UNIVERSITAT POLITÈCNICA DE CATALUNYA

PhD Thesis

STABILIZED FINITE ELEMENT FORMULATIONS FOR SOLVING INCOMPRESSIBLE MAGNETOHYDRODYNAMICS

by **Ramon Planas Badenas**

Barcelona, September 2013

Stabilized finite element formulations for solving incompressible magnetohydrodynamics

Author: **Ramon Planas**

Advisors: **Santiago Badia, Ramon Codina**

Escola Tècnica Superior d'Enginyers de Camins, Canals i Ports

Universitat Politècnica de Catalunya

September 2013

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola de Doctorat

Acta de qualificació de tesi doctoral

Curs acadèmic:

Nom i cognoms

Programa de doctorat

Unitat estructural responsable del programa

Resolució del Tribunal

Reunit el Tribunal designat a l'efecte, el doctorand / la doctoranda exposa el tema de la seva tesi doctoral titulada

_____

_____.

Acabada la lectura i després de donar resposta a les qüestions formulades pels membres titulars del tribunal, aquest atorga la qualificació:

☐ APTA/E ☐ NO APTA/E

(Nom, cognoms i signatura) President/a	(Nom, cognoms i signatura) Secretari/ària	
(Nom, cognoms i signatura) Vocal	(Nom, cognoms i signatura) Vocal	(Nom, cognoms i signatura) Vocal

_____, _____ d'/de _____ de _____

El resultat de l'escrutini dels vots emesos pels membres titulars del tribunal, efectuat per l'Escola de Doctorat, a instància de la Comissió de Doctorat de la UPC, atorga la MENCIÓ CUM LAUDE:

☐ SÍ ☐ NO

(Nom, cognoms i signatura) Presidenta de la Comissió de Doctorat	(Nom, cognoms i signatura) Secretària de la Comissió de Doctorat

Barcelona, _____ d'/de _____ de _____

Acknowledgements

I would like to thank my co-advisors, Santi Badia and Ramon Codina, for their support, advice and never-ending patience to solve my doubts. It's been a great experience to work with them, who have such a vast knowledge of this field.

I also really appreciate the advice, help and discussions with my co-workers during these years, Javier Principe, Alberto Martín, Rubén Otín, Elisabet Mas de les Valls, Oriol Colomés, Alba Hierro, Marc Olm, Joan Baiges, Vladimir Jazarevic, and the rest of people at CIMNE and RMEE-UPC.

The financial support received from the *Universitat Politècnica de Catalunya (UPC)* and from the *Col.legi d'Enginyers de Camins, Canals i Ports de Catalunya* is gratefully acknowledged.

Abstract

Magnetohydrodynamics (MHD) is the physics branch that studies electrically conducting fluids under external magnetic fields. This thesis deals with the numerical approximation using stabilized finite element methods of two different formulations to model incompressible MHD, namely the resistive and inductionless MHD problems. Further, the linear systems of equations resulting from the application of these discrete formulations to simulate real cases are typically ill-conditioned and can have as many as 10^6-10^9 degrees of freedom. An efficient and scalable solver strategy is mandatory in these cases.

On one hand, a new stabilized finite element formulation for the approximation of the resistive magnetohydrodynamics equations has been proposed. The novelty of this formulation with respect to existing ones is that it always converges to the physical solution, even when it is singular, which has been proved through a detailed stability and convergence analysis of the formulation. Moreover, it is inferred from the convergence analysis that a particular type of meshes with a macro-element structure is needed, which can be easily obtained after a straight modification of any original mesh. Finally, different operator splitting schemes have been proposed for solving the transient incompressible resistive MHD system that are unconditionally stable. Two levels of splitting have been considered. On the first level, the segregation of the Lagrange multipliers, the fluid pressure and the magnetic pseudo-pressure, from the vectorial fields computation is achieved. On the second level, the fluid velocity and induction fields are also decoupled. This way, the fully coupled indefinite multiphysics system is transformed into smaller uncoupled one-physics problems.

On the other hand, a stabilized formulation to solve the inductionless magnetohydrodynamic problem using the finite element method is presented. The inductionless MHD problem models the flow of an electrically charged fluid under the influence of an external magnetic field where the magnetic field induced in the fluid by the currents is negligible with respect to the external one. This system of partial differential equations is strongly coupled and highly nonlinear for real cases of interest. Therefore, solving the multiphysics linear systems of equations resulting from the discretization of these equations with finite element methods is a very challenging task which requires efficient and scalable preconditioners. A new family of recursive block LU preconditioners has been designed to improve the convergence of iterative solvers for this problem. These preconditioners are obtained after splitting the fully coupled matrix into one-physics problems for every variable (velocity, pressure, current density and electric potential) that can be optimally solved, e.g. using preconditioned domain decomposition algorithms. Furthermore, these ideas have been extended for developing recursive block LU preconditioners for the thermally coupled inductionless MHD problem.

Resum

La magnetohidrodinàmica (MHD) és la branca de la Física que estudia el moviment de fluids elèctricament conductors que es troben sotmesos a camps magnètics externs. Aquesta tesi tracta de l'aproximació numèrica amb mètodes d'element finits estabilitzats de dues formulacions per modelar el problema de la MHD incompressible, com són la MHD resistiva i la MHD sense inducció. A més a més, els sistemes lineals d'equacions que resulten de l'aplicació d'aquestes formulacions discretes per a simular casos reals solen ser mal condicionats i poden arribar a comprendre entre 10^6-10^9 graus de llibertat. La ressolució d'aquests sistemes lineals d'equacions necessita obligatòriament una estratègia eficient i escalable.

Per una banda, s'ha proposat una nova formulació estabilitzada d'elements finits per a l'aproximació de les equacions de la MHD resistiva. La novetat d'aquesta formulació resideix en el fet que sempre convergeix a la solució física del problema, fins i tot quan és singular, cosa que s'ha demostrat a través de les anàlisis d'estabilitat i convergència del mètode. A més, l'anàlisi de convergència mostra la necessitat de fer servir un tipus particular de malles amb una estructura de macro-element, que es poden obtenir fàcilment a partir de qualsevol malla original. Finalment, s'han proposat diferents esquemes de segregació incondicionalment estables per a resoldre el problema de la MHD resistiva transitòria. S'han considerat dos nivells de segregació. El primer nivell permet la segregació dels multiplicadors de Lagrange, la pressió i la pseudo-pressió magnètica, del càlcul dels camps vectorials. En el segon nivell, es desacobla el càlcul dels camps vectorials, la velocitat i l'inducció magnètica. D'aquesta manera, el sistema de multifísica totalment acoblat es transforma en problemes d'una física desacoblats i més petits.

D'altra banda, també s'ha presentat una formulació estabilitzada per al problema de la MHD sense inducció. Aquest problema permet modelar el flux d'un fluid carregat elèctricament sota l'efecte d'un camp magnètic extern on el camp magnètic induït al fluid pels corrents és negligible respecte del camp magnètic extern. Aquest sistema d'equacions és fortament acoblat i altament no lineal. Llavors, resoldre els sistemes d'equacions lineals que resulten de la discretització amb mètodes d'elements finits d'aquestes equacions és un gran repte que necessita precondicionadors eficients i escalables. S'ha desenvolupat una nova família de precondicionadors LU per blocs recursius per millorar la convergència dels mètodes iteratius per a resoldre aquest problema. Aquests precondicionadors permeten la segregació de la matriu totalment acoblada en problemes d'una física per a cada una de les variables del problema (velocitat, pressió, densitat de corrent i potencial elèctric) que es poden resoldre de forma òptima, per exemple, fent servir algorismes de descomposició de domini precondicionats. A més a més, aquestes idees s'han extès per a desenvolupar precondicionadors LU per blocs recursius per al problema de la MHD sense inducció amb acoblament tèrmic.

Contents

Chapter 1

Introduction

Magnetohydrodynamics (MHD) is the physics branch that studies the motion of electrically conducting fluids under external magnetic fields. This field models a very wide range of examples, from natural processes such as geophysics or astrophysics to industrial applications like MHD pumps, steel casting processes or crystal growth devices. Furthermore, in recent years the interest in numerical tools to solve this problem has increased significantly because of nuclear fusion. It is considered to be a safe energy source but an efficient industrial generation depends on the development of the associated technology. The physical phenomena that take place in fusion reactors are extremely complex. They comprise several areas from physics, such as fluid mechanics, electromagnetics, thermal radiation or plasma physics.

This work aims to develop numerical algorithms based on the finite element method to solve the MHD equations. There exist mainly three modelling approaches to solve a MHD problem depending on the level of complexity. First, we can use asymptotic analyses for high Hartmann numbers. These models are restricted to steady flows and are not accurate. Increasing the complexity, we can apply the inductionless hypothesis, which states that the magnetic field induced by the fluid motion is negligible compared to the external magnetic field. This assumption can be used when the magnetic Reynolds number is very low. The third option is to solve the resistive MHD system, coupling the Navier-Stokes equations from fluid mechanics and Maxwell's equations from electromagnetism where the problem is assumed to be quasi-static, i.e., the so-called displacement currents are neglected.

This thesis is organized as follows. In Chapter 2, a stabilized finite element formulation to solve the resistive MHD system is presented. This system consists of the Navier-Stokes equations from fluid mechanics coupled with Maxwell's equations from electromagnetism. This set of equations can be applied to model general MHD problems, even when the magnetic field induced by the moving fluid is not negligible compared to the externally applied magnetic field. The main feature of the proposed formulation resides in the fact that it always converges to the physical solution of the problem, even if it is non-smooth (singular). This is achieved because the stabilized formulation is able to mimic the correct functional setting of the continuous system of equations.

The stability and convergence analysis of the stabilized formulation for the resistive MHD problem is presented in detail in Chapter 3. The analysis proves the convergence to nonsmooth solutions. Moreover, the need of a particular type of meshes with a macro-

element structure appears from the mentioned convergence analysis. This macro-element structure can be easily obtained from any original mesh in both 2D and 3D cases.

The solution of the linear system of equations arising from applying the stabilized FE formulations to solve engineering or physics problems is a very challenging task. One of the most used techniques in fluid mechanics and by extension in the MHD community consists of operator splitting schemes, also known as fractional step methods. These algorithms aim to uncouple the computation of the several physical variables in order to solve smaller and easier linear systems. In Chapter 4, different splitting procedures for the transient incompressible resistive MHD equations are proposed. The key feature of the developed algorithms is a two-level splitting. On one level, the pressure and magnetic pseudo-pressure are segregated from the vectorial fields computation. On the second level, the velocity and magnetic induction fields are also decoupled. This way, the solution of the fully coupled problem is reduced to solve four one-physics problems where each one of the physical variables are computed uncoupled from the others.

Chapter 5 deals with the numerical approximation of the inductionless MHD problem. As previously stated, the inductionless hypothesis can be applied to the general MHD system when the magnetic field induced by the currents in the fluid is negligible with respect to the external magnetic field. This set of equations can be used to model several industrial processes, such as test blanket modules (TBMs) in nuclear fusion reactors. TBMs will be one of the key components of ITER (International Termonuclear Experimental Reactor), that should demonstrate the scientific reliability of fusion (see www.iter.org for more details). Each of these breeding blankets is designed in a modular shape performing a triple function: 1) heat power extraction from the plasma, 2) tritium generation (breeding) and 3) shielding of the magnets from neutron and gamma radiation. The breeding material used is the eutectic lead-lithium liquid metal. In normal regimes, this liquid metal flow can be modeled by the inductionless MHD equations. The aim to design effective TBMs and the lack of experimental data has increased the demand of numerical methods for this system of equations.

Chapter 6 is devoted to develop new recursive block LU preconditioners for the solution of the thermally coupled incompressible inductionless MHD problem. The application of segregation methods to divide the fully-coupled system of equations into one-physics problems for every variable introduces a splitting error in the solution. Therefore, there exist other approaches to solve the multi-physics monolithic problem to avoid them, based on preconditioned Krylov iterative solvers, such as CG or GMRES. The key ingredient resides in using an efficient and scalable preconditioner to improve the condition number of the system matrix and therefore, increase the convergence rate of the iterative solver. A very interesting family of preconditioners for multi-physics problems are block LU precondtioners, which allow to introduce the same ideas from splitting algorithms at the preconditioner level. This way, the only solvers that appear when applying the preconditioner inverse consist of one-physics solvers for every physical variable. However, it is very important to note that the iterative solver is applied to the fully-coupled monolithic system of equations and therefore, no splitting error is introduced in the solution.

This thesis is closed in Chapter 7 with some conclusions. Furthermore, we summarize some possible future lines of research.

Chapters from 2 to 6 are self contained even if this implies the repetition of some

information. This is due to the fact that each one of these chapters is based on the following scientific papers:

- Chapter 2: On an unconditionally convergent stabilized finite element approximation of resistive magnetohydrodynamics. S. Badia, R. Codina and R. Planas. *Journal of Computational Physics*, 234:399-416, 2013.

- Chapter 3: Analysis of an unconditionally convergent stabilized finite element formulation for incompressible magnetohydrodynamics. S. Badia, R. Codina and R. Planas. *Submitted*, 2013.

- Chapter 4: Unconditionally stable operator splitting algorithms for the incompressible magnetohydrodynamics system discretized by a stabilized finite element formulation based on projections. S. Badia, R. Planas and J.V. Gutiérrez-Santacreu. *International Journal for Numerical Methods in Engineering*, 93:302-328, 2013.

- Chapter 5: Approximation of the inductionless MHD problem using a stabilized finite element method. R. Planas, S. Badia and R. Codina. *Journal of Computational Physics*, 230:2977-2996, 2011.

- Chapter 6: Block recursive LU preconditioners for the thermally coupled incompressible inductionless MHD problem. S. Badia, A. F. Martín and R. Planas. *Submitted*, 2013.

Chapter 2

Resistive MHD problem

In this chapter, a new stabilized finite element formulation for the approximation of the resistive magnetohydrodynamics equations is proposed. The novelty of this formulation is the fact that it always converges to the physical solution, even for singular ones. A detailed set of numerical experiments have been performed in order to validate our approach.

2.1 Introduction

2.1.1 State-of-the-art

In this work, we propose a novel numerical formulation for the approximation of the incompressible visco-resistive magnetohydrodynamics (MHD) system that models incompressible viscous and electrically conducting fluids under the influence of electromagnetic fields (see [61]). Examples of such fluids include liquid metals and plasmas. The numerical approximation of the associated set of partial differential equations (PDEs) is of paramount importance in fusion energy, since it allows us to model liquid metal cooling and He-Tr extraction systems in fusion reactors (the so-called breeding blankets) as well as the plasma confinement in the core [89].

Many conforming numerical approximations to this problem have been proposed so far. There are different equivalent formulations of the continuous magnetic subproblem, namely saddle-point and (weighted) exact penalty formulations (see [111] and [4, 56, 58, 77, 78] respectively). The first one leads to a double-saddle-point formulation for the MHD system. A Galerkin finite element (FE) approximation of the resulting problem has been proposed and analyzed by Schötzau in [111]. It is well-known that saddle-point formulations require to choose particular mixed FE spaces satisfying discrete versions of the so-called inf-sup conditions (see e.g. [34]). It complicates the implementation issues, e.g. the database structure, the computation of the coupling terms and the graphs needed for the compressed storage of the system matrix (see Section 2.1.2). Instead, a weighted exact penalty formulation has been used in [77]. This formulation allows to simplify the aforementioned implementation issues but introduces a new complication, the definition of the weight function (see [58]); it requires an *a priori* knowledge of the exact solution –where the singularities are placed– and the final expressions are hard to integrate accurately using numerical integration. Since the resistive

MHD system loses coercivity as the Reynolds and magnetic Reynolds numbers increase, i.e. convection-type terms become dominant, a mixed FE formulation of the problem is unstable unless the mesh size is sufficientely refined, which is impractical.

Alternative formulations have been proposed for a regularized version of the system, based on an exact penalty formulation.[1] Under some assumptions on the computational domain Ω, the magnetic field is smoother than for the original problem (see e.g. [73]). An inf-sup stable Galerkin FE formulation for the flow sub-problem and the regularized exact penalty formulation for the magnetic sub-problem has been proposed in [76], whereas a stabilized FE version of this formulation can be found in [72].

Non-conforming approximations of discontinuous Galerkin type have been designed in [81]. These methods have good numerical properties, but the increase in CPU cost –degrees of freedom– of these formulations (with respect to conforming formulations) is severe for realistic large-scale applications.

2.1.2 Motivation of the work

The multi-physics nature of the MHD system, and the fact that both sub-problems have a saddle-point structure (before modifications), makes the numerical approximation of this PDE system a challenging task. Further, the extremely complex nature of the phenomena that can be modeled [40, 41], requires the development of low-storage algorithms that allow an easy-to-optimize implementation suitable for massive parallelization.

A straightforward Galerkin FE approximation of the original problem would require very specific FE spaces for the different unknowns in order to satisfy the corresponding inf-sup conditions, e.g. the MINI element could be used for the fluid sub-problem whereas weakly-solenoidal Nédélec bases are needed for the magnetic sub-problem [99, 100]. Clearly, this approach introduces implementational complications in comparison to a straightforward equal interpolation of all the vectorial components and scalar unknowns, e.g. database structures or integration of coupling terms. More important, for $\mathrm{n_{unk}}$ unknowns,[2] we require (in general) $(\mathrm{n_{unk}})^2$ graphs to define the sparse structures of the $(\mathrm{n_{unk}})^2$ block matrices. The optimization of a Galerkin-type MHD implementation is a formidable task. For these reasons, the use of stabilized FE formulations that allow equal interpolation for all the unknowns becomes a very appealing discretization technique for multi-physics applications (see [14]). This way, with only one graph –the mesh-graph– we are able to define the sparse structure of the matrix.

However, implementational issues are not the only (or even the main) justification to use stabilized formulations instead of mixed ones. The additional stabilization terms not only avoid the fulfillment of the inf-sup conditions but add streamline-diffusion to reduce oscillations in dominant convection flows [35].[3] Furthermore, the introduction of stabilization terms changes the nature of the problem, and subsequently the system

[1] All these methods must be used with caution, since they converge to spurious solutions when the exact magnetic field is not smooth.

[2] The MHD system at hand involves fluid velocity and pressure as well as magnetic field for the penalty formulation ($\mathrm{n_{unk}} = 3$) and an additional magnetic pseudo-pressure for the double-saddle-point structure ($\mathrm{n_{unk}} = 4$). Herein we use the concept of *magnetic pseudo-pressure* to denote the Lagrange multiplier that is introduced to enforce the solenoidal constraint on the magnetic field. Do not confuse it with the classical notion of magnetic pressure in MHD (see [61]).

[3] Most of the applications of interest are in this regime.

matrix. Whereas the system matrix obtained by using mixed FEs is indefinite, with null diagonal-blocks, the one from the stabilized FE formulation is positive definite. It simplifies the numerical linear algebra strategy to solve the final linear system in an efficient (optimal) way (see [112] for a detailed discussion).

With regard to parallelization, the possibility to work with the mesh-graph only simplifies the required graph-partitioning needed in domain decomposition approaches [118]. Further, the ease in the definition of effective preconditioners also helps to attain algorithmic scalability [112].

Some stabilized FE formulations have been proposed so far for resistive MHD [23, 24, 53, 54, 72, 73, 112]. These formulations share the benefits listed above but they are based on the regular functional setting of the problem, and so, restricted to smooth or convex domains (see [58]). They are accurate for regular magnetic solutions but tend to spurious (unphysical) solutions otherwise (see Section 2.3 for further discussion). The objective of this work is to propose a novel stabilized FE formulation that always converges to the exact (physical) solution.

More specifically, the objective of this work is to design and analyze a numerical FE formulation of the MHD problem with the following features:

- always converges to the physical solution, even when it is a singular solution,
- allows equal Lagrangian interpolation of every component of the vectorial fields as well as the scalar fields,[4]
- allows for arbitrary-order interpolation,
- introduces effective numerical stability in convection-dominated regimes,
- does not require any *a priori* information of the solution, i.e. it is a fully automatic approach,

always keeping optimal *a priori* error estimates for smooth solutions.

The outline of this chapter is the following. First, the MHD problem of interest is stated in Section 2.2. Some existing FE discretizations (in space) are detailed in Section 2.3 as well as some notation. The linearization and time integration of the resulting problem is presented in Section 2.4. Section 2.5 is devoted to the formulation we propose. A complete set of numerical experiments that validate the formulation are presented in Section 2.6. We finish the chapter by drawing some conclusions in Section 2.7.

2.2 Problem statement

2.2.1 The strong form

The incompressible visco-resistive MHD system of partial differential equations consists of the Navier-Stokes equations coupled to the (simplified) Maxwell equations via the

[4]This feature implies that only one graph is enough for the definition of the sparse structure of the system matrix.

Lorentz force. It reads as follows: find a velocity field $\mathbf{u}(\mathbf{x},t)$, a (dynamic) pressure $\tilde{p}(\mathbf{x},t)$ and an induced magnetic field $\mathbf{b}(\mathbf{x},t)$ such that

$$\begin{aligned} \rho\partial_t\mathbf{u} + \rho\mathbf{u}\cdot\nabla\mathbf{u} - \mu_{\mathrm{f}}\Delta\mathbf{u} + \nabla\tilde{p} - \frac{1}{\mu_{\mathrm{m}}}(\nabla\times\mathbf{b})\times\mathbf{b} &= \mathbf{f}_{\mathrm{u}}, && \text{(2.1a)}\\ \nabla\cdot\mathbf{u} &= 0, && \text{(2.1b)}\\ \partial_t\mathbf{b} + \frac{1}{\mu_{\mathrm{m}}}\nabla\times(\frac{1}{\sigma}\nabla\times\mathbf{b}) - \nabla\times(\mathbf{u}\times\mathbf{b}) &= \mathbf{f}_{\mathrm{b}}, && \text{(2.1c)}\\ \nabla\cdot\mathbf{b} &= 0, && \text{(2.1d)} \end{aligned}$$

in $(\mathbf{x},t)\in\Omega\times(0,T)$, where ∂_t stands for the partial time derivative (d_t will be used for the total one), $\Omega\subset\mathbb{R}^d$ is the spatial open bounded domain filled by the fluid (assumed polyhedral in the finite element approximation), d being the space dimension, and $(0,T)$ is the time interval of interest. $\mathbf{f}_{\mathrm{u}}$ and $\mathbf{f}_{\mathrm{b}}$ are the forcing terms, $\mathbf{f}_{\mathrm{b}}$ being solenoidal. With regard to the physical parameters that describe the fluid, ρ is its density, μ_{f} the fluid viscosity, μ_{m} the magnetic permeability and σ the electric conductivity. In this work, we consider all physical properties constant. These equations must be supplemented with appropriate boundary and initial conditions. The initial conditions are:

$$\mathbf{u}(\mathbf{x},0) = \mathbf{u}_0(\mathbf{x}), \qquad \mathbf{b}(\mathbf{x},0) = \mathbf{b}_0(\mathbf{x}),$$

where $\mathbf{b}_0$ must be solenoidal. This restriction can be weakened for $\mathbf{u}_0$, but it introduces a singularity at $t=0$ (see [117]).

We can easily check that this system is over-constrained, since no Lagrange multiplier has been used to enforce the null-divergence restriction over the magnetic field $\mathbf{b}$. However, the problem has at least one solution; taking the divergence of (2.1c), we obtain $\partial_t(\nabla\cdot\mathbf{b})=0$. This fact, together with the solenoidal initial condition, amounts to say that (2.1d) is satisfied at all times. Unfortunately, this procedure cannot be used for the discretized system when Lagrangian finite elements are used. In this case, it is more suitable to explicitly enforce (2.1d) via a Lagrange multiplier, the magnetic pseudo-pressure $r(\mathbf{x},t)$. Let us introduce the following augmented formulation of (2.1), re-scaled in a more suitable way for the subsequent exposition:

$$\begin{aligned} \partial_t\mathbf{u} + \mathbf{u}\cdot\nabla\mathbf{u} - \nu\Delta\mathbf{u} + \nabla p - (\nabla\times\mathbf{b})\times\varrho\mathbf{b} &= \mathbf{f}_{\mathrm{u}}, && \text{(2.2a)}\\ \nabla\cdot\mathbf{u} &= 0, && \text{(2.2b)}\\ \varrho\partial_t\mathbf{b} + \lambda\nabla\times(\nabla\times\mathbf{b}) + \nabla r - \nabla\times(\mathbf{u}\times\varrho\mathbf{b}) &= \mathbf{f}_{\mathrm{b}}, && \text{(2.2c)}\\ \nabla\cdot\mathbf{b} &= 0, && \text{(2.2d)} \end{aligned}$$

where $\nu := \mu_{\mathrm{f}}\rho^{-1}$, $\varrho := (\rho\mu_{\mathrm{m}})^{-1}$ and $\lambda := (\rho\mu_{\mathrm{m}}^2\sigma)^{-1}$; $p := \rho^{-1}\tilde{p}$ is the kinematic pressure. $\mathbf{f}_{\mathrm{u}}$ and $\mathbf{f}_{\mathrm{b}}$ have been re-defined accordingly.

In order to introduce the boundary conditions, let us consider two disjoint partitions of the domain boundary $\Gamma\equiv\partial\Omega$:

$$\Gamma = \Gamma_{\mathrm{f,e}}\cup\Gamma_{\mathrm{f,n}}, \qquad \Gamma = \Gamma_{\mathrm{m,e}}\cup\Gamma_{\mathrm{m,n}},$$

where the first subscript denotes the subproblem (f for fluid and m for magnetic) and the second one the type of boundary condition (e for essential and n for natural). Then, the fluid sub-problem is supplemented with the standard boundary conditions:

$$\mathbf{u} = \mathbf{u}_\Gamma \quad \text{on } \Gamma_{\mathrm{f,e}}, \qquad -p\mathbf{n} + \nu\mathbf{n}\cdot\nabla\mathbf{u} = \boldsymbol{\sigma}_{n,\Gamma} \quad \text{on } \Gamma_{\mathrm{f,n}},$$

where $\mathbf{u}_\Gamma(\mathbf{x},t)$ and $\boldsymbol{\sigma}_{n,\Gamma}(\mathbf{x},t)$ are the trace and normal stress prescribed; $\mathbf{n}(\mathbf{x})$ denotes the normal vector on Γ pointing outwards from Ω. With regard to the magnetic sub-problem, we consider the set of ideal boundary conditions:

$$\mathbf{n}\times\mathbf{b}=\mathbf{n}\times\mathbf{b}_\Gamma,\quad r=0\quad\text{on }\Gamma_{\mathrm{m,e}},\qquad \mathbf{n}\cdot\mathbf{b}=\mathbf{n}\cdot\mathbf{b}_\Gamma,\quad \mathbf{n}\times(\nabla\times\mathbf{b})=\mathbf{J}_\Gamma\quad\text{on }\Gamma_{\mathrm{m,n}},$$

where $\mathbf{n}\cdot\mathbf{b}_\Gamma$ and $\mathbf{n}\times\mathbf{b}_\Gamma$ are the normal and tangential traces to be prescribed; clearly, $\mathbf{J}_\Gamma\cdot\mathbf{n}$ must vanish.

2.2.2 The weak form

Notation

Let us introduce some notation to set up the weak form of the problem. As usual, Sobolev spaces of functions whose derivatives of order up to m belong to $L^2(\Omega)$ are denoted by $H^m(\Omega)$; $H^1_0(\Omega)$ is the subspace of $H^1(\Omega)$ of functions vanishing on $\partial\Omega$. The space of vector functions with components in $L^2(\Omega)$ and with divergence also in $L^2(\Omega)$ is denoted $H(\mathrm{div};\Omega)$; if the components are $L^2(\Omega)$ and the curl is in $L^2(\Omega)^d$ the space is denoted $H(\mathbf{curl};\Omega)$. $H(\mathrm{div}\,0;\Omega)$ is the subspace of $L^2(\Omega)^d$ of divergence free vector functions.

The inner product of $f,g\in L^2(\Omega)$ is represented as (f,g), whereas $\langle f,g\rangle$ is used to denote the integral $\int_\Omega fg$ whenever it makes sense; this in particular applies for the duality between $H^1_0(\Omega)$ and its topological dual $H^{-1}(\Omega)$. The same notation is used for both scalar and vector valued functions.

Given a normed functional space X, its norm is written as $\|\cdot\|_X$, with the abbreviations $\|\cdot\|_{L^2(\Omega)}\equiv\|\cdot\|$, $\|\cdot\|_{H^m(\Omega)}\equiv\|\cdot\|_m$, $\|\cdot\|_{H^{-1}(\Omega)}\equiv\|\cdot\|_{-1}$, $\|\cdot\|_{H(\mathbf{curl};\Omega)}\equiv\|\cdot\|_{\mathbf{curl}}$. Functions defined on the time interval $(0,T)$ whose X-norm is in $L^2(0,T)$ are denoted by $L^2(0,T;X)$, whereas distributions in $(0,T)$ whose X-norm is bounded are represented by $\mathcal{D}'(0,T;X)$.

Finally, the symbol $\lesssim$ is used to denote $\leq$ up to positive constants.

A double saddle-point formulation

Let us consider the functional setting in which the system of equations (2.2) is well-posed. For the sake of clarity, we will consider homogeneous essential boundary conditions; in any case, the extension to the most general case is standard. We introduce the vectorial functional spaces:

$$\begin{aligned}\boldsymbol{V}&=\{\mathbf{v}\in H^1(\Omega)^d\,\text{such that}\,\mathbf{v}=\mathbf{0}\,\text{on}\,\Gamma\},\\ \boldsymbol{C}&=\{\mathbf{c}\in H(\mathbf{curl};\Omega)\,\text{such that}\,\mathbf{n}\times\mathbf{c}=\mathbf{0}\,\text{on}\,\Gamma\},\end{aligned}\tag{2.3}$$

for the velocity and magnetic field functions, respectively. Further, the space for fluid pressures is $Q\equiv L^2_0(\Omega)$ and the one for magnetic pseudo-pressures $S\equiv H^1_0(\Omega)$. Now, we can state the transient MHD problem at hand in its weak form as follows: find $\mathbf{u}\in L^2(0,T;\boldsymbol{V})$, $\mathbf{b}\in L^2(0,T;\boldsymbol{C})$, $p\in\mathcal{D}'(0,T;Q)$ and $r\in\mathcal{D}'(0,T;S)$ such that

$$\mathrm{d}_t(\mathbf{u},\mathbf{v})+\langle\mathbf{u}\cdot\nabla\mathbf{u},\mathbf{v}\rangle+(\nu\nabla\mathbf{u},\nabla\mathbf{v})-(p,\nabla\cdot\mathbf{v})-\langle(\nabla\times\mathbf{b})\times\varrho\mathbf{b},\mathbf{v}\rangle=\langle\mathbf{f}_\mathrm{u},\mathbf{v}\rangle,\tag{2.4a}$$

$$(q,\nabla\cdot\mathbf{u})=0,\tag{2.4b}$$

$$\mathrm{d}_t\,(\varrho\mathbf{b},\mathbf{c})+(\lambda\nabla\times\mathbf{b},\nabla\times\mathbf{c})-\langle\nabla\times(\mathbf{u}\times\varrho\mathbf{b}),\mathbf{c}\rangle+(\nabla r,\mathbf{c})=\langle\mathbf{f}_\mathrm{b},\mathbf{c}\rangle,\tag{2.4c}$$

$$-(\nabla s,\mathbf{b})=0,\tag{2.4d}$$

for any $(\mathbf{v}, \mathbf{c}, q, s) \in (\boldsymbol{V}, \boldsymbol{C}, Q, S)$ almost everywhere (a.e.) in $(0, T)$. Let us show that $r \equiv 0$ in (2.4). Taking $\mathbf{c} = \nabla r$ (which clearly belongs to $\boldsymbol{C}$) in (2.4c), and using the fact that $\nabla \times \nabla r = \mathbf{0}$ and $\nabla \cdot \mathbf{f}_\mathrm{b} = 0$ a.e. in Ω, we obtain $\|\nabla r\| = 0$. Since r vanishes on $\partial\Omega$, it implies $r \equiv 0$ a.e. in Ω by virtue of Poincaré's inequality. We refer to [78, Propositions 3.4 and 3.5] for the completion of the proof.

