{n+1}, \nabla \boldsymbol{v}_h) \\
&\quad + c(\hat{\boldsymbol{u}}_h^{n+1}, \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h) = \langle \boldsymbol{f}(t^{n+1}), \boldsymbol{v}_h \rangle, \qquad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0} \qquad (4.38a)\\
&\frac{1}{\delta t}(\boldsymbol{u}_h^{n+1} - \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{y}_h) + (\nabla p_h^{n+1}, \boldsymbol{y}_h) = 0, \qquad \forall \boldsymbol{y}_h \in \mathcal{Y}_{h,0}, \qquad (4.38b)\\
&(\nabla \cdot \boldsymbol{u}_h^{n+1}, q_h) = 0, \qquad \forall q_h \in \mathcal{Q}_h. \qquad (4.38c)
\end{aligned}$$

where $\mathcal{V}_{h,0} \times \mathcal{Y}_{h,0} \times \mathcal{Q}_h$ is a finite element approximation space of $\boldsymbol{H}_0^1(\Omega) \times \boldsymbol{H}_0(\mathrm{div}, \Omega) \times L^2(\Omega)/\mathbb{R}$ (see Section 1.3).

For the analysis in this section we need Assumptions 4.1 and 4.2 stated in the previous section and the additional assumptions on the space discretization listed below.

Assumption 4.4. *There exists a constant β_d independent of the mesh size h such that*

$$\inf_{q_h \in \mathcal{Q}_h} \sup_{\boldsymbol{v}_h \in \mathcal{V}_{h,0}} \frac{b(\boldsymbol{v}_h, q_h)}{\|\boldsymbol{v}_h\|_{\mathcal{V}_{h,0}} \|q_h\|_{\mathcal{Q}_h / \ker B_h^t}} \geq \beta_d > 0.$$

We are relying on the inf-sup condition for the analysis of this section.

Assumption 4.5. *The discrete space $\mathcal{V}_{h,0}$ is a subspace of $\mathcal{Y}_{h,0}$, so that*

$$\mathcal{V}_{h,0} \subseteq \mathcal{Y}_{h,0}.$$

Assumption 4.6. *The family of finite element partitions $\{\Theta_h\}_{h>0}$ is quasi-uniform.*

This assumption is needed because if it holds the following inverse estimate (see [23]) can be used: given a $\boldsymbol{v}_h \in \mathcal{V}_h$, there exists a constant C independent of the mesh size h such that

$$\|\boldsymbol{v}_h\|_1 \leq \frac{C}{h} \|\boldsymbol{v}_h\|_0.$$

Assumption 4.7. *There exists a constant $C > 0$ independent of h and δt such that*

$$\delta t \geq C h^2.$$

The proofs of convergence results for this method rely on the auxiliary problem proposed and analyzed in the previous section. We compare the solution of the fully discrete projection method with the solution of the auxiliary problem for which we have optimal convergence results.

Let us denote the *discrete* velocity and pressure errors:

$$\begin{aligned}
\hat{\boldsymbol{e}}_d^{n+1} &= \hat{\boldsymbol{w}}^{n+1} - \hat{\boldsymbol{u}}_h^{n+1},\\
\boldsymbol{e}_d^{n+1} &= \boldsymbol{w}^{n+1} - \boldsymbol{u}_h^{n+1},\\
r_d^{n+1} &= \phi^{n+1} - p_h^{n+1},
\end{aligned}$$

that are governed by the system obtained subtracting (4.38) from (4.32),

$$\frac{1}{\delta t}(\hat{\boldsymbol{e}}_d^{n+1} - \boldsymbol{e}_d^n, \boldsymbol{v}_h) + \nu(\nabla \hat{\boldsymbol{e}}_d^{n+1}, \nabla \boldsymbol{v}_h) = c(\hat{\boldsymbol{u}}_h^{n+1}, \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h) - c(\boldsymbol{u}(t^{n+1}), \boldsymbol{u}(t^{n+1}), \boldsymbol{v}_h) \tag{4.39a}$$

$$\frac{1}{\delta t}(\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}, \boldsymbol{y}_h) + (\nabla r_d^{n+1}, \boldsymbol{y}_h) = 0, \tag{4.39b}$$

$$(\nabla \cdot \boldsymbol{e}_d^{n+1}, q_h) = 0. \tag{4.39c}$$

Likewise, we introduce the *continuous* velocity errors:

$$\hat{\boldsymbol{e}}_c^{n+1} = \boldsymbol{u}(t^{n+1}) - \hat{\boldsymbol{w}}^{n+1},$$
$$\boldsymbol{e}_c^{n+1} = \boldsymbol{u}(t^{n+1}) - \boldsymbol{w}^{n+1}.$$

We also use the following notation for the interpolation error:

$$\begin{aligned}\tilde{\mathcal{I}}_n(h) = {} & \frac{1}{h}\inf_{\boldsymbol{v}_h \in \mathcal{V}_{h,0}} \|\hat{\boldsymbol{w}}^{n+1} - \boldsymbol{v}_h\|_0 + \frac{1}{h}\inf_{\boldsymbol{v}_h \in \mathcal{V}_{h,0}} \|\boldsymbol{w}^{n+1} - \boldsymbol{v}_h\|_0 \\ & + \inf_{\boldsymbol{v}_h \in \mathcal{V}_{h,0}} \|\hat{\boldsymbol{w}}^{n+1} - \boldsymbol{v}_h\|_1 + \inf_{q_h \in \mathcal{Q}_h} \|\phi^{n+1} - q_h\|_0 \\ & + h \inf_{q_h \in Q_h} \|\phi^{n+1} - q_h\|_1 + h \inf_{\boldsymbol{\eta}_h \in \mathcal{V}_h} \|\nabla \phi^{n+1} - \boldsymbol{\eta}_h\|_0 \\ \tilde{\mathcal{I}}(h) = {} & \max_{0 \le n \le N} \tilde{\mathcal{I}}_n(h).\end{aligned}$$

We note that the interpolation error introduced is related to the auxiliary variables.

Under the discrete inf-sup condition, optimal order approximation both in $H_0^1(\Omega)$ and in $L^2(\Omega)$ of solenoidal vector fields can be achieved by means of discretely divergence free finite element functions, that is, $\boldsymbol{w}_h \in \ker B_h$. In fact, one has the following result.

Lemma 4.13. *Let $\boldsymbol{u} \in \boldsymbol{J}_1$ and assume that discrete spaces $\mathcal{V}_{h,0}$ and $\mathcal{Q}_h$ satisfy Assumption 4.4 and 4.6. Then,*

$$\inf_{\boldsymbol{x}_h \in \ker B_h} \|\boldsymbol{u} - \boldsymbol{x}_h\|_1 \le C \inf_{\boldsymbol{v}_h \in \mathcal{V}_h} \|\boldsymbol{u} - \boldsymbol{v}_h\|_1.$$

Moreover, under Assumption 4.2 we also have,

$$\inf_{\boldsymbol{x}_h \in \ker B_h} \|\boldsymbol{u} - \boldsymbol{x}_h\|_0 \le Ch \inf_{\boldsymbol{v}_h \in \mathcal{V}_h} \|\boldsymbol{u} - \boldsymbol{v}_h\|_1.$$

Proof. Let us consider the following Stokes problem: find $(\boldsymbol{y}, a) \in \boldsymbol{H}_0^1(\Omega) \times L^2(\Omega)/\mathbb{R}$ such that, for all $(\boldsymbol{v}, q) \in \boldsymbol{H}_0^1(\Omega) \times L^2(\Omega)/\mathbb{R}$:

$$(\nabla \boldsymbol{y}, \nabla \boldsymbol{v}) - (a, \nabla \cdot \boldsymbol{v}) = (\nabla \boldsymbol{u}, \nabla \boldsymbol{v}),$$
$$(q, \nabla \cdot \boldsymbol{y}) = 0.$$

Given $\boldsymbol{u} \in \boldsymbol{H}_0^1(\Omega)$, $(\nabla \boldsymbol{u}, \nabla \boldsymbol{v})$ is a linear, continuous functional on $\boldsymbol{H}_0^1(\Omega)$, and thus this problem admits a unique solution $(\boldsymbol{y}, a) = (\boldsymbol{u}, 0)$. We next consider the finite element discrete problem of finding $(\boldsymbol{y}_h, a_h) \in \mathcal{V}_{h,0} \times \mathcal{Q}_h$ such that:

$$(\nabla \boldsymbol{y}_h, \nabla \boldsymbol{v}_h) - (a_h, \nabla \cdot \boldsymbol{v}_h) = (\nabla \boldsymbol{u}, \nabla \boldsymbol{v}_h),$$
$$(q_h, \nabla \cdot y_h) = 0.$$

Standard approximation results for this problem (see for instance [81]), allow us to conclude that:

$$\inf_{\boldsymbol{x}_h \in \ker B_h} \|\boldsymbol{u} - \boldsymbol{x}_h\|_1 \leq C \inf_{\boldsymbol{v}_h \in \mathcal{V}_{h,0}} \|\boldsymbol{u} - \boldsymbol{v}_h\|_1,$$

and, if Ω is regular enough,

$$\inf_{\boldsymbol{x}_h \in \ker B_h} \|\boldsymbol{u} - \boldsymbol{x}_h\|_0 \leq Ch \inf_{\boldsymbol{v}_h \in \mathcal{V}_{h,0}} \|\boldsymbol{u} - \boldsymbol{v}_h\|_1.$$

The proof is finished. □

A first error estimate showing that $\{\hat{\boldsymbol{u}}_h^{n+1}\}_{n=0,\dots,N}$ and $\{\boldsymbol{u}_h^{n+1}\}_{n=0,\dots,N}$ are first order approximations to $\{\boldsymbol{u}(t^{n+1})\}_{n=0,\dots,N}$ in $\ell^\infty(\boldsymbol{L}^2(\Omega)) \cap \ell^2(\boldsymbol{H}^1(\Omega))$ is stablished in the following theorem.

Theorem 4.1. *Being valid Assumption 4.1, 4.2 , 4.4, 4.5, 4.6 and 4.7, we have*

$$\begin{aligned}\max_{0\leq n\leq N}\{\|\boldsymbol{e}_d^{N+1}\|_0^2 + \|\hat{\boldsymbol{e}}_d^{N+1}\|_0^2\} + \sum_{n=0}^{N} \nu\delta t\{\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2 + \|\boldsymbol{e}_d^{n+1}\|_1^2\} \\ + \sum_{n=0}^{N}\{\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_1^2 + \|\hat{\boldsymbol{e}}_d^{n+1} - \boldsymbol{e}_d^{n+1}\|_1^2\} + \sum_{n=0}^{N} \delta t^2 \|r_d^{n+1}\|_0^2 \leq C(\tilde{\mathcal{I}}(h)^2 + \delta t^2),\end{aligned}$$

for $\delta t > 0$ small enough.

Proof. Taking $\delta t(\boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1})$ as a test function in (4.39a), we obtain

$$\begin{aligned}&\frac{1}{2}\left(\|\hat{\boldsymbol{e}}_d^{n+1}\|_0^2 - \|\boldsymbol{e}_d^n\|_0^2 + \|\hat{\boldsymbol{e}}_d^{n+1} - \boldsymbol{e}_d^n\|_0^2\right) + \nu\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2 \\ &\quad = \delta t c(\hat{\boldsymbol{u}}_h^{n+1}, \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}) - \delta t c(\boldsymbol{u}(t^{n+1}), \boldsymbol{u}(t^{n+1}), \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}) \\ &\quad + (\hat{\boldsymbol{e}}_d^{n+1} - \boldsymbol{e}_d^n, \hat{\boldsymbol{w}}^{n+1} - \boldsymbol{v}_h) + \nu\delta t(\nabla\hat{\boldsymbol{e}}_d^{n+1}, \nabla(\hat{\boldsymbol{w}}^{n+1} - \boldsymbol{v}_h)).\end{aligned} \tag{4.40}$$

We combine the nonlinear terms as follows:

$$\begin{aligned}&-c(\boldsymbol{u}(t^{n+1}), \boldsymbol{u}(t^{n+1}), \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}) + c(\hat{\boldsymbol{u}}_h^{n+1}, \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}) \\ &\quad = -c(\boldsymbol{u}(t^{n+1}), \hat{\boldsymbol{e}}_c^{n+1}, \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}) - c(\hat{\boldsymbol{u}}_h^{n+1}, \hat{\boldsymbol{e}}_d^{n+1}, \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}) \\ &\qquad - c(\hat{\boldsymbol{e}}_c^{n+1}, \hat{\boldsymbol{w}}^{n+1}, \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}) - c(\hat{\boldsymbol{e}}_d^{n+1}, \hat{\boldsymbol{w}}^{n+1}, \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}) \\ &\quad = -c(\boldsymbol{u}(t^{n+1}), \hat{\boldsymbol{e}}_c^{n+1}, \hat{\boldsymbol{e}}_d^{n+1}) + c(\boldsymbol{u}(t^{n+1}), \hat{\boldsymbol{e}}_c^{n+1}, \hat{\boldsymbol{w}}^{n+1} - \boldsymbol{v}_h) - c(\hat{\boldsymbol{w}}^{n+1}, \hat{\boldsymbol{e}}_d^{n+1}, \hat{\boldsymbol{e}}_d^{n+1}) \\ &\qquad + c(\hat{\boldsymbol{w}}^{n+1}, \hat{\boldsymbol{e}}_d^{n+1}, \hat{\boldsymbol{w}}^{n+1} - \boldsymbol{v}_h) + c(\hat{\boldsymbol{e}}_d^{n+1}, \hat{\boldsymbol{e}}_d^{n+1}, \hat{\boldsymbol{e}}_d^{n+1}) - c(\hat{\boldsymbol{e}}_d^{n+1}, \hat{\boldsymbol{e}}_d^{n+1}, \hat{\boldsymbol{w}}^{n+1} - \boldsymbol{v}_h) \\ &\qquad - c(\hat{\boldsymbol{e}}_c^{n+1}, \hat{\boldsymbol{w}}^{n+1}, \hat{\boldsymbol{e}}_d^{n+1}) + c(\hat{\boldsymbol{e}}_c^{n+1}, \hat{\boldsymbol{w}}^{n+1}, \hat{\boldsymbol{w}}^{n+1} - \boldsymbol{v}_h) - c(\hat{\boldsymbol{e}}_d^{n+1}, \hat{\boldsymbol{w}}^{n+1}, \hat{\boldsymbol{e}}_d^{n+1}) \\ &\qquad + c(\hat{\boldsymbol{e}}_d^{n+1}, \hat{\boldsymbol{w}}^{n+1}, \hat{\boldsymbol{w}}^{n+1} - \boldsymbol{v}_h).\end{aligned}$$

Every nonlinear term can be appropriately bounded as follows:

$$\begin{aligned}
-c(\boldsymbol{u}(t^{n+1}),\hat{\boldsymbol{e}}_c^{n+1},\hat{\boldsymbol{e}}_d^{n+1}) &\le C\|\boldsymbol{u}(t^{n+1})\|_2\|\hat{\boldsymbol{e}}_c^{n+1}\|_0\|\hat{\boldsymbol{e}}_d^{n+1}\|_1\\
&\le C\|\hat{\boldsymbol{e}}_c^{n+1}\|_0^2+\frac{1}{8}\nu\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2\\
&\le C\delta t^2+\frac{1}{8}\nu\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2,\\
c(\boldsymbol{u}(t^{n+1}),\hat{\boldsymbol{e}}_c^{n+1},\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h) &\le C\|\boldsymbol{u}(t^{n+1})\|_2\|\hat{\boldsymbol{e}}_c^{n+1}\|_0\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1\\
&\le C\|\hat{\boldsymbol{e}}_c^{n+1}\|_0^2+C\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1^2\\
&\le C\delta t^2+C\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1^2,\\
-c(\hat{\boldsymbol{w}}^{n+1},\hat{\boldsymbol{e}}_d^{n+1},\hat{\boldsymbol{e}}_d^{n+1}) &= 0,\\
c(\hat{\boldsymbol{w}}^{n+1},\hat{\boldsymbol{e}}_d^{n+1},\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h) &\le \|\hat{\boldsymbol{w}}^{n+1}\|_1\|\hat{\boldsymbol{e}}_d^{n+1}\|_1\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1\\
&\le \frac{1}{8}\nu\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2+C\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1^2,\\
c(\hat{\boldsymbol{e}}_d^{n+1},\hat{\boldsymbol{e}}_d^{n+1},\hat{\boldsymbol{e}}_d^{n+1}) &= 0,\\
-c(\hat{\boldsymbol{e}}_d^{n+1},\hat{\boldsymbol{e}}_d^{n+1},\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h) &\le C\|\hat{\boldsymbol{e}}_d^{n+1}\|_1\|\hat{\boldsymbol{e}}_d^{n+1}\|_1\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1\\
&\le \frac{1}{8}\nu\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2+C\nu\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1^2,\\
-c(\hat{\boldsymbol{e}}_c^{n+1},\hat{\boldsymbol{w}}^{n+1},\hat{\boldsymbol{e}}_d^{n+1}) &\le C\|\hat{\boldsymbol{e}}_c^{n+1}\|_0\|\hat{\boldsymbol{w}}^{n+1}\|_2\|\hat{\boldsymbol{e}}_d^{n+1}\|_1\\
&\le C\|\hat{\boldsymbol{e}}_c^{n+1}\|_0^2+\frac{1}{8}\nu\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2\\
&\le C\delta t^2+\frac{1}{8}\nu\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2,\\
c(\hat{\boldsymbol{e}}_c^{n+1},\hat{\boldsymbol{w}}^{n+1},\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h) &\le C\|\hat{\boldsymbol{e}}_c^{n+1}\|_0\|\hat{\boldsymbol{w}}^{n+1}\|_2\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1\\
&\le C\|\hat{\boldsymbol{e}}_c^{n+1}\|_0^2+C\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1^2\\
&\le C\delta t^2+C\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1^2,\\
-c(\hat{\boldsymbol{e}}_d^{n+1},\hat{\boldsymbol{w}}^{n+1},\hat{\boldsymbol{e}}_d^{n+1}) &\le \|\hat{\boldsymbol{e}}_d^{n+1}\|_0\|\hat{\boldsymbol{w}}^{n+1}\|_2\|\hat{\boldsymbol{e}}_d^{n+1}\|_1\\
&\le C\|\boldsymbol{e}_d^{n+1}\|_0^2+C\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2+\frac{1}{8}\nu\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2,\\
c(\hat{\boldsymbol{e}}_d^{n+1},\hat{\boldsymbol{w}}^{n+1},\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h) &\le \|\hat{\boldsymbol{e}}_d^{n+1}\|_0\|\hat{\boldsymbol{w}}^{n+1}\|_2\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1\\
&\le C\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_0^2+C\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1^2.
\end{aligned}$$

The rest of the terms can also be bounded,

$$\begin{aligned}
(\hat{\boldsymbol{e}}_d^{n+1}-\boldsymbol{e}_d^n,\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h) &\le \frac{1}{3}\|\hat{\boldsymbol{e}}_d^{n+1}-\boldsymbol{e}_d^n\|_0^2+C\frac{\delta t}{h^2}\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_0^2,\\
\nu(\nabla\hat{\boldsymbol{e}}_d^{n+1},\nabla(\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h)) &\le \frac{1}{8}\nu\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2+C\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1^2,
\end{aligned}$$

where Assumption 4.7 is used for the first inequality.

Using all these inequalities in (4.40), invoking the stability estimates for $\hat{\boldsymbol{e}}_d^{n+1}$ and adding up from $n=0$ to $n=N$, we get

$$\begin{aligned}
&\frac{1}{2}\sum_{n=0}^{N}\left(\|\hat{\boldsymbol{e}}_d^{n+1}\|_0^2-\|\boldsymbol{e}_d^n\|_0^2+\|\hat{\boldsymbol{e}}_d^{n+1}-\boldsymbol{e}_d^n\|_0^2\right)+\sum_{n=0}^{N}\nu\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2\\
&\qquad\le C\delta t^2+\frac{6}{8}\sum_{n=0}^{N}\nu\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2+\frac{1}{3}\sum_{n=0}^{N}\|\hat{\boldsymbol{e}}_d^{n+1}-\boldsymbol{e}_d^n\|_0^2
\end{aligned}$$

$$
\begin{aligned}
&+ C\sum_{n=0}^{N}\delta t\left(\|\boldsymbol{e}_d^{n+1}\|_0^2+\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2\right)\\
&+ C(1+\sum_{n=0}^{N}\nu\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2)\tilde{\mathcal{I}}(h)^2\\
&\leq C(\tilde{\mathcal{I}}(h)^2+\delta t^2)+\frac{6}{8}\sum_{n=0}^{N}\nu\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2+\frac{1}{3}\sum_{n=0}^{N}\|\hat{\boldsymbol{e}}_d^{n+1}-\boldsymbol{e}_d^{n}\|_0^2\\
&+ C\sum_{n=0}^{N}\delta t\left(\|\boldsymbol{e}_d^{n+1}\|_0^2+\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2\right). \qquad (4.41)
\end{aligned}
$$

Taking $\delta t(\boldsymbol{x}_h-\boldsymbol{u}_h^{n+1})$ as test function in (4.39b) (valid due to Assumption 3), where $\boldsymbol{x}_h\in\ker B_h$, we obtain

$$\frac{1}{2}\left(\|\boldsymbol{e}_d^{n+1}\|_0^2-\|\hat{\boldsymbol{e}}_d^{n+1}\|_0^2+\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2\right)=(\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1},\boldsymbol{w}^{n+1}-\boldsymbol{x}_h).$$

The optimal approximation properties obtained in the Lemma 4.13 for functions $\boldsymbol{x}_h\in\ker B_h$ allow us to bound the RHS, obtaining

$$\frac{1}{2}\left(\|\boldsymbol{e}_d^{n+1}\|_0^2-\|\hat{\boldsymbol{e}}_d^{n+1}\|_0^2+\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2\right)\leq C\delta t\mathcal{I}_n(h)^2+\frac{1}{3}\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2.$$

Adding up from $n=0$ to $n=N$, we get

$$\frac{1}{2}\sum_{n=0}^{N}\left(\|\boldsymbol{e}_d^{n+1}\|_0^2-\|\hat{\boldsymbol{e}}_d^{n+1}\|_0^2+\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2\right)\leq C\tilde{\mathcal{I}}(h)^2+\frac{1}{3}\sum_{n=0}^{N}\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2. \quad (4.42)$$

Adding (4.41) and (4.42), using the discrete Gronwall lemma (see [98]) and cancelling some terms we obtain,

$$\|\boldsymbol{e}_d^{N+1}\|_0^2+\sum_{n=0}^{N}\left(\|\hat{\boldsymbol{e}}_d^{n+1}-\boldsymbol{e}_d^{n}\|_0^2+\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2\right)+\sum_{n=0}^{N}\nu\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2\leq C(\tilde{\mathcal{I}}(h)^2+\delta t^2).$$

For the pressure error we take benefit from the inf-sup condition stated in Assumption 4.4 and equation (4.39b) in order to get,

$$\delta t\|r_d^{n+1}\|_0\leq C\delta t\sup_{v_h\in V_h}\frac{(\nabla r_d^{n+1},\boldsymbol{v}_h)}{\|\boldsymbol{v}_h\|_1}\leq C\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_{-1}.$$

The bounds for the velocity errors lead to the convergence estimate for the pressure. □

Theorem 4.1 establishes optimal convergence results for the approximated velocity when the classical first order projection method in its fully discretized form is used.

We end our analysis with a corollary that gives the convergence results obtained for the fully discrete fractional step method relying on the inf-sup condition. It is a direct consequence of Corollary 4.3 and Theorem 4.1 using the triangle inequality.

Corollary 4.4 (Convergence)**.** *Under the assumption of Theorem 4.1 and* $\partial_t^3\boldsymbol{f}(t)\in L^\infty(0,T;\boldsymbol{L}^2(\Omega))$ *the following error estimates hold,*

$$
\begin{aligned}
&\max_{0\leq n\leq N}\{\|\boldsymbol{u}(t^{n+1})-\boldsymbol{u}_h^{n+1}\|_0+\|\boldsymbol{u}(t^{n+1})-\hat{\boldsymbol{u}}_h^{n+1}\|_0\\
&\quad+\sqrt{\delta t}\|\boldsymbol{u}(t^{n+1})-\hat{\boldsymbol{u}}_h^{n+1}\|_1\}+\lambda_{n+1}\sqrt{\delta t}\left(\sum_{n=0}^{N}\delta t\|p(t^{n+1})-p_h^{n+1}\|_0^2\right)^{1/2}\leq C(\tilde{\mathcal{I}}(h)+\delta t),
\end{aligned}
$$

for $\delta t>0$ *small enough.*

4.6 Convergence results for a stabilized scheme using the pressure Poisson equation

In order to obtain the pressure of the system analyzed in the previous section, a Stokes-like problem has to be solved. On the other hand, for the fractional step method studied in this section only a simple Poisson equation must be solved. This so-called Pressure Poisson equation is by far the most used approach, not only for its low computational cost, but also because of its inherent stability studied in [47, 52]. At the continuous level, these two approximations are equivalent assuming some regularity properties. But at the fully discrete level, the second procedure introduces a perturbation term in the mass conservation equation.

In this section we analyze a fully discrete fractional step method using a Poisson equation for the pressure. In Section 4.3 we have obtained error estimates for the auxiliary problem at the continuous level and now we compare the fully discrete fractional step method with the auxiliary problem. We begin the analysis of the discrete problem stating the new assumptions we have had to use:

Assumption 4.8. *There exist $\alpha_- > 0$ and $\alpha_+ > 0$ independent of h such that:*

$$\alpha_- h^2 \leq \delta t \leq \alpha_+ h^2.$$

This assumption dictates the behaviour of δt. It could seem a very restrictive hypothesis but we will see in the following analysis that δt is playing the same role as the numerical stabilization parameter of stabilized finite element methods for which this assumption is mandatory (see [15]).

Assumption 4.9. *As in [53, 54], let $\nabla\mathcal{Q}_h$ denote the space*

$$\nabla\mathcal{Q}_h = \{\boldsymbol{v}_h \in \boldsymbol{L}^2(\Omega) \mid \boldsymbol{v}_h = \nabla q_h,\ q_h \in \mathcal{Q}_h\}$$

and define the space $\boldsymbol{E}_h$ by

$$\boldsymbol{E}_h = \mathcal{V}_h + \nabla\mathcal{Q}_h \subset \boldsymbol{L}^2(\Omega).$$

We consider three mutually orthogonal subspaces $\boldsymbol{E}_{h,i}$ of $\boldsymbol{E}_h$ defined by

$$\boldsymbol{E}_{h,1} = \mathcal{V}_{h,0}, \qquad \boldsymbol{E}_{h,2} = \mathcal{V}_{h,0}^{\perp} \cap \mathcal{V}_h, \qquad \boldsymbol{E}_{h,3} = \mathcal{V}_h^{\perp} \cap \boldsymbol{E}_h$$

so that

$$\boldsymbol{E}_h = \boldsymbol{E}_{h,1} \oplus \boldsymbol{E}_{h,2} \oplus \boldsymbol{E}_{h,3}.$$

For $i = 1, 2, 3$ we call Π_i the L^2-projection of $\boldsymbol{E}_h$ onto $\boldsymbol{E}_{h,i}$, and for $i \neq j$, $\Pi_{ij} = \Pi_i + \Pi_j$ and $\boldsymbol{E}_{h,ij} = \boldsymbol{E}_{h,i} \oplus \boldsymbol{E}_{h,j}$. We assume that there is a constant β_0 independent of h such that

$$\|\nabla q_h\|_0 \leq \beta_0 \|\Pi_{13}(\nabla q_h)\|_0 \quad \forall q_h \in Q_h,$$

that is to say, that the second component of the decomposition of every ∇q_h in $\boldsymbol{E}_h$ can be bounded in terms of the other two. This condition can also be written in the form

$$\inf_{q_h \in Q_h} \sup_{\boldsymbol{v}_h \in \boldsymbol{E}_{h,13}} \frac{b(\boldsymbol{v}_h, q_h)}{\|\boldsymbol{v}_h\|_1 \|q_h\|_0} \geq \beta_0 > 0, \tag{4.43}$$

in a similar way to the classical inf-sup condition. However, this requirement is weaker since the space where the supremum is taken, $\boldsymbol{E}_{h,13}$, is larger than in a classical case, $\mathcal{V}_{h,0} = \boldsymbol{E}_{h,1}$. Condition (4.43) was analized in [53], where it was shown to be satisfied by equal order simplicial finite element interpolations.

From the previous definitions we can easily see that,

$$\Pi_3(\nabla q_h) = \Pi_h^{\perp}(\nabla q_h), \tag{4.44}$$

for all $q_h \in \mathcal{Q}_h$, $\Pi_h(\cdot)$ being the L^2-projection onto the finite element space $\mathcal{V}_h$. We will also use the projector $\Pi_{\mathcal{Y}_{h,0}}(\cdot)$, that consist of the L^2-projection onto the finite element space $\mathcal{Y}_{h,0}$. The equations for the intermediate and end-of-step velocities are (4.38a) and (4.38c) respectively. The difference with respect to the previous section is the continuity equation, that now is replaced by a Pressure Poisson equation obtained by testing (4.9a) with $q_h \in \mathcal{Q}_h$ and seeking a pressure $p_h^{n+1} \in \mathcal{Q}_h$ such that

$$(\nabla \cdot \hat{\boldsymbol{u}}_h^{n+1}, q_h) + \delta t(\nabla p_h^{n+1}, \nabla q_h) = 0. \tag{4.45}$$

An alternative version of the pressure Poisson equation more appropriate for the following analysis is obtained using (4.38b), which implies that

$$\hat{\boldsymbol{u}}_h^{n+1} = \boldsymbol{u}_h^{n+1} + \delta t \Pi_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1}).$$

Using that in (4.45) we get

$$\begin{aligned}(\nabla \cdot \boldsymbol{u}_h^{n+1}, q_h) - \delta t(\Pi_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1}), \nabla q_h) + \delta t(\nabla p_h^{n+1}, \nabla q_h) \\ = (\nabla \cdot \boldsymbol{u}_h^{n+1}, q_h) + \delta t(\Pi_{\mathcal{Y}_{h,0}}^{\perp}(\nabla p_h^{n+1}), \nabla q_h) = 0.\end{aligned}$$

Then, the fully discrete method to be analyzed consists of finding $\hat{\boldsymbol{u}}_h^{n+1} \in \mathcal{V}_{h,0}$, $\boldsymbol{u}_h^{n+1} \in \mathcal{Y}_{h,0}$ and $p_h^{n+1} \in \mathcal{Q}_h$ such that:

$$\frac{1}{\delta t}(\hat{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n, \boldsymbol{v}_h) + \nu(\nabla \hat{\boldsymbol{u}}_h^{n+1}, \nabla \boldsymbol{v}_h) + c(\hat{\boldsymbol{u}}_h^{n+1}, \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h) = \langle \boldsymbol{f}(t^{n+1}), \boldsymbol{v}_h \rangle, \tag{4.46a}$$

$$\frac{1}{\delta t}(\boldsymbol{u}_h^{n+1} - \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{y}_h) + b(\boldsymbol{y}_h, p_h^{n+1}) = 0, \tag{4.46b}$$

$$(\nabla \cdot \boldsymbol{u}_h^{n+1}, q_h) + \delta t(\Pi_{\mathcal{Y}_{h,0}}^{\perp}(\nabla p_h^{n+1}), \nabla q_h) = 0, \tag{4.46c}$$

where we have taken the advection velocity implicitly. However, a straightforward extension of the analysis that follows allows to obtain the same estimates for the explicit case. Let us introduce new *discrete* errors, for which we use the same notation as in the previous section:

$$\begin{aligned}\hat{\boldsymbol{e}}_d^{n+1} &= \hat{\boldsymbol{w}}^{n+1} - \hat{\boldsymbol{u}}_h^{n+1}, \\ \boldsymbol{e}_d^{n+1} &= \boldsymbol{w}^{n+1} - \boldsymbol{u}_h^{n+1}, \\ r_d^{n+1} &= \phi^{n+1} - p_h^{n+1},\end{aligned}$$

that we need for the following analysis, together with the error functions defined in the previous section. Subtracting (4.46) from the weak form of (4.32), it can be seen that the discrete errors satisfy the following system, which holds for all $(\boldsymbol{v}_h, \boldsymbol{y}_h, q_h) \in \mathcal{V}_{h,0} \times \mathcal{Y}_{h,0} \times \mathcal{Q}_h$:

$$\begin{aligned}\frac{1}{\delta t}(\hat{\boldsymbol{e}}_d^{n+1} - \boldsymbol{e}_d^n, \boldsymbol{v}_h) + \nu(\nabla \hat{\boldsymbol{e}}_d^{n+1}, \nabla \boldsymbol{v}_h) \\ = c(\hat{\boldsymbol{u}}_h^{n+1}, \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h) - c(\boldsymbol{u}(t^{n+1}), \boldsymbol{u}(t^{n+1}), \boldsymbol{v}_h),\end{aligned} \tag{4.47a}$$

$$\frac{1}{\delta t}(\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}, \boldsymbol{y}_h) + (\nabla r_d^{n+1}, \boldsymbol{y}_h) = 0, \tag{4.47b}$$

$$(\nabla \cdot \boldsymbol{e}_d^{n+1}, q_h) - \delta t(\Pi_{\mathcal{Y}_{h,0}}^{\perp}(\nabla p_h^{n+1}), \nabla q_h) = 0. \tag{4.47c}$$

We also use the following notation for the interpolation error:

$$\begin{aligned}\tilde{\mathcal{I}}_n(h) &= \frac{1}{h}\inf_{\boldsymbol{v}_h\in\mathcal{V}_{h,0}}\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_0+\frac{1}{h}\inf_{\boldsymbol{v}_h\in\mathcal{V}_{h,0}}\|\boldsymbol{w}^{n+1}-\boldsymbol{v}_h\|_0\\ &+\inf_{\boldsymbol{v}_h\in\mathcal{V}_{h,0}}\|\hat{\boldsymbol{w}}^{n+1}-\boldsymbol{v}_h\|_1+\inf_{q_h\in\mathcal{Q}_h}\|\phi^{n+1}-q_h\|_0\\ &+h\inf_{q_h\in Q_h}\|\phi^{n+1}-q_h\|_1+h\inf_{\boldsymbol{y}_h\in\mathcal{Y}_{h,0}}\|\nabla\phi^{n+1}-\boldsymbol{y}_h\|_0\\ \tilde{\mathcal{I}}(h) &= \max_{0\le n\le N}\tilde{\mathcal{I}}_n(h).\end{aligned}$$

The last term of $\tilde{\mathcal{I}}_n(h)$ differs from the expression used in the previous analysis (under the inf-sup condition). This term is of special interest, as it will be commented below.

Before proving the error estimates, we need a previous lemma that gives us a bound for the pressure error. The lemma reads as follows:

Lemma 4.14. *The following bound holds in the range $0\le n\le N$:*

$$\|\nabla r_d^{n+1}\|_0\le C\left(\frac{1}{h}\tilde{\mathcal{I}}_n(h)+\frac{1}{\delta t}\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0+\|\Pi_3(\nabla p_h^{n+1})\|_0\right)$$

for all $\delta t>0$.

Proof. By the triangle inequality and previous definitions, we have

$$\begin{aligned}\|\nabla r_d^{n+1}\|_0 &= \|\nabla(\phi^{n+1}-p_h^{n+1})\|_0\\ &\le \|\nabla\phi^{n+1}-\Pi_{12}(\nabla q_h)\|_0+\|\Pi_1(\nabla q_h-\nabla p_h^{n+1})\|_0\\ &\quad+\|\Pi_2(\nabla q_h-\nabla p_h^{n+1})\|_0+\|\Pi_3(\nabla p_h^{n+1})\|_0\\ &= \widetilde{\mathrm{I}}+\widetilde{\mathrm{II}}+\widetilde{\mathrm{III}}+\widetilde{\mathrm{IV}}.\end{aligned}$$

We bound these terms separately. For the first term we can easily obtain,

$$\begin{aligned}\widetilde{\mathrm{I}}=\|\nabla\phi^{n+1}-\Pi_{12}(\nabla q_h)\|_0 &\le \|\Pi_3(\nabla\phi^{n+1})\|_0+\|\Pi_{12}(\nabla\phi^{n+1}-\nabla q_h)\|_0\\ &\le \|\Pi_3(\nabla\phi^{n+1}-\boldsymbol{y}_h)\|_0+\|\phi^{n+1}-q_h\|_1\\ &\le \|\nabla\phi^{n+1}-\boldsymbol{y}_h\|_0+\|\phi^{n+1}-q_h\|_1,\end{aligned}$$

for all $\boldsymbol{y}_h\in\mathcal{Y}_{h,0}\subset\boldsymbol{E}_{h,12}$ For the second term we have,

$$\begin{aligned}(\widetilde{\mathrm{II}})^2=\|\Pi_1(\nabla q_h-\nabla p_h^{n+1})\|_0^2 &= (\nabla q_h-\nabla p_h^{n+1},\Pi_1(\nabla q_h-\nabla p_h^{n+1}))\\ &= (\nabla q_h-\nabla\phi^{n+1},\Pi_1(\nabla q_h-\nabla p_h^{n+1}))+(\nabla r_d^{n+1},\Pi_1(\nabla q_h-\nabla p_h^{n+1}))\\ &= \widetilde{\mathrm{II}}_a+\widetilde{\mathrm{II}}_b.\end{aligned}$$

The first term is easily bounded,

$$\widetilde{\mathrm{II}}_a=(\nabla q_h-\nabla\phi^{n+1},\Pi_1(\nabla q_h-\nabla p_h^{n+1}))\le\|\phi^{n+1}-q_h\|_1\|\Pi_1(\nabla q_h-\nabla p_h^{n+1})\|_0.$$

For the second term, we take $\boldsymbol{y}_h=\Pi_1(\nabla q_h-\nabla p_h^{n+1}))$ in (4.47b), obtaining

$$\begin{aligned}\widetilde{\mathrm{II}}_b &= (\nabla r_d^{n+1},\Pi_1(\nabla q_h-\nabla p_h^{n+1}))=-\frac{1}{\delta t}(\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1},\Pi_1(\nabla q_h-\nabla p_h^{n+1}))\\ &\le\frac{1}{\delta t}\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0\|\Pi_1(\nabla q_h-\nabla p_h^{n+1})\|_0.\end{aligned}$$

Then,

$$\widetilde{\mathrm{II}} \leq \|\phi^{n+1} - q_h\|_1 + \frac{1}{\delta t}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0.$$

Moreover, due to Assumption 4.9 we can obtain

$$\begin{aligned}\widetilde{\mathrm{III}} &= \|\Pi_2(\nabla q_h - \nabla p_h^{n+1})\|_0 \leq C(\|\Pi_1(\nabla q_h - \nabla p_h^{n+1})\|_0 + \|\Pi_3(\nabla q_h - \nabla p_h^{n+1})\|_0)\\ &\leq C(\widetilde{\mathrm{II}} + \|\Pi_3(\nabla q_h)\|_0 + \|\Pi_3(\nabla p_h^{n+1})\|_0)\\ &\leq C(\widetilde{\mathrm{I}} + \widetilde{\mathrm{II}} + \|\nabla\phi^{n+1} - \nabla q_h\|_0 + 2\|\Pi_3(\nabla p_h^{n+1})\|_0).\end{aligned}$$

Then, taking the infimum with respect to $q_h \in Q_h$, we get

$$\|\nabla r_d^{n+1}\|_0 \leq C\left(\frac{1}{h}\tilde{\mathcal{I}}_n(h) + \frac{1}{\delta t}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0 + \|\Pi_3(\nabla p_h^{n+1})\|_0\right).$$

So that the proof is finished. □

The error estimates we want to obtain are established in the next theorem:

Theorem 4.2. *Under Assumptions 4.1, 4.2, 4.8, 4.6 and 4.9, we have*

$$\begin{aligned}&\max_{0\leq n\leq N}\{\|\boldsymbol{e}_d^{N+1}\|_0^2 + \|\hat{\boldsymbol{e}}_d^{N+1}\|_0^2\} + \sum_{n=0}^{N}\nu\delta t\{\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2 + \|\boldsymbol{e}_d^{n+1}\|_1^2\}\\ &\quad + \sum_{n=0}^{N}\{\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_1^2 + \|\hat{\boldsymbol{e}}_d^{n+1} - \boldsymbol{e}_d^{n+1}\|_1^2\} + \sum_{n=0}^{N}\delta t^2\|\nabla r_d^{n+1}\|_0^2 \leq C(\tilde{\mathcal{I}}(h)^2 + \delta t^2)\end{aligned}$$

for $\delta t > 0$ small enough.

Proof. The first step of this proof is similar to that in the Theorem 4.1. We take $\delta t(\boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1})$ as a test function in (4.47a). Using some bounds from Theorem 4.1, invoking the stability estimates for $\hat{\boldsymbol{e}}_d^{n+1}$ and adding up from $n = 0$ to $n = N$, we get

$$\begin{aligned}&\frac{1}{2}\sum_{n=0}^{N}\left(\|\hat{\boldsymbol{e}}_d^{n+1}\|_0^2 - \|\boldsymbol{e}_d^{n}\|_0^2 + \|\hat{\boldsymbol{e}}_d^{n+1} - \boldsymbol{e}_d^{n}\|_0^2\right) + \sum_{n=0}^{N}\nu\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2\\ &\qquad \leq C(\tilde{\mathcal{I}}(h)^2 + \delta t^2) + \frac{6}{8}\sum_{n=0}^{N}\nu\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2 + \frac{1}{3}\sum_{n=0}^{N}\|\hat{\boldsymbol{e}}_d^{n+1} - \boldsymbol{e}_d^{n}\|_0^2\\ &\qquad + C\sum_{n=0}^{N}\delta t\left(\|\boldsymbol{e}_d^{n+1}\|_0^2 + \|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2\right).\end{aligned} \tag{4.48}$$

The proof goes on taking $\delta t(\boldsymbol{v}_h - \boldsymbol{u}_h^{n+1})$ as a test function in (4.47b), getting

$$\begin{aligned}&\frac{1}{2}\left(\|\boldsymbol{e}_d^{n+1}\|_0^2 - \|\hat{\boldsymbol{e}}_d^{n+1}\|_0^2 + \|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2\right) + \delta t(\nabla r_d^{n+1}, \boldsymbol{e}_d^{n+1})\\ &\quad = (\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}, \boldsymbol{w}^{n+1} - \boldsymbol{v}_h) + \delta t(\nabla r_d^{n+1}, \boldsymbol{w}^{n+1} - \boldsymbol{v}_h).\end{aligned} \tag{4.49}$$

We bound the first term as follows,

$$(\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}, \boldsymbol{w}^{n+1} - \boldsymbol{v}_h) \leq \frac{1}{6}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2 + C\frac{\delta t}{h^2}\|\boldsymbol{w}^{n+1} - \boldsymbol{v}_h\|_0^2.$$

Using Lemma 4.14 we can bound the second term in the RHS of (4.49),

$$\begin{aligned}(\nabla r_d^{n+1}, \boldsymbol{w}^{n+1} - \boldsymbol{v}_h) &\leq C\|\nabla r_d^{n+1}\|_0 \|\boldsymbol{w}^{n+1} - \boldsymbol{v}_h\|_0 \\ &\leq C(\tilde{\mathcal{I}}_n(h) + \frac{1}{\delta t}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0 + \|\Pi_3(\nabla p_h^{n+1})\|_0)\|\boldsymbol{w}^{n+1} - \boldsymbol{v}_h\|_0 \\ &\leq C\tilde{\mathcal{I}}_n(h)^2 + \frac{1}{6\delta t}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2 + \frac{\delta t}{3}\|\Pi_3(\nabla p_h^{n+1})\|_0^2.\end{aligned}$$

Adding up from $n = 0$ to $n = N$, we get

$$\begin{aligned}&\frac{1}{2}\sum_{n=0}^{N}\left(\|\boldsymbol{e}_d^{n+1}\|_0^2 - \|\hat{\boldsymbol{e}}_d^{n+1}\|_0^2 + \|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2\right) + \sum_{n=0}^{N}\delta t(\nabla r_d^{n+1}, \boldsymbol{e}_d^{n+1}) \\ &\qquad \leq \frac{1}{3}\sum_{n=0}^{N}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2 + C\tilde{\mathcal{I}}(h)^2 + \frac{1}{3}\sum_{n=0}^{N}\delta t^2\|\Pi_3(\nabla p_h^{n+1})\|_0^2. \qquad (4.50)\end{aligned}$$

Now we take $\delta t(q_h - p_h^{n+1})$ as a test function in (4.47c), obtaining

$$\begin{aligned}&\delta t(\nabla\cdot\boldsymbol{e}_d^{n+1}, r_d^{n+1}) + \delta t^2\|\Pi^{\perp}_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1})\|_0^2 \\ &\qquad = \delta t(\nabla\cdot\boldsymbol{e}_d^{n+1}, \phi^{n+1} - q_h) + \delta t^2(\Pi^{\perp}_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1}), \nabla q_h) \\ &\qquad = \delta t(\nabla\cdot\boldsymbol{e}_d^{n+1}, \phi^{n+1} - q_h) - \delta t^2(\Pi^{\perp}_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1}), \boldsymbol{y}_h - \nabla q_h) \qquad (4.51)\end{aligned}$$

for any function $\boldsymbol{y}_h \in \mathcal{Y}_{h,0}$. We bound the first term of the RHS as follows,

$$\begin{aligned}(\nabla\cdot\boldsymbol{e}_d^{n+1}, \phi^{n+1} - q_h) &= -(\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}, \nabla\phi^{n+1} - \nabla q_h) + (\nabla\cdot\hat{\boldsymbol{e}}_d^{n+1}, \phi^{n+1} - q_h) \\ &\leq \frac{1}{8}\nu\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2 + C\|\phi^{n+1} - q_h\|_0^2 + \frac{1}{6\delta t}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2 + C\delta t\|\phi^{n+1} - q_h\|_1^2 \\ &\leq \frac{1}{8}\nu\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2 + C\|\phi^{n+1} - q_h\|_0^2 + \frac{1}{6\delta t}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2 \\ &\quad + Ch^2\|\phi^{n+1} - q_h\|_1^2,\end{aligned}$$

and using the triangular inequality and Young's inequality, we have,

$$\begin{aligned}-(\Pi^{\perp}_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1}), \boldsymbol{y}_h - \nabla q_h)) &\leq \|\Pi^{\perp}_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1})\|_0\|\boldsymbol{y}_h - \nabla q_h\|_0 \\ &\leq \frac{1}{3}\|\Pi^{\perp}_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1})\|_0^2 + C\frac{h^2}{\delta t}\|\nabla\phi^{n+1} - \boldsymbol{y}_h\|_0^2 \\ &\quad + C\frac{h^2}{\delta t}\|\phi^{n+1} - q_h\|_1^2.\end{aligned}$$

Adding up from $n = 0$ to $n = N$, we have

$$\begin{aligned}&\sum_{n=0}^{N}\delta t(\nabla\cdot\boldsymbol{e}_d^{n+1}, r_d^{n+1}) + \sum_{n=0}^{N}\delta t^2\|\Pi^{\perp}_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1})\|_0^2 \\ &\leq \frac{1}{8}\sum_{n=0}^{N}\nu\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2 + \frac{1}{6}\sum_{n=0}^{N}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2 + \sum_{n=0}^{N}\frac{\delta t^2}{3}\|\Pi^{\perp}_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1})\|_0^2 + C\tilde{\mathcal{I}}(h)^2. \qquad (4.52)\end{aligned}$$

Taking into account the fact that $\mathcal{Y}_{h,0} \subset \mathcal{V}_h$, we can easily infer that, for any function $q_h \in \mathcal{Q}_h$,

$$\|\Pi_3(\nabla q_h)\|_0 = \|\Pi_h^{\perp}(\nabla q_h)\|_0 \leq \|\Pi^{\perp}_{\mathcal{Y}_{h,0}}(\nabla q_h)\|_0. \qquad (4.53)$$

Summing up (4.48), (4.50) and (4.52) we arrive to

$$\begin{aligned}
&\frac{1}{2}\|\boldsymbol{e}_d^{N+1}\|_0^2+\frac{1}{2}\sum_{n=0}^{N}\left(\|\hat{\boldsymbol{e}}_d^{n+1}-\boldsymbol{e}_d^n\|_0^2+\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2\right)+\sum_{n=0}^{N}\nu\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2\\
&+\sum_{n=0}^{N}\delta t^2\|\Pi^{\perp}_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1})\|_0^2\leq\frac{1}{3}\sum_{n=0}^{N}\|\hat{\boldsymbol{e}}_d^{n+1}-\boldsymbol{e}_d^n\|_0^2+\frac{7}{8}\sum_{n=0}^{N}\nu\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2+C\sum_{n=0}^{N}\delta t\|\boldsymbol{e}_d^{n+1}\|_0^2\\
&\quad+C\sum_{n=0}^{N}\delta t\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2+\frac{1}{3}\sum_{n=0}^{N}\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2+\frac{2}{3}\sum_{n=0}^{N}\delta t^2\|\Pi^{\perp}_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1})\|_0^2\\
&\quad+C\tilde{\mathcal{I}}(h)^2+C\delta t^2,
\end{aligned}$$

where we have used (4.53). After applying Gronwall Lemma we finally get

$$\begin{aligned}
\|\boldsymbol{e}_d^{N+1}\|_0^2+\sum_{n=0}^{N}\left(\|\hat{\boldsymbol{e}}_d^{n+1}-\boldsymbol{e}_d^n\|_0^2+\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2\right)+\sum_{n=0}^{N}\nu\delta t\|\hat{\boldsymbol{e}}_d^{n+1}\|_1^2\\
+\sum_{n=0}^{N}\delta t^2\|\Pi^{\perp}_{\mathcal{Y}_{h,0}}(\nabla p_h^{n+1})\|_0^2\leq C(\tilde{\mathcal{I}}(h)^2+\delta t^2).
\end{aligned}\tag{4.54}$$

From the first and third term of the left hand side of (4.54) we get the bound for $\|\hat{\boldsymbol{e}}_d^{N+1}\|_0$. The bound for the pressure error is straightforward from Lemma 4.14, (4.53) and (4.54). This completes the proof of the theorem. □

The previous *a priori* error estimate is optimal.

Remark 4.2. The *new* term of the interpolation error function,

$$h\inf_{\boldsymbol{y}_h\in\mathcal{Y}_{h,0}}\|\nabla\phi^{n+1}-\boldsymbol{y}_h\|_0$$

keeps the optimality of the error bound. Nevertheless, from the previous theorem we can infer that, if we decide to seek the end-of-step velocity $\boldsymbol{u}_h^{n+1}$ in the finite element space $\mathcal{V}_{h,0}$ (that is, to impose Dirichlet boundary conditions over $\boldsymbol{u}_h^{n+1}$), the situation gets worse. This is the case of pressure segregation methods obtained at the discrete level (that have been used along this work). *The error estimates for a first order fractional step method obtained at the dicrete level is optimal* (under Assumption 4.8) *only for linear elements.* The problem arises from the fact that now the *new* interpolation term is:

$$h\inf_{\boldsymbol{v}_{h,0}\in\mathcal{V}_{h,0}}\|\nabla\phi^{n+1}-\boldsymbol{v}_h\|_0.$$

That is the approximation of a function in $\boldsymbol{L}^2(\Omega)$ that *does not have to vanish on the boundary* (in fact, its trace is not even defined but only its normal component) by finite element functions $\boldsymbol{v}_h\in\mathcal{V}_{h,0}$ with zero trace. We are not aware of any error estimate for this expression. However, this term is expected to be of order $\mathcal{O}(h)$ from the approximation properties of $\mathcal{V}_h$ and the fact that the difference between $\mathcal{V}_h$ and $\mathcal{V}_{h,0}$ are finite element functions different from zero on the boundary elements and zero at the inner nodes.

We end our analysis with a corollary that gives the convergence results obtained for the fully discrete fractional step method using a Pressure Poisson equation.

Corollary 4.5 (Convergence)**.** *Under the assumptions of Theorem 4.2 and* $\partial_t^3 \boldsymbol{f}(t) \in L^\infty(0,T;\boldsymbol{L}^2(\Omega))$*, the following error estimates hold:*

$$\max_{0\le n\le N}\{\|\boldsymbol{u}(t^{n+1})-\boldsymbol{u}_h^{n+1}\|_0+\|\boldsymbol{u}(t^{n+1})-\hat{\boldsymbol{u}}_h^{n+1}\|_0$$
$$+\sqrt{\delta t}\|\boldsymbol{u}(t^{n+1})-\hat{\boldsymbol{u}}_h^{n+1}\|_1\}+\lambda_{n+1}\sqrt{\delta t}\left(\sum_{n=0}^{N}\delta t\|p(t^{n+1})-p_h^{n+1}\|_0^2\right)^{1/2}\le C(\tilde{\mathcal{I}}(h)+\delta t),$$

for $\delta t>0$ *small enough.*

4.7 An alternative convergence analysis under the inf-sup condition

4.7.1 Introduction

In this section we analyze a fully discrete projection method relying on the inf-sup condition where the pressure is obtained from a Stokes problem. In this analysis, dual arguments are needed at the discrete level.

For the convergence analysis of this section we assume extra regularity properties:

(R3) $\sqrt{t}\boldsymbol{u}_{tt}\in L^2(0,T;\boldsymbol{H}^{-1}(\Omega))$.

(R4) $\boldsymbol{u}_{tt}\in L^2(0,T;\boldsymbol{J}_1')$.

Assumption (R4) is automatically satisfied from (see [147])

$$\boldsymbol{f}_t\in L^2(0,T;\boldsymbol{J}_0).$$

We obtain optimal convergence results for the projection method (4.38) with a velocity-pressure interpolation satisfying the inf-sup condition. Furthermore, the end-of-step velocity and the pressure are solved in a coupled way. It implies that the end-of-step velocity is weakly divergence free.

Let us introduce an extra assumption that will be used below.

Assumption 4.10. *We assume that the regularity properties (R3) and (R4) hold.*

The proof of convergence for this method can be understood as a discrete version of the analysis in [147]. First, we compare the fully discrete problem with the continuous Navier-Stokes equation, obtaining a suboptimal error estimate. Afterwards, this estimate is improved introducing an auxiliary problem that allows us to weaken norms at the discrete level. In the second step we rely on a fully discrete monolithic problem for which optimal error estimates hold.

The alternative approach we pursue in this section is more elegant than the previous one. Nevertheless, as stressed in Remark 4.3, we can not extend this strategy to the pressure Poisson equation case. Furthermore, the results of the previous sections are sharper.

4.7.2 A first error estimate

Let us denote the total velocity and pressure errors by

$$\hat{\boldsymbol{e}}_t^{n+1}=\boldsymbol{u}(t^{n+1})-\hat{\boldsymbol{u}}_h^{n+1},$$
$$\boldsymbol{e}_t^{n+1}=\boldsymbol{u}(t^{n+1})-\boldsymbol{u}_h^{n+1}.$$

We also use the following notation for the interpolation error:

$$\begin{aligned}\mathcal{I}_n(h) &= \frac{1}{h}\inf_{v_h\in\mathcal{V}_h}\|\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h\|_0+\inf_{v_h\in\mathcal{V}_h}\|\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h\|_1\\ &\quad+\inf_{q_h\in\mathcal{Q}_h}\|p(t^{n+1})-q_h\|_0+h\inf_{q_h\in\mathcal{Q}_h}\|p(t^{n+1})-q_h\|_1\\ \mathcal{I}(h) &= \max_{0\le n\le N}\mathcal{I}_n(h)\end{aligned}\tag{4.55}$$

Under the discrete inf-sup condition, optimal order of approximation both in $H_0^1(\Omega)$ and in $L^2(\Omega)$ of solenoidal vector fields can be achieved by means of discretely divergence free finite element functions, that is, $\boldsymbol{w}_h\in\ker B_h$. This is what Lemma 4.13 states.

A first error estimate showing that $\{\hat{\boldsymbol{u}}_h^n\}_{n=1}^N$ and $\{\boldsymbol{u}_h^n\}_{n=1}^N$ are half order approximations of $\boldsymbol{u}$ in $\ell^\infty(\boldsymbol{L}^2(\Omega))\cap\ell^2(\boldsymbol{H}^1(\Omega))$ is stablished in the following theorem.

Theorem 4.3. *Being valid Assumptions 4.1, 4.10, 4.4, 4.8 and 4.5 we have*

$$\begin{aligned}&\max_{0\le n\le N}\{\|\boldsymbol{e}_t^{N+1}\|_0^2+\|\hat{\boldsymbol{e}}_t^{N+1}\|_0^2\}+\sum_{n=0}^{N}\nu\delta t\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2\\ &\qquad+\sum_{n=0}^{N}\{\|\boldsymbol{e}_t^{n+1}-\hat{\boldsymbol{e}}_t^{n+1}\|_0^2+\|\hat{\boldsymbol{e}}_t^{n+1}-\boldsymbol{e}_t^{n+1}\|_0^2\}\le C(\delta t+\mathcal{I}(h)^2(1+\delta t^{-1}))\end{aligned}$$

for $\delta t>0$ small enough.

Proof. This proof is similar to that of [147]. But in this case interpolation errors have to be accounted for. We denote by $\boldsymbol{R}^n$ the truncation error defined by

$$\begin{aligned}\frac{1}{\delta t}(\boldsymbol{u}(t^{n+1})-\boldsymbol{u}(t^n))-\nu\Delta\boldsymbol{u}(t^{n+1})+\boldsymbol{u}(t^{n+1})\cdot\nabla\boldsymbol{u}(t^{n+1})+\nabla p(t^{n+1})&\\ =\boldsymbol{f}(t^{n+1})+\boldsymbol{R}^n,&\end{aligned}\tag{4.56}$$

so that

$$\boldsymbol{R}^n=\frac{1}{\delta t}\int_{t^n}^{t^{n+1}}(t-t^n)\boldsymbol{u}_{tt}(t)\,\mathrm{d}t.\tag{4.57}$$

Let us to obtain the system that governs the error functions. Multiplying (4.56) by $\boldsymbol{v}_h\in\mathcal{V}_{h,0}$ we obtain, after integrating by parts,

$$\begin{aligned}\frac{1}{\delta t}(\boldsymbol{u}(t^{n+1})-\boldsymbol{u}(t^n),\boldsymbol{v}_h)-\nu(\nabla\boldsymbol{u}(t^{n+1}),\nabla\boldsymbol{v}_h)+c(\boldsymbol{u}(t^{n+1}),\boldsymbol{u}(t^{n+1}),\boldsymbol{v}_h)+(\nabla p(t^{n+1}),\boldsymbol{v}_h)&\\ =\langle\boldsymbol{f}(t^{n+1}),\boldsymbol{v}_h\rangle+\langle\boldsymbol{R}^n,\boldsymbol{v}_h\rangle&\end{aligned}\tag{4.58}$$

Now we subtract (4.58) from (4.38a), getting the first equation of the error system,

$$\begin{aligned}&\frac{1}{\delta t}(\hat{\boldsymbol{e}}_t^{n+1}-\boldsymbol{e}_t^n,\boldsymbol{v}_h)+\nu(\nabla\hat{\boldsymbol{e}}_t^{n+1},\nabla\boldsymbol{v}_h)\\ &\quad=c(\hat{\boldsymbol{u}}_h^{n+1},\hat{\boldsymbol{u}}_h^{n+1},\boldsymbol{v}_h)-c(\boldsymbol{u}(t^{n+1}),\boldsymbol{u}(t^{n+1}),\boldsymbol{v}_h)+\langle\boldsymbol{R}^n,\boldsymbol{v}_h\rangle-(\nabla p(t^{n+1}),\boldsymbol{v}_h).\end{aligned}\tag{4.59a}$$

The second equation is easily obtained adding and subtracting $(\frac{1}{\delta t}\boldsymbol{u}(t^{n+1}),\boldsymbol{y}_h)$ in (4.38b),

$$\frac{1}{\delta t}(\boldsymbol{e}_t^{n+1}-\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{y}_h)+(\nabla p_h^{n+1},\boldsymbol{y}_h)=0.\tag{4.59b}$$

Due to the fact that $\boldsymbol{u}(t^{n+1}) \in \boldsymbol{J}_1$, we have

$$(\nabla \cdot \boldsymbol{e}_t^{n+1}, q_h) = 0. \tag{4.59c}$$

We now take $\delta t(\boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1})$ as a test function in (4.59a), obtaining

$$\begin{aligned}
&\frac{1}{2}\left(\|\hat{\boldsymbol{e}}_t^{n+1}\|_0^2 - \|\boldsymbol{e}_t^n\|_0^2 + \|\hat{\boldsymbol{e}}_t^{n+1} - \boldsymbol{e}_t^n\|_0^2\right) + \nu\delta t\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2 \\
&\quad = \delta t c(\hat{\boldsymbol{u}}_h^{n+1}, \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}) - \delta t c(\boldsymbol{u}(t^{n+1}), \boldsymbol{u}(t^{n+1}), \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}) \\
&\qquad + (\hat{\boldsymbol{e}}_t^{n+1} - \boldsymbol{e}_t^n, \boldsymbol{u}(t^{n+1}) - \boldsymbol{v}_h) + \nu\delta t(\nabla\hat{\boldsymbol{e}}_t^{n+1}, \nabla\boldsymbol{u}(t^{n+1}) - \nabla\boldsymbol{v}_h) \\
&\qquad + \delta t\langle \boldsymbol{R}^n, \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}\rangle - \delta t(\nabla p(t^{n+1}), \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}).
\end{aligned} \tag{4.60}$$

We can bound the RHS terms of (4.60) related to the truncation error as follows:

$$\begin{aligned}
\delta t\langle \boldsymbol{R}^{n+1}, \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}\rangle &\le \delta t\|\boldsymbol{R}^{n+1}\|_{-1}\|\boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}\|_1 \\
&\le \frac{1}{6}\nu\delta t\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2 + C\delta t\|\boldsymbol{u}(t^{n+1}) - \boldsymbol{v}_h\|_1^2 + C\delta t\|\boldsymbol{R}^{n+1}\|_{-1}^2 \\
&\le \frac{1}{6}\nu\delta t\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2 + C\delta t\|\boldsymbol{u}(t^{n+1}) - \boldsymbol{v}_h\|_1^2 \\
&\qquad + C\delta t^{-1}\|\int_{t^n}^{t^{n+1}} (t - t^n)\boldsymbol{u}_{tt}\,\mathrm{d}t\|_{-1}^2 \\
&\le \frac{1}{6}\nu\delta t\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2 + C\delta t\|\boldsymbol{u}(t^{n+1}) - \boldsymbol{v}_h\|_1^2 \\
&\qquad + C\delta t^{-1}\int_{t^n}^{t^{n+1}} (t - t^n)\|\boldsymbol{u}_{tt}\|_{-1}^2\,\mathrm{d}t \int_{t^n}^{t^{n+1}} (t - t^n)\,\mathrm{d}t \\
&\le \frac{1}{6}\nu\delta t\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2 + C\delta t\|\boldsymbol{u}(t^{n+1}) - \boldsymbol{v}_h\|_1^2 \\
&\qquad + C\delta t\int_{t^n}^{t^{n+1}} t\|\boldsymbol{u}_{tt}\|_{-1}^2\,\mathrm{d}t.
\end{aligned}$$

The pressure gradient term yields

$$\begin{aligned}
-\delta t(\nabla p(t^{n+1}), \boldsymbol{v}_h - \hat{\boldsymbol{u}}_h^{n+1}) =& -\delta t(\nabla p(t^{n+1}), \hat{\boldsymbol{e}}_t^{n+1} - \boldsymbol{e}_t^n) + \delta t(\nabla p(t^{n+1}), \boldsymbol{u}(t^{n+1}) - \boldsymbol{v}_h) \\
&+ \delta t(\nabla p(t^{n+1}), \boldsymbol{e}_t^n) \\
=& -\delta t(\nabla p(t^{n+1}), \hat{\boldsymbol{e}}_t^{n+1} - \boldsymbol{e}_t^n) + \delta t(\nabla p(t^{n+1}), \boldsymbol{u}(t^{n+1}) - \boldsymbol{v}_h) \\
&+ \delta t(\nabla p(t^{n+1}) - \nabla q_h^{n+1}, \boldsymbol{e}_t^n) \\
\le& \frac{1}{4}\|\hat{\boldsymbol{e}}_t^{n+1} - \boldsymbol{e}_t^n\|_0^2 + \frac{C\delta t}{h^2}\|\boldsymbol{u}(t^{n+1}) - \boldsymbol{v}_h\|_0^2 \\
&+ C(\delta t^2 + h^2\delta t)\|\nabla p(t^{n+1})\|_0^2 + h^{-2}\delta t\mathcal{I}_n(h) + C\delta t\|\boldsymbol{e}_t^n\|_0^2,
\end{aligned}$$

where we have invoked the fact that $\boldsymbol{e}_t^n$ is discretely divergence-free in the second step.

It is essential to group the nonlinear terms in a proper manner for the obtention of

appropriate bounds. We have

$$\begin{aligned}
&c(\boldsymbol{u}(t^{n+1}),\boldsymbol{u}(t^{n+1}),\boldsymbol{v}_h-\hat{\boldsymbol{u}}_h^{n+1})-c(\hat{\boldsymbol{u}}_h^{n+1},\hat{\boldsymbol{u}}_h^{n+1},\boldsymbol{v}_h-\hat{\boldsymbol{u}}_h^{n+1})\\
&=c(\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{u}(t^{n+1}),\boldsymbol{v}_h-\hat{\boldsymbol{u}}_h^{n+1})+c(\hat{\boldsymbol{u}}_h^{n+1},\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{v}_h-\hat{\boldsymbol{u}}_h^{n+1})\\
&=c(\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{u}(t^{n+1}),\boldsymbol{v}_h-\hat{\boldsymbol{u}}_h^{n+1})+c(\boldsymbol{u}(t^{n+1}),\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{v}_h-\hat{\boldsymbol{u}}_h^{n+1})\\
&-c(\hat{\boldsymbol{e}}_t^{n+1},\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{v}_h-\hat{\boldsymbol{u}}_h^{n+1})\\
&=c(\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{u}(t^{n+1}),\hat{\boldsymbol{e}}_t^{n+1})-c(\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{u}(t^{n+1}),\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h)\\
&+c(\boldsymbol{u}(t^{n+1}),\hat{\boldsymbol{e}}_t^{n+1},\hat{\boldsymbol{e}}_t^{n+1})-c(\boldsymbol{u}(t^{n+1}),\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h)\\
&-c(\hat{\boldsymbol{e}}_t^{n+1},\hat{\boldsymbol{e}}_t^{n+1},\hat{\boldsymbol{e}}_t^{n+1})+c(\hat{\boldsymbol{e}}_t^{n+1},\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h),
\end{aligned}$$

and now we can bound each term using the fact that $\|\boldsymbol{u}(t^{n+1})\|_2$ is bounded:

$$\begin{aligned}
c(\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{u}(t^{n+1}),\hat{\boldsymbol{e}}_t^{n+1}) &\le C\|\hat{\boldsymbol{e}}_t^{n+1}\|_0\|\boldsymbol{u}(t^{n+1})\|_2\|\hat{\boldsymbol{e}}_t^{n+1}\|_1\\
&\le C\|\hat{\boldsymbol{e}}_t^{n+1}\|_0^2+\frac{1}{6}\nu\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2\\
&\le C\|\boldsymbol{e}_t^{n+1}\|_0^2+C\|\boldsymbol{e}_t^{n+1}-\hat{\boldsymbol{e}}_t^{n+1}\|_0^2+\frac{1}{6}\nu\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2,\\
-c(\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{u}(t^{n+1}),\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h) &\le C\|\hat{\boldsymbol{e}}_t^{n+1}\|_0\|\boldsymbol{u}(t^{n+1})\|_2\|\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h\|_1\\
&\le C\|\boldsymbol{e}_t^{n+1}\|_0^2+C\|\boldsymbol{e}_t^{n+1}-\hat{\boldsymbol{e}}_t^{n+1}\|_0^2+C\|\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h\|_1^2,\\
c(\boldsymbol{u}(t^{n+1}),\hat{\boldsymbol{e}}_t^{n+1},\hat{\boldsymbol{e}}_t^{n+1}) &= 0,\\
-c(\boldsymbol{u}(t^{n+1}),\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h) &\le \|\boldsymbol{u}(t^{n+1})\|_2\|\hat{\boldsymbol{e}}_t^{n+1}\|_0\|\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h\|_1\\
&\le C\|\boldsymbol{e}_t^{n+1}\|_0^2+C\|\boldsymbol{e}_t^{n+1}-\hat{\boldsymbol{e}}_t^{n+1}\|_0^2+C\|\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h\|_1^2,\\
c(\hat{\boldsymbol{e}}_t^{n+1},\hat{\boldsymbol{e}}_t^{n+1},\hat{\boldsymbol{e}}_t^{n+1}) &= 0,\\
-c(\hat{\boldsymbol{e}}_t^{n+1},\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h) &\le C\|\hat{\boldsymbol{e}}_t^{n+1}\|_1\|\hat{\boldsymbol{e}}_t^{n+1}\|_1\|\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h\|_1\\
&\le \frac{1}{6}\nu\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2+C\nu\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2\|\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h\|_1^2.
\end{aligned}$$

Using these bounds in (4.60), we get

$$\begin{aligned}
&\frac{1}{2}(\|\hat{\boldsymbol{e}}_t^{n+1}\|_0^2-\|\boldsymbol{e}_t^{n}\|_0^2+\frac{1}{2}\|\hat{\boldsymbol{e}}_t^{n+1}-\boldsymbol{e}_t^{n}\|_0^2)+\frac{1}{2}\delta t\nu\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2\\
&\quad\le C\delta t\|\boldsymbol{e}_t^{n+1}\|_0^2+C\delta t\|\boldsymbol{e}_t^{n+1}-\hat{\boldsymbol{e}}_t^{n+1}\|_0^2\\
&\qquad+\frac{1}{4}\|\hat{\boldsymbol{e}}_t^{n+1}-\boldsymbol{e}_t^{n}\|_0^2+C\delta t\int_{t^n}^{t^{n+1}} t\|\boldsymbol{u}_{tt}\|_{-1}^2\,\mathrm{d}t+Ch^{-2}\delta t\mathcal{I}_n(h)\\
&\qquad+C(\delta t^2+h^2\delta t)\|\nabla p(t^{n+1})\|_0^2+C\delta t\mathcal{I}_n(h)^2+C\delta t\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2\mathcal{I}_n(h)^2. \qquad (4.61)
\end{aligned}$$

Taking $\delta t(\boldsymbol{w}_h-\boldsymbol{u}_h^{n+1})$ as test function in (4.59b), where $\boldsymbol{w}_h\in\ker B_h$, we obtain:

$$\frac{1}{2}\left(\|\boldsymbol{e}_t^{n+1}\|_0^2-\|\hat{\boldsymbol{e}}_t^{n+1}\|_0^2+\|\boldsymbol{e}_t^{n+1}-\hat{\boldsymbol{e}}_t^{n+1}\|_0^2\right)=(\boldsymbol{e}_t^{n+1}-\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{u}(t^{n+1})-\boldsymbol{w}_h). \qquad (4.62)$$

The optimal approximation properties obtained in the Lemma 4.13 for functions $\boldsymbol{w}_h\in\ker B_h$ and Assumption 4.8 allow us to get the following bound for the RHS,

$$\begin{aligned}
(\boldsymbol{e}_t^{n+1}-\hat{\boldsymbol{e}}_t^{n+1},\boldsymbol{u}(t^{n+1})-\boldsymbol{w}_h) &\le \frac{1}{4}\|\boldsymbol{e}_t^{n+1}-\hat{\boldsymbol{e}}_t^{n+1}\|_0^2+C\frac{\delta t}{h^2}\|\boldsymbol{u}(t^{n+1})-\boldsymbol{w}_h\|_0^2\\
&\le \frac{1}{4}\|\boldsymbol{e}_t^{n+1}-\hat{\boldsymbol{e}}_t^{n+1}\|_0^2+C\delta t\|\boldsymbol{u}(t^{n+1})-\boldsymbol{v}_h\|_1^2,
\end{aligned}$$

for any $\boldsymbol{v}_h \in \mathcal{V}_{h,0}$. Using this in (4.62) we get

$$\frac{1}{2}\left(\|\boldsymbol{e}_t^{n+1}\|_0^2 - \|\hat{\boldsymbol{e}}_t^{n+1}\|_0^2\right) + \frac{1}{4}\|\boldsymbol{e}_t^{n+1} - \hat{\boldsymbol{e}}_t^{n+1}\|_0^2 \leq C\delta t \mathcal{I}_n(h)^2. \tag{4.63}$$

Adding (4.63) to (4.61), we arrive to

$$\begin{aligned}
&\frac{1}{2}(\|\boldsymbol{e}_t^{n+1}\|_0^2 - \|\boldsymbol{e}_t^{n}\|_0^2 + \frac{1}{2}\|\hat{\boldsymbol{e}}_t^{n+1} - \boldsymbol{e}_t^{n}\|_0^2 + \frac{1}{2}\|\boldsymbol{e}_t^{n+1} - \hat{\boldsymbol{e}}_t^{n+1}\|_0^2) + \frac{1}{2}\delta t\nu\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2 \\
&\quad \leq C\delta t\|\boldsymbol{e}_t^{n+1}\|_0^2 + C\delta t\|\boldsymbol{e}_t^{n+1} - \hat{\boldsymbol{e}}_t^{n+1}\|_0^2 + C\delta t\int_{t^n}^{t^{n+1}} t\|\boldsymbol{u}_{tt}\|_{-1}^2 \,\mathrm{d}t \\
&\qquad + C(\delta t^2 + h^2\delta t)\|\nabla p(t^{n+1})\|_0^2 + C(1+h^{-2})\delta t\mathcal{I}_n(h)^2 + C\delta t\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2\mathcal{I}_n(h)^2.
\end{aligned} \tag{4.64}$$

Adding up from $n = 0$ to $n = N$, invoking the bounds assumed for the time step size δt and using the discrete Gronwall lemma (see [98]) we get,

$$\begin{aligned}
\|\boldsymbol{e}_t^{N+1}\|_0^2 + \sum_{n=0}^{N}\|\boldsymbol{e}_t^{n+1} - \hat{\boldsymbol{e}}_t^{n+1}\|_0^2 + \sum_{n=0}^{N}\|\hat{\boldsymbol{e}}_t^{n+1} - \boldsymbol{e}_t^{n}\|_0^2 + \sum_{n=0}^{N}\delta t\nu\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2 \\
\leq C\delta t\int_0^T t\|\boldsymbol{u}_{tt}\|_{-1}^2 \,\mathrm{d}t + C\sum_{n=0}^{N}(\delta t^2 + h^2\delta t)\|\nabla p(t^{n+1})\|_0^2 \\
+ C\mathcal{I}(h)^2(1 + h^{-2} + \sum_{n=0}^{N}\delta t\|\hat{\boldsymbol{e}}_t^{n+1}\|_1^2) \\
\leq C(\mathcal{I}(h)^2(1+h^{-2}) + \delta t + h^2) \\
\leq C(\mathcal{I}(h)^2(1+\delta t^{-1}) + \delta t),
\end{aligned} \tag{4.65}$$

where we have used the stability of $\{\hat{\boldsymbol{u}}_h^n\}_{n=1}^N$ in $\ell^2(\boldsymbol{H}^1(\Omega))$ (see [159]) and the regularity assumptions on the pressure. From the previous bounds we easily find that

$$\|\hat{\boldsymbol{e}}_t^{N+1}\|_0^2 \leq C(\mathcal{I}(h)^2(1+\delta t^{-1}) + \delta t), \tag{4.66}$$

so that the theorem is proved. □

4.7.3 An improved error estimate

At this point we compare the fully discretized fractional step method with a fully discretized monolithic method. The monolithic problem reads as follows,

$$\begin{aligned}
&\frac{1}{\delta t}(\bar{\boldsymbol{u}}_h^{n+1} - \bar{\boldsymbol{u}}_h^{n}, \boldsymbol{v}_h) + \nu(\nabla\bar{\boldsymbol{u}}_h^{n+1}, \nabla\boldsymbol{v}_h) \\
&\qquad + c(\bar{\boldsymbol{u}}_h^{n+1}, \bar{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h) - (\bar{p}_h^{n+1}, \nabla\cdot\boldsymbol{v}_h) = \langle \boldsymbol{f}(t^{n+1}), \boldsymbol{v}_h\rangle, &\text{(4.67a)}\\
&\qquad (\nabla\cdot\bar{\boldsymbol{u}}_h^{n+1}, q_h) = 0, &\text{(4.67b)}
\end{aligned}$$

where $\bar{\boldsymbol{u}}_h^{n+1}, \bar{p}_h^{n+1}$ are the velocity and pressure approximations. For this scheme, a first order approximation can be proved in $\ell^2(0,T;\boldsymbol{L}^2(\Omega)) \cap \ell^\infty(0,T;\boldsymbol{H}^1(\Omega))$ (see [97]). Let us denote the L^2 norm of the error of the fully discrete monolithic system (4.67) as

$$(\mathcal{E}_h^M)^2 = \|\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}_h^{n+1}\|_0^2 \tag{4.68}$$

and the discrete errors,

$$\hat{\boldsymbol{e}}_d^{n+1} = \bar{\boldsymbol{u}}_h^{n+1} - \hat{\boldsymbol{u}}_h^{n+1},$$
$$\boldsymbol{e}_d^{n+1} = \bar{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^{n+1}.$$

The introduction of an auxiliary fully discrete Stokes problem is needed in order to prove the error estimates of this section. Given $\boldsymbol{u} \in \boldsymbol{H}^1(\Omega)$ and $p \in \boldsymbol{L}^2(\Omega)$ let us consider the finite element problem that consists of finding $(\boldsymbol{y}_h, a_h) \in \mathcal{V}_{h,0} \times \mathcal{Q}_h$ such that:

$$\begin{aligned}(\nabla \boldsymbol{y}_h, \nabla \boldsymbol{v}_h) - (a_h, \nabla \cdot \boldsymbol{v}_h) &= (\nabla \boldsymbol{u}, \nabla \boldsymbol{v}_h) - (p, \nabla \cdot \boldsymbol{v}_h)\\ (q_h, \nabla \cdot \boldsymbol{y}_h) &= (q_h, \nabla \cdot \boldsymbol{u}).\end{aligned} \tag{4.69}$$

We collect a useful result extracted from [92] and summarized in the following lemma:

Lemma 4.15. *Under the assumption 4.2, provided $\boldsymbol{u} \in \boldsymbol{H}^2(\Omega)$ and $p \in \boldsymbol{H}^1(\Omega)$, there exists a $C > 0$ such that*

$$\|\boldsymbol{y}_h\|_{0,\infty} + \|\boldsymbol{y}_h\|_{1,3} + \|\nabla a_h\|_0 \leq C(\|\boldsymbol{u}\|_2 + \|\nabla p\|_0).$$

In addition we define the discrete Stokes operator A_h: given $(\boldsymbol{z}_h, \xi_h) \in \mathcal{V}_h \times \mathcal{Q}_h$, $\boldsymbol{e}_h = A_h\{\boldsymbol{z}_h, \xi_h\}$ satisfies

$$\begin{aligned}(\nabla \boldsymbol{z}_h, \nabla \boldsymbol{v}_h) - (\xi_h, \nabla \cdot \boldsymbol{v}_h) &= (\boldsymbol{e}_h, \boldsymbol{v}_h),\\ (q_h, \nabla \cdot \boldsymbol{z}_h) &= 0.\end{aligned} \tag{4.70}$$

Furthermore, we will use the inverse of the discrete Stokes operator, as $\boldsymbol{z}_h = A_h^{-1}\boldsymbol{e}_h$, given a $\boldsymbol{e}_h \in \ker B_h$. It can be easily seen that this operator is self-adjoint. In the next lemma we state some regularity properties of A_h^{-1}.

Lemma 4.16. *Under the assumption 4.2, provided $\boldsymbol{e}_h \in \ker B_h$, there exists a $C > 0$ such that*

$$\|\boldsymbol{z}_h\|_{0,\infty} + \|\boldsymbol{z}_h\|_{1,3} + \|\nabla \xi_h\|_0 \leq C\|\boldsymbol{e}_h\|_0.$$

Proof. It is easily obtained from the previous lemma, with $\{\boldsymbol{u}, p\} = \boldsymbol{A}^{-1}S\boldsymbol{e}_h$ and the $\boldsymbol{H}^2$-regularity of the Stokes operator (1.29),

$$\|\boldsymbol{z}_h\|_{0,\infty} + \|\boldsymbol{z}_h\|_{1,3} + \|\xi_h\|_0 \leq C(\|\boldsymbol{u}\|_2 + \|\nabla p\|_0) \leq C\|\boldsymbol{e}_h\|_0.$$

This proofs the lemma. □

In the next theorem we summarize the error estimates we want to proof. Let us define the following norm: given $\boldsymbol{e}_h \in \ker B_h$, we define the norm

$$\|\boldsymbol{e}_d^{N+1}\|^2_{(\boldsymbol{J}_1)'_h} := (A_h^{-1}\boldsymbol{e}_d^{n+1}, \boldsymbol{e}_d^{n+1}), \tag{4.71}$$

that is the discrete counterpart of (1.30).

Theorem 4.4. *Under the assumptions 4.1, 4.10, 4.4, 4.5, 4.2, 4.8 and 4.7, we have*

$$\max_{0\leq n\leq N}\{\|\boldsymbol{e}_d^{N+1}\|^2_{(\boldsymbol{J}_1)'_h}\} + \sum_{n=0}^{N} \delta t\{\|\hat{\boldsymbol{e}}_d^{n+1}\|_0^2 + \|\boldsymbol{e}_d^{n+1}\|_0^2\} \leq C(\delta t^2 + \mathcal{I}(h)^2) + C(\mathcal{E}_h^M)^2$$

for $\delta t > 0$ small enough.

Proof. We start adding (4.38a) and (4.38b),

$$\frac{1}{\delta t}(\boldsymbol{u}_h^{n+1}-\boldsymbol{u}_h^n,\boldsymbol{v}_h)+\nu(\nabla\hat{\boldsymbol{u}}_h^{n+1},\nabla\boldsymbol{v}_h)+c(\hat{\boldsymbol{u}}_h^{n+1},\hat{\boldsymbol{u}}_h^{n+1},\boldsymbol{v}_h)+(\nabla p_h^{n+1},\boldsymbol{v}_h)$$
$$=\langle\boldsymbol{f}(t^{n+1}),\boldsymbol{v}_h\rangle\quad\forall\boldsymbol{v}_h\in\mathcal{V}_{h,0}. \tag{4.72}$$

This expression can be obtained by invoking Assumption 4.5. Denoting by $r_d^{n+1}=\bar{p}_h^{n+1}-p_h^{n+1}$ the pressure error, we obtain after subtracting (4.72) from (4.67a) and (4.38c) from (4.67b),

$$\frac{1}{\delta t}(\boldsymbol{e}_d^{n+1}-\boldsymbol{e}_d^n,\boldsymbol{v}_h)+\nu(\nabla\hat{\boldsymbol{e}}_d^{n+1},\nabla\boldsymbol{v}_h)+(\nabla r_d^{n+1},\boldsymbol{v}_h)$$
$$=c(\hat{\boldsymbol{u}}_h^{n+1},\hat{\boldsymbol{u}}_h^{n+1},\boldsymbol{v}_h)-c(\bar{\boldsymbol{u}}_h^{n+1},\bar{\boldsymbol{u}}_h^{n+1},\boldsymbol{v}_h). \tag{4.73a}$$
$$(\nabla\cdot\boldsymbol{e}_d^{n+1},q_h)=0 \tag{4.73b}$$

Let $\{\boldsymbol{z}_h,\xi_h\}$ be the solution of the fully discrete Stokes problem (4.70) with $\boldsymbol{e}_h\equiv\boldsymbol{e}_d^{n+1}$. Being A_h the discrete Stokes operator already defined, since $A_h^{-1}\boldsymbol{e}_d^{n+1}\in\ker B_h$, we have that

$$(A_h^{-1}\boldsymbol{e}_d^{n+1},\nabla q_h)=0\quad\forall q_h\in\mathcal{Q}_h. \tag{4.74}$$

We take $\delta tA_h^{-1}\boldsymbol{e}_d^{n+1}\in\mathcal{V}_{h,0}$ as test function in (4.72). It yields

$$(\boldsymbol{e}_d^{n+1}-\boldsymbol{e}_d^n,A_h^{-1}\boldsymbol{e}_d^{n+1})+\nu\delta t(\nabla\hat{\boldsymbol{e}}_d^{n+1},\nabla A_h^{-1}\boldsymbol{e}_d^{n+1})$$
$$=\delta tc(\hat{\boldsymbol{u}}_h^{n+1},\hat{\boldsymbol{u}}_h^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1})-\delta tc(\bar{\boldsymbol{u}}_h^{n+1},\bar{\boldsymbol{u}}_h^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1}). \tag{4.75}$$

Invoking the self-adjointness of A_h^{-1} and due to the fact that $\boldsymbol{e}_d^n\in\ker B_h$, we can express the first term as follows:

$$(\boldsymbol{e}_d^{n+1}-\boldsymbol{e}_d^n,A_h^{-1}\boldsymbol{e}_d^{n+1})=\frac{1}{2}(\|\boldsymbol{e}_d^{n+1}\|^2_{(\boldsymbol{J}_1)'_h}-\|\boldsymbol{e}_d^n\|^2_{(\boldsymbol{J}_1)'_h}+\|\boldsymbol{e}_d^{n+1}-\boldsymbol{e}_d^n\|^2_{(\boldsymbol{J}_1)'_h}).$$

For the second term we get

$$\begin{aligned}(\nabla\hat{\boldsymbol{e}}_d^{n+1},\nabla A_h^{-1}\boldsymbol{e}_d^{n+1})&=(\hat{\boldsymbol{e}}_d^{n+1},\boldsymbol{e}_d^{n+1}-\nabla\xi_h)\\&=(\hat{\boldsymbol{e}}_d^{n+1},\boldsymbol{e}_d^{n+1})+(\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1},\nabla\xi_h)\\&=(\boldsymbol{e}_d^{n+1}+\delta t\nabla p_h^{n+1},\boldsymbol{e}_d^{n+1})+(\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1},\nabla\xi_h)\\&=\|\boldsymbol{e}_d^{n+1}\|_0^2+(\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1},\nabla\xi_h),\end{aligned}$$

and using the regularity properties of A_h, we obtain

$$\begin{aligned}(\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1},\nabla\boldsymbol{\xi}_h)&\le\|\nabla\boldsymbol{\xi}_h\|_0\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0\\&\le\frac{1}{12}\|\boldsymbol{e}_d^{n+1}\|_0^2+C\|\boldsymbol{e}_d^{n+1}-\hat{\boldsymbol{e}}_d^{n+1}\|_0^2.\end{aligned}$$

We group the nonlinear terms in the RHS of (4.75) as follows:

$$\begin{aligned}&c(\hat{\boldsymbol{u}}_h^{n+1},\hat{\boldsymbol{u}}_h^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1})-c(\bar{\boldsymbol{u}}_h^{n+1},\bar{\boldsymbol{u}}_h^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1})\\&=-c(\hat{\boldsymbol{e}}_d^{n+1},\hat{\boldsymbol{u}}_h^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1})-c(\bar{\boldsymbol{u}}_h^{n+1},\hat{\boldsymbol{e}}_d^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1})\\&=c(\hat{\boldsymbol{e}}_d^{n+1},\hat{\boldsymbol{e}}_d^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1})-c(\hat{\boldsymbol{e}}_d^{n+1},\bar{\boldsymbol{u}}_h^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1})-c(\bar{\boldsymbol{u}}_h^{n+1},\hat{\boldsymbol{e}}_d^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1})\\&=c(\hat{\boldsymbol{e}}_d^{n+1},\hat{\boldsymbol{e}}_d^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1})+c(\hat{\boldsymbol{e}}_d^{n+1},\boldsymbol{u}(t^{n+1})-\bar{\boldsymbol{u}}_h^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1})\\&-c(\hat{\boldsymbol{e}}_d^{n+1},\boldsymbol{u}(t^{n+1}),A_h^{-1}\boldsymbol{e}_d^{n+1})+c(\boldsymbol{u}(t^{n+1})-\bar{\boldsymbol{u}}_h^{n+1},\hat{\boldsymbol{e}}_d^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1})\\&-c(\boldsymbol{u}(t^{n+1}),\hat{\boldsymbol{e}}_d^{n+1},A_h^{-1}\boldsymbol{e}_d^{n+1}).\end{aligned}$$

We bound the nonlinear terms (since $\|\boldsymbol{e}_d^{n+1}\|_0 \leq \delta t^{1/2}$ from Theorem 4.3) by invoking the bounds for $A^{-1}\boldsymbol{e}_d^{n+1}$ collected in the Lemma 4.16 and the continuity properties of c listed in (1.24),

$$\begin{aligned}
\delta t c(\hat{\boldsymbol{e}}_d^{n+1}, \hat{\boldsymbol{e}}_d^{n+1}, A_h^{-1}\boldsymbol{e}_d^{n+1}) &\leq C\delta t \|\hat{\boldsymbol{e}}_d^{n+1}\|_0 \|\hat{\boldsymbol{e}}_d^{n+1}\|_1 \\
&\quad \cdot (\|A_h^{-1}\boldsymbol{e}_d^{n+1}\|_{0,\infty} + \|A_h^{-1}\boldsymbol{e}_d^{n+1}\|_{1,3}) \\
&\leq C\delta t \|\hat{\boldsymbol{e}}_d^{n+1}\|_0 \|\hat{\boldsymbol{e}}_d^{n+1}\|_1 \|\boldsymbol{e}_d^{n+1}\|_0 \\
&\leq C\delta t^{3/2} \|\hat{\boldsymbol{e}}_d^{n+1}\|_0 \|\hat{\boldsymbol{e}}_d^{n+1}\|_1 \\
&\leq C\delta t^2 \|\hat{\boldsymbol{e}}_d^{n+1}\|_0^2 + \frac{\nu\delta t}{12}\|\boldsymbol{e}_d^{n+1}\|_0^2 + \frac{\nu\delta t}{12}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2, \\
\delta t c(\hat{\boldsymbol{e}}_d^{n+1}, \boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}_h^{n+1}, A_h^{-1}\boldsymbol{e}_d^{n+1}) &\leq C\delta t \|\hat{\boldsymbol{e}}_d^{n+1}\|_0 \|\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}_h^{n+1}\|_1 \\
&\quad \cdot (\|A_h^{-1}\boldsymbol{e}_d^{n+1}\|_{0,\infty} + \|A_h^{-1}\boldsymbol{e}_d^{n+1}\|_{1,3}) \\
&\leq C\delta t \|\hat{\boldsymbol{e}}_d^{n+1}\|_0 \|\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}_h^{n+1}\|_1 \|\hat{\boldsymbol{e}}_d^{n+1}\|_0 \\
&\leq C\delta t^{3/2} \|\hat{\boldsymbol{e}}_d^{n+1}\|_0 \|\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}_h^{n+1}\|_1 \\
&\leq \frac{\nu\delta t}{12}\|\boldsymbol{e}_d^{n+1}\|_0^2 + \frac{\nu\delta t}{12}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2 \\
&\quad + C\delta t^2 \|\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}_h^{n+1}\|_0^2, \\
\delta t c(\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}_h^{n+1}, \hat{\boldsymbol{e}}_d^{n+1}, A_h^{-1}\boldsymbol{e}_d^{n+1}) &\leq C\delta t \|\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}_h^{n+1}\|_1 \|\hat{\boldsymbol{e}}_d^{n+1}\|_0 \\
&\quad \cdot (\|A_h^{-1}\boldsymbol{e}_d^{n+1}\|_{0,\infty} + \|A_h^{-1}\boldsymbol{e}_d^{n+1}\|_{1,3}), \\
&\leq \frac{\nu\delta t}{12}\|\boldsymbol{e}_d^{n+1}\|_0^2 + \frac{\nu\delta t}{12}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2 \\
&\quad + C\delta t^2 \|\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}_h^{n+1}\|_0^2, \\
\delta t c(\hat{\boldsymbol{e}}_d^{n+1}, \boldsymbol{u}(t^{n+1}), A_h^{-1}\boldsymbol{e}_d^{n+1}) &\leq C\delta t \|\hat{\boldsymbol{e}}_d^{n+1}\|_0 \|\boldsymbol{u}(t^{n+1})\|_2 \|A_h^{-1}\boldsymbol{e}_d^{n+1}\|_1 \\
&\leq C\delta t \|\hat{\boldsymbol{e}}_d^{n+1}\|_0 \|\boldsymbol{e}_d^{n+1}\|_{(\boldsymbol{J}_1)_h'} \\
&\leq \frac{\nu\delta t}{12}\|\boldsymbol{e}_d^{n+1}\|_0^2 + \frac{\nu\delta t}{12}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2 \\
&\quad + C\delta t \|\boldsymbol{e}_d^{n+1}\|_{(\boldsymbol{J}_1)_h'}^2.
\end{aligned}$$

and $c(\boldsymbol{u}(t^{n+1}), \hat{\boldsymbol{e}}_d^{n+1}, A_h^{-1}\boldsymbol{e}_d^{n+1})$ can be bounded exactly as the last term.

Combining all these inequalities we easily get

$$\begin{aligned}
&\frac{1}{2}(\|\boldsymbol{e}_d^{n+1}\|_{(\boldsymbol{J}_1)_h'}^2 - \|\boldsymbol{e}_d^{n}\|_{(\boldsymbol{J}_1)_h'}^2 + \|\boldsymbol{e}_d^{n+1} - \boldsymbol{e}_d^{n}\|_{(\boldsymbol{J}_1)_h'}^2) + \delta t\nu \|\boldsymbol{e}_d^{n+1}\|_0^2 \\
&\leq \frac{\delta t\nu}{2}\|\boldsymbol{e}_d^{n+1}\|_0^2 + \frac{5\delta t\nu}{12}\|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2 + C\delta t^2 \|\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}_h^{n+1}\|_0^2 + C\delta t \|\boldsymbol{e}_d^{n+1}\|_{(\boldsymbol{J}_1)_h'}^2.
\end{aligned}$$

Adding up from $n = 0$ to $n = N$ we obtain

$$\begin{aligned}
&\|\boldsymbol{e}_d^{N+1}\|_{(\boldsymbol{J}_1)_h'}^2 + \sum_{n=0}^{N} \|\boldsymbol{e}_d^{n+1} - \boldsymbol{e}_d^{n}\|_{(\boldsymbol{J}_1)_h'}^2 + \sum_{n=0}^{N} \delta t\nu \|\boldsymbol{e}_d^{n+1}\|_0^2 \\
&\leq C\sum_{n=0}^{N} \|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2 + C\sum_{n=0}^{N} \delta t^2 \|\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}_h^{n+1}\|_0^2 + \sum_{n=0}^{N} C\delta t \|\boldsymbol{e}_d^{n+1}\|_{(\boldsymbol{J}_1)_h'}^2. \qquad (4.76)
\end{aligned}$$

Applying the discrete Gronwall lemma to (4.76) and using the results of Theorem 4.3 we

arrive to

$$\begin{aligned}\|\boldsymbol{e}_d^{N+1}\|^2_{(\boldsymbol{J}_1)'_h} + \sum_{n=0}^{N} \|\boldsymbol{e}_d^{n+1} - \boldsymbol{e}_d^{n}\|^2_{(\boldsymbol{J}_1)'_h} + \sum_{n=0}^{N} \delta t\nu \|\boldsymbol{e}_d^{n+1}\|_0^2 \\ \leq C \sum_{n=0}^{N} \delta t \|\boldsymbol{e}_d^{n+1} - \hat{\boldsymbol{e}}_d^{n+1}\|_0^2 + C \sum_{n=0}^{N} \delta t^2 \|\boldsymbol{u}(t^{n+1}) - \bar{\boldsymbol{u}}_h^{n+1}\|_0^2 \\ \leq C(\delta t^2 + \mathcal{I}(h)^2) + (\mathcal{E}_h^M)^2. \qquad (4.77)\end{aligned}$$

The proof of the theorem has been concluded. □

This theorem allows us to improve the convergence results obtained in Lemma 4.16 for the total errors $\boldsymbol{e}_t^{n+1}$ and $\hat{\boldsymbol{e}}_t^{n+1}$. We summarize them in the next corollary.

Corollary 4.6. *Under the assumptions of Theorem 4.4 the following error estimates hold:*

$$\max_{0\leq n\leq N}\{\|\boldsymbol{e}_t^{N+1}\|^2_{(\boldsymbol{J}_1)'_h}\} + \sum_{n=0}^{N} \delta t\{\|\hat{\boldsymbol{e}}_t^{n+1}\|_0^2 + \|\boldsymbol{e}_t^{n+1}\|_0^2\} \leq C(\delta t^2 + \mathcal{I}(h)^2) + (\mathcal{E}_h^M)^2$$

for $\delta t > 0$ small enough.

Remark 4.3. This strategy can not be applied to the stabilized version of the problem or when using a pressure Poisson equation. It is due to the fact that we have invoked some dual arguments that only hold when the discrete velocity is weakly divergence-free. Unfortunately, the introduction of the pressure Poisson equation or the stabilizing terms affects the continuity equation and the velocity is not weakly divergence-free anymore.

This strategy leads to weaker error estimates than those proved in Corollary 4.4. In Corollary 4.6 we have obtained optimal convergence results for $\{\boldsymbol{u}_h^{n+1}\}_{n=0}^N$ with norm $\ell^\infty(0,T;(\boldsymbol{J}_1)'_h) \cap \ell^2(0,T;\boldsymbol{L}^2(\Omega))$ and for $\{\hat{\boldsymbol{u}}_h^{n+1}\}_{n=0}^N$ with norm $\ell^2(0,T;\boldsymbol{L}^2(\Omega))$, whereas in Section 4.5 we have proved sharper results. In particular, Corollary 4.4 yields optimal error estimates for $\{\boldsymbol{u}_h^{n+1}\}_{n=0}^N$ with norm $\ell^\infty(0,T;\boldsymbol{L}^2(\Omega))$ and $\{\hat{\boldsymbol{u}}_h^{n+1}\}_{n=0}^N$ with norm $\ell^\infty(0,T;\boldsymbol{L}^2(\Omega))$.

4.8 Conclusions

We have obtained *optimal convergence and stability results for a classical first order projection method*. We have analyzed the fully discrete version of the problem. Two different approaches have been pursued.

The first approach is based on the introduction of an auxiliary problem that has allowed the analysis of the final discrete problem. Using this strategy not only optimal error estimates have been obtained under the inf-sup condition, but also using a stabilized finite element formulation and a pressure Poisson equation. As far as we know this is the first attempt for the obtention of convergence results for this problem.

Moreover we have inferred from the numerical analysis that *first order fractional step methods obtained by splitting the discrete system, where Dirichlet boundary conditions are applied over the the end-of-step velocity, keep optimal convergence only for linear elements.* This is the case of the pressure segregation methods used along this work.

We have ended with an alternative analysis of the fully discrete system under the inf-sup condition. This is an extension of the analysis of the semi-discrete (in time) system in [147]. Unfortunately, this approach is not appropriate for the stabilized version.

Chapter 5

Velocity Correction methods based on a Discrete Pressure Poisson Equation

In this Chapter we introduce some pressure segregation methods obtained from a non standard version of the discrete monolithic system, where the continuity equation has been replaced by a pressure Poisson equation obtained at the discrete level. In these methods it is the velocity instead of the pressure the extrapolated unknown. Moreover, predictor corrector schemes are suggested, again motivated by the new monolithic system. Key implementation aspects are discussed. We end with a complete set of numerical examples in order to compare these methods with well settled monolithic and pressure correction schemes. This work is the body of [7].

5.1 Introduction

At the continuous level, from the Navier-Stokes equations for incompressible flows –the momentum and continuity equation– a pressure Poisson equation (PPE from now onwards) can be obtained. Then, an alternative system to the classical Navier-Stokes equations can be the momentum equation together with the PPE instead of the continuity equation. Unfortunately, the numerical approximation of the PPE equation is an involved task due to the presence of third order derivatives for the velocity field. In order to facilitate the numerical approximation, a *simplified* PPE could be considered by assuming that the divergence and the Laplace operators commute and deleting the viscous contribution to the PPE. Nevertheless, as it is shown by Gresho and Sani in [82], this alternative form is not equivalent to the Navier-Stokes equations. Therein it is shown that the system with the *simplified* PPE admits spurious solutions that are not solution of the original Navier-Stokes equations.

In this chapter we explore an alternative version of the fully discretized monolithic system that has been obtained at the discrete level. As for the continuous problem, from the discrete momentum and continuity equation a discrete pressure Poisson equation (DPPE from now onwards) can be easily obtained. This leads to a new fully discrete monolithic system composed by the DPPE and the discrete momentum equation. However, it is not our interest to solve this DPPE monolithic problem.

In this chapter we are interested in the development of new pressure segregation methods. We classify these methods in two families: pressure correction methods (treated in

Chapter 3) and velocity correction methods.

The concept of pressure correction schemes appeared when trying to obtain second order accurate projection methods. In [162] Van Kan introduced a second order pressure correction method where the pressure was extrapolated at the momentum equation with the value of the previous time step. In fact, the classical Chorin-Temam projection method can be understood as a pressure correction scheme with a zero order extrapolation of the pressure.

The appearance of methods where it is the velocity instead of the pressure the extrapolated unknown is very recent. In [94] Guermond and Shen have introduced the velocity correction methods (VC from now onwards).

Different VC schemes can be easily obtained at the discrete level from the DPPE monolithic system. We propose a method where the pressure is obtained from the DPPE using an extrapolated velocity. After it is computed, this pressure is used in the momentum equation in order to obtain the velocity.

The DPPE system matrix is a cumbersome matrix. Then, a further approximation is suggested in order to make the method more appealing from a computational point of view. This approximation can be understood as an improved version of the one widely used for pressure correction methods (see [47]).

From pressure correction methods, predictor corrector methods can be easily obtained (see [56]). Here we design a predictor corrector method where all the terms are motivated from the DPPE monolithic system. This is a main difference compared with some predictor corrector methods based on pressure correction schemes (see Chapter 3) .

The stabilized version of velocity correction schemes using the orthogonal subscale method (see Chapter 2) has been studied and some possibilities discussed.

We have proved some stability bounds for different schemes. We have compared the inherent pressure stability of VC methods with that of pressure correction methods.

The outline of the chapter is given by the following. In Section 5.2 some preliminaries for the continuous and discrete problems are stated for the monolithic system. In Section 5.3, velocity correction methods are introduced and discussed. Section 5.4 is devoted to a new family of predictor corrector methods. The stabilized version of VC schemes is introduced in Section 5.5. In Section 5.6 some implementation aspects are discussed. Section 5.7 is devoted to the numerical analysis of VC methods. Stability results for different cases are obtained. The important point is that no reference at all is made to compatibility conditions between the velocity and pressure interpolating spaces, that is, they do not need to satisfy any inf-sup condition. Section 5.8 presents a set of numerical tests that show the behavior of these new schemes in comparison with the monolithic system and more classical pressure correction methods and, finally, the chapter concludes in Section 5.9 with some final remarks.

5.2 Preliminaries and problem statement

5.2.1 The continuous problem

Let Ω be the domain of $\mathbb{R}^d$ occupied by the fluid, where $d = 2$ or 3 is the number of space dimensions, $\Gamma = \partial\Omega$ its boundary and $[0, T]$ the time interval of analysis. The

Navier-Stokes problem consists of finding a velocity $\boldsymbol{u}$ and a pressure p such that

$$\partial_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nabla p = \boldsymbol{f} \quad \text{in } \Omega,\ t \in (0,T), \tag{5.1a}$$
$$\nabla \cdot \boldsymbol{u} = 0 \quad \text{in } \Omega,\ t \in (0,T), \tag{5.1b}$$
$$\boldsymbol{u} = \boldsymbol{0} \quad \text{on } \Gamma,\ t \in (0,T), \tag{5.1c}$$
$$\boldsymbol{u} = \boldsymbol{u}^0 \quad \text{in } \Omega,\ t = 0, \tag{5.1d}$$

where ν is the kinematic viscosity, $\boldsymbol{f}$ is the force vector and $\boldsymbol{u}^0$ is the velocity initial condition. We have considered the homogeneous Dirichlet boundary condition (5.1c) for simplicity.

Alternative forms of the Navier-Stokes equations where the continuity equation (5.1b) is replaced by a new equation that replaces the mass conservation could be considered. A pressure Poisson equation (PPE) that implies mass conservation can be obtained by taking the divergence of the momentum equation (5.1a) and invoking (5.1c), leading to the system

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \nabla p = \boldsymbol{f}, \tag{5.2a}$$
$$\Delta p = \nabla \cdot (\boldsymbol{f} + \nu \Delta \boldsymbol{u} - \boldsymbol{u} \cdot \nabla \boldsymbol{u}). \tag{5.2b}$$

The pressure boundary condition is obtained by imposing that the normal component of the pressure gradient is equal to the normal component of the term within parenthesis in the right hand side of (5.2b). As commented below, the term $\nabla \cdot (\nu \Delta \boldsymbol{u})$ complicates the finite element approximation of the PPE equation. We can extract this term from the equation by commuting operators, leading to a *simplified* PPE:

$$\Delta p = \nabla \cdot (\boldsymbol{f} - \boldsymbol{u} \cdot \nabla \boldsymbol{u}). \tag{5.3}$$

Unfortunately, this equation is ill-posed. It can be shown that (5.2a)-(5.3) admits extra *spurious* solutions that do not satisfy the original Navier-Stokes equations. These *spurious* solutions satisfy

$$\partial_t (\nabla \cdot \boldsymbol{u}) - \nu \Delta (\nabla \cdot \boldsymbol{u}) = \boldsymbol{0}, \tag{5.4}$$

which does not necessarily imply $\nabla \cdot \boldsymbol{u} = 0$ for $t \in (0,T)$. Thus, the false equivalence between the original continuity equation (5.1b) and the simplified PPE obtained at the continuous level makes (5.2a)-(5.3) an inappropriate alternative to the original Navier-Stokes equations. In [82] there is a deep study of the problems arising when using the simplified PPE.

5.2.2 Weak form

For the evolutionary Navier-Stokes equations let $(\mathcal{V}_0)_t \equiv L^2(0,T;\mathcal{V}_0)$ and $\mathcal{Q}_t \equiv L^1(0,T;\mathcal{Q})$ with $\mathcal{V}_0 \equiv \boldsymbol{H}_0^1(\Omega)$ and $\mathcal{Q} \equiv \boldsymbol{L}^2(\Omega)/\mathbb{R}$ denote the spaces for velocity and pressure, respectively. Then, the weak form of (5.1) consists of finding e.g., $\boldsymbol{u} \in (\mathcal{V}_0)_t$ and $p \in \mathcal{Q}_t$ such that:

$$(\partial_t \boldsymbol{u}, \boldsymbol{v}) + \nu(\nabla \boldsymbol{u}, \nabla \boldsymbol{v}) + (\boldsymbol{u} \cdot \nabla \boldsymbol{u}, \boldsymbol{v}) + \langle \nabla p, \boldsymbol{v} \rangle = \langle \boldsymbol{f}, \boldsymbol{v} \rangle, \qquad \forall \boldsymbol{v} \in \mathcal{V}_0, \tag{5.5a}$$
$$\langle \nabla q, \boldsymbol{u} \rangle = 0, \qquad \forall q \in \mathcal{Q}. \tag{5.5b}$$

We assume the force vector belonging to $\boldsymbol{H}^{-1}(\Omega)$.

To pose the weak form of problem (5.2), we also need $(\tilde{\mathcal{V}}_0)_t \equiv \boldsymbol{L}^2(0,T;\tilde{\mathcal{V}}_0)$ and $\tilde{\mathcal{Q}}_t \equiv L^2(0,T;\tilde{\mathcal{Q}})$ with $\tilde{\mathcal{V}}_0 \equiv \boldsymbol{H}^1_0(\Omega)\cap\boldsymbol{H}^2(\Omega)$ and $\tilde{\mathcal{Q}} \equiv H^1(\Omega)/\mathbb{R}$. Now we assume the force vector belonging to $\boldsymbol{L}^2(\Omega)$. The weak form consists of finding $(\boldsymbol{u},p)\in(\tilde{\mathcal{V}}_0))_t\times(\tilde{\mathcal{Q}}))_t$ such that:

$$(\partial_t\boldsymbol{u},\boldsymbol{v})+\nu(\nabla\boldsymbol{u},\nabla\boldsymbol{v})+(\boldsymbol{u}\cdot\nabla\boldsymbol{u},\boldsymbol{v})+(\nabla p,\boldsymbol{v})=\langle\boldsymbol{f},\boldsymbol{v}\rangle,\qquad\forall\boldsymbol{v}\in\tilde{\mathcal{V}}_0,\tag{5.6a}$$

$$(\nabla p,\nabla q)=(\boldsymbol{f}-\boldsymbol{u}\cdot\nabla\boldsymbol{u}+\nu\Delta\boldsymbol{u},\nabla q),\qquad\forall q\in\tilde{\mathcal{Q}}.\tag{5.6b}$$

We could recover the regularity requirements of the previous formulation by taking the pressure test function in $H^2(\Omega)$. However, either this choice or (5.6b) imply the need to construct finite element approximations to $H^2(\Omega)$, which is an involved task (at least for conforming interpolations, as we consider throughout). Thus, the finite element discretization of the pressure Poisson equation (5.6b) will not be studied. We could try to circumvent the regularity assumptions neglecting the diffusive term in (5.6b). Unfortunately, the simplified Poisson equation is ill-posed, as commented above.

5.2.3 Discrete problem

In this section we study the discretization of (5.1). For the time discretization we propose the trapezoidal rule and backward differencing schemes (BDF). For the space approximation the standard Galerkin finite element method. However, no reference will be made to the need to satisfy any compatibility condition between the velocity and pressure finite element spaces. The introduction of stabilization techniques is deferred until Section 5.5, where a stabilization technique designed to allow equal velocity-pressure interpolation and convection-dominated flows will be presented.

Monolithic time discretization

The backward Euler method will be used for first order pressure correction methods, whereas for second order methods we will consider both the Crank-Nicolson and backward differencing schemes. We will refer to the first and second order backward differencing schemes as BDF1 (which coincides with backward Euler) and BDF2, respectively. These methods and convenient notation have been introduced in Chapter 1.

The time integration of the discretized (in space) Navier–Stokes equations (1.43) using the generalized trapezoidal rule leads to problem (1.48). In this section the values of θ that we will take are $\theta=1/2$, corresponding to the second order Crank-Nicolson scheme, and $\theta=1$, which corresponds to the backward Euler method.

BDF2 time integration schemes will be also adopted for second order velocity correction schemes. In this case we obtain the fully discrete system (1.51).

Finally, we will consider the BDF3 scheme together with appropriate extrapolations in order to design a third order velocity correction method.

Finite element discretization

Let us recall the notation introduced in Section 1.3. Let Θ_h denote a finite element partition of the domain Ω of diameter h, from which we construct the finite element spaces $\mathcal{Q}_h$ and $\mathcal{V}_{h,0}$, approximations to $\mathcal{Q}$ and $\mathcal{V}_0$, respectively. The former is made up with continuous functions of degree k_{q} and the other with continuous vector functions of degree k_{v} verifying the homogeneous Dirichlet boundary conditions. In the following, finite element functions will be identified with a subscript h.

The discrete problem is obtained by approximating $\boldsymbol{u}$ and p. We assume that $\boldsymbol{u}_h^n$ and p_h^n are constructed using the standard finite element interpolation from the nodal values. These are solution of the nonlinear algebraic system

$$\mathsf{M}\frac{1}{\delta t}D_k\mathsf{U}^{n+1} + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{5.7a}$$

$$\mathsf{D}\mathsf{U}^{n+1} = \mathsf{0}, \tag{5.7b}$$

where U^{n+1} and P^{n+1} are the arrays of nodal values for $\boldsymbol{u}_h^{n+1}$ and p_h^{n+1}, respectively, obtained from the monolithic system at the time step $n+1$. Let us recall the matrices needed for the fully discrete problem already defined in Chapter 3. If we denote the node indexes with superscripts a, b, the space indexes with subscripts i, j, and the standard shape function of node a by N^a, the components of the arrays involved in these equations are:

$$\mathsf{M}_{ij}^{ab} = (N^a, N^b)\delta_{ij} \quad (\delta_{ij} \text{ is the Kronecker } \delta),$$

$$\mathsf{K}(\mathsf{U}^{n+\theta})_{ij}^{ab} = (N^a, \boldsymbol{u}_h^{n+\theta}\cdot\nabla N^b)\delta_{ij} + \frac{1}{2}\left(N^a, (\nabla\cdot\boldsymbol{u}_h^{n+\theta})N^b\right)\delta_{ij} + \nu(\nabla N^a, \nabla N^b)\delta_{ij},$$

$$\mathsf{G}_i^{ab} = (N^a, \partial_i N^b),$$

$$\mathsf{D}_j^{ab} = (N^a, \partial_j N^b),$$

$$\mathsf{F}_i^a = \langle N^a, f_i\rangle.$$

It is understood that all the arrays are matrices (except F, which is a vector) whose components are obtained by grouping together the left indexes in the previous expressions (a and possibly i) and the right indexes (b and possibly j). Likewise, (5.7) need to be modified to account for the Dirichlet boundary conditions (matrix G can be replaced by $-\mathsf{D}^{\mathrm{t}}$ when this is done). Observe also that we have used the skew-symmetric form of the convective term, which yields the convective contribution to matrix $\mathsf{K}(\mathsf{U}^{n+\theta})$ skew-symmetric.

A Discrete Pressure Poisson Equation

At the continuous level the PPE is not an appropriate equation to be discretized as it has been commented above. Here we propose a discrete pressure Poisson equation (DPPE) obtained in a matrix setting from the fully discretized form of the classical incompressible Navier-Stokes equations, after multiplying (5.7a) by $\delta t\mathsf{D}\mathsf{M}^{-1}$ and invoking (5.7b) in the resulting equation. Replacing the discrete continuity equation (5.7b) by the DPPE, the equivalent monolithic scheme is

$$\mathsf{M}\frac{1}{\delta t}D_k\mathsf{U}^{n+1} + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{5.8a}$$

$$\gamma_k\delta t\mathsf{D}\mathsf{M}^{-1}\mathsf{G}\mathsf{P}^{n+1} = \gamma_k\delta t\mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1} - \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}) + \mathsf{D}(\sum_{i=0}^{k-1}\alpha_k^i\mathsf{U}^{n-i}) \tag{5.8b}$$

for a kth order BDF time discretization. The parameters γ_k and α_k^i for $i = 0, ..., k-1$ that define this time discretization have been introduced in Section 1.5 for $k = 1$, 2 and 3. Note that matrix $\mathsf{D}\mathsf{M}^{-1}\mathsf{G}$ is a discrete version of the Laplacian operator. Then, equation (5.8b) is a discretization of the Poisson equation though not in the most usual way.

Obviously, for the obtention of scheme (5.8) proposed herein, *no extra regularity conditions are required*, a main difference in comparison to the continuous PPE (5.2b). This

scheme, which has been obtained after algebraic manipulation, *is truly equivalent to the original monolithic discretized scheme* (5.7). Furthermore, boundary conditions arise naturally from the original scheme.

Approximation of $\mathsf{DM}^{-1}\mathsf{G}$

An straightforward implementation of (5.8) is not affordable in most cases. Implementation aspects for this new scheme are studied later. But the critical point is the approximation to the Laplace operator $\mathsf{DM}^{-1}\mathsf{G}$, which is a dense matrix in general. It would be computationally expensive even using a diagonal Gramm matrix M.

At this point, a further approximation is introduced in order to avoid this system matrix. As it is typically done for pressure correction methods, we can approximate

$$\mathsf{DM}^{-1}\mathsf{G} \approx \mathsf{L}, \quad \text{with components } \mathsf{L}^{ab} = -(\nabla N^a, \nabla N^b), \tag{5.9}$$

where the matrix L is the standard approximation to the Laplace operator, only possible when continuous pressure interpolations are employed. As for fractional step methods, this approximation introduces wrong pressure boundary conditions (see [82]). Here we propose an *enhanced* approximation

$$\mathsf{DM}^{-1}\mathsf{GP}^{n+1} = \mathsf{LP}^{n+1} + (\mathsf{DM}^{-1}\mathsf{G} - \mathsf{L})\mathsf{P}^{n+1} \approx \mathsf{LP}^{n+1} + (\mathsf{DM}^{-1}\mathsf{G} - \mathsf{L})\widetilde{\mathsf{P}}_p^{n+1}, \tag{5.10}$$

where $\widetilde{\mathsf{P}}_p^{n+1}$ is an extrapolation of P^{n+1} of order p obtained from previous known values. This new approximation couples space and time errors, yielding better accuracy. Using (5.10) in the DPPE we get

$$\gamma_k \delta t \mathsf{L}(\mathsf{P}^{n+1} - \widetilde{\mathsf{P}}_p^{n+1}) = \gamma_k \delta t \mathsf{DM}^{-1}(\mathsf{F}^{n+1} - \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} - \mathsf{G}\widetilde{\mathsf{P}}_p^{n+1}) + \mathsf{D}(\sum_{i=0}^{k-1} \alpha_k^i \mathsf{U}^{n-k}) \tag{5.11}$$

which is more interesting than (5.8b) from a computational point of view. This new scheme is appropriate in order to obtain pressure segregation methods. Fractional step-like and predictor corrector methods can now be designed.

5.3 Velocity correction methods based on a DPPE

Instead of obtaining the fractional step method from the monolithic system (5.7) we could start from system (5.8), where the continuity equation has been replaced by the discrete pressure Poisson equation. Furthermore, we consider a method where the extrapolated variable is the velocity instead of the pressure. In a first step the pressure is obtained from the approximated DPPE (5.11) using an extrapolation of order q (denoted by $\widetilde{\mathsf{U}}_q^{n+1}$) of the velocity U^{n+1}. Then, U^{n+1} is obtained from the momentum equation. For a k-th order method using BDF, the split scheme read as follows:

$$\gamma_k \delta t \mathsf{DM}^{-1}\mathsf{GP}^{n+1} = \gamma_k \delta t \mathsf{DM}^{-1}(\mathsf{F}^{n+1} - \mathsf{K}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1}) + \mathsf{D}(\sum_{i=0}^{k-1} \alpha_k^i \mathsf{U}^{n-k}), \tag{5.12a}$$

$$\mathsf{M}\frac{1}{\delta t}(D_k \mathsf{U}^{n+1}) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{GP}^{n+1} = \mathsf{F}^{n+1}, \tag{5.12b}$$

with $q = k-1$. Invoking approximation (5.10) for $\mathsf{DM}^{-1}\mathsf{G}$ we obtain the following system:

$$\gamma_k \delta t \mathsf{L}(\mathsf{P}^{n+1} - \widetilde{\mathsf{P}}_p^{n+1}) = \gamma_k \delta t \mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1} - \mathsf{K}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1} - \mathsf{G}\widetilde{\mathsf{P}}_p^{n+1}) + \mathsf{D}(\sum_{i=0}^{k-1} \alpha_k^i \mathsf{U}^{n-k}), \tag{5.13a}$$

$$\mathsf{M}\frac{1}{\delta t}(D_k \mathsf{U}^{n+1}) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{5.13b}$$

with $q = p = k-1$. For instance, we can obtain a first order method (in time) taking $k = 1$ and $q = p = 0$,

$$\delta t \mathsf{L}\mathsf{P}^{n+1} = \delta t \mathsf{D}\mathsf{M}^{-1}\mathsf{F}^{n+1} + \mathsf{D}\mathsf{U}^n,$$
$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}.$$

We can obtain second order accuracy in time with $k = 2$ and $q = p = 1$. In this case the system to be solved is

$$\frac{2}{3}\delta t \mathsf{L}(\mathsf{P}^{n+1} - \mathsf{P}^n) = \frac{2}{3}\delta t \mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1} - \mathsf{K}(\mathsf{U}^n)(\mathsf{U}^n) - \mathsf{G}\mathsf{P}^n) + \mathsf{D}(\frac{4}{3}\mathsf{U}^n - \frac{1}{3}\mathsf{U}^{n-1}), \tag{5.15a}$$

$$\mathsf{M}\frac{1}{2\delta t}(3\mathsf{U}^{n+1} - 4\mathsf{U}^n + \mathsf{U}^{n-1}) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}. \tag{5.15b}$$

Remark 5.1. An interesting feature of these methods is that the splitting error is related to the accuracy of the velocity instead of the pressure (as it happens for pressure correction methods). This is an advantage because it is known from the convergence analysis of different pressure segregation methods that the error estimates for the velocity are sharper than for the pressure.

Similarly, third order methods can be obtained with $k = 3$, that is a BDF3 time integration scheme, and $q = p = 2$. Under approximation (5.10) the third order velocity correction has the following form,

$$\frac{6}{11}\delta t \mathsf{L}(\mathsf{P}^{n+1} - \widetilde{\mathsf{P}}_2^{n+1}) = \frac{6}{11}\delta t \mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1} - \mathsf{K}(\widetilde{\mathsf{U}}_2^{n+1})\widetilde{\mathsf{U}}_2^{n+1} - \mathsf{G}\widetilde{\mathsf{P}}_2^{n+1}) + \mathsf{D}(\frac{18}{11}\mathsf{U}^n - \frac{9}{11}\mathsf{U}^{n-1} + \frac{2}{11}\mathsf{U}^{n-2}),$$
$$\mathsf{M}\frac{1}{6\delta t}(11\mathsf{U}^{n+1} - 18\mathsf{U}^n + 9\mathsf{U}^{n-1} - 2\mathsf{U}^{n-2}) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}.$$

Unfortunately, numerical experimentation dictates that this method is only conditionally stable, as it happens for third order pressure correction methods.

Remark 5.2. Numerical experimentation dictates that VC methods are *unconditionally stable* for $q \leq 1$ (this is proved for some schemes in Section 5.7). The instabilities shown by high order schemes seem to be a common feature for the different sorts of pressure segregation methods. This behavior has been pointed out for pressure correction methods in [148]. This misbehavior is also commented in [88] and [94] for a different version of velocity correction methods without any definitive conclusion.

5.3.1 Equivalent stabilized monolithic formulation

As for pressure correction methods in [47] and Chapter 3 , we could rewrite this system in a monolithic format in order to identify the perturbation terms introduced by the splitting. Taking the difference between (5.13b) after being multiplied by $\delta t \mathsf{D}\mathsf{M}^{-1}$ and (5.13a), we get the following equivalent system:

$$\mathsf{M}\frac{1}{\delta t}D_k\mathsf{U}^{n+1} + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1},$$
$$\mathsf{D}\mathsf{U}^{n+1} + \beta\delta t\mathsf{B}(\mathsf{P}^{n+1} - \widetilde{\mathsf{P}}_p^{n+1}) + \delta t\mathsf{D}\mathsf{M}^{-1}(\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} - \mathsf{K}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1}) = \mathsf{0},$$

where $\mathsf{B} := \mathsf{D}\mathsf{M}^{-1}\mathsf{G} - \mathsf{L}$ and $\beta = 1$ if approximation (5.10) is used and 0 otherwise. The perturbed system obtained in this case is different to the one obtained for pressure correction methods (see [47] and (3.22)), the main advantage being that the momentum equation is not perturbed anymore. This splitting is only introducing perturbation on the continuity equation. The term $\delta t\mathsf{B}(\mathsf{P}^{n+1} - \widetilde{\mathsf{P}}_p^{n+1})$ (that also appears in the classical pressure correction scheme with approximation (5.9)) arises from approximation (5.10) and is not an splitting error.

Remark 5.3. The only term due to the splitting is $\delta t\mathsf{D}\mathsf{M}^{-1}(\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} - \mathsf{K}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1})$, which is formally of order $\mathcal{O}(\delta t^{q+1})$, q being the order of the velocity extrapolation.

5.3.2 An alternative form of velocity correction methods

Instead of using approximation (5.10) we could decide to solve equation (5.8b) exactly. As noted above, $\mathsf{D}\mathsf{M}^{-1}\mathsf{G}$ is a cumbersome system matrix. Still, we could solve the equivalent system

$$\mathsf{M}\frac{1}{\delta t}(\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{5.18a}$$
$$\mathsf{D}\tilde{\mathsf{U}}^{n+1} = \mathsf{0}. \tag{5.18b}$$

We have used BDF1 for the sake of simplicity. The pressure P^{n+1} obtained from (5.18) is the same that we obtain if we solve (5.12) (with $k = 1$). Taking the difference between (5.12b) and (5.18a) we get an equation that allows us to recover the end-of-step velocity,

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} - \mathsf{K}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1} = \mathsf{0}. \tag{5.19}$$

The final scheme (5.18)-(5.19) is an equivalent version of (5.12). However, in this version an intermediate velocity $\tilde{\mathsf{U}}^{n+1}$ has been introduced.

A similar scheme obtained at the continuous level where it is the velocity the extrapolated variable has been recently proposed in [93], where it has been originally called velocity correction method. However, from a computational point of view version (5.18)-(5.19) of the method is not so interesting as (5.13), due to the fact that the pressure and the intermediate velocity are still coupled.

5.4 Predictor corrector methods

5.4.1 Schemes with a single iterative loop

Straightforward from the monolithic scheme (5.8), where the mass conservation is imposed by the DPPE, we can obtain a predictor corrector method. Denoting by a superscript i

the ith iteration of the scheme, the resulting predictor corrector method is, for $k = 1$ and using a Picard linearization of the convective term:

$$\delta t \mathsf{D}\mathsf{M}^{-1}\mathsf{G}\mathsf{P}^{n+1,i+1} = \delta t \mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1} - \mathsf{K}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i}) + \mathsf{D}(\mathsf{U}^{n}), \tag{5.20a}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1,i+1} - \mathsf{U}^{n}) + \mathsf{K}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i+1} + \mathsf{G}\mathsf{P}^{n+1,i+1} = \mathsf{F}^{n+1}, \tag{5.20b}$$

or, when using approximation (5.10),

$$\delta t \mathsf{L}(\mathsf{P}^{n+1,i+1} - \mathsf{P}^{n+1,i}) = \delta t \mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1} - \mathsf{K}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i} - \mathsf{G}\mathsf{P}^{n+1,i}) + \mathsf{D}(\mathsf{U}^{n}), \tag{5.21a}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1,i+1} - \mathsf{U}^{n}) + \mathsf{K}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i+1} + \mathsf{G}\mathsf{P}^{n+1,i+1} = \mathsf{F}^{n+1}, \tag{5.21b}$$

using BDF1 for the time discretization. These methods have to be properly initialized, that is to say, we have to start the process with a splitting error at least of the same order as the scheme. Then, for first order methods we could take $\mathsf{U}^{n+1,0} = \widetilde{\mathsf{U}}_q^{n+1}$ and $\mathsf{P}^{n+1,0} = \widetilde{\mathsf{P}}_p^{n+1}$ with $q = p = 0$. However, it is better to use $q = p = 1$, starting with a second order splitting error. This does not imply extra computational cost and furthermore accelerates convergence.

In this method all the terms are motivated from the monolithic version of the problem, a difference with some predictor corrector methods based on pressure correction schemes (see [56, 57] and Section 3.2.5). Thus, from the DPPE version of the monolithic problem a predictor corrector scheme naturally arises, while it does not occur so for the typical monolithic version.

The second order method using BDF2 is

$$\begin{aligned}\frac{2}{3}\delta t \mathsf{L}(\mathsf{P}^{n+1,i+1} - \mathsf{P}^{n+1,i}) =& \frac{2}{3}\delta t \mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1} - \mathsf{K}(\mathsf{U}^{n+1,i})(\mathsf{U}^{n+1,i}) - \mathsf{G}\mathsf{P}^{n+1,i}) \\ &+ \mathsf{D}(\frac{4}{3}\mathsf{U}^{n} - \frac{1}{3}\mathsf{U}^{n-1}),\end{aligned} \tag{5.22a}$$

$$\mathsf{M}\frac{1}{2\delta t}(3\mathsf{U}^{n+1,i+1} - 4\mathsf{U}^{n} + \mathsf{U}^{n-1}) + \mathsf{K}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i+1} + \mathsf{G}\mathsf{P}^{n+1,i+1} = \mathsf{F}^{n+1}, \tag{5.22b}$$

with the appropriate initializations $\mathsf{U}^{n+1,0} = \widetilde{\mathsf{U}}_q^{n+1}$ and $\mathsf{P}^{n+1,0} = \widetilde{\mathsf{P}}_p^{n+1}$, with $q = p = 1$.

In these schemes we have decoupled velocity and pressure computations. The iterative loop to couple velocity and pressure has been used also for a Picard linearization of the convective term in the momentum equation, although there is the possibility to use nested loops. This alternative is studied later.

As for pressure correction schemes, we can analogously obtain the perturbed monolithic version of the predictor corrector schemes (5.20) and (5.21). We have

$$\begin{aligned}&\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1,i+1} - \mathsf{U}^{n}) + \mathsf{K}(\mathsf{U}^{n+1,i+1})\mathsf{U}^{n+1,i+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \\ &\mathsf{D}\mathsf{U}^{n+1,i+1} + \delta t \beta \mathsf{B}(\mathsf{P}^{n+1,i+1} - \mathsf{P}^{n+1,i}) \\ &\quad + \delta t \mathsf{D}\mathsf{M}^{-1}(\mathsf{K}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i+1} - \mathsf{K}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i}) = 0.\end{aligned}$$

where $\beta = 1$ when using approximation (5.10) and $\beta = 0$ otherwise. The perturbation terms dissapear as the iterative procedure converges, tending to the solution of the monolithic system (5.7).

In [133] and [137] pressure correction methods and their predictor corrector counterparts have been understood as *incomplete block LU factorizations* of the monolithic

matrix. Similarly, the previous predictor corrector method can be interpreted as *a Block Gauss-Seidel iterative solver applied to the monolithic matrix.*

A third order predictor corrector method could also be considered. In this case, a BDF3 time integration scheme has to be used. Furthermore, in order to start with a third order splitting error, the methods have to be initialized with second order extrapolations ($q = p = 2$). In this case the instability of the third order VC method is solved by means of iterations.

The convergence of these methods towards the monolithic system is verified in Section 5.8 using numerical experimentation.

5.4.2 Schemes with nested iterative loops

Instead of treating the nonlinearity with the same loop as for the coupling of variables, we could design a scheme using nested loops, an outer loop for the coupling and an inner loop for the nonlinearity. For the BDF1 time integration scheme with approximation (5.10), this new system could be written as

$$\begin{aligned}\delta t\mathsf{L}(\mathsf{P}^{n+1,i+1}-\mathsf{P}^{n+1,i}) &= \delta t\mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1}-\mathsf{K}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i}-\mathsf{G}\mathsf{P}^{n+1,i})+\mathsf{D}(\mathsf{U}^{n}),\\ \mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1,i+1}-\mathsf{U}^{n})+\mathsf{K}(\mathsf{U}^{n+1,i+1})\mathsf{U}^{n+1,i+1}&+\mathsf{G}\mathsf{P}^{n+1,i+1}=\mathsf{F}^{n+1},\end{aligned}$$

where the superscript i denotes the ith iteration of the outer loop. The momentum equation has not been linearized yet .

Similarly, using BDF2 for the time integration we have the scheme

$$\begin{aligned}\frac{2}{3}\delta t\mathsf{L}(\mathsf{P}^{n+1,i+1}-\mathsf{P}^{n+1,i}) =&\frac{2}{3}\delta t\mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1}-\mathsf{K}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i}-\mathsf{G}\mathsf{P}^{n+1,i})\\ &+\mathsf{D}(\frac{4}{3}\mathsf{U}^{n}-\frac{1}{3}\mathsf{U}^{n-1}),\\ \mathsf{M}\frac{1}{2\delta t}(3\mathsf{U}^{n+1,i+1}-4\mathsf{U}^{n}+\mathsf{U}^{n-1})+\mathsf{K}(\mathsf{U}^{n+1,i+1})\mathsf{U}^{n+1,i+1}&+\mathsf{G}\mathsf{P}^{n+1,i+1}=\mathsf{F}^{n+1}.\end{aligned}$$

These predictor corrector schemes could seem more expensive than the methods introduced in the previous section. However, a faster convergence could make these methods interesting.

Identifying $\mathsf{P}^{n+1,i}$ as $\widetilde{\mathsf{P}}_p^{n+1}$, and $\mathsf{U}^{n+1,i}$ as $\widetilde{\mathsf{U}}_q^{n+1}$, the system to be solved at each iteration is equivalent to the fractional step method (5.15) proposed here taking as approximation for the unknown variables the solution of the previous outer iteration.

5.5 Stabilized velocity correction methods

In this section we consider the stabilization of the previous methods using the OSS stabilized finite element method introduced in Chapter 2.

5.5.1 Matrix version of the stabilized monolithic system

We start writing the orthogonal projection of the convective and pressure gradient terms as

$$P_h^{\perp}(\boldsymbol{u}_h^{n+1}\cdot\nabla\boldsymbol{u}_h^{n+1}+\nabla p_h^{n+1}) = \boldsymbol{u}_h^{n+1}\cdot\nabla\boldsymbol{u}_h^{n+1}+\nabla p_h^{n+1}-\boldsymbol{y}_h^{n+1},$$

where $\boldsymbol{y}^{n+1}$ is the projection of these terms onto the finite element space, that is,

$$(\boldsymbol{y}_h^{n+1}, \boldsymbol{v}_h) = (\boldsymbol{u}_h^{n+1} \cdot \nabla \boldsymbol{u}_h^{n+1} + \nabla p_h^{n+1}, \boldsymbol{v}_h), \qquad \forall \boldsymbol{v}_h \in \mathcal{V}_h. \tag{5.26}$$

From this expression it is easily checked that the discrete variational problem (1.52) for BDF1 together stabilized using OSS with the projection equation (5.26) lead to the nonlinear algebraic system

$$\begin{aligned}
&\mathsf{M}\frac{1}{\delta t}\left(\mathsf{U}^{n+1} - \mathsf{U}^{n}\right) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} \\
&\qquad + \mathsf{S}_{uu}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{S}_{up}(\mathsf{U}^{n+1})\mathsf{P}^{n+1} - \mathsf{S}_{uy}(\mathsf{U}^{n+1})\mathsf{Y}^{n+1} = \mathsf{F}^{n+1}, && \text{(5.27a)} \\
&\qquad \mathsf{D}\mathsf{U}^{n+1} + \mathsf{S}_{pp}\mathsf{P}^{n+1} + \mathsf{S}_{pu}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} - \mathsf{S}_{py}\mathsf{Y}^{n+1} = \mathsf{0}, && \text{(5.27b)} \\
&\qquad \mathsf{M}\mathsf{Y}^{n+1} - \mathsf{C}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} - \mathsf{G}\mathsf{P}^{n+1} = \mathsf{0}, && \text{(5.27c)}
\end{aligned}$$

where Y is an array with the unknown nodal values of $\boldsymbol{y}$. We have adopted a BDF1 scheme in time in order to simplify the exposition. Extension to other time integration schemes is straightforward. With the notation used above, we have introduced the new stabilization matrices:

$$\begin{aligned}
&\mathsf{S}_{uu}(\mathsf{U}^{n+1})_{ij}^{ab} = (\tau \boldsymbol{u}_h^{n+1} \cdot \nabla N^a, \boldsymbol{u}_h^{n+1} \cdot \nabla N^b)\delta_{ij}, \\
&\mathsf{S}_{up}(\mathsf{U}^{n+1})_{i}^{ab} = (\tau \boldsymbol{u}_h^{n+1} \cdot \nabla N^a, \partial_i N^b), \\
&\mathsf{S}_{uy}(\mathsf{U}^{n+1})_{ij}^{ab} = (\tau \boldsymbol{u}_h^{n+1} \cdot \nabla N^a, N^b)\delta_{ij}, \\
&\mathsf{S}_{pp}{}^{ab} = (\tau \nabla N^a, \nabla N^b), \\
&\mathsf{S}_{pu}(\mathsf{U}^{n+1})_{j}^{ab} = (\tau \partial_j N^a, \boldsymbol{u}_h^{n+1} \cdot \nabla N^b), \\
&\mathsf{S}_{py}{}_{j}^{ab} = (\tau \partial_j N^a, N^b), \\
&\mathsf{C}(\mathsf{U}^{n+1})_{ij}^{ab} = (N^a, \boldsymbol{u}_h^{n+1} \cdot \nabla N^b)\delta_{ij}.
\end{aligned}$$

An alternative version of the orthogonal projection terms has been used in [57]. The projection term is split in two least squares parts, after neglecting cross terms. In this case, there are two projection arrays to be introduced, one term for the projection of the pressure gradient and the other associated to the convective term:

$$P_h^{\perp}(\boldsymbol{u}_h^{n+1} \cdot \nabla \boldsymbol{u}_h^{n+1}) = \boldsymbol{u}_h^{n+1} \cdot \nabla \boldsymbol{u}_h^{n+1} - \boldsymbol{y}_h^{n+1}, \tag{5.28}$$

$$P_h^{\perp}(\nabla p_h^{n+1}) = \nabla p_h^{n+1} - \boldsymbol{z}_h^{n+1}, \tag{5.29}$$

where $\boldsymbol{y}_h^{n+1}$ and $\boldsymbol{z}_h^{n+1}$ are the solution of

$$(\boldsymbol{y}_h^{n+1}, \boldsymbol{v}_h) = (\boldsymbol{u}_h^{n+1} \cdot \nabla \boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h), \qquad \forall \boldsymbol{v}_h \in \mathcal{V}_h, \tag{5.30}$$

$$(\boldsymbol{z}_h^{n+1}, \boldsymbol{v}_h) = (\nabla p_h^{n+1}, \boldsymbol{v}_h), \qquad \forall \boldsymbol{v}_h \in \mathcal{V}_h. \tag{5.31}$$

This approximation slightly simplifies the final stabilized system, which now is

$$\begin{aligned}
&\mathsf{M}\frac{1}{\delta t}\left(\mathsf{U}^{n+1} - \mathsf{U}^{n}\right) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} \\
&\qquad + \mathsf{S}_{uu}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} - \mathsf{S}_{uy}(\mathsf{U}^{n+1})\mathsf{Y}^{n+1} = \mathsf{F}^{n+1}, && \text{(5.32a)} \\
&\qquad \mathsf{D}\mathsf{U}^{n+1} + \mathsf{S}_{pp}\mathsf{P}^{n+1} - \mathsf{S}_{py}\mathsf{Z}^{n+1} = \mathsf{0}, && \text{(5.32b)} \\
&\qquad \mathsf{M}\mathsf{Y}^{n+1} - \mathsf{C}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} = \mathsf{0}, && \text{(5.32c)} \\
&\qquad \mathsf{M}\mathsf{Z}^{n+1} - \mathsf{G}\mathsf{P}^{n+1} = \mathsf{0}. && \text{(5.32d)}
\end{aligned}$$

Stabilized pressure correction and predictor corrector systems from systems (5.27) and (5.32) have been obtained in [56] and [57], respectively. The *enhanced* stability properties of these methods has been fully discussed in [47] and [51]. Therein, the stabilization of the convective term is not considered.