Let us re-write system (2.4) in compact manner as:

$$\mathcal{M}((\mathbf{u}, \mathbf{b}), (\mathbf{v}, \mathbf{c})) + \mathcal{A}((\mathbf{u}, \mathbf{b}, p, r), (\mathbf{v}, \mathbf{c}, q, s)) = \langle \mathbf{f}_\mathrm{u}, \mathbf{v} \rangle + \langle \mathbf{f}_\mathrm{b}, \mathbf{c} \rangle, \forall (\mathbf{v}, \mathbf{c}, q, s) \in \boldsymbol{V} \times \boldsymbol{C} \times Q \times S,$$

where the form $\mathcal{M}(\cdot)$ includes all the time derivative terms and $\mathcal{A}(\cdot)$ the rest of left-hand side terms.

The problem is well-posed under the assumption of small data, due to the inf-sup conditions

$$\inf_{q \in Q} \sup_{\mathbf{v} \in \boldsymbol{V}} \frac{(q, \nabla \cdot \mathbf{v})}{\|q\| \|\mathbf{v}\|_1} \geq \beta_\mathrm{f} > 0, \qquad \inf_{s \in S} \sup_{\mathbf{c} \in \boldsymbol{C}} \frac{(\nabla s, \mathbf{c})}{\|s\|_1 \|\mathbf{c}\|_{H(\mathbf{curl};\Omega)}} \geq \beta_\mathrm{m} > 0, \tag{2.5}$$

that are known to be true at the continuous level, as well as the Poincaré-Friedrichs inequalities

$$\begin{aligned} \|\mathbf{v}\|_1 &\leq C_{\mathrm{P},1} \|\nabla \mathbf{v}\| && \text{for } \mathbf{v} \in H_0^1(\Omega)^d, \\ \|\mathbf{c}\|_{H(\mathbf{curl};\Omega)} &\leq C_{\mathrm{P},2} \|\nabla \times \mathbf{c}\| && \text{for } \mathbf{c} \in \boldsymbol{C} \cap H(\mathrm{div}\, 0; \Omega) \end{aligned} \tag{2.6}$$

(see [97, Corollary 3.51]). We refer to [106, 111] for a mathematical discussion about the well-posedness of this problem.

A (weighted) exact penalty formulation for the magnetic sub-problem

As we will discuss later on, the numerical approximation of saddle-point PDEs is complicated. So, we can consider an alternative formulation for the magnetic sub-problem, in which Eqs. (2.4c)-(2.4d) are replaced by

$$\mathrm{d}_t \left(\varrho \mathbf{b}, \mathbf{c}\right) + \left(\lambda \nabla \times \mathbf{b}, \nabla \times \mathbf{c}\right) - \langle \nabla \times (\mathbf{u} \times \varrho \mathbf{b}), \mathbf{c} \rangle + \lambda \left(\nabla \cdot \mathbf{b}, \nabla \cdot \mathbf{c}\right) = \langle \mathbf{f}_\mathrm{b}, \mathbf{c} \rangle. \tag{2.7}$$

We can easily check that systems (2.4) and (2.4a)-(2.4b)-(2.7) are equivalent (see [57,78]). This way, the only inf-sup condition that is needed in order for the problem to be well-posed is the one related to the fluid sub-problem. This formulation can be understood as an exact penalty strategy. Some numerical approximations of the MHD system have considered this statement of the problem [72, 76]. Let us remark that in this work, for the reasons commented below, the numerical methods that are proposed approximate the double-saddle point formulation (2.4).

2.3 Some finite element approximations

In this section, we show some of the different alternatives proposed so far for the numerical approximation of the resistive MHD system (2.1) using finite element techniques. In some cases, the references that are provided considered the stationary version of the

Fluid sub-problem	**Magnetic sub-problem (regularized)**			**Magnetic sub-problem (orig.)**		
	Inf-sup stable	Exact penalty	Stabilized method	Inf-sup stable	Weighted penalty	Stabilized method
Inf-sup stable		Gunzburger et al., [76]		Schötzau, [111]	Hasler et al., [77]	
Stabilized method		Gerbeau, [72]	Codina et al., [54]			This work

Table 2.1: References for different combinations of fluid and magnetic techniques for the MHD problem. For the magnetic sub-problem, we have distinguished among the methods that use the original functional setting and the ones that use the regularized setting (not suitable for singular solutions).

problem, but it does not affect the forthcoming discussion. First, we list FE approximations that are based on the different continuous formulations of the MHD problem. We can distinguish among the approximations that are suitable for singular solutions of the magnetic field, i.e. they are posed in the original functional setting $H(\mathbf{curl};\Omega)\times H^1(\Omega)$, and those that use the regularized formulation, only suitable for convex or smooth enough domains. In this case $\mathbf{b}\in H^1(\Omega)^d$. Table 2.1 shows different combinations of fluid and magnetic sub-problem numerical approximations. We distinguish among methods able to capture singular magnetic solutions and regularized ones. Further, we indicate the references in which every combination has been proposed. All those methods are explained below.

Let us start with formulations suitable for singular magnetic solutions. Since the problem at hand couples two saddle-point sub-problems, a Galerkin FE approximation should satisfy a discrete version of the two inf-sup conditions (2.5) above. In order to do this, we rely on the well-established mixed FE theory (see e.g. [34]). Since the two inf-sup conditions are different, stable approximations for the $(\mathbf{u},p)$ pair differ from those stable for the magnetic pair $(\mathbf{b},r)$. E.g. the fluid sub-problem could be solved by using the MINI element, whereas the magnetic problem should be approximated by a Nédélec FE space (see [99, 100]). The use of inf-sup stable elements for every sub-problem is the strategy used in [111]. From a theoretical point of view, this is a satisfactory approach, because the final MHD solver is able to approximate singular solutions, with $\mathbf{b}\notin H^1(\Omega)^d$. However, this approach is not appealing from a practical point of view, as commented in Section 2.1.2.

A way to simplify the numerical approximation consists in the use of an exact penalty formulation for the magnetic field. Under the following assumption over the domain:

$$\text{The domain } \Omega \text{ is convex or its boundary is } \mathcal{C}^{1,1} \tag{A1}$$

we can prove that the magnetic field does belong to $H^1(\Omega)^d$ (see [73]). This approach was proposed in [76] in the Galerkin case. In [72], a stabilized FE formulation for the fluid sub-problem as well as a Galerkin FE discretization of the exact penalty formulation for the magnetic sub-problem are proposed. The method allows one-graph implementation, arbitrary-order Lagrangian interpolations for all the unknowns (velocity, pressure and

magnetic field), and stabilizes convection. However, the straightforward exact penalty formulation only provides magnetic fields in $H^1(\Omega)^d$ [13, 57]. Further, $H^1(\Omega)^d$ is a strict closed subspace of $H(\mathbf{curl};\Omega) \cap H(\mathrm{div};\Omega)$ and so, H^1-conforming approximations do not have an approximability property (see e.g. [68]). As a result, the formulations based on the exact penalty lead to spurious numerical solutions in general. The formulation proposed in [54] uses a regularized double saddle-point formulation of the problem at hand, in order to explicitly enforce the free-divergence constraint over the magnetic field. Stabilized FE discretizations of both the fluid and magnetic sub-problem are considered. It shares the same benefits and limitations of the previous approach. From the conceptual point of view, the formulation is motivated from the variational multiscale framework (see [83,84]) and the stabilization parameters are carefully designed from the convergence analysis. Both aspects are shared by the formulation we propose in Section 2.5.

In order to solve this situation, less control over $\mathbf{b}$ is required, e.g. with a weighting in the penalty term. Costabel and Dauge proposed in [58] to replace the penalty term in (2.7) by a weighted one (in the frame of the electrostatic Maxwell problem) of the form $(\omega\lambda\nabla\cdot\mathbf{b}, \nabla\cdot\mathbf{c})$ where $\omega(\mathbf{x})$ is the weight function, that can only be effectively defined knowing *a priori* the location of the singularities (see [57]). It has been proved in [57] that a proper choice of this weighting allows one to capture singular solutions. The weighted exact penalty together with an inf-sup stable Galerkin approximation of the fluid problem has been proposed in [77] for the MHD problem (2.1).

2.4 Time discretization and linearization

All the developments in this work are for the spatial discretization of the MHD system (2.4). The stabilized FE formulation we propose can be applied to any time integration scheme. In particular, we can consider θ-methods for the time discretization. After spatial discretization, our time-continuous formulation can be written as a differential-algebraic equation:

$$\dot{\boldsymbol{x}}(t) = A(\boldsymbol{x}(t), \boldsymbol{y}(t), t), \qquad \mathbf{0} = C(\boldsymbol{x}(t), \boldsymbol{y}(t), t),$$

where $\boldsymbol{x}(t) \in \mathbb{R}^n$, $\boldsymbol{y}(t) \in \mathbb{R}^m$, $A : \mathbb{R}^{n+m} \to \mathbb{R}^n$ and $C : \mathbb{R}^{n+m} \to \mathbb{R}^m$. Using the definitions

$$\dot{\boldsymbol{x}}^{n+1} = \frac{\boldsymbol{x}^{n+1} - \boldsymbol{x}^n}{\delta t}, \qquad \boldsymbol{x}^{n+\theta} = \theta\boldsymbol{x}^{n+1} + (1-\theta)\boldsymbol{x}^n,$$

(analogously for $\boldsymbol{y}^{n+\theta}$), the time discretization of this problem with the θ-method simply reads as:

$$\dot{\boldsymbol{x}}^{n+1} = A(\boldsymbol{x}^{n+\theta}, \boldsymbol{y}^{n+\theta}, t^{n+\theta}), \qquad \mathbf{0} = C(\boldsymbol{x}^{n+\theta}, \boldsymbol{y}^{n+\theta}, t^{n+\theta}), \qquad \text{for } \theta \in (0,1].$$

We consider the following time-discretization of (2.2):

$$\begin{aligned}
\dot{\mathbf{u}}^{n+1} + \hat{\mathbf{u}}^n \cdot \nabla\mathbf{u}^{n+\theta} - \nu\Delta\mathbf{u}^{n+\theta} + \nabla p^{n+\theta} - (\nabla\times\mathbf{b}^{n+\theta})\times\varrho\hat{\mathbf{b}}^n &= \mathbf{f}_{\mathrm{u}}, && \text{(2.8a)}\\
\nabla\cdot\mathbf{u}^{n+\theta} &= 0, && \text{(2.8b)}\\
\varrho\dot{\mathbf{b}}^{n+1} + \lambda\nabla\times(\nabla\times\mathbf{b}^{n+\theta}) + \nabla r^{n+\theta} - \nabla\times(\mathbf{u}^{n+\theta}\times\varrho\hat{\mathbf{b}}^n) &= \mathbf{f}_{\mathrm{b}}, && \text{(2.8c)}\\
\nabla\cdot\mathbf{b}^{n+\theta} &= 0, && \text{(2.8d)}
\end{aligned}$$

where $\hat{\mathbf{u}}^n$ and $\hat{\mathbf{b}}^n$ can be either time extrapolations to approximate $\mathbf{u}^{n+\theta}$ and $\mathbf{b}^{n+\theta}$, respectively, or previous iterates within a nonlinear iteration strategy to compute the unknowns at time level $n+1$. In the first case, we may choose $\hat{\mathbf{u}}^n = \mathbf{u}^n$, $\hat{\mathbf{b}}^n = \mathbf{b}^n$ if $\theta \neq 1/2$ (first order schemes) or $\hat{\mathbf{u}}^n = 2\mathbf{u}^n - \mathbf{u}^{n-1}$, $\hat{\mathbf{b}}^n = 2\mathbf{b}^n - \mathbf{b}^{n-1}$ if $\theta = 1/2$ (second order scheme). In the second case, the linearization can be considered of Picard's type. In fact, it can be easily checked that this is the only fixed-point linearization of the problem that leads to an absolutely stable algorithm in time (see [54]). Segregated time-marching schemes that decouple the sub-problems are conditionally stable and not suitable when the coupling is important. For the sake of conciseness, we will consider $\hat{\mathbf{u}}^n = \mathbf{u}^n$, $\hat{\mathbf{b}}^n = \mathbf{b}^n$ in the following.

2.5 A stabilized FE formulation suitable for singular magnetic solutions

Let us present now the spatial discretization we propose. Let $\mathcal{T}_h = \{K\}$ be a FE partition of the domain Ω. For simplicity we assume Ω polyhedral and $\mathcal{T}_h$ quasi-uniform, of diameter h. Summation over all the element domains K is denoted as $\sum_K$. Finite element spaces and FE functions are identified with the subscript h. Only conforming approximations are considered, i.e., the FE spaces where the unknowns are sought are $\boldsymbol{V}_h \subset \boldsymbol{V}$, $\boldsymbol{C}_h \subset \boldsymbol{C}$, $Q_h \subset Q$ and $S_h \subset S$. In particular, in the applications we will be interested in the use of $\mathcal{C}^0$ Lagrangian finite element interpolations of an arbitrary order for all the unknowns. Given two functions f and g piecewise polynomial on each $K \in \mathcal{T}_h$, we define $(f,g)_h := \sum_K \int_K f\, g$ and $\|f\|_h := (f,f)_h^{1/2}$.

Since we assume $\mathcal{T}_h$ quasi-uniform, the following inverse inequality holds:

$$\|\nabla v_h\|_{L^2(K)} \leq \frac{C_{\text{inv}}}{h}\|v_h\|_{L^2(K)}, \qquad K \in \mathcal{T}_h, \tag{2.9}$$

for a positive constant C_{inv} and for all piecewise polynomial functions v_h.

We consider a residual-based stabilized FE formulation for the MHD problem. This type of formulation does not change the statement of the continuous problem but modifies the way the discretization is performed. Instead of considering only those terms that come from a Galerkin discretization, this type of formulation includes additional terms, that are always proportional to some residual, and so, consistent. In order for this approach to be effective, the new terms must provide stability over the Lagrange multiplier-type unknowns, in order to avoid the fulfillment of discrete inf-sup conditions, as well as convection stability (see e.g. [50]). These methods can be motivated as multiscale methods in which the sub-grid component of the solution is properly modeled. Different closures for the sub-grid component have been proposed so far; we can distinguish among standard variational multiscale formulations (also called algebraic sub-grid scales) and orthogonal sub-grid scales, enforced to be orthogonal to the FE functions in L^2 sense. Analogously, we can distinguish among standard quasi-static formulations, in which the time derivative of the sub-grid component is neglected, and dynamic closures; in this last case, the model for the sub-grid scale is an ordinary differential equation (see [53]). It is not the aim of this work to motivate the stabilized FE formulation we

propose. The heuristic motivation of these methods can be found in any of the references above. Since we are presenting a novel formulation, we have considered the most standard type of formulation, an algebraic quasi-static closure. [5]

In order to obtain a numerical algorithm suitable for singular solutions and avoiding the need to define weighting functions that require information about the placement of singularities, we stick to the double saddle-point formulation (2.4).

This way, using the approach indicated above, the stabilized FE formulation we propose is stated in Algorithm 2.1.

Let us stress the key features of our formulation, that allow us to design an unconditionally convergent stabilized method with the interesting properties listed above. The stabilization terms in the Navier-Stokes equations are standard, as well as the definition of the corresponding stabilization parameters τ_1 and τ_2, apart from the coupling with the magnetic field; see [54] for a detailed justification. With regard to the magnetic equation, we have split its residual into two parts $\mathbf{R}_{\mathrm{b},1}$ and $\mathbf{R}_{\mathrm{b},2}$; these two terms are still residuals, since $r = 0$. Further, we have included the solenoidal constraint residual R_r. Now, we are in position to pre-multiply these three residuals with different stabilization parameters: τ_3, τ_4 and τ_5 respectively. On one hand, $\tau_4 \sim \mathcal{O}(1)$ in order to have H^1-stability over r_h. On the other hand, $\tau_3 \sim \mathcal{O}(h^2)$, in order for $(\nabla \times \nabla \times \mathbf{c}_h, \tau_3 \nabla \times \nabla \times \mathbf{b}_h)_h$ to have sense for $\mathbf{b}_h \in H(\mathbf{curl}; \Omega)$ only; we refer to Chapter 3 for the complete numerical analysis of the method. Even though it is not so obvious, we have to pick $\tau_5 \sim \mathcal{O}(h^2)$ in order to have full stability of $\mathbf{b}_h$ in $H(\mathbf{curl}; \Omega)$. We refer to [13] for a detailed justification of this fact via the numerical analysis for the Maxwell operator.[6] Note that the stabilization parameters in Algorithm 2.1, τ_i $(i = 1, ..., 5)$, depend on algorithmic constants c_j $(j = 1, ..., 7)$ as well as on a length scale L_0, and the asymptotic behavior just described assumes that these are fixed. This is common to all stabilized finite element methods, and thus it is not our purpose to discuss how to choose the constants. Results may be sensitive to their value for a fixed mesh, but the asymtotic behavior of the numerical formulation is independent of it, provided they are $\mathcal{O}(1)$. We indicate in the numerical examples how the constants have been taken.

At this point, let us comment the main difference among the proposed formulation (Algorithm 2.1) and the one in [54]. In this reference, the residual was not split, i.e. the gradient of the magnetic pseudo-pressure was included in the residual for $\mathbf{b}$ and $\tau_3 = \tau_4$. Even though this might seem the most natural option, the convergence analysis leads to the need to take the stabilization parameters as $\tau_3 = \tau_4 \sim \mathcal{O}(h^2)$ and $\tau_5 \sim \mathcal{O}(1)$. So, the stabilized formulation in [54] mimics the regularized functional setting and cannot

[5] Roughly speaking, the type of formulation we propose reads as follows. Given a PDE (in strong form) that reads as $\mathcal{L}(x) = f$, where $\mathcal{L}$ is the differential operator, and with weak form $a(x, y) = (f, y)$, after the corresponding integration-by-parts and the definition of suitable functional spaces, the Galerkin formulation consists of replacing infinite-dimensional spaces by finite-dimensional ones, keeping the form of the problem, i.e. $a(x_h, y_h) = (f, y_h)$. The stabilized formulation we consider would replace this discrete problem by

$$a(x_h, y_h) - \sum_{K \in \mathcal{T}_h} \int_K \mathcal{L}(x_h) \tau \mathcal{L}^*(y_h) = (f, y_h) - \sum_{K \in \mathcal{T}_h} \int_K f \tau \mathcal{L}^*(y_h),$$

where $\tau(\mathbf{x})$ is the stabilization parameter, whose expression can be motivated using Fourier analysis [49] or inferred from the numerical analysis [53] and Chapter 5.

[6] $\tau_5 \sim \mathcal{O}(1)$ would provide $\mathbf{b}_h \in H^1(\Omega)^d$ (see [13]) and so, it would not converge to singular solutions.

Algorithm 2.1: Final algorithm

Given $\mathbf{u}_h^n$ and $\mathbf{b}_h^n$ at the previous time step value, find $\mathbf{u}_h^{n+1} \in \boldsymbol{V}_h$, $\mathbf{b}_h^{n+1} \in \boldsymbol{C}_h$, $p_h^{n+1} \in Q_h$ and $r_h^{n+1} \in S_h$ such that

$$
\begin{aligned}
&(\dot{\mathbf{u}}_h^{n+1}, \mathbf{v}_h) + \left\langle (\mathbf{u}_h^n \cdot \nabla)\mathbf{u}_h^{n+\theta}, \mathbf{v}_h \right\rangle + \nu(\nabla \mathbf{u}_h^{n+\theta}, \nabla \mathbf{v}_h) - (p_h^{n+\theta}, \nabla \cdot \mathbf{v}_h) \\
&\quad - \left\langle (\nabla \times \mathbf{b}_h^{n+\theta}) \times \varrho \mathbf{b}_h^n, \mathbf{v}_h \right\rangle \\
&\quad + \left\langle \mathbf{u}_h^n \cdot \nabla \mathbf{v}_h + \nu \Delta \mathbf{v}_h, \tau_1^n \mathbf{R}_{\mathrm{u}}(\mathbf{u}_h^n, \mathbf{b}_h^n; \mathbf{u}_h^{n+\theta}, \mathbf{b}_h^{n+\theta}, p_h^{n+\theta}) \right\rangle_h + \left\langle \nabla \cdot \mathbf{v}_h, \tau_2^n R_p(\mathbf{u}_h^{n+\theta}) \right\rangle_h \\
&\quad - \left\langle \nabla \times (\mathbf{v}_h \times \varrho \mathbf{b}_h^n), \tau_3^n \mathbf{R}_{\mathrm{b},1}(\mathbf{b}_h^n; \mathbf{u}_h^{n+\theta}, \mathbf{b}_h^{n+\theta}) \right\rangle_h = \left\langle \mathbf{f}_{\mathrm{u}}^{n+1}, \mathbf{v}_h \right\rangle, \\
&(q_h, \nabla \cdot \mathbf{u}_h^{n+\theta}) + \left\langle \nabla q_h, \tau_1^n \mathbf{R}_{\mathrm{u}}(\mathbf{u}_h^n, \mathbf{b}_h^n; \mathbf{u}_h^{n+\theta}, \mathbf{b}_h^{n+\theta}, p_h^{n+\theta}) \right\rangle_h = 0, \\
&(\varrho \dot{\mathbf{b}}_h^{n,k+1}, \mathbf{c}_h) + (\lambda \nabla \times \mathbf{b}_h^{n+\theta}, \nabla \times \mathbf{c}_h) - (\nabla \times (\mathbf{u}_h^{n+\theta} \times \varrho \mathbf{b}_h^n), \mathbf{c}_h) \\
&\quad + (\nabla r^{n+\theta}, \mathbf{c}_h) - \left\langle (\nabla \times \mathbf{c}_h) \times \varrho \mathbf{b}^n, \tau_1^n \mathbf{R}_{\mathrm{u}}(\mathbf{u}_h^n, \mathbf{b}_h^n; \mathbf{u}_h^{n+\theta}, \mathbf{b}_h^{n+\theta}, p_h^{n+\theta}) \right\rangle_h \\
&\quad - \left\langle \lambda \nabla \times (\nabla \times \mathbf{c}_h), \tau_3^n \mathbf{R}_{\mathrm{b},1}^{n+\theta}(\mathbf{b}_h^n; \mathbf{u}_h^{n+\theta}, \mathbf{b}_h^{n+\theta}) \right\rangle_h \\
&\quad + \left\langle \nabla \cdot \mathbf{c}_h, \tau_5^n R_r(\mathbf{b}^{n+\theta}) \right\rangle_h = \left\langle \mathbf{f}_{\mathrm{b}}^{n+\theta}, \mathbf{c}_h \right\rangle, \\
&-(\nabla s_h, \mathbf{b}_h^{n+\theta}) + \left\langle \nabla s_h, \tau_4^n \mathbf{R}_{\mathrm{b},2}(r_h^{n+\theta}) \right\rangle_h = 0,
\end{aligned}
$$

where the residuals are:

$$
\begin{aligned}
&\mathbf{R}_{\mathrm{u}}(\mathbf{a}, \mathbf{d}; \mathbf{u}_h, \mathbf{b}_h, p_h) &&:= \dot{\mathbf{u}}_h + \mathbf{a} \cdot \nabla \mathbf{u}_h - \nu \Delta \mathbf{u}_h + \nabla p_h - (\nabla \times \mathbf{b}_h) \times \varrho \mathbf{d} - \mathbf{f}_{\mathrm{u}}, \\
&R_p(\mathbf{u}_h) &&:= \nabla \cdot \mathbf{u}_h, \\
&\mathbf{R}_{\mathrm{b},1}(\mathbf{d}; \mathbf{u}_h, \mathbf{b}_h) &&:= \varrho \dot{\mathbf{b}}_h + \lambda \nabla \times (\nabla \times \mathbf{b}_h) - \nabla \times (\mathbf{u}_h \times \varrho \mathbf{d}) - \mathbf{f}_{\mathrm{b}}, \\
&\mathbf{R}_{\mathrm{b},2}(r_h) &&:= \nabla r_h, \\
&R_r(\mathbf{b}_h) &&:= \nabla \cdot \mathbf{b}_h.
\end{aligned}
$$

The stabilization parameters have the following expressions within each element K:

$$
\begin{aligned}
&\tau_1^m := (\alpha^m)^{-1} \left(1 + \tfrac{\phi^m}{\sqrt{\alpha^m \gamma}}\right)^{-1}, \quad \tau_2 := c_5 \tfrac{h^2}{\tau_1^m}, \quad \tau_3 := \gamma^{-1} \left(1 + \tfrac{\phi^m}{\sqrt{\alpha^m \gamma}}\right)^{-1}, \\
&\tau_4 := c_6 \tfrac{L_0^2}{\lambda}, \qquad\qquad \tau_5 := c_7 \tfrac{h^2 \lambda}{L_0^2},
\end{aligned}
$$

with

$$
\alpha^m := c_1 \frac{\|\mathbf{u}_h^m\|_{L^\infty(K)}}{h} + c_2 \frac{\nu}{h^2}, \qquad \phi^m := c_3 \frac{\varrho \|\mathbf{b}_h^m\|_{L^\infty(K)}}{h}, \qquad \gamma := c_4 \frac{\lambda}{h^2}.
$$

$c_1, \ldots, c_5$ are algorithmic constants that must satisfy $c_1 > \frac{2}{C_{\mathrm{inv}}^2}$ and $c_3 > \frac{2}{C_{\mathrm{inv}}^2}$ and L_0 is a lengthscale of the problem.

capture singular solutions.

As commented above, in this work we have considered stabilization terms of the standard variational multiscale (VMS) type. Straightforwardly, we could consider Galerkin

least-square (GLS) terms (that only differ in some signs with respect to VMS), orthogonal sub-scale methods introducing the proper projections [49] or more involved dynamic sub-grid closures [55]. In a subsequent work, we aim at exploiting these last approaches in the frame of MHD turbulence.

Remark 2.1. *In the numerical analysis of the stabilized FE formulation for the Maxwell problem [13] and its extension to the full MHD system in Chapter 3, convergence to singular solutions of the magnetic field* $\mathbf{b} \notin H^1(\Omega)^d$ *is proved under one assumption about the approximability of the Lagrangian FE spaces. In particular, this assumption holds for meshes with a particular macro-element structure (see [13, Assumption 1 and Corollary 5]). One type of macro-element that satisfies this condition is the Powell-Sabin macro-element (see [114] and [36, Remark 4.1]). Further, in two dimensions, numerical experiments show that both the Powell-Sabin and "crossbox" elements provide excellent results. In the numerical experiments section, we extend this work to three-dimensions; we have considered both the 3D Powell-Sabin element and a 3D extension of the crossbox one; both choices exhibit excellent convergence properties. Finally, let us stress the fact that this macro-element structure is only needed for singular solutions, that appear in non-convex domains. For smooth solutions, with* $\mathbf{b} \in H^1(\Omega)^d$*, the convergence analysis and approximability properties are easy to check and hold for any type of mesh. We refer to [13] for a detailed discussion on this topic, in the framework of the Maxwell operator. Further, in [12] we have observed that the method still converges to singular solutions for general meshes with a significantly lower order of convergence.*

2.6 Numerical experimentation

The numerical examples shown in the next subsections have been computed with some common computational settings. First, linear $\mathcal{C}^0$ Lagrangian FEs have been used, both in the 2D and 3D cases.[7] The solution of the resulting linear systems of equations has been obtained via a direct solver. The problem non-linearity has been solved through Picard iterations until converging to a tolerance of 10^{-4}. Finally, the constants that appear in the definition of the stabilization parameters in Algorithm 2.1 have been chosen as:

$$c_1 = c_3 = 2, \quad c_2 = c_4 = c_7 = 4, \quad \text{and} \quad c_5 = c_6 = 1.$$

2.6.1 Convergence to singular solutions. Case 2D

The stabilization method presented in the previous chapter has been used to solve the MHD problem in a nonconvex L-shaped domain. Due to the re-entrant corner present in the domain, both the Stokes and the Maxwell operators have strong singularities at this corner, where the origin of coordinates is taken. The non-smooth solution for the Stokes

[7] In this situation, all the system matrix blocks have the sparsity pattern defined by the mesh graph.

operator is described in polar coordinates (r, θ) by

$$\begin{aligned} u_x(x,y) =& r^\lambda \left((1+\lambda)\sin(\theta)\psi(\theta) + \cos(\theta)\psi'(\theta) \right), \\ u_y(x,y) =& r^\lambda \left(-(1+\lambda)\cos(\theta)\psi(\theta) + \sin(\theta)\psi'(\theta) \right), \\ p(x,y) =& -\frac{r^{\lambda-1}}{1-\lambda} \left((1+\lambda)^2\psi'(\theta) + \psi'''(\theta) \right), \end{aligned}$$

where

$$\psi(\theta) = \sin((1+\lambda)\theta)\frac{\cos(\lambda\omega)}{1+\lambda} - \cos((1+\lambda)\theta) - \sin((1-\lambda)\theta)\frac{\cos(\lambda\omega)}{1-\lambda} + \cos((1-\lambda)\theta).$$

The value of the parameter λ is the smallest positive solution of

$$\sin(\lambda\omega) + \lambda\sin(\omega) = 0, \quad \text{where } \omega = \frac{3\pi}{2},$$

which is $\lambda \sim 0.54448373678246$. Note that $\mathbf{u} = (u_x, u_y)$ is solenoidal and $(\mathbf{u}, p) \in H^{1+\lambda}(\Omega)^2 \times H^\lambda(\Omega)$.
The singular solution for the Maxwell operator is defined also in polar coordinates as

$$\mathbf{b}(x,y) = \nabla \left(r^{\frac{2n}{3}} \sin\left(\frac{2n}{3}\theta \right) \right), \qquad n \in \mathbb{N}^+.$$

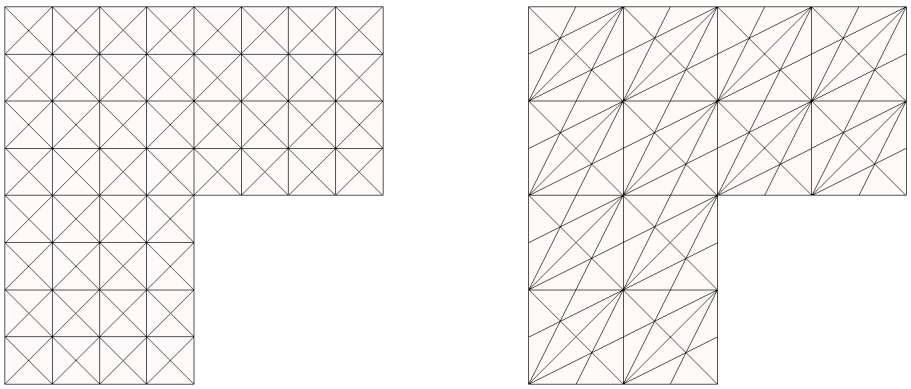
Note that $\nabla \cdot \mathbf{b} = 0$ and $\nabla \times \mathbf{b} = \mathbf{0}$. The magnetic induction field $\mathbf{b} \in H^{\frac{2n}{3}}(\Omega)^2$ and therefore, $\mathbf{b} \notin H^1(\Omega)^2$ for $n = 1$ only.