5.5.2 Stabilized velocity correction system

Again, we can obtain a DPPE from the stabilized monolithic system (5.27) (or alternatively (5.32)). Instead of (5.8b) what we now obtain, for $k = 1$, is:

$$\begin{aligned}\left(\delta t\mathsf{D}\mathsf{M}^{-1}\mathsf{G}-\mathsf{S}_{pp}\right)\mathsf{P}^{n+1} =&\delta t\mathsf{D}\mathsf{M}^{-1}\left(\mathsf{F}^{n+1}-\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}-\mathsf{S}_{uu}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}\right.\\ &\left.-\mathsf{S}_{up}(\mathsf{U}^{n+1})\mathsf{P}^{n+1}+\mathsf{S}_{uy}(\mathsf{U}^{n+1})\mathsf{Y}^{n+1}\right)+\mathsf{D}\mathsf{U}^{n}\\ &+\mathsf{S}_{pu}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}-\mathsf{S}_{py}\mathsf{Y}^{n+1}.\end{aligned}\tag{5.33}$$

We need to make some further approximations in order to obtain a computationally appealing stabilized velocity correction system. As in Section 5.3, the velocity in the right hand side of (5.33) is extrapolated. Now, this extrapolation is needed not only for the viscous and convective term, but also for the stabilization terms associated to the momentum and continuity equations. Further, we extrapolate the projection array Y. It has been shown in [15] and [55] that treating the projection term explicitly for the monolithic system even better stability results are obtained. In Section 5.8 the orthogonal projection has been treated explicitly for velocity correction methods. For VC predictor corrector schemes, this projection is treated implicitly due to the fact that its computation is inside the external loop. In the worst case, the error induced by these extrapolations is of order $\mathcal{O}(\tau\delta t)$. If we assume that $\tau \leq C\delta t$, we do not spoil the accuracy for first and second order methods.

For the momentum equation no assumptions are required. Thus, the stabilized version of system (5.12) is,

$$\begin{aligned}\left(\delta t\mathsf{D}\mathsf{M}^{-1}\mathsf{G}-\mathsf{S}_{pp}\right)\mathsf{P}^{n+1} = \delta t\mathsf{D}\mathsf{M}^{-1}&\left(\mathsf{F}^{n+1}-\mathsf{K}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1}-\mathsf{S}_{uu}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1}\right.\\ &\left.-\mathsf{S}_{up}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{P}}_p^{n+1}+\mathsf{S}_{uy}(\widetilde{\mathsf{U}}_q^{n+1})\mathsf{Y}^{n}\right)+\mathsf{D}\mathsf{U}^{n}\\ &+\mathsf{S}_{pu}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1}-\mathsf{S}_{py}\mathsf{Y}^{n},\\ \mathsf{M}\frac{1}{\delta t}\left(\mathsf{U}^{n+1}-\mathsf{U}^{n}\right)+\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}&+\mathsf{G}\mathsf{P}^{n+1}\\ +\mathsf{S}_{uu}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}+\mathsf{S}_{up}(\mathsf{U}^{n+1})\mathsf{P}^{n+1}&-\mathsf{S}_{uy}(\mathsf{U}^{n+1})\mathsf{Y}^{n}=\mathsf{F}^{n+1},\\ \mathsf{M}\mathsf{Y}^{n+1}-\mathsf{C}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}&-\mathsf{G}\mathsf{P}^{n+1}=0,\end{aligned}$$

where we have adopted a BDF1 time integration scheme.

Alternatively, when the starting stabilized system is (5.32), the VC system is

$$\begin{aligned}\left(\delta t\mathsf{D}\mathsf{M}^{-1}\mathsf{G}-\mathsf{S}_{pp}\right)\mathsf{P}^{n+1} = \delta t\mathsf{D}\mathsf{M}^{-1}&\left(\mathsf{F}^{n+1}-\mathsf{K}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1}-\mathsf{S}_{uu}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1}\right.\\ &\left.+\mathsf{S}_{uy}(\widetilde{\mathsf{U}}_q^{n+1})\mathsf{Y}^{n}\right)+\mathsf{D}\mathsf{U}^{n}-\mathsf{S}_{py}\mathsf{Z}^{n},\\ \mathsf{M}\frac{1}{\delta t}\left(\mathsf{U}^{n+1}-\mathsf{U}^{n}\right)+\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}&+\mathsf{G}\mathsf{P}^{n+1}\\ +\mathsf{S}_{uu}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}-\mathsf{S}_{uy}(\mathsf{U}^{n+1})\mathsf{Y}^{n}&=\mathsf{F}^{n+1},\\ \mathsf{M}\mathsf{Y}^{n+1}-\mathsf{C}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}&=0,\\ \mathsf{M}\mathsf{Z}^{n+1}-\mathsf{G}\mathsf{P}^{n+1}&=0.\end{aligned}$$

At this point, approximation (5.10) can be applied in order to avoid dealing with $\mathsf{DM}^{-1}\mathsf{G}$.

5.5.3 Stabilized predictor corrector scheme

As shown in section 5.4, the monolithic system solution can be obtained using a predictor corrector method, interpreted as a block Gauss-Seidel iterative procedure. Using similar arguments to those used for the velocity correction method presented above, we can easily get an stabilized version of the predictor corrector system (5.20), which now is

$$\begin{aligned}
\left(\delta t \mathsf{DM}^{-1}\mathsf{G} - \mathsf{S}_{pp}\right)\mathsf{P}^{n+1,i+1} = \delta t \mathsf{DM}^{-1}\big(&\mathsf{F}^{n+1} - \mathsf{K}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i} - \mathsf{S}_{uu}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i} \\
&-\mathsf{S}_{up}(\mathsf{U}^{n+1,i})\mathsf{P}^{n+1,i} + \mathsf{S}_{uy}(\mathsf{U}^{n+1,i})\mathsf{Y}^{n+1,i}\big) + \mathsf{DU}^{n} \\
&+\mathsf{S}_{pu}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i} - \mathsf{S}_{py}\mathsf{Y}^{n+1,i}, \\
\mathsf{M}\frac{1}{\delta t}\left(\mathsf{U}^{n+1,i+1} - \mathsf{U}^{n}\right) + \mathsf{K}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i+1} &+ \mathsf{GP}^{n+1,i+1} \\
+\mathsf{S}_{uu}(\mathsf{U}^{n+1,i})\mathsf{U}^{n+1,i+1} + \mathsf{S}_{up}(\mathsf{U}^{n+1,i})\mathsf{P}^{n+1,i+1} &- \mathsf{S}_{uy}(\mathsf{U}^{n+1,i})\mathsf{Y}^{n+1,i} = \mathsf{F}^{n+1}, \\
\mathsf{MY}^{n+1,i+1} - \mathsf{C}(\mathsf{U}^{n+1,i+1})\mathsf{U}^{n+1,i+1} &- \mathsf{GP}^{n+1,i+1} = 0.
\end{aligned}$$

In this case we do not need any approximation when treating the monolithic system. Alternatively, a predictor corrector scheme starting from (5.32) can be easily obtained. A stabilized version of the predictor corrector methods with inner and outer iteration loops (as in Section 5.4) is straightforward.

5.6 Implementation aspects

At this point we discuss how to treat non-homogeneous boundary conditions when using velocity correction methods. For the sake of clarity we use a BDF1 time integration scheme and neglect the stabilization terms.

Let us introduce some new notation. Given an array $\widetilde{\mathsf{X}}$ with all the nodal values of an unknown $\boldsymbol{x}$, we split this array in two parts, X associated to free nodes and X_d associated to the nodes belonging to the Dirichlet boundary for the velocity, that is,

$$\widetilde{\mathsf{X}} = [\mathsf{X}, \mathsf{X}_d]^T.$$

Furthermore, for every matrix $\widetilde{\mathsf{A}} = \left[\widetilde{\mathsf{A}}^{ab}\right]$, being a and b all the nodes of the mesh, we define the block $\mathsf{A} = \left[\mathsf{A}^{ab}\right]$, where a and b are free nodes, and $\mathsf{A}_d = \left[\mathsf{A}^{ab}\right]$, where a is a free node and b a fixed node. In all cases, fixed or free is understood with respect to the velocity unknown.

The discrete momentum equation for non-homogeneous Dirichlet boundary conditions reads:

$$\begin{aligned}
\mathsf{M}\frac{1}{\delta t}\mathsf{U}^{n+1} + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{GP}^{n+1} & \\
&= \mathsf{F}^{n+1} - \mathsf{M}_d\frac{1}{\delta t}(\mathsf{U}_d^{n+1} - \mathsf{U}_d^{n}) - \mathsf{K}_d(\mathsf{U}^{n+1})\mathsf{U}_d^{n+1} + \mathsf{M}_d\frac{1}{\delta t}\mathsf{U}^{n} \\
&=: {\mathsf{F}_+}^{n+1}
\end{aligned} \tag{5.37}$$

where F_+ accounts for the force terms together with Dirichlet boundary conditions and the last term comes from the time integration.

On the other hand, with the notation introduced above, we can rewrite the discrete continuity equation as

$$\mathsf{D}\mathsf{U}^{n+1} = -\mathsf{D}_d\mathsf{U}_d^{n+1}. \tag{5.38}$$

Multiplying (5.37) by $\delta t\mathsf{M}^{-1}\mathsf{G}$, invoking (5.38) and using the velocity extrapolation, we get

$$\delta t\mathsf{D}\mathsf{M}^{-1}\mathsf{G}\mathsf{P}^{n+1} = \mathsf{D}\mathsf{M}^{-1}\left(\mathsf{F}_+^{n+1} - \mathsf{K}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1}\right) + \mathsf{D}_d\mathsf{U}_d^{n+1} + \mathsf{D}\mathsf{U}^n, \tag{5.39}$$

where we have extrapolated U^{n+1} in order to obtain the system to be solved for the velocity correction system. At this point, we are able to use approximation (5.10).

In order to solve system (5.37)-(5.39), we introduce an auxiliary nodal array X^{n+1}, obtained from the following system,

$$\mathsf{M}\mathsf{X}^{n+1} = \mathsf{F}_+^{n+1} - \mathsf{K}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1}.$$

It allows us to write equation (5.39) as

$$\delta t\mathsf{D}\mathsf{M}^{-1}\mathsf{G}\mathsf{P}^{n+1} = \mathsf{D}\mathsf{X}^{n+1} + \mathsf{D}_d\mathsf{U}_d^{n+1} + \mathsf{D}\mathsf{U}^n.$$

We summarize the implementation of this velocity correction method in a compact manner in Box 5.1, both with approximation (5.10) ($\beta = 1$) and without this approximation ($\beta = 0$). A simple fixed point method has been used to linearize the convective term.

The Laplacian approximation makes the method computationally more flexible and has been used in all the numerical tests of Section 5.8.

The implementation of the rest of predictor-corrector methods presented is straightforward, and also their stabilized versions.

Remark 5.4. We end this section with a comment about the system matrix $\mathsf{D}\mathsf{M}^{-1}\mathsf{G}$. This matrix is dense when a consistent matrix M is used. The obtention of this matrix is only computationally feasible when using a diagonal Gramm matrix. However, for iterative solvers where the matrix is not explicitly required and only matrix-vector multiplications are needed, $\mathsf{D}\mathsf{M}^{-1}\mathsf{G}$ can be treated without any extra approximation. Each matrix-vector product implies to solve a linear system with matrix M. We stress the fact that *no artificial boundary conditions are introduced* in this case.

5.7 Stability of velocity correction methods

In this section we present a complete set of stability results for the original DPPE Velocity correction schemes proposed above. We study this method using BDF1 and Crank-Nicolson time integration schemes. For simplicity in the exposition we will identify the methods as BDF1-Uq-Pp, q being the order of the velocity extrapolation and p the order of the pressure extrapolation. Similarly, for the Crank-Nicolson scheme we will use CN-Uq-Pp. The parameters q and p will take the values 0 and 1 for BDF1 and $p = q = 1$ for Crank-Nicolson.

Box 5.1. Velocity Correction Algorithm

- Solve $\quad \mathsf{M}\mathsf{X}^{n+1} = \mathsf{F}_+^{n+1} - \mathsf{K}(\widetilde{\mathsf{U}}_q^{n+1})\widetilde{\mathsf{U}}_q^{n+1} - \beta\mathsf{G}\widetilde{\mathsf{P}}_p^{n+1}.$
- Add $\quad \widetilde{\mathsf{X}}^{n+1} = \left[\mathsf{X}^{n+1}; 0\right]^T + \left[\mathsf{U}^n; \mathsf{U}_d^{n+1}\right]^T.$

IF $\beta = 0$:

- Solve $\quad \delta t\mathsf{D}\mathsf{M}^{-1}\mathsf{G}\mathsf{P}^{n+1} = \widetilde{\mathsf{D}}\widetilde{\mathsf{X}}^{n+1}.$

ELSE:

- Solve $\quad \delta t\mathsf{L}\mathsf{P}^{n+1} = \widetilde{\mathsf{D}}\widetilde{\mathsf{X}}^{n+1} + \delta t\widetilde{\mathsf{L}\mathsf{P}}_p^{n+1}.$

END IF

- Set $\mathsf{U}^{n+1,0} = \mathsf{U}^n$ and $i = 0$.

DO UNTIL CONVERGENCE:

- Solve $\quad \left[\frac{1}{\delta t}\mathsf{M} + \mathsf{K}(\mathsf{U}^{n+1,i})\right]\mathsf{U}^{n+1,i+1} = \mathsf{F}_+^{n+1} - \mathsf{G}\mathsf{P}^{n+1}.$
- Set $i \leftarrow i + 1$.

ENDDO

Let us recall some notation previously introduced in Section 3.3. If X, Y are arrays, $\{\mathsf{X}^n\}_{n=0,1,...,N}$ is a sequence of arrays of $N+1$ terms and A a symmetric positive semi-definite matrix, we define

$$
\begin{aligned}
(\mathsf{X},\mathsf{Y})_\mathsf{A} &:= \mathsf{X}\cdot\mathsf{A}\mathsf{Y},\\
\|\mathsf{X}\|_\mathsf{A} &:= (\mathsf{X}\cdot\mathsf{A}\mathsf{X})^{1/2},\\
\|\mathsf{Y}\|_{-\mathsf{A}} &:= \sup_{\mathsf{X}\neq 0}\frac{\mathsf{Y}\cdot\mathsf{X}}{\|\mathsf{X}\|_\mathsf{A}} \quad \text{(here } \mathsf{A} \text{ is assumed to be positive definite)},\\
\{\mathsf{X}^n\}\in\ell^\infty(\mathsf{A}) &\iff \|\mathsf{X}^n\|_\mathsf{A}\le C<\infty \quad \forall\, n=0,1,...,N,\\
\{\mathsf{X}^n\}\in\ell^p(\mathsf{A}) &\iff \sum_{n=0}^{N}\delta t\|\mathsf{X}^n\|_\mathsf{A}^p\le C<\infty, \quad 1\le p<\infty.
\end{aligned}
$$

Here and in the following, C denotes a positive constant, not necessarily the same at different appearances. Moreover, we denote by $N = [T/\delta t]$.

A remark is needed when $\mathsf{A} = \mathsf{K}$. This matrix is not symmetric, but it has the contribution from the convective term, which is skew-symmetric, and the contribution from the viscous term, K_{visc}, which is symmetric and positive definite. We will simply write $\mathsf{U}\cdot\mathsf{K}(\mathsf{U})\mathsf{U} = \mathsf{U}\cdot\mathsf{K}_{\text{visc}}\mathsf{U} \equiv \|\mathsf{U}\|_\mathsf{K}^2$.

These definitions will allow us to express our stability results in a compact manner.

For obtaining stability bounds in this section the basic assumption in all the cases will be that

Assumption 5.1. *The force vector* $\{\mathsf{F}^n\}_{n=0,1,\ldots,N}$ *satisfies*

$$\sum_{n=0}^{N} \delta t \|\mathsf{F}^n\|_{\mathsf{M}}^2 \leq C < \infty, \tag{5.40}$$

for all $\delta t > 0$.

This bound for $\{\mathsf{F}^n\}$, corresponding to requiring $\boldsymbol{f} \in L^2(0,T;\boldsymbol{L}^2(\Omega))$ for the continuous problem, will be used in the following. Apart from this, *no other regularity assumptions will be required.*

5.7.1 Stability of methods using BDF1

The first scheme to be studied is the simplest BDF1-U0-P0, together with the scheme BDF1-U0 *without pressure extrapolation*, that is to say, without making use of approximation (5.10). As before, we will distinguish both possibilities according to the parameter β, and we will denote the resulting methods by BDF1-U0-(P0). For these methods we will obtain the following stability results:

Stability of BDF1-U0-(P0):

$$\{\mathsf{U}^n\} \in \ell^\infty(\mathsf{M}) \cap \ell^2(\mathsf{K}), \quad \{\sqrt{\delta t}\,\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\} \in \ell^2(\mathsf{M}), \quad \{\sqrt{\delta t}\,\mathsf{P}^n\} \in \ell^2(\beta\mathsf{B})$$

Recall that matrix B is defined as $\mathsf{B} = \mathsf{D}\mathsf{M}^{-1}\mathsf{G} - \mathsf{L}$. This estimate is optimal for the velocity. The stability for the pressure is certainly not optimal (see Remark 5.5 below), but the important point is that we will obtain it without relying on the classical inf-sup condition for the velocity-pressure interpolation. Observe also that this stability is lost when $\beta = 0$.

In the previous estimate we have also displayed the additional control we have on the norm of the viscous plus convective terms.

First, let us write the scheme to be analyzed as

$$\delta t[\beta\mathsf{L} + (1-\beta)\mathsf{D}\mathsf{M}^{-1}\mathsf{G}]\mathsf{P}^{n+1} = \delta t\mathsf{D}\mathsf{M}^{-1}\mathsf{F}^{n+1} + \mathsf{D}\mathsf{U}^n, \tag{5.41a}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{5.41b}$$

where, as before, $\beta = 1$ using approximation (5.10) and $\beta = 0$ otherwise. Instead of analyzing (5.41) we will work with the equivalent form of this method, introducing an intermediate velocity, as for pressure-correction methods. In this new format we have

$$\mathsf{M}\frac{1}{\delta t}(\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n) + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{5.42a}$$

$$\mathsf{D}\tilde{\mathsf{U}}^{n+1} + \delta t\beta\mathsf{B}\mathsf{P}^{n+1} = 0, \tag{5.42b}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} = 0. \tag{5.42c}$$

Thus, the intermediate velocity we have introduced is not divergence-free when approximation (5.10) is used ($\beta = 1$). When $\beta = 0$ the second term in the left-hand-side of (5.42b) disappears together with the stability results associated to the norm $\|\cdot\|_{\beta\mathsf{B}}$.

The stability bounds for the BDF1-U0-P0 and BDF1-U0 methods are summarized in the next theorem.

Theorem 5.1. *Under Assumption 5.1 the following stability estimates hold for the BDF1-U0-P0 method:*

$$\max_{0\le n\le N}\{\|\mathsf{U}^n\|_{\mathsf{M}}^2\}+\sum_{n=1}^{N}\delta t\{\|\mathsf{U}^n\|_{\mathsf{K}}^2+\|\sqrt{\delta t}\,\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\|_{\mathsf{M}}^2+\|\sqrt{\delta t}\,\mathsf{P}^n\|_{\mathsf{B}}^2\}\le C,$$

for all $\delta t>0$.

Proof. Let us prove the stability results. Taking the inner product of (5.42a) with $2\delta t\tilde{\mathsf{U}}^{n+1}$ and using the identity

$$(2a,a-b):=a^2-b^2+(a-b)^2,$$

we get

$$\|\tilde{\mathsf{U}}^{n+1}\|_{\mathsf{M}}^2-\|\mathsf{U}^n\|_{\mathsf{M}}^2+\|\tilde{\mathsf{U}}^{n+1}-\mathsf{U}^n\|_{\mathsf{M}}^2+2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{G}\mathsf{P}^{n+1}=2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{F}^{n+1}. \tag{5.43}$$

On the other hand, (5.42c) can be reordered in order to obtain

$$\mathsf{U}^{n+1}+\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}=\tilde{\mathsf{U}}^{n+1}. \tag{5.44}$$

Squaring both terms of this equation with the inner product $(\cdot,\cdot)_{\mathsf{M}}$ it is found that

$$\|\mathsf{U}^{n+1}\|_{\mathsf{M}}^2+2\delta t\|\mathsf{U}^{n+1}\|_{\mathsf{K}}^2+\|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}\|_{\mathsf{M}}^2=\|\tilde{\mathsf{U}}^{n+1}\|_{\mathsf{M}}^2. \tag{5.45}$$

Multiplying (5.44) by $2\delta t\mathsf{F}^{n+1}$ we get

$$2\delta t\mathsf{U}^{n+1}\cdot\mathsf{F}^{n+1}-2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{F}^{n+1}+2\delta t^2\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}\cdot\mathsf{F}^{n+1}=0. \tag{5.46}$$

Adding up (5.43), (5.45) and subtracting (5.46) we obtain

$$\begin{aligned}&\|\mathsf{U}^{n+1}\|_{\mathsf{M}}^2-\|\mathsf{U}^n\|_{\mathsf{M}}^2+\|\tilde{\mathsf{U}}^{n+1}-\mathsf{U}^n\|_{\mathsf{M}}^2+2\delta t\|\mathsf{U}^{n+1}\|_{\mathsf{K}}^2+\|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}\|_{\mathsf{M}}^2\\&\quad+2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{G}\mathsf{P}^{n+1}=2\delta t\mathsf{U}^{n+1}\cdot\mathsf{F}^{n+1}+2\delta t^2\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}\cdot\mathsf{F}^{n+1}\\&\quad\le\delta t\|\mathsf{F}^{n+1}\|_{-\mathsf{K}}^2+\delta t\|\mathsf{U}^{n+1}\|_{\mathsf{K}}^2+2\delta t^2\|\mathsf{M}^{-1}\mathsf{F}^{n+1}\|_{\mathsf{M}}^2+\frac{1}{2}\|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}\|_{\mathsf{M}}^2\end{aligned} \tag{5.47}$$

From (5.42b) we can bound the term involving the pressure, obtaining

$$2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{G}\mathsf{P}^{n+1}=-2\delta t\mathsf{P}^{n+1}\cdot\mathsf{D}\tilde{\mathsf{U}}^{n+1}=2\delta t^2\|\mathsf{P}^{n+1}\|_{\beta\mathsf{B}}^2. \tag{5.48}$$

Replacing (5.44) in (5.43), and summing from $n=0$ to $n=N-1$, an arbitrary time level, we find that

$$\begin{aligned}&\|\mathsf{U}^N\|_{\mathsf{M}}^2+\sum_{n=1}^{N}\|\tilde{\mathsf{U}}^n-\mathsf{U}^{n-1}\|_{\mathsf{M}}^2+\sum_{n=1}^{N}\delta t\|\mathsf{U}^n\|_{\mathsf{K}}^2+\sum_{n=1}^{N}\delta t\|\sqrt{\delta t}\,\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\|_{\mathsf{M}}^2\\&\qquad+\sum_{n=1}^{N}\delta t\|\sqrt{\delta t}\mathsf{P}^n\|_{\beta\mathsf{B}}^2\le C.\end{aligned} \tag{5.49}$$

This proofs the Theorem. □

The stability analysis of the method BDF1-U0-P1 follows in a straightforward way. Obviously, this method only makes sense if approximation (5.10) is used ($\beta=1$, with the previous notation). The stability results we are going to obtain are:

Stability of BDF1-U0-P1:
$\{\mathsf{U}^n\} \in \ell^\infty(\mathsf{M}) \cap \ell^2(\mathsf{K}), \quad \{\sqrt{\delta t}\,\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\} \in \ell^2(\mathsf{M}),$
$\{\delta t \mathsf{P}^n\} \in \ell^\infty(\mathsf{B}), \quad \{\sqrt{\delta t}\,\delta \mathsf{P}^n\} \in \ell^2(\mathsf{B})$

Observe that the pressure stability now is weaker than for the BDF1-U0-P0 method, where the pressure was extrapolated only up to zero order. Control in $\ell^\infty(\mathsf{B})$ is obtained only for $\{\delta t \mathsf{P}^n\}$, whereas the optimal would be $\{\sqrt{\delta t}\mathsf{P}^n\}$ if $\delta t = \mathcal{O}(h^2)$ (see Remark 5.5 below). In general, a better approximation for the pressure implies less stability (which has to be found either from the use of stabilization methods or by invoking an inf-sup condition). Now the scheme reads as follows:

$$\delta t \mathsf{L}(\mathsf{P}^{n+1} - \mathsf{P}^n) = \delta t \mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1} - \mathsf{G}\mathsf{P}^n) + \mathsf{D}\mathsf{U}^n \tag{5.50a}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}. \tag{5.50b}$$

As above, we will work with an equivalent form of this method, introducing a intermediate velocity:

$$\mathsf{M}\frac{1}{\delta t}(\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n) + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{5.51a}$$

$$\mathsf{D}\tilde{\mathsf{U}}^{n+1} + \delta t \mathsf{B}(\mathsf{P}^{n+1} - \mathsf{P}^n) = 0, \tag{5.51b}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} = 0. \tag{5.51c}$$

Theorem 5.2. *Under Assumption 5.1 the following stability estimates hold for the BDF1-U0-P1 method*

$$\max_{0\le n\le N}\{\|\mathsf{U}^n\|_{\mathsf{M}}^2 + \|\delta t \mathsf{P}^n\|_{\mathsf{B}}^2\} + \sum_{n=1}^{N} \delta t\{\|\mathsf{U}^n\|_{\mathsf{K}}^2 + \|\sqrt{\delta t}\, M^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\|_{\mathsf{M}}^2 + \|\sqrt{\delta t}\,\delta \mathsf{P}^n\|_{\mathsf{B}}^2\} \le C,$$

for all $\delta t > 0$.

Proof. In this case the only place where the analysis of this method differs from the previous one is in (5.48), which now has to be replaced by

$$2\delta t \tilde{\mathsf{U}}^{n+1} \cdot \mathsf{G}\mathsf{P}^{n+1} = -2\delta t \mathsf{P}^{n+1} \cdot \mathsf{D}\tilde{\mathsf{U}}^{n+1} = \delta t^2(\|\mathsf{P}^{n+1}\|_{\mathsf{B}}^2 - \|\mathsf{P}^n\|_{\mathsf{B}}^2 + \|\delta \mathsf{P}^{n+1}\|_{\mathsf{B}}^2). \tag{5.52}$$

Then, we obtain

$$\|\mathsf{U}^N\|_{\mathsf{M}}^2 + \sum_{n=1}^{N} \|\tilde{\mathsf{U}}^n - \mathsf{U}^{n-1}\|_{\mathsf{M}}^2 + \sum_{n=1}^{N} \delta t \|\mathsf{U}^n\|_{\mathsf{K}}^2 + \sum_{n=1}^{N} \delta t \|\delta t \mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\|_{\mathsf{M}}^2$$
$$+ \|\delta t \mathsf{P}^N\|_{\mathsf{B}}^2 + \sum_{n=1}^{N} \delta t \|\sqrt{\delta t}\delta \mathsf{P}^n\|_{\mathsf{B}}^2 \le C. \tag{5.53}$$

instead of (5.49). As a conclusion, a better approximation for the pressure implies less stability.□

Now we will obtain the following stability results for the BDF1-U1-P0:

Stability of BDF1-U1-P0:
$\{\mathsf{U}^n\} \in \ell^\infty(\mathsf{M}) \cap \ell^2(\mathsf{K}), \quad \{\delta t \mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\} \in \ell^\infty(\mathsf{M}), \quad \{\sqrt{\delta t}\,\mathsf{P}^n\} \in \ell^2(\mathsf{B})$

This method reads as follows:

$$\delta t \mathsf{L}\mathsf{P}^{n+1} = \delta t \mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1} - \mathsf{K}(\mathsf{U}^n)\mathsf{U}^n) + \mathsf{D}\mathsf{U}^n, \tag{5.54a}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}. \tag{5.54b}$$

The equivalent version that will be analyzed is

$$\mathsf{M}\frac{1}{\delta t}(\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\mathsf{U}^n)\mathsf{U}^n + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{5.55a}$$

$$\mathsf{D}\tilde{\mathsf{U}}^{n+1} + \delta t \mathsf{B}\mathsf{P}^{n+1} = \mathsf{0}, \tag{5.55b}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} - \mathsf{K}(\mathsf{U}^n)\mathsf{U}^n = \mathsf{0}. \tag{5.55c}$$

taking at the first time step $\mathsf{U}^0 = \mathsf{0}$. That is to say, at the first time step the first order method BDF1-U0-P0 is used.

Theorem 5.3. *Under Assumption 5.1, the following stability estimates hold for the BDF1-U1-P0 method:*

$$\max_{0\le n\le N}\{\|\mathsf{U}^n\|_{\mathsf{M}}^2 + \|\delta t M^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\|_{\mathsf{M}}^2\}\} + \sum_{n=1}^{N}\delta t\{\|\mathsf{U}^n\|_{\mathsf{K}}^2 + \|\sqrt{\delta t}\,\mathsf{P}^n\|_{\mathsf{B}}^2\} \le C,$$

for all $\delta t > 0$.

Proof. After multiplying (5.55a) by $2\delta t\tilde{\mathsf{U}}^{n+1}$ we get

$$\begin{aligned}\|\tilde{\mathsf{U}}^{n+1}\|_{\mathsf{M}}^2 - \|\mathsf{U}^n\|_{\mathsf{M}}^2 + \|\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n\|_{\mathsf{M}}^2 + 2\delta t\tilde{\mathsf{U}}^{n+1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n \\ + 2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{G}\mathsf{P}^{n+1} = 2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{F}^{n+1}.\end{aligned} \tag{5.56}$$

We can reorder (5.55c) in the form:

$$\mathsf{U}^{n+1} + \delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} = \tilde{\mathsf{U}}^{n+1} + \delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n. \tag{5.57}$$

After squaring (5.57) with the inner product $(\cdot,\cdot)_{\mathsf{M}}$, we have

$$\begin{aligned}\|\mathsf{U}^{n+1}\|_{\mathsf{M}}^2 + 2\delta t\|\mathsf{U}^{n+1}\|_{\mathsf{K}}^2 + \|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}\|_{\mathsf{M}}^2 \\ = \|\tilde{\mathsf{U}}^{n+1}\|_{\mathsf{M}}^2 + 2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n + \|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\|_{\mathsf{M}}^2.\end{aligned} \tag{5.58}$$

On the other hand, multiplying (5.57) by $2\delta t\mathsf{F}^{n+1}$, we get

$$\begin{aligned}2\delta t\mathsf{U}^{n+1}\cdot\mathsf{F}^{n+1} + 2\delta t^2\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}\cdot\mathsf{F}^{n+1} \\ - 2\delta t^2\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\cdot\mathsf{F}^{n+1} = 2\delta t\tilde{\mathsf{U}}^{n+1}\cdot\mathsf{F}^{n+1}.\end{aligned} \tag{5.59}$$

Finally, the term involving the pressure in (5.56) is bounded using (5.48). From (5.56), (5.58), (5.59) and (5.48) we have

$$\begin{aligned}&\|\mathsf{U}^{n+1}\|_{\mathsf{M}}^2 - \|\mathsf{U}^n\|_{\mathsf{M}}^2 + \|\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n\|_{\mathsf{M}}^2 + 2\delta t\|\mathsf{U}^{n+1}\|_{\mathsf{K}}^2 + \|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}\|_{\mathsf{M}}^2 \\ &\quad - \|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\|_{\mathsf{M}}^2 + 2\delta t\|\sqrt{\delta t}\,\mathsf{P}^{n+1}\|_{\mathsf{B}}^2 \\ &= 2\delta t\mathsf{U}^{n+1}\cdot\mathsf{F}^{n+1} + 2\delta t^2\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}\cdot\mathsf{F}^{n+1} - 2\delta t^2\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\cdot\mathsf{F}^{n+1} \\ &\le \delta t\|\mathsf{F}^{n+1}\|_{-\mathsf{K}}^2 + \delta t\|\mathsf{U}^{n+1}\|_{\mathsf{K}}^2 + 4\alpha\delta t\|\mathsf{M}^{-1}\mathsf{F}^{n+1}\|_{\mathsf{M}}^2 + \frac{\delta t}{2\alpha}\|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}\|_{\mathsf{M}}^2 \\ &\quad + \frac{\delta t}{2\alpha}\|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\|_{\mathsf{M}}^2,\end{aligned} \tag{5.60}$$

being $\alpha := 2\max\{1, \delta t\}$. Summing up from $n = 0$ to $n = N - 1$, (or to an arbitrary time level), we obtain,

$$\|\mathsf{U}^N\|^2_{\mathsf{M}} + \sum_{n=1}^{N} \|\tilde{\mathsf{U}}^n - \mathsf{U}^{n-1}\|^2_{\mathsf{M}} + \sum_{n=1}^{N} \delta t \|\mathsf{U}^n\|^2_{\mathsf{K}} + \|\delta t \mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^N)\mathsf{U}^N\|^2_{\mathsf{M}}$$
$$+ \sum_{n=1}^{N} \delta t \|\sqrt{\delta t}\mathsf{P}^n\|^2_{\mathsf{B}} \leq C + \sum_{n=1}^{N} \frac{\delta t}{\alpha} \|\delta t \mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\|^2_{\mathsf{M}}. \tag{5.61}$$

The proof of the stability results required is finished after using the discrete Gronwall inequality (see [98]) in (5.49). We stress the fact that $\delta t/\alpha < 1$. Therefore, the discrete Gronwall lemma can be applied without any restriction over the time step size.□

For the BDF1-U1-P1 scheme, originally written as

$$\delta t \mathsf{L}(\mathsf{P}^{n+1} - \mathsf{P}^n) = \delta t \mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1} - \mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n - \mathsf{G}\mathsf{P}^n) + \mathsf{D}\mathsf{U}^n, \tag{5.62a}$$
$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1} \tag{5.62b}$$

which can also be written as:

$$\mathsf{M}\frac{1}{\delta t}(\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\mathsf{U}^n)\mathsf{U}^n + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{5.63a}$$
$$\mathsf{D}\tilde{\mathsf{U}}^{n+1} + \delta t \mathsf{B}(\mathsf{P}^{n+1} - \mathsf{P}^n) = 0, \tag{5.63b}$$
$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}) + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} - \mathsf{K}(\mathsf{U}^n)\mathsf{U}^n = 0. \tag{5.63c}$$

The stability analysis differs from the previous one just in the pressure term to be bounded. Using (5.52) instead of (5.44) it is easily obtained that:

Stability of BDF1-U1-P1:
$\{\mathsf{U}^n\} \in \ell^\infty(\mathsf{M}) \cap \ell^2(\mathsf{K})$, $\{\delta t \mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\} \in \ell^\infty(\mathsf{M})$,
$\{\delta t \mathsf{P}^n\} \in \ell^\infty(\mathsf{B})$, $\{\sqrt{\delta t}\,\delta \mathsf{P}^n\} \in \ell^2(\mathsf{B})$

5.7.2 Stability of methods using Crank-Nicolson

The Crank-Nicolson time integration scheme will be the last to be analyzed. We only present the stability results of the CN-U1-P1 method, since in order to maintain the second order accuracy of the Crank-Nicolson scheme the velocity and the pressure need to be extrapolated to first order. Again, we will consider that approximation (5.10) is used; otherwise, the pressure bounds presented next disappear. This method reads as follows:

$$\delta t \mathsf{L}(\mathsf{P}^{n+1} - \mathsf{P}^n) = \delta t \mathsf{D}\mathsf{M}^{-1}(\mathsf{F}^{n+1} - \mathsf{K}(\mathsf{U}^{n-1/2})\mathsf{U}^{n-1/2} - \mathsf{G}\mathsf{P}^n) + \mathsf{D}(\mathsf{U}^n), \tag{5.64a}$$
$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\mathsf{U}^{n+1/2})\mathsf{U}^{n+1/2} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{5.64b}$$

and its equivalent form to be studied is:

$$\mathsf{M}\frac{1}{\delta t}(\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\mathsf{U}^{n-1/2})\mathsf{U}^{n-1/2} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{5.65a}$$
$$\mathsf{D}\tilde{\mathsf{U}}^{n+1} + \delta t \mathsf{B}\mathsf{P}^{n+1} = 0, \tag{5.65b}$$
$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}) + \mathsf{K}(\mathsf{U}^{n+1/2})\mathsf{U}^{n+1/2} - \mathsf{K}(\mathsf{U}^{n-1/2})\mathsf{U}^{n-1/2} = 0. \tag{5.65c}$$

At the first time step, we adopt the first order BDF1-U0-P0 for simplicity. It does not affect the overall second order accuracy of the method. In the following setting this initialization is equivalent to take $\mathsf{U}^{-1/2} = \mathsf{0}$. The stability results we prove here are:

$$\boxed{\begin{array}{c}\textit{Stability of CN-U1-P1:}\\ \{\mathsf{U}^n\} \in \ell^\infty(\mathsf{M}) \cap \ell^2(\mathsf{K}), \quad \{\delta t \mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n-1/2})\mathsf{U}^{n-1/2}\} \in \ell^\infty(\mathsf{M}),\\ \{\delta t \mathsf{P}^n\} \in \ell^\infty(\mathsf{B}), \quad \{\sqrt{\delta t}\,\delta \mathsf{P}^n\} \in \ell^2(\mathsf{B})\end{array}}$$

These results are summarized in the following theorem.

Theorem 5.4. *Under Assumption 5.1, the following stability estimates hold for the CN-U1-P1 method*

$$\max_{0\leq n\leq N}\{\|\mathsf{U}^n\|_{\mathsf{M}}^2 + \|\delta t M^{-1}\mathsf{K}(\mathsf{U}^{n-1/2})\mathsf{U}^{n-1/2}\|_{\mathsf{M}}^2 + \|\delta t \mathsf{P}^n\|_{\mathsf{B}}^2\} + \sum_{n=1}^{N}\delta t\{\|\mathsf{U}^n\|_{\mathsf{K}}^2 + \|\sqrt{\delta t}\,\delta \mathsf{P}^n\|_{\mathsf{B}}^2\} \leq C,$$

for all $\delta t > 0$.

Proof. Following the same steps as for the proof of the stability estimates for the BDF1-U1-P0 and bounding the pressure term in the momentum equation as for the BDF1-U0-P1, we find that:

$$\begin{aligned}
&\|\mathsf{U}^{n+1}\|_{\mathsf{M}}^2 - \|\mathsf{U}^n\|_{\mathsf{M}}^2 + \|\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n\|_{\mathsf{M}}^2 + 2\delta t\|\mathsf{U}^{n+1/2}\|_{\mathsf{K}}^2 + \|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1/2})\mathsf{U}^{n+1/2}\|_{\mathsf{M}}^2\\
&- \|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n-1/2})\mathsf{U}^{n-1/2}\|_{\mathsf{M}}^2 + 2\delta t\|\sqrt{\delta t}\,\mathsf{P}^{n+1}\|_{\mathsf{B}}^2 + \delta t^2(\|\mathsf{P}^{n+1}\|_{\mathsf{B}}^2 - \|\mathsf{P}^n\|_{\mathsf{B}}^2 + \|\delta\mathsf{P}^{n+1}\|_{\mathsf{B}}^2)\\
&= 2\delta t\mathsf{U}^{n+1}\cdot\mathsf{F}^{n+1} + 2\delta t^2\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1}\cdot\mathsf{F}^{n+1} - 2\delta t^2\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^n)\mathsf{U}^n\cdot\mathsf{F}^{n+1}\\
&\leq \delta t\|\mathsf{F}^{n+1}\|_{-\mathsf{K}}^2 + \delta t\|\mathsf{U}^{n+1}\|_{\mathsf{K}}^2 + 4\alpha\delta t\|\mathsf{M}^{-1}\mathsf{F}^{n+1}\|_{\mathsf{M}}^2 + \frac{\delta t}{2\alpha}\|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n+1/2})\mathsf{U}^{n+1/2}\|_{\mathsf{M}}^2\\
&\quad + \frac{\delta t}{2\alpha}\|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n-1/2})\mathsf{U}^{n-1/2}\|_{\mathsf{M}}^2, \qquad (5.66)
\end{aligned}$$

with the expression of the positive constant α defined in the previous theorem. Summing up from $n = 0$ to $n = N - 1$, an arbitrary time level, we get

$$\begin{aligned}
&\|\mathsf{U}^N\|_{\mathsf{M}}^2 + \sum_{n=1}^{N}\|\tilde{\mathsf{U}}^n - \mathsf{U}^{n-1}\|_{\mathsf{M}}^2 + \|\delta t\mathsf{P}^N\|_{\mathsf{B}}^2 + \sum_{n=1}^{N}\delta t\|\mathsf{U}^n\|_{\mathsf{K}}^2 + \|\delta t\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{N-1/2})\mathsf{U}^{N-1/2}\|_{\mathsf{M}}^2\\
&+ \sum_{n=1}^{N}\delta t\|\sqrt{\delta t}\mathsf{P}^{n+1}\|_{\mathsf{B}}^2 \leq C + \sum_{n=1}^{N}\frac{\delta t}{\alpha}\|\mathsf{M}^{-1}\mathsf{K}(\mathsf{U}^{n-1/2})\mathsf{U}^{n-1/2}\|_{\mathsf{M}}^2. \qquad (5.67)
\end{aligned}$$

Again, as pointed out in the previous theorem, the discrete Gronwall lemma can be applied for any time step size, concluding the proof.□

Remark 5.5. We can easily see from the previous stability bounds that the inherent pressure stability of velocity correction methods seems insufficient. We only have some pressure stability under approximation (5.10). And even in this case, the stability is under the norm associated to B (a difference between discrete Laplacians that tends to zero with h). Thus, their behavior is different from pressure correction methods, which have a *stronger* inherent pressure stability (see Section 3.3). For first order splitting error, using

pressure correction methods with approximation (5.10) we have control over the whole pressure gradient $\|\sqrt{\delta t}\nabla p_h^{n+1}\|_0$. For velocity correction methods, the bound for the finite element projection (weighted with $\sqrt{\delta t}$) does not appear. However, as proved in Section 3.3.2, we can recover the control over the whole gradient by using the stability provided by the momentum equation (proved in the following theorem). Summarizing, even though the stability bounds of the VC methods seem weaker, we also have stability over the whole pressure gradient (under approximation (5.10)).

Let us recall some notation previously used in Section 3.3.2. We will need some projections of ∇p_h^{n+1} with respect to the L^2-inner product

$\boldsymbol{\pi}_1$: projection onto $\mathcal{V}_{h,0}$
$\boldsymbol{\pi}_2$: projection onto $(\mathcal{V}_{h,0})^\perp \cap \mathcal{V}_h$
$\boldsymbol{\pi}_3$: projection onto $(\mathcal{V}_h)^\perp$

and we will denote $\boldsymbol{\pi}_{ij} := \boldsymbol{\pi}_i + \boldsymbol{\pi}_j$ and $\boldsymbol{\pi}_h = \boldsymbol{\pi}_{12}$.

Theorem 5.5. *Under Assumption 5.1 and assuming a quasi-uniform finite element partition, the following stability estimate holds for the BDF1-U0-Pp method:*

$$\sum_{n=1}^{N} \delta t \|h\boldsymbol{\pi}_1^n\|_0^2 \leq C. \tag{5.68}$$

For the BDF1-U1-Pp and CN-U1-Pp, it holds

$$\sum_{n=1}^{N} \delta t \|h\boldsymbol{\pi}_1^n\|_0 \leq C, \tag{5.69}$$

being $p = 0$ or 1, for all $\delta t > 0$.

Proof. Let us write inequalities (5.42a), (5.55a) and (5.65a) in an appropriate format in order to analyze the different methods together. We use the following variational form:

$$\begin{aligned}\frac{1}{\delta t}(\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n, \boldsymbol{v}_h) + \gamma\nu(\nabla \boldsymbol{u}_h^{n+\delta-1}, \nabla \boldsymbol{v}_h) + \gamma(\boldsymbol{u}_h^{n+\delta-1} \cdot \nabla \boldsymbol{u}_h^{n+\delta-1}, \boldsymbol{v}_h) \\ + (\nabla p_h^{n+1}, \boldsymbol{v}_h) = \langle \boldsymbol{f}^{n+\delta}, \boldsymbol{v}_h\rangle\end{aligned} \tag{5.70}$$

which must hold for all test functions $\boldsymbol{v}_h \in \mathcal{V}_{h,0}$, where $\delta = 1$ for the BDF1 scheme and $\delta = 1/2$ for the CN scheme. Furthermore, $\gamma = 0$ for the BDF1-U0-Pp and $\gamma = 1$ for the rest of methods. We take $\boldsymbol{\pi}_1^{n+1}$ as test function in (5.70) and use the inverse estimate (1.37) valid for quasi-uniform finite element partitions, obtaining

$$\begin{aligned}\|\boldsymbol{\pi}_1^{n+1}\|_0^2 \leq &\|\boldsymbol{f}^{n+\delta}\|_{-1}\frac{C_{\text{inv}}}{h}\|\boldsymbol{\pi}_1^{n+1}\|_0 + \frac{1}{2\delta t}\|\tilde{\boldsymbol{u}}_h^{n+1} - \boldsymbol{u}_h^n\|_0\|\boldsymbol{\pi}_1^{n+1}\|_0 \\ &+ \gamma(N_a\|\tilde{\boldsymbol{u}}_h^{n+\delta-1}\|_1 + N_c\|\tilde{\boldsymbol{u}}_h^{n+\delta-1}\|_1^2)\frac{C_{\text{inv}}}{h}\|\boldsymbol{\pi}_1^{n+1}\|_0.\end{aligned}$$

where N_a and N_c are the norms of the viscous and convective terms, respectively. We can easily get the bounds stated in the theorem from the previous inequality bounding the right hand side terms using inequalities (5.49), (5.61) and (5.67). □

5.8 Numerical tests

In this section we present some numerical results to test the velocity correction schemes introduced in this chapter. In all cases we use approximation (5.10). We compare the behavior of these methods with the stabilized monolithic, pressure correction and predictor corrector systems studied in Section 3.4 . The results shown along this section are obtained with the stabilized versions of the methods using OSS. It has allowed to use equal velocity-pressure interpolation. In particular, we have taken $k_{\mathrm{q}} = k_{\mathrm{v}} = 1$, with the notation of Section 5.2. Both for pressure and velocity correction methods only second order splitting error methods have been analyzed, due to its clear superiority in comparison to first order methods (see [52]) without extra computational cost. The only possible reason that would (hardly) justify pressure correction methods with a first order splitting error is their inherent stability. However, the inherent stability is weaker in the case of velocity correction methods, as stressed in Remark 5.5.

However, the introduction of consistent stabilization techniques, such as OSS, makes this stability unnecessary, improving the accuracy.

The numerical examples presented in this section are the same as in Section 3.4. Likewise, very similar comments apply.

5.8.1 Convergence test

The first example we consider is a simple convergence test whose goal is to check numerically the rate of convergence in time for some of the numerical methods described.

The computational domain is the unit square, discretized using a uniform triangular mesh of 11×11 nodal points (200 triangles). The boundary and initial conditions and the force term are prescribed so that the analytic solution is $\boldsymbol{u} = (y, -x)\sin(\pi t/10)\exp(t/25)$ and $p = 0$. Note that the exact solution belongs to the finite element space, and thus the only source of numerical error is the time approximation.

Results are shown in Figure 5.1. The error E is measured in the ℓ^2 norm of the sequence $\{\boldsymbol{u}^n - \boldsymbol{u}(t^n)\}$. It is seen that all the methods show the expected rate of convergence. This is particularly relevant for the predictor-corrector schemes, whose error is affected by the convergence tolerance adopted in the iterative loop of each time step.

5.8.2 Flow in a cavity

In this second example we solve the classical cavity flow problem at a Reynolds number $Re = 100$. The computational domain is the unit square, discretized using a mesh of 21×21 nodal points (400 triangles). The velocity is fixed to zero everywhere except on the top boundary, where it is prescribed to (1,0).

Even though the solution in this simple example is stationary, we obtain it by stepping in time. The goal of this test is precisely to check the properties of the schemes proposed for the long-term time integration of stationary solutions (very often difficult to obtain in a stationary calculation) and, particularly, their numerical dissipation. The time step employed is $\delta t = 1$.

Figure 5.2 shows the evolution towards the steady state case for velocity correction methods compared with the monolithic one. We take a VC method with a second order splitting error ($p = q = 1$). It is observed that the VC methods reach the steady state case slower than the monolithic system, which is more dissipative.

The VC predictor corrector method, as expected, follows very closely the plot of the monolithic system. Furthermore, when the residual reaches the tolerance of the predic-

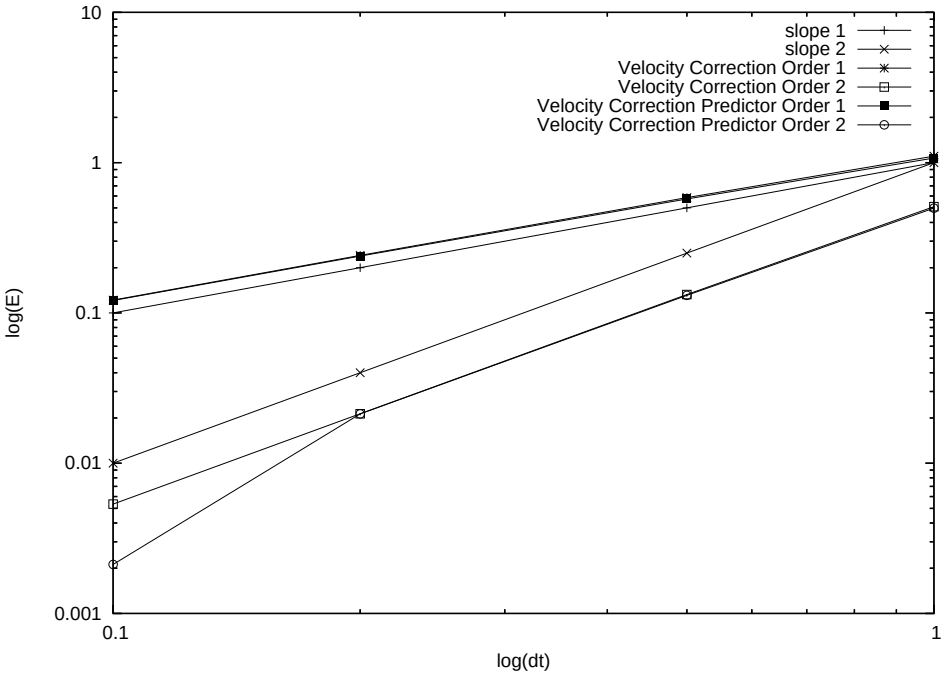


Figure 5.1: Convergence test

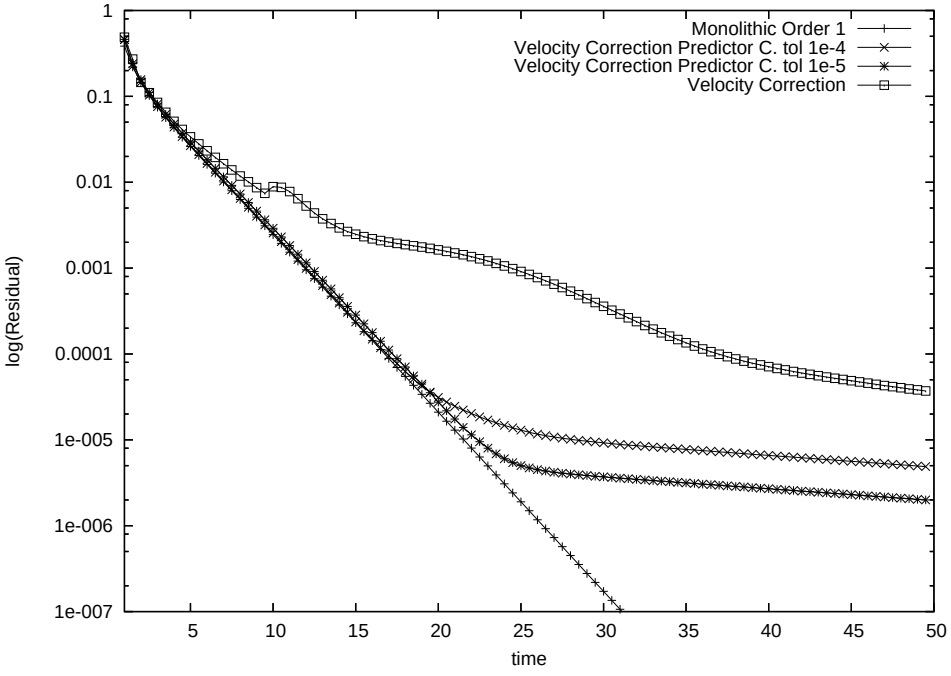


Figure 5.2: Evolution towards the steady-state for the cavity flow problem using monolithic and velocity correction schemes

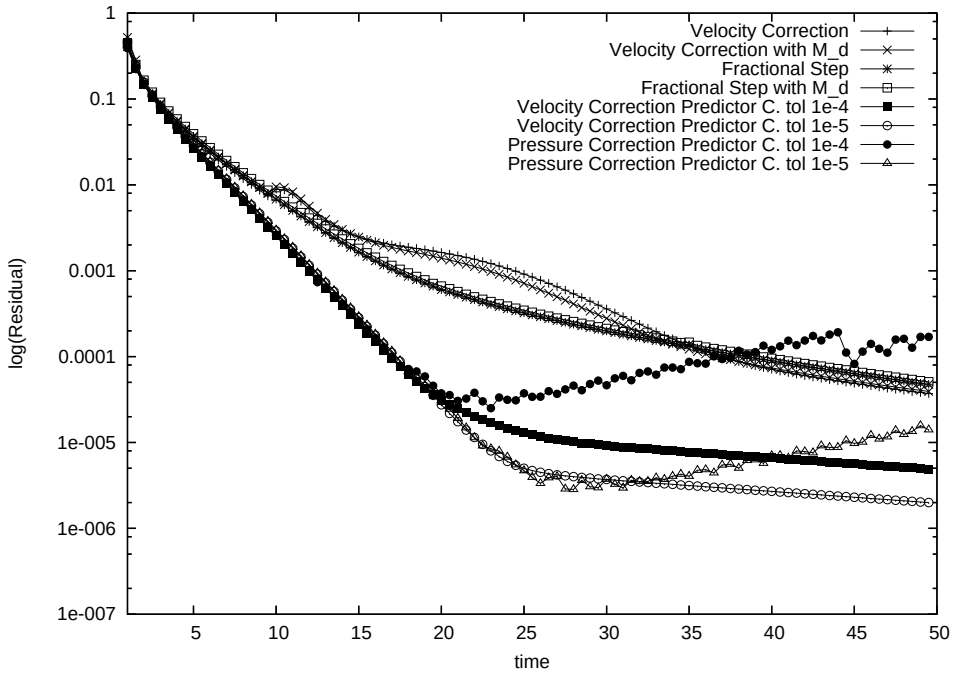


Figure 5.3: Evolution towards the steady-state for the cavity flow problem using velocity and pressure correction schemes

tor corrector loop, the plots of the VC predictor corrector methods detaches from the monolithic one, tending slower to the steady state case, as it could be expected.

On the other hand, the use of the consistent or lumped mass matrix seems that does not affect essentially the results obtained.

In Figure 5.3, the velocity correction method is compared to its dual pressure correction method. Its assymptotic behavior is very similar. However, their predictor corrector versions give very different plots. Pressure correction predictor corrector methods have a weird behavior when tending to the steady state case. For VC predictor corrector methods better convergence is attained. In Figure 5.3, M_d stands for the diagonal mass matrix obtained from a nodal quadrature.

Concerning the convergence of the methods, in Figures 5.4, 5.5 and 5.6 we plot the number of iteration performed versus time step in order to compare velocity and pressure correction methods (with a maximum of 10 iterations allowed). These iterations deal with the non linearity of the convective term for velocity and pressure correction methods as well as with the convergence to the monolithic system for predictor corrector schemes. For this specific test, velocity correction schemes reach convergence slightly faster than pressure correction schemes in all cases. Due to the fact that both pressure correction and velocity correction methods have a very similar computational cost per iteration (in both cases we have to solve a Poisson equation for the pressure and a momentum equation for the velocity) the CPU time used by each method is proportional to the number of iterations. VC turn out to be cheaper for this example.

5.8.3 Flow over a cylinder

The last example is also a classical benchmark, namely, the flow over a cylinder. The computational domain is $\bar{\Omega} = [0, 16] \times [0, 8] \setminus D$, with the cylinder D of diameter 1 and centered at $(4, 4)$. The velocity at $x = 0$ is prescribed to $(1, 0)$, whereas at $y = 0$ and $y = 8$ the y-velocity component is prescribed to 0 and the x-component is left free. The

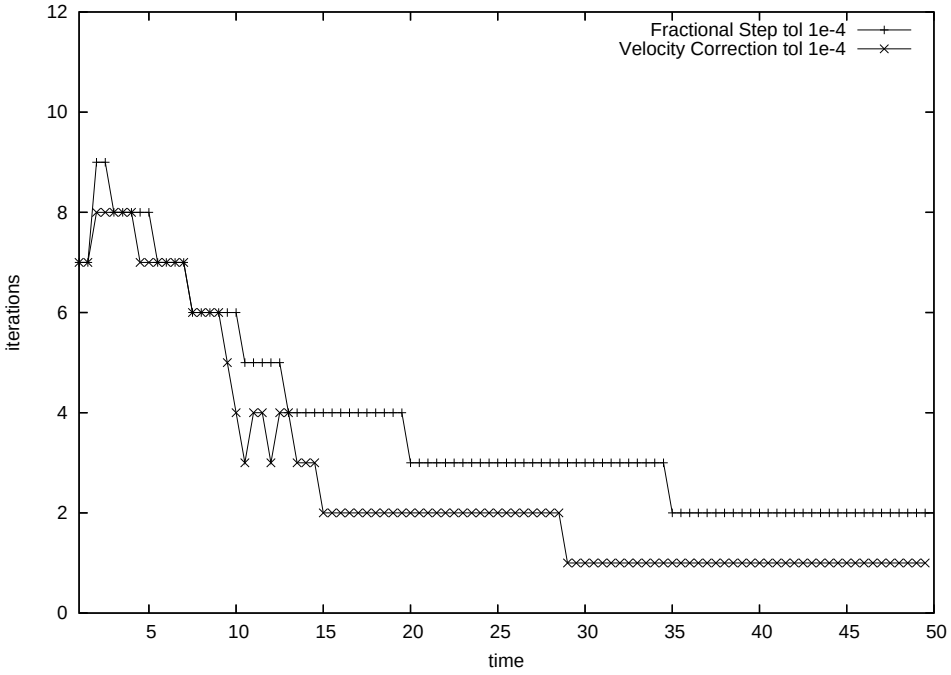


Figure 5.4: Number of iterations per time step using different velocity and pressure correction schemes

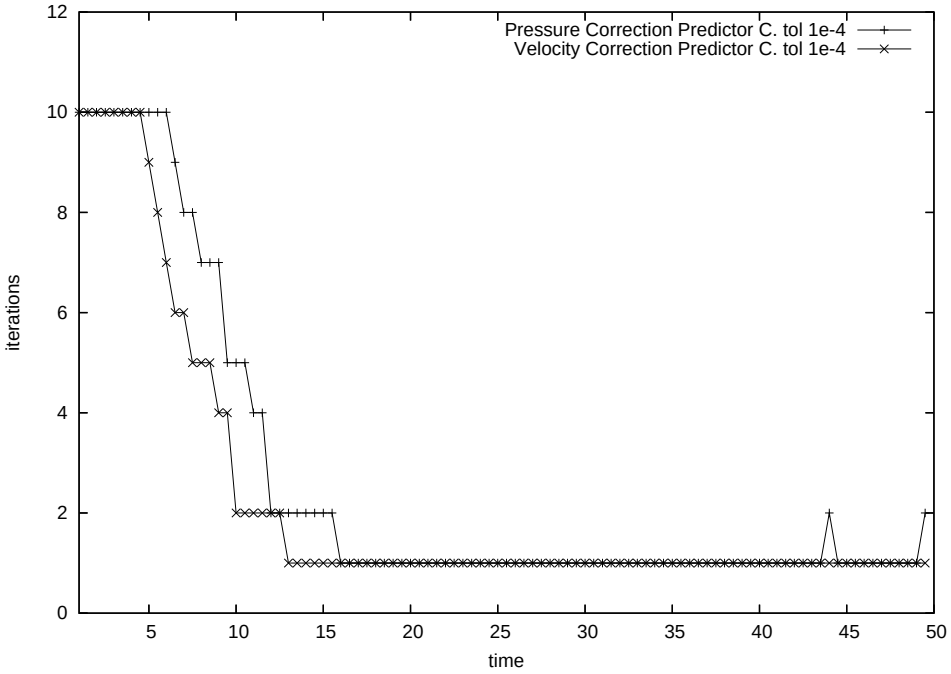


Figure 5.5: Number of iterations per time step using different velocity and pressure correction schemes

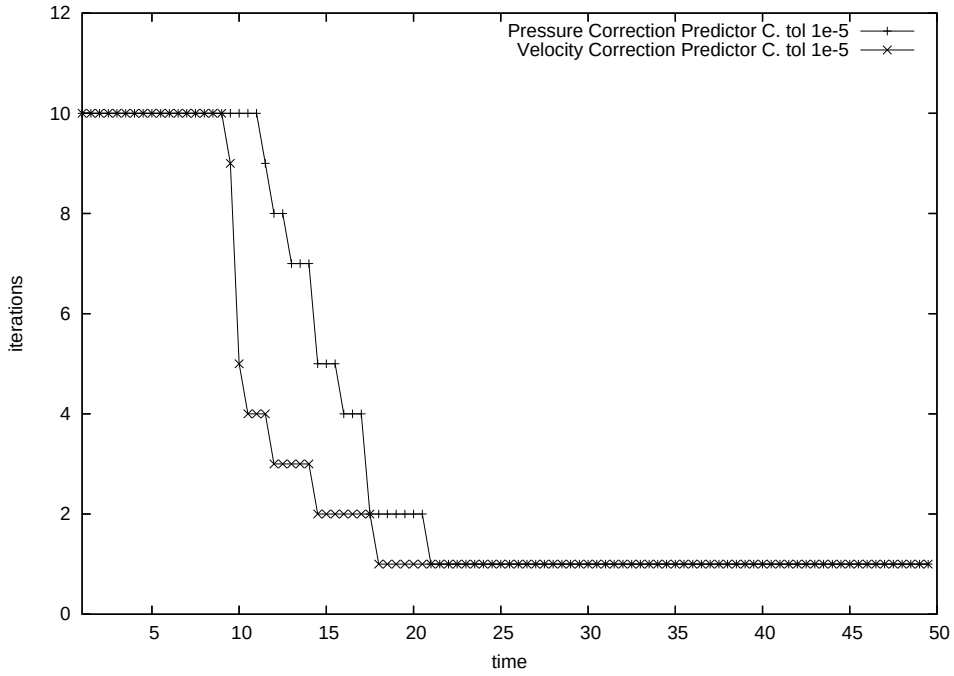


Figure 5.6: Number of iterations per time step using different velocity and pressure correction schemes

outflow (where both the x- and y-components are free) is $x = 16$. The Reynolds number is 100, based on the cylinder diameter and the prescribed inflow velocity. The finite element mesh employed consists of 3604 linear triangles, with 1902 nodal points. This example has been used in Section 3.4 to analyze pressure correction methods in time. Here we try to show the same case with VC methods. Only second order schemes have been considered.

The evolution of the y-velocity component at the control point located at (6,4) is shown in Figure 5.7. The time step size used in all the cases is $\delta t = 0.05$. A maximum of only 5 iterations has been permitted. Similar results to those of the pressure correction methods have been obtained (see [52]). Again, the predictor corrector method lays between the monolithic and velocity correction method.

5.9 Conclusions

The new kind of velocity correction methods presented in this chapter and obtained at the discrete level from a pressure Poisson equation are appropriate methods for the simulation of fluid dynamics, avoiding the problems that arise when using a pressure Poisson equation obtained at the continuous level. This requires a finite element approximation to $H^2(\Omega)$, which is avoided by our purely algebraic approach.

The VC methods proposed herein have a similar behavior than the widely used pressure correction methods. Moreover, in both cases we observe that third order methods (in time) become conditionally stable. It is a surprising result, because the perturbation introduced by the splitting is very different in both cases.

In all cases it seems that the inherent pressure stability of VC schemes is weaker than the one of pressure correction methods. Whereas for first order pressure correction methods an stability bound of the gradient pressure can be obtained (under some usual assumptions), only part of this term is controlled using VC methods. Nevertheless, we can recover control over the rest of the pressure gradient from the momentum equation.