The computation has been done fully-coupled, solving the implicit scheme stated in Algorithm 2.1 and using two different mesh structures suitable to approximate the Maxwell operator singularity, namely the Powell-Sabin and the crossbox element. Figure 2.1 shows an example of the two different mesh structures.

Remark 2.2. *Let us stress the fact that, since both crossbox and Powell-Sabin macro-element meshes allow to converge to singular solutions, we can start with any triangle or quadrilateral mesh, and after a simple processing of the mesh, solve the problem on the modified mesh. Therefore, the previous macro-element topology can be attained for any original mesh.*

In both cases, the problem has been solved on several meshes with different mesh sizes, from the coarsest one with $h = 2^{-2}$ to the finest one with $h = 2^{-7}$. Tables 2.2 and 2.3 contain the numerical error norms for the case of crossbox element meshes. In Table 2.2, we show the numerical errors for the velocity in the L^2-norm and the H^1-norm and the error for the pressure in the L^2-norm. In Table 2.3, we have listed the numerical errors for the magnetic unknowns, that is, the error for the magnetic induction in the L^2-norm and the $H(\mathbf{curl})$-norm and the errors for the magnetic pseudo-pressure in both the L^2-norm and the H^1-norm.

Furthermore, Figure 2.2 shows the convergence plots of the numerical errors presented in Tables 2.2 and 2.3. It is clearly seen that the proposed method is able to converge to the singular solutions when using meshes of crossbox elements.



(a) Crossbox element (b) Powell-Sabin element

Figure 2.1: Mesh structures for the 2D case.

h	$\|\|\mathbf{e}_{\mathrm{u}}\|\|$	$\|\|\nabla\mathbf{e}_{\mathrm{u}}\|\|$	$\|\|e_p\|\|$
2^{-2}	$8.49 \cdot 10^{-2}$ (-)	$1.30 \cdot 10^{0}$ (-)	$1.69 \cdot 10^{0}$ (-)
2^{-3}	$3.45 \cdot 10^{-2}$ (1.30)	$8.95 \cdot 10^{-1}$ (0.54)	$1.13 \cdot 10^{0}$ (0.58)
2^{-4}	$1.45 \cdot 10^{-2}$ (1.25)	$6.16 \cdot 10^{-1}$ (0.54)	$7.66 \cdot 10^{-1}$ (0.56)
2^{-5}	$6.28 \cdot 10^{-3}$ (1.21)	$4.24 \cdot 10^{-1}$ (0.54)	$5.22 \cdot 10^{-1}$ (0.55)
2^{-6}	$2.79 \cdot 10^{-3}$ (1.17)	$2.91 \cdot 10^{-1}$ (0.54)	$3.57 \cdot 10^{-1}$ (0.55)
2^{-7}	$1.27 \cdot 10^{-3}$ (1.14)	$2.00 \cdot 10^{-1}$ (0.54)	$2.44 \cdot 10^{-1}$ (0.55)

Table 2.2: Numerical errors for hydrodynamic unknowns and rate of convergence in brackets. Crossbox element in 2D.

Tables 2.4 and 2.5 display the numerical errors obtained when using meshes with a Powell-Sabin macro-element structure to approximate the singular solutions. We show in these tables the same error quantities as before, for the fluid and the magnetic subproblems respectively.

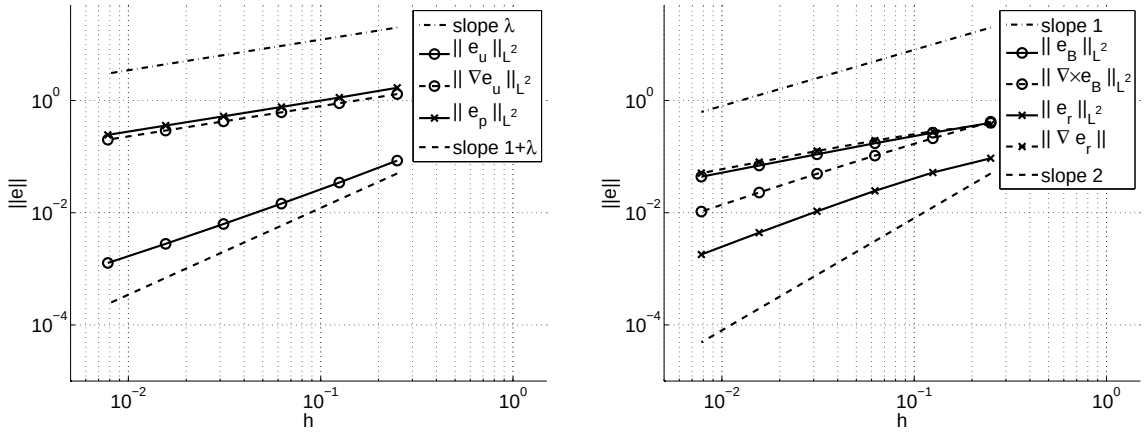
Figure 2.3 shows the convergence plots of the numerical errors presented in Tables 2.4 and 2.5. Similarly to the previous case, the proposed method with a Powell-Sabin mesh is clearly suitable to approximate non-smooth solutions for both the Stokes and the Maxwell operator.

2.6.2 Convergence to singular solutions. Extension to the 3D case

The easiest problem with singular solutions in dimension three consists of the two-dimensional solution for the x-y components and a constant in the third component. In this case, the third component (z-axis) and its corresponding derivatives are null. Therefore, the computational domain considered is the volume generated by the extrusion of the 2D L-shaped domain (x-y plane) in the z-axis, for a given thickness. Periodic

h	$\|\|\mathbf{e}_\mathrm{b}\|\|$	$\|\|\nabla \times \mathbf{e}_\mathrm{b}\|\|$	$\|\|e_r\|\|$	$\|\|\nabla e_r\|\|$
2^{-2}	$4.01 \cdot 10^{-1}$ (-)	$4.19 \cdot 10^{-1}$ (-)	$9.34 \cdot 10^{-2}$ (-)	$3.77 \cdot 10^{-1}$ (-)
2^{-3}	$2.67 \cdot 10^{-1}$ (0.59)	$2.12 \cdot 10^{-1}$ (0.98)	$5.21 \cdot 10^{-2}$ (0.84)	$2.83 \cdot 10^{-1}$ (0.41)
2^{-4}	$1.73 \cdot 10^{-1}$ (0.63)	$1.04 \cdot 10^{-1}$ (1.03)	$2.45 \cdot 10^{-2}$ (1.09)	$1.93 \cdot 10^{-1}$ (0.55)
2^{-5}	$1.10 \cdot 10^{-1}$ (0.65)	$4.94 \cdot 10^{-2}$ (1.07)	$1.06 \cdot 10^{-2}$ (1.21)	$1.26 \cdot 10^{-1}$ (0.62)
2^{-6}	$6.94 \cdot 10^{-2}$ (0.66)	$2.29 \cdot 10^{-2}$ (1.11)	$4.42 \cdot 10^{-3}$ (1.26)	$8.02 \cdot 10^{-2}$ (0.65)
2^{-7}	$4.38 \cdot 10^{-2}$ (0.66)	$1.05 \cdot 10^{-2}$ (1.12)	$1.80 \cdot 10^{-3}$ (1.30)	$5.07 \cdot 10^{-2}$ (0.66)

Table 2.3: Numerical errors for magnetic unknowns and rate of convergence in brackets. Crossbox element in 2D.



(a) Hydrodynamic unknowns

(b) Magnetic unknowns

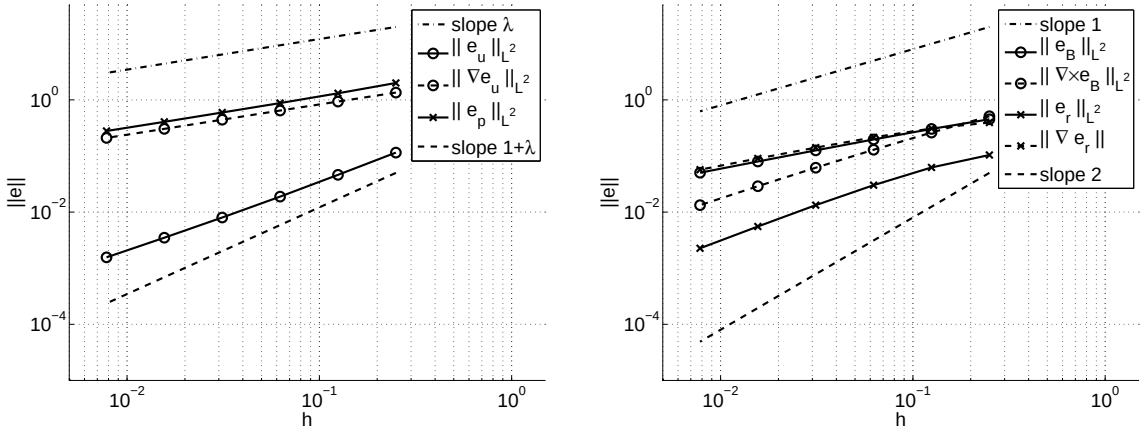
Figure 2.2: Convergence plots for the singular solutions in the 2D case. Crossbox element.

h	$\|\|\mathbf{e}_\mathrm{u}\|\|$	$\|\|\nabla \mathbf{e}_\mathrm{u}\|\|$	$\|\|e_p\|\|$
2^{-2}	$1.15 \cdot 10^{-1}$ (-)	$1.35 \cdot 10^{0}$ (-)	$2.00 \cdot 10^{0}$ (-)
2^{-3}	$4.61 \cdot 10^{-2}$ (1.32)	$9.35 \cdot 10^{-1}$ (0.53)	$1.31 \cdot 10^{0}$ (0.61)
2^{-4}	$1.89 \cdot 10^{-2}$ (1.29)	$6.45 \cdot 10^{-1}$ (0.54)	$8.79 \cdot 10^{-1}$ (0.58)
2^{-5}	$7.99 \cdot 10^{-3}$ (1.24)	$4.44 \cdot 10^{-1}$ (0.54)	$5.97 \cdot 10^{-1}$ (0.56)
2^{-6}	$3.49 \cdot 10^{-3}$ (1.19)	$3.05 \cdot 10^{-1}$ (0.54)	$4.07 \cdot 10^{-1}$ (0.55)
2^{-7}	$1.56 \cdot 10^{-3}$ (1.16)	$2.10 \cdot 10^{-1}$ (0.54)	$2.79 \cdot 10^{-1}$ (0.54)

Table 2.4: Numerical errors for hydrodynamic unknowns and rate of convergence in brackets. Powell-Sabin element in 2D.

h	$\|\|\mathbf{e}_\mathrm{b}\|\|$	$\|\|\nabla \times \mathbf{e}_\mathrm{b}\|\|$	$\|\|e_r\|\|$	$\|\|\nabla e_r\|\|$
2^{-2}	$4.51 \cdot 10^{-1}$ (-)	$5.11 \cdot 10^{-1}$ (-)	$1.04 \cdot 10^{-1}$ (-)	$3.95 \cdot 10^{-1}$ (-)
2^{-3}	$3.04 \cdot 10^{-1}$ (0.57)	$2.61 \cdot 10^{-1}$ (0.97)	$6.24 \cdot 10^{-2}$ (0.74)	$3.09 \cdot 10^{-1}$ (0.35)
2^{-4}	$1.97 \cdot 10^{-1}$ (0.63)	$1.29 \cdot 10^{-1}$ (1.02)	$3.02 \cdot 10^{-2}$ (1.05)	$2.15 \cdot 10^{-1}$ (0.52)
2^{-5}	$1.26 \cdot 10^{-1}$ (0.64)	$6.19 \cdot 10^{-2}$ (1.06)	$1.33 \cdot 10^{-2}$ (1.18)	$1.41 \cdot 10^{-1}$ (0.61)
2^{-6}	$7.98 \cdot 10^{-2}$ (0.66)	$2.89 \cdot 10^{-2}$ (1.10)	$5.54 \cdot 10^{-3}$ (1.26)	$9.04 \cdot 10^{-2}$ (0.64)
2^{-7}	$5.03 \cdot 10^{-2}$ (0.67)	$1.33 \cdot 10^{-2}$ (1.12)	$2.26 \cdot 10^{-3}$ (1.29)	$5.70 \cdot 10^{-2}$ (0.67)

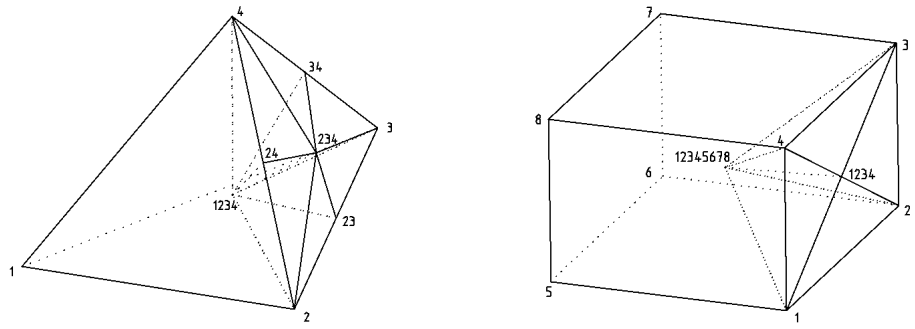
Table 2.5: Numerical errors for magnetic unknowns and rate of convergence in brackets. Powell-Sabin element in 2D.



(a) Hydrodynamic unknowns

(b) Magnetic unknowns

Figure 2.3: Convergence plots for the singular solutions in the 2D case. Powell-Sabin element.



(a) Tetrahedron remeshing (b) Hexahedron remeshing

Figure 2.4: Remeshing structure suitable for interpolation of Maxwell singular solutions.

boundary conditions have been set in the planes x-y.

Analogously to the bi-dimensional case, the approximation of non-smooth solutions of the Maxwell operator requires solving the problem on a proper modification of the original mesh with a macro-element structure. In the three-dimensional case, the 3D extensions of the bi-dimensional crossbox and the Powell-Sabin macro-elements displayed in Figure 2.4 are suitable to converge to the singular solutions. On one hand, the Powell-Sabin macro-element in 3D consists of the division of every tetrahedron of the original mesh into 24 new tetrahedra [114]. Each tetrahedron face is decomposed into 6 parts as the Powell-Sabin element in 2D. Each one of these parts forms a tetrahedron together with the original tetrahedron centroid (see Figure 2.4(a)). In this way, 6 new tetrahedra are obtained for each one of the original tetrahedron faces. On the other hand, the extension to three dimensions of the crossbox element requires an original (possibly) non-structured mesh of hexahedra. Each hexahedron face is divided into 4 triangles as the crossbox element in 2D and then, 4 tetrahedra are obtained together with the hexahedron centroid (see Figure 2.4(b)). Therefore, each original hexahedron is divided into 24 new tetrahedra.

Remark 2.3. *An equivalent remark to the bi-dimensional case can be made. In three dimensions, we can use any tetrahedral or hexahedral mesh and easily process this mesh in order to get one with the 3D crossbox or 3D Powell-Sabin macro-element structure.*

The results obtained for different mesh-sizes using crossbox elements in 3D are displayed in Tables 2.6 and 2.7. The error norms calculated are the same as in the 2D cases. It is clearly seen that, similarly to the 2D case, the numerical solution is able to converge to singular (non-smooth) solutions. Figure 2.5 shows graphically the convergence plots displaying the obtained slopes for every computed error norm.

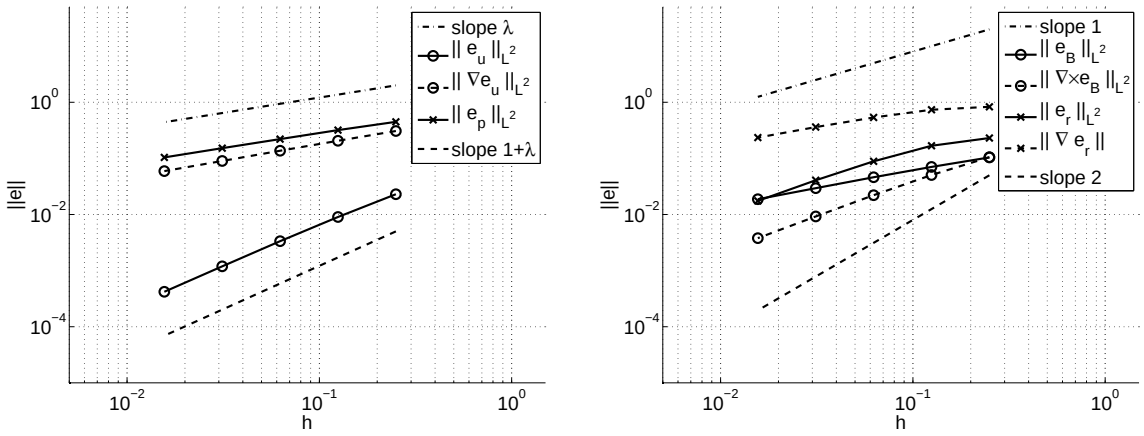
The results obtained when using the Powell-Sabin elements in 3D instead of the crossbox elements are shown in Tables 2.8 and 2.9. The error norms computed in this case are the same as in the previous case with crossbox elements, and the behavior of the numerical solution is the same. The results show clearly that the proposed method is able to converge to singular solutions. Figure 2.6 displays the convergence plots showing graphically the obtained results.

h	$\|\|\mathbf{e}_\mathrm{u}\|\|$	$\|\|\nabla\mathbf{e}_\mathrm{u}\|\|$	$\|\|e_p\|\|$
2^{-2}	$2.30\cdot10^{-2}$ (-)	$3.08\cdot10^{-1}$ (-)	$4.49\cdot10^{-1}$ (-)
2^{-3}	$9.04\cdot10^{-3}$ (1.35)	$2.06\cdot10^{-1}$ (0.58)	$3.19\cdot10^{-1}$ (0.49)
2^{-4}	$3.34\cdot10^{-3}$ (1.44)	$1.37\cdot10^{-1}$ (0.59)	$2.21\cdot10^{-1}$ (0.53)
2^{-5}	$1.19\cdot10^{-3}$ (1.49)	$8.99\cdot10^{-2}$ (0.61)	$1.52\cdot10^{-1}$ (0.54)
2^{-6}	$4.19\cdot10^{-4}$ (1.51)	$5.94\cdot10^{-2}$ (0.60)	$1.04\cdot10^{-1}$ (0.55)

Table 2.6: Numerical errors for hydrodynamic unknowns and rate of convergence in brackets. Crossbox element in 3D.

h	$\|\|\mathbf{e}_\mathrm{b}\|\|$	$\|\|\nabla\times\mathbf{e}_\mathrm{b}\|\|$	$\|\|e_r\|\|$	$\|\|\nabla e_r\|\|$
2^{-2}	$1.04\cdot10^{-1}$ (-)	$1.05\cdot10^{-1}$ (-)	$2.31\cdot10^{-1}$ (-)	$8.31\cdot10^{-1}$ (-)
2^{-3}	$7.05\cdot10^{-2}$ (0.56)	$5.06\cdot10^{-2}$ (1.05)	$1.68\cdot10^{-1}$ (0.46)	$7.38\cdot10^{-1}$ (0.17)
2^{-4}	$4.61\cdot10^{-2}$ (0.61)	$2.21\cdot10^{-2}$ (1.20)	$8.87\cdot10^{-2}$ (0.92)	$5.37\cdot10^{-1}$ (0.46)
2^{-5}	$2.95\cdot10^{-2}$ (0.64)	$9.24\cdot10^{-3}$ (1.26)	$4.07\cdot10^{-2}$ (1.12)	$3.61\cdot10^{-1}$ (0.57)
2^{-6}	$1.87\cdot10^{-2}$ (0.66)	$3.79\cdot10^{-3}$ (1.29)	$1.75\cdot10^{-2}$ (1.22)	$2.35\cdot10^{-1}$ (0.62)

Table 2.7: Numerical errors for magnetic unknowns and rate of convergence in brackets. Crossbox element in 3D.



(a) Hydrodynamic unknowns

(b) Magnetic unknowns

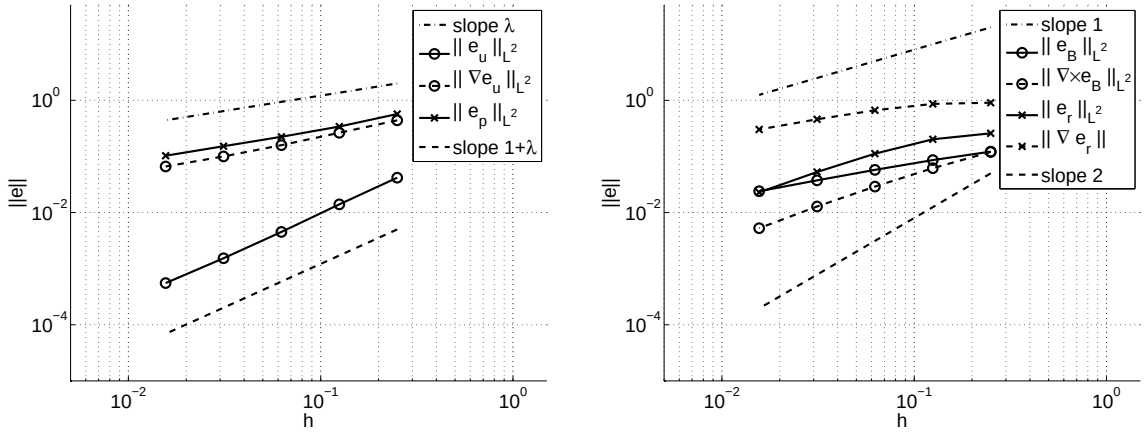
Figure 2.5: Convergence plots for the singular solutions in the 3D case. Crossbox element.

h	$\|\|\mathbf{e}_\mathrm{u}\|\|$	$\|\|\nabla\mathbf{e}_\mathrm{u}\|\|$	$\|\|e_p\|\|$
2^{-2}	$4.18\cdot10^{-2}$ (-)	$4.41\cdot10^{-1}$ (-)	$5.71\cdot10^{-1}$ (-)
2^{-3}	$1.40\cdot10^{-2}$ (1.58)	$2.64\cdot10^{-1}$ (0.74)	$3.41\cdot10^{-1}$ (0.74)
2^{-4}	$4.51\cdot10^{-3}$ (1.63)	$1.59\cdot10^{-1}$ (0.73)	$2.24\cdot10^{-1}$ (0.61)
2^{-5}	$1.53\cdot10^{-3}$ (1.56)	$1.00\cdot10^{-1}$ (0.67)	$1.52\cdot10^{-1}$ (0.56)
2^{-6}	$5.53\cdot10^{-4}$ (1.47)	$6.63\cdot10^{-2}$ (0.59)	$1.03\cdot10^{-1}$ (0.56)

Table 2.8: Numerical errors for hydrodynamic unknowns and rate of convergence in brackets. Powell-Sabin element in 3D.

h	$\|\|\mathbf{e}_\mathrm{b}\|\|$	$\|\|\nabla\times\mathbf{e}_\mathrm{b}\|\|$	$\|\|e_r\|\|$	$\|\|\nabla e_r\|\|$
2^{-2}	$1.22\cdot10^{-1}$ (-)	$1.20\cdot10^{-1}$ (-)	$2.58\cdot10^{-1}$ (-)	$9.08\cdot10^{-1}$ (-)
2^{-3}	$8.60\cdot10^{-2}$ (0.50)	$6.17\cdot10^{-2}$ (0.96)	$2.02\cdot10^{-1}$ (0.35)	$8.65\cdot10^{-1}$ (0.07)
2^{-4}	$5.76\cdot10^{-2}$ (0.58)	$2.90\cdot10^{-2}$ (1.09)	$1.12\cdot10^{-1}$ (0.85)	$6.64\cdot10^{-1}$ (0.38)
2^{-5}	$3.75\cdot10^{-2}$ (0.62)	$1.28\cdot10^{-2}$ (1.18)	$5.28\cdot10^{-2}$ (1.08)	$4.60\cdot10^{-1}$ (0.53)
2^{-6}	$2.40\cdot10^{-2}$ (0.64)	$5.26\cdot10^{-3}$ (1.28)	$2.29\cdot10^{-2}$ (1.21)	$3.02\cdot10^{-1}$ (0.61)

Table 2.9: Numerical errors for magnetic unknowns and rate of convergence in brackets. Powell-Sabin element in 3D.



(a) Hydrodynamic unknowns

(b) Magnetic unknowns

Figure 2.6: Convergence plots for the singular solutions in the 3D case. Powell-Sabin element.

2.6.3 Classical MHD problems with analytical solution. Shercliff's case

The stabilized finite element method proposed in this paper has also been tested to simulate some classical problems of MHD, namely Shercliff's [113] and Hunt's [86] cases. Shercliff's problem consists of a fully developed flow in a squared-section channel where both the Hartmann walls, perpendicular to the externally applied magnetic field, and the side walls, parallel to the external magnetic field, are considered electrically insulating. Considering that the external magnetic field, $\mathbf{b}_{\text{ext}}$ is applied in the y-direction and the longitudinal axis of the channel is the z-direction, the fluid flows driven by an imposed pressure gradient, with a velocity field where only the z-component is not null. This problem has an analytical solution in the form of Fourier series, whose expression can be found in [113] and Chapter 5.

The computational domain has been set as a channel slice of width 1/100 times the section sides. The boundary conditions on the inflow and outflow sections have been set as periodic conditions in order to enforce the condition of fully developed flow. The pressure gradient that drives the flow has been set as an external hydrodynamic body force, with value

$$\frac{\mathrm{d}p}{\mathrm{d}z} = \frac{KL^3}{\rho\nu^2 \mathrm{Re}}$$

$$K = \frac{\mathrm{Ha}}{1 - 0.825\mathrm{Ha}^{-1/2} - \mathrm{Ha}^{-1}}, \qquad \mathrm{Re} = \frac{\mathrm{UL}}{\nu},$$

where Ha and Re are the Hartmann [8] and Reynolds [9] numbers, respectively.

Finally, the boundary conditions on the walls, both the Hartmann and side walls, have been set as non-slip conditions for the velocity, $\mathbf{u} = \mathbf{0}$, and as the essential boundary conditions for the magnetic unknowns because the walls are electrically insulating,

$$\mathbf{b} \times \mathbf{n} = \mathbf{b}_{\text{ext}} \times \mathbf{n} \quad \text{and} \quad r = 0. \tag{2.10}$$

The physical properties of the problem, density, viscosity, electrical conductivity and magnetic permeability have been set to one when solving this problem. Therefore, the Hartmann number Ha corresponds to the norm of the externally applied magnetic field. This problem has been solved on several meshes with different mesh sizes h in order to study the convergence properties of the method. Two different cases have been simulated depending on the Hartmann number, namely Ha = 10 and Ha = 100, in order to

[8] The Hartmann number is the ratio between electromagnetic and viscous forces. Its expression is:

$$\mathrm{Ha} = BL\sqrt{\frac{\sigma}{\rho\nu}},$$

where L is a characteristic length of the problem.

[9] The Reynolds number is the ratio between inertial and viscous forces in a fluid. It can be computed from:

$$\mathrm{Re} = \frac{UL}{\nu},$$

where L is a characteristic length and U is a characteristic velocity of the problem.

h	$\|\|\mathbf{e}_u\|\|$	$\|\|\nabla\mathbf{e}_u\|\|$	$\|\|\mathbf{e}_b\|\|$	$\|\|\nabla\times\mathbf{e}_b\|\|$
2^{-2}	$3.32\cdot10^{-3}$ (-)	$5.62\cdot10^{-2}$ (-)	$1.90\cdot10^{-3}$ (-)	$4.20\cdot10^{-2}$ (-)
2^{-3}	$1.25\cdot10^{-3}$ (1.41)	$3.19\cdot10^{-2}$ (0.82)	$5.01\cdot10^{-4}$ (1.92)	$2.13\cdot10^{-2}$ (0.98)
2^{-4}	$3.73\cdot10^{-4}$ (1.74)	$1.56\cdot10^{-2}$ (1.03)	$1.59\cdot10^{-4}$ (1.66)	$1.08\cdot10^{-2}$ (0.98)
2^{-5}	$9.72\cdot10^{-5}$ (1.94)	$6.88\cdot10^{-3}$ (1.18)	$6.13\cdot10^{-5}$ (1.38)	$5.43\cdot10^{-3}$ (0.99)

Table 2.10: Numerical errors and rate of convergence in brackets. Shercliff's case with Ha=10.

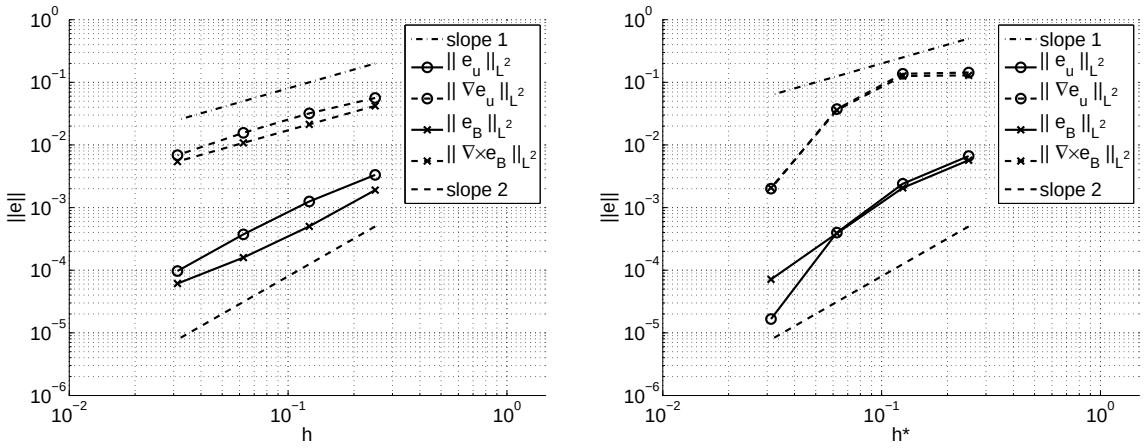
h^*	$\|\|\mathbf{e}_u\|\|$	$\|\|\nabla\mathbf{e}_u\|\|$	$\|\|\mathbf{e}_b\|\|$	$\|\|\nabla\times\mathbf{e}_b\|\|$
2^{-2}	$6.65\cdot10^{-3}$ (-)	$1.43\cdot10^{-1}$ (-)	$5.70\cdot10^{-3}$ (-)	$1.29\cdot10^{-1}$ (-)
2^{-3}	$2.40\cdot10^{-3}$ (1.47)	$1.37\cdot10^{-1}$ (0.06)	$2.06\cdot10^{-3}$ (1.47)	$1.26\cdot10^{-1}$ (0.03)
2^{-4}	$3.99\cdot10^{-4}$ (2.59)	$3.74\cdot10^{-2}$ (1.87)	$3.88\cdot10^{-4}$ (2.41)	$3.59\cdot10^{-2}$ (1.81)
2^{-5}	$1.66\cdot10^{-5}$ (4.59)	$2.00\cdot10^{-3}$ (4.22)	$7.13\cdot10^{-5}$ (2.44)	$2.07\cdot10^{-3}$ (4.12)

Table 2.11: Numerical errors and rate of convergence in brackets. Shercliff's case with Ha=100.

show that the method has optimal convergence properties independently of the coupling between the hydrodynamic and the magnetic problems.

On one hand, Shercliff's case has been solved for Ha = 10 and Reynolds number Re = 10 on a series of uniformly structured meshes composed of linear tetrahedral elements. The obtained results are shown in Table 2.10. The error norms computed correspond to the L^2-norm of the velocity error $||\mathbf{e}_u||$, the velocity gradient error $||\nabla\mathbf{e}_u||$, the magnetic induction error $||\mathbf{e}_b||$ and the curl of the magnetic induction error $||\nabla\times\mathbf{e}_b||$. The number in brackets is the rate of convergence. The same results have been plotted in Figure 2.7(a), where it can be clearly seen that the expected optimal order of convergence is achieved.

The same problem has been solved for Ha = 100 and Re = 10. In this case, the meshes used for the computations are also structured meshes of linear tetrahedra but concentrating the elements near the walls to be able to capture the sharper boundary layers that appear in this case. Therefore, the mesh size h^* used in the convergence study of this problem corresponds to the mesh size of a uniformly structured mesh with the same number of degrees of freedom. Similarly to the previous case, the computed error norms are displayed in Table 2.11. Figure 2.7(b) shows graphically the behavior of the error norms with h^*. It is also very clear that the proposed method has very good convergence properties in this case.



(a) Ha=10 (b) Ha=100

Figure 2.7: Convergence plots of velocity and induction error norms for Shercliff's case.

2.6.4 Classical MHD problems with analytical solution. Hunt's case

Hunt's problem simulates a fully developed flow in a channel with square section under an external magnetic field $\mathbf{b}_{\text{ext}}$. In this case, the Hartmann walls are considered to be perfectly conducting whereas the side walls are electrically insulating. Similarly to the previous Shercliff's case, this problem has an analytical solution in terms of Fourier series, whose expression can be found in [86] and Chapter 5.

The computational domain where this problem has been solved is the same that in Shercliff's case, see Section 2.6.3. Similarly, the boundary conditions on the inflow and outflow have been set to periodic conditions. In both the Hartmann walls and side walls, the hydrodynamic boundary conditions have been set as the non-slip condition, $\mathbf{u} = \mathbf{0}$. The magnetic boundary conditions for the side walls are the essential boundary conditions written in (2.10) because the walls are electrically insulating. However, the Hartmann walls in this case are suposed to be perfectly conducting, which changes the boundary conditions to be imposed. Therefore, the boundary conditions imposed in a perfectly conducting wall are the natural conditions,

$$\mathbf{b} \cdot \mathbf{n} = \mathbf{b}_{\text{ext}} \cdot \mathbf{n} \quad \text{and} \quad \mathbf{n} \times (\nabla \times \mathbf{b}) = \mathbf{0}.$$

The computation of the pressure gradient that drives the flow differs slightly from Shercliff's case,

$$\frac{\mathrm{d}p}{\mathrm{d}z} = \frac{KL^3}{\rho\nu^2 \mathrm{Re}}, \quad \text{where} \quad K = \frac{\mathrm{Ha}}{1 - 0.95598\mathrm{Ha}^{-1/2} - \mathrm{Ha}^{-1}}.$$

The Hartmann number Ha corresponds to the norm of the external magnetic field because all the physical properties of the problem have been set to one. The proposed stabilized finite element method has been used to solve Hunt's problem for two different Hartmann

h	$\|\|\mathbf{e}_u\|\|$	$\|\|\nabla\mathbf{e}_u\|\|$	$\|\|\mathbf{e}_b\|\|$	$\|\|\nabla\times\mathbf{e}_b\|\|$
2^{-2}	$8.98\cdot10^{-4}$ (-)	$1.63\cdot10^{-2}$ (-)	$5.25\cdot10^{-4}$ (-)	$9.97\cdot10^{-3}$ (-)
2^{-3}	$2.55\cdot10^{-4}$ (1.82)	$8.53\cdot10^{-3}$ (0.93)	$1.38\cdot10^{-4}$ (1.93)	$4.84\cdot10^{-3}$ (1.04)
2^{-4}	$6.92\cdot10^{-5}$ (1.88)	$4.08\cdot10^{-3}$ (1.06)	$4.21\cdot10^{-5}$ (1.71)	$2.42\cdot10^{-3}$ (1.00)
2^{-5}	$1.87\cdot10^{-5}$ (1.89)	$1.81\cdot10^{-3}$ (1.17)	$1.40\cdot10^{-5}$ (1.59)	$1.21\cdot10^{-3}$ (1.00)

Table 2.12: Numerical errors and rate of convergence in brackets. Hunt's case with Ha=10.

h^*	$\|\|\mathbf{e}_u\|\|$	$\|\|\nabla\mathbf{e}_u\|\|$	$\|\|\mathbf{e}_b\|\|$	$\|\|\nabla\times\mathbf{e}_b\|\|$
2^{-2}	$1.34\cdot10^{-3}$ (-)	$2.82\cdot10^{-2}$ (-)	$1.19\cdot10^{-3}$ (-)	$2.33\cdot10^{-2}$ (-)
2^{-3}	$1.54\cdot10^{-4}$ (3.12)	$8.62\cdot10^{-3}$ (1.71)	$1.16\cdot10^{-4}$ (3.36)	$6.09\cdot10^{-3}$ (1.94)
2^{-4}	$1.17\cdot10^{-5}$ (3.72)	$1.18\cdot10^{-3}$ (2.87)	$1.32\cdot10^{-5}$ (3.14)	$1.41\cdot10^{-3}$ (2.11)
2^{-5}	$2.13\cdot10^{-6}$ (2.46)	$8.61\cdot10^{-5}$ (3.78)	$1.86\cdot10^{-6}$ (2.83)	$1.52\cdot10^{-4}$ (3.21)

Table 2.13: Numerical errors and rate of convergence in brackets. Hunt's case with Ha=100.

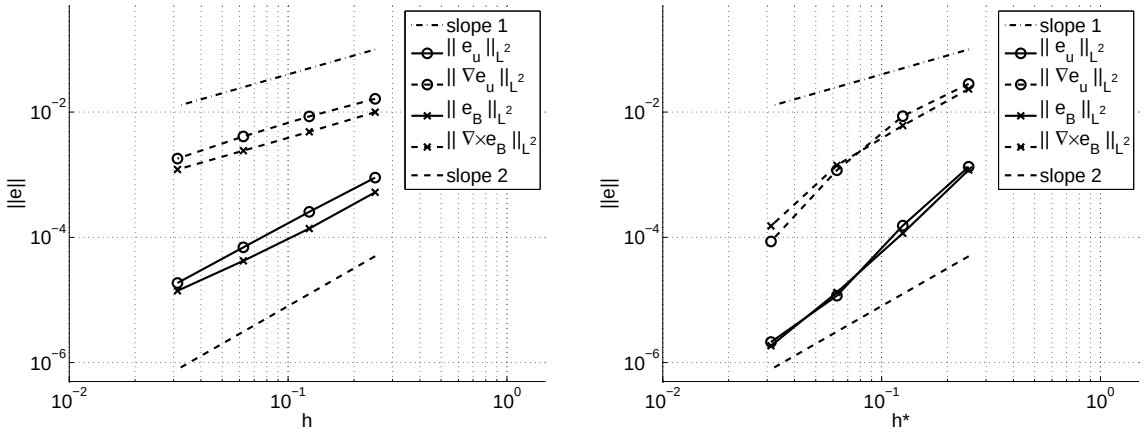
number, Ha = 10 and Ha = 100. In both cases, the Reynolds number has been set to Re = 10. Again, each case has been solved with several meshes in order to be able to compare the numerical behavior of the method with regard to its theoretical convergence properties.

On one hand, the case with Ha = 10 and Re = 10 has been computed on a series of uniformly structured meshes of linear tetrahedral elements. Table 2.12 shows the same error norms as in the previous test, obtained for every mesh size h. The rate of convergence between two consecutive meshes is shown in brackets. These results have been displayed in Figure 2.8(a). It shows that the results obtained are in good agreement with the theory.

On the other hand, Hunt's problem with Ha = 100 and Re = 10 has been solved on a collection of structured meshes of linear tetrahedral elements. However, the meshes in this case are not uniformly structured, but with element concentration near the walls, because the boundary layers are much thinner than in the previous case of Ha=10. Table 2.13 shows the computed error norms for the same quantities that in the case of Ha=10. Again, the mesh-size h^* corresponds to the equivalent mesh-size h of a uniformly structured mesh with the same number of degrees of freedom. In Figure 2.8(b), the numerical results have been plotted. The same conclusion as in the previous results applies here, that is, the numerical behavior of the proposed method is very good in terms of convergence.

2.6.5 Clogging of nozzles in steel casting processes.

This numerical example serves two objectives. First, we aim at solving a convection-dominated problem to check that the designed formulation has the expected behavior in



(a) Ha=10 (b) Ha=100

Figure 2.8: Convergence plots of velocity and induction error norms for Hunt's case.

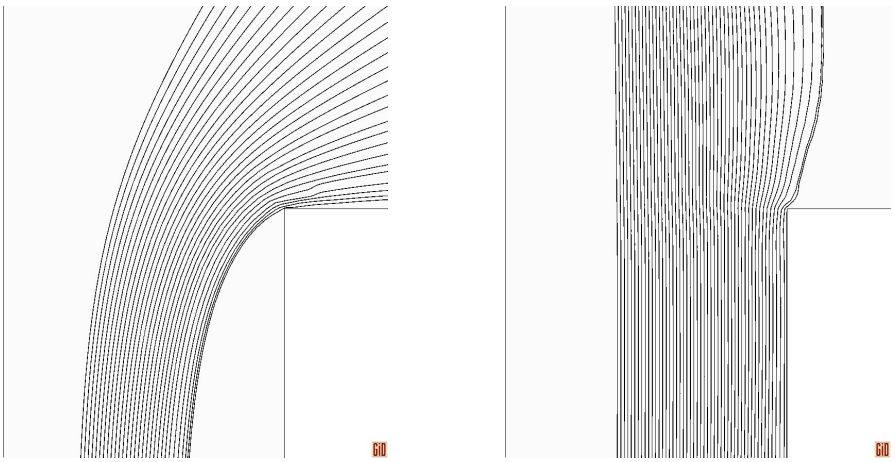
this regime and second, we want to solve a real problem with technological applications. The chosen problem is the clogging of nozzles in continuous casting of steel; see [54, 90] for a more detailed description. In continuous steel casting processes, the metal flow from the tundish to the mould is basically controlled by the tundish nozzle. Under particular flow conditions, there appears a serious problem which is the clogging of the nozzles due to the build-up of oxides. These build-ups can even block the flow through the nozzle. One of the solutions to this problem is the use of electromagnetic forces that reduce substantially the stagnation zone.

The L-shaped computational domain has been set as $[0, 0.00795] \times [0, 0.075] \cup [0, 0.200] \times [0.075, 0.525]$ due to the axial simmetry of the problem. This domain has been meshed with 21305 linear triangular elements and 10981 nodes, concentrating them in the nozzle zone. The hydrodynamic boundary conditions are a fixed velocity $\mathbf{u} = (0, -0.05)$ at the inlet $\{y = 0.525\}$, a free surface condition at the outlet $\{y = 0\}$, symmetric conditions $u_x = 0$ at the symmetry axes $\{x = 0\}$ and no-slip conditions $\mathbf{u} = (0, 0)$ elsewhere. The boundary conditions for the magnetic unknowns have been set as symmetric conditions $b_x = 0$ at $\{x = 0\}$, and a fixed radial magnetic induction field $b_x = b_{\text{ext},x}$ elsewhere. Finally, the gravity $\boldsymbol{g} = (0, -9.81)$ is set as an external body force.

The physical properties of the problem have been taken as $\rho = 7 \times 10^3$, $\nu = 2 \times 10^{-6}$, $\sigma = 10^6$ and $\mu_{\text{m}} = 8 \times 10^{-7}$. The flow parameters and the nozzle length lead to a Reynolds number of $\text{Re} \approx 20000$. The problem has been solved with two Hartmann numbers ($\text{Ha} = 1,\ 100$) to study the flow behavior under weak and strong external magnetic fields.

The obtained results are shown in Figure 2.9, where the velocity field streamlines in the nozzle zone are plotted. Figure 2.9(a) shows the case with $\text{Ha} = 1$ where it is clearly seen that the flow separates from the internal wall of the nozzle generating a stagnation zone which produces the clogging. However, increasing the external magnetic field reduces drastically this effect. Figure 2.9(b) shows the velocity streamlines for $\text{Ha} = 100$. In this case, the flow becomes attached in the nozzle zone, reducing the recirculation zones and

therefore eliminating the clogging of the nozzles.



(a) Ha=1 (b) Ha=100

Figure 2.9: Velocity streamlines in the nozzle zone.

2.7 Conclusions

A finite element approximation of the resistive MHD problem has been proposed in this work. The formulation falls within the category of stabilized finite element methods and, as such, is intended to avoid the need for using finite element approximations satisfying the compatibility conditions of the continuous problem and dealing with ranges of the physical parameters in which first order derivatives dominate second order ones.

Particular features of the formulation proposed are that it is of residual type, can be based on the VMS framework and the stabilization parameters are designed from the numerical analysis in [13] and Chapter 3, accounting for the coupling between the fluid and magnetic sub-problems. However, the most salient feature is that it allows to converge to singular solutions even when using a continuous Lagrangian approximation for the magnetic induction field. To our knowledge, this is the first time this is achieved. From the technical point of view, this possibility relies on the splitting of the residual of the finite element equation for the magnetic induction into two parts, one consisting of the gradient of the magnetic pseudo-pressure and the other the remaining terms. This is possible because the magnetic pseudo-pressure is zero for the continuous problem. The two resulting residuals are multiplied by stabilization parameters with different asymptotic behavior, a key point to mimic the correct functional setting of the continuous problem at the discrete level.

Support to assess the feasibility of our formulation arises from two facts. On the one hand, its applicability to the full transient and nonlinear problem has been demonstrated in a set of numerical experiments which have shown its excellent performance. On the other hand, we have performed a complete numerical analysis of the linearized and

stationary MHD problem in Chapter 3. In particular, optimal stability and convergence results have been obtained.

Chapter 3

Analysis of the stabilized formulation for the resistive MHD problem

In this chapter, we analyze the stabilized finite element formulation for the approximation of the resistive magnetohydrodynamics equations proposed in Chapter 2. The novelty of this formulation with respect to existing ones is the fact that it always converges to the physical solution, even when it is singular. We have performed a detailed stability and convergence analysis of the formulation in a simplified setting. From the convergence analysis, we infer that a particular type of meshes with a macro-element structure is needed, which can be easily obtained after a straight modification of any original mesh.

Let us stress that, even though it consists of the numerical analysis of the formulation presented in Chapter 2, this chapter is self-contained because it is based on the following scientific paper:

- Analysis of an unconditionally convergent stabilized finite element formulation for incompressible magnetohydrodynamics. S. Badia, R. Codina and R. Planas. *Submitted*, 2013.

3.1 Introduction

In this chapter, we analyze a numerical formulation for the approximation of the incompressible visco-resistive magnetohydrodynamics (MHD) system, which models incompressible viscous and electrically conducting fluids under electromagnetic fields (see [61]). Many conforming numerical approximations to this problem have been proposed so far. There are different equivalent formulations of the continuous magnetic sub-problem, namely saddle-point and (weighted) exact penalty formulations (see [111] and [4, 56, 58, 77, 78], respectively). The first one leads to a double-saddle-point formulation for the MHD system. A Galerkin finite element (FE) approximation of the resulting problem has been proposed and analyzed by Schötzau in [111]. It is well-known that saddle-point formulations require to choose particular mixed FE spaces satisfying discrete versions of the so-called inf-sup conditions (see e.g. [34]). Instead, a weighted exact penalty formulation has been used in [77]. This formulation allows to simplify

implementation issues but introduces a new complication, the definition of the weight function (see [58]). Alternative formulations have been proposed for a regularized version of the system, based on an exact penalty formulation [76]. These methods must be used with caution, since they converge to spurious solutions when the exact magnetic field is not smooth. Non-conforming approximations of discontinuous Galerkin type have been designed in [81]. They have good numerical properties, but the increase in CPU cost –degrees of freedom– of these formulations (with respect to conforming formulations) is severe for realistic large-scale applications. For the Maxwell equations alone, alternative approximations based on nodal Lagrangian FEs can be found e.g. in [31,32,64]. We refer to [6, 82] for the application of residual-free bubbles to MHD and [22] for a two-level stabilization method with Scott-Vogelius FEs.

Since the resistive MHD system loses coercivity as the Reynolds and magnetic Reynolds numbers increase, i.e. convection-type terms become dominant, the previous formulations are unstable unless the mesh size is sufficientely refined, which is impractical. In order to treat this problem, as well as the previous ones, some stabilized FE formulations have been proposed for resistive MHD in [23, 24, 53, 54, 72, 73, 112]. These formulations are appealing in terms of implementation issues, since arbitrary order Lagrangian FE spaces can be used for all the unknowns and include convection-type stabilization. However, these formulations are based on the regularized functional setting of the problem, and so, restricted to smooth or convex domains (see [58]). They are accurate for regular magnetic solutions but tend to spurious (unphysical) solutions otherwise. A further improvement is the formulation in Chapters 2 and 4, which always converges to the exact (physical) solution, even when it is singular. In this chapter, we carry out a numerical analysis of this formulation in order to prove stability and unconditional convergence in the correct norms while keeping optimal *a priori* error estimates for smooth solutions.

The outline of this chapter is the following. First, the MHD problem of interest is stated in Section 3.2. The stabilized FE formulation is introduced in Section 3.3. We present a detailed stability and convergence analysis for the stationary and linearized problem in Sections 3.4 and 3.5 respectively. The possible extension of these results to nonlinear problem is analyzed in Section 3.6. Section 3.7 is devoted to the numerical experiments. We finish the chapter by drawing some conclusions in Section 3.8.

3.2 Problem statement

3.2.1 The strong form

The incompressible visco-resistive MHD system of partial differential equations consists of the Navier-Stokes equations coupled to the (simplified) Maxwell equations via the Lorentz force. A linearized version of this system of equations reads as follows: find a velocity field $\mathbf{u}(\mathbf{x}, t)$, a (kinematic) pressure $p(\mathbf{x}, t)$, an induced magnetic field $\mathbf{b}(\mathbf{x}, t)$ and

a magnetic pseudo-pressure $r(\mathbf{x},t)$ such that

$$\begin{aligned} \mathbf{a}\cdot\nabla\mathbf{u}-\nu\Delta\mathbf{u}+\nabla p-(\nabla\times\mathbf{b})\times\mathbf{d} &= \mathbf{f}_{\mathrm{u}}, && (3.1a)\\ \nabla\cdot\mathbf{u} &= g_{\mathrm{u}}, && (3.1b)\\ \lambda\nabla\times(\nabla\times\mathbf{b})+\nabla r-\nabla\times(\mathbf{u}\times\mathbf{d}) &= \mathbf{f}_{\mathrm{b}}, && (3.1c)\\ \nabla\cdot\mathbf{b} &= g_{\mathrm{b}}, && (3.1d) \end{aligned}$$

in $(\mathbf{x},t)\in\Omega$, where $\Omega\subset\mathbb{R}^d$ is the spatial open bounded domain filled by the fluid (assumed polyhedral in the finite element approximation), d being the space dimension. With regard to the physical parameters that describe the fluid, ρ is its density, μ_{f} the fluid viscosity, μ_{m} the magnetic permeability and σ the electric conductivity. Further, $\nu:=\mu_{\mathrm{f}}\rho^{-1}$ and $\lambda:=(\rho\mu_{\mathrm{m}}^2\sigma)^{-1}$. In this work, we consider all physical properties constant; we refer to [12] for discontinuous physical coefficients for the electromagnetic problem. $(\mathbf{a},\mathbf{d})$ is the point around which the system has been linearized. In order to recover the nonlinear case, $(\mathbf{a},\mathbf{d})$ must be replaced by $(\mathbf{u},\varrho\mathbf{b})$, where $\varrho:=(\rho\mu_{\mathrm{m}})^{-1}$. Regularity conditions for $(\mathbf{a},\mathbf{d})$ are discussed later. $\mathbf{f}_{\mathrm{u}}$ and $\mathbf{f}_{\mathrm{b}}$ are the forcing terms, $\mathbf{f}_{\mathrm{b}}$ being solenoidal. Let us remark the fact that we have introduced g_{b} and g_{u}. We are only interested in the case when both functions are zero, but this generalization will allow us to re-use the following stability results in the convergence analysis. These equations must be supplemented with appropriate boundary conditions.

In order to introduce the boundary conditions, let us consider two disjoint partitions of the domain boundary $\Gamma\equiv\partial\Omega$:

$$\Gamma=\Gamma_{\mathrm{f,e}}\cup\Gamma_{\mathrm{f,n}},\qquad\qquad \Gamma=\Gamma_{\mathrm{m,e}}\cup\Gamma_{\mathrm{m,n}},$$

where the first subscript denotes the subproblem (f for fluid and m for magnetic) and the second one the type of boundary condition (e for essential and n for natural). Then, the fluid sub-problem is supplemented with the standard boundary conditions:

$$\mathbf{u}=\mathbf{u}_\Gamma\quad\text{on }\Gamma_{\mathrm{f,e}},\qquad -p\mathbf{n}+\nu\mathbf{n}\cdot\nabla\mathbf{u}=\boldsymbol{\sigma}_{n,\Gamma}\quad\text{on }\Gamma_{\mathrm{f,n}},$$

where $\mathbf{u}_\Gamma(\mathbf{x},t)$ and $\boldsymbol{\sigma}_{n,\Gamma}(\mathbf{x},t)$ are the trace and normal stress prescribed; $\mathbf{n}(\mathbf{x})$ denotes the normal vector on Γ pointing outwards from Ω. With regard to the magnetic sub-problem, we consider the set of ideal boundary conditions:

$$\mathbf{n}\times\mathbf{b}=\mathbf{n}\times\mathbf{b}_\Gamma,\quad r=0\quad\text{on }\Gamma_{\mathrm{m,e}},\qquad \mathbf{n}\cdot\mathbf{b}=\mathbf{n}\cdot\mathbf{b}_\Gamma,\quad \mathbf{n}\times(\nabla\times\mathbf{b})=\mathbf{J}_\Gamma\quad\text{on }\Gamma_{\mathrm{m,n}},$$

where $\mathbf{n}\cdot\mathbf{b}_\Gamma$ and $\mathbf{n}\times\mathbf{b}_\Gamma$ are the normal and tangential traces to be prescribed; clearly, $\mathbf{J}_\Gamma\cdot\mathbf{n}$ must vanish.

Taking the divergence of (3.1c), we easily infer that $r=0$. Unfortunately, this is not true in general for the discretized system. For numerical purposes, it is more suitable to explicitly enforce (3.1d) via a Lagrange multiplier, the magnetic pseudo-pressure $r(\mathbf{x},t)$.

3.2.2 The weak form

Let us introduce some notation to set up the weak form of the problem. As usual, Sobolev spaces of functions whose derivatives of order up to m belong to $L^2(\Omega)$ are denoted by

$H^m(\Omega)$; $H^1_0(\Omega)$ is the subspace of $H^1(\Omega)$ of functions vanishing on $\partial\Omega$. The space of vector functions with components in $L^2(\Omega)$ and with divergence also in $L^2(\Omega)$ is denoted $H(\mathrm{div};\Omega)$; if the components are $L^2(\Omega)$ and the curl is in $L^2(\Omega)^d$ the space is denoted $H(\mathbf{curl};\Omega)$. $H(\mathrm{div}\,0;\Omega)$ is the subspace of $L^2(\Omega)^d$ of divergence free vector functions.

The inner product of $f, g \in L^2(\Omega)$ is represented as (f,g), whereas $\langle f,g\rangle$ is used to denote the integral $\int_\Omega fg$ whenever it makes sense; this in particular applies for the duality between $H^1_0(\Omega)$ and its topological dual $H^{-1}(\Omega)$. The same notation is used for both scalar and vector valued functions. Given a normed functional space X, its norm is written as $\|\cdot\|_X$, with the abbreviations $\|\cdot\|_{L^2(\Omega)} \equiv \|\cdot\|$, $\|\cdot\|_{H^m(\Omega)} \equiv \|\cdot\|_m$, $\|\cdot\|_{H^{-1}(\Omega)} \equiv \|\cdot\|_{-1}$, $\|\cdot\|_{H(\mathbf{curl};\Omega)} \equiv \|\cdot\|_{\mathbf{curl}}$. Finally, the symbol $\lesssim$ is used to denote $\leq$ up to positive constants that do not depend on numerical or physical parameters.

Let us consider the functional setting in which the system of equations (3.1) is well-posed. For the sake of clarity, we will consider homogeneous essential boundary conditions; in any case, the extension to the most general case is standard. We introduce the vectorial functional spaces:

$$\begin{aligned}\boldsymbol{V} &= \{\mathbf{v} \in H^1(\Omega)^d \,\text{such that}\, \mathbf{v} = \mathbf{0}\,\text{on}\,\Gamma\},\\ \boldsymbol{C} &= \{\mathbf{c} \in H(\mathbf{curl};\Omega)\,\text{such that}\, \mathbf{n}\times\mathbf{c} = \mathbf{0}\,\text{on}\,\Gamma\},\end{aligned}$$

for the velocity and magnetic field functions, respectively. Further, the space for fluid pressures is $Q \equiv L^2_0(\Omega)$ and the one for magnetic pseudo-pressures $S \equiv H^1_0(\Omega)$. Now, we can state the stationary MHD problem at hand in its weak form as follows: find $\mathbf{u} \in \boldsymbol{V}$, $\mathbf{b} \in \boldsymbol{C}$, $p \in Q$ and $r \in S$ such that

$$\langle \mathbf{a}\cdot\nabla\mathbf{u},\mathbf{v}\rangle + (\nu\nabla\mathbf{u},\nabla\mathbf{v}) - (p,\nabla\cdot\mathbf{v}) - \langle(\nabla\times\mathbf{b})\times\mathbf{d},\mathbf{v}\rangle = \langle\mathbf{f}_\mathrm{u},\mathbf{v}\rangle, \tag{3.2a}$$

$$(q,\nabla\cdot\mathbf{u}) = \langle g_\mathrm{u}, q\rangle, \tag{3.2b}$$

$$(\lambda\nabla\times\mathbf{b},\nabla\times\mathbf{c}) - \langle\nabla\times(\mathbf{u}\times\mathbf{d}),\mathbf{c}\rangle + (\nabla r,\mathbf{c}) = \langle\mathbf{f}_\mathrm{b},\mathbf{c}\rangle, \tag{3.2c}$$

$$-(\nabla s,\mathbf{b}) = \langle g_\mathrm{b}, s\rangle, \tag{3.2d}$$

for any $(\mathbf{v},\mathbf{c},q,s) \in \boldsymbol{V}\times\boldsymbol{C}\times Q\times S$. Let us show that $r \equiv 0$ in (3.2). Taking $\mathbf{c} = \nabla r$ (which clearly belongs to $\boldsymbol{C}$) in (3.2c), and using the fact that $\nabla\times\nabla r = \mathbf{0}$ and $\nabla\cdot\mathbf{f}_\mathrm{b} = 0$ a.e. in Ω, we obtain $\|\nabla r\| = 0$. Since r vanishes on $\partial\Omega$, it implies $r \equiv 0$ a.e. in Ω by virtue of Poincaré's inequality. We refer to [78, Propositions 3.4 and 3.5] for the completion of the proof.

Let us re-write system (3.2) in compact manner as:

$$\mathcal{A}((\mathbf{u},\mathbf{b},p,r),(\mathbf{v},\mathbf{c},q,s)) = \mathcal{F}(\mathbf{v},\mathbf{c},q,s), \quad \forall(\mathbf{v},\mathbf{c},q,s) \in \boldsymbol{V}\times\boldsymbol{C}\times Q\times S.$$

with the obvious definition of the bilinear form $\mathcal{A}$ and the linear form $\mathcal{F}$.