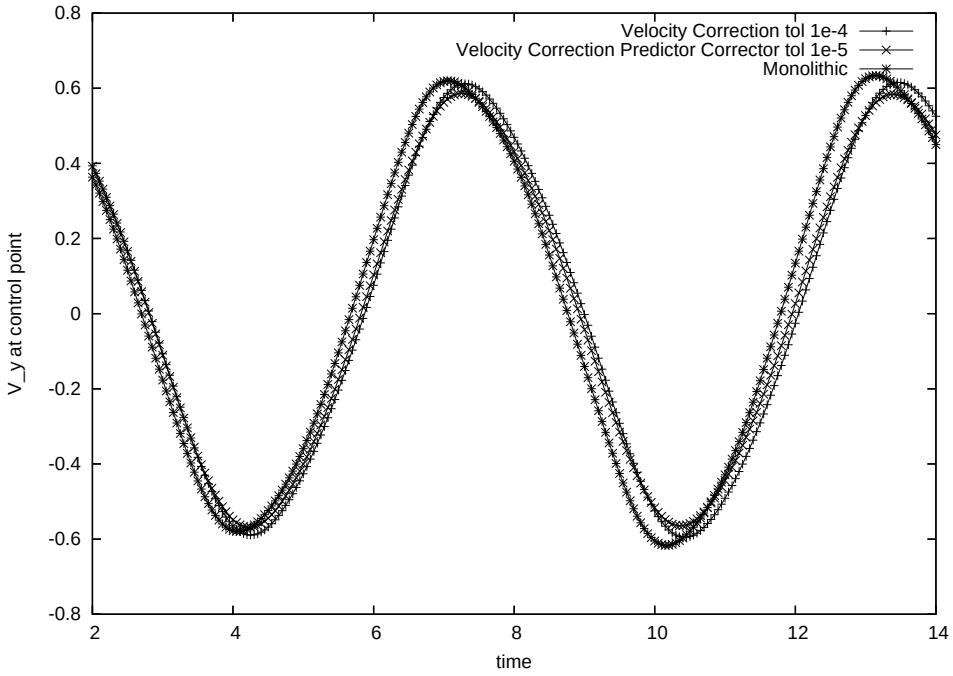


Figure 5.7: Temporal evolution of the y-velocity component at the control point

A nice feature of the DPPE monolithic system is the fact that VC predictor corrector schemes arise naturally. We have compared these schemes to predictor corrector schemes motivated from pressure correction methods. Numerical experimentation shows that these new methods exhibit a similar or, in most cases, better behavior.

Chapter 6

The ALE Framework

In this chapter we introduce the ALE framework that we need in order to write the Navier-Stokes equations on moving domains. This framework will be used in Chapter 7 for the statement of the coupled fluid-structure system. Moreover, we analyze the approximation to the convection-diffusion equation on moving domains using an ALE framework and the OSS stabilized finite element method. As basic numerical strategy, we discretize the equation in time using first and second order backward differencing (BDF) schemes, whereas space is discretized using a stabilized finite element method (the orthogonal subgrid scale formulation) to deal with convection dominated flows. The analysis of this chapter has motivated [4]

6.1 Introduction

In this chapter we propose and analyze two time integration schemes, of first and second order, for the numerical approximation of the transient convection-diffusion equation in moving domains. This equation is written in an ALE framework, in which the temporal derivatives are expressed with respect to the reference of a moving domain Ω_t obtained from a mapping of the domain at the initial time. The space discretization is carried out using a stabilized finite element method that allows us to deal with convection dominated flows.

The ALE framework, initially used with a finite element approximation in [64], has become widely popular when simulating fluid-structure interaction problems. Even tough one can find a lot of numerical experimentation using the ALE approach, some aspects have remained on the dark side for a long time. For instance, the meaning and effect of the *Geometric conservation Law* (GCL) or how the accuracy of a numerical method in fixed domains is spoiled when introducing moving domains with an ALE formulation were not clear. Farhat and co-authors have shown in [68] that the GCL makes the numerical scheme preserve a maximum principle. In [80], the authors have shown that this condition is not necessary to obtain second order ALE schemes in a finite volume framework. More recently, in a finite element setting and taking the transient convection-diffusion equation as the *model* equation, works like [74] and [130] have also allowed to clarify the effect of the GCL on the stability properties, the different behavior between conservative and non-conservative forms, and also some convergence results have been proved. Further analyses, for second order schemes, have been developed in later works, as [75] and [19]. Herein we use the mathematical setting used, e.g., in [130] for the description of this method.

The ALE framework does not introduce any error by itself at the continuous level.

However, when the problem is discretized in time, some errors due to the ALE description arise. At this step, for fixed domains, the only source error is the time derivative of the unknown. In addition, for moving domains, also the error from the evaluation of the mesh velocity has to be accounted for. This velocity is calculated as the time derivative of the space position of a particle. Thus, an error is induced when this time derivative is calculated numerically.

On the other hand, in practical applications the mesh velocity belongs to the finite element space and does not introduce any interpolation error. Thus, we consider that the ALE formulation is better understood analyzing the problem semi-discretized in time. However, most numerical analysis (see [74], [75] and [19]) first study the semi-discrete problem in space, and then the fully discretized problem.

The convection-diffusion equation (as the Navier-Stokes equations) when discretized in space with the standard Galerkin formulation shows numerical oscillations if the convective term is dominant. With the aim of developing a finite element method free of spurious oscillations many methods have been proposed during the last twenty years, such as SUPG (see [33]), Galerkin/Least-squares (see [107]) or the Subgrid Scale stabilization (see [102]). A comparative of different stabilization methods can be found in [45]. The *Orthogonal Subgrid Scale* method (OSS) used along this work belongs to this last family and was introduced by Codina in [46]. The method is designed taking as starting point the Subgrid Scale variational setting proposed by Hughes in [104] and modeling the subgrid problem in a particular way, in particular taking the subgrid scales orthogonal to the finite element space (see Chapter 2). The common aspect of all these methods is found in the convergence analysis of the discrete problem in space. For the Galerkin approximation, the error estimate bound depends on the physical properties (the Péclet number for the convection-diffusion equation), and increases as the convective term is more dominant. In fact, the stability bound blows up as diffusion goes to zero, reflecting the fact that the continuous problem is a singularly perturbed one. But when using stabilized methods this negative feature does not appear anymore. This is explained because the new terms introduced by the stabilization control the convective term norm. In the present analysis we have been able to obtain appropriate error estimates only controlling a part of the convective term, which is an innovative result.

As far as we know, most of the existing stabilization techniques are extended to transient problems using the framework of the discontinuous Galerkin space-time formulation, increasing notably the computer cost for schemes in time of order two or higher. This situation has been improved by Guermond in [87], where he analyzes the introduction of a certain numerical subgrid viscosity. Optimal convergence results are obtained for an evolutionary equation. The key point is the uncoupling of the stabilization terms with the temporal derivative of the unknown. Another stabilization method with this feature is presented in [22].

Codina and Blasco analyze in [55] the transient convection-diffusion-reaction equation discretized in space using the OSS method and in time with the backward Euler time integration. Further, they consider the *tracking* of the subscales in time. Optimal convergence and stability results are obtained.

The present work can be viewed as an extension of [55]. We generalize the situation to moving domains (using an ALE approach). In addition, first and second order backward differencing (BDF) time integration schemes are considered, which will be denoted by BDF1 and BDF2, respectively. The blend of a stabilized finite element method with the use of an ALE framework is one of the innovative aspects of this chapter.

In order to analyze the stabilized method for transient problems, the following strategy

is adopted in [55]: first the semi-discrete problem is studied (where no stabilization terms appear) and later the fully discrete method is analyzed. As it is shown in [55], this provides a natural way to deal with the subscales whose approximation enhances the stability and accuracy of the formulation. The main drawback of this strategy is that space regularity for the convergence analysis needs to be assumed for the semidiscrete solution, not for the continuous one.

The first time integration scheme considered uses the classical backward Euler formula for the approximation of both the time derivative of the unknown and the calculation of the mesh velocity. We label this method as:

- BDF1-BDF1$_{\delta t}$ for the problem semi-discretized in time,
- BDF1-BDF1$_{\delta t,h}$ for the fully discretized problem using the classical Galerkin approximation in space,
- BDF1-BDF1-OSS$_{\delta t,h}$ for the fully discretized problem using the OSS method in space,
- BDF1-OSS$_{\delta t,h}$ for the fully discretized problem using the OSS method in space on fixed domains (not in an ALE framework).

In the second method the time integration makes use of the second order BDF formula. Again, we use the following notation:

- BDF2-BDF2$_{\delta t}$ for the problem semi-discretized in time,
- BDF2-BDF2$_{\delta t,h}$ for the fully discretized problem using the classical Galerkin approximation in space,
- BDF2-BDF2-OSS$_{\delta t,h}$ for the fully discretized problem using the OSS method in space,
- BDF2-OSS$_{\delta t,h}$ for the fully discretized problem using the OSS method in space on fixed domains (not in an ALE framework).

Let us underline what is new in each case. The BDF1-BDF1$_{\delta t,h}$ method has been analyzed in [74]. As explained above, we change the order of the discretization: first we analyze BDF1-BDF1$_{\delta t}$, and then BDF1-BDF1-OSS$_{\delta t,h}$, introducing the appropriate stabilization terms. For fixed domains, the BDF1-OSS$_{\delta t,h}$ has been analyzed in [55]. However, the analysis herein is slightly different. The analysis of convergence and stability of the semidiscrete method BDF2-BDF2$_{\delta t}$ is new, and also its fully discrete stabilized version BDF2-BDF2-OSS$_{\delta t,h}$. We specially remark the fact that convergence results independent of the physical properties can be obtained without the full norm of the convective term. Even for fixed domains, the stability and convergence results for BDF2-OSS$_{\delta t,h}$ are new. In all cases the long time behavior has been considered.

Numerical experimentation with the ALE methods (for diffusion dominated problems using the Galerkin method) BDF1-BDF1$_{\delta t,h}$ and BDF2-BDF2$_{\delta t,h}$ can be found in [75], [19] and [130], showing the expected behavior. The application of BDF1-OSS$_{\delta t,h}$ and BDF2-OSS$_{\delta t,h}$, can be found in [47] and [52] for the solution of fluid problems. Finally, the blend of these methods, BDF1-BDF1-OSS$_{\delta t,h}$ and BDF2-BDF2-OSS$_{\delta t,h}$, has been used for simulating engineering problems in the following chapters.

The chapter is organized as follows. In Section 6.2 we state the governing equations for moving domains in an ALE framework. Some important ingredients in order to define the

ALE approach are introduced. The semi-discrete problem is formulated both for BDF1 and BDF2. The section ends with the presentation of the OSS stabilization method and the fully discrete problem. Section 6.3 is devoted to the semi-discrete problem. First and second order methods are considered for which stability and optimal convergence estimates are obtained. Section 6.4 presents an analogous analysis to that of Section 6.3 but for the fully discrete problem. Finally, some conclusions are drawn in Section 6.5.

6.2 Problem statement

6.2.1 The continuous problem

In order to study the ALE framework together with a stabilized finite element method, we take as a model test problem the transient convection-diffusion equation. The problem written in an Eulerian framework consists in finding a function u such that

$$\frac{\partial u}{\partial t} - \nu\Delta u + \boldsymbol{a}\cdot\nabla u = f \quad \text{in } \Omega_t \times (0,T), \tag{6.1a}$$

$$u = 0 \quad \text{on } \partial\Omega_t \times (0,T), \tag{6.1b}$$

$$u(\boldsymbol{x}_0, 0) = u_0 \quad \text{in } \Omega_0 \times \{0\}, \tag{6.1c}$$

where $\Omega_t \subset \mathbb{R}^d$ (d=2,3) is a bounded and polyhedral domain (moving in time), $[0,T]$ is the time interval of analysis, $\boldsymbol{a}$ is a divergence-free velocity field and $\nu > 0$ is the diffusion coefficient. Homogeneous boundary conditions are assumed to clarify the analysis. We also assume the following regularity of the data:

$$f \in L^2(0,T;H^{-1}(\Omega_t)), \quad u_0 \in L^2(\Omega_0), \quad \boldsymbol{a} \in \boldsymbol{L}^\infty(\Omega_t),$$

assuring the existence of a unique solution $u(t) \in L^2(0,T;H^1(\Omega_t)) \cap \mathcal{C}^0(0,T;L^2(\Omega_t))$.

We introduce some key ingredients of an ALE framework. Let $\mathcal{A}_t$ be a family of mappings, which for all $t \in [0,T]$ map a point $\boldsymbol{x}_0 \in \Omega_0$ into a point $\boldsymbol{x} \in \Omega_t$,

$$\mathcal{A}_t : \Omega_0 \longrightarrow \Omega_t, \quad \boldsymbol{x}(\boldsymbol{x}_0, t) = \mathcal{A}_t(\boldsymbol{x}_0).$$

We assume that $\mathcal{A}_t$ is invertible with inverse $\mathcal{A}_t^{-1}$. For $t_1, t_2 \in [0,T]$ we define

$$\mathcal{A}_{t_1,t_2} : \Omega_{t_1} \longrightarrow \Omega_{t_2}, \quad \mathcal{A}_{t_1,t_2} = \mathcal{A}_{t_2} \circ \mathcal{A}_{t_1}^{-1}.$$

We note that the family of mappings is arbitrary. Several techniques have been suggested in order to construct this ALE mapping. If $\mathcal{A}_t$ is the mapping arising from the motion of the particles, the resulting formulation would be of pure Lagrangian type.

Let us consider a function $f : \Omega_t \times [0,T] \longrightarrow \mathbb{R}$. We indicate with $\hat{f} = f \circ \mathcal{A}_t$ the corresponding function in the ALE frame:

$$\hat{f} : \Omega_0 \times [0,T] \longrightarrow \mathbb{R}, \quad \hat{f}(\boldsymbol{x}_0, t) = f(\mathcal{A}_t(\boldsymbol{x}_0), t).$$

Furthermore, the time derivatives in the ALE frame are defined as follows:

$$\left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{x}_0} : \Omega_t \times [0,T] \longrightarrow \mathbb{R}, \quad \left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{x}_0}(\boldsymbol{x}, t) = \frac{\partial \hat{f}}{\partial t}(\boldsymbol{x}_0, t).$$

The domain velocity $\boldsymbol{w}$ is calculated using the following expression:

$$\boldsymbol{w}(\boldsymbol{x}, t) = \left.\frac{\partial \boldsymbol{x}}{\partial t}\right|_{\boldsymbol{x}_0} = \frac{\partial \mathcal{A}_t(\boldsymbol{x}_0)}{\partial t},$$

and the Jacobian of the ALE mapping is given by

$$J_t = \det(\boldsymbol{J}_t), \quad \boldsymbol{J}_t = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{x}_0}.$$

We recall the *Reynolds transport formula*. Let $\psi(\boldsymbol{x}, t)$ be a function defined in Ω_t. Then, for any subdomain $V_t \subseteq \Omega_t$ such that $V_t = \mathcal{A}_t(V_0)$ with $V_0 \subseteq \Omega_0$ it holds that

$$\frac{d}{dt}\int_{V_t} \psi(\boldsymbol{x}, t)\,\mathrm{d}V = \int_{V_t} \left(\left.\frac{\partial \psi}{\partial t}\right|_{\boldsymbol{x}_0} + \psi \nabla \cdot \boldsymbol{w} \right) \mathrm{d}V.$$

In particular, if $v : \Omega_t \longrightarrow \mathbb{R}$, that is, if v does not depend explicitly on time, we have that

$$\frac{d}{dt}\int_{\Omega_t} v\,\mathrm{d}\Omega = \int_{\Omega_t} v \nabla \cdot \boldsymbol{w}\,\mathrm{d}\Omega. \tag{6.2}$$

With all this notation introduced, we are ready to write (6.1) in the ALE framework. It now reads

$$\left.\frac{\partial u}{\partial t}\right|_{\boldsymbol{x}_0} - \nu \Delta u + (\boldsymbol{a} - \boldsymbol{w}) \cdot \nabla u = \boldsymbol{f} \quad \text{in } \Omega_t \times (0, T), \tag{6.3a}$$

$$u = 0 \quad \text{on } \partial\Omega_t \times (0, T), \tag{6.3b}$$

$$\boldsymbol{u}(\boldsymbol{x}_0, 0) = \boldsymbol{u}_0 \quad \text{in } \Omega_0 \times \{0\}. \tag{6.3c}$$

The funcional space

$$\mathcal{V}(\Omega_t) := \left\{ v : \Omega_t \to \mathbb{R}, v = \hat{v} \circ \mathcal{A}_t^{-1}, \hat{v} \in H_0^1(\Omega_0) \right\}, \quad t \in (0, T),$$

allows us to write (6.3) in its variational form. The variational problem reads: find $u(t) \in \mathcal{V}(\Omega_t)$ for all $t \in (0, T)$ such that

$$\left(\frac{\partial u(t)}{\partial t}, v \right)_{\Omega_t} + \nu \left(\nabla u(t), \nabla v \right)_{\Omega_t} + \left((\boldsymbol{a} - \boldsymbol{w}(t)) \cdot \nabla u(t), v \right)_{\Omega_t} = \langle f(t), v \rangle_{\Omega_t}, \tag{6.4}$$

$\forall v \in \mathcal{V}(\Omega_t)$, where $(\cdot, \cdot)_{\Omega_t}$ stands for the $L^2(\Omega_t)$ inner product and $\langle \cdot, \cdot \rangle_{\Omega_t}$ for the duality pairing in $H^{-1}(\Omega_t) \times H_0^1(\Omega_t)$.

Let us re-scale the time variable as $t \leftarrow t/T$, so that the new time interval is $[0, 1]$ and the coefficient $1/T$ has to be inserted in front of the time derivatives. The reason of this change is to display which terms in the stability and convergence results disapear as $T \to \infty$, that is, the long time behavior. After re-scaling, problem (6.4) is transformed to,

$$\frac{1}{T}\left(\frac{\partial u(t)}{\partial t}, v \right)_{\Omega_t} + \nu \left(\nabla u(t), \nabla v \right)_{\Omega_t} + \left((\boldsymbol{a} - \boldsymbol{w}(t)) \cdot \nabla u(t), v \right)_{\Omega_t} = \langle f(t), v \rangle_{\Omega_t}, \tag{6.5}$$

and now the domain velocity is,

$$\boldsymbol{w}(\boldsymbol{x}, t) = \frac{1}{T} \left.\frac{\partial \boldsymbol{x}}{\partial t}\right|_{\boldsymbol{x}_0}. \tag{6.6}$$

We take into account this re-scaling in property (6.2), which now reads

$$\frac{1}{T}\frac{d}{dt}\int_{\Omega_t} v\,\mathrm{d}\Omega = \int_{\Omega_t} v \nabla \cdot \boldsymbol{w}\,\mathrm{d}\Omega. \tag{6.7}$$

6.2.2 The semi-discrete problem in time

We refer to Section 1.5 for the introduction of the notation and time integration schemes used for discretization carried out below.

Let us discretize problem (6.5) in time, once t has been normalized. We assume the force term is continuous in time and denote the time level by a superscript. We start using the BDF1 time integration scheme. It leads to the following problem: for $= 0, 1, ..., N-1$, given u^n, find $u^{n+1} \in \mathcal{V}(\Omega_{t^{n+1}})$ such that

$$\begin{aligned}\frac{1}{T}\left(u^{n+1}-u^n, v^{n+1}\right)_{\Omega_{t^{n+1}}} + \delta t\nu\left(\nabla u^{n+1}, \nabla v^{n+1}\right)_{\Omega_{t^{n+1}}} \\ + \delta t\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla u^{n+1}, v^{n+1}\right)_{\Omega_{t^{n+1}}} = \delta t\langle f^{n+1}, v^{n+1}\rangle_{\Omega_{t^{n+1}}},\end{aligned} \tag{6.8}$$

with $u^0 = u_0$ in $L^2(\Omega_0)$.

Furthermore, we discretize in time the ALE mapping using a linear interpolation. The discretized ALE mapping $\mathcal{A}_{\delta t}^{n+1}$ is defined for a given time slab $[t^n, t^{n+1}]$ as

$$\mathcal{A}_{\delta t}^{n+1}(\boldsymbol{x}_0, t) = \frac{t-t^n}{\delta t}\mathcal{A}_{t^{n+1}}(\boldsymbol{x}_0) + \frac{t^{n+1}-t}{\delta t}\mathcal{A}_{t^n}(\boldsymbol{x}_0).$$

Thus, the mesh velocity is constant on each time step and given by

$$\hat{\boldsymbol{w}}^{n+1}(\boldsymbol{x}_0) = \frac{\mathcal{A}_{t^{n+1}}(\boldsymbol{x}_0) - \mathcal{A}_{t^n}(\boldsymbol{x}_0)}{T\delta t},$$

and $\boldsymbol{w}^{n+1}(\boldsymbol{x}, t) = \hat{\boldsymbol{w}}^{n+1}((\mathcal{A}_{\delta t}^{n+1})^{-1}(\boldsymbol{x}))$ for $t \in (t^n, t^{n+1}]$. Equation (6.8) with this mesh velocity defines the BDF1-BDF1$_{\delta t}$ method. Note that the superscript $n+1$ in $\boldsymbol{w}$ denotes that it varies with time within the time interval $(t^n, t^{n+1}]$ where it is defined. However, in what follows we will simply denote $\boldsymbol{w}^{n+1} \equiv \boldsymbol{w}^{n+1}(\boldsymbol{x}, t^{n+1})$. Since $\mathcal{A}_{t^{n+1}}^{n+1} = \mathcal{A}_{t^{n+1}}$, we will write $\boldsymbol{w}^{n+1}(\boldsymbol{x}, t^{n+1}) = \hat{\boldsymbol{w}}^{n+1}(\mathcal{A}_{t^{n+1}}^{-1}(\boldsymbol{x}))$ or, for $\boldsymbol{x}$ arbitrary, $\boldsymbol{w}^{n+1} = \hat{\boldsymbol{w}}^{n+1} \circ \mathcal{A}_{t^{n+1}}^{-1}$.

For the numerical analysis we rewrite the transient problem using a different setting. The sequence of problems (6.8) can be written in a unified manner as: find a sequence $U = \{u^0, u^1, u^2, ..., u^N\}$ such that

$$B(U, V) = L(V) \tag{6.9}$$

for all sequences V, where

$$\begin{aligned}B(U,V) := \frac{1}{2T}\left(u^0, v^0\right)_{\Omega_0} + \sum_{n=0}^{N-1}\Bigg[\frac{1}{T}\left(\delta u^{n+1}, v^{n+1}\right)_{\Omega_{t^{n+1}}} + \delta t\nu\left(\nabla u^{n+1}, \nabla v^{n+1}\right)_{\Omega_{t^{n+1}}} \\ + \delta t\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla u^{n+1}, v^{n+1}\right)_{\Omega_{t^{n+1}}}\Bigg],\end{aligned} \tag{6.10}$$

$$L(V) := \frac{1}{2T}\left(u^0, v^0\right)_{\Omega_0} + \sum_{n=0}^{N-1}\delta t\langle f^{n+1}, v^{n+1}\rangle_{\Omega_{t^{n+1}}}. \tag{6.11}$$

Observe that the initial condition has been embedded in the variational problem.

In order to reach second order accuracy in time, the BDF2 integration scheme is used. It leads to the following time discretization of (6.5):

$$\begin{aligned}\frac{1}{2T}\left(3u^{n+1}-4u^n+u^{n-1}, v^{n+1}\right)_{\Omega_{t^{n+1}}} + \delta t\nu\left(\nabla u^{n+1}, \nabla v^{n+1}\right)_{\Omega_{t^{n+1}}} \\ + \delta t\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla u^{n+1}, v^{n+1}\right)_{\Omega_{t^{n+1}}} = \delta t\langle f^{n+1}, v^{n+1}\rangle_{\Omega_{t^{n+1}}}.\end{aligned} \tag{6.12}$$

This problem has to be initialized. For instance, we can obtain u^1 with (6.8) and $u^0 = u_0$ in $L^2(\Omega_0)$ keeping the order of convergence of the method. In order to keep this accuracy, a quadratic interpolation is used to approximate the ALE mapping. For a given time slab $[t^n, t^{n+1}]$, this interpolation is given by

$$\begin{aligned}\mathcal{A}_{\delta t}^{n+1}(\boldsymbol{x}_0, t) =& \frac{(t-t^n)(t-t^{n-1})}{2\delta t^2}\mathcal{A}_{t^{n+1}}(\boldsymbol{x}_0) \\ &- \frac{(t-t^{n+1})(t-t^{n-1})}{2\delta t^2}\mathcal{A}_{t^n}(\boldsymbol{x}_0) + \frac{(t-t^{n+1})(t-t^n)}{2\delta t^2}\mathcal{A}_{t^{n-1}}(\boldsymbol{x}_0).\end{aligned}$$

Thus, the mesh velocity on each time step is linear in time and given by

$$\begin{aligned}\hat{\boldsymbol{w}}^{n+1}(\boldsymbol{x}_0, t) =& \frac{2t-t^n-t^{n-1}}{2T\delta t^2}\mathcal{A}_{t^{n+1}}(\boldsymbol{x}_0) \\ &- \frac{2t-t^{n+1}-t^{n-1}}{2T\delta t^2}\mathcal{A}_{t^n}(\boldsymbol{x}_0) + \frac{2t-t^{n+1}-t^n}{2T\delta t^2}\mathcal{A}_{t^{n-1}}(\boldsymbol{x}_0),\end{aligned}$$

and $\boldsymbol{w}^{n+1}(\boldsymbol{x}, t) = \hat{\boldsymbol{w}}^{n+1}((\mathcal{A}_{\delta t}^{n+1})^{-1}(\boldsymbol{x}), t)$ for $t \in (t^n, t^{n+1}]$. It is easily checked that at t^{n+1} we recover the BDF2 formula for the mesh velocity.

Again, we can rewrite the transient problem as an abstract 'variarional' problem (6.9), now with the bilinear form

$$\begin{aligned}B(U,V) =& \frac{1}{T}\left(u^1-u^0, v^1\right)_{\Omega_{t^1}} + \delta t\nu\left(\nabla u^1, \nabla v^1\right)_{\Omega_{t^1}} + \delta t\left((\boldsymbol{a}-\boldsymbol{w}^1)\cdot\nabla u^1, v^1\right)_{\Omega_{t^1}} \\ &+ \frac{1}{2T}\left(u^0, v^0\right)_{\Omega_0} + \sum_{n=1}^{N-1}\Bigg[\frac{1}{2T}\left(3u^{n+1}-4u^n+u^{n-1}, v^{n+1}\right)_{\Omega_{t^{n+1}}} \\ &+\delta t\nu\left(\nabla u^{n+1}, \nabla v^{n+1}\right)_{\Omega_{t^{n+1}}} + \delta t\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla u^{n+1}, v^{n+1}\right)_{\Omega_{t^{n+1}}}\Bigg], \qquad (6.13)\end{aligned}$$

and the linear form

$$L(V) := \frac{1}{2T}\left(u^0, v^0\right)_{\Omega_0} + \sum_{n=0}^{N-1}\delta t\langle f^{n+1}, v^{n+1}\rangle_{\Omega_{t^{n+1}}}. \qquad (6.14)$$

We end this subsection giving the norm for which stability and convergence results are obtained in Section 6.3 for the previous semi-discrete problems, which is

$$|||V|||^2 = \frac{1}{T}\sup_{n\in[0,N]}\|v^n\|^2_{L^2(\Omega_{t^n})} + \sum_{n=0}^{N-1}\delta t\nu\left\|\nabla v^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})}. \qquad (6.15)$$

Given a normed space X, for $1 \le q < \infty$ we define the space $\ell^q(X)$ as that of sequences $V = \{v^n\}_{n=0}^N$ such that $\sum_{n=0}^N \delta t\|v^n\|_X^q < \infty$, and $\ell^\infty(X)$ the space of sequences such that $\sup_{n=0,\dots,N}\|v^n\|_X < \infty$. With this notation, the norm defined in (6.15) can be considered that of $\ell^\infty(L^2(\Omega_t)) \cap \ell^2(H_0^1(\Omega_t))$. Here, the subscript t has to be understood as t^n for the n-th component of the sequence.

6.2.3 The fully discrete problem

At this point we treat the space discretization of systems (6.8) and (6.12). The BDF1-BDF1-OSS$_{\delta t,h}$ reads as follows: for $n = 0, 1, ..., N-1$, given u_h^n, find $u_h^{n+1} \in \mathcal{V}_h(\Omega_t)$ such

that

$$\begin{aligned}\frac{1}{T}\left(u_h^{n+1}-u_h^n,v_h^{n+1}\right)_{\Omega_{t^{n+1}}}&+\delta t\nu\left(\nabla u_h^{n+1},\nabla v_h^{n+1}\right)_{\Omega_{t^{n+1}}}\\&+\delta t\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla u_h^{n+1},v_h^{n+1}\right)_{\Omega_{t^{n+1}}}\\&+\delta t\left(\Pi_h^{\perp}\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla u_h^{n+1}\right),\tau^{n+1}(\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v_h^{n+1}\right)_{\Omega_{t^{n+1}}}\\&=\delta t\langle f^{n+1},v_h^{n+1}\rangle_{\Omega_{t^{n+1}}},\end{aligned}\tag{6.16}$$

where $\mathcal{V}_h(\Omega_t)$ is a finite element approximation space of $\mathcal{V}(\Omega_t)$, τ^{n+1} is a mesh dependent parameter, that we will call *stabilization parameter*, whose expression is detailed later, and $\Pi_h^{\perp}(\cdot) =: Id(\cdot)-\Pi_h(\cdot)$, with Id the identity in $L^2(\Omega_t)$ and $\Pi_h(\cdot)$ the L^2-projection onto this finite element space (and therefore $\Pi_h^{\perp}(\cdot)$ is the projection orthogonal to the finite element space). The description and motivation of this formulation, that we call *Orthogonal Subgrid Scale* (OSS) stabilization, has been explained in Chapter 2.

We refer to Section 1.3 for the ingredients needed for the space discretization. Let Θ_h^t be a finite element partition of the domain Ω_t in a family of elements $\{K_e\}_{e=1}^{n_{el}}$, n_{el} being the number of elements. The finite element spaces introduced before and that we will use in the following are:

$$\begin{aligned}&\mathcal{V}_h(\Omega_0)=\{\widehat{v}_h\in\mathcal{C}^0(\Omega_0)\mid\widehat{v}_h|_K=\tilde{v}\circ F_K^{-1},\ \tilde{v}\in R_p(\tilde{K}),\ K\in\Theta_h^t\},\\&\mathcal{V}_{h,0}(\Omega_0)=\{v_h\in\mathcal{V}_h(\Omega_0)\mid v_h|_{\partial\Omega_0}=0\},\\&\mathcal{V}_h(\Omega_t)=\{v_h\in\mathcal{C}^0(\Omega_t)\mid v_h=\widehat{v}_h\circ\mathcal{A}_t^{-1},\ \widehat{v}_h\in\mathcal{V}_h(\Omega_0)\},\\&\mathcal{V}_{h,0}(\Omega_t)=\{v_h\in\mathcal{C}^0(\Omega_t)\mid v_h=\widehat{v}_h\circ\mathcal{A}_t^{-1},\ \widehat{v}_h\in\mathcal{V}_{h,0}(\Omega_0)\}.\end{aligned}$$

Moreover, Θ_h^t is assumed to be quasi-uniform (see Section 1.3). This will simplify the analysis and, in particular, will allow us to use stabilization parameters constant in space.

Let us note that in practical applications $\mathcal{A}_{t^{n+1}}$ maps Θ_h^0 onto Θ_h^{n+1}. Therefore, it is easily checked the fact that $\boldsymbol{w}^{n+1}\in(\mathcal{V}_h(\Omega_{t^{n+1}}))^d$. In the following we will not distinguish between $\boldsymbol{w}^{n+1}$ and $\boldsymbol{w}_h^{n+1}$.

Also in this case we can write the problem using a *variational* formalism. The fully discrete sequence of problems given by (6.16) can be written as: find a sequence $U_h=\{u_h^0,u_h^1,...,u_h^N\}$ such that

$$B_h(U_h,V_h)=L(V_h),\tag{6.17}$$

for all sequences V_h, with the bilinear form B_h given by

$$\begin{aligned}B_h(U_h,V_h)=&\frac{1}{2T}\left(u_h^0,v_h^0\right)_{\Omega_0}\\&+\sum_{n=0}^{N-1}\left[\frac{1}{T}\left(u_h^{n+1}-u_h^n,v_h^{n+1}\right)_{\Omega_{t^{n+1}}}+b_h\left(\boldsymbol{w}^{n+1};u_h^{n+1},v_h^{n+1}\right)_{\Omega_{t^{n+1}}}\right],\end{aligned}\tag{6.18}$$

where b_h is defined as

$$\begin{aligned}b_h\left(\boldsymbol{w}^{n+1};u_h^{n+1},v_h^{n+1}\right)_{\Omega_{t^{n+1}}}&=\delta t\nu\left(\nabla u_h^{n+1},\nabla v_h^{n+1}\right)_{\Omega_{t^{n+1}}}\\&+\delta t\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla u_h^{n+1},v_h^{n+1}\right)_{\Omega_{t^{n+1}}}\\&+\delta t\left(\Pi_h^{\perp}\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla u_h^{n+1}\right),\tau^{n+1}(\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v_h^{n+1}\right)_{\Omega_{t^{n+1}}}.\end{aligned}\tag{6.19}$$

The OSS method modifies the discretized equation of the classical Galerkin method introducing the last term, which enhances the stability of the original method. The value of the stabilization parameter τ^{n+1} has been justified in [48]. In an ALE framework it depends on the difference between the advection velocity $\boldsymbol{a}$ and the mesh velocity $\boldsymbol{w}$. The expression we use is

$$\tau^{n+1} = \left(c_1 \frac{\nu}{h^2} + c_2 \frac{\|\boldsymbol{a} - \boldsymbol{w}\|_{L^\infty(\Omega_{t^{n+1}})}}{h} \right)^{-1}, \tag{6.20}$$

that is constant in space. Here, c_1 and c_2 are algorithmic constants that depend on the order of the finite element interpolation. As will be shown later (see (6.43)), they are related to the constant C_{inv} in the inverse estimates introduced in (6.37).

As in [55], we will make further assumptions. We assume that for each n the parameter τ^n satisfies

$$\tau^n \leq CT\delta t, \tag{6.21}$$

which in particular implies that we can not let $\delta t \to 0$ without refining the finite element mesh.

For the space discretization of the second order method (6.12), the bilinear form is given by

$$\begin{aligned} &B_h\left(U_h, V_h\right) \\ &\quad = \sum_{n=1}^{N-1} \left[\frac{1}{2T} \left(3u_h^{n+1} - 4u_h^n + u_h^{n-1}, v_h^{n+1} \right)_{\Omega_{t^{n+1}}} + b_h\left(\boldsymbol{w}^{n+1}; u_h^{n+1}, v_h^{n+1} \right)_{\Omega_{t^{n+1}}} \right] \\ &\qquad + \frac{1}{T} \left(u_h^1 - u_h^0, v_h^1 \right)_{\Omega_{t^1}} + b_h\left(\boldsymbol{w}^1; u_h^1, v_h^1 \right)_{\Omega_{t^1}} + \frac{1}{T} \left(u_h^0, v_h^0 \right)_{\Omega_0}. \end{aligned} \tag{6.22}$$

We end this section with two norms useful in the following numerical analysis. The first is a norm that we will call *weak*, and given by

$$\begin{aligned} |||V|||_w^2 =& \frac{1}{T} \sup_{n\in[0,N]} \|v^n\|^2_{L^2(\Omega_{t^n})} + \sum_{n=0}^{N-1} \delta t\nu \left\|\nabla v^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})} \\ &+ \sum_{n=0}^{N-1} \delta t\tau^{n+1} \left\| \Pi_h^\perp \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla v^{n+1} \right) \right\|^2_{L^2(\Omega_{t^{n+1}})}. \end{aligned}$$

Observe that only the orthogonal projection of the convective term appears. The full convective term appears in the norm that we will call *strong*, given by

$$\begin{aligned} |||V|||_s^2 =& \frac{1}{T} \sup_{n\in[0,N]} \|v^n\|^2_{L^2(\Omega_{t^n})} + \sum_{n=0}^{N-1} \delta t\nu \left\|\nabla v^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})} \\ &+ \sum_{n=0}^{N-1} \delta t\tau^{n+1} \left\| (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla v^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})} \\ =& |||V|||_w^2 + \sum_{n=0}^{N-1} \delta t\tau^{n+1} \left\| \Pi_h \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla v^{n+1} \right) \right\|^2_{L^2(\Omega_{t^{n+1}})}. \end{aligned}$$

6.3 Analysis of the semi-discrete problem

In this section we analyze problems BDF1-BDF1$_{\delta t}$ and BDF2-BDF2$_{\delta t}$. In both cases, stability and error estimates will be given. We denote by C a positive constant, possibly with different values at different appearances.

6.3.1 Analysis of BDF1-BDF1$_{\delta t}$

Let us define by $U_{\text{ex}} = \{u_0, u(t^1), u(t^2), ..., u(t^N)\}$ the sequence of solutions of the contiunous problem (6.4) and $U = \{u^0, u^1, u^2, ..., u^N\}$ the sequence of solutions of the semi-discrete problem (in time) (6.9)-(6.11). We start obtaining a stability result for this method. With this aim, first we prove that the bilinear form (6.10) that governs the semi-discrete problem is coercive.

Theorem 6.1 (Coercivity)**.** *There exists δt^1_{cr} such that for $0 < \delta t < \delta t^1_{\text{cr}}$ the bilinear form $B(\cdot,\cdot)$ defined in (6.10) is coercive, that is, for every sequence $V = \{v^n\}_{n=0}^N$, with $v^n \in \mathcal{V}(\Omega_{t^n})$,*

$$B(V,V) \geq \beta_1 |||V|||^2$$

for a certain constant $\beta_1 \geq 0$ independent of h.

Proof. We know, from the definition of the bilinear form, that

$$\begin{aligned} B(V,V) = \sum_{n=0}^{N-1} \Bigg[\frac{1}{T} \left(v^{n+1} - v^n, v^{n+1}\right)_{\Omega_{t^{n+1}}} + \delta t \nu \left\|\nabla v^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})} \\ + \delta t \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla v^{n+1}, v^{n+1}\right)_{\Omega_{t^{n+1}}} \Bigg] + \frac{1}{2T} \left\|v^0\right\|^2_{L^2(\Omega_0)}. \end{aligned}$$

We can rewrite the term coming from the time derivative as follows:

$$\begin{aligned} &\frac{1}{T} \left(v^{n+1} - v^n, v^{n+1}\right)_{\Omega_{t^{n+1}}} \\ &\quad = \frac{1}{2T} \left[\left\|v^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})} - \left\|v^n\right\|^2_{L^2(\Omega_{t^{n+1}})} + \left\|v^{n+1} - v^n\right\|^2_{L^2(\Omega_{t^{n+1}})} \right]. \end{aligned}$$

Integrating equation (6.7) from t^n to t^{n+1} for the function v^n we get

$$\frac{1}{T} \left\|v^n\right\|^2_{L^2(\Omega_{t^{n+1}})} = \frac{1}{T} \left\|v^n\right\|^2_{L^2(\Omega_{t^n})} + \int_{t^n}^{t^{n+1}} \int_{\Omega_s} (\nabla \cdot \boldsymbol{w}^{n+1})(v^n)^2 \, \mathrm{d}\Omega \, \mathrm{d}s,$$

where we have profited from the fact that the discrete mesh velocity is constant at every time step. On the other hand, due to the fact that the convective velocity $\boldsymbol{a}$ is divergence-free, we get

$$\begin{aligned} \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla v^{n+1}, v^{n+1}\right)_{\Omega_{t^{n+1}}} &= -\frac{1}{2} \int_{\Omega_{t^{n+1}}} \boldsymbol{w}^{n+1} \cdot \nabla (v^{n+1})^2 \, \mathrm{d}\Omega \\ &= \frac{1}{2} \int_{\Omega_{t^{n+1}}} (\nabla \cdot \boldsymbol{w}^{n+1})(v^{n+1})^2 \, \mathrm{d}\Omega. \end{aligned}$$

We bound the terms associated to the mesh velocity as follows:

$$\begin{aligned} &\int_{t^n}^{t^{n+1}} \int_{\Omega_s} (\nabla \cdot \boldsymbol{w}^{n+1})(v^n)^2 \, \mathrm{d}\Omega \, \mathrm{d}s \\ &\qquad \leq \delta t \sup_{s \in (t^n, t^{n+1})} \left\| J_{\mathcal{A}_{t^{n+1},s}} \nabla \cdot \boldsymbol{w}^{n+1} \right\|_{L^\infty(\Omega_s)} \left\|v^n\right\|^2_{L^2(\Omega_{t^{n+1}})}, \end{aligned}$$

$$-\delta t \int_{\Omega_{t^{n+1}}} \boldsymbol{w}^{n+1} \cdot \nabla (v^{n+1})^2 \, \mathrm{d}\Omega = \delta t \int_{\Omega_{t^{n+1}}} (\nabla \cdot \boldsymbol{w}^{n+1})(v^{n+1})^2 \, \mathrm{d}\Omega$$
$$\leq \delta t \left\| \nabla \cdot \boldsymbol{w}^{n+1} \right\|_{L^\infty(\Omega_{t^{n+1}})} \left\| v^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})}.$$

Let us define the parameters

$$\gamma_1^{n+1} = T \sup_{s \in (t^n, t^{n+1})} \left\| J_{\mathcal{A}_{t^{n+1},s}} \nabla \cdot \boldsymbol{w}^{n+1} \right\|_{L^\infty(\Omega_s)}, \tag{6.23}$$

for $n = -1, ..., N-2$ and $\gamma_1^N = 0$, together with

$$\gamma_2^{n+1} = T \left\| \nabla \cdot \boldsymbol{w}^n \right\|_{L^\infty(\Omega_{t^{n+1}})} \tag{6.24}$$

for $n = 0, ..., N-1$ and $\gamma_2^0 = 0$.

With the inequalities just proved we can easily obtain that

$$B(V,V) + \frac{1}{2T} \sum_{n=-1}^{N-1} \delta t (\gamma_1^{n+1} + \gamma_2^{n+1}) \left\| v^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})}$$
$$\geq \sup_{n \in [-1, N-1]} \frac{1}{2T} \left\| v^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})} + \sum_{n=0}^{N-1} 2\delta t \nu \left\| \nabla v^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})}.$$

If the maximum of $\|v^n\|_{L^2(\Omega_{t^n})}$ is achieved at $n = N_m$, the sequence

$$\{v^0, v^1, ..., v^{N_m}, 0, ..., 0\}$$

has to be added to the test sequence. Sometimes along this chapter we obtain the maximum using this technique. Invoking the Gronwall lemma (see [98]), we can absorb the second term of the left hand side with the first one of the right hand side for a δt small enough. More precisely, the time step must be such that

$$\delta t < \frac{1}{\sup_{n \in [0,N]} (\gamma_1^n + \gamma_2^n)} =: \delta t_{\mathrm{cr}}^1.$$

We note that this is the time step size of the *normalized* problem in time. The original δt_{cr}^1 does not depend on T anymore.□

This result, together with the continuity of $L(\cdot)$ proved in next lemma, will lead us to a classical stability bound.

Lemma 6.1 (Continuity)**.** *The following inequality holds:*

$$L(V) \leq \sum_{n=0}^{N-1} \frac{\delta t}{2\beta\nu} \left\| f^{n+1} \right\|^2_{H^{-1}(\Omega_{t^{n+1}})} + \sum_{n=0}^{N-1} \frac{\delta t \beta \nu}{2} \left\| \nabla v^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})}$$

for all $\beta > 0$.

Proof. The right hand side has the following expression:

$$L(V) = \sum_{n=0}^{N-1} \delta t \langle f^{n+1}, v^{n+1} \rangle_{\Omega_{t^{n+1}}}.$$

From this expression we have that

$$L(V) \leq \left(\sum_{n=0}^{N-1} \frac{\delta t}{\nu} \left\| f^{n+1} \right\|^2_{H^{-1}(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}} \left(\sum_{n=0}^{N-1} \delta t \nu \left\| \nabla v^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}}.$$

The proof is finished invoking Young's inequality.□

From Theorem 6.1 and Lemma 6.1 the following stability result is straightforward.

Corollary 6.1 (Stability)**.** *There exists δt^1_{cr} such that, for $0 < \delta t < \delta t^1_{\text{cr}}$, the sequence U, solution of problem (6.9)-(6.11), is bounded as follows:*

$$|||U|||^2 \leq C \sum_{n=0}^{N-1} \frac{\delta t}{\nu} \left\| f^{n+1} \right\|^2_{H^{-1}(\Omega_{t^{n+1}})}.$$

Remark 6.1. The BDF1 method is unconditionally stable for fixed domains. However, for moving domains this property is not maintained anymore. In this case only conditional stability can be proved, with the critical time step value obtained above.

The next task is to obtain an optimal convergence result. In the following theorem, relying on the stability properties proved in Corollary 6.1, *optimal* error estimates are obtained. We denote by $e^{n+1} := u(t^{n+1}) - u^{n+1}$ the error introduced by the time integration at time t^{n+1}, and by $E := U_{\text{ex}} - U$ the sequence of these errors.

Theorem 6.2 (Convergence)**.** *There exists δt^1_{cr} such that, for $0 < \delta t < \delta t^1_{\text{cr}}$, the sequence of errors $E = U_{\text{ex}} - U$ satisfies the following error estimate:*

$$|||E|||^2 \leq C \frac{\delta t^2}{T} \sum_{n=0}^{N-1} \delta t \left(\left\| \left. \frac{\partial^2 u}{\partial t^2} \right|_{\boldsymbol{x}_0} \right\|^2_{L^2(\Omega_{t^{n+1}})} + \sup_{s \in (t^n, t^{n+1})} \left\| \left| \frac{\partial^2 \mathcal{A}_s}{\partial t^2} \right|^2 \right\|_{L^\infty(\Omega_0)} \left\| u^{n+1} \right\|^2_{H^1(\Omega_{t^{n+1}})} \right). \tag{6.25}$$

Proof. We start taking the exact solution sequence U_{ex} in the bilinear form. We get:

$$\begin{aligned} B(U_{\text{ex}}, V) =& L(V) + \sum_{n=0}^{N-1} \frac{1}{T} \left(u(t^{n+1}) - u(t^n) - \delta t \left. \frac{\partial u}{\partial t} \right|_{t^{n+1}}, v^{n+1} \right)_{\Omega_{t^{n+1}}} \\ & - \sum_{n=0}^{N-1} \delta t \left((\boldsymbol{w}^{n+1} - \boldsymbol{w}(t^{n+1})) \cdot \nabla u(t^{n+1}), v^{n+1} \right)_{\Omega_{t^{n+1}}}. \end{aligned}$$

We subtract the equation for the semi-discrete sequence of solutions to the previous equations and arrive to

$$\begin{aligned} B(U - U_{\text{ex}}, V) =& - \sum_{n=0}^{N-1} \frac{1}{T} \left(u(t^{n+1}) - u(t^n) - \delta t \left. \frac{\partial u}{\partial t} \right|_{t^{n+1}}, v^{n+1} \right)_{\Omega_{t^{n+1}}} \\ & + \sum_{n=0}^{N-1} \delta t \left((\boldsymbol{w}^{n+1} - \boldsymbol{w}(t^{n+1})) \cdot \nabla u(t^{n+1}), v^{n+1} \right)_{\Omega_{t^{n+1}}}. \end{aligned}$$

We test the previous equation with $V = U - U_{\text{ex}} = E$, obtaining

$$\begin{aligned} B(E,E) = &- \sum_{n=0}^{N-1} \frac{1}{T} \left(u(t^{n+1}) - u(t^n) - \delta t \left.\frac{\partial u}{\partial t}\right|_{t^{n+1}}, e^{n+1} \right)_{\Omega_{t^{n+1}}} \\ &+ \sum_{n=0}^{N-1} \delta t \left((\boldsymbol{w}^{n+1} - \boldsymbol{w}(t^{n+1})) \cdot \nabla u(t^{n+1}), e^{n+1} \right)_{\Omega_{t^{n+1}}}. \end{aligned}$$

Exploiting the fact that the bilinear form is coercive the remaining ingredient is an appropriate bound for the error terms associated to the time discretization. Let us start with the terms related to the time derivative. We use the following Taylor formula for u:

$$\frac{u(\boldsymbol{x}_0, t^{n+1}) - u(\boldsymbol{x}_0, t^n)}{T\delta t} - \frac{1}{T} \left.\frac{\partial u}{\partial t}\right|_{\boldsymbol{x}_0} (t^{n+1}) = -\frac{1}{T\delta t} \int_{t^n}^{t^{n+1}} (s - t^n) \left.\frac{\partial^2 u}{\partial t^2}\right|_{\boldsymbol{x}_0} (s)\, \mathrm{d}s, \quad (6.26)$$

and for the mesh velocity

$$\boldsymbol{w}^{n+1} - \boldsymbol{w}(t^{n+1}) = -\frac{1}{T\delta t} \left(\int_{t^n}^{t^{n+1}} (s - t^n) \frac{\partial^2 \mathcal{A}_s}{\partial t^2}\, \mathrm{d}s \right) \circ \mathcal{A}_{t^{n+1}}^{-1}. \quad (6.27)$$

As explained in Section 2, it is understood with this notation that this equality holds for arbitrary $\boldsymbol{x} \in \Omega_t$.

With (6.26) we get a bound for the term associated to the time derivative of u as follows:

$$\begin{aligned} &\int_{\Omega_{t^{n+1}}} e^{n+1} \cdot \left(\int_{t^n}^{t^{n+1}} (s - t^n) \left.\frac{\partial^2 u}{\partial t^2}\right|_{\boldsymbol{x}_0} (\boldsymbol{x}_0, s)\, \mathrm{d}s \right) \circ \mathcal{A}_{t^{n+1}}^{-1}\, \mathrm{d}\Omega \\ &\leq \int_{t^n}^{t^{n+1}} \int_{\Omega_0} J_{\mathcal{A}_{t^{n+1}}} (s - t^n) \widehat{e^{n+1}} \widehat{\frac{\partial^2 u}{\partial t^2}}\, \mathrm{d}\Omega\, \mathrm{d}s \\ &\leq \left(\int_{t^n}^{t^{n+1}} (s - t^n)^2 \left\| e^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}} \\ &\qquad \times \left(\int_{t^n}^{t^{n+1}} \int_{\Omega_0} J_{\mathcal{A}_{t^{n+1}}} \left(\widehat{\frac{\partial^2 u}{\partial t^2}} \right)^2 \mathrm{d}\Omega\, \mathrm{d}s \right)^{\frac{1}{2}} \\ &\leq \frac{\beta_1 \delta t}{2} \left\| e^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})} + C\delta t^3 \left\| \left.\frac{\partial^2 u}{\partial t^2}\right|_{\boldsymbol{x}_0} \right\|^2_{L^2(\Omega_{t^{n+1}})}, \end{aligned}$$

where β_1 is the coercivity constant introduced in Theorem 6.1. Similarly, using (6.27) for the term related to the time derivative of the mapping, we get

$$\begin{aligned} &- \int_{\Omega_{t^{n+1}}} e^{n+1} \left(\int_{t^n}^{t^{n+1}} (s - t^n) \frac{\partial^2 \mathcal{A}_s}{\partial t^2}\, \mathrm{d}s \right) \circ \mathcal{A}_{t^{n+1}}^{-1} \cdot \nabla u^{n+1}\, \mathrm{d}\Omega \\ &\leq \int_{t^n}^{t^{n+1}} \int_{\Omega_0} J_{\mathcal{A}_{t^{n+1}}} (s - t^n) \widehat{e^{n+1}} \frac{\partial^2 \mathcal{A}_s}{\partial t^2} \cdot \widehat{\nabla u^{n+1}}\, \mathrm{d}\Omega\, \mathrm{d}s \\ &\leq \left(\int_{t^n}^{t^{n+1}} (s - t^n)^2 \left\| e^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})}\, \mathrm{d}s \right)^{\frac{1}{2}} \end{aligned}$$

$$\times \left(\int_{t^n}^{t^{n+1}} \left\| \left| \frac{\partial^2 \mathcal{A}_s}{\partial t^2} \right|^2 \right\|_{L^\infty(\Omega_0)} \left\| u^{n+1} \right\|^2_{H^1(\Omega_{t^{n+1}})} \, \mathrm{d}s \right)^{\frac{1}{2}}$$

$$\leq \frac{\beta_1 \delta t}{2} \left\| e^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})} + C\delta t^3 \sup_{s \in (t^n, t^{n+1})} \left\| \left| \frac{\partial^2 \mathcal{A}_s}{\partial t^2} \right|^2 \right\|_{L^\infty(\Omega_0)} \left\| u^{n+1} \right\|^2_{H^1(\Omega_{t^{n+1}})}.$$

With these results we can write

$$B(E,E) \leq \frac{1}{T} \sum_{n=0}^{N-1} \left(\delta t \beta_1 \left\| e^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})} + C\delta t^3 \left\| \left. \frac{\partial^2 u}{\partial t^2} \right|_{\boldsymbol{x}_0} \right\|^2_{L^2(\Omega_{t^{n+1}})} \right.$$

$$\left. + C\delta t^3 \sup_{s \in (t^n, t^{n+1})} \left\| \left| \frac{\partial^2 \mathcal{A}_s}{\partial t^2} \right|^2 \right\|_{L^\infty(\Omega_0)} \left\| u^{n+1} \right\|^2_{H^1(\Omega_{t^{n+1}})} \right). \tag{6.28}$$

At this point we invoke the coercivity property of the bilinear form proved in Theorem 6.1. Thus, the first term of the right hand side in (6.28) can be absorbed using the Gronwall lemma . We note that in this case we can apply the Gronwall lemma without any extra condition over the time step size (see [98]). □

Clearly, the second term in the right-hand-side of (6.25) is bounded if the second time derivatives of the ALE mapping are uniformly bounded in $[0,T]$. In this case, its norm in the space $L^\infty(0,T;L^\infty(\Omega_0))$ can be taken out of the sum and the stability estimate of Corollary 6.1 allows us to bound the remaining term. However, we have kept expression (6.25) to display the structure of the error bound.

We conclude this subsection with an improved stability estimate:

Corollary 6.2 (Stability in $\ell^\infty(H^2(\Omega_t))$)**.** *Under the conditions of Theorem 6.2, suppose additionally that the right-hand-side of (6.25) is bounded, that $u \in L^\infty(0,T;H^2(\Omega_t))$ and that the domain Ω_t is such that $\Delta u \in L^2(\Omega_t)$ implies $u \in H^2(\Omega_t)$. Then, $U \in \ell^\infty(H^2(\Omega_t))$.*

Proof. At each time step we can write the error equation

$$\nu \Delta(u^{n+1} - u(t^{n+1})) = (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla(u^{n+1} - u(t^{n+1}))$$

$$+ (\boldsymbol{w}(t^{n+1}) - \boldsymbol{w}^{n+1}) \cdot \nabla u(t^{n+1}) + \frac{1}{\delta t}(u^{n+1} - u^n) - \left. \frac{\partial u}{\partial t} \right|_{t^{n+1}}.$$

By virtue of Theorem 6.2, all the terms in the right-hand-side are bounded in $L^2(\Omega_{t^{n+1}})$ for $n = 0, ..., N-1$. Since

$$\|\Delta u^{n+1}\|_{L^2(\Omega_{t^{n+1}})} \leq \|\Delta u^{n+1} - \Delta u(t^{n+1})\|_{L^2(\Omega_{t^{n+1}})} + \|\Delta u(t^{n+1})\|_{L^2(\Omega_{t^{n+1}})},$$

it follows that $\{\Delta u^{n+1}\}_{n=0}^{N-1} \in \ell^\infty(L^2(\Omega_t))$. The assumption on the domain Ω_t implies that $\{u^{n+1}\}_{n=0}^{N-1} \in \ell^\infty(H^2(\Omega_t))$.□

This justifies our strategy of first analyzing the problem semi-discretized in time and then the fully discrete problem. When we will require $U \in \ell^2(H^{p+1}(\Omega_t))$ to obtain optimal order of convergence in space, we know that at least for $p = 1$ this holds under the same condition on the domain Ω_t as for the sequence of solutions of the continuous problem, U_{ex}. It is well known that this condition on Ω_t holds for example if it is convex and polyhedral (see, for example, [81]).

6.3.2 Analysis of BDF2-BDF2$_{\delta t}$

For the second order method we follow the same procedure used above. In this case the problem that we analyze can be written using equation (6.9) together with the bilinear form (6.13) and the right hand side linear form (6.14), and we denote by $U = \{u^0, u^1, u^2, ..., u^N\}$ the sequence of solutions of this problem.

We start again proving that the corresponding bilinear form is coercive.

Theorem 6.3 (Coercivity)**.** *There exists δt^2_{cr} such that for $0 < \delta t < \delta t^2_{\text{cr}}$ the bilinear form $B(\cdot,\cdot)$ defined in (6.12) is coercive, that is, for every sequence $V = \{v^n\}_{n=0}^N$, with $v^n \in \mathcal{V}(\Omega_{t^n})$,*

$$B(V,V) \geq \beta_2 |||V|||^2$$

for a certain constant $\beta_2 \geq 0$ independent of h.

Proof. We know, from the definition of the bilinear form, that,

$$\begin{aligned} B(V,V) = &\sum_{n=0}^{N-1} \left[\delta t \left\| \nabla v^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})} + \delta t \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla v^{n+1}, v^{n+1} \right)_{\Omega_{t^{n+1}}} \right] \\ &+ \sum_{n=1}^{N-1} \frac{1}{2T} \left(3v^{n+1} - 4v^n + v^{n-1}, v^{n+1} \right)_{\Omega_{t^{n+1}}} + \frac{1}{T} \left(v^1 - v^0, v^1 \right)_{\Omega_{t^1}} \\ &+ \frac{1}{2T} \left\| v^0 \right\|^2_{L^2(\Omega_0)} . \end{aligned} \tag{6.29}$$

Integrating equation (6.7) from t^n to t^{n+1} for the functions v^n and $2v^n - v^{n-1}$, we can express the term corresponding to discrete time derivative as follows:

$$\begin{aligned} &\frac{1}{2T}(3v^{n+1} - 4v^n + v^{n-1}, 4v^{n+1})_{\Omega_{t^{n+1}}} \\ &= \frac{1}{T} \left(\left\| v^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})} - \left\| v^n \right\|^2_{L^2(\Omega_{t^n})} + \left\| 2v^{n+1} - v^n \right\|^2_{L^2(\Omega_{t^{n+1}})} \right. \\ &\quad \left. - \left\| 2v^n - v^{n-1} \right\|^2_{L^2(\Omega_{t^n})} + \left\| \delta^2 v^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})} \right) \\ &\quad + \int_{t^n}^{t^{n+1}} \int_{\Omega_s} (\nabla \cdot \boldsymbol{w}^{n+1}(s))(v^n)^2 \, \mathrm{d}\Omega \, \mathrm{d}s \\ &\quad + \int_{t^n}^{t^{n+1}} \int_{\Omega_s} (\nabla \cdot \boldsymbol{w}^{n+1}(s))(2v^n - v^{n-1})^2 \, \mathrm{d}\Omega \, \mathrm{d}s. \end{aligned} \tag{6.30}$$

The mesh velocity terms are bounded as follows:

$$\begin{aligned} &\int_{t^n}^{t^{n+1}} \int_{\Omega_s} (\nabla \cdot \boldsymbol{w}^{n+1}(s))(v^n)^2 \, \mathrm{d}\Omega \, \mathrm{d}s + \int_{t^n}^{t^{n+1}} \int_{\Omega_s} (\nabla \cdot \boldsymbol{w}^{n+1}(s))(2v^n - v^{n-1})^2 \, \mathrm{d}\Omega \, \mathrm{d}s \\ &\leq \delta t \sup_{s \in (t^n, t^{n+1})} \left\| J_{\mathcal{A}_{t^{n+1},s}} \nabla \cdot \boldsymbol{w}^{n+1}(s) \right\|_{L^\infty(\Omega_s)} \\ &\qquad \times \left(\left\| v^n \right\|^2_{L^2(\Omega_{t^{n+1}})} + \left\| 2v^n - v^{n-1} \right\|^2_{L^2(\Omega_{t^{n+1}})} \right). \end{aligned} \tag{6.31}$$

On the other hand, we can exploit the fact that the convective velocity $\boldsymbol{a}$ is divergence-free,

obtaining for the convective term that

$$\begin{aligned}\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v^{n+1},4v^{n+1}\right)_{\Omega_{t^{n+1}}} &= -2\delta t\int_{\Omega_{t^{n+1}}}\boldsymbol{w}^{n+1}\cdot\nabla(v^{n+1})^2\,\mathrm{d}\Omega\\ &= 2\delta t\int_{\Omega_{t^{n+1}}}(\nabla\cdot\boldsymbol{w}^{n+1})(v^{n+1})^2\,\mathrm{d}\Omega\\ &\leq 2\left\|\nabla\cdot\boldsymbol{w}^{n+1}\right\|_{L^\infty(\Omega_{t^{n+1}})}\left\|u_h^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})}. \end{aligned}\tag{6.32}$$

We use inequalities (6.30) and (6.31) in (6.29) and invoke again the Gronwall lemma. This leads to the desired bound for a time step size:

$$\delta t < \frac{1}{\sup_{n\in[0,N]}\left(\gamma_1^n+2\gamma_2^n\right)} =: \delta t^2_{\mathrm{cr}},$$

slightly different from the one obtained for the first order method. □

The previous theorem and Lemma 6.1 allow us to obtain the same stability result as for the previous case, stated in the next corollary.

Corollary 6.3 (Stability)**.** *There exists δt^2_{cr} such that for $0<\delta t<\delta t^2_{\mathrm{cr}}$ the sequence U solution of problem (6.9)-(6.13)-(6.14) is bounded as follows:*

$$|||U|||^2 \leq C\sum_{n=0}^{N-1}\frac{\delta t}{\nu}\left\|f^{n+1}\right\|^2_{H^{-1}(\Omega_{t^{n+1}})}.$$

Furthermore, we can obtain optimal error estimates under some regularity assumptions. For the sake of clearness we assume that the initialization is calculated exactly. It can be easily checked from Theorem 6.2 that the error introduced by the initialization is optimal.

Theorem 6.4 (Convergence)**.** *There exist δt^2_{cr} such that for $0<\delta t<\delta t^2_{\mathrm{cr}}$ the sequence of errors $E=U_{ex}-U$ satisfies the following error estimate:*

$$\begin{aligned}|||E|||^2 \leq C\frac{\delta t^4}{T}\sum_{n=0}^{N-1}\delta t\Bigg(&\left\|\left.\frac{\partial^3 u}{\partial t^3}\right|_{\boldsymbol{x}_0}\right\|^2_{L^2(\Omega_{t^{n+1}})}\\ &+\sup_{s\in(t^n,t^{n+1})}\left\|\left|\frac{\partial^3\mathcal{A}_s}{\partial t^3}\right|^2\right\|_{L^\infty(\Omega_0)}\left\|u^{n+1}\right\|^2_{H^1(\Omega_{t^{n+1}})}\Bigg).\end{aligned}$$

Proof. We start taking the exact solution sequence U_{ex} in the bilinear form. We get:

$$\begin{aligned}B(U_{ex},V) =& L(V)+\sum_{n=0}^{N-1}\frac{1}{2T}\left(3u(t^{n+1})-4u(t^n)+u(t^{n-1})-\delta t\left.\frac{\partial u}{\partial t}\right|_{t^{n+1}},v^{n+1}\right)_{\Omega_{t^{n+1}}}\\ &-\sum_{n=0}^{N-1}\delta t\left((\boldsymbol{w}^{n+1}-\boldsymbol{w}(t^{n+1}))\cdot\nabla u(t^{n+1}),v^{n+1}\right)_{\Omega_{t^{n+1}}}.\end{aligned}$$

Now we subtract the equation for the semi-discrete sequence of solutions to the previous equations and arrive to

$$\begin{aligned}B(U-U_{ex},V) =& -\sum_{n=0}^{N-1}\frac{1}{T}\left(3u(t^{n+1})-4u(t^n)+u(t^{n-1})-\delta t\left.\frac{\partial u}{\partial t}\right|_{t^{n+1}},v^{n+1}\right)_{\Omega_{t^{n+1}}}\\ &+\sum_{n=0}^{N-1}\delta t\left((\boldsymbol{w}^{n+1}-\boldsymbol{w}(t^{n+1}))\cdot\nabla u(t^{n+1}),v^{n+1}\right)_{\Omega_{t^{n+1}}}.\end{aligned}$$

We test the previous equation with $V = U - U_{ex} = E$, obtaining

$$B(E,E) = -\sum_{n=0}^{N-1} \frac{1}{T}\left(3u(t^{n+1}) - 4u(t^n) + u(t^{n-1}) - \delta t \left.\frac{\partial u}{\partial t}\right|_{t^{n+1}}, e^{n+1}\right)_{\Omega_{t^{n+1}}} + \sum_{n=0}^{N-1} \delta t \left((\boldsymbol{w}^{n+1} - \boldsymbol{w}(t^{n+1})) \cdot \nabla u(t^{n+1}), e^{n+1}\right)_{\Omega_{t^{n+1}}}.$$

The truncation error introduced by the time integration scheme BDF2 is evaluated using the following Taylor formula:

$$\frac{3u(\boldsymbol{x}_0,t^{n+1}) - 4u(\boldsymbol{x}_0,t^n) + u(\boldsymbol{x}_0,t^{n-1})}{T\delta t} - \frac{1}{T}\left.\frac{\partial u}{\partial t}\right|_{\boldsymbol{x}_0}(t^{n+1}) = -\frac{1}{T\delta t}\int_{t^{n-1}}^{t^{n+1}} (s-t^n)^2 \left.\frac{\partial^3 u}{\partial t^3}\right|_{\boldsymbol{x}_0}(s)\,\mathrm{d}s - \frac{1}{T\delta t}\int_{t^n}^{t^{n+1}} (s-t^n)^2 \left.\frac{\partial^3 u}{\partial t^3}\right|_{\boldsymbol{x}_0}(s)\,\mathrm{d}s. \quad (6.33)$$

The evaluation of the mesh velocity (6.6) requires a time derivative. Its numerical approximation using the second order BDF2 scheme can be written again as a truncation error:

$$\boldsymbol{w}^{n+1} - \boldsymbol{w}(t^{n+1}) = -\frac{1}{T\delta t}\left(\int_{t^n-1}^{t^{n+1}} (s-t^n)^2 \frac{\partial^3 \mathcal{A}_s}{\partial t^3}\,\mathrm{d}s + \int_{t^n}^{t^{n+1}} (s-t^n)^2 \frac{\partial^3 \mathcal{A}_s}{\partial t^3}\,\mathrm{d}s\right) \circ \mathcal{A}_{t^{n+1}}^{-1}, \quad (6.34)$$

which holds for all $\boldsymbol{x} \in \Omega_t$. Recall that $\boldsymbol{w}^{n+1}$ stands for the mesh velocity evalutated at t^{n+1}.

The error related to the time derivative of u can be bounded using the following inequality:

$$\begin{aligned}\int_{\Omega_{t^{n+1}}} e^{n+1} \cdot \Bigg(&\int_{t^{n-1}}^{t^{n+1}} (s-t^n)^2 \left.\frac{\partial^3 u}{\partial t^3}\right|_{\boldsymbol{x}_0}(s)\,\mathrm{d}s \\ &-\frac{1}{T\delta t}\int_{t^n}^{t^{n+1}} (s-t^n)^2 \left.\frac{\partial^3 u}{\partial t^3}\right|_{\boldsymbol{x}_0}(s)\,\mathrm{d}s\Bigg) \circ \mathcal{A}_{t^{n+1}}^{-1}\,\mathrm{d}\Omega \\ &\le \frac{\beta_2 \delta t}{2}\left\|e^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})} + C\delta t^5 \left\|\left.\frac{\partial^2 u}{\partial t^2}\right|_{\boldsymbol{x}_0}\right\|^2_{L^2(\Omega_{t^{n+1}})}, \end{aligned} \quad (6.35)$$

where β_2 is the coercivity constant introduced in Theorem 6.3.

We obtain the following inequality in order to bound the error introduced by the evaluation of the mesh velocity,

$$\begin{aligned}-\int_{\Omega_{t^{n+1}}} e^{n+1} \Bigg(&\int_{t^n-1}^{t^{n+1}} (s-t^n)^2 \frac{\partial^3 \mathcal{A}_s}{\partial t^3}\,\mathrm{d}s \\ &+\int_{t^n}^{t^{n+1}} (s-t^n)^2 \frac{\partial^3 \mathcal{A}_s}{\partial t^3}\,\mathrm{d}s\Bigg) \circ \mathcal{A}_{t^{n+1}}^{-1} \cdot \nabla u^{n+1}\,\mathrm{d}\Omega \\ &\le \frac{\beta_2 \delta t}{2}\left\|e^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})} + \delta t^5 \sup_{s\in(t^{n-1},t^{n+1})} \left\|\left|\frac{\partial^3 \mathcal{A}_s}{\partial t^3}\right|^2\right\|_{L^\infty(\Omega_0)} \left\|u^{n+1}\right\|^2_{H^1(\Omega_{t^{n+1}})}. \end{aligned} \quad (6.36)$$

Using the error expressions (6.33) and (6.34) and bounds (6.35) and (6.36) we get

$$B(E,E) \leq \frac{1}{T}\sum_{n=0}^{N-1}\delta t\beta_2 \left\|e^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})} + C\frac{\delta t^4}{T}\sum_{n=0}^{N-1}\delta t \left\| \left.\frac{\partial^3 u}{\partial t^3}\right|_{\boldsymbol{x}_0}\right\|^2_{L^2(\Omega_{t^{n+1}})}$$
$$+ C\frac{\delta t^4}{T}\sum_{n=0}^{N-1}\delta t \sup_{s\in(t^n,t^{n+1})}\left\|\left|\frac{\partial^3 \mathcal{A}_s}{\partial t^3}\right|^2\right\|_{L^\infty(\Omega_0)} \left\|u^{n+1}\right\|^2_{H^1(\Omega_{t^{n+1}})}.$$

Again, we can apply the Gronwall lemma without any extra condition over the time step size. □

6.4 The fully discrete problem

In this section we analyze the fully discrete problems BDF1-BDF1-OSS$_{\delta t,h}$ and BDF2-BDF2-OSS$_{\delta t,h}$. In both cases, stability and error estimates are obtained.

Observe from (6.20) that τ^n has been taken constant in space. Further, we assume Θ_h^t quasi-uniform. In this case, the following inverse estimate holds (see [23]):

$$\|\nabla v_h\|_{L^2(\Omega_t)} \leq \frac{C_{\text{inv}}}{h}\|v_h\|_{L^2(\Omega_t)}. \tag{6.37}$$

In order to obtain optimal convergence results, we assume that $u^{n+1} \in H^{p+1}(\Omega_t)$ for $n = 0, ..., N-1$, where p is the degree of the polynomial defining the finite element space $\mathcal{V}_h$. As stated in Section 1.3, we know from the finite element approximation theory that for any function $v \in H^{p+1}(\Omega_t)$ there exists a finite element interpolation $\pi_h(v)$ such that,

$$\|v - \pi_h(v)\|_{H^m(\Omega_t)} \leq C_h h^{p+1-m}\|v\|_{H^{p+1}(\Omega_t)}.$$

We need to prove that the L^2-projection onto the finite element space is an optimal interpolation in the $L^2(\Omega_t)$ norm and the seminorm $\|\nabla(\cdot)\|_{L^2(\Omega_t)}$. We show this in the following lemma.

Lemma 6.2. *Given a function $v \in H^{p+1}(\Omega_t)$ with $p \geq 1$, its L^2-projection onto the finite element space $\Pi_h(v)$ satisfies*

$$\|v - \Pi_h(v)\|_{L^2(\Omega_t)} \leq C_h h^{p+1}\|v\|_{H^{p+1}(\Omega_t)}, \tag{6.38}$$

and also

$$h^2\|\Delta v - \Pi_h(\Delta v)\|_{L^2(\Omega_t)} \leq C h^{p+1}\|v\|_{H^{p+1}(\Omega_t)}. \tag{6.39}$$

If the inverse estimate (6.37) hold true

$$\|\nabla(v - \Pi_h(v))\|_{L^2(\Omega_t)} \leq C h^{p}\|v\|_{H^{p+1}(\Omega_t)}, \tag{6.40}$$

are satisfied.

Proof. The first inequality is obvious:

$$\|v - \Pi_h(v)\|_{L^2(\Omega_t)} \leq \|v - \pi_h(v)\|_{L^2(\Omega_t)} \leq C_h h^{p+1}\|v\|_{H^{p+1}(\Omega_t)}.$$

Using the previous inequality for Δv we obtain the second result.

Making use of the inverse estimate (6.37) we prove the last inequality:

$$\begin{aligned}\|\nabla\left(v-\Pi_h\left(v\right)\right)\|_{L^2(\Omega_t)} &\leq \|\nabla\left(v-\pi_h(v)\right)\|_{L^2(\Omega_t)} + \|\nabla\left(\pi_h(v)-\Pi_h\left(v\right)\right)\|_{L^2(\Omega_t)} \\ &= \|\nabla\left(v-\pi_h(v)\right)\|_{L^2(\Omega_t)} + \|\nabla\left(\Pi_h\left(v-\pi_h(v)\right)\right)\|_{L^2(\Omega_t)} \\ &\leq \|\nabla\left(v-\pi_h(v)\right)\|_{L^2(\Omega_t)} + \frac{C_{\text{inv}}}{h}\|\Pi_h\left(v-\pi_h(v)\right)\|_{L^2(\Omega_t)} \\ &\leq (1+C_{\text{inv}})C_h h^p \|v\|_{H^{p+1}(\Omega_t)}.\end{aligned}$$

□

As in the previous section, C is a positive constant, possibly with different values at different appearances.

6.4.1 Analysis of BDF1-BDF1-OSS$_{\delta t,h}$

In this subsection we analyze the fully discrete problem (6.17) with the bilinear form $B_h(\cdot,\cdot)$ defined in (6.18) and right-hand side (6.14). We denote by $U=\{u^0,u^1,u^2,...,u^N\}$ the sequence of solutions of the semi-discrete problem (in time) (6.9)-(6.11) and $U_h=\{u_h^0,u_h^1,u_h^2,...,u_h^N\}$ its fully discrete counterpart, solution of (6.17)-(6.18)-(6.14).

We start proving the coercivity of the bilinear form for the *weak* norm $|||\cdot|||_w$. This result will be used in the convergence analysis.

Theorem 6.5 (Coercivity)**.** *There exists δt_{cr}^1 such that for $0<\delta t<\delta t_{\text{cr}}^1$ the bilinear form $B_h(\cdot,\cdot)$ defined in (6.18) is coercive. That is, for every sequence $V=\{v^n\}_{n=0}^N$, with $v^n\in\mathcal{V}(\Omega_{t^n})$,*

$$B_h\left(V,V\right)\geq\beta_1|||V|||_w^2$$

for a certain constant $\beta_1>0$.

Proof. The bilinear form analyzed in this theorem is equal to the one for which coercivity is proved in Theorem 6.1 plus the stabilization term. We can easily get

$$\begin{aligned}B_h(V,V) &= \frac{1}{2T}\left\|v^N\right\|_{L^2(\Omega^N)}^2 + \frac{1}{2T}\sum_{n=0}^{N-1}\left\|\delta v^{n+1}\right\|_{L^2(\Omega_{t^{n+1}})}^2 \\ &+ \sum_{n=0}^{N-1}\left[\delta t\nu\left\|\nabla v^{n+1}\right\|_{L^2(\Omega_{t^{n+1}})}^2 + \delta t\tau^{n+1}\left\|\Pi_h^{\perp}\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v^{n+1}\right)\right\|_{L^2(\Omega_{t^{n+1}})}^2\right] \\ &+ \frac{1}{2}\sum_{n=0}^{N-1}\delta t\left(\nabla\cdot\boldsymbol{w}^{n+1},(v^{n+1})^2\right)_{\Omega_{t^{n+1}}} + \frac{1}{2}\sum_{n=0}^{N-1}\int_{t^n}^{t^{n+1}}\int_{\Omega_t}(\nabla\cdot\boldsymbol{w}^{n+1})(v^{n+1})^2\,\mathrm{d}\Omega. \quad (6.41)\end{aligned}$$

Due to the fact that the stabilization term does not affect the treatment of the mesh velocity terms in Theorem 6.1, we refer to this one for the remaining of the proof.□

Let us define the Λ-coercivity property associated to a bilinear form that will be used in the following analysis.

Definition 6.1 (Λ-coercivity)**.** Let $\mathcal{V}$ be a functional space and $\zeta:\mathcal{V}\times\mathcal{V}\longrightarrow\mathbb{R}$ a bilinear form. We say that ζ is Λ-coercive with respect to the norm $|||\cdot|||$ and the linear operator $\Lambda:\mathcal{V}\longrightarrow\mathcal{V}$ if there exists and a constant $\beta>0$ such that,

$$\zeta(v,\Lambda(v))\geq\beta|||v|||^2,\quad\forall v\in\mathcal{V}. \quad (6.42)$$

The bilinear form $\zeta(\cdot,\cdot)$ also satisfies an *inf-sup* condition under the conditions of the following lemma.

Lemma 6.3. *If Λ is continuous with respect to the norm $|||\cdot|||$ and $\zeta(\cdot,\cdot)$ is Λ-coercive, then there exists $\gamma > 0$ such that*

$$\inf_{u\in\mathcal{V}} \sup_{v\in\mathcal{V}} \frac{\zeta(u,v)}{|||u||| \, |||v|||} \geq \gamma.$$

The proof of the previous lemma is straightforward from Definition 6.1 and the continuity of the operator $\Lambda(\cdot)$.

We now show that the bilinear form $B_h(\cdot,\cdot)$ of our problem is Λ-coercive for the *strong* norm $|||\cdot|||_s$.

Theorem 6.6 (Λ-coercivity)**.** *Let $V = \{v^n\}_{n=0}^N$ be a sequence of functions such that $v^n \in \mathcal{V}(\Omega_{t^n})$ and consider the operator*

$$\Lambda(V) = V + \{0, \frac{1}{2}\{\tau^{n+1}\Pi_h\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v^{n+1}\right)\}_0^{N-1}\}.$$

Then, there exists $\delta t^1_{\rm cr}$ such that, for $0 < \delta t < \delta t^1_{\rm cr}$, the bilinear form $B_h(\cdot,\cdot)$ is Λ-coercive:

$$B_h\left(V, \Lambda(V)\right) \geq \beta_1 |||V|||_s^2$$

for a certain constant $\beta_1 > 0$.

Proof. Testing (6.18) with the sequence of functions that belong to the finite element space

$$\Pi_0(\tau, V) := \{0, \{\tau^{n+1}\Pi_0(v^{n+1})\}\}_{n=0}^{N-1} := \{0, \{\tau^{n+1}\Pi_h\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v^{n+1}\right)\}_{n=0}^{N-1}\}$$

we have

$$\begin{aligned} B_h(V, \Pi_0(\tau, V)) \geq & \sum_{n=0}^{N-1} \phi^{n+1}\delta t\tau^{n+1} \left\|\Pi_h\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v^{n+1}\right)\right\|^2_{L^2(\Omega_{t^{n+1}})} \\ & - \sum_{n=0}^{N-1}\left[\frac{1}{T}\left\|\delta v^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})} + \delta t\nu \left\|\nabla v^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})}\right. \\ & \left. + \delta t\tau^{n+1}\left\|\Pi_h^{\perp}\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v^{n+1}\right)\right\|^2_{L^2(\Omega_{t^{n+1}})}\right], \end{aligned} \tag{6.43}$$

where

$$\phi^{n+1} := 1 - \frac{1}{4}\frac{\tau^{n+1}}{T\delta t} - \frac{1}{4}\tau^{n+1}\frac{\nu C_{\rm inv}^2}{h^2} - \frac{1}{4}(\tau^{n+1})^2\frac{\left\|\boldsymbol{a}-\boldsymbol{w}^{n+1}\right\|^2_{L^\infty(\Omega_{t^{n+1}})} C_{\rm inv}^2}{h^2}. \tag{6.44}$$

To obtain (6.43) we have made use of Young's inequality and the inverse estimate (6.37). Assuming now that the constants c_1 and c_2 in (6.20) are such that $c_1 \leq C_{\rm inv}^2$, $c_2 \leq C_{\rm inv}$ and the constant C in (6.21) is $C \leq 1$ it follows that $\phi^{n+1} \geq 1/4$.

The combination of (6.43) and (6.41) leads to

$$
\begin{aligned}
B_h(V,2V+\Pi_0(\tau,V)) \geq & \frac{1}{T}\left\|v^N\right\|^2_{L^2(\Omega^N)} + \sum_{n=0}^{N-1}\delta t\nu\left\|\nabla v^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})} \\
& + C\sum_{n=0}^{N-1}\delta t\tau^{n+1}\left\|(\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})} \\
& + \sum_{n=0}^{N-1}\delta t\left(\nabla\cdot\boldsymbol{w}^{n+1},(v^{n+1})^2\right)_{\Omega_{t^{n+1}}} + \sum_{n=0}^{N-1}\int_{t^n}^{t^{n+1}}\int_{\Omega_s}(\nabla\cdot\boldsymbol{w}^{n+1})(v^{n+1})^2\,\mathrm{d}\Omega\,\mathrm{d}s \\
\geq & \frac{1}{T}\left\|v^N\right\|^2_{L^2(\Omega^N)} + \sum_{n=0}^{N-1}\delta t\nu\left\|\nabla v^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})} \\
& + C\sum_{n=0}^{N-1}\delta t\tau^{n+1}\left\|\Pi_h\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v^{n+1}\right)\right\|^2_{L^2(\Omega_{t^{n+1}})} \\
& - \sum_{n=0}^{N-1}\delta t\gamma_{n+1}\left\|v^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})},
\end{aligned}
$$

with $\gamma_{n+1} := \gamma_1^{n+1}+\gamma_2^{n+1}$, and γ_1^{n+1}, γ_2^{n+1} defined in (8.3) and (6.24). Using the Gronwall lemma, we finally get the coercivity stated in the theorem. We point out that the critical time step δt^1_{cr}in this case is identical to the one obtained for the semi-discrete problem. □

In order to satisfy the continuity of $\Lambda(\cdot)$ needed for to obtain the *inf-sup* condition in Lemma 6.3 we have to restrict the situation to the discrete finite element space $\mathcal{V}_h$.

Lemma 6.4 (Continuity)**.** *Let $V_h = \{v_h^n\}_{n=0}^N$ be a finite element sequence such that $v_h^n \in \mathcal{V}_h(\Omega_{t^n})$ and consider the operator Λ introduced in Theorem 6.6. Then, $\Lambda(\cdot)$ is continuous with respect to the norm $|||\cdot|||_s$ for every finite element sequence V_h:*

$$
|||\Lambda(V_h)|||_s \leq \rho|||V_h|||_s \tag{6.45}
$$

for a certain constant $\rho > 0$.

Proof. Defining $\Pi_0(\tau,V_h)$ as in the proof of the previous theorem, we have from the definition of the norm that

$$
\begin{aligned}
|||\Pi_0(\tau,V_h)|||_s^2 = & \frac{1}{T}\sup_{n\in[0,N-1]}\left\|\tau^{n+1}\Pi_0(v_h^{n+1})\right\|^2_{L^2(\Omega_{t^{n+1}})} \\
& + \sum_{n=0}^{N-1}\delta t\nu\left\|\tau^{n+1}\nabla\Pi_0(v_h^{n+1})\right\|^2_{L^2(\Omega_{t^{n+1}})} \\
& + \sum_{n=0}^{N-1}\delta t\tau^{n+1}\left\|\tau^{n+1}(\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla\Pi_0(v_h^{n+1})\right\|^2_{L^2(\Omega_{t^{n+1}})}.
\end{aligned} \tag{6.46}
$$

Invoking the expression for τ^{n+1} and the inverse estimate (6.37) we can easily bound every term by $|||V|||_s^2$.□

Remark 6.2. The fact that we need to use the inverse estimate (6.37) in order to bound *the first term in* (6.46) restricts the continuity of $\Lambda(\cdot)$ to finite element sequences (for the rest of the terms the inverse estimate is applied to derivatives of $\Pi_h\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v^{n+1}\right)$, a finite element function even if v^{n+1} is not in the finite element space). However, this restriction does not complicate the convergence analysis, where only the Λ-coercivity is invoked.

From Lemmas 6.3 and 6.4 we obtain the *discrete inf-sup* condition.

Corollary 6.4 (Discrete inf-sup condition)**.** *Let $U_h = \{u_h^n\}_{n=0}^N$ and $V_h = \{v_h^n\}_{n=0}^N$ be sequences of finite element functions such that $u^n, v^n \in \mathcal{V}(\Omega_{t^n})$. There exists δt_{cr}^1 such that, for $0 < \delta t < \delta t_{\text{cr}}^1$, the bilinear form $B_h(\cdot,\cdot)$ satisfies the following condition:*

$$\inf_{U_h \in \mathcal{V}_h} \sup_{V_h \in \mathcal{V}_h} \frac{B_h\left(U_h, V_h\right)}{|||U_h|||_s \, |||V_h|||_s} \geq \widetilde{\beta}_1$$

for a certain constant $\widetilde{\beta}_1 > 0$.

At this point, the only ingredient that remains in order to have a stability result is the continuity of the *force term*, provided by Lemma 6.1. The stability result is stated in the next corollary.

Corollary 6.5 (Stability)**.** *There exists δt_{cr}^1 such that, for $0 < \delta t < \delta t_{\text{cr}}^1$, the sequence U_h, solution of problem (6.17)-(6.18)-(6.11), is bounded as follows:*

$$|||U_h|||_s^2 \leq C \sum_{n=0}^{N-1} \frac{\delta t}{\nu} \left\| f^{n+1} \right\|_{H^{-1}(\Omega_{t^{n+1}})}^2 .$$

For the convergence analysis, let us define the difference between the solution of (6.8) and (6.16) as $e_d^{n+1} := u_h^{n+1} - u^{n+1}$, and the sequence of these errors by E_d. From Theorem 6.6, that proves the Λ-coercivity of the bilinear form B_h for Λ defined in this theorem, we know that

$$B_h\left(E_d, \Lambda(E_d)\right) \geq \beta_1 |||E_d|||_s^2. \tag{6.47}$$

We subtract the discrete bilinear form (6.18) from its semi-discrete counterpart (6.10) tested with finite element sequences in order to get

$$\begin{aligned} B_h\left(E_d, V_h\right) &= \epsilon_c(V_h) \\ &:= -\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left(\Pi_h^{\perp} \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla u^{n+1} \right), (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla v_h^{n+1} \right)_{\Omega_{t^{n+1}}}, \end{aligned}$$

where $\epsilon_c(V_h)$ accounts for the consistency error. After some manipulations, we can write

$$\begin{aligned} B_h\left(E_d, \Lambda(E_d)\right) &= B_h\left(E_d, E_d\right) + \frac{1}{2} B_h\left(E_d, \Pi_0(\tau, E_d)\right) \\ &= B_h\left(E_d, \Pi_h(U) - U\right) + \epsilon_c(U_h - \Pi_h(U)) + \frac{1}{2}\epsilon_c(\Pi_0(\tau, E_d)), \end{aligned}$$

where $\Pi_h(U) := \{\Pi_h(u^n)\}_{n=0}^N$.

We distinguish between *interpolation* error, the first term of the right hand side, and the *consistency* error associated to the second and third terms. In the following two lemmas we bound these error terms. We start with the interpolation error, obtaining the result stated in the following lemma:

Lemma 6.5 (Interpolation error)**.** *The error sequence $E_d = U_h - U$ satisfies the following inequality:*

$$B_h\left(E_d, \Pi_h(U) - U\right) \leq C |||E_d|||_w \left(h^{2(p+1)} \sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left\| u^{n+1} \right\|_{H^{p+1}(\Omega_{t^{n+1}})}^2 \right)^{\frac{1}{2}}. \tag{6.48}$$

Proof. Let us expand the expression of the interpolation error making use of the definition of the bilinear form associated to the problem we are analyzing:

$$
\begin{aligned}
&B_h\left(E_d, \Pi_h(U) - U\right) \\
&\quad = \sum_{n=0}^{N-1} \Bigg[\frac{1}{T} \left(e_d^{n+1} - e_d^n, \Pi_h\left(u^{n+1}\right) - u^{n+1}\right)_{\Omega_{t^{n+1}}} \\
&\quad + \delta t \nu \left(\nabla e_d^{n+1}, \nabla(\Pi_h\left(u^{n+1}\right) - u^{n+1})\right)_{\Omega_{t^{n+1}}} \\
&\quad + \delta t \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla e_d^{n+1}, \Pi_h\left(u^{n+1}\right) - u^{n+1}\right)_{\Omega_{t^{n+1}}} \\
&\quad + \delta t \tau^{n+1} \left(\Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla e_d^{n+1}\right), (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla \left(\Pi_h\left(u^{n+1}\right) - u^{n+1}\right)\right)_{\Omega_{t^{n+1}}} \Bigg].
\end{aligned}
$$

We must control each term separately. Let us start with the discrete time derivative term. Using assumption (6.21) we have that

$$
\begin{aligned}
&\sum_{n=0}^{N-1} \frac{1}{T} \left(e_d^{n+1} - e_d^n, \Pi_h\left(u^{n+1}\right) - u^{n+1}\right)_{\Omega_{t^{n+1}}} \\
&\leq C \left(\sum_{n=0}^{N-1} \frac{1}{T} \left\| e_d^{n+1} - e_d^n \right\|^2_{L^2(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}} \times \left(h^{2(p+1)} \sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left\| u^{n+1} \right\|^2_{H^{p+1}(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}}.
\end{aligned}
$$

For the viscosity term, using the definition of τ^{n+1} and the inverse estimate (6.41), we have that

$$
\begin{aligned}
&\sum_{n=0}^{N-1} \delta t \nu \left(\nabla e_d^{n+1}, \nabla(\Pi_h\left(u^{n+1}\right) - u^{n+1})\right)_{\Omega_{t^{n+1}}} \\
&\leq C \left(\sum_{n=0}^{N-1} \delta t \nu \left\| \nabla e_d^{n+1} \right\|^2_{L^2(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}} \times \left(h^{2(p+1)} \sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left\| u^{n+1} \right\|^2_{H^{p+1}(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}}.
\end{aligned}
$$

Similar arguments allow us to obtain a bound for the convective term:

$$
\begin{aligned}
&\sum_{n=0}^{N-1} \delta t \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla e_d^{n+1}, \Pi_h\left(u^{n+1}\right) - u^{n+1}\right)_{\Omega_{t^{n+1}}} \\
&\qquad \leq C \left(\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left\| \Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla e_d^{n+1}\right) \right\|^2_{L^2(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}} \\
&\qquad\quad \times \left(h^{2(p+1)} \sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left\| u^{n+1} \right\|^2_{H^{p+1}(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}},
\end{aligned} \tag{6.49}
$$

and for the stabilization term we obtain

$$
\begin{aligned}
&\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left(\Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla e_d^{n+1}\right), (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla \left(\Pi_h\left(u^{n+1}\right) - u^{n+1}\right)\right)_{\Omega_{t^{n+1}}} \\
&\qquad \leq C \left(\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left\| \Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla e_d^{n+1}\right) \right\|^2_{L^2(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}} \\
&\qquad\quad \times \left(h^{2(p+1)} \sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left\| u^{n+1} \right\|^2_{H^{p+1}(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}}.
\end{aligned}
$$

All the terms have been bounded by the right-hand-side of (6.48), and therefore the proof is finished.□

Remark 6.3. Invoking the interpolation error (6.38) in (6.49) has allowed us to obtain an optimal bound for the interpolation error without the control of the full convective term in the norm $|||\cdot|||_w$. This fact will be used for the analysis of the second order method.

The following lemma is devoted to the control of the consistency error. Since we are interested in smooth solutions, say $u \in L^2(0,T;H^{p+1}(\Omega_t))$ (with the obvious modifications for u less regular), we assume that f is also smooth, in particular $f \in L^2(0,T;H^{p-1}(\Omega_t))$. Thus, for $p \geq 1$, $\langle f, v_h\rangle_{\Omega_t} = (\Pi_h(f), v_h)_{\Omega_t}$. Therefore, the finite element solution is not altered if we assume $\Pi_h^{\perp}(f) = 0$.

Lemma 6.6 (Consistency error)**.** *The following inequality holds:*

$$\epsilon_c(U_h - \Pi_h(U) + \frac{1}{2}\Pi_0(\tau, E_d)) \leq C \left(h^{2(p+1)} \sum_{n=0}^{N-1} \delta t(\tau^{n+1})^{-1} \left\| u^{n+1} \right\|^2_{H^{p+1}(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}} \times \left(|||E_d|||_s^2 + h^{2(p+1)} \sum_{n=0}^{N-1} \delta t(\tau^{n+1})^{-1} \left\| u^{n+1} \right\|^2_{H^{p+1}(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}}. \tag{6.50}$$

Proof. From the expression of the consistency error we arrive to:

$$\begin{aligned}
&- \epsilon_c(U_h - \Pi_h(U)) \\
&= \sum_{n=0}^{N-1} \delta t \tau^{n+1} \left(\Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla u^{n+1} \right), (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla (u_h^{n+1} - \Pi_h\left(u^{n+1}\right)) \right)_{\Omega_{t^{n+1}}} \\
&= \sum_{n=0}^{N-1} \delta t \tau^{n+1} \left(\Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla u^{n+1} \right), (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla e_d^{n+1} \right)_{\Omega_{t^{n+1}}} \qquad (6.51) \\
&+ \sum_{n=0}^{N-1} \delta t \tau^{n+1} \left(\Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla u^{n+1} \right), (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla (u^{n+1} - \Pi_h\left(u^{n+1}\right)) \right)_{\Omega_{t^{n+1}}}.
\end{aligned}$$

On the other hand, from the equation for the semi-discrete unknown (6.8), we can easily check that

$$\begin{aligned}
&\left(\Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla u^{n+1} \right), v^{n+1} \right)_{\Omega_{t^{n+1}}} \\
&\quad = \left(\Pi_h^{\perp}\left(\nu \Delta u^{n+1} - \frac{1}{T\delta t}(u^{n+1} - u^n) \right), v^{n+1} \right)_{\Omega_{t^{n+1}}} \\
&\quad =: \left(\Pi_h^{\perp}\left(\lambda(u^{n+1}) \right), v^{n+1} \right)_{\Omega_{t^{n+1}}}.
\end{aligned} \tag{6.52}$$

where $\lambda(\cdot) := \nu\Delta(\cdot) - \frac{\delta(\cdot)}{T\delta t}$. Note that we have not included $\Pi_h^{\perp}(f)$ in the previous equation.

Now, using (6.52) in (6.51) we can split the error in two different terms bounded as

follows:

$$\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left(\Pi_h^{\perp} \left(\lambda(u^{n+1}) \right), (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla e_d^{n+1} \right)_{\Omega_{t^{n+1}}}$$
$$\leq C \left(\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left\| \Pi_h^{\perp} \left(\lambda(u^{n+1}) \right) \right\|^2_{L^2(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}}$$
$$\times \left(\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left\| \Pi_h^{\perp} \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla e_d^{n+1} \right) \right\|^2_{L^2(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}},$$

$$\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left(\Pi_h^{\perp} \left(\lambda(u^{n+1}) \right), (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla \left(u^{n+1} - \Pi_h \left(u^{n+1} \right) \right) \right)_{\Omega_{t^{n+1}}}$$
$$\leq C \left(\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left\| \Pi_h^{\perp} \left(\lambda(u^{n+1}) \right) \right\|^2_{L^2(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}}$$
$$\times \left(h^{2(p+1)} \sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left\| u^{n+1} \right\|^2_{H^{p+1}(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}}.$$

On the other hand, the term related to the perturbation of the test function $\Pi_0(\tau, E_d)$ appearing in (6.50) can be bounded using similar arguments, leading to

$$\begin{aligned}
&\epsilon_c(\Pi_0(\tau, E_d)) \\
&= \sum_{n=0}^{N-1} \delta t \tau^{n+1} \left(\Pi_h^{\perp} \left(\lambda(u^{n+1}) \right), (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla \left(\tau^{n+1} \Pi_h \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla e_d^{n+1} \right) \right) \right)_{\Omega_{t^{n+1}}} \\
&\leq C \left(\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left\| \Pi_h^{\perp} \left(\lambda(u^{n+1}) \right) \right\|^2_{L^2(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}} \\
&\quad \times \left(\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left\| \Pi_h \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla e_d^{n+1} \right) \right\|^2_{L^2(\Omega_{t^{n+1}})} \right)^{\frac{1}{2}}.
\end{aligned}$$

It only remains to prove that

$$\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left\| \Pi_h^{\perp} \left(\lambda(u^{n+1}) \right) \right\|^2_{L^2(\Omega_{t^{n+1}})} \leq C h^{2(p+1)} \sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left\| u^{n+1} \right\|^2_{H^{p+1}(\Omega_{t^{n+1}})}.$$

This inequality can be easily obtained from the expression of τ^{n+1}, assumption (6.19) and the interpolation error estimate (6.39).□

We end this section with the main convergence result, which is a direct consequence of inequality (6.47), Lemma 6.5 and Lemma 6.6:

Theorem 6.7 (Convergence)**.** *There exist δt_{cr}^1 such that, for $0 < \delta t < \delta t_{\text{cr}}^1$, the sequence of errors $E_d = U_h - U$ satisfies the following error estimate:*

$$|||E_d|||_s^2 \leq C h^{2(p+1)} \sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left\| u^{n+1} \right\|^2_{H^{p+1}(\Omega_{t^{n+1}})}.$$

6.4.2 Analysis of BDF2-BDF2-OSS$_{\delta t,h}$

In this subsection we analyze the fully discrete problem (6.17) with the bilinear form $B_h(\cdot,\cdot)$ defined in (6.22) and right-hand side (6.14). We denote by $U = \{u^0, u^1, u^2, ..., u^N\}$ the sequence of solutions of the second order semi-discrete problem (in time) (6.9)-(6.13)-(6.14) and $U_h = \{u_h^0, u_h^1, u_h^2, ..., u_h^N\}$ its fully discrete counterpart, solution of (6.17)-(6.22)-(6.14).

We have obtained the results of this section using the *weak* norm $|||\cdot|||_w$. Let us start with a theorem proving coercivity under the *weaker* norm.

Theorem 6.8 (Coercivity)**.** *There exists δt_{cr}^2 such that, for $0 < \delta t < \delta t_{\text{cr}}^2$, the bilinear form $B_h(\cdot,\cdot)$ defined in (6.22) is coercive. That is, for every sequence $V = \{v^n\}_{n=0}^N$ with $v^n \in V(\Omega_{t^n})$*

$$B_h\left(V,V\right) \geq \beta_2 |||V|||_w^2$$

for a certain constant $\beta_2 > 0$.

Proof. It can be easily shown that

$$\begin{aligned}
B_h\left(V,4V\right) \geq & \frac{1}{T}\left(\left\|v^N\right\|_{L^2(\Omega^N)}^2 + \sum_{n=0}^{N-1}\left\|\delta^2 v^{n+1}\right\|_{L^2(\Omega_{t^{n+1}})}^2\right) \\
& + \sum_{n=0}^{N-1} 4\delta t\nu \left\|\nabla v^{n+1}\right\|_{L^2(\Omega_{t^{n+1}})}^2 + \sum_{n=0}^{N-1} 4\delta t\tau^{n+1}\left\|\Pi_h^{\perp}\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v^{n+1}\right)\right\|_{L^2(\Omega_{t^{n+1}})}^2 \\
& + \int_{t^n}^{t^{n+1}}\int_{\Omega_s}(\nabla\cdot\boldsymbol{w}(s))(v^n)^2\,\mathrm{d}\Omega\,\mathrm{d}s + \int_{t^n}^{t^{n+1}}\int_{\Omega_s}(\nabla\cdot\boldsymbol{w}(s))(2v^n - v^{n-1})^2\,\mathrm{d}\Omega\,\mathrm{d}s.
\end{aligned}$$

Manipulating the mesh velocity as for the BDF2-BDF2$_{\delta t}$ formulation (see Theorem 6.3) and applying the Gronwall Lemma we obtain the desired result.□

Stability is now straightforward from Theorem 6.8 and Lemma 6.1:

Corollary 6.6 (Stability)**.** *There exists δt_{cr}^2 such that, for $0 < \delta t < \delta t_{\text{cr}}^2$, the sequence U_h, solution of problem (6.17)-(6.22)-(6.14), is bounded as follows:*

$$|||U_h|||_w^2 \leq C\sum_{n=0}^{N-1}\frac{\delta t}{\nu}\left\|f^{n+1}\right\|_{H^{-1}(\Omega_{t^{n+1}})}^2.$$

This stability result can be considered *weak*. However, we will see that this result is enough in order to obtain error estimates without constants depending on the Péclet number (except, of course, in the dependence of the continuous solution with this number), the original motivation of stabilization methods for convection-diffusion problems.

Let us obtain now error estimates for the BDF2-BDF2-OSS$_{\delta t,h}$ formulation. We start with an auxiliary lemma that will be useful in the following:

Lemma 6.7. *Let $X = \{x^n\}_{n=0}^N$ and $V = \{v^n\}_{n=0}^N$ be two sequences of functions such that $x^n, v^n \in H^{p+1}(\Omega_{t^n})$. Then, the bilinear form (6.22) satisfies the following bound:*

$$\begin{aligned}
B_h\left(X,\Pi_h^{\perp}\left(V\right)\right) \leq & C\left(|||X|||_w^2 + \sum_{n=-1}^{N-1}\delta t(\tau^{n+1})^{-1}\|\Pi_h^{\perp}(x^{n+1})\|_{L^2(\Omega_{t^{n+1}})}^2\right)^{\frac{1}{2}} \\
& \times\left(h^{2(p+1)}\sum_{n=-1}^{N-1}\delta t(\tau^{n+1})^{-1}\left\|v^{n+1}\right\|_{H^{p+1}(\Omega_{t^{n+1}})}^2\right)^{\frac{1}{2}}.
\end{aligned}$$

Proof. From (6.22) we have that

$$\begin{aligned} B_h\left(X, \Pi_h^{\perp}(V)\right) = & \sum_{n=0}^{N-1} b_h\left(\boldsymbol{w}^{n+1}; x^{n+1}, \Pi_h^{\perp}\left(v^{n+1}\right)\right)_{\Omega_{t^{n+1}}} \\ & + \sum_{n=1}^{N-1} \frac{1}{2T}\left(3x^{n+1} - 4x^n + x^{n-1}, \Pi_h^{\perp}\left(v^{n+1}\right)\right)_{\Omega_{t^{n+1}}} \\ & + \frac{1}{T}\left(x^1 - x^0, \Pi_h^{\perp}\left(v^1\right)\right)_{\Omega_{t1}} + \frac{1}{T}\left(x^0, \Pi_h^{\perp}\left(v^0\right)\right)_{\Omega_0}, \end{aligned}$$

where

$$\begin{aligned} & \sum_{n=0}^{N-1} b_h\left(\boldsymbol{w}^{n+1}; x^{n+1}, \Pi_h^{\perp}\left(v^{n+1}\right)\right)_{\Omega_{t^{n+1}}} \\ & = \sum_{n=0}^{N-1} \delta t \left[\nu\left(\nabla x^{n+1}, \nabla \Pi_h^{\perp}\left(v^{n+1}\right)\right)_{\Omega_{t^{n+1}}} + \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla x^{n+1}, \Pi_h^{\perp}\left(v^{n+1}\right)\right)_{\Omega_{t^{n+1}}}\right. \\ & \quad \left. + \tau^{n+1}\left(\Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla x^{n+1}\right), (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla \Pi_h^{\perp}\left(v^{n+1}\right)\right)_{\Omega_{t^{n+1}}}\right]. \end{aligned}$$

Now we have to bound every term of the right hand side in order to complete the proof. We start with the first term:

$$\begin{aligned} & \sum_{n=0}^{N-1} \delta t \nu \left(\nabla x^{n+1}, \nabla(\Pi_h^{\perp}\left(v^{n+1}\right))\right)_{\Omega_{t^{n+1}}} \\ & \quad \leq \left(\sum_{n=0}^{N-1} \delta t \nu \left\|\nabla x^{n+1}\right\|^2_{L^2(\Omega_{t^{n+1}})}\right)^{\frac{1}{2}} \left(\sum_{n=0}^{N-1} \delta t \nu \left\|\nabla \Pi_h^{\perp}\left(v^{n+1}\right)\right\|^2_{L^2(\Omega_{t^{n+1}})}\right)^{\frac{1}{2}}. \end{aligned}$$

The second term in the right-hand side can be bounded as

$$\begin{aligned} & \sum_{n=0}^{N-1} \delta t \left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla x^{n+1}, \Pi_h^{\perp}\left(v^{n+1}\right)\right)_{\Omega_{t^{n+1}}} \\ & \quad \leq \left(\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left\|\Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla x^{n+1}\right)\right\|^2_{L^2(\Omega_{t^{n+1}})}\right)^{\frac{1}{2}} \\ & \qquad \times \left(\sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left\|\Pi_h^{\perp}\left(v^{n+1}\right)\right\|^2_{L^2(\Omega_{t^{n+1}})}\right)^{\frac{1}{2}}, \end{aligned}$$

and the third term as

$$\begin{aligned} & \sum_{n=0}^{N-1} \delta t \tau^{n+1} \left(\Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla x^{n+1}\right), (\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla \Pi_h^{\perp}\left(v^{n+1}\right)\right)_{\Omega_{t^{n+1}}} \\ & \quad \leq \left(\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left\|\Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla x^{n+1}\right)\right\|^2_{L^2(\Omega_{t^{n+1}})}\right)^{\frac{1}{2}} \\ & \qquad \times \left(\sum_{n=0}^{N-1} \delta t \tau^{n+1} \left\|\Pi_h^{\perp}\left((\boldsymbol{a} - \boldsymbol{w}^{n+1}) \cdot \nabla \Pi_h^{\perp}\left(v^{n+1}\right)\right)\right\|^2_{L^2(\Omega_{t^{n+1}})}\right)^{\frac{1}{2}}. \end{aligned}$$

The term related to the time derivative is bounded after recalling assumption (6.21) for the stabilization parameter τ^{n+1}:

$$\sum_{n=1}^{N-1}\frac{1}{T}\left(3x^{n+1}-4x^{n}+x^{n-1},\Pi_h^{\perp}\left(v^{n+1}\right)\right)_{\Omega_{t^{n+1}}}+\frac{1}{T}\left(x^{1}-x^{0},\Pi_h^{\perp}\left(v^{1}\right)\right)_{\Omega_{t^{1}}}$$
$$+\frac{1}{T}\left(x^{0},\Pi_h^{\perp}\left(v^{0}\right)\right)_{\Omega_{0}}\leq C\left(\sum_{n=-1}^{N-1}\delta t(\tau^{n+1})^{-1}\left\|\Pi_h^{\perp}\left(x^{n+1}\right)\right\|_{L^2(\Omega_{t^{n+1}})}\right)^{\frac{1}{2}}$$
$$\times\left(\sum_{n=-1}^{N-1}\delta t(\tau^{n+1})^{-1}\left\|\Pi_h^{\perp}\left(v^{n+1}\right)\right\|_{L^2(\Omega_{t^{n+1}})}\right)^{\frac{1}{2}}.$$

We now have to use (6.40) of Lemma 6.2 and the expression (6.20) of the stabilization parameter τ^{n+1} to conclude the proof.□

To obtain the error estimate, we also need to invoke the coercivity of $B_h(\cdot,\cdot)$, which leads to

$$B_h\left(E_d,E_d\right)\geq\beta_2|||E_d|||_w^2.$$

Subtracting the equation for the semi-discrete velocity and the discrete velocity we get

$$B_h\left(E_d,V_h\right)=:\epsilon_c(V_h)$$
$$=-\sum_{n=0}^{N-1}\delta t\tau^{n+1}\left(\Pi_h^{\perp}\left((\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla u^{n+1}\right),(\boldsymbol{a}-\boldsymbol{w}^{n+1})\cdot\nabla v_h^{n+1}\right)_{\Omega_{t^{n+1}}}.$$

Using the previous equation we can obtain

$$B_h\left(E_d,E_d\right)=B_h\left(E_d,\Pi_h(U)-U\right)+\epsilon_c(U_h-\Pi_h(U))$$

The first term is due to the interpolation error, whereas the second one is the consistency error. In the following lemma we obtain a bound for the interpolation error:

Lemma 6.8 (Interpolation error)**.** *The following inequality holds:*

$$B_h\left(E_d,\Pi_h^{\perp}\left(U\right)\right)\leq C\left(|||E_d|||_w^2+h^{2(p+1)}\sum_{n=-1}^{N-1}\delta t(\tau^{n+1})^{-1}\left\|u^{n+1}\right\|_{H^{p+1}(\Omega_{t^{n+1}})}^2\right)^{\frac{1}{2}}$$
$$\times\left(h^{2(p+1)}\sum_{n=-1}^{N-1}\delta t(\tau^{n+1})^{-1}\left\|u^{n+1}\right\|_{H^{p+1}(\Omega_{t^{n+1}})}^2\right)^{\frac{1}{2}}.$$

Proof. Invoking lemma 6.7 and using the fact that $\Pi_h\left(U\right)-U=-\Pi_h^{\perp}\left(U\right)$ and $\Pi_h^{\perp}\left(E_d\right)=-\Pi_h^{\perp}\left(U\right)$, we immediately get the result. □

In order to bound the consistency error we follow again the technique developed in Lemma 6.6. The only difference between these two cases is the term associated to the time derivative, which does not affect essentially the proof:

Lemma 6.9 (Consistency error)**.** *The following inequality holds:*

$$\epsilon_c(U_h-\Pi_h(U))\leq C\left(h^{2(p+1)}\sum_{n=0}^{N-1}\delta t(\tau^{n+1})^{-1}\left\|u^{n+1}\right\|_{H^{p+1}(\Omega_{t^{n+1}})}^2\right)^{\frac{1}{2}}$$
$$\times\left(|||E_d|||_w^2+h^{2(p+1)}\sum_{n=0}^{N-1}\delta t(\tau^{n+1})^{-1}\left\|u^{n+1}\right\|_{H^{p+1}(\Omega_{t^{n+1}})}^2\right)^{\frac{1}{2}}.$$

Again, we end with the desired convergence result, which is straight from Lemma 6.9 for the bound of the consistency error, Lemma 6.8 for the bound of the interpolation error and Theorem 6.8, that gives coercivity of the bilinear form:

Theorem 6.9 (Convergence)**.** *There exist δt_{cr}^2 such that, for $0 < \delta t < \delta t_{\text{cr}}^2$, the sequence of errors $E_d = U_h - U$ satisfies the following error estimate*

$$|||E_d|||_w^2 \leq Ch^{2(p+1)} \sum_{n=-1}^{N-1} \delta t (\tau^{n+1})^{-1} \left\| u^{n+1} \right\|_{H^{p+1}(\Omega_{t^{n+1}})}^2 .$$

This error estimate is optimal.

From this analysis, we can easily obtain stability and convergence results when the domain is fixed, that is, when the mesh velocity vanishes.

6.4.3 Analysis of BDF2-OSS$_{\delta t,h}$

The previous results are new even for fixed domains. The OSS stabilization method was analyzed in [55] using the backward Euler time integration. It can be easily seen that for fixed domains, i.e. when $\boldsymbol{w}^{n+1} = \mathbf{0}$, there is no critical time step size, the method becoming unconditionally stable. In this case, the problem to be solved reads as follows: find a sequence of finite element functions U_h such that

$$B_h(U_h, V_h) = L(V_h) \tag{6.53}$$

with the bilinear form

$$\begin{aligned} B_h\left(U_h, V_h\right) = \sum_{n=1}^{N-1} \left[\frac{1}{2T} \left(3u_h^{n+1} - 4u_h^n + u_h^{n-1}, v_h^{n+1} \right) + b_h \left(u_h^{n+1}, v_h^{n+1} \right) \right] \\ + \frac{1}{T} \left(u_h^1 - u_h^0, v_h^1 \right) + b_h \left(u_h^1, v_h^1 \right) + \frac{1}{T} \left(u_h^0, v_h^0 \right), \end{aligned} \tag{6.54}$$

where now $b_h(u_h^{n+1}, v_h^{n+1})$ denotes $b_h(\mathbf{0}; u_h^{n+1}, v_h^{n+1})$, with $b_h(\boldsymbol{w}^{n+1}; u_h^{n+1}, v_h^{n+1})$ defined in (6.19). The right-hand side linear form is given again by (6.14).

In this case two different sets of results are obtained. The first one with the *weak* norm $|||\cdot|||_w$, and the second one with the *strong* norm $|||\cdot|||_s$. The main difference is that in the second norm, $B_h(\cdot,\cdot)$, looses coercivity. This complicates the analysis.

We state the results with the norm $|||\cdot|||_w$ in the following corollaries. Their proofs are straightforward from the previous analysis.

Corollary 6.7 (Stability)**.** *The sequence U_h solution of problem (6.53) is bounded as follows:*

$$|||U_h|||_w^2 \leq C \sum_{n=0}^{N-1} \frac{\delta t}{\nu} \left\| f^{n+1} \right\|_{H^{-1}(\Omega)}^2 .$$

for all $\delta t > 0$.

Again, we denote by $U = \{u^0, u^1, u^2, ..., u^N\}$ the sequence of solutions of the second order semi-discrete problem (in time) (6.9)-(6.13)-(6.14), now with $\Omega_t \equiv \Omega$.

Corollary 6.8 (Convergence)**.** *The error sequence $E_d = U_h - U$ satisfies the following error estimate:*

$$|||E_d|||_w^2 \leq Ch^{2(p+1)} \sum_{n=-1}^{N-1} \delta t(\tau^{n+1})^{-1} \|u^{n+1}\|^2_{H^{p+1}(\Omega)},$$

for all $\delta t > 0$.

The remaining of this section is devoted to improve these stability and convergence estimates. The improvement consists in obtaining estimates in the *stronger* norm $|||\cdot|||_s$. This is possible for *fixed* domains, but we have not been able to obtain similar estimates as those presented next for *moving* domains. Nevertheless, some additional assumptions will be required. We will also remark the aspects that make the analysis of the BDF2-OSS$_{\delta t,h}$ method much more involved than that of the BDF1-OSS$_{\delta t,h}$ formulation.

Let us introduce some new notation. We modify the bilinear form as follows:

$$\begin{aligned} B_h^*(U_h, V_h) = &\sum_{n=0}^{N-1} b_h\left(u_h^{n+1}, v_h^{n+1}\right) + \sum_{n=1}^{N-1} \frac{1}{2T}(3u_h^{n+1} - 4u_h^n + u_h^{n-1}, v_h^{n+1}) \\ &+ \frac{1}{T}(u_h^1 - u_h^0, v_h^1) + \frac{1}{T}(u_h^0, v_h^0) + \frac{1}{T\delta t}(u_h^{-1}, v_h^{-1}), \end{aligned} \tag{6.55}$$

and the right-hand-side linear form to:

$$L^*(V_h) = \sum_{n=0}^{N-1} \delta t\langle f^{n+1}, v_h^{n+1}\rangle + \frac{1}{T}(u_0, v_h^0) + \frac{1}{T\delta t}(u_{1,h} - \Pi_h(u_0), v_h^{-1}), \tag{6.56}$$

where u_0 is obviously the initial condition and $u_{1,h}$ is the solution at the first time step obtained with the scheme used to initialize the BDF2 scheme. For example, the BDF1 scheme can be used, and this is precisely what is assumed in the expression of $B_h^*(\cdot,\cdot)$. Note that now the sequences of finite element functions start at $n = -1$.

It is easily checked that the solution of the equation (6.9) with the bilinear form (6.12) is equivalent to

$$B_h^*(U_h, V_h) = L^*(V_h).$$

Observe that this problem yields $u_h^{-1} = u_{1,h} - \Pi_h(u_0)$, $u_h^0 = \Pi_h(u_0)$ and $u_h^1 = u_{1,h}$. The rest of the terms of the sequence of unknowns $U = \{u_h^{-1}, u_h^0, u_h^2, .., u_h^N\}$ is the same as the solution of problem (6.53).

Let us introduce some additional ingredients. Given a sequence

$$V = \{v^{-1}, v^0, v^1, v^2, ..., v^N\}$$

we define

$$\begin{aligned} d^{1,*}(V) &= \{0, 0, 0, \delta v^2, \delta v^3, ..., \delta v^{N-1}, \delta v^N\}, \\ d^{2,*}(V) &= \{0, 0, -\delta v^2, -\delta^2 v^2, -\delta^2 v^3, ..., -\delta^2 v^N, \delta v^N\}. \end{aligned}$$

These operators on sequences have the following property: for all sequences $X = \{x^n\}_{n=-1}^N$ it holds that

$$\begin{aligned} B_h^*\left(X, d^{2,*}(V)\right) = &\sum_{n=1}^{N-1} b_h\left(\delta x^{n+1}, \delta v^{n+1}\right) + \sum_{n=2}^{N-1} \frac{1}{2T}(3\delta x^{n+1} - 4\delta x^n + \delta x^{n-1}, \delta v^{n+1}) \\ &+ \frac{3}{2T}(\delta x^2 - \delta x^1, \delta v^2) \\ = &B_h^*\left(d^{1,*}(X), d^{1,*}(V)\right) + \frac{1}{2T}(\delta x^1, \delta v^3 - 3\delta v^2). \end{aligned} \tag{6.57}$$

Remark 6.4. The previous property is not satisfied for moving domains due to the fact that the convective velocity changes at every time step. It introduces an extra term $b_h\left(\delta \boldsymbol{w}^{n+1}; u^n, \delta v^{n+1}\right)_{\Omega_{t^{n+1}}}$ that can not be bounded as required in the following analysis.

In the next theorem we obtain Λ-coercivity for the norm $|||\cdot|||_s$.

Theorem 6.10 (Λ-coercivity)**.** *Let $V = \{v^n\}_{n=-1}^N$ be a sequence of functions such that $v^n \in \mathcal{V}(\Omega)$, $n = 0, 1, ..., N$, and $v^{-1} = v^1 - v^0$, and consider the operator*

$$\Lambda(V) = V + \{0, 0, \frac{1}{4}\{\tau^{n+1}\Pi_h\left(\boldsymbol{a}\cdot\nabla v^{n+1}\right)\}_0^{N-1}\} + \delta t^{-1} d^{2,*}(V).$$

Then, the bilinear form $B_h^*(\cdot,\cdot)$ is Λ-coercive. In particular, the following inequality holds:

$$B_h^*(V, \Lambda(V)) \geq \beta_2 \left(|||V|||_s^2 + \delta t^{-1} |||d^{1,*}(V)|||_w^2 + \frac{1}{T\delta t}\|v^{-1}\|_{L^2(\Omega)}^2 \right)$$

for a certain constant $\beta_2 > 0$.

Proof. It can be easily shown that

$$\begin{aligned}
B_h^*(V, 4V) &= \sum_{n=0}^{N-1} 4 b_h\left(v^{n+1}, v^{n+1}\right) + \sum_{n=2}^{N-1} \frac{4}{2T}\left(3v^{n+1} - 4v^n + v^{n-1}, v^{n+1}\right) \\
&\quad + \frac{4}{T}\left(v^1 - v^0, v^1\right) + \frac{4}{T}\left(v^0, v^0\right) + \frac{4}{T\delta t}\left(v^{-1}, v^{-1}\right) \\
&\geq \sum_{n=0}^{N-1} 4\left[\delta t\nu \left\|\nabla v^{n+1}\right\|_{L^2(\Omega)}^2 + \delta t \tau^{n+1} \left\|\Pi_h^{\perp}\left(\boldsymbol{a}\cdot\nabla v^{n+1}\right)\right\|_{L^2(\Omega)}^2\right] \\
&\quad + \frac{1}{T}\left[\left\|v^{N+1}\right\|_{L^2(\Omega)}^2 + \sum_{n=1}^{N-1}\left\|\delta^2 v^{n+1}\right\|_{L^2(\Omega)}^2 + 2\left\|v^0\right\|_{L^2(\Omega)}^2 + \frac{4}{\delta t}\|v^{-1}\|_{L^2(\Omega)}^2\right].
\end{aligned}$$

In order to obtain stability for the component of the convective term in the finite element space, we use as test function the sequence $\{0, 0, \{\tau^{n+1}\Pi_h(\boldsymbol{a}\cdot\nabla v^{n+1})\}_0^{N-1}\} =: \Pi_0(\tau, V)$, that is starting with 0 in the component -1 and 0. Exactly as in the proof of Theorem 6.6, we now obtain:

$$\begin{aligned}
B_h^*(V, \Pi_0(\tau, V)) &\geq \sum_{n=0}^{N-1} \phi^{n+1}\delta t \tau^{n+1} \left\|\Pi_h\left(\boldsymbol{a}\cdot\nabla v^{n+1}\right)\right\|_{L^2(\Omega)}^2 \\
&\quad - \sum_{n=0}^{N-1}\left[\delta t\nu\left\|\nabla v^{n+1}\right\|_{L^2(\Omega)}^2 + \delta t\tau^{n+1}\left\|\Pi_h^{\perp}\left(\boldsymbol{a}\cdot\nabla v^{n+1}\right)\right\|_{L^2(\Omega)}^2\right] \\
&\quad - \sum_{n=1}^{N-1}\frac{1}{4T}\left\|3v^{n+1} - 4v^n + v^{n-1}\right\|_{L^2(\Omega)}^2 - \frac{1}{T}\left\|v^1 - v^0\right\|_{L^2(\Omega)}^2,
\end{aligned} \tag{6.58}$$

with the expression of ϕ^{n+1} given in (6.44). We do not have control over the term related to the time derivative, needing a further step. We use now as test function $d^{2,*}(V)$. From the first step in (6.57) it follows that

$$\begin{aligned}
\delta t^{-1} B_h^*\left(V, 4d^{2,*}(V)\right) &\geq \delta t^{-1}|||d^{1,*}(V)|||_w^2 - \frac{3}{T\delta t}\left\|\delta v^1\right\|_{L^2(\Omega)}^2 \\
&= \delta t^{-1}|||d^{1,*}(V)|||_w^2 - \frac{3}{T\delta t}\left\|v^{-1}\right\|_{L^2(\Omega)}^2.
\end{aligned} \tag{6.59}$$

Combining the previous inequalities and invoking the Gronwall lemma (without any assumption over the time step size) we can conclude the proof of the theorem.□

Remark 6.5. In equation (6.58) we do not have control over the term associated to the time derivative. It makes the analysis for the second order method more intricate than for the first order one, for which the time derivative term is easily controlled (see (6.43)). The control of this term has motivated the introduction of $d^{2,*}(V)$ in the test sequence used.

In order to obtain stability it remains to prove some kind of continuity with respect to the operator Λ. This is what the next theorem states:

Theorem 6.11 (Λ-continuity)**.** *The following inequality holds:*

$$\begin{aligned}
&L^*(\Lambda(V)) \\
&\leq \left(\sum_{n=0}^{N-1} \frac{\delta t}{\nu} \left\|f^{n+1}\right\|^2_{H^{-1}(\Omega)} + \sum_{n=1}^{N-1} \frac{\delta t^2}{\nu} \left\|D_t f^{n+1}\right\|^2_{H^{-1}(\Omega)}\right. \\
&\qquad \left. + \frac{1}{T}\left\|u_0\right\|^2_{L^2(\Omega)} + \frac{\delta t}{T}\left\|\frac{u_{1,h} - \Pi_h\left(u_0\right)}{\delta t}\right\|^2_{L^2(\Omega)}\right)^{\frac{1}{2}} \\
&\qquad \times \left(|||V|||_s^2 + \delta t^{-1}|||d^{1,*}(V)|||_w^2 + \frac{1}{T}\left\|v^0\right\|^2_{L^2(\Omega)} + \frac{1}{T\delta t}\|v^{-1}\|^2_{L^2(\Omega)}\right)^{\frac{1}{2}}.
\end{aligned}$$

Proof. The following inequalities can be easily obtained:

$$\begin{aligned}
L^*(V) \leq &\left(\sum_{n=0}^{N-1} \frac{\delta t}{\nu} \left\|f^{n+1}\right\|^2_{H^{-1}(\Omega)} + \frac{1}{T}\left\|u_0\right\|^2_{L^2(\Omega)} + \frac{\delta t}{T}\left\|\frac{u_{1,h} - \Pi_h\left(u_0\right)}{\delta t}\right\|^2_{L^2(\Omega)}\right)^{\frac{1}{2}} \\
&\times \left(\sum_{n=0}^{N-1} \delta t\nu \left\|\nabla v^{n+1}\right\|^2_{L^2(\Omega)} + \frac{1}{T}\left\|v^0\right\|^2_{L^2(\Omega)} + \frac{1}{T\delta t}\left\|v^{-1}\right\|^2_{L^2(\Omega)}\right)^{\frac{1}{2}},
\end{aligned}$$

$$L^*(\delta t^{-1} d^{2,*}(V)) \leq \left(\sum_{n=1}^{N-1} \frac{\delta t^2}{\nu} \left\|D_t f^{n+1}\right\|^2_{H^{-1}(\Omega)}\right)^{\frac{1}{2}} \left(\sum_{n=1}^{N-1} \delta t^2 \nu \left\|D_t v^{n+1}\right\|^2_{H^1(\Omega)}\right)^{\frac{1}{2}},$$

$$\begin{aligned}
L^*(\Pi_0(\tau, V)) \leq &\left(\sum_{n=0}^{N-1} \frac{\delta t}{\nu} \left\|f^{n+1}\right\|^2_{H^{-1}(\Omega)}\right)^{\frac{1}{2}} \\
&\times \left(\sum_{n=0}^{N-1} \delta t\nu \left\|\tau^{n+1}\nabla(\Pi_h\left(\boldsymbol{a}\cdot\nabla v^{n+1}\right))\right\|^2_{L^2(\Omega)}\right)^{\frac{1}{2}},
\end{aligned}$$

and

$$\begin{aligned}
\nu\left\|\tau^{n+1}\nabla(\Pi_h\left(\boldsymbol{a}\cdot\nabla v^{n+1}\right))\right\|^2_{L^2(\Omega)} \leq &\frac{C_{\text{inv}}^2\nu}{h^2}(\tau^{n+1})^2\left\|\Pi_h\left(\boldsymbol{a}\cdot\nabla v^{n+1}\right)\right\|^2_{L^2(\Omega)} \\
\leq &C\tau^{n+1}\left\|\boldsymbol{a}\cdot\nabla v^{n+1}\right\|^2_{L^2(\Omega)}.
\end{aligned}$$

From all these inequalities the theorem follows easily.□

The two previous theorems lead to the following stability result:

Corollary 6.9 (Stability II)**.** *The sequence U_h, solution of problem (6.53), is bounded as follows:*

$$\begin{aligned}|||U_h|||_s^2+&\delta t^{-1}|||d^{1,*}U_h|||_w^2\\ &\leq C\left(\sum_{n=0}^{N-1}\frac{\delta t}{\nu}\left\|f^{n+1}\right\|_{H^{-1}(\Omega)}^2+\sum_{n=1}^{N-1}\frac{\delta t^2}{\nu}\left\|D_t f^{n+1}\right\|_{H^{-1}(\Omega)}^2\right.\\ &\left.+\frac{1}{T}\left\|u^0\right\|_{L^2(\Omega)}^2+\frac{\delta t}{T}\left\|\frac{u_{1,h}-\Pi_h\left(u_0\right)}{\delta t}\right\|_{L^2(\Omega)}^2\right)\end{aligned}$$

for all $\delta t>0$.

Obviously, this stability bounds makes sense if the initialization is such that the last term on the right-hand-side is bounded. Using for example the backward Euler scheme, this poses a mild condition on δt and h. In particular, if $h^{p+1}\leq C\delta t$, it is easy to show that this last term is bounded.

The final result we obtain is an error estimate in the *strong* norm $|||\cdot|||_s$. At this point we introduce the sequence $U=\{u_h^{-1},u^0,u^1,u^2,...,u^N\}$, that consists of the sequence of solutions of the semi-discrete problem (6.9)-(6.11) supplemented with u_h^{-1} at $n=-1$. It can be easily checked that this sequence satisfies

$$B_h^*\left(U,V\right)=L^*(V)-\epsilon_c(V).$$

Thus, $E_d:=U_h-U=\{0,u_h^0-u^0,u_h^1-u^1,...,u_h^N-u^N\}$ satisfies

$$B_h^*\left(E_d,V_h\right)=\epsilon_c(V_h).$$

We point out that for fixed domains the critical time step size does not appear anymore due to the fact that $\boldsymbol{w}=\mathbf{0}$. The method is unconditionally stable, as expected.

We stress the fact that $e_d^{-1}\neq e_d^1-e_d^0$, and therefore E_d does not verify the statement of Theorem 6.10. The only place where the fact that $v^{-1}=v^1-v^0$ is used is in (6.59). When the test sequence does not satisfy the assumption $v^{-1}=v^1-v^0$ of Theorem 6.10, we have to modify the Λ-coercivity proved in this theorem as follows:

$$\frac{4}{T\delta t}\left\|\delta e_d^1\right\|_{L^2(\Omega)}^2+B_h^*\left(E_d,\Lambda(E_d)\right)\geq\beta_2\left(|||E_d|||_s^2+\delta t^{-1}|||d^{1,*}(E_d)|||_w^2\right). \tag{6.60}$$

With the expression of $\Lambda(\cdot)$ given in Theorem 6.10 we arrive to

$$\begin{aligned}B_h^*\left(E_d,\Lambda(E_d)\right)&=B_h^*\left(E_d,E_d\right)+\frac{1}{4}\epsilon_c(\Pi_0(\tau,E_d))+\delta t^{-1}B_h^*\left(E_d,d^{2,*}E_d\right)\\ &=B_h^*\left(E_d,\Pi_h\left(U\right)-U\right)+\epsilon_c(U_h-\Pi_h\left(U\right))+\frac{1}{4}\epsilon_c(\Pi_0(\tau,E_d))\\ &\quad+\delta t^{-1}B_h^*\left(E_d,d^{2,*}(\Pi_h\left(U\right)-U)\right)+\delta t^{-1}\epsilon_c(d^{2,*}(U_h-\Pi_h\left(U\right))).\end{aligned} \tag{6.61}$$

Again, we group the different terms as interpolation and consistency errors and bound them separately in the next lemmas.

Lemma 6.10 (Interpolation error)**.** *The following inequality holds:*

$$
\begin{aligned}
&B_h\left(E_d, \Pi_h^{\perp}(U)\right) + \delta t^{-1} B_h^*\left(E_d, d^{2,*}(\Pi_h^{\perp}(U)\right) \\
&\leq \Bigg(|||E_d|||_w^2 + \delta t^{-1} |||d^{1,*}(E_d)|||_w^2 \\
&\quad + h^{2(p+1)} \sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left(\left\| u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 + \left\| \sqrt{\delta t} D_t u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 \right) \Bigg)^{\frac{1}{2}} \\
&\quad \times \left(h^{2(p+1)} \sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left(\left\| u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 + \left\| \sqrt{\delta t} D_t u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 \right) \right)^{\frac{1}{2}}.
\end{aligned}
$$

Proof. The bound for the first term of the left-hand side of the inequality is easily obtained from the proof of Lemma 6.8, since $e_d^{-1} = 0$. For the second term we use property (6.57) and again the fact that $e_d^{-1} = 0$, getting

$$
\begin{aligned}
&B_h^*\left(E_d, d^{2,*}(\Pi_h^{\perp}(U))\right) \\
&= B_h(d^{1,*}(E_d), d^{1,*}(\Pi_h^{\perp}(U))) - \frac{1}{2T}\left(\delta e_d^1, \delta(\Pi_h\left(u^3\right) - 3\Pi_h\left(u^2\right))\right).
\end{aligned}
$$

Note that when we write $B_h(d^{1,*}(E_d), d^{1,*}(\Pi_h^{\perp}(U)))$ we eliminate the element -1 of the sequences to apply the bilinear form $B_h(\cdot,\cdot)$.

Using Lemma 6.7 we get

$$
\begin{aligned}
&\delta t^{-1} B_h(d^{1,*}(E_d), d^{1,*}(\Pi_h^{\perp}(U))) \\
&\leq C \left(\delta t^{-1} |||d^{1,*} E_d|||_w^2 + h^{2(p+1)} \sum_{n=1}^{N-1} \delta t (\tau^{n+1})^{-1} \left\| \sqrt{\delta t} D_t u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 \right)^{\frac{1}{2}} \\
&\quad \times \left(h^{2(p+1)} \sum_{n=1}^{N-1} \delta t (\tau^{n+1})^{-1} \left\| \sqrt{\delta t} D_t u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 \right)^{\frac{1}{2}}.
\end{aligned}
$$

Exploiting the fact that $\Pi_h^{\perp}\left(e_d^1\right) = \Pi_h\left(u^1\right) - u^1$ we can easily get that

$$
\begin{aligned}
&\frac{1}{2T\delta t}\left(\delta e_d^1, \delta\left(\Pi_h\left(u^3\right) - 3\Pi_h\left(u^2\right)\right)\right) \\
&\leq C h^{2(p+1)} \sum_{n=0}^{2} \delta t (\tau^{n+1})^{-1} \left\| \sqrt{\delta t} D_t u^{n+1} \right\|_{H^{p+1}(\Omega)}^2.
\end{aligned}
$$

The proof is concluded.□

Lemma 6.11 (Consistency error)**.** *The following inequality holds:*

$$
\begin{aligned}
&\epsilon_c(U_h - \Pi_h(U) + \frac{1}{4}\Pi_0(\tau, E_d) + \delta t^{-1} d^{2,*}(U_h - \Pi_h(U))) \\
&\leq C \left(h^{2(p+1)} \sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left(\left\| u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 + \left\| \sqrt{\delta t} D_t u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 \right) \right)^{\frac{1}{2}} \times \\
&\left(|||E_d|||_w^2 + h^{2(p+1)} \sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left(\left\| u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 + \left\| \sqrt{\delta t} D_t u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 \right) \right)^{\frac{1}{2}}.
\end{aligned}
$$

Proof. Due to the fact that $e_d^{-1} = 0$ we can take profit from the bounds obtained in Lemmas 6.6 and 6.9. The remaining term associated to $d^{2,*}(\cdot)$ can be bounded as follows:

$$\begin{aligned}
\epsilon_c(\delta t^{-1} d^{2,*}(U_h - \Pi_h(U))) &= \sum_{n=1}^{N-1} \tau^{n+1} \left(\Pi_h^{\perp} \left(\boldsymbol{a} \cdot \nabla \delta u^{n+1} \right), \boldsymbol{a} \cdot \nabla \Pi_h^{\perp} \left(\delta u^{n+1} \right) \right) \\
&= \sum_{n=1}^{N-1} \tau^{n+1} \left(\Pi_h^{\perp} \left(\lambda(\delta u^{n+1}) \right), \boldsymbol{a} \cdot \nabla \Pi_h^{\perp} \left(\delta u^{n+1} \right) \right) \\
&\leq C \left(\sum_{n=1}^{N-1} \delta t \tau^{n+1} \left\| \Pi_h^{\perp} \left(\sqrt{\delta t} \lambda(D_t u^{n+1}) \right) \right\|_{L^2(\Omega)}^2 \right)^{\frac{1}{2}} \\
&\quad \times \left(h^{2(p+1)} \sum_{n=1}^{N-1} \delta t (\tau^{n+1})^{-1} \left\| \sqrt{\delta t} D_t u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 \right)^{\frac{1}{2}},
\end{aligned}$$

with $\lambda(\cdot)$ introduced in Lemma 6.6. The term related to $\lambda(D_t u^{n+1})$ can be easily bounded from the expression of τ^{n+1}, assumption (6.19) and the interpolation error estimate (6.39), as pointed out in Lemma 6.6.□

We end with the convergence result of the method in the norm $|||\cdot|||_s$:

Theorem 6.12 (Convergence II)**.** *The sequence of errors $E_d = U_h - U$ satisfies the following error estimate:*

$$\begin{aligned}
|||E_d|||_s^2 \leq & C h^{2(p+1)} \left[\sum_{n=0}^{N-1} \delta t (\tau^{n+1})^{-1} \left(\left\| u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 + \left\| \sqrt{\delta t} D_t u^{n+1} \right\|_{H^{p+1}(\Omega)}^2 \right) \right. \\
& \left. + (\tau^1)^{-1} \left\| u^1 \right\|_{H^{p+1}(\Omega)} + (\tau^1)^{-1} \left\| u^0 \right\|_{H^{p+1}(\Omega)} \right]
\end{aligned} \tag{6.62}$$

for all $\delta t > 0$.