In the following, we consider infima and suprema with respect to functions in some space different from the zero function. For the sake of brevity, we will omit the fact that the zero function cannot be picked. Problem (3.2) is well-posed due to the inf-sup conditions

$$\inf_{q\in Q}\sup_{\mathbf{v}\in\boldsymbol{V}}\frac{(q,\nabla\cdot\mathbf{v})}{\|q\|\|\mathbf{v}\|_1} \geq \beta_\mathrm{f} > 0, \qquad \inf_{s\in S}\sup_{\mathbf{c}\in\boldsymbol{C}}\frac{(\nabla s,\mathbf{c})}{\|s\|_1\|\mathbf{c}\|_{H(\mathbf{curl};\Omega)}} \geq \beta_\mathrm{m} > 0, \tag{3.3}$$

that are known to be true at the continuous level, as well as the Poincaré-Friedrichs inequalities

$$\begin{aligned} \|\mathbf{v}\|_1 &\leq C_{\mathrm{P},1}\|\nabla\mathbf{v}\| && \text{for } \mathbf{v} \in H_0^1(\Omega)^d, \\ \|\mathbf{c}\|_{H(\mathbf{curl};\Omega)} &\leq C_{\mathrm{P},2}\|\nabla\times\mathbf{c}\| && \text{for } \mathbf{c} \in \boldsymbol{C} \cap H(\operatorname{div} 0;\Omega), \end{aligned} \tag{3.4}$$

e.g., see [97, Corollary 3.51].

From the standard theory of saddle-point problems, well-posedness of the MHD system (3.2) is proved in the next theorem.

Theorem 3.1. *The following inf-sup condition is satisfied,*

$$\inf_{(\mathbf{u},\mathbf{b},p,r)\in\boldsymbol{V}\times\boldsymbol{C}\times Q\times S} \sup_{(\mathbf{v},\mathbf{c},q,s)\in\boldsymbol{V}\times\boldsymbol{C}\times Q\times S} \frac{\mathcal{A}((\mathbf{u},\mathbf{b},p,r),(\mathbf{v},\mathbf{c},q,s))}{\|(\mathbf{u},\mathbf{b},p,r)\|_{\mathrm{Gal}}\times\|(\mathbf{v},\mathbf{c},q,s)\|_{\mathrm{Gal}}} \geq \beta > 0. \tag{3.5}$$

As a consequence, formulation (3.2) is well-posed.

Proof: We can easily check that $\mathcal{A}((\mathbf{u},\mathbf{b},0,0),(\mathbf{v},\mathbf{c},0,0))$ is a bilinear, continuous and coercive form when it is restricted to $\boldsymbol{V}\cap H(\operatorname{div} 0;\Omega)\times\boldsymbol{C}\cap H(\operatorname{div} 0;\Omega)$. It is a direct consequence of the Poincaré-Friedrichs inequalities (3.4). This result, together with the inf-sup conditions (3.3) are necessary and sufficient conditions for proving (3.5) (see [68, Proposition 2.36]). We know from the theory of saddle-point problems that (3.2) is well-posed if and only if condition (3.5) is satisfied (see [68, Theorem 2.34]). □

3.3 A stabilized FE formulation suitable for singular magnetic solutions

Let us present now the spatial discretization we propose. Let $\mathcal{T}_h = \{K\}$ be a FE partition of the domain Ω. For simplicity we assume Ω polyhedral and $\mathcal{T}_h$ quasi-uniform, of diameter h. Summation over all the element domains K is denoted as $\sum_K$. Finite element spaces and FE functions are identified with the subscript h. Only conforming approximations are considered, i.e., the FE spaces where the unknowns are sought are $\boldsymbol{V}_h \subset \boldsymbol{V}$, $\boldsymbol{C}_h \subset \boldsymbol{C}$, $Q_h \subset Q$ and $S_h \subset S$. In particular, we will use $\mathcal{C}^0$ Lagrangian finite element interpolations of an arbitrary order for all the unknowns. Given two functions f and g piecewise polynomial on each $K \in \mathcal{T}_h$, we define $(f,g)_h := \sum_K \int_K f\,g$ and $\|f\|_h := (f,f)_h^{1/2}$.

Since we assume $\mathcal{T}_h$ quasi-uniform, the following inverse inequality holds:

$$\|\nabla v_h\|_{L^2(K)} \leq \frac{C_{\mathrm{inv}}}{h}\|v_h\|_{L^2(K)}, \qquad K \in \mathcal{T}_h, \tag{3.6}$$

for a positive constant C_{inv} and for all piecewise polynomial functions v_h.

We consider a residual-based stabilized FE formulation for the MHD problem. This type of formulation does not change the statement of the continuous problem but modifies the way the discretization is performed. Instead of considering only those terms that come from a Galerkin discretization, this type of formulation includes additional terms, that are always proportional to some residual, and so, consistent. In order for

this approach to be effective, the new terms must provide stability over the Lagrange multiplier-type unknowns which allows one to circumvent discrete inf-sup conditions, as well as convection stability (see e.g. [50]). In order to obtain a numerical algorithm suitable for singular solutions and avoiding the need to define weighting functions that require information about the placement of singularities, we stick to the double saddle-point formulation (3.2). The resulting method has been stated in Algorithm 3.1; we refer to Chapter 2 for a motivation of the method.

Algorithm 3.1: Stabilized FE formulation

Find $(\mathbf{u}_h, \mathbf{b}_h, p_h, r_h) \in \boldsymbol{V}_h \times \boldsymbol{C}_h \times Q_h \times S_h$ such that

$$\mathcal{A}_{\text{stab}}((\mathbf{u}_h, \mathbf{b}_h, p_h, r_h), (\mathbf{v}_h, \mathbf{c}_h, q_h, s_h)) = \mathcal{F}_{\text{stab}}(\mathbf{v}_h, \mathbf{c}_h, q_h, s_h), \tag{3.7}$$

for any $(\mathbf{v}_h, \mathbf{c}_h, q_h, s_h) \in \boldsymbol{V}_h \times \boldsymbol{C}_h \times Q_h \times S_h$, where

$$\begin{aligned}\mathcal{A}_{\text{stab}}((\mathbf{u}_h, \mathbf{b}_h, p_h, r_h), (\mathbf{v}_h, \mathbf{c}_h, q_h, s_h)) =& \mathcal{A}((\mathbf{u}_h, \mathbf{b}_h, p_h, r_h), (\mathbf{v}_h, \mathbf{c}_h, q_h, s_h)) \\ &+ \mathcal{S}((\mathbf{u}_h, \mathbf{b}_h, p_h, r_h), (\mathbf{v}_h, \mathbf{c}_h, q_h, s_h)),\end{aligned}$$

with the stabilization terms

$$\begin{aligned}\mathcal{S}((\mathbf{u}_h, \mathbf{b}_h, p_h, r_h), (\mathbf{v}_h, \mathbf{c}_h, q_h, s_h)) =& (\tau_1(X_{\text{u}}(\mathbf{u}_h, p_h, \mathbf{b}_h) - \nu\Delta\mathbf{u}_h), X_{\text{u}}(\mathbf{v}_h, q_h, \mathbf{c}_h) + \nu\Delta\mathbf{v}_h)_h \\ &+ (\tau_2\nabla\cdot\mathbf{u}_h, \nabla\cdot\mathbf{v}_h) \\ &+ (\tau_3(\nabla\times(\mathbf{u}_h\times\mathbf{d}) - \lambda\nabla\times(\nabla\times\mathbf{b}_h)), \nabla\times(\mathbf{v}_h\times\mathbf{d}) + \lambda\nabla\times(\nabla\times\mathbf{c}_h))_h \\ &+ (\tau_4\nabla r_h, \nabla s_h) + (\tau_5\nabla\cdot\mathbf{b}_h, \nabla\cdot\mathbf{c}_h),\end{aligned}$$

and

$$\begin{aligned}\mathcal{F}_{\text{stab}}(\mathbf{v}_h, \mathbf{c}_h, q_h, s_h) =& \mathcal{F}(\mathbf{v}_h, \mathbf{c}_h, q_h, s_h) + (\tau_1\mathbf{f}_{\text{u}}, X_{\text{u}}(\mathbf{v}_h, q_h, \mathbf{c}_h) + \nu\Delta\mathbf{v}_h)_h + (\tau_2 g_{\text{u}}, \nabla\cdot\mathbf{v}_h) \\ &- (\tau_3\mathbf{f}_{\text{b}}, \nabla\times(\mathbf{v}_h\times\mathbf{d}) + \lambda\nabla\times(\nabla\times\mathbf{c}_h))_h + (\tau_5 g_{\text{b}}, \nabla\cdot\mathbf{c}_h).\end{aligned}$$

We use the notation $X_{\text{u}}(\mathbf{v}_h, q_h, \mathbf{c}_h) := \mathbf{a}\cdot\nabla\mathbf{v}_h + \nabla q_h - (\nabla\times\mathbf{c}_h)\times\mathbf{d}$. The stabilization parameters have the following expressions within each element K:

$$\begin{aligned}&\tau_1 := (\alpha)^{-1}\left(1 + \tfrac{\phi}{\sqrt{\alpha\gamma}}\right)^{-1}, \quad \tau_2 := c_5\tfrac{h^2}{\tau_1}, \quad \tau_3 := \gamma^{-1}\left(1 + \tfrac{\phi}{\sqrt{\alpha\gamma}}\right)^{-1}, \\ &\tau_4 := c_6\tfrac{L_0^2}{\lambda}, \qquad\qquad \tau_5 := c_7\tfrac{h^2\lambda}{L_0^2},\end{aligned}$$

with

$$\alpha := c_1\frac{\|\mathbf{a}\|_{L^\infty(\Omega)}}{h} + c_2\frac{\nu}{h^2}, \qquad \phi := c_3\frac{\|\mathbf{d}\|_{L^\infty(\Omega)}}{h}, \qquad \gamma := c_4\frac{\lambda}{h^2}.$$

$c_1, \ldots, c_5$ are algorithmic constants that must satisfy $c_1 > \frac{2}{C_{\text{inv}}^2}$ and $c_3 > \frac{2}{C_{\text{inv}}^2}$ and L_0 is a length scale of the problem.

We consider the following norms that will be used hereafter:

$$\|(\mathbf{v},\mathbf{c},q,s)\|_{\mathrm{Gal}} = \nu^{\frac{1}{2}}\|\mathbf{v}\|_1 + \lambda^{\frac{1}{2}}\|\mathbf{c}\|_{\mathbf{curl}} + \frac{1}{\nu^{\frac{1}{2}}}\|q\| + \frac{L_0}{\lambda^{\frac{1}{2}}}\|s\|_1, \tag{3.8a}$$

$$|(\mathbf{v},\mathbf{c},q,s)|_{\mathrm{stab}} = \|\tau_1^{\frac{1}{2}} X_{\mathrm{u}}(\mathbf{v},q,\mathbf{c})\|_h + \|\tau_2^{\frac{1}{2}}\nabla\cdot\mathbf{v}\| + \|\tau_3^{\frac{1}{2}}\nabla\times(\mathbf{v}\times\mathbf{d})\| + \|\tau_4^{\frac{1}{2}}\nabla s\| + \|\tau_5^{\frac{1}{2}}\nabla\cdot\mathbf{c}\|, \tag{3.8b}$$

$$\|(\mathbf{v},\mathbf{c},q,s)\|_{\mathrm{stab,w}} = \nu^{\frac{1}{2}}\|\mathbf{v}\|_1 + \lambda^{\frac{1}{2}}\|\nabla\times\mathbf{c}\| + |(\mathbf{v},\mathbf{c},q,s)|_{\mathrm{stab}}, \tag{3.8c}$$

$$\|(\mathbf{v},\mathbf{c},q,s)\|_{\mathrm{stab,s}} = \frac{\lambda^{\frac{1}{2}}}{L_0}\|\mathbf{c}\| + \frac{1}{\nu^{\frac{1}{2}}}\|q\| + \|(\mathbf{v},\mathbf{c},q,s)\|_{\mathrm{stab,w}}, \tag{3.8d}$$

where L_0 is a length scale of the problem (see [11] for a discussion about its meaning and possible ways to choose it).

Norm (3.8a) is the continuous norm in which the problem is well-posed, and the Galerkin norm when using stable mixed FEs. The extra stability due to the form $\mathcal{S}$ is given by the semi-norm (3.8b). Norm (3.8c) is the one that adds the stability that comes from the coercive terms in $\mathcal{A}$ to the one that comes from $\mathcal{S}$. Finally, norm (3.8d) is the sum of the weak stability norm $\|\cdot\|_{\mathrm{stab,w}}$ terms and the additional terms that are present in $\|\cdot\|_{\mathrm{Gal}}$. Obviously, $\|(\mathbf{v},\mathbf{c},q,s)\|_{\mathrm{Gal}} \leq \|(\mathbf{v},\mathbf{c},q,s)\|_{\mathrm{stab,s}}$.

3.4 Stability analysis

In this section, we analyze the stability properties of the stabilized FE formulation in Algorithm 3.1. First, we prove coercivity of the stabilized form $\mathcal{A}_{\mathrm{stab}}$ in the weak stabilized norm. Next, we prove a weak inf-sup which includes $\|\mathbf{b}\|$ and $\|p\|$ control. We attain this result relying on the continuous inf-sup condition (3.5) proved in Theorem 3.1. We absorb the length scale coefficients in the constants, since it clarifies the exposition.

Lemma 3.1. *Forms $\mathcal{A}_{\mathrm{stab}}$ and $\mathcal{A}$ satisfy the following properties:*

(i) Weak coercivity of $\mathcal{A}_{\mathrm{stab}}$: Assuming that $\mathbf{a}\in H^1(\Omega)^d$ and $\mathbf{d}\in H(\mathbf{curl};\Omega)$, it holds

$$\frac{1}{2}\|(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h)\|^2_{\mathrm{stab,w}} \leq \mathcal{A}_{\mathrm{stab}}((\mathbf{u}_h,\mathbf{b}_h,p_h,r_h),(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h)),$$

for any $(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h)\in \boldsymbol{V}_h\times\boldsymbol{C}_h\times Q_h\times S_h$.

(ii) Weak inf-sup condition for $\mathcal{A}$: Assuming that $\mathbf{a}\in H^1(\Omega)^d$ and $\mathbf{d}\in L^{d+\epsilon}(\Omega)^d$, it holds

$$\|(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h)\|_{\mathrm{Gal}} - \sigma\|(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h)\|_{\mathrm{stab,w}} \lesssim \sup_{(\mathbf{v}_h,s_h)\in\boldsymbol{V}_h\times S_h} \frac{\mathcal{A}((\mathbf{u}_h,\mathbf{b}_h,p_h,r_h),(\mathbf{v}_h,\mathbf{0},0,s_h))}{\|(\mathbf{v}_h,\mathbf{0},0,s_h)\|_{\mathrm{Gal}}}$$

for any $(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h)\in\boldsymbol{V}_h\times\boldsymbol{C}_h\times Q_h\times S_h$, where

$$\sigma = c_\sigma\left(1+\frac{h}{\nu^{\frac{1}{2}}\tau_1^{\frac{1}{2}}} + \frac{\|\mathbf{d}\|_{L^{d+\epsilon}(\Omega)}}{\sqrt{\nu\lambda}}\right)$$

for $\epsilon \in (0,3)$ arbitrary small and c_σ a positive constant independent of physical and numerical parameters.

Proof: Let us prove the first result. Using the equality $(a+b)(a-b) = a^2 - b^2$, it is straightforward to check that

$$\begin{aligned}\mathcal{A}_{\text{stab}}((\mathbf{u}_h, \mathbf{b}_h, p_h, r_h), (\mathbf{u}_h, \mathbf{b}_h, p_h, r_h)) = \nu\|\nabla \mathbf{u}_h\|^2 + \lambda\|\nabla \times \mathbf{b}_h\|^2 + |(\mathbf{u}_h, \mathbf{b}_h, p_h, r_h)|^2_{\text{stab}} \\ - \|\tau_1^{\frac{1}{2}}\nu\Delta \mathbf{u}_h\|_h^2 - \|\tau_3^{\frac{1}{2}}\lambda\nabla \times (\nabla \times \mathbf{b}_h)\|_h^2.\end{aligned}$$

We have used the relation $\int_\Omega ((\nabla \times \mathbf{b}) \times \mathbf{d}) \cdot \mathbf{u} = -\int_\Omega \nabla \times (\mathbf{u} \times \mathbf{d}) \cdot \mathbf{b}$ that holds for any $\mathbf{b}$, $\mathbf{d} \in H(\mathbf{curl}; \Omega)$ and $\mathbf{u} \in H^1(\Omega)^3$. The last two terms can be bounded for FE functions, by using the inverse inequality (3.6) elementwise:

$$\|\tau_1^{\frac{1}{2}}\nu\Delta \mathbf{u}_h\|_h^2 + \|\tau_3^{\frac{1}{2}}\lambda\nabla \times (\nabla \times \mathbf{b}_h)\|_h^2 \leq \frac{1}{2}(\nu\|\nabla \mathbf{u}_h\|^2 + \lambda\|\nabla \times \mathbf{b}_h\|^2),$$

since $\frac{\tau_1\nu}{h^2} \leq \frac{1}{2}$ and $\frac{\tau_3\lambda}{h^2} \leq \frac{1}{2}$ from the definition of the stabilization parameters. Full control over $\mathbf{u}_h$ in $H^1(\Omega)^3$ is consequence of Poincaré's inequality. It proves the weak coercivity.

In order to prove the weak inf-sup condition, let us invoke the continuous inf-sup condition (3.5) for the full MHD system, which can be stated as follows: for any $(\mathbf{u}_h, \mathbf{b}_h, p_h, r_h) \in \boldsymbol{V}_h \times \boldsymbol{C}_h \times Q_h \times S_h$ there exists $(\mathbf{v}, \mathbf{c}, q, s) \in \boldsymbol{V} \times \boldsymbol{C} \times Q \times S$ with unit Galerkin norm such that

$$\|(\mathbf{u}_h, \mathbf{b}_h, p_h, r_h)\|_{\text{Gal}} \leq \mathcal{A}((\mathbf{u}_h, \mathbf{b}_h, p_h, r_h), (\mathbf{v}, \mathbf{c}, q, s)).$$

Now, we have that

$$\begin{aligned}\mathcal{A}((\mathbf{u}_h, \mathbf{b}_h, p_h, r_h), (\mathbf{v}, \mathbf{c}, q, s)) =& \mathcal{A}((\mathbf{u}_h, \mathbf{b}_h, p_h, r_h), (\varepsilon_h(\mathbf{v}), \mathbf{c}, q, \varepsilon_h(s))) \\ &+ \mathcal{A}((\mathbf{u}_h, \mathbf{b}_h, p_h, r_h), (\pi_h(\mathbf{v}), \mathbf{0}, 0, \pi_h(s))),\end{aligned} \tag{3.9}$$

where $\varepsilon_h(v) := v - \pi_h(v)$ and $\pi_h(v)$ is a continuous FE interpolant in $H^1(\Omega)$ of a function v (scalar or vector valued) with optimal interpolation properties that preserves null traces, e.g. the Scott-Zhang interpolant [33].

Let us define the following functions:

$$q(\delta, d) = \begin{cases} \frac{2}{1-2\delta}, & \text{for } d = 2 \\ \frac{12-2\delta}{4-\delta}, & \text{for } d = 3 \end{cases}, \qquad q(\delta, d)' = \begin{cases} \frac{1}{\delta}, & \text{for } d = 2 \\ 6 - \delta, & \text{for } d = 3 \end{cases}.$$

where $\delta \in (0, \frac{1}{2})$. We note that $\frac{1}{q(\delta,d)} + \frac{1}{q(\delta,d)'} = \frac{1}{2}$. Now, we can prove the continuity result

$$(\nabla \times (\mathbf{u} \times \mathbf{d}), \mathbf{c}) \lesssim \|\mathbf{u}\|_{L^{q(\delta,d)'}(\Omega)}\|\mathbf{d}\|_{L^{q(\delta,d)}(\Omega)}\|\nabla \times \mathbf{c}\| \lesssim \|\mathbf{u}\|_1\|\mathbf{d}\|_{L^{q(\delta,d)}(\Omega)}\|\nabla \times \mathbf{c}\|,$$

which holds for any $\delta \in (0, \frac{1}{2})$.We have used the compact imbedding $H^1(\Omega) \hookrightarrow L^q(\Omega)$ that holds for $q \in [1, \infty)$ in dimension two and for $q \in [1, 6)$ in dimension three. Noting that $q(\delta, d)'$ belongs to these intervals in both dimensions, we prove the result. Now, we

bound the first term in the RHS of (3.9) as follows:

$$
\begin{aligned}
&\mathcal{A}((\mathbf{u}_h,\mathbf{b}_h,p_h,r_h),(\varepsilon_h(\mathbf{v}),\mathbf{c},q,\varepsilon_h(s)))\\
&\quad\lesssim \|X_{\mathrm{u}}(\mathbf{u}_h,p_h,\mathbf{b}_h)\|\|\varepsilon_h(\mathbf{v})\|+\|\nabla\cdot\mathbf{b}_h\|\|\varepsilon_h(s)\|+\nu\|\nabla\mathbf{u}_h\|\|\nabla\varepsilon_h(\mathbf{v})\|\\
&\qquad+\|\nabla\cdot\mathbf{u}_h\|\|q\|+\lambda\|\nabla\times\mathbf{b}_h\|\|\nabla\times\mathbf{c}\|+\|\nabla r_h\|\|\mathbf{c}\|+\|\mathbf{u}_h\|_1\|\mathbf{d}\|_{L^{q(\delta,d)}(\Omega)}\|\nabla\times\mathbf{c}\|\\
&\quad\lesssim \frac{h}{\nu^{\frac{1}{2}}}\|X_{\mathrm{u}}(\mathbf{u}_h,p_h,\mathbf{b}_h)\|+h\lambda^{\frac{1}{2}}\|\nabla\cdot\mathbf{b}_h\|+(1+\frac{\|\mathbf{d}\|_{L^{q(\delta,d)}(\Omega)}}{\sqrt{\nu\lambda}})\nu^{\frac{1}{2}}\|\mathbf{u}_h\|_1\\
&\qquad+\lambda^{\frac{1}{2}}\|\nabla\times\mathbf{b}_h\|+\frac{1}{\lambda^{\frac{1}{2}}}\|\nabla r_h\|\\
&\quad\lesssim \left(1+\frac{h}{\nu^{\frac{1}{2}}\tau_1^{\frac{1}{2}}}+\frac{\|\mathbf{d}\|_{L^{q(\delta,d)}(\Omega)}}{\sqrt{\nu\lambda}}\right)|||(\mathbf{u}_h,p_h,\mathbf{b}_h,r_h)|||_{\mathrm{stab,w}},
\end{aligned}
$$

where we have used the interpolation error estimate $\|\varepsilon_h(v)\|\lesssim h\|v\|_1$, integration-by-parts, Schwarz's inequality and the previous continuity result. The second term in the right-hand side of (3.9) is easily handled by using the $H^1(\Omega)$ stability of the projector $\pi_h(\cdot)$. Noting that for any $\epsilon\in(0,3)$ there exists $\delta>0$ such that $d+\epsilon>q(\delta,d)$, we prove the result. Let us stress the fact that ϵ can be taken arbitrarily small. □

Combining the previous lemmas, we readily get a weak inf-sup condition which provides control in the strong stabilized form.

Corollary 3.1. *Under the assumption that* $\mathbf{a}\in H^1(\Omega)^d$ *and* $\mathbf{d}\in L^{d+\epsilon}(\Omega)^d$ *for an arbitrarily small* $\epsilon>0$*, the following inequality holds for any* $(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h)\in\boldsymbol{V}_h\times\boldsymbol{C}_h\times Q_h\times S_h$*,*

$$
\begin{aligned}
\|(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h)\|_{\mathrm{stab,s}}\lesssim&\sup_{(\mathbf{v}_h,s_h)\in\boldsymbol{V}_h\times S_h}\frac{\mathcal{A}((\mathbf{u}_h,\mathbf{b}_h,p_h,r_h),(\mathbf{v}_h,\mathbf{0},0,s_h))}{\|(\mathbf{v}_h,\mathbf{0},0,s_h)\|_{\mathrm{Gal}}}\\
&+\sigma\,\mathcal{A}_{\mathrm{stab}}((\mathbf{u}_h,\mathbf{b}_h,p_h,r_h),(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h))^{\frac{1}{2}}.
\end{aligned}
\tag{3.10}
$$

Now, we are in position to provide bounds of the FE solution of the MHD problem in Algorithm 3.1 with respect to the data.

Theorem 3.2. *The solution* $(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h)$ *of the FE problem (3.7) satisfies:*

(i) Weak stability: For $\mathbf{a}\in H^1(\Omega)^d$ *and* $\mathbf{d}\in H(\mathbf{curl};\Omega)$*, it holds*

$$
\|(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h)\|_{\mathrm{stab,w}}\leq\sup_{(\mathbf{v}_h,\mathbf{c}_h,q_h,s_h)\in\boldsymbol{V}_h\times\boldsymbol{C}_h\times Q_h\times S_h}\frac{\mathcal{F}_{\mathrm{stab}}(\mathbf{v}_h,\mathbf{c}_h,q_h,s_h)}{\|\mathbf{v}_h,\mathbf{c}_h,q_h,s_h\|_{\mathrm{stab,w}}}
\tag{3.11}
$$

(ii) Strong stability: For $\mathbf{a}\in L^\infty(\Omega)^d$ *and* $\mathbf{d}\in W^{1,d+\epsilon}(\Omega)^d\cap L^\infty(\Omega)^d$ *for an arbitrarily small* $\epsilon>0$*, it holds*

$$
\begin{aligned}
\|(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h)\|_{\mathrm{stab,s}}\leq&\xi\|(\mathbf{u}_h,\mathbf{b}_h,p_h,r_h)\|_{\mathrm{stab,w}}\\
&+\sup_{(\mathbf{v}_h,s_h)\in\boldsymbol{V}_h\times S_h}\frac{|\mathcal{F}_{\mathrm{stab}}(\mathbf{v}_h,\mathbf{0},0,s_h)|}{|||(\mathbf{v}_h,\mathbf{0},0,s_h)|||_{\mathrm{stab,s}}},
\end{aligned}
\tag{3.12}
$$

with the constant

$$\xi = 1 + \frac{\tau_1^{\frac{1}{2}}}{\nu^{\frac{1}{2}}}\|\mathbf{a}\|_{L^\infty(\Omega)} + \frac{\tau_3^{\frac{1}{2}}}{\nu^{\frac{1}{2}}}(\|\nabla\mathbf{d}\|_{L^{d+\epsilon}(\Omega)} + \|\mathbf{d}\|_{L^\infty(\Omega)}) + \frac{\|\mathbf{d}\|_{L^{d+\epsilon}(\Omega)}}{\sqrt{\nu\lambda}} + \left(1 + \frac{\|\mathbf{a}\|_{L^\infty(\Omega)}h}{\nu}\right)^{\frac{1}{2}} \left(1 + \frac{\|\mathbf{d}\|_{L^\infty(\Omega)}h}{\sqrt{\lambda\nu}\sqrt{1 + \frac{\|\mathbf{a}\|_{L^\infty(\Omega)}h}{\nu}}}\right)^{\frac{1}{2}}.$$

Proof: We readily prove the weak stability invoking Lemma 3.1 and the stabilized FE system (3.7). Strong stability is proved using Corollary 3.1. The second term in the right-hand side of (3.10) is readily handled by the weak stability (3.11). On the other hand, the first term is bounded as follows:

$$\begin{aligned}
&\mathcal{A}((\mathbf{u}_h, \mathbf{b}_h, p_h, r_h), (\pi_h(\mathbf{v}), \mathbf{0}, 0, \pi_h(s))) = \\
&- (\tau_1(X_{\mathrm{u}}(\mathbf{u}_h, p_h, \mathbf{b}_h) - \nu\Delta\mathbf{u}_h), \mathbf{a}\cdot\nabla\pi_h(\mathbf{v}) + \nu\Delta\pi_h(\mathbf{v}))_h \\
&- (\tau_2\nabla\cdot\mathbf{u}_h, \nabla\cdot\pi_h(\mathbf{v})) - (\tau_3(\nabla\times(\mathbf{u}_h\times\mathbf{d}) - \lambda\nabla\times(\nabla\times\mathbf{b}_h)), \nabla\times(\pi_h(\mathbf{v})\times\mathbf{d}))_h \\
&- (\tau_4\nabla r_h, \nabla\pi_h(s)) + \mathcal{F}_{\mathrm{stab}}(\pi_h(\mathbf{v}), \mathbf{0}, 0, \pi_h(s)) \\
&\leq \left(|||(\mathbf{u}_h, \mathbf{b}_h, p_h, r_h)|||_{\mathrm{stab,w}} + \sup_{(\mathbf{v}_h, s_h)\in \boldsymbol{V}_h\times S_h} \frac{|\mathcal{F}_{\mathrm{stab}}(\mathbf{v}_h, \mathbf{0}, 0, s_h)|}{|||(\mathbf{v}_h, \mathbf{0}, 0, s_h)|||_{\mathrm{stab,s}}}\right) \\
&\quad\times (\tau_1^{\frac{1}{2}}\|\mathbf{a}\cdot\nabla\pi_h(\mathbf{v})\| + \tau_1^{\frac{1}{2}}\|\nu\Delta\pi_h(\mathbf{v})\|_h + \tau_2^{\frac{1}{2}}\|\nabla\cdot\pi_h(\mathbf{v})\| + \tau_3^{\frac{1}{2}}\|\nabla\times(\pi_h(\mathbf{v})\times\mathbf{d})\| \\
&\qquad + \tau_4^{\frac{1}{2}}\|\nabla\pi_h(s)\| + \nu^{\frac{1}{2}}\|\pi_h(\mathbf{v})\|_1\|).
\end{aligned} \tag{3.13}$$

Let us work on the right-hand side of the previous inequality. Using the stability properties of the projector $\pi_h(\cdot)$, we obtain:

$$\tau_1^{\frac{1}{2}}\|\mathbf{a}\cdot\nabla\pi_h(\mathbf{v})\| \leq \tau_1^{\frac{1}{2}}\|\mathbf{a}\|_{L^\infty(\Omega)}\|\nabla\mathbf{v}\|.$$

The same arguments allow us to get the following bound:

$$\begin{aligned}
\tau_3^{\frac{1}{2}}\|\nabla\times(\pi_h(\mathbf{v})\times\mathbf{d})\| &\lesssim \tau_3^{\frac{1}{2}}(\|\mathbf{v}\|_{L^{q(\delta,d)'}(\Omega)}\|\nabla\mathbf{d}\|_{L^{q(\delta,d)}(\Omega)} + \|\nabla\mathbf{v}\|\|\mathbf{d}\|_{L^\infty(\Omega)}) \\
&\lesssim \tau_3^{\frac{1}{2}}\|\mathbf{v}\|_1(\|\nabla\mathbf{d}\|_{L^{q(\delta,d)}(\Omega)} + \|\mathbf{d}\|_{L^\infty(\Omega)}),
\end{aligned}$$

where we have used the vector analysis formula

$$\nabla\times(\mathbf{a}\times\mathbf{b}) = \mathbf{b}\cdot\nabla\mathbf{a} - \mathbf{b}(\nabla\cdot\mathbf{a}) - \mathbf{a}\cdot\nabla\mathbf{b} + \mathbf{a}(\nabla\cdot\mathbf{b}),$$

that holds for any smooth vector fields $\mathbf{a}$ and $\mathbf{b}$, and similar arguments as those above. Invoking these bounds in (3.13), we easily get:

$$\begin{aligned}
\mathcal{A}((\mathbf{u}_h, \mathbf{b}_h, p_h, r_h), (\pi_h(\mathbf{v}), \mathbf{0}, 0, \pi_h(s))) &\lesssim \|(\mathbf{v}_h, \mathbf{0}, 0, s_h)\|_{\mathrm{Gal}} \times |||(\mathbf{u}_h, \mathbf{b}_h, p_h, r_h)|||_{\mathrm{stab,w}} \\
&\quad\times \left(\frac{\tau_1^{\frac{1}{2}}}{\nu^{\frac{1}{2}}}\|\mathbf{a}\|_{L^\infty(\Omega)} + \frac{\tau_3^{\frac{1}{2}}}{\nu^{\frac{1}{2}}}(\|\nabla\mathbf{d}\|_{L^{q(\delta,d)}(\Omega)} + \|\mathbf{d}\|_{L^\infty(\Omega)})\right).
\end{aligned}$$

Now, let us rewrite some coefficients by using the definition of the stabilization parameters:

$$\frac{h}{\nu^{\frac{1}{2}}\tau_1^{\frac{1}{2}}} = \left(1+\frac{\|\mathbf{a}\|_{L^\infty(\Omega)}h}{\nu}\right)^{\frac{1}{2}}\left(1+\frac{\|\mathbf{d}\|_{L^\infty(\Omega)}h}{\sqrt{\nu\lambda}\sqrt{1+\frac{\|\mathbf{a}\|_{L^\infty(\Omega)}h}{\nu}}}\right)^{\frac{1}{2}}.$$

Using again that for small enough $\epsilon>0$ there exists $\delta>0$ such that $d+\epsilon>q(\delta,d)$, we prove the theorem. □

Remark 3.1. *In practice, both* $\mathbf{a}$ *and* $\mathbf{d}$ *will be FE functions.*[1] *In this situation, we can clearly reduce the assumptions over these two functions in Theorem 3.2.*

The following corollary consists of the restriction of the previous analysis to FE functions.

Corollary 3.2. *Let us assume that* $\mathbf{a}_h\in\boldsymbol{V}_h$ *and* $\mathbf{d}_h\in\boldsymbol{C}_h\cap L^{d+\epsilon}(\Omega)^d$ *for some* $\epsilon>0$. *The FE solution of (3.7) satisfies inequality (3.12) for*

$$\xi = 1+\frac{h^{\frac{1}{4}}\|\mathbf{a}_h\|_1^{\frac{1}{2}}}{\nu^{\frac{1}{2}}}+\frac{\|\mathbf{d}_h\|_{L^{d+\epsilon}(\Omega)}}{\sqrt{\nu\lambda}}+\left(1+\frac{\|\mathbf{a}_h\|_1 h^{\frac{1}{2}}}{\nu}\right)^{\frac{1}{2}}\left(1+\frac{\|\mathbf{d}_h\|_{L^{d+\epsilon}(\Omega)}}{\sqrt{\nu\lambda}}\right)^{\frac{1}{2}}. \tag{3.14}$$

Proof: Let us introduce the inequality

$$\|v_h\|_{W_p^l(\Omega)}\lesssim h^{m-l+\frac{d}{p}-\frac{d}{q}}\|v_h\|_{W_q^m(\Omega)} \tag{3.15}$$

that holds for FE functions v_h, where d is the space dimension (see [33, Th. 4.5.11]). In dimension three, we have that:

$$\tau_1^{\frac{1}{2}}\|\mathbf{a}_h\|_{L^\infty(\Omega)}\lesssim\min(h^{\frac{1}{2}}\|\mathbf{a}_h\|_{L^\infty(\Omega)}^{-\frac{1}{2}},h\nu^{-\frac{1}{2}})\|\mathbf{a}_h\|_{L^\infty(\Omega)}\lesssim\min\left(h^{\frac{1}{4}}\|\mathbf{a}_h\|_1^{\frac{1}{2}},h^{\frac{1}{2}}\nu^{-\frac{1}{2}}\|\mathbf{a}_h\|_1\right).$$

Analogously, we obtain:

$$\tau_3^{\frac{1}{2}}(\|\nabla\mathbf{d}_h\|_{L^{d+\epsilon}(\Omega)}+\|\mathbf{d}_h\|_{L^\infty(\Omega)})\lesssim\tau_3^{\frac{1}{2}}h^{-1}\|\mathbf{d}_h\|_{L^{d+\epsilon}(\Omega)}\lesssim\lambda^{-\frac{1}{2}}\|\mathbf{d}_h\|_{L^{d+\epsilon}(\Omega)},$$

where we have used the inequality (3.15). With regard to the last term in the definition of ξ, we have:

$$\frac{h}{\nu^{\frac{1}{2}}\tau_1^{\frac{1}{2}}}\lesssim\left(1+\frac{\|\mathbf{a}_h\|_1 h^{\frac{1}{2}}}{\nu}\right)^{\frac{1}{2}}\left(1+\frac{\|\mathbf{d}_h\|_{L^{d+\epsilon}(\Omega)}}{\sqrt{\nu\lambda}}\right)^{\frac{1}{2}}.$$

Therefore, using the fact that $1+\min(a,a^2)\lesssim 1+a$ for any $a>0$, the previous corollary applies for the definition of ξ in (3.14). □

Remark 3.2. *In the previous results, we observe that stability bounds for some terms of the Galerkin norm, as well as the extra stability that comes from the stabilization terms do not deteriorate in asymptotic regimes. However, control over* $\|p\|$ *and* $\|\mathbf{b}\|$ *can deteriorate for fixed grids in some singular limits of the continuous problem that imply a coercivity loss. Anyway, the same behavior has been observed for the Navier-Stokes problem alone when solved by using stabilized FE techniques.*

[1] In any case, we can always project the continuous fields into the FE spaces using proper projections.

3.5 Convergence analysis

Once we have proved stability of the problem, we look at the convergence properties of the numerical algorithm. We are interested in both convergence towards the exact solution (even when it is rough) and optimal order of convergence, i.e. *a priori* error estimates when the solution is smoother.

Using the fact that the stabilized FE problem (3.7) is consistent, i.e. the exact solution satisfies the FE equality, we have that:

$$\begin{aligned}\mathcal{A}_{\text{stab}}((\chi(\mathbf{u}_h,\mathbf{u}),\chi(\mathbf{b}_h,\mathbf{b}),\chi(p_h,p),\chi(r_h,r)),(\mathbf{v}_h,\mathbf{c}_h,q_h,s_h))\\ =\mathcal{A}_{\text{stab}}((\varepsilon_h(\mathbf{u}),\varepsilon_h(\mathbf{b}),\varepsilon_h(p),\varepsilon_h(r)),(\mathbf{v}_h,\mathbf{c}_h,q_h,s_h)),\end{aligned}\tag{3.16}$$

where $\chi(v,w):=v-\pi_h(w)$; as stated above, $\varepsilon_h(v):=v-\pi_h(v)$ and $\pi_h(v)$ is a continuous FE interpolant in $H^1(\Omega)$ with optimal interpolation properties that preserves null traces, e.g. the Scott-Zhang interpolant. Let us also define the following norms:

$$\begin{aligned}\|(\mathbf{v},\mathbf{c},q,s)\|_{\widetilde{\text{stab,w}}}&=\nu^{\frac{1}{2}}\|\mathbf{v}\|_1+\lambda^{\frac{1}{2}}\|\nabla\times\mathbf{c}\|+\|\tau_1^{\frac{1}{2}}X_{\text{u}}(\mathbf{v},q,\mathbf{c})\|_h+\|\tau_2^{\frac{1}{2}}\nabla\cdot\mathbf{v}\|\\&\quad+\|\tau_3^{\frac{1}{2}}\nabla\times(\mathbf{v}\times\mathbf{d})\|+\|\tau_4^{\frac{1}{2}}\nabla s\|,\\ \|(\mathbf{v},\mathbf{c},q,s)\|_{\widetilde{\text{stab,s}}}&=\frac{\lambda^{\frac{1}{2}}}{L_0}\|\mathbf{c}\|+\frac{1}{\nu^{\frac{1}{2}}}\|q\|+\|(\mathbf{v},\mathbf{c},q,s)\|_{\widetilde{\text{stab,w}}}.\end{aligned}$$

So, the norms with the tilde have all the terms of those without the tilde except the stabilization term related to τ_5. Let us also define the error functions:

$$\begin{aligned}E_{\mathcal{A}}(h):=&\,\tau_1^{-\frac{1}{2}}\|\varepsilon_h(\mathbf{u})\|+\nu^{\frac{1}{2}}\|\nabla\varepsilon_h(\mathbf{u})\|+\lambda^{\frac{1}{2}}\|\varepsilon_h(\mathbf{b})\|_{\mathbf{curl}}\\&+\min\left\{\nu^{-\frac{1}{2}}L_0^{\frac{1}{2}}\|\mathbf{d}\|_{L^{d+\epsilon}(\Omega)}\|\nabla\times\varepsilon_h(\mathbf{b})\|,\tau_3^{-\frac{1}{2}}\|\varepsilon_h(\mathbf{b})\|\right\}+\tau_2^{-\frac{1}{2}}\|\varepsilon_h(p)\|,\\ E_{\mathcal{S}}(h):=&\,\tau_1^{\frac{1}{2}}\|X_{\text{u}}(\varepsilon_h(\mathbf{u}),\varepsilon_h(p),\varepsilon_h(\mathbf{b}))-\nu\Delta\varepsilon_h(\mathbf{u})\|+\tau_2^{\frac{1}{2}}\|\nabla\cdot\varepsilon_h(\mathbf{u})\|\\&+\tau_3^{\frac{1}{2}}\|\nabla\times(\varepsilon_h(\mathbf{u})\times\mathbf{d})+\lambda\nabla\times(\nabla\times\varepsilon_h(\mathbf{b})))\|+\lambda^{\frac{1}{2}}L_0^{-1}\|\varepsilon_h(\mathbf{b})\|\\&+\lambda^{\frac{1}{2}}L_0^{-1}\left(\sum_{K\in\mathcal{T}_h}h\|\varepsilon_h(\mathbf{b})\|^2_{L^2(\partial K)}\right)^{\frac{1}{2}},\end{aligned}$$

for some $\epsilon>0$. In order for the error function $E_{\mathcal{S}}(h)$ to be well-defined, we require $-\nu\Delta\mathbf{u}+X_{\text{u}}(\mathbf{u},\mathbf{b},p)\in L^2(\Omega)^d$ and $\lambda\nabla\times(\nabla\times\mathbf{b})-\nabla\times(\mathbf{u}\times\mathbf{d})\in L^2(\Omega)^d$, which are true for $\mathbf{f}_{\text{u}}\in L^2(\Omega)^d$ and $\mathbf{f}_{\text{b}}\in L^2(\Omega)^d$, respectively; this is easily inferred from the continuous problem. Furthermore, the boundary terms are well defined since $\mathbf{b}_h\in L^2(\partial K)$ for every $K\in\mathcal{T}_h$ (see [13, Corollary 3.8]).

Theorem 3.3. *Under the conditions of Corollary 3.1, the following* a priori *error estimate holds*

$$|||(\mathbf{u}-\mathbf{u}_h,\mathbf{b}-\mathbf{b}_h,p-p_h,r-r_h)|||_{\widetilde{\text{stab,w}}}\lesssim E_{\mathcal{A}}(h)+E_{\mathcal{S}}(h),$$

for the error functions defined above.

Proof: Let us bound the right hand side of the error system (3.16), first Galerkin terms and second stabilization ones. Proceeding as in the proof of Lemma 3.1, we obtain

$$\langle \varepsilon_h(\mathbf{b}), \nabla \times (\mathbf{v} \times \mathbf{d}) \rangle \lesssim L_0^{\frac{1}{2}} \|\nabla \times \varepsilon_h(\mathbf{b}_h)\| \|\mathbf{v}\|_1 \|\mathbf{d}\|_{L^{d+\epsilon}(\Omega)}.$$

This result, together with a straight use of integration-by-parts, Schwarz's inequality, the fact that $\varepsilon_h(r) = 0$ and the definition of τ_4, lead to:

$$\begin{aligned}
&\mathcal{A}((\varepsilon_h(\mathbf{u}), \varepsilon_h(\mathbf{b}), \varepsilon_h(p), \varepsilon_h(r)), (\mathbf{v}_h, \mathbf{c}_h, q_h, s_h)) \\
&\quad = -\langle \varepsilon_h(\mathbf{u}), X_{\mathrm{u}}(\mathbf{v}_h, q_h, \mathbf{c}_h) \rangle + \nu(\nabla \varepsilon_h(\mathbf{u}), \nabla \mathbf{v}_h) - (\varepsilon_h(p), \nabla \cdot \mathbf{v}_h) \\
&\qquad + \langle \varepsilon_h(\mathbf{b}), \nabla \times (\mathbf{v}_h \times \mathbf{d}) \rangle + \lambda(\nabla \times \varepsilon_h(\mathbf{b}), \nabla \times \mathbf{c}_h) - (\varepsilon_h(\mathbf{b}), \nabla s_h) \\
&\quad \lesssim E_{\mathcal{A}}(h) |\!|\!| (\mathbf{v}_h, \mathbf{c}_h, q_h, s_h) |\!|\!|_{\mathrm{stab,w}},
\end{aligned} \tag{3.17}$$

In order to bound the stabilization terms, we proceed as follows:

$$\begin{aligned}
\tau_5(\nabla \cdot \varepsilon_h(\mathbf{b}), \nabla \cdot \mathbf{c}_h) &= \tau_5 \sum_{K \in \mathcal{T}_h} \int_K \varepsilon_h(\mathbf{b}) \nabla \nabla \cdot \mathbf{c}_h - \tau_5 \sum_{K \in \mathcal{T}_h} \int_{\partial K} \mathbf{n} \cdot \varepsilon_h(\mathbf{b}) \nabla \cdot \mathbf{c}_h \\
&\lesssim (h^{-1} \tau_5^{\frac{1}{2}} \|\varepsilon_h(\mathbf{b})\| + h^{-\frac{1}{2}} \tau_5^{\frac{1}{2}} \|\varepsilon_h(\mathbf{b})\|_{L^2(\partial K)}) \tau_5^{\frac{1}{2}} \|\nabla \cdot \mathbf{c}_h\|.
\end{aligned}$$

where we have used the inequality $\|v\|_{L^2(\partial K)} \leq h^{-\frac{1}{2}} \|v\|_{L^2(K)}$ that holds for FE functions (see [33]). Using the fact that $h^{-1} \tau_5^{\frac{1}{2}} = c_5 L_0^{-1} \lambda^{\frac{1}{2}}$, and Schwarz's inequality we easily obtain:

$$\mathcal{S}(\varepsilon_h(\mathbf{u}), \varepsilon_h(\mathbf{b}), \varepsilon_h(p), \varepsilon_h(r)), (\mathbf{v}, \mathbf{c}, q, s)) \leq E_{\mathcal{S}}(h) |\!|\!| (\mathbf{v}, \mathbf{c}, q, s) |\!|\!|_{\mathrm{stab,w}}. \tag{3.18}$$

So, using the weak coercivity in Lemma 3.1 and the previous results, we straightforwardly get:

$$\|(\chi(\mathbf{u}_h, \mathbf{u}), \chi(\mathbf{b}_h, \mathbf{b}), \chi(p_h, p), \chi(r_h, r))\|_{\mathrm{stab,w}} \lesssim E_{\mathcal{A}}(h) + E_{\mathcal{S}}(h). \tag{3.19}$$

Finally, using the triangle inequality and the fact that

$$|\!|\!| (\varepsilon_h(\mathbf{u}), \varepsilon_h(\mathbf{c}), \varepsilon_h(p), \varepsilon_h(s)) |\!|\!|_{\widetilde{\mathrm{stab,w}}} \lesssim E_{\mathcal{A}}(h) + E_{\mathcal{S}}(h)$$

we prove the theorem. □

On the other hand, we can recover convergence on the strong stabilized norm as follows:

Theorem 3.4. *Under the conditions of Corollary 3.2, the following* a priori *error estimate holds*

$$|\!|\!| (\mathbf{u} - \mathbf{u}_h, \mathbf{b} - \mathbf{b}_h, p - p_h, r - r_h) |\!|\!|_{\widetilde{\mathrm{stab,s}}} \lesssim \xi(E_{\mathcal{A}}(h) + E_{\mathcal{S}}(h)),$$

for the error functions defined above.

Proof: First, invoking Theorem 3.2, we have

$$\begin{aligned}
&\|(\chi(\mathbf{u}_h,\mathbf{u}),\chi(\mathbf{b}_h,\mathbf{b}),\chi(p_h,p),\chi(r_h,r))\|_{\text{stab,s}}\\
&\lesssim \xi\|(\chi(\mathbf{u}_h,\mathbf{u}),\chi(\mathbf{b}_h,\mathbf{b}),\chi(p_h,p),\chi(r_h,r))\|_{\text{stab,w}}\\
&\quad+\sup_{(\mathbf{v}_h,s_h)\in\boldsymbol{V}_h\times S_h}\frac{\mathcal{A}_{\text{stab}}((\varepsilon_h(\mathbf{u}),\varepsilon_h(\mathbf{b}),\varepsilon_h(p),\varepsilon_h(r)),(\mathbf{v}_h,\mathbf{0},0,s_h))}{|||(\mathbf{v}_h,\mathbf{0},0,s_h)|||_{\text{stab,s}}}.
\end{aligned}$$

We can readily bound the right hand side using the bounds in (3.17)-(3.18) and (3.19), in order to get:

$$\|(\chi(\mathbf{u}_h,\mathbf{u}),\chi(\mathbf{b}_h,\mathbf{b}),\chi(p_h,p),\chi(r_h,r))\|_{\text{stab,s}}\lesssim\xi(E_{\mathcal{A}}(h)+E_{\mathcal{S}}(h)).$$

Using the relation $\tau_2^{-\frac{1}{2}}\lesssim\nu^{-\frac{1}{2}}$ we easily get

$$|||(\varepsilon_h(\mathbf{u}),\varepsilon_h(\mathbf{c}),\varepsilon_h(p),\varepsilon_h(s))|||_{\widetilde{\text{stab,s}}}\lesssim E_{\mathcal{A}}(h)+E_{\mathcal{S}}(h).$$

Combining the last two inequalities and the triangle inequality, we prove the theorem. □

In the asymptotic limit when $h\searrow 0$, we can easily see that

$$\begin{aligned}
E_{\mathcal{A}}(h)+E_{\mathcal{S}}(h)\ \sim\ &h^{-1}\|\varepsilon_h(\mathbf{u})\|+\|\varepsilon_h(\mathbf{u})\|_1+h\|X_{\text{u}}(\varepsilon_h(\mathbf{u}),\varepsilon_h(\mathbf{b}),\varepsilon_h(p))-\nu\Delta\varepsilon_h(\mathbf{u})\|\\
&+\|\varepsilon_h(p)\|+\|\varepsilon_h(\mathbf{b})\|_{\mathbf{curl}}+\left(\sum_{K\in\mathcal{T}_h}h\|\varepsilon_h(\mathbf{b})\|^2_{L^2(\partial K)}\right)^{\frac{1}{2}}\\
&+h\|\nabla\times(\varepsilon_h(\mathbf{u})\times\mathbf{d})+\lambda\nabla\times(\nabla\times\varepsilon_h(\mathbf{b}))\|.
\end{aligned}$$

So, the convergence results are optimal, in the sense that optimal rates are obtained for smooth enough functions. In order to get convergence to singular components of the magnetic field, the following approximability condition must hold:

$$\lim_{h\to0}\left\{\|\varepsilon_h(\mathbf{b})\|_{\mathbf{curl}}+\left(\sum_{K\in\mathcal{T}_h}h\|\varepsilon_h(\mathbf{b})\|^2_{L^2(\partial K)}\right)^{\frac{1}{2}}\right\}=0. \tag{3.20}$$

This condition is true for Lagrangian FEs for meshes with a particular macro-element structure (see [13, Assumption 1 and Corollary 5]). One type of element that satisfies this condition is the Powell-Sabin macro-element (see [104,114] and [36, Remark 4.1]); we note that in two space dimensions this macroelement requires its circumcenter to remain inside the FE. Further, numerical experiments in 2D and 3D show that both the Powell-Sabin and *criss-cross* elements provide excellent results (see Chapter 2). In the numerical experiments section, we have extended these results, evaluating the effect of not using this type of meshes in the convergence towards singular solutions. Finally, let us stress the fact that this macro-element structure is only needed for singular solutions, that appear in non-convex domains. For smooth solutions, with $\mathbf{b}\in H^1(\Omega)^d$, the convergence analysis and approximability properties are easy to check and hold for any type of mesh. We refer to [13] for a detailed discussion on this topic, in the framework of the Maxwell operator.

Remark 3.3. *It is not the target of the method developed in this chapter the eigenvalue MHD problem [29]. Since the requirements for a method to be useful for initial and boundary value problems, viz. stability and convergence estimates, are different to the ones for eigenvalue problems (see [26]), it is unclear whether this approach would serve in this last case. However, due to the large amount of problems of interest that are governed by the initial and boundary value MHD problem, efficient methods for this problem, as the one analyzed herein, are useful for the MHD community. We refer to [27] for the application of a similar algorithm to the eigenvalue Maxwell problem.*

Remark 3.4. *The previous version of the stabilized MHD problem in [53, 54] (not converging to singular solutions) has extensively been used for large-scale problems by Shadid and co-workers (see e.g. [60, 112]); they have observed that stabilized MHD systems are much easier to solve than inf-sup stable formulations, especially for the parallel algebraic multigrid method in TRILINOS (see [79, 80]), and favor their use for large scale MHD problems.*

Remark 3.5. *This type of stabilized FE formulations have also been used succesfully for transient problems in Chapter 2. It is due to the fact that the divergence-free constraint is explicitly enforced at every time step via the introduction of the pseudo-pressure r.*

Remark 3.6. *Let us note that a method that introduces similar stabilization terms has been proposed in [27] for electromagnetic eigenvalue problems. The method in [27] depends on a coefficient α and corresponds to the method proposed in [13] for $\alpha = 1$ with the only difference that no restriction over the FE spaces or meshes is assumed. Unfortunately, the convergence analysis in [27] does not apply for $\alpha = 1$, since the analysis says that polynomials of infinite order would be required in this case. See [12] and the numerical experiments section for further details.*

3.6 Some comments on the nonlinear analysis

The numerical analysis of FE methods for the incompressible nonlinear MHD equations is a hard issue; let us recall that this system is obtained taking $(\mathbf{a}, \mathbf{d})$ as $(\mathbf{u}, \varrho\mathbf{b})$ in (3.1). Using inf-sup stable elements, this analysis has been carried out in [74, 111]. It heavily uses the nice properties of Nédélec type FEs and cannot be applied to nodal FEs. For weighted regularization techniques, the full nonlinear analysis has been published in [77]. As stated by the authors, the results in three dimensions are quite restrictive. Finally, a closer formulation to the one presented herein, based on discontinuous Galerkin nodal FEs, is presented in [81]. However, both the numerical analysis and experiments (dealing with singular solutions) are for the linearized problem only. The authors say that the nonlinear extension is an open problem, and up to our knowledge it keeps open so far.

Therefore, the extension of the linearized analysis above to the nonlinear problem is not straightforward. For the nonlinear problem, $\mathbf{a}_h$ and $\mathbf{d}_h$ are in fact the solution of the previous iterate, when using a Picard-type linearization. Thus, we cannot assume regularity over $\mathbf{d}_h$, namely $\mathbf{d}_h \in L^{d+\epsilon}(\Omega)^d$ for some $\epsilon > 0$.

We may proceed by induction. Let us denote the iteration counter with a superscript within parentheses. In the nonlinear version of system (3.1) for $i \geq 1$ we replace $\mathbf{a} \leftarrow \mathbf{u}_h^{(i-1)}$, $\mathbf{d} \leftarrow \varrho\mathbf{b}_h^{(i-1)}$, and denote the continuous solution of this problem as

$(\mathbf{u}^{(i-1)}, \mathbf{b}^{(i-1)}, p^{(i-1)}, r^{(i-1)})$. Next, we obtain the new iterate $(\mathbf{u}_h^{(i)}, \mathbf{b}_h^{(i)}, p_h^{(i)}, r_h^{(i)})$ by solving the discretized problem (3.7). We assume in what follows that the physical properties are such that this iterative scheme converges. Let us also assume that:

$$\mathbf{b}_h^{(i-1)} \text{ is such that } \|\mathbf{b}^{(i-1)} - \mathbf{b}_h^{(i-1)}\| \lesssim h^{\frac{d-2}{2}+\delta^{(i-1)}} \text{ for some } \delta^{(i-1)} > 0 \tag{A1}$$

Let us introduce the inverse inequality $\|v_h\|_{L^q(\Omega)} \lesssim h^{\frac{d}{q}-\frac{d}{p}}\|v_h\|_{L^p(\Omega)}$, where $1 \leq p \leq \infty$ and $1 \leq q \leq \infty$ (see [33, Theorem 4.5.11]). We infer that $\|v_h\|_{L^{d+\epsilon}(\Omega)} \lesssim h^{-\theta_\epsilon(d)}\|v_h\|$, with $\theta_\epsilon(2) = \frac{\epsilon}{2+\epsilon}$ and $\theta_\epsilon(3) = \frac{3(1+\epsilon)}{2(3+\epsilon)}$ for any $\epsilon > 0$. Using the stability of the Scott-Zhang projector $\pi_h(\cdot)$, we finally get:

$$\begin{aligned}\|\mathbf{b}_h^{(i-1)}\|_{L^{d+\epsilon}(\Omega)} &\lesssim \|\pi_h(\mathbf{b}^{(i-1)}) - \mathbf{b}_h^{(i-1)}\|_{L^{d+\epsilon}(\Omega)} + \|\pi_h(\mathbf{b}^{(i-1)})\|_{L^{d+\epsilon}(\Omega)} \\ &\lesssim h^{\frac{d-2}{2}+\delta^{(i-1)}-\theta_\epsilon(d)} + h^{-\theta_\epsilon(d)}\|\mathbf{b}^{(i-1)} - \pi_h(\mathbf{b}^{(i-1)})\| + \|\mathbf{b}^{(i-1)}\|_{L^{d+\epsilon}(\Omega)}.\end{aligned}$$

Next, we use the continuous imbedding of $\boldsymbol{C} \cap H(\mathrm{div};\Omega)$ into $H^{\frac{1}{2}+\delta_1}(\Omega)^d$ for some $\delta_1 > 0$ and the interpolation properties of the Scott-Zhang projector (see [33, Theorem 4.8.12]) to infer that $\|\mathbf{b}^{(i-1)} - \pi_h(\mathbf{b}^{(i-1)})\| \lesssim h^{\frac{1}{2}+\delta_1}\|\mathbf{b}^{(i-1)}\|_{\mathbf{curl}}$ On the other hand, since $\boldsymbol{C} \cap H(\mathrm{div};\Omega)$ is compactly imbedded into $L^{3+\delta_2}(\Omega)$, for some $\delta_2 > 0$ (see [111, Proposition 2.3] and [2, Proposition 3.7]), we can pick an $\epsilon = \epsilon(\delta^{(i-1)}, \delta_1, \delta_2) > 0$ small enough such that $\|\mathbf{b}_h^{(i-1)}\|_{L^{d+\epsilon}(\Omega)} \lesssim \|\mathbf{b}^{(i-1)}\|_{\mathbf{curl}}$.

From Theorem 3.4, we have that $\|\mathbf{b}^{(i)} - \mathbf{b}_h^{(i)}\| \lesssim \xi(E_{\mathcal{A}}(h) + E_{\mathcal{S}}(h))$, where ξ is now bounded. Assuming that:

$$E_{\mathcal{A}}(h) + E_{\mathcal{S}}(h) \lesssim h^{\frac{d-2}{2}+\delta^{(i)}} \text{ for some } \delta^{(i)} > 0, \tag{A2}$$

we finally obtain that $\|\mathbf{b}^{(i)} - \mathbf{b}_h^{(i)}\| \lesssim h^{\frac{1}{2}+\delta^{(i)}}$.

Assumption A1 is easily satisfied. In fact, we only need to start the process with a $\mathbf{b}_h^{(0)} \in \boldsymbol{C} \cap H(\mathrm{div};\Omega)$, e.g. $\mathbf{b}_h^{(0)} = \mathbf{0}$ and $\mathbf{b}^{(0)} = \mathbf{0}$, so that the effective initial guess is a solution of a linear problem for which we know that A1 holds [13].

Assumption A2 is true for $(\mathbf{u}^{(i-1)}, p^{(i-1)}) \in H^{\frac{d}{2}+\epsilon}(\Omega)^d \times H^{\frac{d-2}{2}+\epsilon}(\Omega)$; the terms related to $\mathbf{b}$ can be treated as in [13, Corollary 3.12]. In dimension three, it requires $(\mathbf{u}^{(i-1)}, p^{(i-1)}) \in H^{\frac{3}{2}+\epsilon}(\Omega)^d \times H^{\frac{1}{2}+\epsilon}(\Omega)$ for an arbitrary small $\epsilon > 0$.

3.7 Numerical experimentation

The objective of the following numerical experiment is to compare the approximation of singular solutions for the MHD problem using several mesh structures. We will show the difference between a mesh with a suitable macro-element structure, the crossbox element, which has been observed numerically that verifies the approximability condition (3.20), against mesh structures that do not satisfy it, both in terms of convergence rates and the solution itself.