Proof. Using Lemma 6.10 and Lemma 6.11 in expressions (6.60) and (6.61), we can easily get the desired bound for $|||E_d|||_s^2$ in terms of $\frac{4}{T\delta t} \left\| \delta e_d^1 \right\|_{L^2(\Omega)}^2$. Using as initialization the backward Euler scheme and the convergence result of Theorem 6.7 for the semidiscrete problem, it follows that

$$\begin{aligned}
\frac{1}{T\delta t} \left\| \delta e_d^1 \right\|_{L^2(\Omega)}^2 &\leq \frac{C}{T\delta t} \left(\left\| e_d^1 \right\|_{L^2(\Omega)}^2 + \left\| u^0 - \Pi_h(u^0) \right\|_{L^2(\Omega)}^2 \right) \\
&\leq C (\tau^1)^{-1} h^{2(p+1)} \left(\left\| u^1 \right\|_{H^{p+1}(\Omega)} + \left\| u^0 \right\|_{H^{p+1}(\Omega)} \right),
\end{aligned}$$

from which we obtain the desired result.□

Remark 6.6. From (6.62) it is seen that we need $\{\sqrt{\delta t} D_t u^{n+1}\}$ bounded in the norm of $\ell^2(H^{p+1}(\Omega))$. This can be understood as additional regularity on the data or as an additional assumption on the asymptotic behavior of the time step size in terms of h. From the semidiscrete equation, it is immediate to bound $\|D_t u^{n+1}\|_{H^q(\Omega)}$ in terms of the $H^q(\Omega)$-norm of the rest of the terms of the equation. In particular, the viscous term implies that the $H^q(\Omega)$-norm of $D_t u^{n+1}$ can be bounded in terms of the $H^{q+2}(\Omega)$-norm of u^{n+1}. If only the $H^{p+1}(\Omega)$-norm of u^{n+1} is bounded, we have to take $q = p - 1$, and thus $h^{2(p+1)} \|\sqrt{\delta t} D_t u^{n+1}\|_{H^{p+1}(\Omega)}^2$ has to be replaced by $h^{2(p-1)} \|\sqrt{\delta t} D_t u^{n+1}\|_{H^{p-1}(\Omega)}^2$, and therefore we need $\delta t \leq C h^4$ in order to maintain the optimal order of accuracy.

6.5 Conclusions

In this chapter we have analyzed a stabilized finite element method for the approximation of the convection-diffusion equation on moving domains. The Orthogonal Subgrid Scale formulation has been used as stabilization technique and an ALE framework has been used in order to deal with moving domains.

In the first part of this chapter we have analyzed the semi-discrete problem (in time). Two methods have been considered. A first order accurate method, where the time derivatives are computed using the BDF1 scheme, and a second order accurate method, where the BDF2 scheme has been used. In this analysis it is easy to identify the error introduced by the ALE formulation. The mesh velocity is computed as the time derivative of the mesh displacement. The numerical approximation of this time derivative is the only source of error introduced by the ALE formulation. As a conclusion, *in order to keep the accuracy of a k-th order (in time) method on fixed domains, we must compute the mesh velocity using a time integration scheme of, at least, order k of accuracy.* The only negative aspect is that *unconditional stable methods for fixed domains become conditionally stable.*

In the second part of this chapter we have analyzed a stabilized transient convection-diffusion equation in an ALE framework. We have introduced the concept of Λ-coercivity that has been used for obtaining stability results and error estimates. It has been shown that the OSS method can be easily extended to transient problems. For the BDF1 time integration scheme we have stability of the convective term norm, as usual when using stabilization techniques. The analysis of BDF2 is more complicated. We only have control over the *orthogonal projection* of the convective term. However, optimal convergence results with constants that do not depend on the Péclet number can be proved. Finally, for fixed domains, we have been able to recover *stronger* stability and convergence involving the full norm of the convective term, but the analysis is much more involved and requires more regularity assumptions.

Chapter 7

Fluid-Structure Interaction problems

In this chapter we introduce the framework used for the statement of the fluid-structure interaction problem. Different kinds of iterative coupling algorithms are introduced. The blend of fixed point iterative algorithms and pressure segregation methods are suggested for the real engineering applications simulated in the next chapter. Reference [6] contains this work.

7.1 Introduction

The interaction between a fluid and a structure appears in a wide variety of fields. Probably, the most analyzed fluid-structure interaction problem is the aeroelastic one (specially for aeronautical applications), for instance in the simulation of the action of a fluid (air) over a structure (such as a wing or a bridge...). Recently, an increasing interest in the simulation of haemodynamics has motivated a lot of research on fluid-structure algorithms appropriate for the blood-vessel system.

The implementation of a coupled problem can be done using two different global strategies. The *monolithic* strategy implies the solution of the coupled problems simultaneously (see [17]). *Partitioned* methods are usually used in order to keep software modularity and allow the use of the numerical methods developed for every field separately. When using pressure segregation methods for the fluid problem (as in this work) *partitioned* procedures are naturally adapted., since a global iterative scheme is already needed to couple the velocity and pressure calculations.

The numerical simulation of the fluid-structure coupled problem is complicated. It does not only inherit the difficulties associated to the fluid and solid simulation, but the coupling of these two systems is also cumbersome in many situations. The difficulties arising from this coupled system depend strongly on the physical properties of the case to be simulated. Thus, the choice of an appropriate algorithm that deals well with the coupling varies with the problem to solve. For instance, applications in aeroelasticity and haemodynamics have very different behavior for the same coupling algorithm. Whereas for aeroelastic problems there is a clear tendency to solve the coupled system using explicit procedures, these methods are not appropriate in most haemodynamics applications. In the last case, the use of special implicit procedures for the coupling are required in order to reach good convergence. This situation can be explained by the *added mass* effect (see [37]). When the structure density ρ_s is much larger than the fluid density ρ_f (as it

happens in aeroelasticity), the coupling procedure is more stable. On the contrary, when the fluid and structure densities are of the same order (as in haemodynamics) the *added mass* introduced by the fluid over the structural problem makes the convergence of the coupling algorithm much more involved.

The use of explicit procedures for the coupling has been deeply studied by Farhat and co-authors in the framework of aeroelasticity in [67, 69, 70, 135]. Therein they suggest improved explicit procedures that minimize the virtual energy introduced by the explicit algorithm, and added emphasis is placed on paralelization. The application they have in mind is the interaction between an aircraft and a compressible flow surrounding it. Fluid-structure algorithms for aeroelastic problems have also been used in civil and mechanical engineering.

On the other side, for haemodynamics, more elaborated algorithms are needed for the coupling. In order to obtain convergence, Newton and quasi-Newton algorithms have been suggested (see [72, 79]). In [154, 61] some methods motivated from a domain decomposition approach to the fluid-structure problem have been proposed. The relaxation of these methods is a key aspect in order to reach convergence when dealing with these problems, and some possibilities have been used (see [129, 60]). In [37] a simplified blood-vessel system is studied, giving a nice explanation of the *added mass* effect and the big impact of the relaxation on the convergence. Alternatively, some kind of relaxation can be introduced with a pseudo-compressibility. The introduction of a pseudo-compressibility that vanishes when the convergence of the coupling is reached has been used in [139] for the simulation of a fluid in an elastic cavity.

We can say that, for a given time step size, explicit procedures are cheaper than implicit procedures. However, explicit procedures are also less accurate. Moreover, when using explicit techniques we are restricted to *small enough* time step sizes or otherwise the solution explodes. Implicit procedures allow larger time step sizes. But, depending on the problem, convergence can be a delicate aspect and involved implicit procedures can be required.

Herein we want to obtain appropriate algorithms for the simulation of fluid-structure problems using finite element methods. The interpretation of the coupling of the fluid and structure as a domain decomposition method without overlapping used in [138] is adopted. Further, the *linearization* of the Steklov-Poincaré operator associated to the fluid is suggested and exploited. We apply these algorithms to the aeroelastic analysis of bridges (see Chapter 8). We assume a Newtonian and incompressible fluid. The structure, as it is usually done in the analysis of these problems, is considered a rigid body with elastic coefficients in the rigid body motion degrees of freedom.

Let us list what we need in order to solve a fluid-structure problem. In these problems the displacement of the structure changes the domain of the fluid. Then, the fluid equations have to be able to deal with moving domains. With this aim we use an ALE (*Arbitrary Lagrangian Eulerian*) approach. Some ALE formulations have been analyzed in [74, 80, 19, 4] and Chapter 6 . Comments about the relationship between the stability of these methods and the *geometric conservation law* can be found in [130, 68]. The ALE scheme has an intrinsic error in time that can spoil the accuracy of the fluid solver in fixed domains. For this reason, an appropriate ALE scheme depends on the time accuracy of the fluid solver for fixed domains. The ALE approach involves the movement of the domain (mesh) with appropriate Dirichlet boundary conditions. This movement is defined by a mesh displacement. Different techniques have been proposed for its computation. The most widely used is the harmonic extension of the Dirichlet functions on the boundaries, being this methodology the one adopted in the present work.

The fluid solver for incompressible flows is a key point of the algorithm because it consumes most of the CPU time. The monolithic treatment of the Navier-Stokes equations is involved (for the system solver) and time consuming. In order to improve the situation, we suggest the use of pressure segregation methods in their fractional step and predictor corrector forms (see [57, 52, 56] and Chapter 3). On the other hand, we use the orthogonal subgrid scale stabilized finite element method (see Chapter 2 and [48]) for the space discretization, that allows the use of equal velocity-pressure interpolation.

Less attention is paid to the structure solver. The following exposition can be applied to any kind of structural problem, with linear or nonlinear material behavior. Nevertheless, in the application we have considered, the structure is considered a rigid body. Thus, the computational cost of the structure is much lower than the computational cost of the fluid.

We have organized the present work as follows: In Section 7.2 we state every field problem in its continuous level and some notation is introduced. We write the strong and weak form of the governing equations of the coupled problem. In Section 7.3 we write the interface equation associated to the problem under consideration, using a Domain Decomposition framework. Some methods have been listed. Finally, at the fully discrete level, we introduce the fluid solvers and appropriate coupling procedures (Section 7.4). In particular, pressure segregation methods are suggested. In Section 7.5 we justify the algorithms chosen for the numerical experimentation and applications. In the next chapter we apply these methods to the simulation of bridge aerodynamics.

7.2 The continuous problem

In this section we introduce the fluid-structure problem at the continuous level. Firstly, we treat some aspects about the problem domain, the definition of its movement and its restriction to the fluid and structure, the domain velocity and the matching conditions that these restrictions satisfy on the interface. Secondly, we state the governing equations of the fluid and structure problems and suggest how to calculate the domain displacement. We conclude this section with the matching conditions (that is, continuity of some values) that have to be imposed over the interface between the fluid and the solid.

We denote by Ω_t the domain occupied by the heterogeneous mechanical system at a given time $t > 0$. This domain is divided into the structure domain Ω_t^s and its complement Ω_t^f occupied by the fluid. We denote by $\Sigma_t \equiv \partial\Omega_t^f \cap \partial\Omega_t^s$ the fluid-structure interface. Further, $\boldsymbol{n}_f$ is the outward normal of Ω_t^f on Σ_t and $\boldsymbol{n}_s$ its counterpart for the structure.

The total domain Ω_t is defined at every time instant by a family of mappings $\mathcal{A}_t$

$$\mathcal{A}_t : \Omega_0 \longrightarrow \Omega_t,$$

where Ω_0 is the reference domain associated to $t = 0$. We also define its restriction $\mathcal{A}_t^s := \mathcal{R}_{\Omega_t^s}(\mathcal{A}_t)$ over Ω_t^s and $\mathcal{A}_t^f := \mathcal{R}_{\Omega_t^f}(\mathcal{A}_t)$ over Ω_t^f, such that

$$\mathcal{A}_t^s : \Omega_0^s \longrightarrow \Omega_t^s,$$
$$\mathcal{A}_t^f : \Omega_0^f \longrightarrow \Omega_t^f,$$

being again Ω_0^s and Ω_0^f the fluid and structure domains at $t = 0$.

From the trace theorem (see [138]) applied to $\mathcal{A}_t$, we know that

$$\mathcal{A}_t^s|_{\Sigma_t} = \mathcal{A}_t^f|_{\Sigma_t},$$

where $|_{\Sigma_t}$ denotes the restriction to Σ_t. We stress the fact that $\mathcal{A}_t$ is arbitrary.

Let us introduce some notation. Given a function $f : \Omega_t \times [0,T] \longrightarrow \mathbb{R}$ defined at the current domain we indicate by $\hat{f} = f \circ \mathcal{A}_t$ the corresponding function defined at the initial configuration,

$$\hat{f} : \Omega_0 \times [0,T] \longrightarrow \mathbb{R}, \quad \hat{f}(\boldsymbol{x}_0, t) = f(\mathcal{A}_t(\boldsymbol{x}_0), t).$$

Furthermore, the time derivatives at the initial configuration are defined as follows:

$$\left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{x}_0} : \Omega_t \times (0,T) \longrightarrow \mathbb{R}, \quad \left.\frac{\partial f}{\partial t}\right|_{\boldsymbol{x}_0}(\boldsymbol{x}, t) = \frac{\partial \hat{f}}{\partial t}(\boldsymbol{x}_0, t).$$

We denote by $\boldsymbol{d}(\boldsymbol{x}, t)$ the displacement of the domain evaluated at the current configuration. Then, we could write the mapping $\mathcal{A}_t$ as $\mathcal{A}_t(\boldsymbol{x}_0, t) = \boldsymbol{x}_0 + \widehat{\boldsymbol{d}}(\boldsymbol{x}_0, t)$. As before, we split the domain displacement into its fluid and structure restriction as $\boldsymbol{d} = \mathcal{R}_{\Omega_t^s}\boldsymbol{d} + \mathcal{R}_{\Omega_t^f}\boldsymbol{d} =: \boldsymbol{d}^s + \boldsymbol{d}^f$. Again, from the trace theorem we know that

$$\boldsymbol{d}^s|_{\Sigma_t} = \boldsymbol{d}^f|_{\Sigma_t} \tag{7.1}$$

has to be satisfied. Moreover, we define

$$\boldsymbol{w} = \left.\frac{\partial \boldsymbol{d}^f}{\partial t}\right|_{\boldsymbol{x}_0}, \tag{7.2}$$

which is the domain velocity that we will require in order to write the fluid equations in an ALE framework.

In the present work we assume a Newtonian incompressible fluid. We use the ALE formulation in order to write the Navier-Stokes equations on moving domains. In what follows we only consider the boundary conditions on Σ_t. The rest of boundary conditions are essential for the definition of the problem but do not affect the following exposition. For this reason we have omitted them for the sake of clarity. The Navier-Stokes equations that govern the fluid problem read as follows: find a velocity field $\boldsymbol{u}$ and a pressure field p such that

$$\rho_f \frac{\partial \boldsymbol{u}}{\partial t} - \mu\Delta\boldsymbol{u} + \rho_f \boldsymbol{u}\cdot\nabla\boldsymbol{u} + \nabla p = \rho_f \boldsymbol{f}_f \qquad \text{in } \Omega_t^f \times (0,T), \tag{7.3a}$$

$$\nabla\cdot\boldsymbol{u} = 0 \qquad \text{in } \Omega_t^f \times (0,T), \tag{7.3b}$$

where ρ_f is the density and μ the viscosity. The Cauchy stress tensor for the fluid is $\boldsymbol{\sigma}^f = -p\boldsymbol{I} + 2\mu\boldsymbol{\epsilon}(\boldsymbol{u})$ where $\boldsymbol{\epsilon}(\boldsymbol{u}) = (\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T)/2$ is the strain rate tensor and $\boldsymbol{I}$ the identity matrix. We denote by $\boldsymbol{\sigma}^f_{\boldsymbol{n}} := \boldsymbol{\sigma}^f|_{\Sigma_t}\cdot\boldsymbol{n}_f$ the normal stress on Σ_t.

In (7.3) we have slightly changed the notation introduced in the Preliminaries. We denote by p the pressure field, instead of the kynematic pressure. This format is more common in fluid-structure works.

Let us recall the *Reynolds transport formula*. Let $\psi(\boldsymbol{x}, t)$ be a function defined on Ω_t. Then, for any subdomain $V_t \subseteq \Omega_t$ such that $V_t = \mathcal{A}_t(V_0)$ with $V_0 \subseteq \Omega_0$ it holds that

$$\frac{d}{dt}\int_{V_t}\psi(\boldsymbol{x}, t)\,\mathrm{d}V = \int_{V_t}\left(\left.\frac{\partial\psi}{\partial t}\right|_{\boldsymbol{x}_0} + \psi\nabla\cdot\boldsymbol{w}\right)\mathrm{d}V. \tag{7.4}$$

At this point, using expression (7.4) for the time derivative, we can write the fluid equations (7.3) in the ALE frameworkas follows: find a velocity $\boldsymbol{u}$ and a pressure p such that

$$\begin{aligned} \rho_f \left.\frac{\partial \boldsymbol{u}}{\partial t}\right|_{\boldsymbol{x}_0} - \mu\Delta\boldsymbol{u} + \rho_f(\boldsymbol{u}-\boldsymbol{w})\cdot\nabla\boldsymbol{u} + \nabla p = \rho_f \boldsymbol{f}_f \qquad & \text{in } \Omega_t^f \times (0,T), \\ \nabla\cdot\boldsymbol{u} = 0 \qquad & \text{in } \Omega_t^f \times (0,T). \end{aligned} \tag{7.5}$$

Remark 7.1. We remark that formulations (7.3) and (7.5) are equivalent at the continuous level.

The structure can easily handle with moving domains using a fully Lagrangian framework. For instance, if we consider an elastic structure, the problem that governs the displacement field on the structure is: find $\widehat{\boldsymbol{d}}^s(\boldsymbol{x}_0,t)$ such that

$$\rho_s \frac{\partial^2 \widehat{\boldsymbol{d}}^s}{\partial t^2} - \nabla|_{\boldsymbol{x}_0}\cdot(\widehat{\boldsymbol{\sigma}}^s(\widehat{\boldsymbol{d}}^s)) = \rho_s \boldsymbol{f}_s \qquad \text{in } \Omega_0^s \times (0,T), \tag{7.6}$$

where ρ_s is the solid density, $\boldsymbol{f}_s$ is the vector of body forces exerted on the solid and $\widehat{\boldsymbol{\sigma}}^s(\boldsymbol{x}_0,t)$ is the Piola-Kirchoff stress tensor for the solid at the reference configuration. We denote by $\boldsymbol{\sigma}^s_{\boldsymbol{n}} := \boldsymbol{\sigma}^s|_{\Sigma_t}\cdot\boldsymbol{n}_s$ the normal stress on Σ_t.

The displacement of the structure domain $\boldsymbol{d}^s$ has been assumed equal to the structure displacement obtained from (7.6).

The introduction of boundary conditions on Σ_t for problems (7.5) and (7.6) in order for the heterogeneous problem to be well-posed are stated below.

The fluid displacement $\boldsymbol{d}^f$ is arbitrary but has to satisfy condition (7.1). Thus, we can write $\boldsymbol{d}^f$ as an arbitrary extension of $\boldsymbol{d}^s|_{\Sigma_t}$ into Ω_t^f, that we denote by

$$\boldsymbol{d}^f = Ext\left(\boldsymbol{d}^s|_{\Sigma_t}\right).$$

Different choices of the lifting operator $\mathrm{Ext}(\cdot)$ have been proposed in the literature. Herein, we adopt an harmonic extension evaluated at the current domain Ω_t^f. In this case, $\boldsymbol{d}^f$ is solution of the Laplace problem

$$\Delta\boldsymbol{d}^f = \boldsymbol{0} \qquad \text{in } \Omega_t^f \times (0,T), \tag{7.7a}$$

$$\boldsymbol{d}^f = \boldsymbol{d}^s \qquad \text{on } \Sigma_t \times (0,T). \tag{7.7b}$$

This extension is different from the harmonic extension evaluated at Ω_0^f, used, e.g., in [130].

At this point, suitable matching conditions have to be applied on the interface Σ_t. These are continuity of normal stresses (due to the action-reaction principle) and velocities (due to the perfect adherence of the fluid to the structure):

$$\boldsymbol{u} = \frac{\partial \widehat{\boldsymbol{d}}^s}{\partial t} \qquad \text{on } \Sigma_t \times (0,T), \tag{7.8}$$

$$\boldsymbol{\sigma}^f_{\boldsymbol{n}} + \boldsymbol{\sigma}^s_{\boldsymbol{n}} = \boldsymbol{0} \qquad \text{on } \Sigma_t \times (0,T). \tag{7.9}$$

Then, the fluid-structure coupled problem is completely defined by the fluid problem (7.5), the structure problem (7.6), the fluid domain displacement (7.7) and the interface

matching conditions (7.1), (7.8) and (7.9). For the space discretization of the equations, let us to write the weak form of the system. Given $t \in (0,T)$, the functional spaces

$$\begin{aligned}
\mathcal{V}(\Omega_t^f) &:= \left\{ \boldsymbol{v} : \Omega_t^f \to \mathbb{R}^d, \boldsymbol{v} = \hat{\boldsymbol{v}} \circ (\mathcal{A}_t^f)^{-1}, \hat{\boldsymbol{v}} \in (H^1(\Omega_0^f))^d \right\}, \\
\mathcal{V}_0(\Omega_t^f) &:= \left\{ \boldsymbol{v} \in \mathcal{V}(\Omega_t^f) | \, \boldsymbol{v}|_{\Sigma_t} = \boldsymbol{0} \right\}, \\
\mathcal{Y}(\Omega_t^s) &:= \left\{ \boldsymbol{y} : \Omega_t^s \to \mathbb{R}^d, \boldsymbol{y} = \hat{\boldsymbol{y}} \circ (\mathcal{A}_t^s)^{-1}, \hat{\boldsymbol{y}} \in (H^1(\Omega_0^s))^d \right\}, \\
\mathcal{Q}(\Omega_t^f) &:= \left\{ q : \Omega_t^f \to \mathbb{R}, q = \hat{q} \circ (\mathcal{A}_t^f)^{-1}, \hat{q} \in L^2(\Omega_0^f) \right\}, \\
\Gamma(\Sigma_t) &:= \left\{ \boldsymbol{\gamma} : \Sigma_t \to \mathbb{R}^d, \boldsymbol{\gamma} = \hat{\boldsymbol{\gamma}} \circ (\mathcal{A}_t|_{\Sigma_t})^{-1}, \hat{\boldsymbol{\gamma}} \in (H^{1/2}(\Sigma_0))^d \right\},
\end{aligned}$$

will allow us to write the governing equations of the fluid (7.5) and structure (7.6) in their weak forms. The notation used here is as follows: $L^2(\omega)$ denotes the space of square integrable functions in a spatial domain ω, $H^1(\omega)$ is the space of functions in $L^2(\omega)$ with first derivatives in $L^2(\omega)$, and $H^{1/2}(\sigma)$ is the space of functions defined on a $d-1$-manifold σ that are the trace of functions in $H^1(\omega)$, with $\sigma \subset \partial\omega$. For functions f and g defined on a d- or $d-1$-manifold, we write $\langle f, g \rangle_\omega := \int_\omega fg \, \mathrm{d}\omega$, omitting the subscript when ω is the domain where the problem under consideration is posed. For σ a $d-1$-manifold and $f \in H^{1/2}(\sigma)$, the space of functions g such that $\langle f, g \rangle_\sigma < \infty$ is denoted by $H^{-1/2}(\sigma)$. Finally, $(\cdot, \cdot)$ denotes the usual L^2 product in the domain where the problem considered is posed.

Assuming that $\boldsymbol{u}(t)$ is continuous in time for simplicity, the variational form of (7.5) for a given time value $t \in (0,T)$ reads: find $\boldsymbol{u}(t) \in \mathcal{V}(\Omega_t^f)$ and $p(t) \in \mathcal{Q}(\Omega_t^f)$ such that

$$\begin{aligned}
\rho_f \left(\frac{\partial \boldsymbol{u}}{\partial t}, \boldsymbol{v} \right) + \mu \left(\nabla \boldsymbol{u}, \nabla \boldsymbol{v} \right) + \rho_f \left((\boldsymbol{u} - \boldsymbol{w}) \cdot \nabla \boldsymbol{u}, \boldsymbol{v} \right) - \left(p, \nabla \cdot \boldsymbol{v} \right) &= \rho_f \langle \boldsymbol{f}_f, \boldsymbol{v} \rangle \quad \forall \boldsymbol{v} \in \mathcal{V}_0(\Omega_t^f), \\
\left(\nabla \cdot \boldsymbol{u}, q \right) &= 0 \qquad\qquad \forall q \in \mathcal{Q}(\Omega_t^f).
\end{aligned}$$

The weak form of the structure system (7.6) for a given time value $t \in (0,T)$ is: find $\widehat{\boldsymbol{d}}^s(t) \in \mathcal{Y}(\Omega_0^s)$ such that

$$\rho_s \left(\frac{\partial^2 \widehat{\boldsymbol{d}}^s}{\partial t^2}, \boldsymbol{y} \right) + \left((\widehat{\boldsymbol{\sigma}}^s(\widehat{\boldsymbol{d}}^s)), \nabla|_{\boldsymbol{x}_0} \boldsymbol{y} \right) = \rho_s \langle \boldsymbol{f}_s, \boldsymbol{y} \rangle + \langle \boldsymbol{\sigma}_{\boldsymbol{n}}^s, \boldsymbol{y} \rangle_{\Sigma_t} \qquad \forall \boldsymbol{y} \in \mathcal{Y}(\Omega_0^s).$$

Finally, the boundary conditions that have to be imposed on Σ_t for the *weak* formulation of the coupled problem are:

$$\begin{aligned}
\langle \boldsymbol{\sigma}_{\boldsymbol{n}}^s, \boldsymbol{\gamma} \rangle_{\Sigma_t} + \langle \boldsymbol{\sigma}_{\boldsymbol{n}}^f, \boldsymbol{\gamma} \rangle_{\Sigma_t} &= 0 \qquad\qquad \forall \boldsymbol{\gamma} \in \Gamma(\Sigma_t) \times (0,T), \\
\boldsymbol{u} &= \left. \frac{\partial \boldsymbol{d}^s}{\partial t} \right|_{\Sigma_t} \qquad \text{on } \Sigma_t \times (0,T),
\end{aligned}$$

where $\boldsymbol{\sigma}_{\boldsymbol{n}}^s$ and $\boldsymbol{\sigma}_{\boldsymbol{n}}^f$ belong to $(H^{-1/2}(\Sigma_t))^d$.

7.3 The domain decomposition approach

In this section we reformulate the fluid-structure problem in a Domain Decomposition (DD onwards) framework, as done in [154] and later works [61, 60]. First, the fluid problem is introduced in this framework, and after that, the structure problem. The resulting

interface equation is written in different forms, in order to justify the use of different algorithms suggested in the literature for the fluid-structure problem.

Let us consider the time discretized version of (7.5) using BDF for the time integration at the time step $t^{n+1} = (n+1)\delta t$, being $\delta t > 0$ the time step size. We use the notation introduced in Section 1.5 for the time discretization.

At a fixed time step $n+1$, let us denote by $\boldsymbol{\lambda}$ the interface variable corresponding to the displacement on the fluid-structure interface $\boldsymbol{d}|_{\Sigma_{t^{n+1}}}$. We denote by $\mathsf{FL}_{\delta t}$ the operator that gives the velocity and pressure field at t^{n+1} for a given $\boldsymbol{\lambda}$,

$$\begin{aligned} \mathsf{FL}_{\delta t} : \Gamma(\Sigma_{t^{n+1}}) &\to \mathcal{V}(\Omega^f_{t^{n+1}}) \times \mathcal{Q}(\Omega^f_{t^{n+1}}) \\ \boldsymbol{\lambda} &\mapsto (\boldsymbol{u}^{n+1}, p^{n+1}) \end{aligned}$$

There are multiple choices for the $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda})$ operator, corresponding to the different possibilities for the time approximation of the incompressible Navier-Stokes equations, such as the monolithic system or the fractional step version at the continuous level in space (see [157]). Let us start with the monolithic scheme, denoted by $\mathsf{MN}_{\delta t}(\boldsymbol{\lambda})$. In this case, $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda}) = (\boldsymbol{u}^{n+1}, p^{n+1})$ is computed by solving the problem: given $\boldsymbol{\lambda} \in \Gamma(\Sigma_{t^{n+1}})$, find $\boldsymbol{u}^{n+1} \in \mathcal{V}(\Omega^f_{t^{n+1}})$ and $p^{n+1} \in \mathcal{Q}(\Omega^f_{t^{n+1}})$ such that

$$\frac{\rho_f}{\delta t}\left(D_k \boldsymbol{u}^{n+1}, \boldsymbol{v}\right) + \mu\left(\nabla \boldsymbol{u}^{n+1}, \nabla \boldsymbol{v}\right) + \rho_f\left((\boldsymbol{u}^{n+1} - \boldsymbol{w}^{n+1}) \cdot \nabla \boldsymbol{u}^{n+1}, \boldsymbol{v}\right)$$

$$-(p^{n+1}, \nabla \cdot \boldsymbol{v}) = \rho_f \langle \boldsymbol{f}_f^{n+1}, \boldsymbol{v} \rangle \qquad \forall \boldsymbol{v} \in \mathcal{V}_0(\Omega^f_{t^{n+1}}), \tag{7.10a}$$

$$\left(\nabla \cdot \boldsymbol{u}^{n+1}, q\right) = 0 \qquad \forall q \in \mathcal{Q}(\Omega^f_{t^{n+1}}), \tag{7.10b}$$

$$\boldsymbol{u}^{n+1} = \frac{1}{\delta t \gamma_k}\left(\boldsymbol{\lambda} + \sum_{i=0}^{k-1} \alpha_k^i \boldsymbol{d}^{n-i}\right) \qquad \text{on } \Sigma_{t^{n+1}}. \tag{7.10c}$$

being γ_p and α_p^i the parameters that define the BDF scheme of order p, listed in Section 1.5.

Borrowing classical concepts from domain decomposition methods, we can define the *Steklov-Poincaré interface operator* (see [138]) for the fluid as follows: $\mathcal{S}_f$ is the Dirichlet-to-Neumann map in Ω_t^f such that

$$\begin{aligned} \mathcal{S}_f : H^{1/2}(\Sigma_t) &\to H^{-1/2}(\Sigma_t) \\ \boldsymbol{\lambda} &\mapsto \boldsymbol{\sigma}_{\boldsymbol{n}}^f. \end{aligned} \tag{7.11}$$

This operator consists of solving the fluid problem given a value for the interface variable $\boldsymbol{\lambda}$, that is $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda})$, and recover the normal stress on the interface $\boldsymbol{\sigma}_{\boldsymbol{n}}^f$. Thus, this is a mapping between the trace of the displacement field $\boldsymbol{d}$ and the space of normal stresses exerted by the fluid. Obviously, this operator depends on the fluid solver used, $\mathsf{FL}_{\delta t}$.

We point out that the Steklov-Poincaré operator $\mathcal{S}_f$ for the fluid is nonlinear. It involves two different non-linearities: one associated to the convective term of the Navier-Stokes equations, and a second one due to the fact that the fluid domain $\Omega_t^f \equiv \Omega_t^f(\boldsymbol{\lambda})$ does depend on the interface variable (shape non-linearity). This implies that the superposition of problems cannot be used and thus $\mathcal{S}_f$ has to deal also with forcing terms and non-homogeneous boundary conditions.

Analogously for the structure, we define the Steklov-Poincaré operator: $\mathcal{S}_s$ is the Dirichlet-to-Neumann map in Ω_t^s such that

$$\begin{aligned} \mathcal{S}_s : H^{1/2}(\Sigma_t) &\to H^{-1/2}(\Sigma_t) \\ \boldsymbol{\lambda} &\mapsto \boldsymbol{\sigma}_{\boldsymbol{n}}^s. \end{aligned} \tag{7.12}$$

In this case $\mathcal{S}_s$ consists of solving the structure problem using $\boldsymbol{\lambda}$ as Dirichlet boundary condition for $\boldsymbol{d}^s$ on Σ_t and extract the value of the normal stress $\boldsymbol{\sigma}_{\boldsymbol{n}}^s$ on Σ_t. Therefore, this is a mapping between the trace of the displacement field $\boldsymbol{d}$ and the space of normal stresses exerted by the structure. Again, this operator is nonlinear even for linear constitutive equations (as the elastic case considered) because of the *shape derivative* (the deformation of the solid domain). Let us introduce also $\mathcal{S}_s^{-1}$, which is the so called *Poincaré-Steklov interface operator*: $\mathcal{S}_s^{-1}$ is the Neumann-to-Dirichlet map in Ω_t^s such that

$$\begin{aligned} \mathcal{S}_s^{-1} : H^{-1/2}(\Sigma_t) &\to H^{1/2}(\Sigma_t) \\ \boldsymbol{\sigma}_{\boldsymbol{n}}^s &\mapsto \boldsymbol{\lambda}. \end{aligned} \tag{7.13}$$

The operator $\mathcal{S}_s^{-1}$ consists of solving the structure problem using $\boldsymbol{\sigma}_{\boldsymbol{n}}^s$ as Neumann boundary condition on Σ_t and recover $\boldsymbol{d}^s$ on the boundary. $\mathcal{S}_s^{-1}$ will be used for fixed point algorithms.

At this point the interface condition (7.9) that involves continuity of normal stresses on Σ_t can be easily rewritten as: find $\boldsymbol{\lambda} \in \Gamma(\Sigma_{t^{n+1}})$ such that

$$\mathcal{S}_f(\boldsymbol{\lambda}) + \mathcal{S}_s(\boldsymbol{\lambda}) = \mathbf{0}. \tag{7.14}$$

Thus, using the DD approach the initial coupled problem has been reduced to an interface equation.

An alternative form of the interface equation, obtained by applying the inverse of the Steklov-Poincaré operator $\mathcal{S}_s^{-1}$ in (7.14), reads as: find $\boldsymbol{\lambda} \in \Gamma(\Sigma_{t^{n+1}})$ such that

$$-\mathcal{S}_s^{-1}(\mathcal{S}_f(\boldsymbol{\lambda})) = \boldsymbol{\lambda}. \tag{7.15}$$

This expression motivates the use of the fixed point algorithm (see [38]). The iterative fixed point procedure can be written as: given $\boldsymbol{\lambda}^k$ with $k \geq 0$, find $\boldsymbol{\lambda}^{k+1}$ such that

$$-\mathcal{S}_s^{-1}(\mathcal{S}_f(\boldsymbol{\lambda}^k)) = \boldsymbol{\lambda}^{k+1} \tag{7.16}$$

where $\mathcal{S}_f(\boldsymbol{\lambda})$ is associated to an appropriate semi-discrete fluid solver $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda})$. The initialization $\boldsymbol{\lambda}^0$ of the iterative process is treated in Section 7.5. Let us explain this equation: given a value for the interface displacement $\boldsymbol{\lambda}^k$, we solve the fluid problem for this $\boldsymbol{\lambda}^k$ using $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda}^k)$ and recover the normal stresses on the interface $\boldsymbol{\sigma}_{\boldsymbol{n}}^f$, that is to say, we compute $\mathcal{S}_f(\boldsymbol{\lambda}^k)$. Then, we calculate the structure problem with $\boldsymbol{\sigma}_{\boldsymbol{n}}^s = \boldsymbol{\sigma}_{\boldsymbol{n}}^f$ as boundary condition on the fluid-structure interface. It gives a new value of the interface displacement that now we call $\boldsymbol{\lambda}^{k+1}$. In this case we solve the *Neumann-to-Dirichlet* Poincaré-Steklov interface operator $-\mathcal{S}_s^{-1}(\boldsymbol{\sigma}_{\boldsymbol{n}}^f)$. This procedure is repeated until convergence.

Remark 7.2. The solution of the fluid problem $\mathsf{FL}_{\delta t}(\boldsymbol{\lambda})$ requires nonlinear iterations. Thus, algorithm (7.16) involves the use of nested iterative loops.

We are also interested on a linearized version of $\mathcal{S}_f$. We denote by $\mathsf{FL}_{\delta t}(\boldsymbol{u}_*^{n+1}; \gamma)$ the linearized fluid operator that differs from the non-linearized version, i.e. (7.10), in the fact that the convective term in the momentum equation of the fluid has been replaced

by $\boldsymbol{u}_*^{n+1} \cdot \nabla \boldsymbol{u}^{n+1}$ with $\boldsymbol{u}_*^{n+1}$ given. We also denote by $\widetilde{\mathcal{S}}_f(\boldsymbol{u}_*^{n+1})$ the linearization of $\mathcal{S}_f$ *around the point* $\boldsymbol{u}_*^{n+1}$, that is, involving the solution of the linearized fluid problem with $\mathsf{FL}_{\delta t}(\boldsymbol{u}_*^{n+1}; \gamma)$. In the next section we suggest the use of the *semi*-linear interface operator in some cases. We stress the fact that $\widetilde{\mathcal{S}}_f(\boldsymbol{u}_*^{n+1})$ is non linear due to the *shape derivative*.

A different version of the fixed point algorithm (7.16) is obtained when using the *semi-linearized* version of the interface operator $\mathcal{S}_f$ for the fluid. In this case the fixed point algorithm reads as follows: given $\boldsymbol{\lambda}^k$ and $\boldsymbol{u}^{n+1,k}$ with $k > 0$, compute $\boldsymbol{\lambda}^{k+1}$ by

$$-\mathcal{S}_s^{-1}(\widetilde{\mathcal{S}}_f(\boldsymbol{u}^{n+1,k}; \boldsymbol{\lambda}^k)) = \boldsymbol{\lambda}^{k+1}. \tag{7.17}$$

and obtain $\boldsymbol{u}^{n+1,k+1}$ from $\mathsf{FL}_{\delta t}(\boldsymbol{u}^{n+1,k}; \boldsymbol{\lambda}^k)$. The procedure is repeated until a selected norm of $\boldsymbol{u}^{n+1,k+1} - \boldsymbol{u}^{n+1,k}$ and (or) $\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k$ is below a threshold tolerance.

Remark 7.3. When using the algorithm (7.17) the same loop deals with the coupling of the fluid and structure systems and the nonlinearity of the fluid equations.

Remark 7.4. The *semi-linearized* fixed point algorithm (7.17) involves the domain update at each iteration. This situation can be relaxed by using some criteria over $(\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k)$ in order to decide to update or to freeze the domain at the current iteration (that is to say, to neglect or not the shape derivative). Alternatively, instead of freezing the domain, we can use a transpiration method (cheaper than the movement of the domain), as suggested in [62] in order to accelerate the iterative process.

Alternative forms of the interface equation (7.14) motivate different iterative algorithms for the coupling. For instance, if we rewrite (7.14) as

$$-\mathcal{S}_s^{-1}(\mathcal{S}_f(\boldsymbol{\lambda})) - \boldsymbol{\lambda} = \mathbf{0}, \tag{7.18}$$

it motivates the use of a root finding technique. The use of the Newton algorithm in order to obtain the root of (7.18) has been explored in recent works (see e.g. [72]). It involves the computation of the tangent operators of $\mathcal{S}_f$ and $\mathcal{S}_s$. Again, these tangent operators account for the non-linearity of the fluid equations and the shape derivative. Its computation is an involved task. Approximate Jacobians invoking different approximations lead to a variety of Quasi-Newton methods (see [79, 127, 128]).

Deparis *et al.* in [61] use the approach adopted herein in order to motivate new algorithms. These algorithms, widely used as DD methods, are applied to the fluid-structure problem. They consider the preconditioned Richardson method to solve (7.14): given $\boldsymbol{\lambda}^k$, for $k > 0$, find $\boldsymbol{\lambda}^{k+1}$ such that

$$\mathcal{P}_k(\boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k) = -\mathcal{S}_f(\boldsymbol{\lambda}^k) - \mathcal{S}_s(\boldsymbol{\lambda}^k) \tag{7.19}$$

where $\mathcal{P}_k$ is a preconditioner of the Jacobian of $\mathcal{S}_f(\boldsymbol{\lambda}^k) + \mathcal{S}_s(\boldsymbol{\lambda}^k)$. Some alternative choices of $\mathcal{P}_k$ are suggested in [61].

Besides the iterative algorithm for the coupling, a relaxation method is advisable in order to improve the convergence properties of all the previous algorithms. The impact of the relaxation parameter on the convergence of the iterative algorithm for a simple test case has been analyzed in [37]. The Aitken acceleration method is the most widely used. The value of the optimal relaxation parameter for the Aitken technique has a known value for scalar equations. Different alternatives for the extension to the vector case have been proposed in [113, 60].

7.4 The discrete problem

This section is devoted to the fully discretized version of the coupling problem. We are focused on the discretization of the fluid. Three different sorts of methods are considered: monolithic, pressure correction and predictor corrector. Every method is introduced and stated. In the applications we consider the stabilized versions of these schemes using orthogonal subgrid scales. However, for the sake of clarity, we omit the stabilization terms in the formulation. We refer the reader to a set of articles that deal with stabilized pressure segregation methods [47, 56, 57, 52, 51]. The use of a stabilized space discretization allows us to use the same low-order finite element space for the interpolation of velocity and pressure. After the exposition of the alternative methods for the fluid problem, we state the discrete extension operator used for the obtention of the fluid domain movement. Finally, we suggest some coupling procedures taking into account the fluid solver used. These procedures are stated for being used in the next chapter.

7.4.1 The discrete fluid problem

The fully discretized version of the monolithic scheme (7.10), denoted by $\mathsf{MN}_{\delta t,h}(\boldsymbol{\lambda}_h)$ reads as follows: for $n = 0, 1, 2, ...$, given $\boldsymbol{\lambda}_h \in \Gamma_h(\Sigma_{t^{n+1}})$ (understood as the displacement on the solid boundary at the time step n), find $\boldsymbol{u}_h^{n+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ and $p_h^{n+1} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$ such that,

$$\frac{\rho_f}{\delta t}\left(D_k \boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h\right) + \mu\left(\nabla \boldsymbol{u}_h^{n+1}, \nabla \boldsymbol{v}_h\right) + \rho_f\left((\boldsymbol{u}_h^{n+1} - \boldsymbol{w}^{n+1}) \cdot \nabla \boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h\right) \tag{7.20a}$$

$$-(p_h^{n+1}, \nabla \cdot \boldsymbol{v}_h) = \rho_f \langle \boldsymbol{f}_f, \boldsymbol{v}_h \rangle \qquad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}), \tag{7.20b}$$

$$\left(\nabla \cdot \boldsymbol{u}_h^{n+1}, q_h\right) = 0 \qquad \forall q_h \in \mathcal{Q}_h(\Omega^f_{t^{n+1}}), \tag{7.20c}$$

$$\boldsymbol{u}_h^{n+1} = \frac{1}{\delta t \gamma_p}\left(\boldsymbol{\lambda}_h + \sum_{i=1}^{p} \alpha_p^i \boldsymbol{d}_h^{n+1-i}\right) \qquad \text{on } \Sigma_{t^{n+1}}, \tag{7.20d}$$

where $\Gamma_h(\Sigma_{t^{n+1}})$, $\mathcal{V}_h(\Omega^f_{t^{n+1}})$ and $\mathcal{Q}_h(\Omega^f_{t^{n+1}})$ are finite element approximation spaces of the functional spaces $\Gamma(\Sigma_{t^{n+1}})$, $\mathcal{V}(\Omega^f_{t^{n+1}})$ and $\mathcal{Q}(\Omega^f_{t^{n+1}})$ respectively. We refer to Section 1.3 for the introduction of the notation and concepts needed for the space discretization. Let Θ_h^t be a finite element partition of the domain Ω_t^f in a family of elements $\{K_e\}_{e=1}^{n_{el}}$, n_{el} being the number of elements. We also introduce the finite element partition Ξ_h^t of the interface Σ_t, which is completely defined by Θ_h^t. For simplicity, we consider that the finite element partitions of the fluid and solid meshes match on Σ_t, or alternatively, the structure is considered a rigid-body (as in the next chapter).

The finite element spaces introduced before and that we will use in the following are:

$$\begin{aligned}
\mathcal{V}_h(\Omega_0^f) &= \{\widehat{\boldsymbol{v}}_h \in \mathcal{C}^0(\Omega_0^f) \mid \widehat{\boldsymbol{v}}_h|_K = \tilde{\boldsymbol{v}} \circ F_K^{-1},\ \tilde{\boldsymbol{v}} \in R_k(\tilde{K}),\ K \in \Theta_h^t\},\\
\mathcal{Q}_h(\Omega_0^f) &= \{\widehat{q}_h \in \mathcal{C}^0(\Omega_0^f) \mid \widehat{q}_h|_K = \tilde{q} \circ F_K^{-1},\ \tilde{q} \in R_k(\tilde{K}),\ K \in \Theta_h^t\},\\
\Gamma_h(\Sigma_0) &= \{\widehat{\gamma}_h \in \mathcal{C}^0(\Sigma_0) \mid \widehat{\gamma}_h|_K = \tilde{\gamma} \circ F_K^{-1},\ \tilde{\gamma} \in R_k(\tilde{K}),\ K \in \Xi_h^t\},\\
\mathcal{V}_h(\Omega^f_{t^{n+1}}) &= \{\boldsymbol{v}_h \in \mathcal{C}^0(\Omega^f_{t^{n+1}}) \mid \boldsymbol{v}_h = \widehat{\boldsymbol{v}}_h \circ \mathcal{A}_t^{-1},\ \widehat{\boldsymbol{v}}_h \in \mathcal{V}_h(\Omega_0^f)\},\\
\mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}) &= \left\{\boldsymbol{v}_h \in \mathcal{V}_h(\Omega^f_{t^{n+1}}) | \, \boldsymbol{v}_h|_{\Sigma_{t^{n+1}}} = \boldsymbol{0}\right\},\\
\mathcal{Q}_h(\Omega^f_{t^{n+1}}) &= \{q_h \in \mathcal{C}^0(\Omega^f_{t^{n+1}}) \mid q_h = \widehat{q}_h \circ \mathcal{A}_t^{-1},\ \widehat{q}_h \in \mathcal{Q}_h(\Omega_0^f)\},\\
\Gamma_h(\Sigma_t) &= \{\gamma_h \in \mathcal{C}^0(\Sigma_t) \mid \gamma_h = \widehat{\gamma}_h \circ \mathcal{A}_t^{-1},\ \widehat{\gamma}_h \in \Gamma_h(\Sigma_0)\}.
\end{aligned}$$

Again, we consider the linearized version of $\mathsf{MN}_{\delta t,h}(\boldsymbol{\lambda}_h)$, around $\boldsymbol{u}_{*,h}^{n+1}$, denoted by $\mathsf{MN}_{\delta t,h}(\boldsymbol{u}_{*,h}^{n+1};\boldsymbol{\lambda}_h)$. $\mathsf{MN}_{\delta t,h}(\boldsymbol{\lambda}_h)$ implies the computation of velocities and pressure together. A substantial reduction of the computational cost is obtained when using a splitting technique. These techniques allow the uncoupling of velocity and pressure computation. Herein, we consider a pressure correction method obtained at the discrete level (see Chapter 3) . We denote by $\mathsf{FS}_{\delta t,h}(\boldsymbol{\lambda}_h)$ the following problem: given $\boldsymbol{\lambda}_h \in \Gamma_h(\Sigma_{t^{n+1}})$, find $\boldsymbol{u}_h^{n+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ and $p_h^{n+1} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$ from the following scheme:

1. Find $\hat{\boldsymbol{u}}_h^{n+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ such that

$$\begin{aligned}\frac{\rho_f}{\gamma_p \delta t}\left(\hat{\boldsymbol{u}}_h^{n+1} - \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{u}_h^{n-i}, \boldsymbol{v}_h\right) + \mu\left(\nabla \hat{\boldsymbol{u}}_h^{n+1}, \nabla \boldsymbol{v}_h\right) \\ + \rho_f\left((\hat{\boldsymbol{u}}_h^{n+1} - \boldsymbol{w}^{n+1}) \cdot \nabla \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h\right) \\ - (\widetilde{p}_h^{n+1}, \nabla \cdot \boldsymbol{v}_h) = \rho_f \langle \boldsymbol{f}_f^{n+1}, \boldsymbol{v}_h\rangle \quad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}) \end{aligned} \tag{7.21a}$$

$$\hat{\boldsymbol{u}}_h^{n+1} = \frac{1}{\delta t \gamma_p}\left(\boldsymbol{\lambda}_h - \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{d}_h^{n-i}\right) \qquad \text{on } \Sigma_{t^{n+1}}. \tag{7.21b}$$

2. Find $p_h^{n+1} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$ such that

$$\begin{aligned} -\gamma_p \delta t\left(\Pi_h\left(\nabla p_h^{n+1} - \nabla \widetilde{p}_h^{n+1}\right), \nabla q_h\right) \\ = \rho_f\left(\hat{\boldsymbol{u}}_h^{n+1}, \nabla q_h\right) \quad \forall q_h \in \mathcal{Q}_h(\Omega^f_{t^{n+1}}) \end{aligned} \tag{7.22}$$

3. Find $\boldsymbol{u}_h^{n+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ such that

$$\begin{aligned}\frac{\rho_f}{\delta t \gamma_p}\left(\boldsymbol{u}_h^{n+1} - \hat{\boldsymbol{u}}_h^{n+1}, \boldsymbol{v}_h\right) \\ - (p_h^{n+1} - \widetilde{p}_h^{n+1}, \nabla \cdot \boldsymbol{v}_h) = \boldsymbol{0} \quad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}) \end{aligned} \tag{7.23a}$$

$$\boldsymbol{u}_h^{n+1} = \frac{1}{\delta t \gamma_p}\left(\boldsymbol{\lambda}_h + \sum_{i=0}^{p-1} \alpha_p^i \boldsymbol{d}_h^{n-i}\right) \qquad \text{on } \Sigma_{t^{n+1}}. \tag{7.23b}$$

In step 2, $\widetilde{p}_h^{n+1}$ is *an appropriate approximation to* p_h^{n+1} and Π_h is the L^2 projection onto the velocity space. We consider an *incremental* fractional step method when $\widetilde{p}_h^{n+1} = p_h^n$. This method has an splitting error of order $\mathcal{O}(\delta t^2)$. The results are much better than for *total* projection methods, where $\widetilde{p}_h^{n+1} = 0$, without extra computational cost. Equation (7.22) of the second step of the method can be approximated by the pressure Poisson equation (see Eq. (3.16)): find $p_h^{n+1} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$ such that

$$-\gamma_p \delta t\left(\nabla p_h^{n+1} - \nabla \widetilde{p}_h^{n+1}, \nabla q_h\right) = \rho_f\left(\hat{\boldsymbol{u}}_h^{n+1}, \nabla q_h\right) \qquad \forall q_h \in \mathcal{Q}_h(\Omega^f_{t^{n+1}}). \tag{7.24}$$

Remark 7.5. This approximation introduces the same artificial boundary condition that we find when we do the splitting at the continuous level (see [157]), that is, $\partial p^{n+1}/\partial n = 0$ on the Dirichlet boundary of the velocity. This misbehavior is of special interest for fluid-structure interaction problems, due to the fact that the fluid-structure interface is a Dirichlet boundary. Thus, an artificial boundary condition over the pressure is imposed on the surface where the pressure is integrated for the calculation of the stresses exerted by the fluid. We defend the use of (7.22) if we want to avoid the artificial boundary conditions.

The system matrix associated to (7.22) is cumbersome, but can be tackled when using an iterative solver, case in which only matrix-vector products are needed. Furthermore, the use of a closed integration rule for approximating the Gramm (mass) matrix that appears in (7.22) reduces considerably the computational cost. Nevertheless, it would be interesting to assess the impact of the artificial boundary condition on fluid-structure problems.

For the pressure correction method we only consider the fixed point iteration algorithm using nested loops, as justified below.

When we use an iterative implicit procedure for the coupling the fluid problem is evaluated (at least) as many times as coupling iterations. Thus, it is natural to put in the momentum equation $\widetilde{p}_h^{n+1} = p_h^{n+1,k}$, $p_h^{n+1,k}$ being the pressure obtained at the previous iteration. In fact, if the resulting scheme converges, *the intermediate velocity* $\boldsymbol{u}_h^{n+1}$ *converges to the end-of-step velocity* $\hat{\boldsymbol{u}}_h^{n+1}$. Furthermore, $\boldsymbol{u}_h^{n+1}$ *converges to the solution of the monolithic fluid system.* Thus we do not need to distinguish between $\hat{\boldsymbol{u}}_h^{n+1}$ and $\boldsymbol{u}_h^{n+1}$ and (7.23) can be ignored. The final system to be solved at every coupling iteration is the following: given $\boldsymbol{\lambda}_h^k \in \Gamma_h(\Sigma_{t^{n+1}})$ and $p_h^{n+1,k} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$, find $\boldsymbol{u}_h^{n+1,k+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ and $p_h^{n+1,k+1} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$ such that,

$$\begin{aligned}
&\frac{\rho_f}{\delta t}\left(D_k\boldsymbol{u}_h^{n+1,k+1},\boldsymbol{v}_h\right)+\mu\left(\nabla\boldsymbol{u}_h^{n+1,k+1},\nabla\boldsymbol{v}_h\right)\\
&\qquad+\rho_f\left((\boldsymbol{u}_h^{n+1,k+1}-\boldsymbol{w}^{n+1})\cdot\nabla\boldsymbol{u}_h^{n+1,k+1},\boldsymbol{v}_h\right)\\
&\qquad\qquad-(p_h^{n+1,k},\nabla\cdot\boldsymbol{v}_h)=\rho_f\langle\boldsymbol{f}_s,\boldsymbol{v}_h\rangle\quad\forall\boldsymbol{v}_h\in\mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}),\qquad(7.25a)\\
&-\gamma_p\delta t\left(\Pi_h\left(\nabla p_h^{n+1,k+1}-\nabla p_h^{n+1,k}\right),\nabla q_h\right)\\
&\qquad\qquad=\rho_f\left(\boldsymbol{u}_h^{n+1,k+1},\nabla q_h\right)\quad\forall q_h\in\mathcal{Q}_h(\Omega^f_{t^{n+1}}),\qquad(7.25b)\\
&\boldsymbol{u}_h^{n+1,k+1}=\frac{1}{\delta t\gamma_p}\left(\boldsymbol{\lambda}_h^k+\sum_{i=0}^{p-1}\alpha_p^i\boldsymbol{d}_h^{n-i}\right)\qquad\text{on }\Sigma_{t^{n+1}}.\qquad(7.25c)
\end{aligned}$$

This problem is denoted by $\mathsf{PC}_{\delta t,h}(p_h^{n+1,k};\boldsymbol{\lambda}_h^k)$. We remark that in the case presented nested loops are needed: an internal loop to deal with the nonlinearity of the convective term and an external for the convergence to the monolithic fluid system (for fluid problems) or the *monolithic* coupling system (for fluid-structure problems). Again, there is the possibility to use one loop for everything. In this case, the final system is: given $\boldsymbol{\lambda}_h^k \in \Gamma_h(\Sigma_{t^{n+1}})$, $\boldsymbol{u}_h^{n+1,k} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ and $p_h^{n+1,k} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$, find $\boldsymbol{u}_h^{n+1,k+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ and $p_h^{n+1,k+1} \in \mathcal{Q}_h(\Omega^f_{t^{n+1}})$ such that,

$$\begin{aligned}
&\rho_f\left(D_k\boldsymbol{u}_h^{n+1,k+1},\boldsymbol{v}_h\right)+\mu\left(\nabla\boldsymbol{u}_h^{n+1,k+1},\nabla\boldsymbol{v}_h\right)\\
&\qquad+\rho_f\left((\boldsymbol{u}_h^{n+1,k}-\boldsymbol{w}^{n+1})\cdot\nabla\boldsymbol{u}_h^{n+1,k+1},\boldsymbol{v}_h\right)\\
&\qquad\qquad-(p_h^{n+1,k},\nabla\cdot\boldsymbol{v}_h)=\rho_f\langle\boldsymbol{f}_s,\boldsymbol{v}_h\rangle\quad\forall\boldsymbol{v}_h\in\mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}),\qquad(7.26a)\\
&-\gamma_p\delta t\left(\Pi_h\left(\nabla p_h^{n+1,k+1}-\nabla p_h^{n+1,k}\right),\nabla q_h\right)\\
&\qquad\qquad=\rho_f\left(\boldsymbol{u}_h^{n+1,k+1},\nabla q_h\right)\quad\forall q_h\in\mathcal{Q}_h(\Omega^f_{t^{n+1}}),\qquad(7.26b)\\
&\boldsymbol{u}_h^{n+1,k+1}=\frac{1}{\delta t\gamma_p}\left(\boldsymbol{\lambda}_h^k+\sum_{i=0}^{p-1}\alpha_p^i\boldsymbol{d}_h^{n-i}\right)\qquad\text{on }\Sigma_{t^{n+1}}.\qquad(7.26c)
\end{aligned}$$

In this case the fluid solver is denoted by $\mathsf{PC}_{\delta t,h}(\boldsymbol{u}_h^{n+1,k}, p_h^{n+1,k}; \boldsymbol{\lambda}^k)$.

Methods (7.25) and (7.26) are *predictor-corrector* schemes (see Section 3.2.5) . These methods have been introduced in [56, 57] without the fluid-structure motivation. In these references the stabilization terms omitted in the present exposition are carefully treated.

Remark 7.6. Along this section we have considered $\boldsymbol{w}^{n+1}$ independent of the iterative process for the sake of clarity. However, this is not the general case. How to treat this mesh velocity in the iterative algorithm has been pointed out in Remark 7.4.

7.4.2 The discrete fluid domain movement

As commented in the previous section, we use a harmonic extension operator on Ω_t^f in order to obtain $\boldsymbol{d}_h^f$. The discrete problem reads as follows: given $\boldsymbol{\lambda}_h^k \in \Gamma_h(\Sigma_{t^{n+1}})$, find $(\boldsymbol{d}_h^f)^{n+1} \in \mathcal{V}_h(\Omega^f_{t^{n+1}})$ such that

$$\left(\nabla(\boldsymbol{d}_h^f)^{n+1}, \nabla \boldsymbol{v}_h\right) = \boldsymbol{0} \qquad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,0}(\Omega^f_{t^{n+1}}), \tag{7.27a}$$

$$(\boldsymbol{d}_h^f)^{n+1} = \boldsymbol{\lambda}_h, \tag{7.27b}$$

We call $(\boldsymbol{d}_h^f)^{n+1} = \mathrm{Ext}_h(\boldsymbol{\lambda}_h)$. The harmonic operator is applied on Ω_t^f because it allows to solve this problem using the same mesh that we use to compute the fluid problem.

7.4.3 Coupling algorithms for the discrete problem

In this section we propose three different coupling procedures for the pressure segregation methods listed in Section 7.3, exploiting their properties. We only consider the fixed point algorithms (7.16) and (7.17) for the coupling, but these ideas can be easily extended to methods motivated from (7.18) and (7.19). Let us start with the pressure correction method (7.26). As commented above the use of this method will be restricted to cases where an explicit procedure is used for the coupling. In this case the resulting iterative algorithm is: given $\widetilde{\boldsymbol{\lambda}_h}^{n+1}$, find $\boldsymbol{\lambda}_h^{n+1}$ such that

$$\boldsymbol{\lambda}_h^{n+1} = -\mathcal{S}_s^{-1}(\mathcal{S}_f(\widetilde{\boldsymbol{\lambda}}_h^{n+1})) \tag{7.28}$$

and $(\boldsymbol{u}_h^{n+1}, p_h^{n+1}) = \mathsf{FS}_{\delta t,h}(\widetilde{\boldsymbol{\lambda}}_h^{n+1})$. Here, $\widetilde{\boldsymbol{\lambda}}_h^{n+1}$ is an appropriate approximation of $\boldsymbol{\lambda}_h^{n+1}$. Different alternatives have been suggested in the literature. A first order approximation in time is $\widetilde{\boldsymbol{\lambda}_h}^{n+1} = \boldsymbol{\lambda}_h^n$. A more accurate second order approximation that reduces the artificial energy introduced to the system is proposed in [135]. However, numerical instabilities occur much earlier with the second order predictor (see the numerical experimentation in [129]). In this work we have adopted as initial condition

$$\widetilde{\boldsymbol{\lambda}}^{n+1} = -\mathcal{S}_s^{-1}((\boldsymbol{\sigma}_{\boldsymbol{n}}^f)^n), \tag{7.29}$$

that is, we solve the structure problem at t^{n+1} using as Neumann boundary condition the normal stress $(\boldsymbol{\sigma}_{\boldsymbol{n}}^f)^n$ exerted by the fluid at the previous time step. A second order method ofthis type is

$$\widetilde{\boldsymbol{\lambda}}^{n+1} = -\mathcal{S}_s^{-1}(2(\boldsymbol{\sigma}_{\boldsymbol{n}}^f)^n - (\boldsymbol{\sigma}_{\boldsymbol{n}}^f)^{n-1}). \tag{7.30}$$

A stability analysis of an aeroelastic test case (similar to the one in [135]) using (7.29) and (7.30) together with explicit procedures has been developed in [141] with good results.

When using implicit procedures for the coupling we have claimed predictor corrector schemes are superior. As commented above there are some possibilities for the iterative process. Let us start with the one loop algorithm. For every coupling iteration $k \geq 0$, the problem to be solved is: given $\boldsymbol{\lambda}_h^{n+1,k}$, $\boldsymbol{u}_h^{n+1,k}$ and $p_h^{n+1,k}$, find $\boldsymbol{\lambda}_h^{n+1,k+1}$ such that

$$\boldsymbol{\lambda}_h^{n+1,k+1} = -\mathcal{S}_s^{-1}(\widetilde{\mathcal{S}_f}(\boldsymbol{u}_h^{n+1,k}; \boldsymbol{\lambda}_h^{n+1,k})) \tag{7.31}$$

with $(\boldsymbol{u}_h^{n+1,k+1}, p_h^{n+1,k+1}) = \mathsf{PC}_{\delta t,h}(\boldsymbol{u}_h^{n+1,k}, p_h^{n+1,k}; \boldsymbol{\lambda}_h^{n+1,k})$. Thus, in the implicit coupling process, we have to solve (7.31) until convergence. In this method the same loop deals with the non-linearity of the convective term and the convergence to the monolithic system. Some other alternatives for the treatment of the iterations are possible. For instance, the use of nested loops, one for the coupling and one for the non-linearity. This case is similar to (7.31) but using $\mathcal{S}_f(\boldsymbol{\lambda}_h^{n+1,k})$ together with the fluid solver (7.25). Further, a third algorithm could be used. At every coupling iteration k we could iterate over the predictor-corrector method until convergence to the monolithic fluid system. However, for simplicity, we only use (7.28) and (7.31) in the numerical experimentation. Alternative versions of (7.31) can be tested for every application in order to identify which is faster.

Remark 7.7. In the algorithm (7.28) associated to the pressure correction method, we use the Steklov-Poincaré operator $\mathcal{S}_f(\cdot)$ that involves nonlinear iterations due to the convective term. However, for the predictor corrector coupling algorithm (7.31) the semi-linearized version $\mathcal{S}_f(\boldsymbol{u}_h^{n+1,k}; \cdot)$ that does not involve nonlinear iterations, has been used.

7.5 Conclusions

As explained above, the appropriate algorithm for the solution of the coupled system depends on the kind of problem to be solved. In this chapter we have in mind aeroelastic problems. Let us draw some features about this sort of applications:

- The fluid solver consumes much more CPU time than the structure. By this reason, the number of fluid evaluations has to be minimized in order to optimize the computational cost.

- The convergence of the coupling iterative process is *easy*. As explained in Section 7.1, this behavior is associated to the fact that the structure density is much larger than the fluid density.

The use of Newton and Quasi-Newton methods, or the use of preconditioners together with a Richardson iterative process are justified in some cases (for instance haemodynamics) when the convergence of fixed point algorithms is very slow. However, for aeroelastic problems the convergence rate of the last method is good. That, together with the fact that the fixed point algorithm minimizes the number of fluid evaluations per iteration, has motivated its choice for the applications of Chapter 8.

Besides, the bottle neck of the coupling method is the fluid solver. We can use explicit and implicit fluid solvers. The first class of solvers is cheaper but the time step allowed is restricted to be smaller than a critical time step size δt_{cr}. Using an implicit procedure the cost per evaluation is more expensive but δt can be larger. It is commonly accepted that to capture well the physics of the flow, a time step size δt of the order of $10 - 100\delta t_{cr}$ should be used.

In order to reduce the computational cost associated to the fluid solver we suggest the use of fractional step methods. Pressure correction methods (7.21)-(7.23) and predictor corrector methods (7.25) and (7.26) are considered.

The pressure correction scheme (in its implicit, semi-implicit or explicit version) is a good choice when using explicit procedures for the coupling. The coupling problem to be solved in this case is the one defined in (7.28). This method introduces a splitting error (due to the splitting of the Navier-Stokes equations) and a coupling error (due to the explicit procedure for the coupling). Furthermore, the coupling error affects the stability of the coupling procedure (see e.g. [129]).

When using implicit procedures, as explained above for the fully discrete problem, the use of a predictor corrector scheme is more appropriate, because we can profit from the coupling iterations in order for the fluid solver to tend to the monolithic system (decreasing the splitting error). The nice property of this method is that, when reaching convergence, the solution is the same as that obtained by the *monolithic* approach to the coupled problem. In (7.31) we have stated the algorithm using only a single loop for the fluid and the coupling.

Chapter 8

Bridge Aerodynamics

The coupling algorithms designed in the previous chapter for the fluid-structure interaction problem are applied to the aeroelastic simulation of suspension bridges. We assess flexural and torsional frequencies for a given inflow velocity. Moreover, increasing this velocity we reach the value for which the flutter phenomenon appears. This chapter is part of [6].

8.1 Introduction

Among the different topologies of bridges, *suspension* bridges span the greatest distances. However, the bending moments acting on the deck sections of this sort of bridges are relatively small. Even though the span between piles is very large, the distance between cables, that in fact are working as piles, is small. For this reason these structures are flexible and light.

These features make suspension bridges very influenced by wind actions. While for other topologies the aeroelastic behavior is not considered important, for suspension bridges it implies a key aspect of the design process.

The action of a fluid over a structure can induce three different phenomena:

- *Divergence*: It can be considered a *static* instability. It happens when the deformation induced by the fluid to the structure increases the fluid action until failure. It is similar to the *buckling* of a pile loaded by an axial force. This phenomenon is of little importance in bridge design because it happens for very high wind speeds.

- *Buffeting*: This dynamic phenomenon is associated to the effect of the fluctuations of the inflow over the structure. For suspended bridges, the inflow is usually very homogeneous. However, in some conditions it can induce low frequency vibrations. Even though these vibrations do not endanger the structure they can induce fatigue effects.

- *Flutter*: This dynamic phenomenon is induced by the fluid-structure coupling (the energy transfer). The flutter happens when the damping induced by the fluid to the structure makes the overall structure damping negative. Then, the oscillations of the bridge increase until failure. It happens for high velocities ($\sim 60 - 70 m/s$) of the inflow (but much smaller than the divergence velocity).

In the application described now we consider only the flutter phenomenon, which is the most important aeroelastic effect when designing suspension bridges. When this aeroelastic phenomenon was not taken into account by the engineers it caused some historical

failures of bridges. The Tacoma bridge is probably the best known case. Suspension bridges have a very low structural damping that make them sensitive to this effect. One of the most important criteria of design is the *flutter limit* velocity (when flutter occurs). An acceptable structure must have a large enough flutter limit velocity. A large gap between the maximum velocity of design and this limit is required.

The flutter analysis has been developed by experimentation in wind tunnels. For instance, the design of the Great Belt bridge (Denmark) involved more than 16 sections (see [121]). We point out that in wind tunnels the flutter limit is not obtained directly, that is, increasing the inflow velocity of the wind tunnel until failure. This flutter limit is obtained evaluating the *aeroelastic derivatives*. This methodology, that was originated in aeronautics, was extrapolated to bridge aerodynamics by Scanlan and Tomko in [143]. When using this methodology, the assessment of the effect of the fluid over the structure is made with an inflow velocity far from the flutter limit and prescribed deck motions. This experimentation process is very expensive and time consuming. Further, the Reynolds number of the real problem cannot be reproduced in conventional wind tunnels.

The increasing in the capability of computers together with the improvement of numerical methods have motivated in the last decade the use of computer methods for the analysis of bridge aerodynamics [144]. For the bridge analyzed herein, the Great Belt bridge, we refer now to some previous works. Jenssen and Kvamsdal in [115] analyze this bridge using the finite volume method and an explicit procedure for the coupling. The aeroelastic derivatives are computed to obtain the flutter limit. Selvam *et al.* in [144] use the finite element method in a moving (non-inertial) frame of reference for a direct simulation of the flutter. The more recent work of Frandsen [78] uses the finite element method and a monolithic approximation of the coupled fluid-structure problem. This reference includes a good review on this topic.

In our numerical simulation no turbulence modeling has been considered. Due to the fact that the bridge deck is a bluff body, the flow is detached and the influence of the turbulence effects for this case is less important than for the aeroelastic analysis of wings. However, for wide decks, the flow re-attaches at a given point. Nevertheless, we use a stabilized finite element method motivated by a multiscale approach. There is a recent trend among the computational mechanics community to claim that this kind of methods can replace conventional turbulence models (see [108, 109, 110, 21]).