We have chosen to solve the same test problem in Chapter 2 Section 2.6.1 but extending the previous results to finer meshes with up to 12.58M elements. This problem with non-smooth solution corresponds to solve the MHD equations in a nonconvex L-shaped domain $\Omega = (-1,1) \times (-1,1) \setminus [0,1] \times [0,-1]$. Both the hydrodynamic and magnetic solutions have strong singularities at the re-entrant corner, where the origin of coordinates

is taken. The singular solution for the Stokes operator is described in polar coordinates (r, θ) by

$$\begin{aligned} u_x(x,y) =& r^\lambda \left((1+\lambda)\sin(\theta)\psi(\theta) + \cos(\theta)\psi'(\theta) \right), \\ u_y(x,y) =& r^\lambda \left(-(1+\lambda)\cos(\theta)\psi(\theta) + \sin(\theta)\psi'(\theta) \right), \\ p(x,y) =& -\frac{r^{\lambda-1}}{1-\lambda} \left((1+\lambda)^2\psi'(\theta) + \psi'''(\theta) \right), \end{aligned}$$

where

$$\psi(\theta) = \sin((1+\lambda)\theta)\frac{\cos(\lambda\omega)}{1+\lambda} - \cos((1+\lambda)\theta) - \sin((1-\lambda)\theta)\frac{\cos(\lambda\omega)}{1-\lambda} + \cos((1-\lambda)\theta).$$

The value of the parameter λ is the smallest positive solution of

$$\sin(\lambda\omega) + \lambda\sin(\omega) = 0, \quad \text{where } \omega = \frac{3\pi}{2},$$

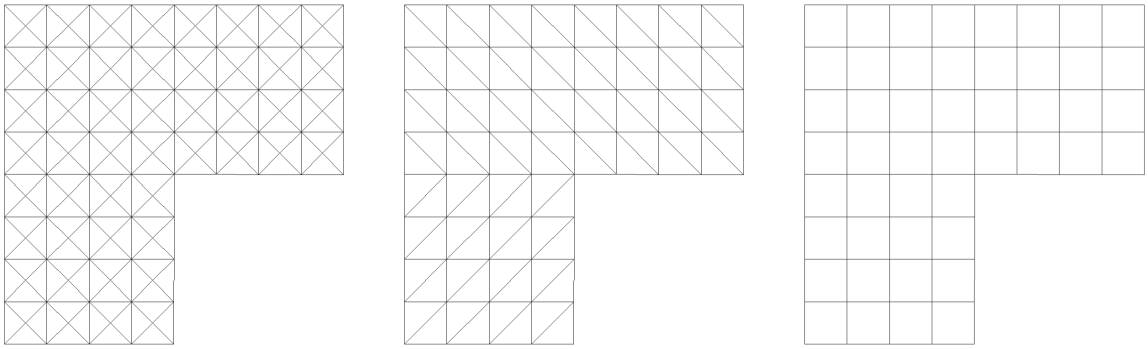
which is $\lambda \sim 0.54448373678246$. Note that $\mathbf{u} = (u_x, u_y)$ is solenoidal and $(\mathbf{u}, p) \in H^{1+\lambda}(\Omega)^2 \times H^\lambda(\Omega)$.
The singular solution for the Maxwell operator is defined, also in polar coordinates, as

$$\mathbf{b}(x,y) = \nabla \left(r^{\frac{2n}{3}} \sin\left(\frac{2n}{3}\right) \right), \qquad n \in \mathbb{N}^+.$$

Note that $\nabla \cdot \mathbf{b} = 0$ and $\nabla \times \mathbf{b} = \mathbf{0}$. The magnetic induction field $\mathbf{b} \in H^{\frac{2n}{3}}(\Omega)^2$ and therefore, $\mathbf{b} \notin H^1(\Omega)^2$ for $n = 1$.

This problem has been solved using several mesh structures. On one hand, we have used meshes with a macro-element structure, namely the crossbox element, which satisfies the approximability condition (3.20). On the other hand, we have used meshes composed of linear elements, both triangular meshes (P1) and quadrilateral meshes (Q1) that do not verify (3.20). Figure 3.1 shows an example of the three different mesh structures for $h = 2^{-2}$. Moreover, let us stress that we have solved the problem fully coupled with a non-linear tolerance of 10^{-4} and setting every physical parameter to 1.

For the three cases, the problem has been solved in several meshes with different mesh sizes, from the coarsest one with $h = 2^{-2}$ to the finest one with $h = 2^{-10}$, which consists of 12,58 million elements for the mesh composed of crossbox elements. Tables 3.1-3.6 contain the numerical error norms of the hydrodynamical and magnetic variables for the three cases, crossbox, P1 and Q1 elements respectively. In Tables 3.1, 3.3 and 3.5, related to the hydrodynamical variables, we show the numerical errors for the velocity in the L^2-norm and the H^1-norm, and the error for the pressure in the L^2-norm. In Tables 3.2, 3.4 and 3.6, related to the magnetic unknowns, we have listed the numerical errors for the magnetic induction in the L^2-norm and the $H(\mathbf{curl})$-norm, and the errors for the magnetic pseudo-pressure in both the L^2-norm and the H^1-norm. Furthermore, Figure 3.2 shows the convergence plots of the computed numerical errors. It is clearly seen that using a mesh with a macro-element structure is crucial in order to obtain the theoretical convergence rates to singular solutions. It is also shown that linear elements, both P1 and Q1 meshes, have a much lower convergence rate.



(a) Crossbox element (b) P1 element (c) Q1 element

Figure 3.1: Mesh structures.

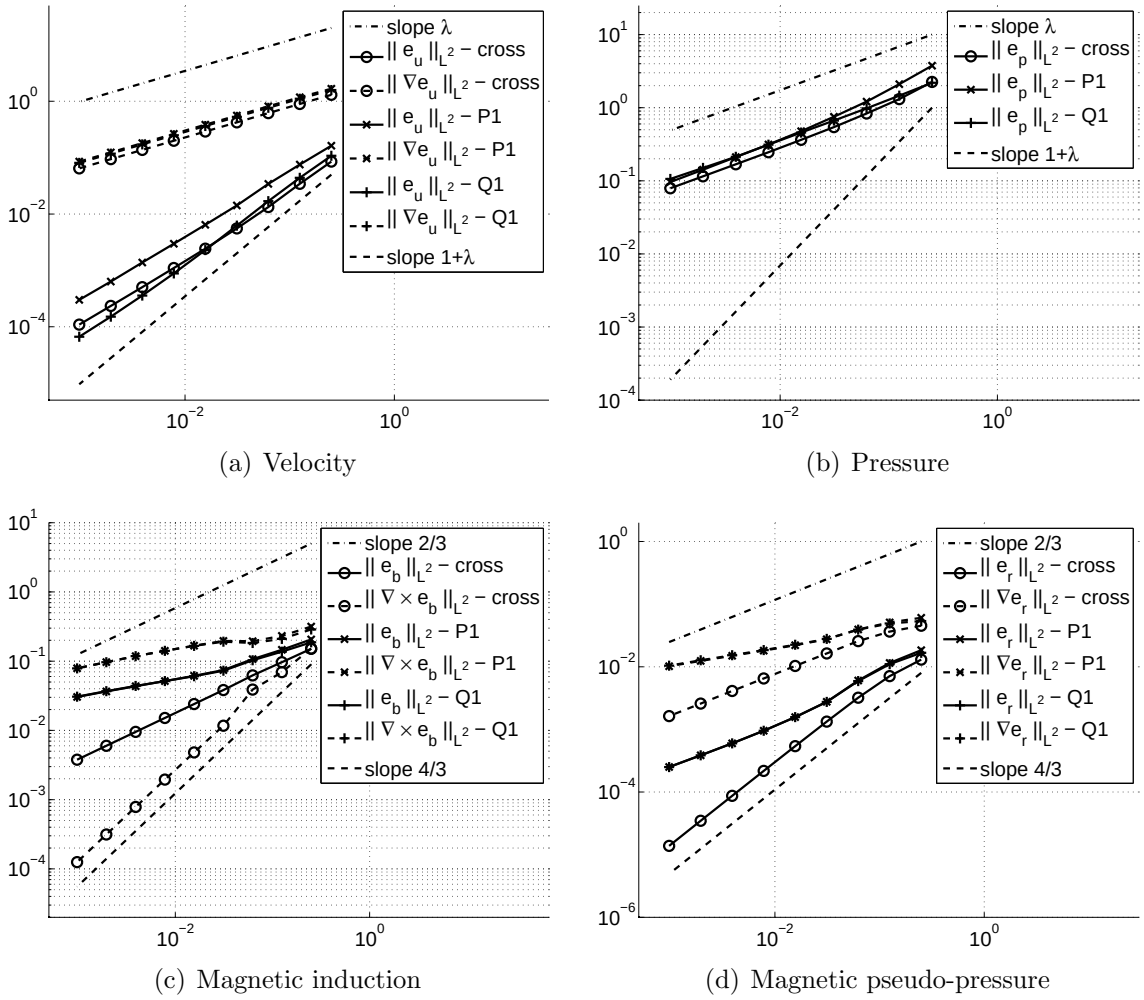
However, when using meshes of linear elements, both P1 and Q1, the method provides a solution for the magnetic induction $\mathbf{b}$ with spurious discontinuities around the corner. Figure 3.3 displays the magnetic induction solution for the three different mesh structures. Besides, Figure 3.4 shows a zoom of the same magnetic induction fields around the corner in order to highlight the discontinuities that appear when computing with meshes that do not satisfy condition (3.20).

Finally, we observe that 1) the convergence rates for $\|\nabla \mathbf{e}_{\mathrm{u}}\|$, $\|e_p\|$, $\|\mathbf{e}_{\mathrm{b}}\|$ and $\|\nabla e_r\|$ are *exactly* those predicted by the numerical analysis of the uncoupled stabilized Stokes and Maxwell problems, 2) super-convergence is observed for the error quantities $\|\mathbf{e}_{\mathrm{u}}\|$ and $\|\nabla \times \mathbf{e}_{\mathrm{b}}\|$. In fact, we can infer from the numerical analysis that the error due to the coupling term is affected by the two quantities that exhibit super-convergence. As a result, the coupling terms are small compared to those related to the uncoupled fluid and magnetic sub-problems, leading to the first observation.

3.8 Conclusions

The finite element approximation of the resistive MHD problem in Chapter 2 has been analyzed in this chapter. The formulation falls within the category of stabilized finite element methods and, as such, is intended to avoid the need for using finite element approximations satisfying the compatibility conditions of the continuous problem and dealing with ranges of the physical parameters in which first order derivatives dominate second order ones.

Particular features of the formulation analyzed are that it is of residual type, can be based on the VMS framework and the stabilization parameters are designed from the numerical analysis, accounting for the coupling between the fluid and magnetic sub-problems. However, the most salient feature is that it allows to converge to singular solutions even when using a continuous Lagrangian approximation for the magnetic induction field. To our knowledge, this is the first time this is achieved. From the technical



(a) Velocity

(b) Pressure

(c) Magnetic induction

(d) Magnetic pseudo-pressure

Figure 3.2: Convergence plots.

h	$\|\|\mathbf{e}_\mathrm{u}\|\|$	$\|\|\nabla\mathbf{e}_\mathrm{u}\|\|$	$\|\|e_p\|\|$
2^{-2}	$8.52 \cdot 10^{-2}$ (-)	$1.30 \cdot 10^{0}$ (-)	$2.25 \cdot 10^{0}$ (-)
2^{-3}	$3.43 \cdot 10^{-2}$ (1.31)	$8.96 \cdot 10^{-1}$ (0.54)	$1.31 \cdot 10^{0}$ (0.78)
2^{-4}	$1.33 \cdot 10^{-2}$ (1.37)	$6.17 \cdot 10^{-1}$ (0.54)	$8.34 \cdot 10^{-1}$ (0.65)
2^{-5}	$5.54 \cdot 10^{-3}$ (1.26)	$4.23 \cdot 10^{-1}$ (0.54)	$5.42 \cdot 10^{-1}$ (0.62)
2^{-6}	$2.43 \cdot 10^{-3}$ (1.19)	$2.91 \cdot 10^{-1}$ (0.54)	$3.63 \cdot 10^{-1}$ (0.58)
2^{-7}	$1.10 \cdot 10^{-3}$ (1.14)	$1.99 \cdot 10^{-1}$ (0.55)	$2.46 \cdot 10^{-1}$ (0.56)
2^{-8}	$5.04 \cdot 10^{-4}$ (1.13)	$1.37 \cdot 10^{-1}$ (0.54)	$1.68 \cdot 10^{-1}$ (0.55)
2^{-9}	$2.34 \cdot 10^{-4}$ (1.11)	$9.40 \cdot 10^{-2}$ (0.54)	$1.15 \cdot 10^{-1}$ (0.55)
2^{-10}	$1.09 \cdot 10^{-4}$ (1.10)	$6.44 \cdot 10^{-2}$ (0.55)	$7.89 \cdot 10^{-2}$ (0.54)

Table 3.1: Numerical errors for hydrodynamic unknowns and rate of convergence in brackets. Crossbox element.

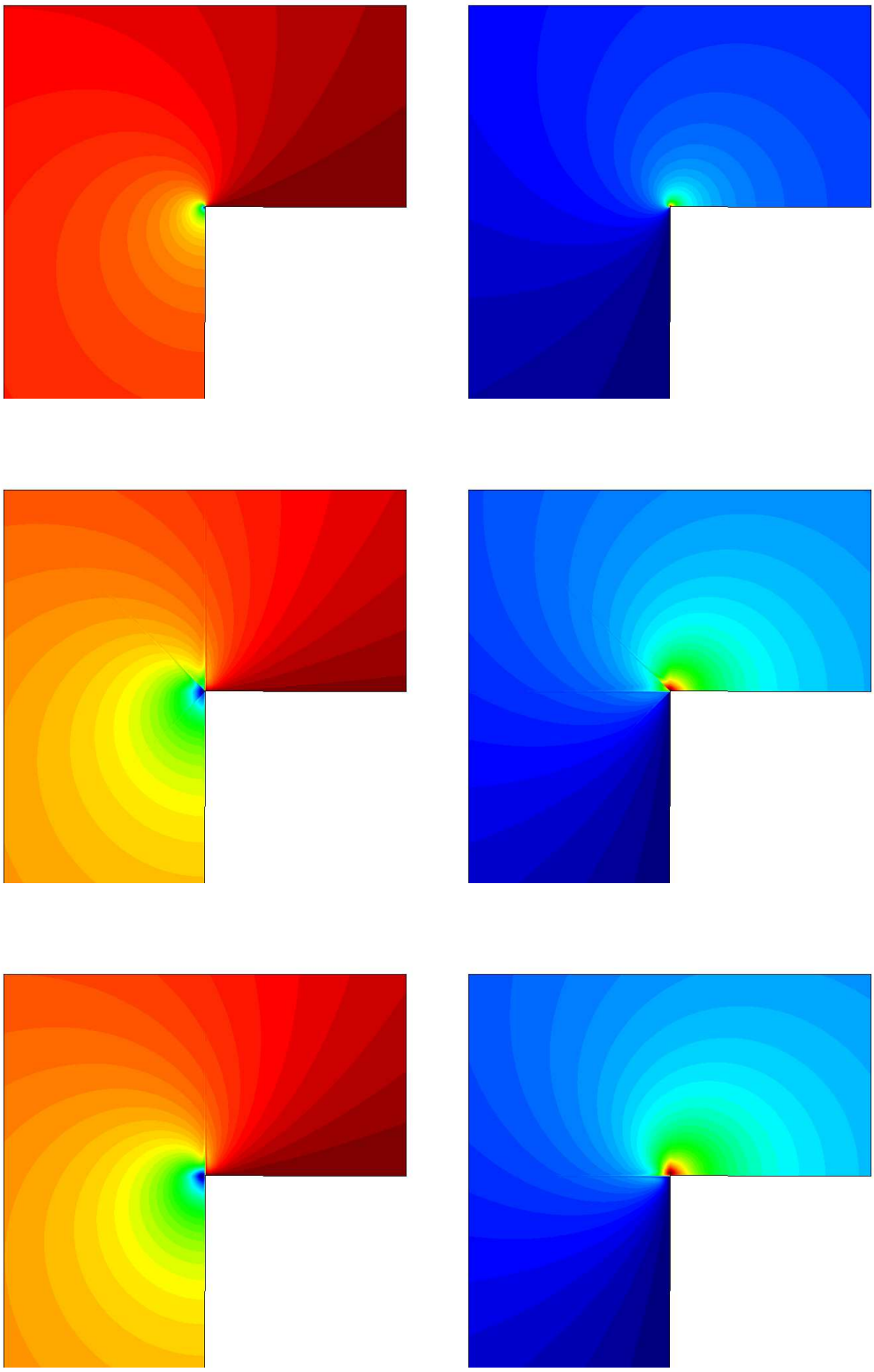


Figure 3.3: Magnetic induction solution for $h = 2^{-8}$. x-component (left) and y-component (right). Crossbox (top), P1 (middle) and Q1 (bottom) elements.

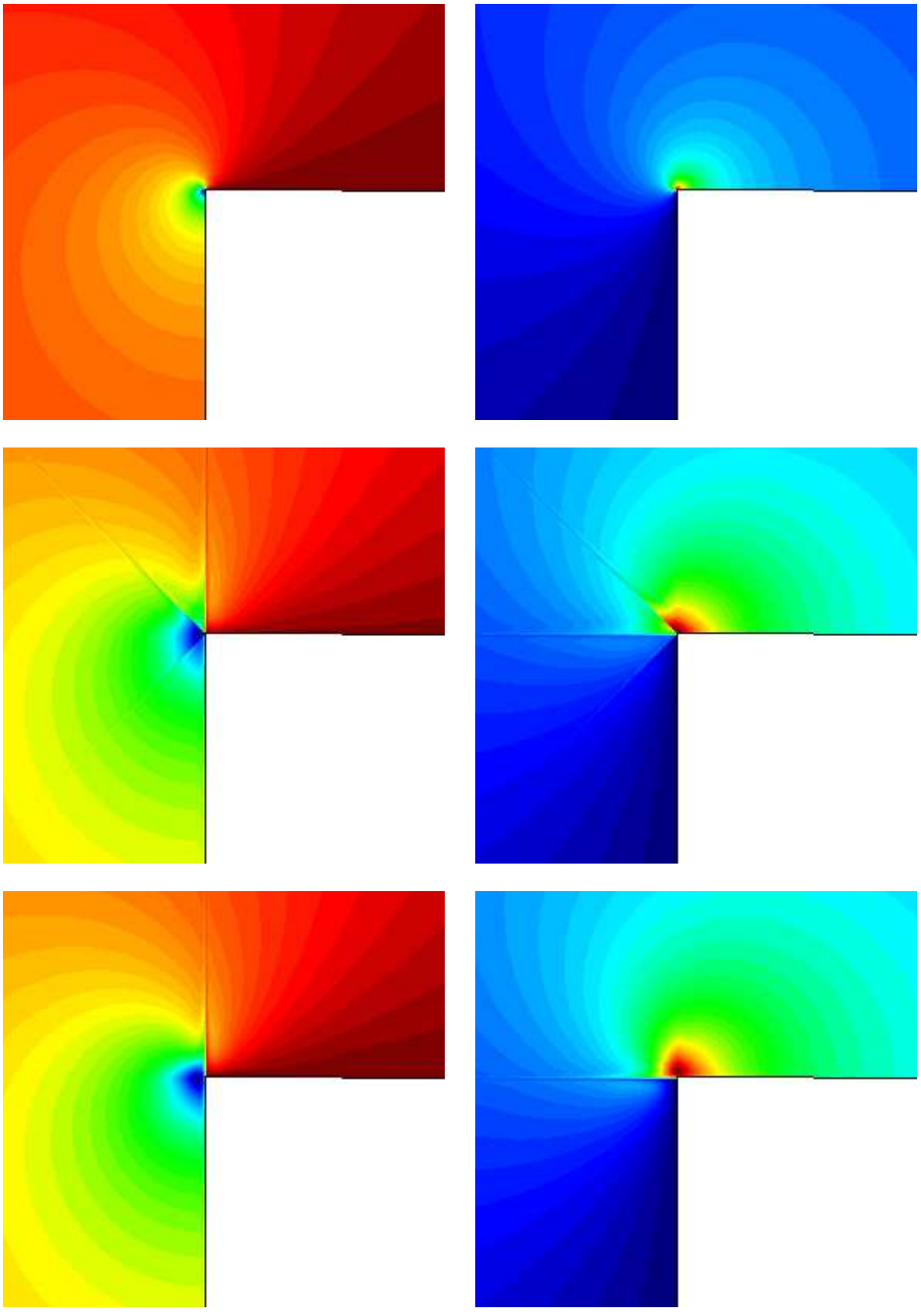


Figure 3.4: Zoom of the magnetic induction solution for $h = 2^{-8}$ around the corner. x-component (left) and y-component (right). Crossbox (top), P1 (middle) and Q1 (bottom) elements.

h	$\|\|\mathbf{e}_\mathrm{b}\|\|$	$\|\|\nabla \times \mathbf{e}_\mathrm{b}\|\|$	$\|\|e_r\|\|$	$\|\|\nabla e_r\|\|$
2^{-2}	$1.51 \cdot 10^{-1}$ (-)	$1.52 \cdot 10^{-1}$ (-)	$1.30 \cdot 10^{-2}$ (-)	$4.52 \cdot 10^{-2}$ (-)
2^{-3}	$9.66 \cdot 10^{-2}$ (0.64)	$6.94 \cdot 10^{-2}$ (1.13)	$7.08 \cdot 10^{-3}$ (0.88)	$3.64 \cdot 10^{-2}$ (0.31)
2^{-4}	$6.22 \cdot 10^{-2}$ (0.64)	$3.89 \cdot 10^{-2}$ (0.84)	$3.21 \cdot 10^{-3}$ (1.14)	$2.57 \cdot 10^{-2}$ (0.50)
2^{-5}	$3.83 \cdot 10^{-2}$ (0.70)	$1.17 \cdot 10^{-2}$ (1.73)	$1.33 \cdot 10^{-3}$ (1.27)	$1.63 \cdot 10^{-2}$ (0.66)
2^{-6}	$2.40 \cdot 10^{-2}$ (0.67)	$4.80 \cdot 10^{-3}$ (1.29)	$5.39 \cdot 10^{-4}$ (1.30)	$1.03 \cdot 10^{-2}$ (0.66)
2^{-7}	$1.51 \cdot 10^{-2}$ (0.67)	$1.95 \cdot 10^{-3}$ (1.30)	$2.17 \cdot 10^{-4}$ (1.31)	$6.50 \cdot 10^{-3}$ (0.66)
2^{-8}	$9.51 \cdot 10^{-3}$ (0.67)	$7.83 \cdot 10^{-4}$ (1.32)	$8.69 \cdot 10^{-5}$ (1.32)	$4.10 \cdot 10^{-3}$ (0.66)
2^{-9}	$5.99 \cdot 10^{-3}$ (0.67)	$3.14 \cdot 10^{-4}$ (1.32)	$3.47 \cdot 10^{-5}$ (1.32)	$2.58 \cdot 10^{-3}$ (0.67)
2^{-10}	$3.77 \cdot 10^{-3}$ (0.67)	$1.25 \cdot 10^{-4}$ (1.33)	$1.38 \cdot 10^{-5}$ (1.33)	$1.63 \cdot 10^{-3}$ (0.66)

Table 3.2: Numerical errors for magnetic unknowns and rate of convergence in brackets. Crossbox element.

h	$\|\|\mathbf{e}_\mathrm{u}\|\|$	$\|\|\nabla \mathbf{e}_\mathrm{u}\|\|$	$\|\|e_p\|\|$
2^{-2}	$1.63 \cdot 10^{-1}$ (-)	$1.67 \cdot 10^{0}$ (-)	$3.76 \cdot 10^{0}$ (-)
2^{-3}	$7.55 \cdot 10^{-2}$ (1.11)	$1.18 \cdot 10^{0}$ (0.50)	$2.10 \cdot 10^{0}$ (0.84)
2^{-4}	$3.42 \cdot 10^{-2}$ (1.14)	$8.14 \cdot 10^{-1}$ (0.54)	$1.21 \cdot 10^{0}$ (0.80)
2^{-5}	$1.42 \cdot 10^{-2}$ (1.27)	$5.59 \cdot 10^{-1}$ (0.54)	$7.51 \cdot 10^{-1}$ (0.69)
2^{-6}	$6.43 \cdot 10^{-3}$ (1.14)	$3.84 \cdot 10^{-1}$ (0.54)	$4.77 \cdot 10^{-1}$ (0.65)
2^{-7}	$2.97 \cdot 10^{-3}$ (1.11)	$2.64 \cdot 10^{-1}$ (0.54)	$3.14 \cdot 10^{-1}$ (0.60)
2^{-8}	$1.38 \cdot 10^{-3}$ (1.11)	$1.81 \cdot 10^{-1}$ (0.54)	$2.11 \cdot 10^{-1}$ (0.57)
2^{-9}	$6.34 \cdot 10^{-4}$ (1.12)	$1.24 \cdot 10^{-1}$ (0.55)	$1.42 \cdot 10^{-1}$ (0.57)
2^{-10}	$3.00 \cdot 10^{-4}$ (1.08)	$8.53 \cdot 10^{-2}$ (0.54)	$9.61 \cdot 10^{-2}$ (0.56)

Table 3.3: Numerical errors for hydrodynamic unknowns and rate of convergence in brackets. P1 element.

h	$\|\|\mathbf{e}_\mathrm{b}\|\|$	$\|\|\nabla\times\mathbf{e}_\mathrm{b}\|\|$	$\|\|e_r\|\|$	$\|\|\nabla e_r\|\|$
2^{-2}	$2.06\cdot10^{-1}$ (-)	$3.16\cdot10^{-1}$ (-)	$1.84\cdot10^{-2}$ (-)	$6.06\cdot10^{-2}$ (-)
2^{-3}	$1.46\cdot10^{-1}$ (0.50)	$2.32\cdot10^{-1}$ (0.45)	$1.15\cdot10^{-2}$ (0.68)	$5.11\cdot10^{-2}$ (0.25)
2^{-4}	$1.07\cdot10^{-1}$ (0.45)	$1.94\cdot10^{-1}$ (0.26)	$6.04\cdot10^{-3}$ (0.93)	$3.94\cdot10^{-2}$ (0.38)
2^{-5}	$7.44\cdot10^{-2}$ (0.52)	$1.96\cdot10^{-1}$ (-0.01)	$2.76\cdot10^{-3}$ (1.13)	$2.75\cdot10^{-2}$ (0.52)
2^{-6}	$6.13\cdot10^{-2}$ (0.28)	$1.67\cdot10^{-1}$ (0.23)	$1.57\cdot10^{-3}$ (0.81)	$2.21\cdot10^{-2}$ (0.32)
2^{-7}	$5.16\cdot10^{-2}$ (0.25)	$1.41\cdot10^{-1}$ (0.24)	$9.51\cdot10^{-4}$ (0.72)	$1.81\cdot10^{-2}$ (0.29)
2^{-8}	$4.35\cdot10^{-2}$ (0.25)	$1.18\cdot10^{-1}$ (0.26)	$5.98\cdot10^{-4}$ (0.67)	$1.50\cdot10^{-2}$ (0.27)
2^{-9}	$3.65\cdot10^{-2}$ (0.25)	$9.68\cdot10^{-2}$ (0.29)	$3.85\cdot10^{-4}$ (0.64)	$1.24\cdot10^{-2}$ (0.27)
2^{-10}	$3.04\cdot10^{-2}$ (0.26)	$7.89\cdot10^{-2}$ (0.29)	$2.51\cdot10^{-4}$ (0.62)	$1.02\cdot10^{-2}$ (0.28)

Table 3.4: Numerical errors for magnetic unknowns and rate of convergence in brackets. P1 element.

h	$\|\|\mathbf{e}_\mathrm{u}\|\|$	$\|\|\nabla\mathbf{e}_\mathrm{u}\|\|$	$\|\|e_p\|\|$
2^{-2}	$1.08\cdot10^{-1}$ (-)	$1.58\cdot10^{0}$ (-)	$2.19\cdot10^{0}$ (-)
2^{-3}	$4.39\cdot10^{-2}$ (1.30)	$1.11\cdot10^{0}$ (0.51)	$1.46\cdot10^{0}$ (0.58)
2^{-4}	$1.69\cdot10^{-2}$ (1.38)	$7.67\cdot10^{-1}$ (0.53)	$9.80\cdot10^{-1}$ (0.58)
2^{-5}	$6.23\cdot10^{-3}$ (1.44)	$5.29\cdot10^{-1}$ (0.54)	$6.67\cdot10^{-1}$ (0.56)
2^{-6}	$2.29\cdot10^{-3}$ (1.44)	$3.64\cdot10^{-1}$ (0.54)	$4.57\cdot10^{-1}$ (0.55)
2^{-7}	$8.82\cdot10^{-4}$ (1.38)	$2.50\cdot10^{-1}$ (0.54)	$3.12\cdot10^{-1}$ (0.55)
2^{-8}	$3.57\cdot10^{-4}$ (1.30)	$1.72\cdot10^{-1}$ (0.54)	$2.13\cdot10^{-1}$ (0.55)
2^{-9}	$1.51\cdot10^{-4}$ (1.24)	$1.18\cdot10^{-1}$ (0.54)	$1.50\cdot10^{-1}$ (0.51)
2^{-10}	$6.66\cdot10^{-5}$ (1.18)	$8.08\cdot10^{-2}$ (0.55)	$1.06\cdot10^{-1}$ (0.50)

Table 3.5: Numerical errors for hydrodynamic unknowns and rate of convergence in brackets. Q1 element.

h	$\|\|\mathbf{e}_b\|\|$	$\|\|\nabla \times \mathbf{e}_b\|\|$	$\|\|e_r\|\|$	$\|\|\nabla e_r\|\|$
2^{-2}	$1.87 \cdot 10^{-1}$ (-)	$2.87 \cdot 10^{-1}$ (-)	$1.67 \cdot 10^{-2}$ (-)	$5.38 \cdot 10^{-2}$ (-)
2^{-3}	$1.37 \cdot 10^{-1}$ (0.45)	$2.10 \cdot 10^{-1}$ (0.45)	$1.10 \cdot 10^{-2}$ (0.60)	$4.92 \cdot 10^{-2}$ (0.13)
2^{-4}	$1.03 \cdot 10^{-1}$ (0.41)	$1.83 \cdot 10^{-1}$ (0.20)	$5.91 \cdot 10^{-3}$ (0.90)	$3.92 \cdot 10^{-2}$ (0.33)
2^{-5}	$7.27 \cdot 10^{-2}$ (0.50)	$1.91 \cdot 10^{-1}$ (-0.06)	$2.75 \cdot 10^{-3}$ (1.10)	$2.79 \cdot 10^{-2}$ (0.49)
2^{-6}	$6.06 \cdot 10^{-2}$ (0.26)	$1.66 \cdot 10^{-1}$ (0.20)	$1.57 \cdot 10^{-3}$ (0.81)	$2.25 \cdot 10^{-2}$ (0.31)
2^{-7}	$5.15 \cdot 10^{-2}$ (0.23)	$1.41 \cdot 10^{-1}$ (0.24)	$9.50 \cdot 10^{-4}$ (0.72)	$1.85 \cdot 10^{-2}$ (0.28)
2^{-8}	$4.38 \cdot 10^{-2}$ (0.23)	$1.18 \cdot 10^{-1}$ (0.26)	$5.98 \cdot 10^{-4}$ (0.67)	$1.53 \cdot 10^{-2}$ (0.27)
2^{-9}	$3.69 \cdot 10^{-2}$ (0.25)	$9.66 \cdot 10^{-2}$ (0.29)	$3.85 \cdot 10^{-4}$ (0.64)	$1.27 \cdot 10^{-2}$ (0.27)
2^{-10}	$3.07 \cdot 10^{-2}$ (0.27)	$7.86 \cdot 10^{-2}$ (0.30)	$2.50 \cdot 10^{-4}$ (0.62)	$1.04 \cdot 10^{-2}$ (0.29)

Table 3.6: Numerical errors for magnetic unknowns and rate of convergence in brackets. Q1 element.

point of view, this possibility relies on the fact that we mimic the correct functional setting of the continuous problem at the discrete level.