The present application is devoted to the evaluation of flexural and torsional frequencies of the Great Belt bridge for a given inflow velocity and the direct flutter simulation using the methods introduced in the previous chapter. The finite element method together with stabilized predictor corrector and pressure correction fluid solvers for the coupling have been used. The ALE framework has allowed to forulate the flow problem in moving domains. First and second order accurate methods (in time) have been considered.

8.2 The bridge model

For the numerical aeroelastic analysis of bridges, the 3D problem is usually reduced to a 2D problem. In fact, this is also the usual procedure for wind tunnel tests. In order to simulate the correct natural frequencies in the fundamental symmetric flexural and torsional modes, spring stiffnesses are applied to the elastic center of the cross-section. Lumped mass and moment of inertia on the gravity center have been introduced to simulate the mass and moment of inertia per unit length. Furthermore, the 2D cross-section is considered a rigid body.

In order to obtain the equations governing the displacement of the bridge section, the

Newton's law is formulated on the gravity center, and the spring force depending on the displacement of the structure is applied to the elastic center. When the gravity center and the elastic center are in different positions, the resulting governing equations are nonlinear. However, assuming that the rotation angle is *small*, the equations can be easily linearized. Thus, the linearized ordinary differential equation (ODE) that governs the displacement of the structure reads as follow: find the displacement vector $\boldsymbol{d}_s \in \mathbb{R}^3$ (for a $2d$ problem) such that

$$\boldsymbol{M}\ddot{\boldsymbol{d}}_s + \boldsymbol{C}\dot{\boldsymbol{d}}_s + \boldsymbol{K}\boldsymbol{d}_s = \boldsymbol{f}, \tag{8.1}$$

where $\boldsymbol{M}$ is the *mass* matrix, $\boldsymbol{C}$ is the *damping* matrix, $\boldsymbol{K}$ is the *stiffness* matrix and $\boldsymbol{f}$ is the external force exerted over the structure. The displacement vector contains the translation and rotation of the structure, that is,

$$\boldsymbol{d}_s = \begin{pmatrix} d_x \\ d_y \\ d_\theta \end{pmatrix}.$$

where d_x and d_y are the displacements along the x and y directions, respectively, and d_θ is the rotation. The *linearized mass* matrix has the following expression:

$$\boldsymbol{M} = \begin{pmatrix} m & 0 & -s_x \\ 0 & m & s_y \\ -s_x & s_y & I_\theta \end{pmatrix}$$

where m, I_θ, s_x and s_y denote the mass, inertial moment and static moments associated to the elastic center (per unit length), respectively. The stiffness matrix is given by

$$\boldsymbol{K} = \begin{pmatrix} k_x & 0 & 0 \\ 0 & k_y & 0 \\ 0 & 0 & k_\theta \end{pmatrix}$$

where k_x, k_y and k_θ are the corresponding stiffness coefficients. The damping coefficients of each degree of freedom define the damping matrix, which is taken as

$$\boldsymbol{C} = \begin{pmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & c_\theta \end{pmatrix}.$$

The damping coefficients are usually given as a *percentage logarithmic decrement*. A $l\%$ logarithmic decrement implies a damping coefficient

$$c = \frac{l}{\pi}\sqrt{mk},$$

m and k being the mass and stiffness coefficients, respectively.

The external force vector (including force and moment) over the bridge exerted by the fluid at a given time is

$$\boldsymbol{f} = \begin{pmatrix} f_x \\ f_y \\ m_\theta \end{pmatrix}$$

and can be obtained as

$$[f_x, f_y]^T = \int_{\Sigma_t} \boldsymbol{\sigma}_{\boldsymbol{n}}^f \, \mathrm{d}\Sigma$$
$$m_\theta = \int_{\Sigma_t} \boldsymbol{\sigma}_{\boldsymbol{n}}^f \times \boldsymbol{r} \, \mathrm{d}\Sigma.$$

where, as before, $\boldsymbol{\sigma}_{\boldsymbol{n}}^f$ is the normal stress on Σ_t exerted by the fluid, and $\boldsymbol{r}$ is the position vector (being the frame of reference centered on the elastic center). We point out that this external force depends on the displacement of the structure, that is, $\boldsymbol{f} = \boldsymbol{f}(\boldsymbol{d}_s)$, and thus the problem is nonlinear.

For the time integration of the ODE (8.2) we use the unconditional stable *constant-average-acceleration* scheme, also called *trapezoidal rule*, which is described by the following set of equations:

$$\begin{cases} \boldsymbol{M}\ddot{\boldsymbol{d}}_s^{\,n+1} + \boldsymbol{C}\dot{\boldsymbol{d}}_s^{\,n+1} + \boldsymbol{K}\boldsymbol{d}_s^{n+1} = \boldsymbol{f}^{n+1}, \\ \boldsymbol{d}_s^{n+1} = \boldsymbol{d}_s^n + \delta t \dot{\boldsymbol{d}}_s^{\,n} + \frac{\delta t^2}{4}\left(\ddot{\boldsymbol{d}}_s^{\,n+1} + \ddot{\boldsymbol{d}}_s^{\,n}\right), \\ \dot{\boldsymbol{d}}_s^{\,n+1} = \dot{\boldsymbol{d}}_s^{\,n} + \frac{\delta t}{2}\left(\ddot{\boldsymbol{d}}_s^{\,n+1} + \ddot{\boldsymbol{d}}_s^{\,n}\right). \end{cases}$$

This second order accurate scheme is particularly appropriate for the case under consideration due to the fact that preserves the energy of the structure, given by

$$E_s = \frac{1}{2}\dot{\boldsymbol{d}}_s \cdot \boldsymbol{M}\dot{\boldsymbol{d}}_s + \frac{1}{2}\boldsymbol{d}_s \cdot \boldsymbol{K}\boldsymbol{d}_s, \tag{8.2}$$

which is an important feature when analyzing the aeroelastic stability of the structure.

8.3 The coupling model

In this section we describe the fluid solver on moving domains and the coupling procedure that will be used for the direct analysis of flutter.

The coupling procedure that we use herein for the simulation of this phenomenon is implicit. As it is widely known, explicit procedures introduce artificial energy to the system that can lead to undesirable numerical instability (see [135] and [129]). Due to the fact that we want to assess the stability of the coupling problem, intimately related to the energy transfer between fluid and structure, it is justified the use of an implicit procedure that avoids this artificial energy. Further, the implicit procedure tends to the solution of the monolithic coupled system, eliminating the splitting error associated to staggered procedures.

Due to the complexity of external flows that appear in aeroelastic applications, and its highly transient behavior, the use of second order methods are worth it, and even more when no extra computational cost is introduced. We have used here the BDF2 scheme, both for the time integration of the momentum equation and for the evaluation of the mesh velocity in the fluid domain. By doing that, and as it is proved in Chapter 6 for the convection-diffusion equation, the ALE formulation does not spoil the second order of accuracy of the fluid solver. The movement of the fluid domain has been computed by solving the discrete problem (7.27).

The formulation for the fluid problem that will be used is a stabilized pressure segregation method. More specifically, a predictor corrector method is considered because of the fact that we use an implicit procedure, as justified in the previous chapter.

We point out that, when the structure is considered a rigid body, the interface equation (7.15) has the following *integrated* form,

$$\mathcal{S}_s^{-1}\left(\int_{\Sigma_t}\mathcal{S}_f(\boldsymbol{\lambda})\,\mathrm{d}\Sigma\right)=\boldsymbol{\lambda}. \tag{8.3}$$

where $\boldsymbol{\lambda}=\boldsymbol{\lambda}(\boldsymbol{d}_s)$ and $\mathcal{S}_s(\boldsymbol{\lambda})$ gives the forces and moments (not the stresses) that cause a displacement $\boldsymbol{\lambda}$. Likewise, $\mathcal{S}_f(\boldsymbol{\lambda})$ contains not only the components of the normal stress exerted by the fluid, but also the moments per unit of area (length if $d=2$), and therefore the integral of $\mathcal{S}_f(\boldsymbol{\lambda})$ gives the total force and moment exerted by the fluid on Σ_t. In this case we use a fixed point iterative method to solve the nonlinear interface problem (8.3). More precisely, the method used here is the integral version of the iteration scheme stated in (7.31).

Even though this kind of problems have a good convergence, we have used the Aitken acceleration technique for scalar equations. We define the residual of the interface equation as,

$$\boldsymbol{r}(\boldsymbol{\lambda}^k)=\mathcal{S}_s^{-1}\left(\int_{\Sigma_t}\mathcal{S}_f(\boldsymbol{\lambda}^k)\,\mathrm{d}\Sigma\right)-\boldsymbol{\lambda}^k.$$

Exploiting the fact that the structure is considered as a rigid body, the relaxation parameter can be obtained from the expression for scalar equations. In this case, we consider the *diagonal relaxation matrix*,

$$\boldsymbol{\omega}^k=\begin{pmatrix}\omega_x^k & 0 & 0\\ 0 & \omega_y^k & 0\\ 0 & 0 & \omega_\theta^k\end{pmatrix},$$

that verifies

$$\boldsymbol{\omega}^k(\boldsymbol{r}(\boldsymbol{\lambda}^k)-\boldsymbol{r}(\boldsymbol{\lambda}^{k-1}))=\boldsymbol{\lambda}^k-\boldsymbol{\lambda}^{k-1}.$$

The *relaxed* version of a fixed point iteration applied to (8.3) is

$$\boldsymbol{\lambda}^{k+1}=-\boldsymbol{\omega}^k\mathcal{S}_s^{-1}\left(\int_{\Sigma_t}\mathcal{S}_f(\boldsymbol{\lambda}^k)\,\mathrm{d}\Sigma\right)+(Id-\boldsymbol{\omega}^k)\boldsymbol{\lambda}^k.$$

A deep study of relaxation methods in a fluid-structure framework can be found in [60].

8.4 Assessment of frequencies and direct flutter simulation

This section is devoted to the numerical simulation of the flutter limit and the assessment of frequencies of the Great Belt bridge (Denmark). The parameters that define the problem have been summarized in Table 8.1 and have been extracted from [144]. The problem domain and its finite element discretization is shown in Figure 8.4. We have used an unstructured mesh of 48453 linear triangles for this simulation. A time step size of 0.01 s has been considered. The horizontal movement is restricted, as it is usually assumed. We do not know which are the appropriate elastic coefficients when analyzing the real sized problem with the real inflow velocity. By this reason we have assumed the elastic coefficients used for the dimensionless analyzed by Selvam *et al.* in [144]. It has to be

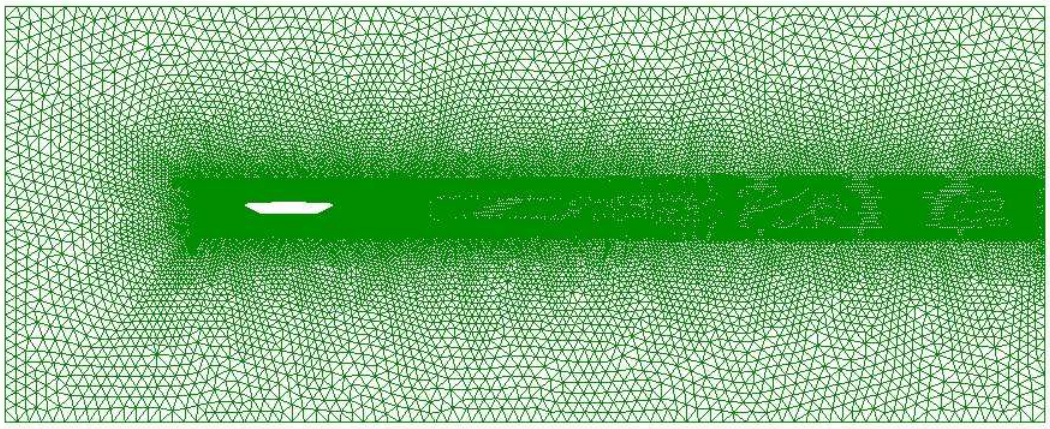


Figure 8.1: Space domain of analysis and mesh used for the simulation

Mass per unit length, m $[Kg/m]$	2.27×10^3
Vertical static moment on elastic centre per unit length, s_y $[Kg \cdot m/m]$	1.61×10^4
Mass moment of inertia on elastic centre per unit length, I_θ $[Kg \cdot m^2/m]$	2.47×10^6
Vertical spring stiffness, k_y $[N/m^2]$	8.78×10^3
Torsional spring stiffness, k_θ $[N \cdot m/m^2]$	7.21×10^6
Vertical logarithmic damping, l_y [%]	1
Torsional logarithmic damping, l_θ [%]	0.6

Table 8.1: Properties of the Great Belt Bridge

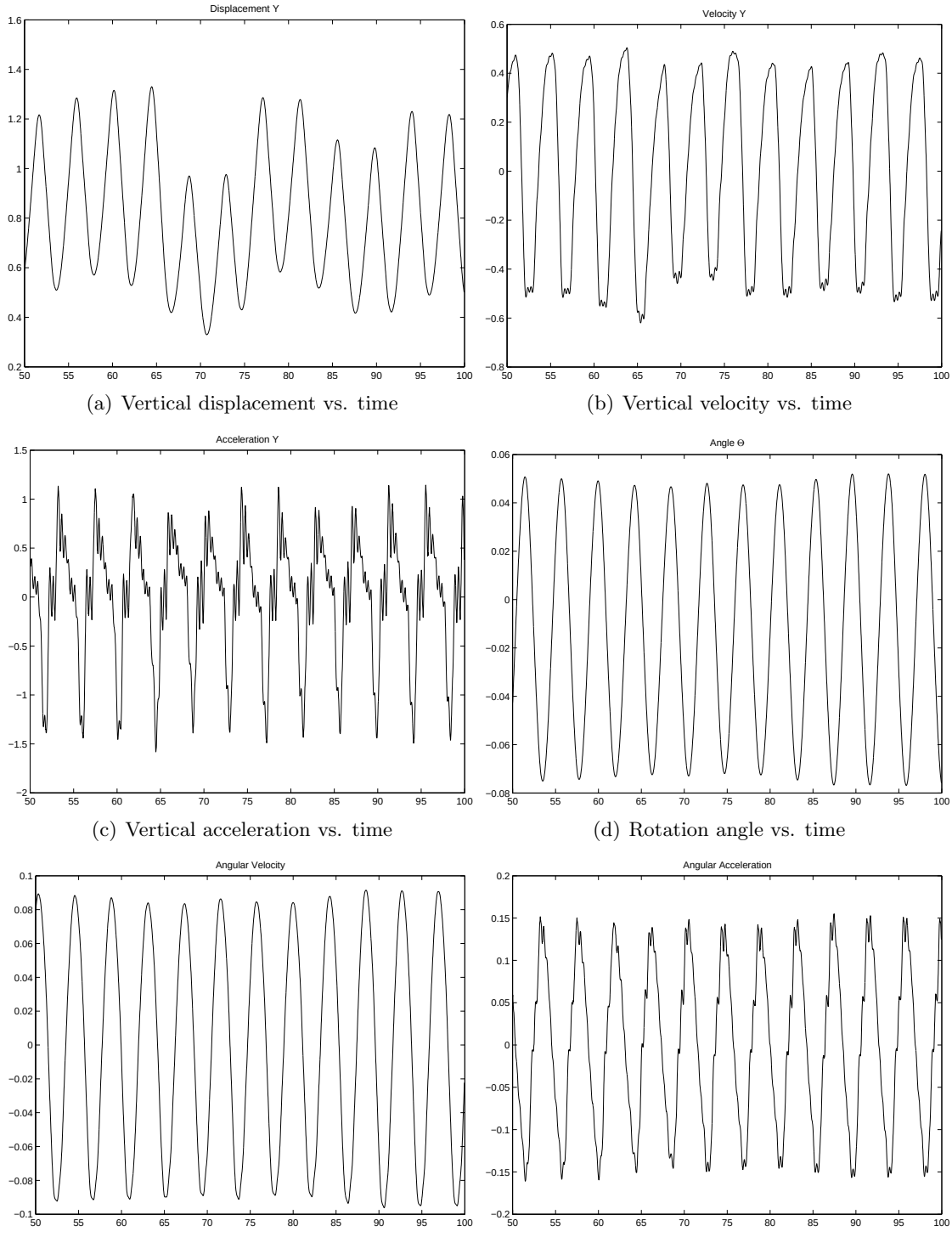
taken into account that this assumption affects the obtained results and complicates the comparison to wind tunnel experiments.

Firstly, given an inflow velocity of $\boldsymbol{u}_{in} = (50, 0)m/s$, we obtain the temporary response of the bridge. In figures 8.2(a), 8.2(b) and 8.2(c) we show the vertical displacement, velocity and acceleration. Figures 8.2(d), 8.2(e) and 8.2(f) show the rotation angle, angular velocity and angular acceleration. We plot the results after some time of computation. In figure 8.3 we plot the energy of the structure, defined in (8.2). These plots prove the stability of the structure.

Using a *Fourier Fast Transform* we have obtained the frequencies associated to the vertical displacement (flexural frequency) and rotation angle (torsional frequency). We show these results in Figures 8.4(a) and 8.4(b). In both cases a clear dominant frequency governs the movement.

We show contours of the velocity norm and pressure at different time steps in Figures 8.5 and 8.6, respectively.

The average number of iterations needed for the convergence of the integral version of method (7.31) to the monolithic system for a given time step is around *4 iterations per time step* for an inflow velocity of $50m/s$.



(a) Vertical displacement vs. time

(b) Vertical velocity vs. time

(c) Vertical acceleration vs. time

(d) Rotation angle vs. time

(e) Angular velocity vs. time

(f) Angular acceleration vs. time

Figure 8.2: Movement of the bridge for inflow velocity $\boldsymbol{u}_{in} = (50, 0)m/s$

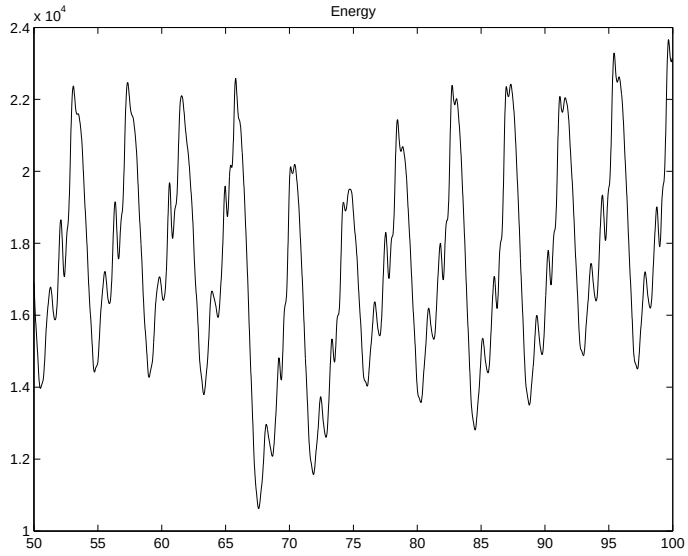
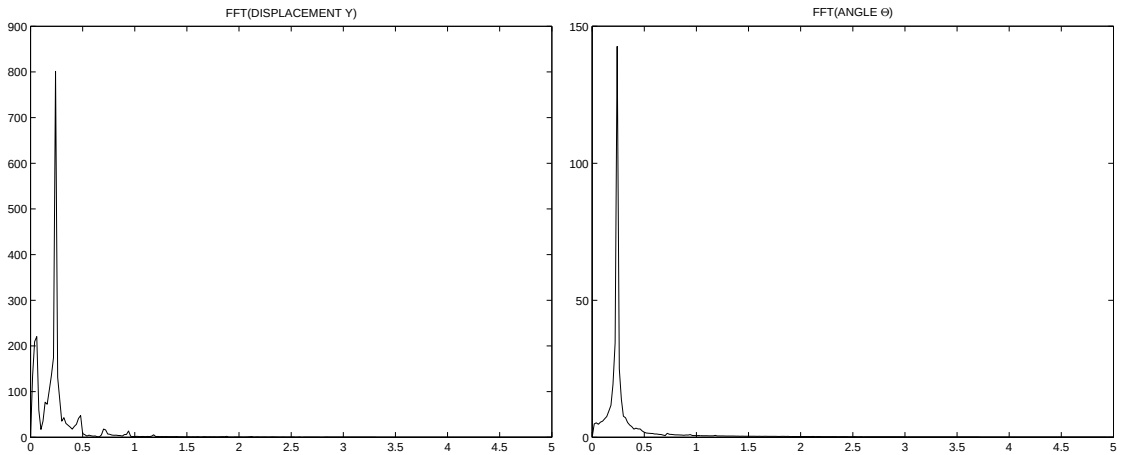


Figure 8.3: Bridge energy vs. time for inflow velocity $\boldsymbol{u}_{in} = (50, 0)\ m/s$

In a second step, we increase the inflow velocity until we reach the aeroelastic instability. The flutter phenomenon appears for an inflow velocity of $55m/s$. We plot the same values as before in figures 8.7- 8.8. We easily see in this case that the flutter instability appears for this velocity. In fact, the instability is translational and torsional (see Figures 8.7(a) and 8.7(d)). We plot velocities and accelerations for vertical displacement and rotation angle in Figures 8.7(b)- 8.7(c) and 8.7(e)- 8.7(f). The aerodynamic instability is clearly shown from the increase of the structure energy (Figure 8.8).

Obviously, the number of iterations needed for the inflow velocity of $55m/s$ increases with the structure energy. We end this section with the plots of the velocity norm and pressure at different time steps in Figures 8.9 and 8.10, respectively.



(a) Vertical displacement: amplitude vs. frequency (b) Rotation angle: amplitude vs. frequency

Figure 8.4: Fourier transform of vertical displacement and rotation angle of the bridge for inflow velocity $\boldsymbol{u}_{in} = (50, 0) m/s$

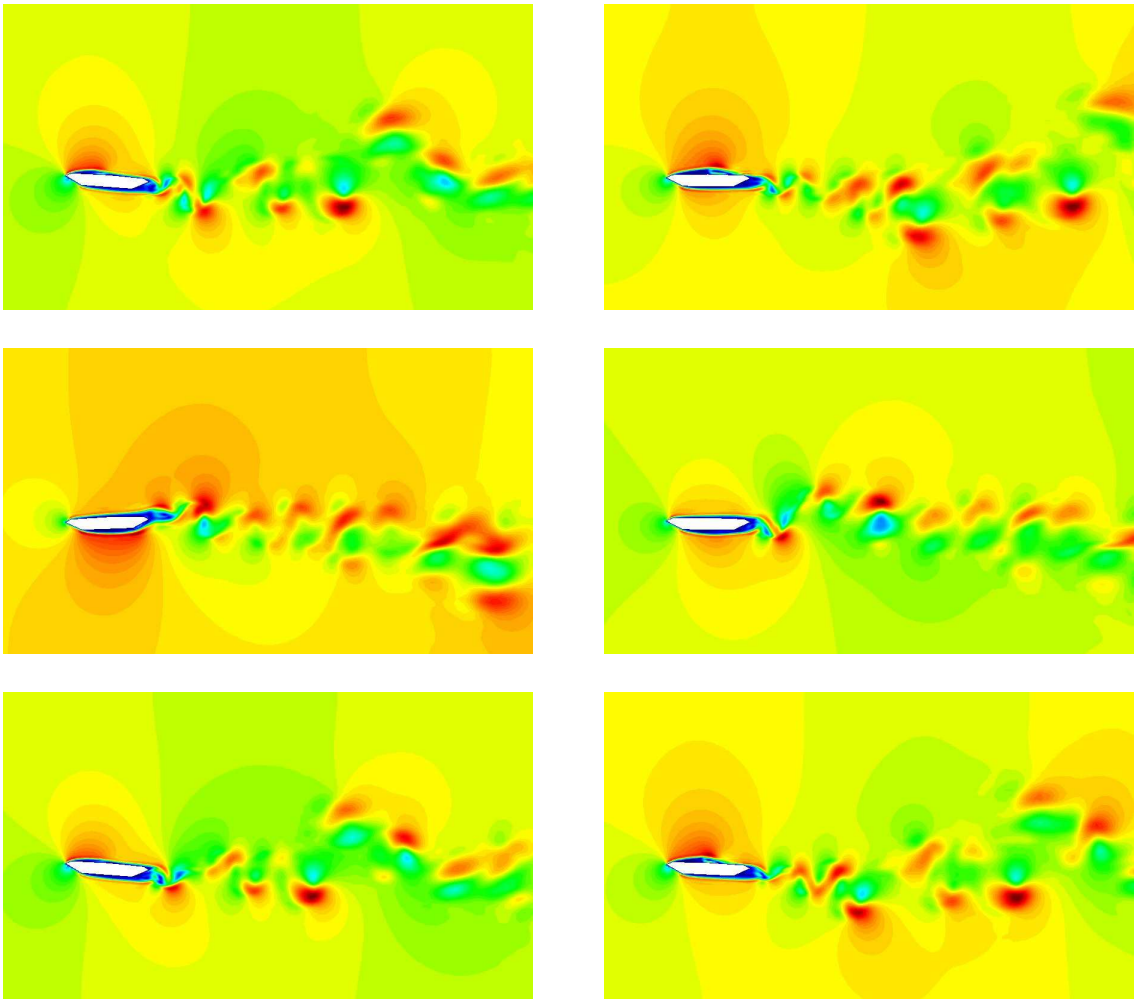


Figure 8.5: Contours of the velocity norm at different time steps (increasing time from left to right and from top to bottom) for inflow velocity $\boldsymbol{u}_{in} = (50, 0)m/s$

8.5 Aeroelastic derivatives using numerical experimentation

A different approach to the direct flutter simulation is the calculation of the aeroelastic derivatives. This is the usual procedure when using wind tunnel tests. We refer to [143] for an introduction to this methodology.

In [141] Rossi has used the coupling method (7.28) for the assessment of aeroelastic derivatives. In fact, our fluid software has been used for the assessment of the flutter derivatives. This method involves the use of a pressure correction method together with an explicit coupling procedure. The bridge model is identical to the one presented herein.

In this case, the results obtained are in agreement with the wind tunnel experiments. The key difference, compared to the direct flutter simulation presented here, is the fact that in the reference mentioned the problem considered is the same as in the wind tunnel, and thus the results can be fairly compared.

8.6 Conclusions

The coupling methods proposed herein for the calculation of the flutter limit have shown an excellent behavior in aeroelastic applications. The *coupling method proposed converges*

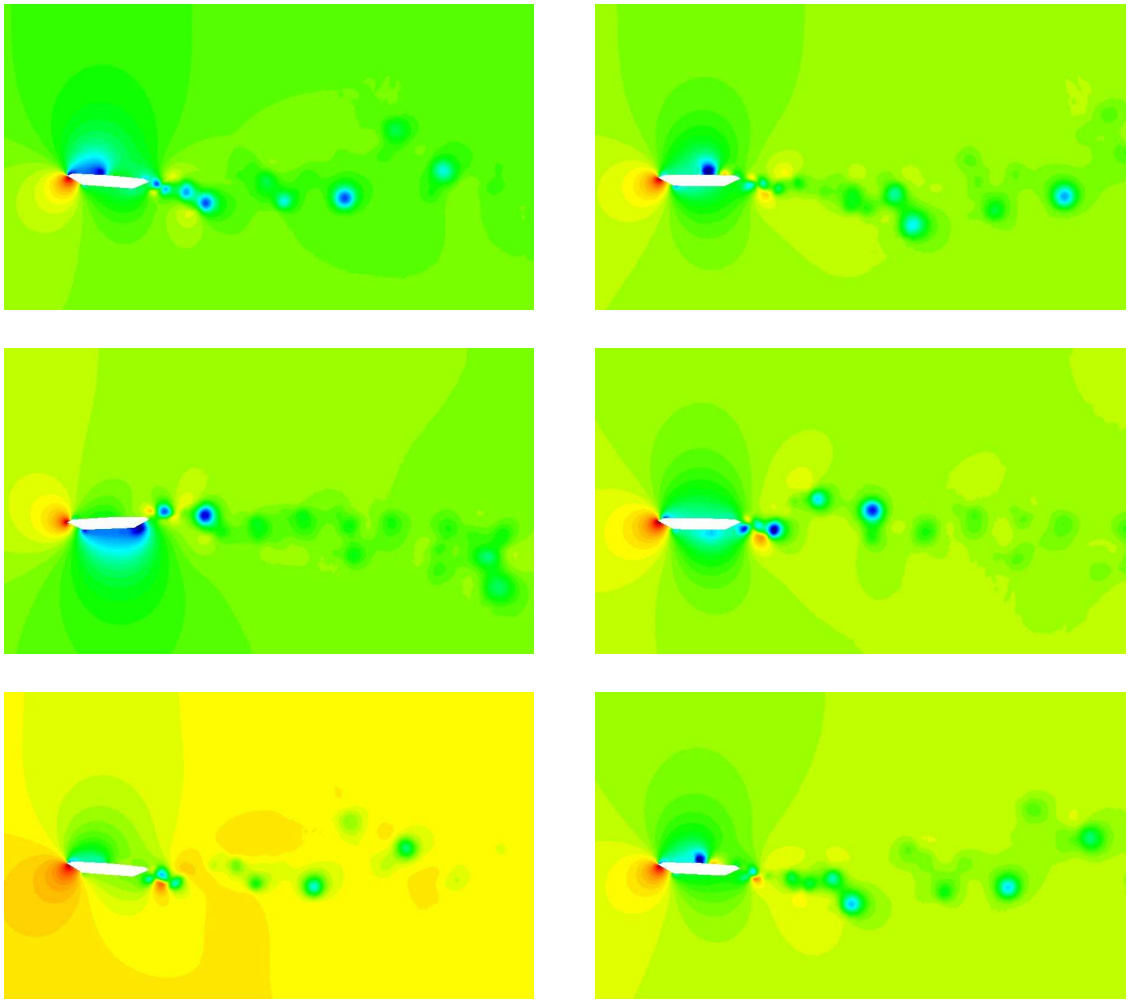


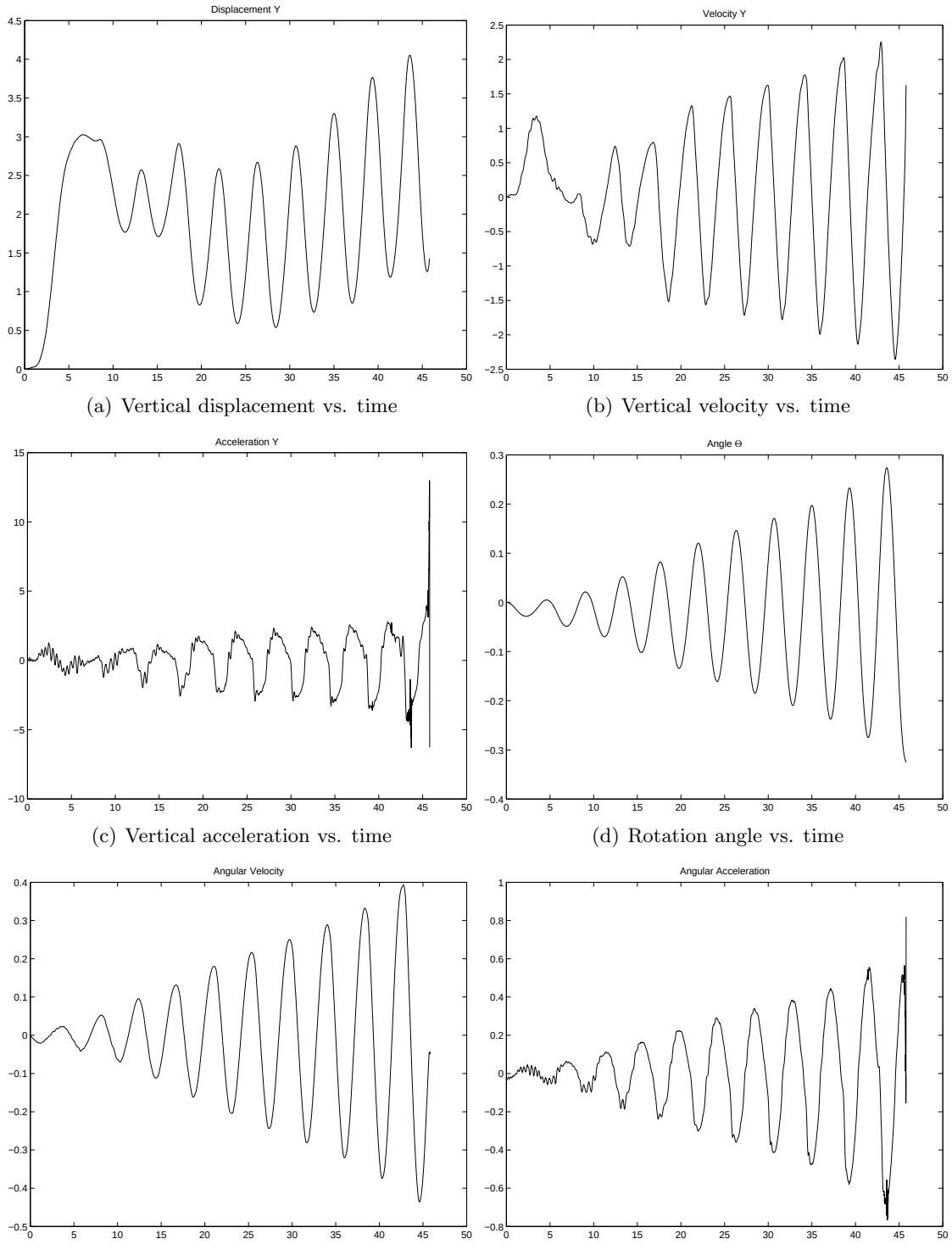
Figure 8.6: Contours of the pressure at different time steps (increasing time from left to right and from top to bottom) for inflow velocity $\boldsymbol{u}_{in} = (50, 0)m/s$

to the monolithic problem (for the predictor-corrector solver). That is, the coupling process does not introduced any extra error (apart from tolerance stop criteria). Furthermore, this method shows a good convergence behavior for this kind of problems.

The other key point is the fact that *the present methods uncouple the velocity and pressure computation*, that implies a high reduction of the computational cost of the fluid problem, the bottle neck of aeroelastic simulations.

Summarizing, the key feature of the formulation we propose are the use of second order stabilized pressure segregation methods (both pressure correction and predictor corrector versions) together with a second order ALE formulation, a second order structure solver and a coupling iterative procedure that tends to the monolithic system. Thus, *the overall fluid-structure coupling procedure* proposed herein *is second order accurate in time*, an important property for highly transient external flows that appear in aeroelastic applications.

We have applied this methods to the aeroelastic analysis of a bridge deck. The flutter velocity of 55 m/s obtained herein differs from the 65-70 m/s obtained from the aeroelastic derivatives assessed with wind tunnel tests. However, this gap could be expected, since the problems solved are different. It seems that the elastic coefficient that should be used for the direct analysis of flutter in dimensional form has to be higher than the one used



(a) Vertical displacement vs. time

(b) Vertical velocity vs. time

(c) Vertical acceleration vs. time

(d) Rotation angle vs. time

(e) Angular velocity vs. time

(f) Angular acceleration vs. time

Figure 8.7: Movement of the bridge for inflow velocity $\boldsymbol{u}_{in} = (55, 0)m/s$

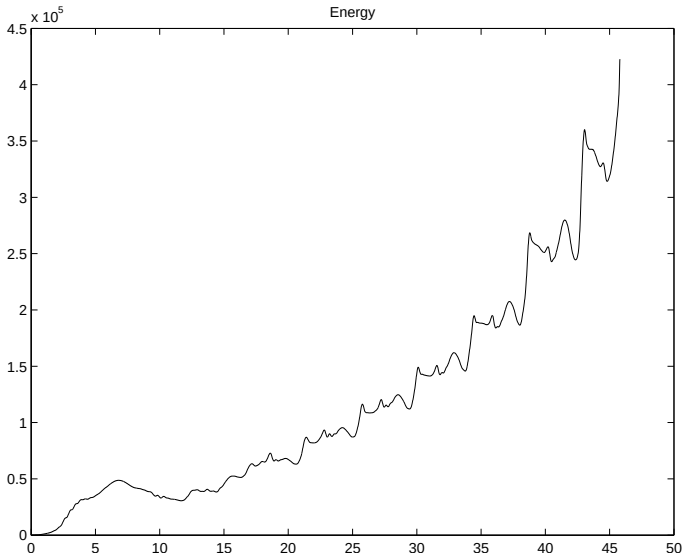


Figure 8.8: Bridge energy vs. time for inflow velocity $\boldsymbol{u}_{in} = (55, 0)m/s$

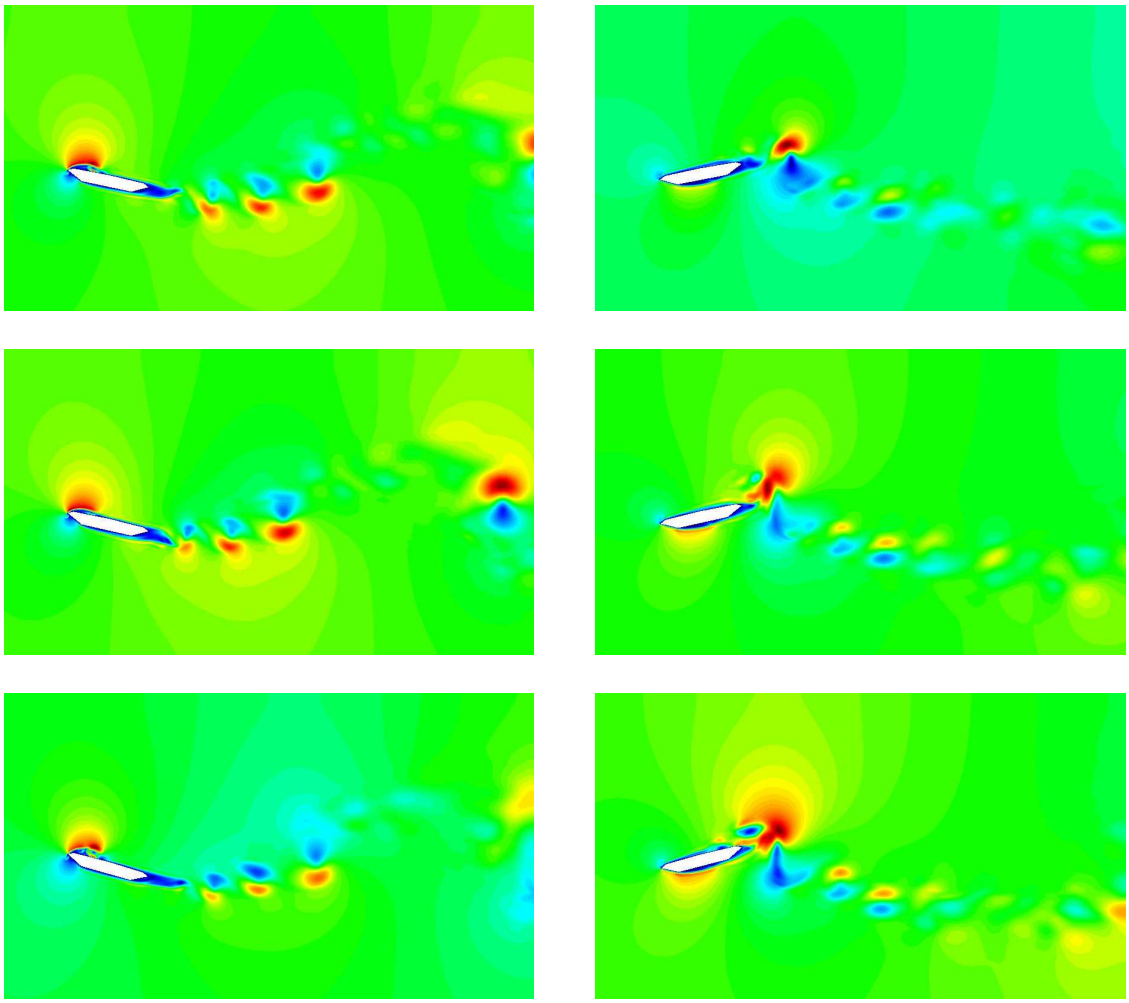


Figure 8.9: Contours of the velocity norm at different time steps (increasing time from left to right and top to bottom) for inflow velocity $\boldsymbol{u}_{in} = (55, 0)m/s$

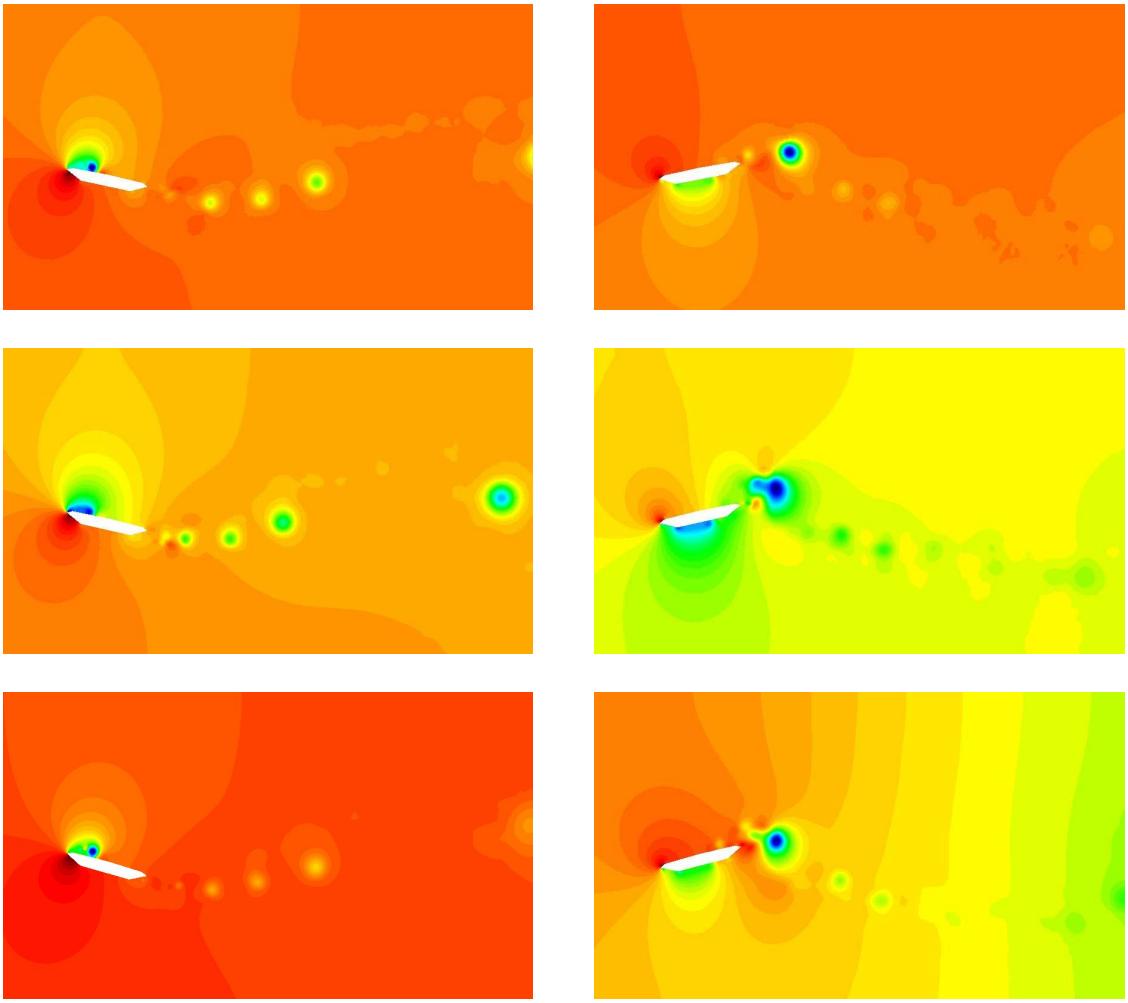


Figure 8.10: Contours of the pressure at different time steps (increasing time from left to right and top to bottom) for inflow velocity $\boldsymbol{u}_{in} = (55, 0)m/s$

for the scaled problem.

In fact, numerical experiments using the same method (even the same software) are in exceptional agreement with the wind tunnel results when assessing the aeroelastic derivatives, as reported in [141]. This is even more relevant considering that in this reference an explicit coupling procedure has been used together with a pressure correction method, thus introducing a splitting error and artificial energy.

Chapter 9

Wind Turbines

This chapter is devoted to the discussion of different numerical methods that allow the simulation of problems involving fixed and rotary bodies that modify the problem domain. Two different strategies are considered: a geometrical domain decomposition approach and a selective remeshing technique. The pros and cons of these two choices are listed, having in mind the fact that pressure segregation methods are being used. After that, using a particular remeshing procedure, we assess aeroelastic forces exerted by the wind over a wind turbine.

9.1 Introduction

The production of renewable energies must increase in the near future due to the commitment to the Kyoto Protocol that demands to reduce the CO_2 emissions. For this reason, renewable energy sources are appealing an increasing number of companies.

Wind turbines are the most popular renewable energy devices. They are playing a key role in the generation of electrical power in USA, and specially Europe. For this reason, the successful development of large, flexible and highly optimized wind turbines is one of the aspects that will dictate the future of wind energy.

The aeroelastic behavior of wind turbines is very complicated, but it is mandatory a deep understanding of the aeroelasic response for the subsequent optimization. Thus, the development of appropriate numerical methods that can handle well the simulation of the aeroelastic behavior of wind turbines is a current challenge of special interest.

The simulation of wind turbines involves a fluid solver and a structure solver. Herein we are focussed on the fluid simulation, assuming the different parts of the wind turbine rigid bodies. However, the bottle neck of this simulation is the fluid solver, that consumes most of the CPU time. For this reason the use of pressure segregation methods are obviously justified, since these methods reduce the computational cost of the fluid problem.

The consideration of the whole wind turbine complicates the numerical simulation. There are parts of the device that are fixed (piles and generator) together with rotary parts (blades). Further, we could be interested on the positioning of a set of wind turbines, or even the effect of the topology of the terrain. It implies that the problem domain is being deformed in time. Moreover, the rotation of the blades makes the ALE approach by itself (used in the previous chapter for bridge aerodynamics) inappropriate, only allowing small rotation angles without remeshing.

In this chapter we introduce two different algorithms that can handle well the domain deformation without a great impact on the CPU time: a *geometrical* domain decomposi-

tion (DD onwards) approach and a selective *geometrical* remeshing strategy.

The domain decomposition approach is called *geometrical* to stress the fact that the algorithm is introduced in order to handle the combination of fixed and rotary bodies, clearly different from DD methods related to computational efficiency (parallelization). We split the domain into two parts (with or without overlapping). One is fixed and the other is moving fixed to the rotary body. We discuss the problems linked to the transmission conditions when using pressure segregation methods. We address some important aspects to take into account when choosing the DD methods to be used. Further, we suggest how to improve the imposition of Neumann and Dirichlet boundary conditions.

Again, we call the second approach *geometrical* remeshing to remark that the procedure considered herein is not introduced, for example, to refine the mesh under some *a posteriori* error estimate. The remeshing procedure is a time consuming task, that not only implies the generation of the mesh but also the projection of some values from the old mesh (or meshes) to the new mesh (introducing a projection error), the re-allocation of arrays of unknowns and the system matrix and reconstruction of the mesh graph, etc... By this reason, it is highly advisable to restrict the remeshing to a small part of the domain, keeping the rest of the mesh untouched. With this aim we split the domain in three different non-overlapped parts: the fixed domain, the moving domain glued to the rotating components, and the *transmission* domain between the fixed and moving domains. The remeshing procedure is only carried out on the transmission domain. Further, as initial attempt, we have decided to use a specific procedure for the remeshing of the transmission domain, suggested by Behr and Tezduyar when applying their *DSD/SST* method to the simulation of rotating components in [10, 11]. Another key aspect of the remeshing approach is the fact that we account for the movement of the rotary domain using an special ALE framework.

Finally, we have assessed the forces exerted by the wind over a real wind turbine. This simulation has been carried out mixing the following ingredients: a second order (in time) stabilized pressure correction scheme, a selective remeshing strategy using the SSL algorithm and an *exact* ALE formulation. We assume that the wind turbine is working under a constant angular velocity. The coupling of the structure and the fluid is only considered in one direction. However, in future works, we would consider fully coupled situations applying the fluid-structure algorithms developed in Chapter 7.

This chapter is organized as follows. In Section 9.2 we introduce the geometrical DD approach considering that pressure correction methods are used. In Section 9.3 the selective remeshing procedure is discussed and the SSL reconnecting algorithm is commented. We end the chapter with the numerical simulation of a wind turbine, evaluating the aeroelastic forces.

9.2 A geometrical domain decomposition approach

Apart from the powerful application of domain decomposition methods for the distribution of the computational cost, these methods facilitate the management of complex geometries. *Geometrically* oriented domain decomposition methods can be found in the literature (for instance, see [101, 100]). This is the case under consideration in the present work.

One of the most widely used DD methods for simplifying the meshing is the *Chimera* method (see [13, 14, 96, 152, 101]). The Chimera method consists of the coupling of a background mesh that covers the whole domain together with grids (usually associated to moving objects) that are patched to the background mesh by means of transmission conditions on their boundaries. This method is appropriate for applications where there

are moving objects with large translational and rotational displacements that make the ALE approach insufficient.

When dealing with rotary components, more appropriate procedures can be considered. In this case the most effective approach is to split the domain into two parts. The moving mesh that is glued to the rotating component, that is to say, is rotating with the angular velocity of the rotating object, and the fixed mesh. These two meshes will have an overlapping or not depending on the kind of DD method used.

Let us stress some aspects to be considered when using pressure segregation methods. It is widely known that pressure correction methods motivated at the continuous level introduce artificial boundary conditions for the pressure, $\partial p/\partial \boldsymbol{n} = 0$ on the (velocity) Dirichlet boundary. This misbehavior can be repared when using methods obtained at the discrete level, as we do in Chapter 3. In this case, as we point out in Remark 3.1, using $\mathsf{DM}^{-1}\mathsf{G}$ (see Eq. (3.15b)) as system matrix, we do not introduce any artificial boundary condition over the pressure. Even though this system matrix increases the computational cost of the pressure Poisson equation, it can be justified in the case under consideration, because the interface boundary between subdomains is usually an *interest zone*.

If Neumann transmission conditions are imposed, the situation is easier. Let us comment how to deal with Neumann boundary conditions when using pressure segregation methods. Recalling the setting used in Chapter 3, we can write the discrete monolithic system (using BDF1 for simplicity) as

$$\mathsf{M}\frac{1}{\delta t}D_1\mathsf{U}^{n+1} + \mathsf{K}(\mathsf{U}^{n+1})\mathsf{U}^{n+1} + \mathsf{G}\mathsf{P}^{n+1} = \mathsf{F}^{n+1}, \tag{9.1a}$$

$$\mathsf{D}\mathsf{U}^{n+1} = \mathsf{0}, \tag{9.1b}$$

where the only modification with respect to Chapter 3 is the force vector F, that now incorporates the Neumann boundary conditions,

$$\mathsf{F}_i^a = \langle N^a, f_i\rangle + \langle N^a, \boldsymbol{t}_i^{n+1}\rangle_{\Gamma_N},$$

being $\boldsymbol{t}_i^{n+1}$ the normal stress on the Neumann boundary Γ_N. The splitting of the momentum equation leads again to

$$\mathsf{M}\frac{1}{\delta t}(\tilde{\mathsf{U}}^{n+1} - \mathsf{U}^n) + \mathsf{K}(\tilde{\mathsf{U}}^{n+1})\tilde{\mathsf{U}}^{n+1} + \gamma\mathsf{G}\mathsf{P}^n = \mathsf{F}^{n+1}, \tag{9.2a}$$

$$\delta t\mathsf{D}\mathsf{M}^{-1}\mathsf{G}(\mathsf{P}^{n+1} - \gamma\mathsf{P}^n) = \mathsf{D}\tilde{\mathsf{U}}^{n+1}, \tag{9.2b}$$

$$\mathsf{M}\frac{1}{\delta t}(\mathsf{U}^{n+1} - \tilde{\mathsf{U}}^{n+1}) + \mathsf{G}(\mathsf{P}^{n+1} - \gamma\mathsf{P}^n) = \mathsf{0}, \tag{9.2c}$$

or, if we want to consider a further approximation (see Eq. (3.16)), we replace (9.2b) by

$$\delta t\mathsf{L}(\mathsf{P}^{n+1} - \gamma\mathsf{P}^n) = \mathsf{D}\tilde{\mathsf{U}}^{n+1}, \tag{9.3}$$

that introduces artificial Neumann conditions over the pressure. Equations (9.2b) and (9.3) need an extra pressure Dirichlet condition on the (velocity) Neumann boundary in order for the pressure correction system to be well-posed. From the Neumann boundary condition a pressure Dirichlet boundary condition can be easily obtained. We know at the continuous level that

$$\boldsymbol{\sigma} = -p\boldsymbol{I} + 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u})$$

being ρ the fluid density and p the pressure field. Thus, at the (velocity) Neumann boundary, we know that

$$p^n = -\mathbf{t}^n \cdot \boldsymbol{n} + 2\mu \boldsymbol{n} \cdot \varepsilon(\boldsymbol{u}^n) \cdot \boldsymbol{n}. \tag{9.4}$$

This is the expression obtained when writing the viscous term as $-2\nabla \cdot (\nu\varepsilon(\boldsymbol{u}^n))$. The Laplacian expression of the viscous term $-\nu\Delta\boldsymbol{u}^n$ modifies the Neumann boundary conditions, yielding

$$p^n = -\mathbf{t}^n \cdot \boldsymbol{n} + \mu \boldsymbol{n} \cdot \nabla \boldsymbol{u}^n \cdot \boldsymbol{n}.$$

Boundary condition (9.4) is not appropriate at the discrete level, because the normal vector and velocity derivatives are not defined at the nodes. The appropriate way to understand (9.4) is to consider this condition in a weak sense,

$$(p^n, \gamma)_{\Gamma_N} = -\left(\boldsymbol{t}^n \cdot \boldsymbol{n}, \gamma\right)_{\Gamma_N} + 2\mu \left(\boldsymbol{n} \cdot \varepsilon(\boldsymbol{u}^n) \cdot \boldsymbol{n}, \gamma\right)_{\Gamma_N} \tag{9.5}$$

for all γ belonging to the finite element partition of the $(d-1)$-dimensional manifold Γ_N (being d the space dimension).

Summarizing, the DD approach obligates the imposition of boundary conditions in an *interest zone*. The imposition of boundary conditions for pressure correction methods is complicated and can be source of errors. We can easily infer that Neumann boundary conditions are prefered to Dirichlet boundary conditions.

The exploration of DD methods that would only involve Neumann transmission conditions could be a nice choice (for instance, as in FETI-like methods, see [71, 123]). However, if we impose Dirichlet boundary conditions and we want to avoid artificial conditions over the pressure, the best choice (from our point of view) is to solve the subdomain where Dirichlet boundary conditions are being imposed with $\mathsf{DM}^{-1}\mathsf{G}$ (preferably, the subdomain with less number of nodes).

In the present work we have not used the DD approach for the numerical simulation of wind turbines. As commented below, the use of a selective remeshing strategy seems superior for this particular case. But in other situations this approach can be very useful. The study of the ideas commented above using numerical experimentation would be part of future works.

9.3 Selective remeshing and ALE

In this section we suggest the use of a selective *geometrical* remeshing strategy together with an ALE formulation that will allow us to solve the wind turbine problem.

The remeshing strategy is usually used in order to refine the mesh at the places where an *a posteriori* error estimate dictates. This procedure is expensive because involves the generation of a new mesh, the projection between old and new meshes (that induces a projection error), to store as old meshes as backward steps needed for the time integration (one for BDF1 and two for BDF2), to resize arrays of unknowns and the system matrix, to compute again the mesh graph, etc... Summarizing, the remeshing strategy is an involved and CPU time consuming process.

Again, we say *geometrical* to stress the fact that the introduction of this method is justified in order to simplify the treatment of the complex time dependent domain. Exploiting the particular features of our goal, we can simplify drastically the remeshing procedure.

Let us divide the domain Ω_t (where the subscript t denotes the time dependence) in three different parts: the *fixed domain* Ω^f, the *moving domain* Ω_t^m glued to the rotating object under consideration and the *transmission domain* Ω^{tr} between them. These domains satisfy,

$$\Omega^f \cup \Omega_t^m \cup \Omega^{tr} \equiv \Omega_t, \qquad \overline{\Omega^f} \cap \overline{\Omega_t^m} = \emptyset, \qquad \overline{\Omega^f} \cap \overline{\Omega^{tr}} = \emptyset, \qquad \overline{\Omega_t^m} \cap \overline{\Omega^{tr}} = \emptyset. \tag{9.6}$$

Moreover, we consider the following interfaces

$$\Omega^f \cap \Omega^{tr} \equiv \Sigma^{ft}, \qquad \Omega_t^m \cap \Omega^{tr} \equiv \Sigma^{mt} \tag{9.7}$$

called *fixed transmission interface* and *moving transmission interface*, respectively. We point out that, even though Σ^{mt} is the boundary of Ω_t^m, it is also boundary of Ω^{tr}, and thus, does not change in time.

Now, we consider the finite element partitions and finite element spaces associated to these domains. We refer to Section 1.3 for the ingredients needed for the space discretization. Let Θ_h^f be a finite element partition of the domain Ω^f in a family of elements $\{K_e\}_{e=1}^{n_{el}^f}$, n_{el}^f being the number of elements in this domain. We point out that Θ_h^f does not change in time. We define the following finite element spaces,

$$\begin{aligned}
\mathcal{Q}_h^f &= \{q_h \in \mathcal{C}^0(\Omega_0) \mid q_h|_K = \tilde{q} \circ F_K^{-1},\ \tilde{q} \in R_{k_\mathrm{p}}(\tilde{K}),\ K \in \Theta_h^f\},\\
\mathcal{V}_h^f &= \{\boldsymbol{v}_h \in (\mathcal{C}^0(\Omega_0))^d \mid \boldsymbol{v}_h|_K = \tilde{\boldsymbol{v}} \circ F_K^{-1},\ \tilde{\boldsymbol{v}} \in R_{k_\mathrm{v}}(\tilde{K}),\ K \in \Theta_h^f\}.
\end{aligned}$$

Further, we denote by Ξ_h^f the restriction of Θ_h^f onto Σ^{ft}, that is a finite element partition of the $(d-1)$-manifold Σ^{ft}.

Analogously, we define $(\Theta_h^m)_0$ as the finite element partition of the moving domain Ω_0^m at the reference configuration (assumed for simplicity at $t = 0$) in a family of elements $\{K_e\}_{e=1}^{n_{el}^m}$, n_{el}^m being the number of elements in this domain. We define the *reference* finite element spaces,

$$\begin{aligned}
(\mathcal{Q}_h^m)_0 &= \{\widehat{q}_h \in \mathcal{C}^0(\Omega_0^m) \mid \widehat{q}_h|_K = \tilde{q} \circ F_K^{-1},\ \tilde{q} \in R_{k_\mathrm{p}}(\tilde{K}),\ K \in (\Theta_h^m)_0\},\\
(\mathcal{V}_h^m)_0 &= \{\widehat{\boldsymbol{v}}_h \in (\mathcal{C}^0(\Omega_0^m))^d \mid \widehat{\boldsymbol{v}}_h|_K = \tilde{\boldsymbol{v}} \circ F_K^{-1},\ \tilde{\boldsymbol{v}} \in R_{k_\mathrm{v}}(\tilde{K}),\ K \in (\Theta_h^m)_0\}.
\end{aligned}$$

Let us define the moving domain at every time value $t \in [0, T]$ by means of a mapping as

$$\Omega_t^m = \mathcal{A}_t(\Omega_0^m) \tag{9.8}$$

where the mapping (see Chapter 6) must represent the movement of the domain (glued to the rotating component),

$$\mathcal{A}_t = \mathbf{T}(\boldsymbol{\omega} t), \tag{9.9}$$

$\boldsymbol{\omega}$ being the angular velocity vector that dictates the movement of the rotary body and $\mathbf{T}$ the rotation matrix of angle $\boldsymbol{\omega} t$. With these ingredients we can extend the *reference* finite element spaces to an arbitrary time step value t as,

$$\begin{aligned}
(\mathcal{Q}_h^m)_t &= \{q_h \in \mathcal{C}^0(\Omega_t) \mid q_h = \widehat{q}_h \circ \mathcal{A}_t^{-1},\ \widehat{q}_h \in (\mathcal{Q}_h^m)_0\},\\
(\mathcal{V}_h^m)_t &= \{\boldsymbol{v}_h \in (\mathcal{C}^0(\Omega_t))^d \mid \boldsymbol{v}_h = \widehat{\boldsymbol{v}}_h \circ \mathcal{A}_t^{-1},\ \widehat{\boldsymbol{v}}_h \in (\mathcal{V}_h^m)_0\},
\end{aligned}$$

and we also write, symbolically,

$$(\Theta_h^m)_t = \mathcal{A}_t((\Theta_h^m)_0),$$

that is to say, the mesh at each time t is a rotation of the initial mesh. Let us also define by $(\Xi_h^m)_t$ the restriction of $(\Theta_h^m)_t$ onto Σ^{mt}.

In order to account for the mesh movement we have considered the ALE formulation. In this particular case, the domain velocity $\boldsymbol{w}$ is,

$$\boldsymbol{w} = \boldsymbol{\omega} \times \boldsymbol{r}, \tag{9.10}$$

$\boldsymbol{r}$ being the position vector respect to the rotation center of the moving object. Thus, the mesh velocity $\boldsymbol{w}_h \in (\mathcal{V}_h^m)_t$ is such that its nodal values are equal to $\boldsymbol{w}(\boldsymbol{x}, t)$.

Remark 9.1. Given a known angular velocity, the ALE formulation is *exact*. As showed in Chapter 6, the ALE approach introduces an error related to the approximation of the time derivative of the displacement needed for the assessment of the mesh velocity. However, the computation of the mesh velocity defined above does not introduce any error. In case of calculating numerically the angular velocity, the ALE approach is affected by the error related to the assessment of the angular velocity.

Finally, we consider the transmission domain. This is the only place where we carry out the remeshing strategy. Given a time value t, $(\Theta_h^{tr})_t$ is a finite element partition of the domain Ω^{tr} in a family of elements $\{K_e\}_{e=1}^{n_{el}^{tr}}$, n_{el}^{tr} being the number of elements of this domain. We denote by Ξ_h^{ft} and $(\Xi_h^{mt})_t$ the finite element partitions defined as the restrictions of $(\Theta_h^{tr})_t$ onto Σ^{ft} and Σ^{mt}. We introduce the following finite element space (for a time value t) associated to this partition:

$$(\mathcal{Q}_h^{tr})_t = \{\widehat{q}_h \in \mathcal{C}^0(\Omega_0) \mid \widehat{q}_h|_K = \tilde{q} \circ F_K^{-1},\ \tilde{q} \in R_{k_{\mathrm{p}}}(K),\ K \in (\Theta_h^{tr})_t\},$$
$$(\mathcal{V}_h^{tr})_t = \{\widehat{\boldsymbol{v}}_h \in (\mathcal{C}^0(\Omega_0))^d \mid \widehat{\boldsymbol{v}}_h|_K = \tilde{\boldsymbol{v}} \circ F_K^{-1},\ \tilde{\boldsymbol{v}} \in R_{k_{\mathrm{v}}}(K),\ K \in (\Theta_h^{tr})_t\}.$$

The matching conditions that must hold $(\Theta_h^{tr})_t$ in order to have an appropriate finite element partition over the whole domain Ω_t are

$$\Xi_h^{ft} = \Xi_h^f \tag{9.11}$$
$$(\Xi_h^{mt})_t = (\Xi_h^m)_t. \tag{9.12}$$

These conditions mean that the overall mesh has to hold *continuity* on the interfaces. In Figure 9.2 we sketch the process for a simple $2d$ case. Θ_h^f is fixed, $(\Theta_h^m)_t$ is rotating, and $(\Theta_h^{tr})_t$ is generated every time step matching the previous meshes. If these conditions hold, the set of finite elements,

$$(\Theta_h)_t = \{\Theta_h^f, (\Theta_h^m)_t, (\Theta_h^{tr})_t\} \tag{9.13}$$

becomes an appropriate finite element partition of the whole domain Ω_t. With all these ingredients we can define the overall finite element spaces

$$(\mathcal{Q}_h)_t = \mathcal{Q}_h^f \oplus (\mathcal{Q}_h^m)_t \oplus (\mathcal{Q}_h^{tr})_t, \tag{9.14}$$
$$(\mathcal{V}_h)_t = \mathcal{V}_h^f \oplus (\mathcal{V}_h^m)_t \oplus (\mathcal{V}_h^{tr})_t, \tag{9.15}$$
$$(\mathcal{V}_{h,0})_t = \{\boldsymbol{v}_h \in (\mathcal{V}_h)_t \mid \boldsymbol{v}_h|_{\Gamma_{D_t}} = 0\}, \tag{9.16}$$

where Γ_{D_t} is the Dirichlet boundary at the time step value t. Thus, we have build up the pressure and velocity finite element spaces.

Therefore, with the introduction of the ALE framework and splitting the domain in three different parts, we are able to construct a method that only involves a remeshing in

a tiny part of the domain. This method does not introduce any new source of error and does not deserve any iterative procedure. Moreover, it does not involve the imposition of any transmission condition, highly advisable when using pressure segregation methods. These features justify the choice of this methodology for the numerical simulation showed later on.

Now, we must consider a pending ingredient, the *remeshing* process. In our particular case, the remeshing procedure consists of the mesh generation of a volume (or surface for $2d$ problems) given the surface (or line) meshes Ξ_h^{ft} and $(\Xi_h^{mt})_t$.

We adopt herein a particular algorithm called *shear slip layer* (SSL), in order to simplify this remeshing procedure. This method has been proposed and used by Behr and Tezduyar in [10, 11] in conjunction with their *DSD/SST* methodology. The SSL algorithm allows the generation of the transmission domain mesh just using a reconnecting procedure. However, this methodology can only be applied to some specific sorts of regular partitions of the interfaces.

The SSL method can be used if and only if there exists a finite element partition $(\Theta_h^{tr})_t$ such that connects the surface meshes Ξ_h^{ft} and $(\Xi_h^{mt})_t$ without the introduction of extra nodes. Unfortunately, the existence of this kind of reconnecting meshes is restricted to very special situations. For $2d$ problems, this methodology is very effective (see Figure 9.2). It is easy to generate *surface* meshes allowing the SSL method. Moreover, the projection is obvious due to the fact that we do not introduce or eliminate nodes. However, for $3d$ problems, surface meshes allowing the SSL reconnection are too much restrictive, and in most cases show large aspect ratios (see Figure 9.8).

As pointed out above, the movement of the domain is considered by an ALE formulation. The Navier-Stokes equations written in an ALE framework are:

$$\partial_t \boldsymbol{u} + (\boldsymbol{u} - \boldsymbol{w}) \cdot \nabla \boldsymbol{u} - \frac{1}{\rho} \nabla \cdot \boldsymbol{\sigma} = \boldsymbol{f} \quad \text{in } \Omega \times (0, T), \tag{9.17a}$$

$$\nabla \cdot \boldsymbol{u} = 0 \quad \text{in } \Omega \times (0, T), \tag{9.17b}$$

where the mesh velocity $\boldsymbol{w}$ is

$$\boldsymbol{w}(\boldsymbol{x}, t) = \begin{cases} \boldsymbol{\omega}(t) \times \boldsymbol{r} & \text{if } \boldsymbol{x} \in \Omega_t^m \\ \boldsymbol{0} & \text{otherwise} \end{cases} \tag{9.18}$$

The mesh velocity is only different from zero in the moving domain. The finite element approximation of (9.17)-(9.18) using the finite element spaces (9.14), (9.15) and (9.16) together with the discrete mesh velocity $\boldsymbol{w}_h \in \mathcal{V}_h$ such that its nodal values are equal to $\boldsymbol{w}(\boldsymbol{x}, t)$ leads to the method to be used.

In the next chapter, we consider a selective remeshing procedure for the simulation of wind turbines. Initially, a $2d$ problem is solved in order to explain the proposed methodology. In this case the selective remeshing strategy using the SSL algorithm is a really nice choice. After that, we simulate wind turbines, obtaining aeroelastic forces. In this $3d$ case, the SSL can also be used and it does not introduce any problem apart from the difficult task of generating meshes satisfying the condition between Ξ_h^{ft} and $(\Xi_h^{mt})_t$ described above and the high aspect ratio of the overall mesh.

9.4 Numerical experimentation

9.4.1 A $2d$ test problem

Let us start with a simple test problem that will allow us to show the remeshing procedure introduced above in order to handle problems involving rotating bodies. The problem

under consideration consists of a body that rotates around its geometrical center with a known angular velocity. In Figure 9.1 we show the domain under consideration. We can easily distinguish between the moving domain, where it is allocated the body, the transmission domain (the layer in white), and the fixed domain. The moving domain mesh is rotating fixed to the object. Thus, the transmission domain (that is not meshed in Figure 9.1) will be meshed in order for the fixed and moving meshes to match. In Figure 9.2 we sketch the procedure for the remeshing of the transmission domain. In particular, we show the SSL methodology, where there is just a reconnection between "surface" meshes.

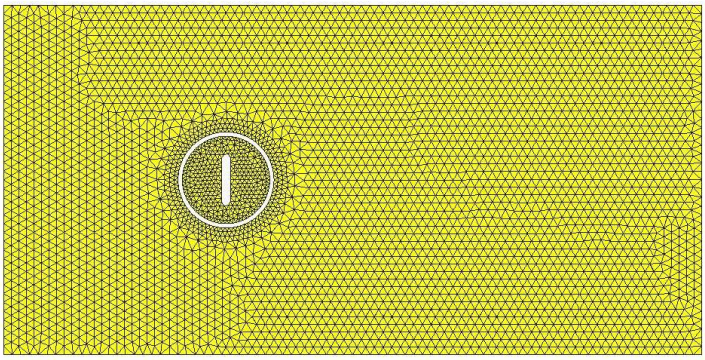


Figure 9.1: Mesh used for the $2d$ test problem

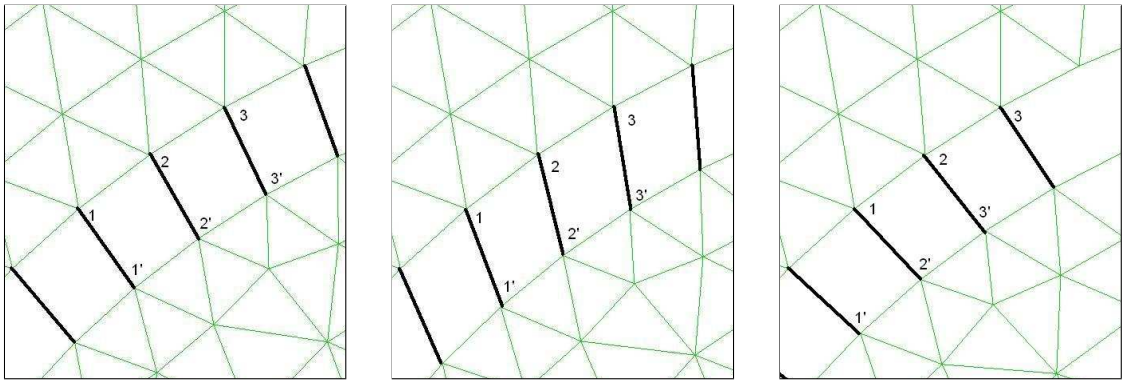


Figure 9.2: SSL reconnecting process for $2d$ problems

It can be easily seen from Figure 9.2 that for $2d$ problems the obtention of meshes that can be connected using SSL is an easy task. In particular, taking Σ^{ft} and Σ^{mt} as concentric circles, we have to impose their line meshes to be regular and with the same number of line elements.

Another aspect is the element type we will use on the different meshes. Different combinations are possible. Herein, we use triangle elements for the partition of the fixed and moving meshes. In the transmission domain we consider quadrilaterals (see Figure 9.2) that are naturally obtained from the reconnection of line elements. However, it would be possible to use triangles (see [10, 11]).

We have simulated this $2d$ problem using the following properties for the fluid: density $\rho = 1000$ and viscosity $\mu = 0.001$, that implies a Reynolds number of $Re = 10^6$. We

force the body to be rotating with a known angular velocity $\omega = 0.1\ rad/s$ and on the inflow boundary (right hand side edge) velocity components $u_x = 1\ m/s$ and $u_y = 0\ m/s$ are prescribed (x being the horizontal direction). In Figures 9.3 and 9.4 we show some velocity and pressure contour plots. The mesh velocity is composed by 8046 triangular elements plus 121 quadrilateral elements (belonging to the transmission mesh, that are being changed in time). We stress the fact that we have been able to solve this problem using a remeshing procedure that only involves a 13% of the overall mesh. This relationship is even lower for $3d$ problems.

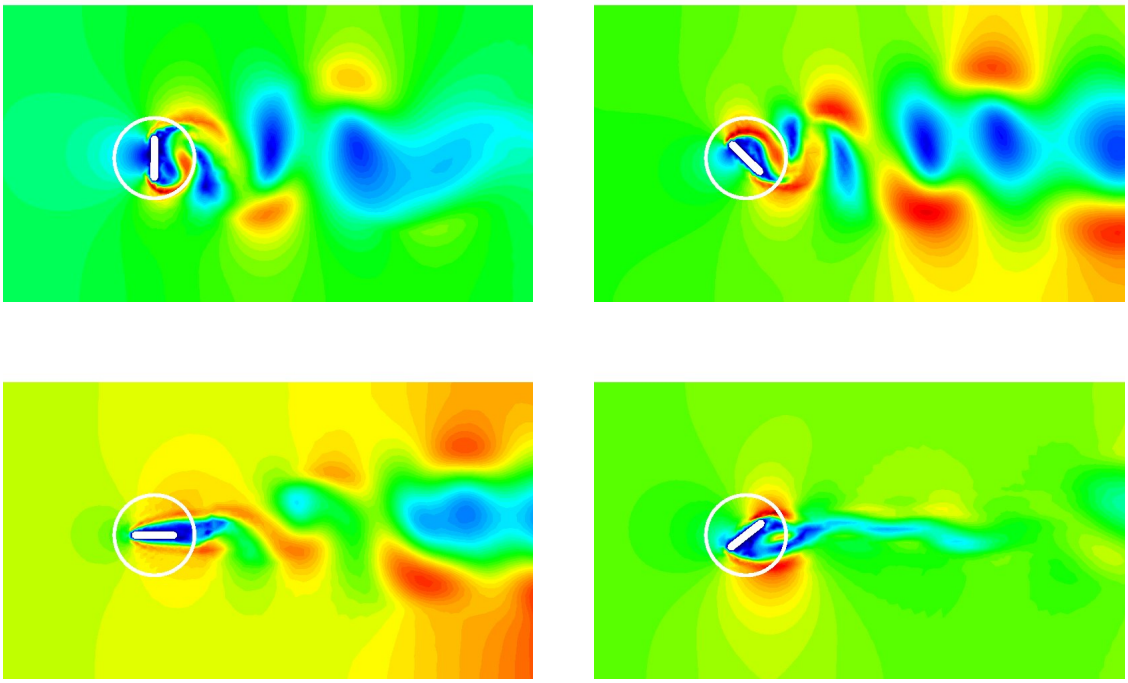


Figure 9.3: Test problem: Contours of velocity modulus for one cycle of the $2d$ rotating object

9.4.2 Wind turbines

The aim of this chapter is the simulation of a real wind turbine. The wind turbine for which we carry out the following analysis is shown in Figure 9.5. This wind turbine is a prototype model developed by Gamesa Eolica Co. (Navarra, Spain). This work has been developed in the frame of the Minister of Science and Technology Research Project ADEL, with reference BIA2003-09078-CO2-00.

We prove that with the ingredients developed along this work we are able to simulate wind turbines. This is a good starting point. However, this topic deserves further research that would be developed in a near future and that has been pointed out at the end of this chapter.

Let us state the problem to be simulated. We consider that the wind turbine is surrounded by air, characterized by the following physical properties: density $\rho = 1.21\ Kg/m^3$ and viscosity $\mu = 1.8 \cdot 10^{-5}\ Kg/ms$. All the components of the wind turbine are assumed rigid bodies. We consider a wind velocity of 10 m/s in the $x-$direction. The whole domain

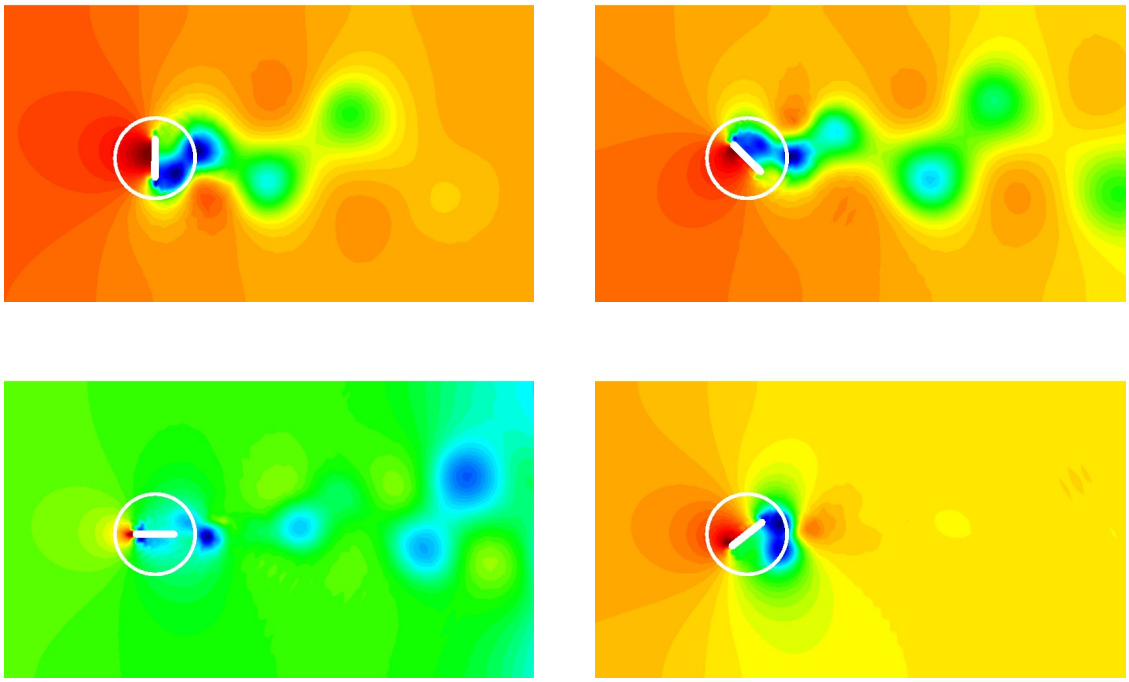


Figure 9.4: Test problem: Contours of pressure for one cycle of the $2d$ rotating object

Figure 9.5: The wind turbine geometry (Courtesy of Gamesa Eolica Co.)

of analysis is shown in Figure 9.6 together with the reference axes.

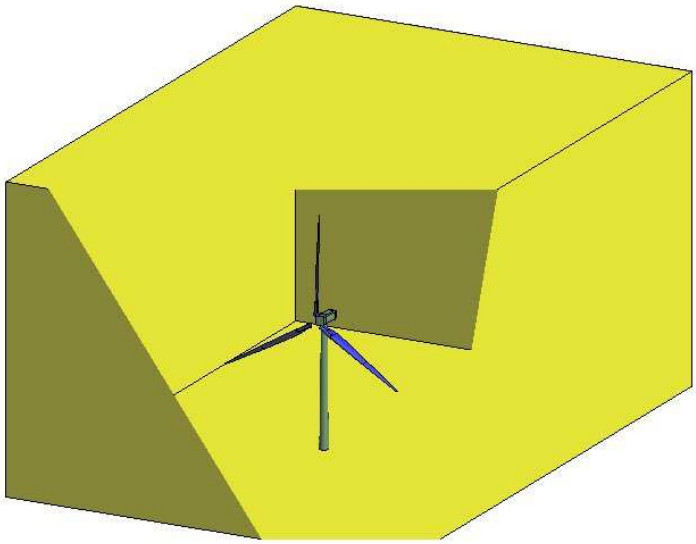


Figure 9.6: Whole domain of analysis

The domain has been meshed using 509286 tetrahedral elements together with 41040 triangular prisms that have been used to mesh the transmission domain. In this case, the SSL surface meshes to be reconnected (around a 7% of the overall mesh) are obviously partitioned into triangles. Thus, from the connection of two appropriate triangles we easily obtain a triangular prism (see Figure 9.9). This is the finite element used on Ω^{tr}. Nevertheless, this triangular prism could also be partitioned into tetrahedra.

We have split the mesh generation in three parts. First, we generate appropriate surface partitions of the interfaces Σ^{ft} and Σ^{mt} in order for these partitions to allow the SSL reconnecting procedure. Thus, respecting these surface meshes, we create the mesh of the fixed and moving domains separately. Details of the generated fixed and moving meshes and the interface surfaces are shown in Figures 9.7 and 9.8 respectively. Now, we must connect the SSL surface meshes in order to obtain the mesh of the transmission domain. We can see a detail of the transmission domain and the SSL meshes to be reconnected in Figure 9.9. It is easy to infer that the generation of meshes that would allow the SSL procedure for $3d$ problems is an involved task.

Besides the mesh, a second order pressure segregation method using BDF2 for the time integration has been chosen. This method has been stated in the set of Equations (3.18). Moreover we use the ALE framework in order to account for the movement of the rotating domain. This ALE framework does not introduce any error, because, as pointed out in Remark 9.1, the only perturbation of this formulation is the one arising from the numerical approximation of the mesh velocity, and in this case this mesh velocity is evaluated exactly. Thus, the overall fluid solver is second order accurate in time.

We have solved the wind turbine problem for a period of 80 seconds with a constant time step size $\delta t = 0.025s$. In Figure 9.10 we show the plot of the aerodynamic forces and moments exerted by the wind over the blades (in time). The behavior of these aerodynamic forces is highly transient, feature that justifies the use of second order accurate schemes. We stress the fact that the high frequencies that can be seen in these plots are not numerical

instabilities. There are enough time steps for capturing these frequencies.

The Fourier transform of the magnitudes of more interest (the $x-$force and $x-$moment) allows us to identify the dominant frequencies. According to the position of the blades, the geometry is repeated every $2\pi/3$ rad, that implies, given an angular velocity of $\omega = 0.25$ rad/s, a period of 8 seconds, and thus a frequency of 0.125 Hz. As expected, the maximum amplitude coincides with this frequency. Moreover, in both cases, there is a smaller peak for a frequency around 2.1 Hz.

We have captured some isosurfaces of velocity and pressure that allow to visualize some aspects of the $3d$ problem. Figures 9.12 and 9.13 show the isosurfaces of velocity modulus, for 8 m/s and 9 m/s, respectively. These velocities are lower than the inflow velocity. We can easily see the wake due to the pile and the blades. The wake of the blades shows a helicoidal shape, due to the rotation. Figures 9.14 and 9.15 show the velocity isosurfaces for 10.5 m/s and 11 m/s, higher than the inflow velocity.

The pressure isosurfaces for negative values of Figures 9.16 and 9.17 are positioned behind the wind turbine, as expected. Obviously, for $p = -40$ N/m^2, the isosurfaces are closer to the blades than for $p = -10$ N/m^2. In Figures 9.18 and 9.19 we show isosurfaces of pressure for positive values.

In Figure 9.20 some particle trajectories have been drawn behind the wind turbine.

9.5 Conclusions

The present chapter has been devoted to the numerical simulation of problems that involve fixed and rotary objects and, consequently, the domain changes in time. In particular, we have analyzed a wind turbine.

We have suggested two different numerical techniques that can manage this situation: domain decomposition methods and remeshing procedures. In both cases, we consider that a pressure segregation method is being used for the simulation of the flow.

The domain decomposition approach has been commented, underlining the aspects related to the fact that the fluid solver is a pressure correction scheme. We have addressed some important aspects to take into account when imposing the transmission conditions.

After that, we have suggested the use of a selective remeshing procedure that only affects a tiny portion of the whole domain, used together with an ALE formulation. This strategy does not introduce transmission conditions, and in the particular case under consideration (rotating bodies) seems clearly superior. On the other hand, we have considered a simplified remeshing procedure (the SSL algorithm) that consists of a reconnection between appropriate meshes.

We have ended with the numerical experimentation. First, a $2d$ test case has been introduced in order to explain the remeshing procedure by using a simple problem . After that, we have stated the wind turbine problem and shown the results that have been obtained by means of numerical experimentation. In particular, the time evolution of the different aerodynamic forces has been obtained.

The selective remeshing strategy together with an ALE formulation seems to be a powerful technique for the problems simulated herein. The SSL algorithm works well for $2d$ problems, but makes the generation of the mesh for $3d$ problems rigid and involved.

The previous results are a motivating starting point. However, the numerical simulation applied to wind energy deserves much more research, such as the introduction of fluid-structure interaction algorithms (see Chapter 7), the use of frontal methods (or Delaunay methods) for the generation of the *transmission* mesh or the use of appropriate structure solvers for the blades (that are made of composite materials).

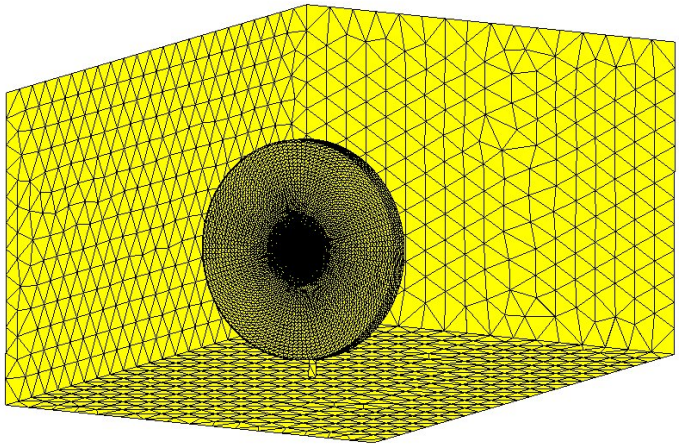


Figure 9.7: Detail of surface meshes of the fixed domain (fictitious boundary and SSL external surface mesh Ξ_h^f)

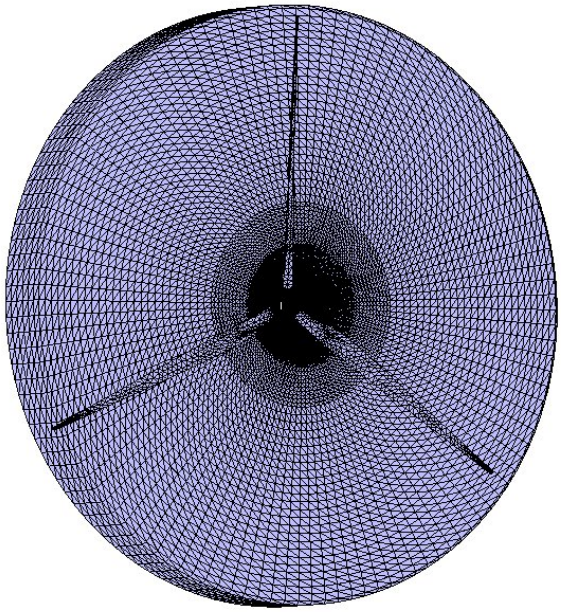


Figure 9.8: Detail of surface meshes of the moving domain(SSL internal mesh $(\Xi_h^m)_t$ and blades)

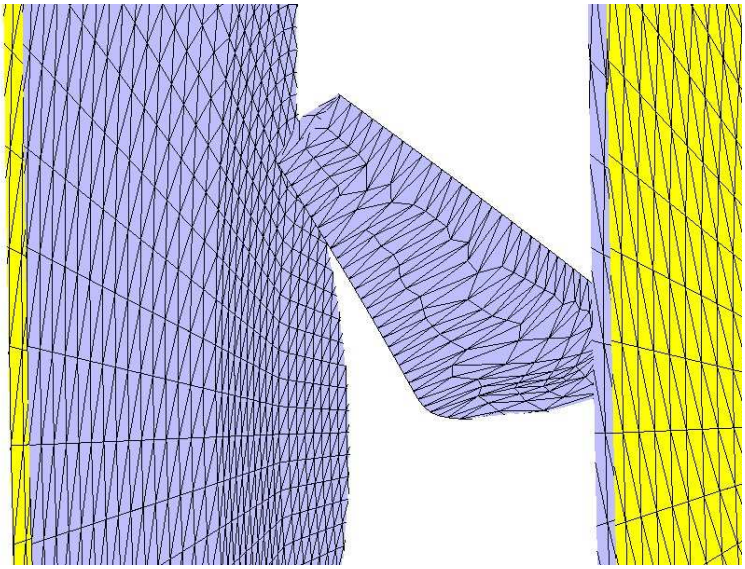
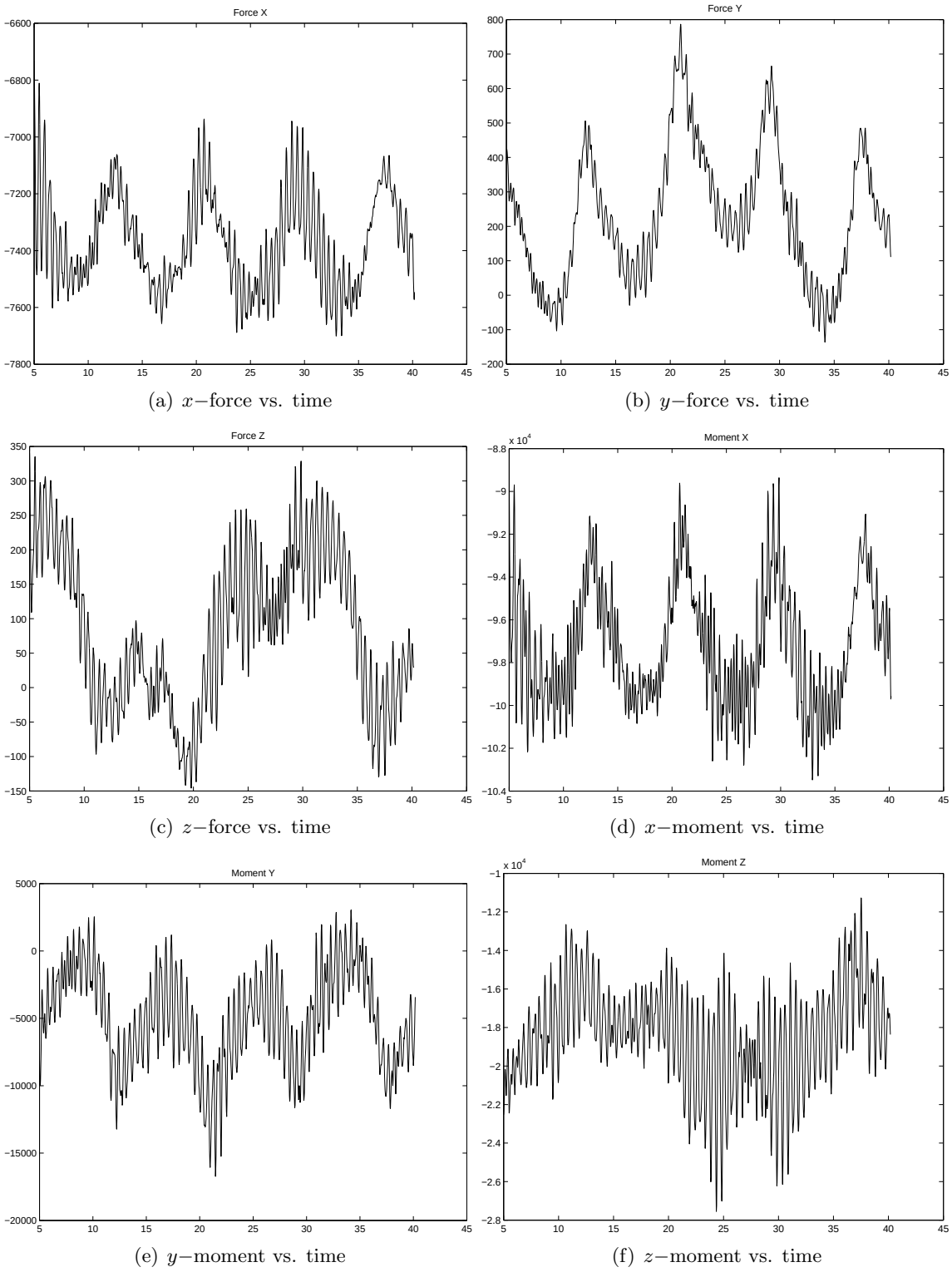


Figure 9.9: Detail of the transmission domain Ω^{tr} between the external (yellow) and internal (blue) SSL surface meshes



(a) $x-$force vs. time

(b) $y-$force vs. time

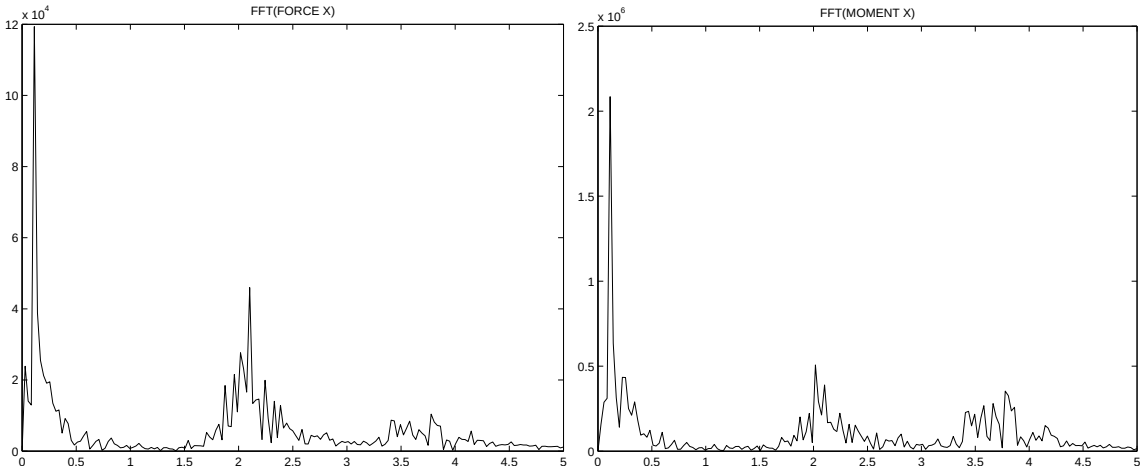
(c) $z-$force vs. time

(d) $x-$moment vs. time

(e) $y-$moment vs. time

(f) $z-$moment vs. time

Figure 9.10: Time evolution of the value of aerodynamic forces exerted over the blades



(a) $x-$force: amplitude vs. frequency (b) $x-$moment: amplitude vs. frequency

Figure 9.11: Fast Fourier transform of the x-force and x-moment over the blades

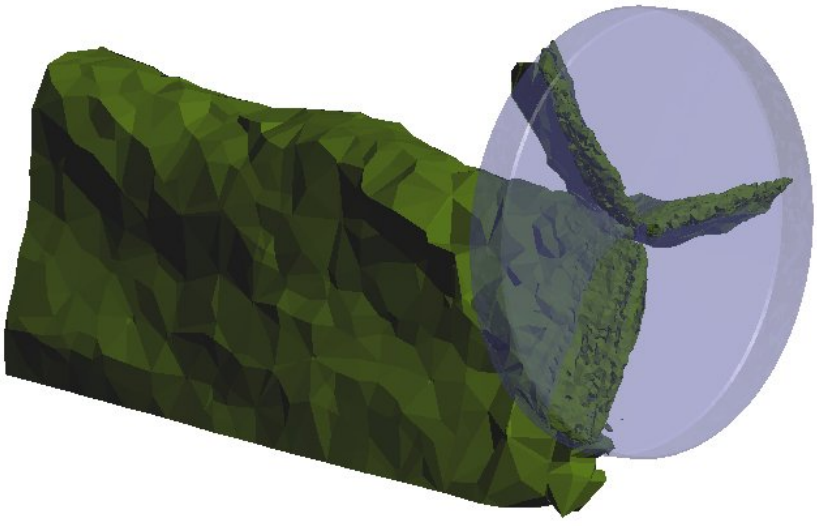


Figure 9.12: Isosurfaces of velocity modulus for $|\boldsymbol{u}| = 8\ m/s$

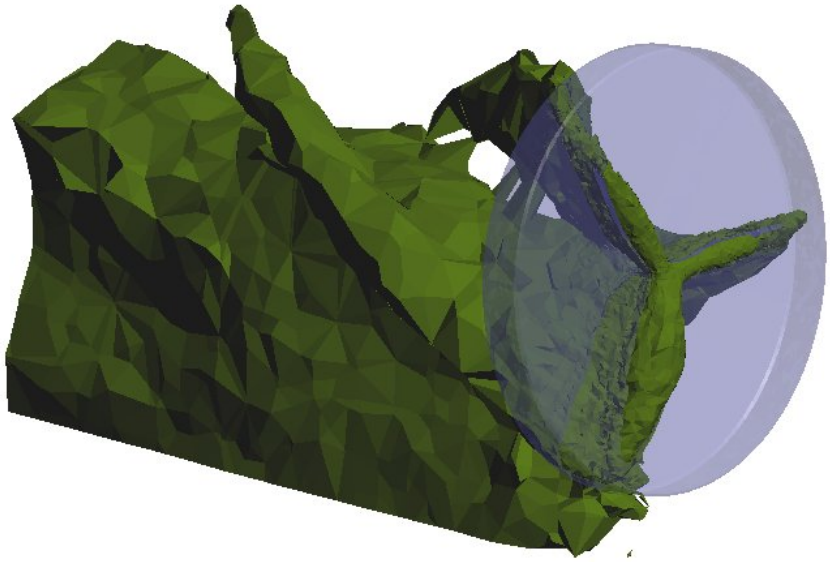


Figure 9.13: Isosurfaces of velocity modulus for $|\boldsymbol{u}| = 9\ m/s$

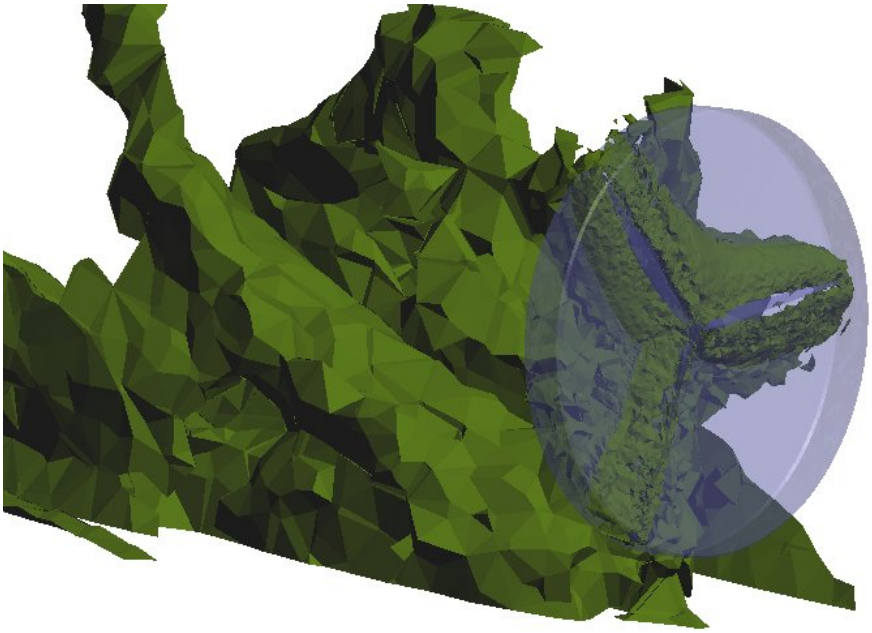


Figure 9.14: Isosurfaces of velocity modulus for $|\boldsymbol{u}| = 10.5\ m/s$

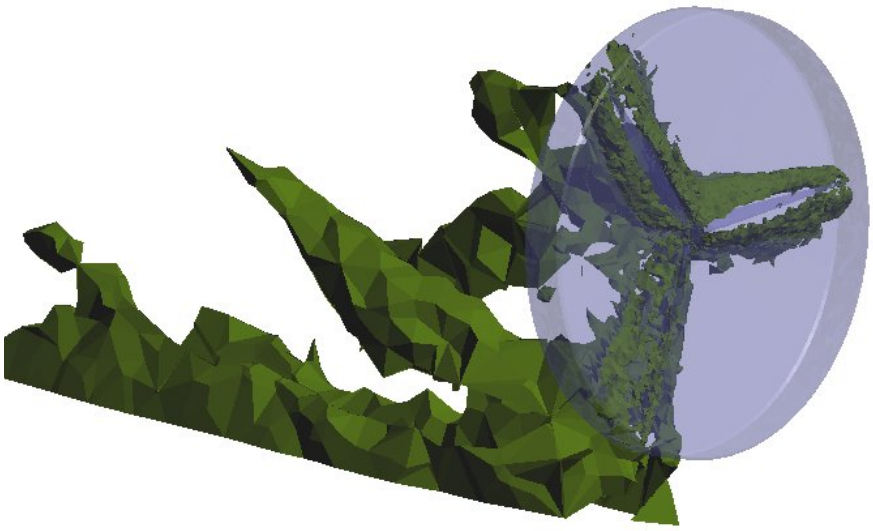


Figure 9.15: Isosurfaces of velocity modulus for $|\boldsymbol{u}| = 11\ m/s$

Figure 9.16: Isosurfaces of pressure for $p = -40\ N/m^2$

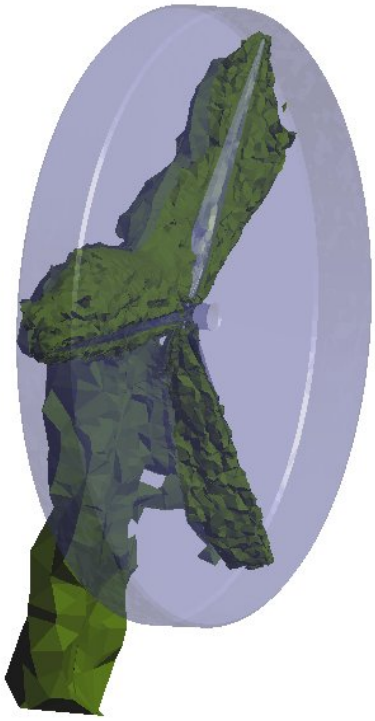


Figure 9.17: Isosurfaces of pressure for $p = -10\ N/m^2$

Figure 9.18: Isosurfaces of pressure for $p = 10\ N/m^2$

Figure 9.19: Isosurfaces of pressure for $p = 40\ N/m^2$

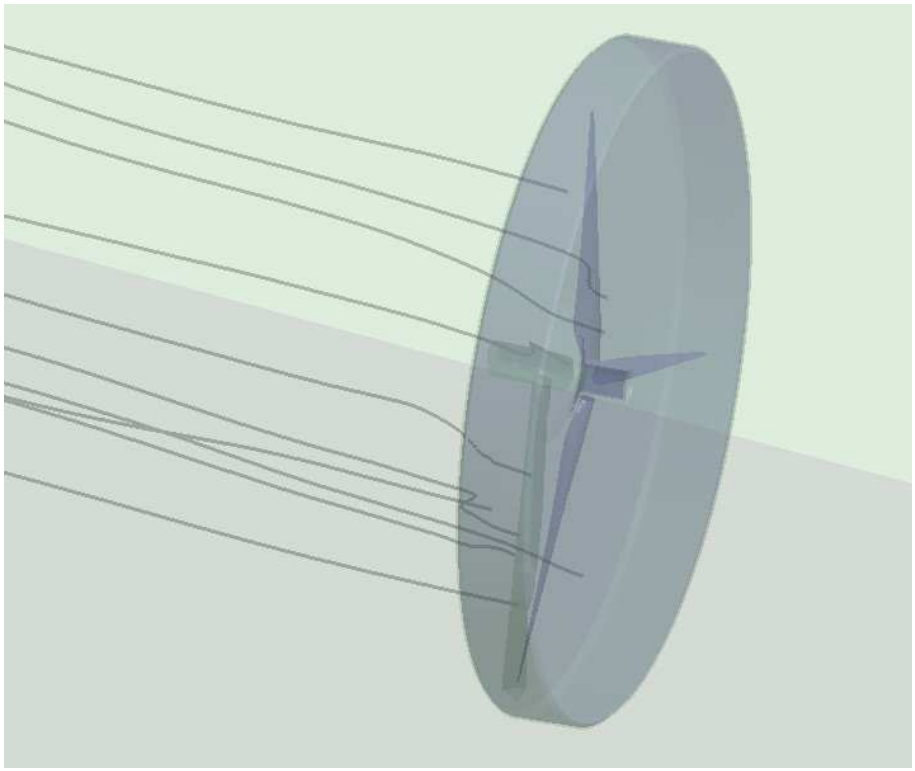


Figure 9.20: Trajectories of particles behind the wind turbine

Open lines of research

The aim of this section is not to state a list of conclusions. We found more appropriate to draw the conclusions at the chapter level due to the heterogeneous nature of this work. This section is devoted to the statement of some open aspects of this work that could be object of further research. These research lines are:

- The most important question about pressure segregation methods that still remains open is the unstable behavior of pressure correction methods with a third (or higher) order splitting error. We have found a discrepancy in the literature about third order pressure correction methods and the only attempt to prove this instability seems that is not correct. The situation is even more mysterious when we have found that a different kind of pressure segregation methods, velocity correction methods, also exhibit this misbehavior.

- About velocity correction methods, it would be interesting to apply these methods to real applications, specially predictor corrector schemes. Further, some aspects about the artificial boundary conditions, specially for predictor corrector methods, remain unclear. Moreover, we consider that a minor modification of the discrete pressure Poisson equation could introduce consistent boundary conditions over the pressure.

- We have found that it would be nice to extend the numerical analysis of the first order pressure correction method using a Poisson equation for the pressure (not satisfying the inf-sup condition) to a second order one (that requires a stabilization technique). We envisage that the pressure estimates will improve with the introduction of a pressure extrapolation in the momentum equation.

- The fluid-structure algorithm problem that we have used herein consists of a predictor corrector scheme that profits from the coupling iterations of a fixed point algorithm. The behavior of the artificial boundary conditions over the solid boundary with the predictor corrector iterations deserves more attention, as well as its impact on the results.

- The fluid-structure applications of the present monograph have been restricted to aeroelastic problems. We find that the use of pressure correction methods together with more involved coupling procedures (Newton-like, for instance) could be a useful tool for the numerical simulation of the blood-vessel system.

- There is a lot of work to do on the numerical simulation of wind turbines. The selective remeshing procedure together with an ALE formulation seems to be an excellent choice. However, even though we have obtained good results, the use of the SSL reconnecting strategy makes the mesh generation tedious and involved and restricts the geometries that can be solved. The introduction of a frontal (or Delaunay)

technique for the generation of the transmission mesh would increase substantially the flexibility of the overall method. Other aspects that must be considered in a future are the introduction of fluid-structure interaction algorithms and the use of appropriate structure solvers for the blades (that are made of composite materials). These methods would have to be validated comparing numerical and experimental results.

- In this work we have not used any turbulence model for the numerical simulations. However, we are using a stabilized finite element method motivated from the split of the solution into fine and coarse scales. The relationship between stabilized methods obtained from a multiscale setting and turbulence modelling is a current subject of research and is becoming one of the most active topics in computational fluid dynamics. Thus, the development of stabilized methods based on the orthogonal subgrid idea that would replace conventional turbulence models are a promising line of research.

References

[1] A.A.O. Ammi and M. Marion. Nonlinear Galerkin methods and mixed finite elements: two-grid algorithms for the Navier-Stokes equations. *Numerische Mathematik*, 68:189–213, 1994.

[2] F. Armero and J.C. Simo. Long-term dissipativity of time-stepping algorithms for an abstract evolution equation with application to MHD and Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 131:41–90, 1996.

[3] I. Babuŝka. Error bounds for the finite element method. *Numerische Mathematik*, 16:322–333, 1971.

[4] S. Badia and R. Codina. Analysis of a stabilized finite element approximation of the transient convection-diffusion equation using an ALE framework. *SIAM Journal on Numerical Analysis*, submitted.

[5] S. Badia and R. Codina. Convergence analysis of the FEM approximation of the first order projection method for incompressible flows with and without the inf-sup condition. *Numerische Mathematik*, submitted.

[6] S. Badia and R. Codina. On some fluid-structure iterative algorithms using pressure segregation methods. Application to aeroelasticity. *International Journal for Numerical Methods in Engineering*, submitted.

[7] S. Badia and R. Codina. Velocity correction methods based on a discrete pressure Poisson equation. An algebraic approach. *Computer Methods in Applied Mechanics and Engineering*, submitted.

[8] C. Baiocchi, F. Brezzi, and L.P. Franca. Virtual bubbles and Galerkin/least-squares type methods (Ga.L.S). *Computer Methods in Applied Mechanics and Engineering*, 105:125–141, 1993.

[9] G.A. Baker, V.A. Dougalis, and A. Karakashian. On a higher order accurate fully discrete Galerkin approximation to the Navier-Stokes equations. *Mathematics of Computation*, 39:339–375, 1982.

[10] M. Behr and T.E. Tezduyar. The shear-slip mesh update method. *Computer Methods in Applied Mechanics and Engineering*, 174:261–274, 1999.

[11] M. Behr and T.E. Tezduyar. Shear-slip mesh update in 3d computation of complex flow problems with rotating mechanical components. *Computer Methods in Applied Mechanics and Engineering*, 190:3189–3200, 2001.

[12] J.B. Bell, P. Colella, and H.M. Glaz. A second-order projection method for the incompressible Navier-Stokes equations. *Journal of Computational Physics*, 85:257–283, 1989.

[13] J.A. Benek, P.G. Buning, and J.L. Steger. A 3-D Chimera grid enbedding technique. *AIAA*, Paper 85-1523CP, 1985.

[14] J.A. Benek, T.L. Tonegan, and N.E. Suhs. Extended Chimera grid embedding system with application to viscous flow. *AIAA*, Paper 87-1126, 1987.

[15] J. Blasco and R. Codina. Space and time error estimates for a first order, pressure stabilized finite element method for the incompressible Navier-Stokes equations. *Applied Numerical Mathematics*, 38:475–497, 2001.

[16] J. Blasco, R. Codina, and A. Huerta. A fractional step method for the incompressible Navier-Stokes equations related to a predictor-multicorrector algorithm. *International Journal for Numerical Methods in Fluids*, 28:1391–1419, 1998.

[17] F.J. Blom. A monolithic fluid-structure interaction algorithm applied to the piston problem. *Computer Methods in Applied Mechanics and Engineering*, 167:369–391, 1998.

[18] P. Bochev, Z. Cai, T.A. Manteuffel, and S.F. McCormick. Analysis of velocity-flux first-order system least-squares principles for the Navier-Stokes equations: Part I. *SIAM Journal on Numerical Analysis*, 35:990–1009, 1998.

[19] D. Boffi and L. Gastaldi. Stability and geometric conservation laws for ALE formulation. *Computer Methods in Applied Mechanics and Engineering*, 193:4717–4739, 2004.

[20] J.M. Boland and R.A. Nicolaides. Stability of finite elements under divergence constraints. *SIAM Journal on Numerical Analysis*, 20:722–731, 1983.

[21] M. Braack and E. Burman. A multiscale method towards turbulent flow based on local projection stabilization. *SIAM Journal on Numerical Analysis*, submitted.

[22] M. Braack and E. Burman. Local projection stabilization for the Oseen problem and its interpretation as a variational multiscale method. *SIAM Journal on Numerical Analysis*, to appear.

[23] S.C. Brenner and L.R. Scott. *The mathematical theory of finite element methods.* Springer–Verlag, 1994.

[24] F. Brezzi. On the existence, uniqueness and approximation of saddle point problems arising from lagrange multipliers. *RAIRO Anal. Numer.*, 8:129–151, 1974.

[25] F. Brezzi and K.J. Bathe. A discourse on the stability conditions for mixed finite element formulations. *Computer Methods in Applied Mechanics and Engineering*, 82:27–57, 1990.

[26] F. Brezzi, M.O. Bristeau, L. Franca, M. Mallet, and G. Rogé. A relationship between stabilized finite element methods and the Galerkin method with bubble functions. *Computer Methods in Applied Mechanics and Engineering*, 96:117–129, 1992.

[27] F. Brezzi and J. Douglas. Stabilized mixed methods for the Stokes problem. *Numerische Mathematik*, 53:225–235, 1988.

[28] F. Brezzi and M. Fortin. *Mixed and hybrid finite element methods*. Springer Verlag, 1991.

[29] F. Brezzi, D. Marini, and E. Süli. Residual-free bubbles for advection-diffusion problems: the general error analysis. *Numerische Mathematik*, 85:31–47, 2000.

[30] F. Brezzi, J. Rappaz, and P.A. Raviart. Finite dimensional approximation of nonlinear problems. Part I: Branches of non-singular solutions. *Numerische Mathematik*, 36:1–25, 1981.

[31] F. Brezzi, J. Rappaz, and P.A. Raviart. Finite dimensional approximation of nonlinear problems. Part II: Limit points. *Numerische Mathematik*, 37:1–28, 1981.

[32] F. Brezzi, J. Rappaz, and P.A. Raviart. Finite dimensional approximation of nonlinear problems. Part III: Simple bifurcation points. *Numerische Mathematik*, 38:1–30, 1981.

[33] A.N. Brooks and T.J.R. Hughes. Streamline upwind / Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equation. *Computer Methods in Applied Mechanics and Engineering*, 32:199–259, 1982.

[34] R.M. Brown, P. Perry, and Z. Shen. The additive turbulent decomposition for the two-dimensional incompressible Navier-Stokes equations: convergence theorems and error estimates. *SIAM Journal on Numerical Analysis*, 59:139–155, 1998.

[35] J.B. Burie and M. Marion. Multilevel methods in space and time for the Navier-Stokes equations. *SIAM Journal on Numerical Analysis*, 34:1574–1599, 1997.

[36] C. Canuto and V. van Kemenade. Bubble-stabilized spectral methods for the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 135:35–61, 1996.

[37] P. Causin, J.F. Gerbeau, and F. Nobile. Added-mass effect in the design of partitioned algorithms for fluid-structure problems. *Computer Methods in Applied Mechanics and Engineering*, 194(42-44):4506–4527, 2001.

[38] M. Cervera, R. Codina, and M. Galindo. On the computational efficiency and implementation of block-iterative algorithms for nonlinear coupled problems. *Engineering Computations*, 13(6):4–30, 1996.

[39] A.J. Chorin. A numerical method for solving incompressible viscous problems. *Journal of Computational Physics*, 2:12–26, 1967.

[40] A.J. Chorin. The numerical solution of the Navier–Stokes equations for an incompressible fluid. AEC Research and Development Report, NYO-1480-82. New York University, New York, 1967.

[41] A.J. Chorin. Numerical solution of the Navier–Stokes equations. *Mathematics of Computation*, 22:745–762, 1968.

[42] A.J. Chorin. On the convergence of discrete approximation to the Navier–Stokes equations. *Mathematics of Computation*, 23, 1969.

[43] P.G. Ciarlet. *The finite element method for elliptic problems*. North-Holland, Amsterdam, 1978.

[44] Clay Mathematics Institute. *http://www.claymath.org/millenium/*. Millenium Problems, 2000.

[45] R. Codina. Comparison of some finite element methods for solving the diffusion-convection-reaction equation. *Computer Methods in Applied Mechanics and Engineering*, 156:185–210, 1998.

[46] R. Codina. Stabilization of incompressibility and convection through orthogonal subscales in finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 190:1579–1599, 2000.

[47] R. Codina. Pressure stability in fractional step finite element methods for incompressible flows. *Journal of Computational Physics*, 170:112–140, 2001.

[48] R. Codina. A stabilized finite element method for generalized stationary incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 190:2681–2706, 2001.

[49] R. Codina. Stabilized finite element approximation of transient incompressible flows using orthogonal subscales. *Computer Methods in Applied Mechanics and Engineering*, 191:4295–4321, 2002.

[50] R. Codina and S. Badia. Métodos de predicción-corrección basados en esquemas de pasos fraccionados para la resolución de las ecuaciones de Navier-Stokes. In *Congreso de Métodos Computacionais em Engenharia*, Lisboa, Portugal, 2003.

[51] R. Codina and S. Badia. Second order fractional step schemes for the incompressible Navier-Stokes equations. Inherent pressure stability and pressure stabilization. In *Proceedings of WCCM VI*, Beijing, China, 2004.

[52] R. Codina and S. Badia. On some pressure segregation methods of fractional-step type for the finite element approximation of incompressible flow problems. *Computer Methods in Applied Mechanics and Engineering*, in press.

[53] R. Codina and J. Blasco. A finite element formulation for the Stokes problem allowing equal velocity-pressure interpolation. *Computer Methods in Applied Mechanics and Engineering*, 143:373–391, 1997.

[54] R. Codina and J. Blasco. Analysis of a pressure-stabilized finite element approximation of the stationary Navier-Stokes equations. *Numerische Mathematik*, 87:59–81, 2000.

[55] R. Codina and J. Blasco. Analysis of a stabilized finite element approximation of the transient convection-diffusion-reaction equation using orthogonal subscales. *Computing and Visualization in Science*, 4:167–174, 2002.

[56] R. Codina and A. Folch. A stabilized finite element predictor–corrector scheme for the incompressible Navier–Stokes equations using a nodal based implementation. *International Journal for Numerical Methods in Fluids*, 44:483–503, 2004.

[57] R. Codina and O. Soto. Approximation of the incompressible Navier–Stokes equations using orthogonal–subscale stabilization and pressure segregation on anisotropic finite element meshes. *Computer Methods in Applied Mechanics and Engineering*, 193:1403–1419, 2004.

[58] P. Constantin and C. Foias. *Navier-Stokes Equations*. University of Chicago Press, Chicago and London, 1988.

[59] G. de Rham. *Variétés différentiables formes, courants, formes harmoniques*. Paris Hermann, 1973.

[60] S. Deparis. *Numerical analysis of axisymmetric flows and methods for fluid-structure interaction arising in blood flow simulation*. PhD thesis, École Polytechnique Fédérale de Lausanne, 2004.

[61] S. Deparis, M. Discacciati, and A. Quarteroni. A domain decomposition framework for fluid-structure interaction problems. In *Proceedings of ICCFD3*, Toronto, 2004.

[62] S. Deparis, M.A. Fernández, and L. Formaggia. Acceleration of a fixed point algorithm for fluid-structure interaction using transpiration conditions. *Mathematical Modelling and Numerical Analysis*, 37(4):601–616, 2003.

[63] J. Donea. A Taylor-Galerkin method for convection transport problems. *International Journal for Numerical Methods in Engineering*, 20:101–119, 1984.

[64] J. Donea, P. Fasoli-Stella, and S. Giuliani. Lagrangian and Eulerian finite element techniques for transient fluid structure interaction problems. In *Transactions Fourth SMIRT*, page B1/2, 1977.

[65] J. Douglas and T. Russel. Numerical methods for convection dominated problems based on combining the method of characteristics with finite elements or finite difference procedures. *SIAM Journal on Numerical Analysis*, 19:871–885, 1982.

[66] W. E and J.G. Liu. Projection method I: Convergence and numerical boundary layers. *SIAM Journal on Numerical Analysis*, 32:1017–1057, 1995.

[67] C. Farhat, P. Geuzaine, and G. Brown. Application of a three-field nonlinear fluid-structure formulation to the prediction of the aeroelastic parameters of an F-16 fighter. *Computers & Fluids*, 32:3–29, 2003.

[68] C. Farhat, P. Geuzaine, and C. Grandmont. The discrete geometric conservation law and the non-linear stability of ALE schemes for the solution of flow problems on moving grids. *Journal of Computational Physics*, 174:669–692, 2001.

[69] C. Farhat, M. Lesoinne, and N. Maman. Mixed explicit/implicit time integration of coupled aerolastic problems: three-field formulation, geometric conservation and distributed solution. *International Journal for Numerical Methods in Fluids*, 21:807–835, 1995.

[70] C. Farhat, M. Lesoinne, and P. Le Tallec. Load and motion transfer algorithms for fluid/structure interaction problems with non-matching discrete interfaces: Momentum and energy conservation, optimal discretization and application to aeroelasticity. *Computer Methods in Applied Mechanics and Engineering*, 157:95–114, 1998.

[71] C. Farhat and F. Roux. An unconventional domain decomposition method for an efficient parallel solution of large-scale finite element systems. *SIAM Journal on Scientific and Statistical Computing*, 13:379–396, 1992.

[72] M.A. Fernández and M. Moubachir. A Newton method using exact Jacobians for solving fluid-structure coupling. *Computers & Structures*, 83(2-3):127–142, 2005.

[73] C. Foias, O. Manley, and R. Temam. Modelling of the interaction of small and large eddies in two-dimensional turbulent flows. *Mathematical Modelling and Numerical Analysis*, 22:93–118, 1988.

[74] L. Formaggia and F. Nobile. A stability analysis for the Arbitrary Lagrangian Eulerian formulation with finite elements. *East-West J. Num. Math.*, 7:105–132, 1999.

[75] L. Formaggia and F. Nobile. Stability analysis of second-order time accurate schemes for ALE-FEM. *Computer Methods in Applied Mechanics and Engineering*, 193:4097–4116, 2004.

[76] L. Franca and R. Stenberg. Error analysis of some Galerkin least-squares methods for the elasticity equations. *SIAM Journal on Numerical Analysis*, 28:1680–1697, 1991.

[77] L.P. Franca, A. Nesliturk, and M. Stynes. On the stability of residual free bubbles for convection-diffusion problems and their approximation by a two-level finite element method. *Computer Methods in Applied Mechanics and Engineering*, 166:35–49, 1998.

[78] J.B. Frandsen. Numerical bridge deck studies using finite elements. Part I: Flutter. *Journal of Fluids and Structures*, 19(2):171–191, 2003.

[79] J.F. Gerbeau and M. Vidrascu. Quasi-Newton algorithm based on a reduced model for fluid-structure interaction problems in blood flows. *Mathematical Modelling and Numerical Analysis*, 37(4):631–647, 2003.

[80] P. Geuzaine, C. Grandmont, and C. Farhat. Design and analysis of ALE schemes with provable second-order time accuracy for inviscid and viscous flow simulations. *Journal of Computational Physics*, 191(1):206–227, 2003.

[81] V. Girault and P.A. Raviart. *Finite element methods for Navier-Stokes equations*. Springer–Verlag, 1986.

[82] P. M. Gresho and R.L. Sani. *Incompressible flow and the finite element method*. John Wiley & Sons, 2000.

[83] P.M. Gresho. On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix. Part I: Theory. *International Journal for Numerical Methods in Fluids*, 11:587–620, 1990.

[84] P.M. Gresho, S.T. Chan, M.A. Christon, and A.C. Hindmarsh. A little more on stabilized q_1q_1 for transient viscous incompressible flow. *International Journal for Numerical Methods in Fluids*, 21:837–856, 1995.

[85] J.L. Guermond. Remarques sur les méthodes de projection pour l'approximation des équations de Navier-Stokes. *Numerische Mathematik*, 67:465–473, 1994.

[86] J.L. Guermond. Stabilization of Galerkin approximations of transport equations by subgrid modeling. *Mathematical Modelling and Numerical Analysis*, 33:1293–1316, 1999.

[87] J.L. Guermond. Subgrid stabilization of Galerkin approximations of linear contraction semi-groups of class $\mathcal{C}^0$. *Computing and Visualization in Science*, 2:131–138, 1999.

[88] J.L. Guermond, P. Minev, and J. Shen. An overview of projection methods for incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, to appear.

[89] J.L. Guermond, J.T. Oden, and S. Prudhomme. Mathematical perspectives on large eddy simulation models for turbulent flows. *Journal of Mathematical Fluid Mechanics*, 6:194–248, 2004.

[90] J.L. Guermond and S. Prudhomme. On the construction of suitable solutions to the Navier-Stokes equations and questions regarding the definition of large eddy simulation. *Physica D*, 207:64–78, 2005.

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

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

[93] J.L. Guermond and J. Shen. A new class of truly consistent splitting schemes for incompressible flows. *Journal of Computational Physics*, 192:262–276, 2003.

[94] J.L. Guermond and J. Shen. Velocity-correction projection methods for incompressible flows. *SIAM Journal on Numerical Analysis*, 41:112–134, 2003.

[95] J.L. Guermond and J. Shen. On the error estimates for the rotational pressure-correction projection methods. *Mathematics of Computation*, 73:1719–1737, 2004.

[96] E. Guilmineau, J. Piquet, and P. Queutey. Two-dimensional turbulent viscous flow simulation past airfoils at fixed incidence. *Computers & Fluids*, 26(2):135–162, 1997.

[97] J.G. Heywood and R. Rannacher. Finite element approximation of the nonstationary Navier-Stokes problem. I: Regularity of solutions and second-order error estimates for spatial discretization. *SIAM Journal on Numerical Analysis*, 19:275–311, 1982.

[98] J.G. Heywood and R. Rannacher. Finite element approximation of the nonstationary Navier-Stokes problem. IV: Error analysis for second-order time discretization. *SIAM Journal on Numerical Analysis*, 27:353–384, 1990.

[99] E. Hopf. Uber die aufangswertaufgabe für die hydrodynamischen grundgliechungen. *Mathematische Nachrichten*, 4:213–231, 1951.

[100] G. Houzeaux and R. Codina. A Chimera method based on a Dirichlet–Neumann (Robin) coupling for the Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 192:3343–3377, 2003.

[101] G. Houzeaux and R. Codina. An overlapping iteration-by-subdomain Dirichlet–Robin domain decomposition method for advection–diffusion problems. *Journal of Computational and Applied Mathematics*, 158:243–276, 2003.

[102] T.J.R. Hughes. Multiscale phenomena: Green's function, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized formulations. *Computer Methods in Applied Mechanics and Engineering*, 127:387–401, 1995.

[103] T.J.R. Hughes and A.N. Brooks. A multidimensional upwind scheme with no cross-wind diffusion. In T.J.R. Hughes, editor, *FEM for Convection Dominated Flows*. ASME, New York, 1979.

[104] T.J.R. Hughes, G.R. Feijóo, L. Mazzei, and J.B. Quincy. The variational multiscale method–a paradigm for computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 166:3–24, 1998.

[105] T.J.R. Hughes, L.P. Franca, and M. Balestra. A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuŝka-Brezzi condition: a stable Petrov-Galerkin formulation for the Stokes problem accommodating equal-order interpolations. *Computer Methods in Applied Mechanics and Engineering*, 59:85–99, 1986.

[106] T.J.R. Hughes, L.P. Franca, and G.M. Hulbert. 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. *Computer Methods in Applied Mechanics and Engineering*, 65:85–96, 1987.

[107] T.J.R. Hughes, L.P. Franca, and G.M. Hulbert. A new finite element formulation for computational fluid dynamics: VIII. The Galerkin / least-squares method for advective-diffusive equations. *Computer Methods in Applied Mechanics and Engineering*, 73:173–189, 1989.

[108] T.J.R. Hughes, L. Mazzei, and K.E. Jansen. Large eddy simulation and the variational multiscale method. *Computing and Visualization in Science*, 3:47–59, 2000.

[109] T.J.R. Hughes, L. Mazzei, and A.A. Oberai. The multiscale formulation of large eddy simulation: decay of homogeneous isotropic turbulence. *Physics of Fluids*, 13–2, 2001.

[110] T.J.R. Hughes, A.A. Oberai, and L. Mazzei. Large eddy simulation of turbulent channel flows by the variational multiscale method. *Physics of Fluids*, 13–6, 2001.

[111] T.J.R. Hughes, G. Scovazzi, P. Bochev, and A. Buffa. A multiscale discontinuous Galerkin method with the computational structure of a continuous Galerkin method. *Computer Methods in Applied Mechanics and Engineering*, submitted.

[112] W. Hundsdorfer and J.G. Verwer. *Numerical Solution of Time-Dependent Advection-Diffusion-Reaction Equations*. Springer, 2003.

[113] B. Irons and R.C.Tuck. A version of the Aitken accelerator for computer implementation. *International Journal for Numerical Methods in Engineering*, 1:275–277, 1969.

[114] F. Jauberteau, C. Rosier, and R. Temam. A nonlinear Galerkin method for the Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 80:245–260, 1990.

[115] C.B. Jensen and T. Kvamsdal. Computational methods for FSI-simulation of slender bridges on high performance computers. In T. Kvamsdal, editor, *Computational Methods for Fluid-Structure Interaction*, pages 31–40, Tapir Forlag, Trondheim, Norway, 1999.

[116] G.E. Karniadakis, M. Israeli, and S.E. Orszag. High order splitting methods for the incompressible Navier-Stokes equations. *Journal of Computational Physics*, 59:414–443, 1991.

[117] D.W. Kelly, S. Nakazawa, O.C. Zienkiewicz, and J.C. Heinrich. A note on upwinding and anisotropic balancing dissipation in finite element approximations to convective diffusion problems. *International Journal for Numerical Methods in Engineering*, 15:1705–1711, 1980.

[118] J. Kim and P. Moin. Application of the fractional step method to incompressible Navier-Stokes equations. *Journal of Computational Physics*, 59:308–323, 1985.

[119] O. Ladyzhenskaya. New equations for the description of motion of viscous incompressible fluids and solvability in the large of boundary value problems for them. *Proc. Steklov Inst. Math.*, 102:95–118, 1967.

[120] O. Ladyzhenskaya. *The mathematical theory of viscous incompressible flow*. Gordon and Breach, New York, 1969.

[121] A. Larsen and A.S. Jacobsen. Aerodynamic design of the great belt east bridge. In A. Larsen, editor, *Aerodynamics of Large Bridges*. Balkema, Rotterdam, Netherlands.

[122] J. Leray. Essai sur les mouvements d'un liquide visqueux emplissant l'espace. *Acta Mathematica*, 63:193–248, 1934.

[123] J. Li. A Dual-Primal FETI for incompressible Stokes equations. Technical report 816, Courant Institute of Mathematical Sciences, 2001.

[124] J.L. Lions. Sur certaines équations paraboliques non linéaires. *Bulletin de la Société Mathématique de France*, 93:155–175, 1965.

[125] J.L. Lions. *Quelques méthodes de résolution des problémes aux limites non linéaires*. Ed. Dunod, 1969.

[126] M. Marion and R. Temam. Nonlinear Galerkin methods. *SIAM Journal on Numerical Analysis*, 26:1139–157, 1989.

[127] H.G. Matthies and J. Stein. Partitioned but strongly coupled iteration schemes for nonlinear fluid-structure interaction. *Computers & Structures*, 80:1991–1999, 2002.

[128] H.G. Matthies and J. Stein. Partitioned strong coupled algorithms for fluid-structure interaction. *Computers & Structures*, 81:805–812, 2003.

[129] D.P. Mok and W.A. Wall. Partitioned analysis schemes for transient interaction of incompressible flows and nonlinear flexible structures. In *Trends in computational structural mechanics (W.A. Wall, K.U. Bletzinger and K. Schweizerhof, Eds.)*, CIMNE, Barcelona, Spain, 2001.

[130] F. Nobile. *Numerical Approximation of Fluid-Structure Interaction problems with application to Haemodynamics*. PhD thesis, École Polytechnique Fédérale de Lausanne, 2001.

[131] J.T. Oden and L.Z. Demkowicz. *Applied Funtional Analysis*. CRC Press, 1996.

[132] S.A. Orszag, M. Israeli, and M. Deville. Boundary conditions for incompressible flows. *J. Sci. Comput.*, 1:75–111, 1986.

[133] J.B. Perot. An analysis of the fractional step method. *Journal of Computational Physics*, 108:51–58, 1993.

[134] R. Pierre. Optimal selection of the bubble function in the stabilization of the P1-P1 element for the Stokes problem. *SIAM Journal on Numerical Analysis*, 32:1210–1224, 1995.

[135] S. Piperno and C. Farhat. Partitioned prodecures for the transient solution of coupled aeroelastic problems-Part II: energy transfer analysis and three-dimensional applications. *Computer Methods in Applied Mechanics and Engineering*, 190:3147–3170, 2001.

[136] A. Prohl. *Projection and Quasi-Compressibility Methods for Solving the Incompressible Navier-Stokes Equations*. B.G. Teubner Stuttgart, 1997.

[137] A. Quarteroni, F. Saleri, and A. Veneziani. Factorization methods for the numerical approximation of Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 188:505–526, 2000.

[138] A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations*. Oxford Science Publications, 1999.

[139] P. Raback, J. Ruokolainen, M. Lyly, and E. Jarvinen. Fluid-structure interaction boundary conditions by artificial compressibility. In *ECCOMAS Computational Fluid Dynamics Conference*, Swansea, Wales, UK, 2001.

[140] R. Rannacher. *On Chorin's projection method for incompressible Navier-Stokes equations,* Lecture Notes in Mathematics, volume 1530, pages 167–183. Springer, Berlin, 1992.

[141] R. Rossi. *Light weight Structures: Structural Analysis and Coupling Issues*. PhD thesis, Universitá di Bologna, 2005.

[142] A. Russo. Bubble stabilization of finite element methods for the linearized incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 132:335–343, 1996.

[143] R.H. Scanlan and J.J. Tomko. Airfoil and bridge deck flutter derivatives. *Journal of Engineering Mechanics division ASCE 97 (EM6)*, pages 1717–1737, 1971.

[144] R.P. Selvam and S. Govindaswamy. Aeroelasic analysis of bridge girder section using computer modelling. Technical Report, University of Arkansas, 2001.

[145] F. Shakib and T.J.R. Hughes. A new finite element formulation for computational fluid dynamics: IX. Fourier analysis of space-time Galerkin/least-squares algorithms. *Computer Methods in Applied Mechanics and Engineering*, 87:35–58, 1991.

[146] J. Shen. On error estimates for some higher order projection and penalty-projection methods for Navier-Stokes equations. *Numerische Mathematik*, 62:49–73, 1992.

[147] J. Shen. On error estimates of projection methods for Navier-Stokes equations: first order squemes. *SIAM Journal on Numerical Analysis*, 29:57–77, 1992.

[148] J. Shen. A remark on the projection-3 method. *International Journal for Numerical Methods in Fluids*, 16:249–253, 1993.

[149] J. Shen. Remarks on the pressure error estimates for the projection methods. *Numerische Mathematik*, 67:513–520, 1994.

[150] J. Shen. On error estimates of the projection methods for the Navier-Stokes equations: second-order squemes. *Mathematics of Computation*, 65:1039–1065, 1996.

[151] J.C. Simo and F. Armero. Unconditional stability and long term behavior of transient algorithms for the incompressible Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 111:111–154, 1994.

[152] J.L. Steger and J.A. Benek. On the use of composite grid schemes in computational aerodynamics. *Computer Methods in Applied Mechanics and Engineering*, 64:301–320, 1987.

[153] G. Strang and J. Fix. *An Analysis of the Finite Element Method*. Prentice Hall, Englewood Cliffs, 1973.

[154] P. Le Tallec and J. Mouro. Fluid structure interaction with large structural displacements. *Computer Methods in Applied Mechanics and Engineering*, 190:3039–3067, 2001.

[155] R. Temam. Sur la stabilité et la convergence de la méthode des pas fractionaires. *Ann. Mat. Pura Appl.*, LXXIV:191–380, 1968.

[156] R. Temam. Une méthode d'approximations de la solution des equations de Navier–Stokes. *Bull. Soc. Math. France*, 98:115–152, 1968.

[157] R. Temam. Sur l'approximation de la solution des équations de Navier–Stokes par la méthode des pas fractionaires (I). *Archives for Rational Mechanics and Analysis*, 32:135–153, 1969.

[158] R. Temam. Sur l'approximation de la solution des équations de Navier–Stokes par la méthode des pas fractionaires (II). *Archives for Rational Mechanics and Analysis*, 33:377–385, 1969.

[159] R. Temam. *Navier-Stokes equations*. North-Holland, 1984.

[160] R. Temam. Remark on the pressure boundary condition for the projection method. *Theoretical Computational Fluid Dynamics*, 3:181–184, 1991.

[161] L.J.P. Timmermans, P.D. Minev, and F.N. Van de Vosse. An approximate projection scheme for incompressible flow using espectral elements. *International Journal for Numerical Methods in Fluids*, 22:673–688, 1996.

[162] J. van Kan. A second–order accurate pressure correction scheme for viscous incompressible flow. *SIAM Journal of Sci. Stat. Comp.*, 7:870–891, 1986.

[163] J. von Neumann and R.D. Richtmyer. A method for the numerical calculation of hydrodynamical shocks. *J. Appl. Phys.*, 21:232, 1950.