The stability and convergence analysis in this chapter support the feasibility of our formulation, and complements the numerical experimentation in Chapter 2. We have restricted ourselves to some simplifying assumptions (quasi-uniform meshes, constant stabilization parameters, conforming finite element spaces) which have allowed us to avoid excessive technicalities but to highlight the main analytical reasons for the success of our formulation.[2]

[2]The analysis for non-degenerate meshes and variable stabilization parameters introduces some additional technicalities. We refer to [11, 50] for the numerical analysis of stabilized FE discretizations of Navier-Stokes and Maxwell problems respectively, in this more general setting.

Chapter 4

Operator splitting solvers for the resistive MHD problem

In this chapter we propose different splitting procedures for the transient incompressible MHD system that are unconditionally stable. We consider two levels of splitting, on one side we perform the segregation of the fluid pressure and magnetic pseudo-pressure from the vectorial fields computation. At the second level, the fluid velocity and induction fields are also decoupled. This way, we transform a fully coupled indefinite multi-physics system into a set of smaller definite ones, clearly reducing the CPU cost. With regard to the finite element approximation, we stick to an unconditionally convergent stabilized finite element formulation, since it introduces convection stabilization, allows to circumvent inf-sup conditions (clearly simplifying implementation issues) and is able to capture nonsmooth solutions of the magnetic sub-problem. However, residual-based finite element formulations are not suitable for segregation, since they lose the skew-symmetry of the off-diagonal blocks. Therefore, in this chapter we have proposed a novel term-by-term stabilization of the MHD system based on projections that is still unconditionally convergent.

Let us note here that we have slightly changed the notation for this chapter. The following work will be done at the discrete (algebraic) level and we have decided not to use bold symbols for the vectorial fields.

4.1 Introduction

Incompressible full magnetohydrodynamics (MHD) model incompressible, viscous and electrically conducting fluids under electromagnetic fields. Examples of such fluids are liquid metals, with important applications in the development of fusion technology and casting processes (see [61]). In these applications, the capability to simulate the corresponding set of partial differential equations is of great importance, e.g. in order to design the breeding blankets for the future fusion reactors, since no experimentation is possible nowadays.

A way to simulate MHD phenomena is to discretize this system by finite element techniques. However, the nature of the problem at hand is very complex. It couples the incompressible Navier-Stokes equations with a simplified version of the Maxwell system,

which corresponds to the induction equation and the divergence-free constraint for the magnetic field. Therefore, the resulting system has a double saddle point structure. A straightforward Galerkin finite element formulation of the problem requires to consider finite element spaces satisfying two different inf-sup conditions (see [111]). Alternative formulations have been derived, based on exact penalty formulations that can be obtained by the fact that the zero divergence condition is automatically satisfied from the induction equation; the resulting system transforms the magnetic sub-problem into a positive definite operator. This idea was followed by Gunzburger and co-workers in [76], but it requires enough regularity of the physical domain. Unfortunately, the numerical discretization of the resulting system does not convergence to the correct (physical) solution in general, leading to wrong results. A way to circumvent this problem has been proposed in [77], based on a weighted penalty term initally introduced by Costabel and Dauge for the Maxwell operator in [58]. On top of all that, the incompressible MHD system includes some convective terms that can deteriorate the performance of stable Galerkin finite element formulations when they become dominant if the mesh size is not reduced accordingly.

Residual-based stabilized finite element formulations, that include additional (stabilizing) terms keeping optimal convergence properties, have been considered so far. Initially, Gerbeau proposed in [72] a stabilized finite element formulation of the method in [76] that introduced convection stabilization. In [23, 24, 53] a further step was taken, facing the double saddle-point formulation (by adding a magnetic pseudo-pressure, as in [111]) but circumventing the inf-sup conditions via stabilization. In any case, the method was developed assuming the regularized functional setting for the magnetic field and again, the proposed method could not converge to nonsmooth solutions. The first stabilized formulation that always converges to the good (physical) solution has been proposed in Chapter 2, by considering a stabilized finite element formulation that mimics the correct functional setting, as in [111], but also introducing full convection stabilization. We refer to Chapter 2 for a detailed design of the method, its numerical analysis and numerical experiments.

Some of the applications of incompressible MHD (e.g. the design process of breeding blankets in fusion reactors) are very challenging, requiring large computational resources. The use of large scale computers, based on distributed platforms, require domain decomposition techniques (see e.g. [118]). In order to have scalable algorithms, coarse correction solvers are needed, e.g. by using FETI-DP methods or balanced domain decomposition [63,70,92,93]. The mathematical theory for these algorithms has been developed for positive definite linear systems, and it has only been extended to indefinite linear systems for very specific cases [103]. A previous step in order to deal with realistic large scale incompressible MHD simulations is to transform the original indefinite system into a set of definite ones. Furthermore, this way we are also decoupling the original multi-physics problem (coupling all the unknowns) into a set of possibly one-physics sub-problems, reducing CPU cost. This transformation of the original system is carried out by a set of methods that have different names in the literature: operator splitting, fractional step or segregated algorithms. They are widely used in computational fluid dynamics for incompressible flows (see e.g. [9, 75]) and are also related to block preconditioners and physics-based preconditioners [41, 65, 67]. The main objective of this work is to consider segregated algorithms for the incompressible MHD system that arises from a stabilized

finite element formulation of the problem.

Pressure segregation algorithms for the incompressible Navier-Stokes equations rely on the fact that the off-diagonal blocks that couple velocity and pressure, i.e. the gradient and divergence operator, are skew-symmetric. In this chapter, we proceed first with the (fluid and magnetic) pressure segregation using a so-called pressure correction algorithm (see [9, 47, 75]). The pressure segregation transforms the indefinite monolithic problem into a set of positive-definite systems. It allows now to use domain decomposition techniques with coarse solvers. However, the fluid velocity and the magnetic field are still coupled, in a convection-diffusion-reaction type system. Observing that this system has again skew-symmetric off-diagonal blocks, we can perform a subsequent splitting of the vectorial fields.

The original continuous problem satisfies the previous skew-symmetric properties and the Galerkin approximation straightforwardly inherits them. However, the stabilized residual-based formulations as the one in Chapter 2 modify the off-diagonal blocks in such a way that these properties are not true anymore, not allowing to obtain unconditionally stable splitting algorithms. As an alternative, in this work we design a projection-based term-by-term stabilization that introduces only those terms that are required for stability purposes, and keep accuracy by introducing particular projections. There are different types of schemes that can be considered in this family, which is experiencing an increasing interest (see e.g. [21, 30, 48, 50]). The key aspect of the new formulation is that it keeps the unconditional convergence of the stabilized formulation in Chapter 2 and the off-diagonal skew-symmetric properties of the continuous problem. We have considered L^2 orthogonal projections onto the finite element spaces in the design of the stabilization terms, the so-called orthogonal subscales approach in [48,50], but other approaches could also fit into the framework, as the local projections in [21, 30].

Finally, let us mention that the label MHD is applied to a large variety of physical phenomena. Herein we are focused on incompressible resistive MHD applied to liquid metals, but different MHD systems are applied to other physical problems. A different MHD application is the coupling of electromagnetics with nonlinear mechanics of metals which undergo very fast plastic deformations and even transitions to liquid, where hydrocodes are used for the simulations (see e.g. [119]). Those applications usually make use of Lagrangian frames of reference and remap techniques. As an example, we refer to [107], where an operator splitting technique is used for the first-order version of the Maxwell equations, as well as an explicit treatment of the magnetic-solid coupling.

The outline of the chapter is the following. In Section 4.2 the continuous problem is stated in strong and weak form, as well as its Galerkin finite element discretization. Section 4.3 is devoted to the linearization of the algorithm, considering both implicit and semi-implicit algorithms. The term-by-term stabilized finite element formulation that preserves the block-diagonal skew-symmetry is presented in Section 4.4. Section 4.5 deals with the splitting techniques, first for the pressure segregation and subsequently the vectorial fields uncoupling. Numerical experiments are included in Section 4.6 and some conclusions are drawn in Section 4.7.

4.2 Problem statement

4.2.1 Continuous problem

The incompressible MHD system of partial differential equations consists of the (incompressible) Navier-Stokes equations coupled to the (simplified) Maxwell equations via the Lorentz force. It reads as follows: find the velocity field $u(x,t)$, the (kinematic) pressure $p(x,t)$, the magnetic induction field $b(x,t)$ and the magnetic pseudo-pressure $r(x,t)$[1] such that

$$\partial_t u + (u\cdot\nabla)u - \nu\Delta u - (\nabla\times b)\times \varrho b + \nabla p = f_u, \tag{4.1a}$$
$$\nabla\cdot u = 0, \tag{4.1b}$$
$$\varrho\partial_t b + \lambda\nabla\times\nabla\times b - \nabla\times(u\times\varrho b) + \nabla r = f_b, \tag{4.1c}$$
$$\nabla\cdot b = 0, \tag{4.1d}$$

in $(x,t)\in\Omega\times(0,T)$, where ∂_t stands for the partial time derivative, $\Omega\subset\mathbb{R}^d$ is the spatial open bounded domain filled by the fluid, d being the space dimension, and $(0,T)$ is the time interval of interest. f_u and f_b are the forcing terms, f_b being solenoidal. We define $\nu := \mu_{\mathrm{f}}\rho^{-1}$, $\varrho := (\rho\mu_{\mathrm{m}})^{-1}$ and $\lambda := (\rho\mu_{\mathrm{m}}^2\sigma)^{-1}$, where ρ is the fluid density, μ_{f} the fluid viscosity, μ_{m} the magnetic permeability and σ the electric conductivity. In this work, we consider all physical properties constant. These equations must be supplemented with appropriate boundary and initial conditions. The initial conditions are:

$$u(x,0) = u_0(x), \qquad b(x,0) = b_0(x),$$

in $x\in\Omega$, where b_0 and u_0 must be solenoidal. Let us assume Dirichlet-type boundary conditions for simplicity:

$$u = u_\Gamma, \quad n\times b = n\times b_\Gamma, \quad r = 0$$

on the domain boundary $\partial\Omega$, where $u_\Gamma(x,t)$ and $n\times b_\Gamma$ are the velocity trace and tangent trace of the magnetic field to be prescribed, respectively; $n(x)$ denotes the normal vector on $\partial\Omega$ pointing outwards from Ω. In the analysis, we will assume homogeneous boundary conditions.

4.2.2 Weak form

Let us introduce some notation to set up the weak form of system (4.1). Let $L^p(\Omega)$, with $1\le p<\infty$, denote the space of pth-integrable real-valued functions defined on Ω for the Lebesgue measure; the subset of functions in $L^2(\Omega)$ with zero mean value is denoted as $L^2_0(\Omega)$. $L^\infty(\Omega)$ denotes the space of essentially bounded functions. As usual, Sobolev spaces of functions whose derivatives of order up to m belong to $L^2(\Omega)$ are denoted by $H^m(\Omega)$; $H^1_0(\Omega)$ is the subspace of $H^1(\Omega)$ of functions vanishing on $\partial\Omega$. The space of vector functions with their components in an arbitrary functional space X is denoted as

[1] We refer to Chapter 2 for the need to introduce this additional unknown to enforce the divergence-free constraint over b at the discrete problem. At the continuous level, we can easily check that r is identically zero.

X^d, where d is the number of space dimension. The subspace of $L^2(\Omega)^d$ vector functions with their curl in $L^2(\Omega)^d$ is denoted as $H(\mathrm{curl};\Omega)$. $H_0(\mathrm{curl};\Omega)$ is the subspace of functions in $H(\mathrm{curl};\Omega)$ with null tangent trace.

The inner product of $f, g \in L^2(\Omega)$ is represented as (f, g). Given an arbitrary Banach space X, its dual space is denoted by X'. The duality pairing between $f \in X$ and $g \in X'$ is represented by $\langle f, g\rangle_{X\times X'}$, which denotes the integral $\int_\Omega fg$ whenever it makes sense; this in particular applies for the duality between $H_0^1(\Omega)$ and its topological dual $H^{-1}(\Omega)$. The sub-script denoting the spaces in which the duality pairing is understood will be omitted when there is no confusion. The same notation is used for both scalar and vector valued functions. Furthermore, when X is a finite-dimensional space, we will also make use of the notation $f \cdot g$ to denote the duality pairing.

Given a normed functional space X, its norm is written as $\|\cdot\|_X$, with the abbreviation $\|\cdot\|_{L^2(\Omega)} \equiv \|\cdot\|$. For a semi-positive definite operator $P : X \longrightarrow X'$, we define the semi-norm $|x|_P^2 := \langle x, Px\rangle$ for any $x \in X$.

In the subsequent analyses, we will extensively use the Cauchy-Schwarz inequality $\langle f, g\rangle_{X\times X'} \leq \|f\|_X\|g\|_{X'}$, that holds for an arbitrary Banach space X, as well as the generalized Young's inequality $2ab \leq \alpha a^2 + \frac{1}{\alpha}b^2$, for any positive value α. Finally, the symbol $\lesssim$ is used to denote $\leq$ up to positive constants.

Since the aim of this work is to consider a Faedo-Galerkin approximation of system (4.1), we proceed to state the problem in a weak sense. Let us consider the following functional spaces:

$$V_u \equiv H_0^1(\Omega)^d, \quad V_b \equiv H_0(\mathrm{curl};\Omega), \quad Q_p \equiv L_0^2(\Omega), \quad Q_r \equiv H_0^1(\Omega).$$

We define the dimensionally consistent norms:

$$\|u\|_{V_u} := \frac{\nu^{\frac{1}{2}}}{L_0}\|u\| + \nu^{\frac{1}{2}}\|\nabla u\|, \qquad \|b\|_{V_b} := \frac{\lambda^{\frac{1}{2}}}{L_0}\|b\| + \lambda^{\frac{1}{2}}\|\nabla\times b\|,$$
$$\|p\|_{Q_p} := \frac{1}{\nu^{\frac{1}{2}}}\|p\|, \qquad \|r\|_{Q_r} := \frac{1}{\lambda^{\frac{1}{2}}}\|r\| + \frac{L_0}{\lambda^{\frac{1}{2}}}\|\nabla r\|,$$

where L_0 is a characteristic dimension length of the problem. The weak form of the problem reads as follows: find $(u(t), p(t), b(t), s(t)) \in V_u \times Q_p \times V_b \times Q_r$ such that

$$M_u\partial_t u + K_u(u,u) + C_u(b,b) + G_u p = f_u \quad \text{in} \quad V_u', \qquad D_u u = 0 \quad \text{in} \quad Q_p', \tag{4.2a}$$
$$\varrho M_b\partial_t b + K_b b + C_b(b,u) + G_b r = f_b \quad \text{in} \quad V_b', \qquad D_b b = 0 \quad \text{in} \quad Q_r', \tag{4.2b}$$

almost everywhere in $t \in (0, T)$, where the operators involved in (4.2) are given by (4.1) and the definition of weak derivative as follows.

For the fluid sub-problem, we define the linear operators $M_u : V_u \longrightarrow V_u'$ as $M_u v := (v, \cdot)$, $G_u : Q_p \longrightarrow V_u'$ as $G_u q := -(q, \nabla\cdot(\cdot))$ and $D_u := -G_u^T$, where the superscript T indicates the transpose operator.

The bilinear operator $K_u(u,u) : V_u \times V_u \longrightarrow V_u'$ is split into its symmetric and skew-symmetric parts $K_u(w,v) := K_{\mathrm{sym}}v + K_{\mathrm{skw}}(w,v)$, where

$$K_{\mathrm{sym}}v := \nu\,(\nabla v, \nabla(\cdot)) \quad \text{and} \quad K_{\mathrm{skw}}(w,v) := \left\langle (w\cdot\nabla)v + \frac{1}{2}(\nabla\cdot w)v, \cdot \right\rangle_{V_u'\times V_u}$$

include the viscous and convection terms, respectively.[2] The Lorentz force term $C_u : V_b \times V_b \longrightarrow V_u'$ reads as $C_u(d,c) := -\langle (\nabla \times c) \times \varrho d, \cdot \rangle_{V_u' \times V_u}$.

Analogously, for the magnetic sub-problem $M_b : V_b \longrightarrow V_b'$ is defined as $M_b c := (c, \cdot)$, $K_b : V_b \longrightarrow V_b'$ is defined as $K_b c := \lambda(\nabla \times c, \nabla \times (\cdot))$, $G_b : Q_r \longrightarrow V_b'$ as $G_b s := (\nabla s, \cdot)$ and $D_b = -G_b^T$. The bilinear coupling operator $C_b : V_b \times V_u \longrightarrow V_b'$ is defined as $C_b(d,u) := -\langle \nabla \times (u \times \varrho d), \cdot \rangle_{V_b' \times V_b}$.

The operator-type notation of the problem in (4.2) will be used as long as possible, in order to make the exposition clear and concise. Furthermore, we will make use of the following notation for the bilinear operators. We define the linear operator $K_{\text{skw}}(w) : V_u \longrightarrow V_u'$ as $K_{\text{skw}}(w, \cdot)$, and analogously for $C_u(d) := C_u(d, \cdot)$ and $C_b(d) := C_b(d, \cdot)$. We note that $C_u(d) = -C_b(d)^T$, for any $d \in V_b$.

The well-posedness of (4.2) comes from the coercivity of $K_u(u)$, and the inf-sup condition

$$\beta_u \leq \inf_{q \in Q_p} \sup_{v \in V_u} \frac{\langle G_u q, v \rangle}{\|q\|_{Q_p} \|v\|_{V_u}} \tag{4.3}$$

for the fluid problem, as well as

$$\beta_{\text{m}} \leq \inf_{(c,r) \in V_b \times Q_r} \sup_{(d,t) \in V_b \times Q_r} \frac{c_{\text{Mxw}}(c,r,d,t)}{(\|c\|_{V_b} + \|r\|_{Q_r}) \cdot (\|d\|_{V_b} + \|t\|_{Q_r})} \tag{4.4}$$

for the magnetic problem, where $c_{\text{Mxw}}(c,r,d,t) := \langle K_b c, d \rangle + \langle G_b r, d \rangle + \langle D_b c, t \rangle$ is the Maxwell operator (see [13, Theorem 2.1]). We refer to [73] for a detailed mathematical analysis of system (4.2); in particular, it is proved that the problem is well-posed and satisfies an energy inequality.

4.2.3 Galerkin finite element approximation and time integration

Let $\mathcal{T}_h = \{k\}$ be a finite element (FE) partition of the domain Ω. For simplicity we assume Ω polyhedral and $\mathcal{T}_h$ quasi-uniform, of diameter h. FE spaces and functions are identified with the subscript h. Only conforming approximations are considered, i.e. the FE spaces where the unknowns are sought satisfy $V_{uh} \subset V_u$, $V_{bh} \subset V_b$, $Q_{ph} \subset Q_p$ and $Q_{rh} \subset Q_r$. Further, we denote by $\pi_{X_h}(x)$ the $L^2(\Omega)$ projection of a function x over the FE space X_h, and $\pi_{X_h}^{\perp}(x) := x - \pi_{X_h}(x)$.

In order to state the Galerkin approximation, we consider the discrete version of the continuous operators in (4.2). Given an operator $A : X \longrightarrow X'$ and a FE space $X_h \subset X$, we define its discrete operator (matrix, for the linear case) $A_h : X_h \longrightarrow X_h'$ as follows: $\langle A_h x_h, y_h \rangle := \langle A x_h, y_h \rangle$ for any $x_h, y_h \in X_h$. For the sake of conciseness, we will omit the h subscript for the discrete operators, since the continuous counterpart will not be used onwards. On the other hand, the linear functional $f_u \in V_u'$ must be replaced by $f_{uh} \in V_{uh}'$ such that $\langle f_{uh}, v_h \rangle = \langle f_u, v_h \rangle$ for any $v_h \in V_{uh}$ (analogously for f_b). For the sake of conciseness, we will omit the h subscript for FE functions, linear functionals on

[2] We have considered the skew-symmetric form of the convective term, instead of $\langle (w \cdot \nabla) v, \cdot \rangle_{V_u' \times V_u}$, which is only skew-symmetric for w solenoidal. This way, we inherit the skew-symmetry of the operator at the discrete level.

FE spaces and discrete operators, since their continuous counterparts will not be used onwards if not explictly stated.

Let us consider a uniform partition of the time interval $(0,T)$ into N elements (t^n, t^{n+1}) for $n = 0, ..., N-1$, where $t^n := n\delta t$. The element size is denoted by $\delta t := \frac{T}{N}$. With regard to time integration, we consider the θ-method.[3] We can split our unknowns into the primal unknowns $y := (u, b)$ and the Lagrange multipliers $z := (p, r)$. This way, the time-continuous, spatial FE formulation can be written as a differential-algebraic equation:

$$\dot{y}(t) = A(y(t), z(t), t), \qquad 0 = C(y(t), t),$$

where $y(t) \in \mathbb{R}^n$, $z(t) \in \mathbb{R}^m$, $A(t) : \mathbb{R}^{n+m} \to \mathbb{R}^n$ and $C(t) : \mathbb{R}^{n+m} \to \mathbb{R}^m$. Using the definitions

$$\delta y^{n+1} = y^{n+1} - y^n, \qquad \dot{y}^{n+1} = \frac{\delta y^{n+1}}{\delta t}, \qquad y^{n+\theta} = \theta y^{n+1} + (1-\theta) y^n, \tag{4.5}$$

(analogously for $z^{n+\theta}$), the time discretization of this problem with the θ-method simply reads as:

$$\dot{y}^{n+1} = A(y^{n+\theta}, z^{n+\theta}, t^{n+\theta}), \qquad 0 = C(y^{n+\theta}, t^{n+\theta}), \qquad \text{for } \theta \in [0,1].$$

Assuming that the initial condition satisfies the constraint, i.e. $C(y^0, 0) = 0$, and that the constraint operator $C(\cdot)$ is linear, we can easily check that the previous problem is equivalent to

$$\dot{y}^{n+1} = A(y^{n+\theta}, z^{n+1}, t^{n+\theta}), \qquad 0 = C(y^{n+1}, t^{n+1}),$$

where we have used the shift $z^{n+1} \leftarrow z^{n+\theta}$; it does not affect the algorithm, since no time derivatives of z appear in the equation. Even though the solution of both systems is the same (up to the pressure shift), the second one allows to simplify the analysis of the pressure segregation algorithms when using $\theta = \frac{1}{2}$ in Theorem 4.4.

Once the conforming FE spaces V_{uh}, Q_{ph}, V_{bh} and Q_{rh} have been chosen, we can state the discretization of (4.2) by using a Galerkin FE approximation in space and the θ-method in time. The fully discrete problem at every time step value reads as: given $(u^n, b^n) \in V_{uh} \times V_{bh}$, find the solution $(u^{n+1}, p^{n+1}, b^{n+1}, r^{n+1}) \in V_{uh} \times Q_{ph} \times V_{bh} \times Q_{rh}$ of the problem:

$$\frac{1}{\delta t} M_u \delta u^{n+1} + K_u(u^{n+\theta}, u^{n+\theta}) + C_u(b^{n+\theta}, b^{n+\theta}) + G_u p^{n+1} = f_u^{n+\theta}, \quad D_u u^{n+1} = 0\ , \tag{4.6a}$$

$$\frac{\varrho}{\delta t} M_b \delta b^{n+1} + K_b b^{n+\theta} + C_b(b^{n+\theta}, u^{n+\theta}) + G_b r^{n+1} = f_b^{n+\theta}, \quad D_b b^{n+1} = 0, \tag{4.6b}$$

for $n = 0, \ldots, N-1$. The problem is started e.g. with $u^0 = \pi_{V_{uh}}(u_0)$ and $b^0 = \pi_{V_{bh}}(b_0)$.

In order to get a well-posed discrete problem (4.6), the discrete operators must satisfy the following assumption.

Assumption 4.1. *The discrete operators in (4.6) satisfy the following properties:*

1. *K_u is coercive with respect to V_{uh}, i.e. $\langle K_u v, v\rangle \geq \alpha_u \|v\|^2_{V_u}$ for any $v \in V_{uh}$, with $\alpha_u > 0$.*

[3] The extension of this work to backward differencing (BDF) time integrators is straightforward.

2. *K_b is coercive with respect to* $\mathrm{Ker}(D_b)$ *, i.e.* $\langle K_b c, c\rangle \geq \alpha_b \|c\|^2_{V_b}$ *for any* $c \in V_{bh}$ *such that* $D_b c = 0$*, with* $\alpha_b > 0$*.*

3. *For any* $q \in Q_{ph}$*, there exists* $\beta_u > 0$ *such that* $\beta_u \|q\|_{Q_p} \leq \sup_{v \in V_{uh}} \frac{\langle G_u(q), v\rangle}{\|v\|_{V_u}}$*.*

4. *For any* $s \in Q_{rh}$ *there exists* $\beta_b > 0$ *such that* $\beta_b \|s\|_{Q_r} \leq \sup_{c \in V_{bh}} \frac{\langle G_b(s), c\rangle}{\|c\|_{V_b}}$*.*

5. $G_u = -D_u^T$*,* $G_b = -D_b^T$ *and* $C_u(d) = -C_b(d)^T$*, for any* $d \in V_{bh}$*.*

The first and last properties are automatically inherited from the continuous problem for any conforming FE space, since we have considered the skew-symmetric form of the convective term. Unfortunately, the other three properties are not true in general, introducing restrictions over the FE spaces to be used. Property 3 is the classical discrete inf-sup condition for the fluid problem, which is known to be satisfied for some particular choices of $V_{uh} \times Q_{ph}$ (see e.g. [34]). Properties 2 and 4 are known to be satisfied for the so-called Nédélec FEs (see [99, 100]) and are enough to satisfy the discrete version of (4.4). We refer to [73, 111] for a detailed discussion about the numerical approximation of the full incompressible MHD system (4.2) using a Galerkin FE approximation.

Let us introduce the following notation, $\beta_\theta(P; v, w) := (\theta - \frac{1}{2})|v - w|^2_P$, for an arbitrary positive-definite (PD) operator P. In the next theorem, we state the stability of the discrete (Galerkin) problem (4.6).

Theorem 4.2. *Given a set of FE spaces* V_{uh}*,* V_{bh}*,* Q_{ph} *and* Q_{rh} *that satisfy Assumption 4.1, the sequence of solutions of system (4.6) for* $m = 0, \ldots, N-1$ *satisfies the energy inequality:*

$$\begin{aligned} &\|u^{m+1}\|^2 + \varrho\|b^{m+1}\|^2 + \sum_{n=0}^{m} (\beta_\theta(M_u; u^{n+1}, u^n) + \varrho\beta_\theta(M_b; b^{n+1}, b^n)) \\ &+ \sum_{n=0}^{m} \delta t(\alpha_u \|u^{n+\theta}\|^2_{V_u} + \alpha_b \|b^{n+\theta}\|^2_{V_b}) \leq \|u_0\|^2 + \varrho\|b_0\|^2 \\ &+ \sum_{n=0}^{m} \delta t(\frac{1}{\alpha_u}\|f_u^{n+\theta}\|_{V_u'} + \frac{1}{\alpha_b}\|f_b^{n+\theta}\|_{V_b'}), \end{aligned} \tag{4.7}$$

with $\theta \in [\frac{1}{2}, 1]$*.*

Proof. Using the skew-symmetry in property 5 of Assumption 4.1, together with the fact that $(a-b)(\theta a + (1-\theta)b) = \frac{1}{2}(a^2 - b^2 + (2\theta - 1)(a-b)^2)$, we get:

$$\begin{aligned} &|u^{n+1}|^2_{M_u} - |u^n|^2_{M_u} + \beta_\theta(M_u; u^{n+1}, u^n) + 2\delta t|u^{n+\theta}|^2_{K_{\mathrm{sym}}} \\ &+ \varrho|b^{n+1}|^2_{M_b} - \varrho|b^n|^2_{M_b} + \varrho\beta_\theta(M_b; b^{n+1}, b^n) + 2\delta t|b^{n+\theta}|^2_{K_b} = 2\delta t\langle f_u^{n+\theta}, u^{n+\theta}\rangle + 2\delta t\langle f_b^{n+\theta}, b^{n+\theta}\rangle. \end{aligned}$$

Now, using properties 1 and 2 in Assumption 4.1, as well as Cauchy-Schwarz's and Young's inequality for the right hand side, and adding up for $n = 0, \ldots, m$, we prove the theorem. □

Remark 4.1. *Pressure stability is obtained* a posteriori *via the inf-sup conditions (properties 3 and 4) in Assumption 4.1.*

4.3 Linearization and semi-implicit algorithms

In this section, we consider a Picard-type linearization of the problem that is unconditionally stable when considering semi-implicit techniques, i.e. no nonlinear iterations are performed. Given an arbitrary bilinear operator $P(x, y)$, the Picard linearization of $P(x, x)$ (with respect to the first argument) in the vicinity of a point $\bar{x}$ is obtained approximating $P(x, x) \approx P(\bar{x}, x)$. We use this linearization for K_{skw}, C_u and C_b. Since all these operators are skew-symmetric with respect to the second argument, their Picard linearizations keep the skew-symmetry. Let us denote by $\bar{u}^{n+\theta}$ an approximation of $u^{n+\theta}$ (idem for $b^{n+\theta}$). At every time step value, the *nonlinear* FE system (4.6) can always be stated in compact algebraic form as:

$$A(u^{n+\theta}, b^{n+\theta})x^{n+\theta} = g^{n+\theta}(u^{n+\theta}, b^{n+\theta}; u^n, b^n), \quad \text{where } x^{n+\theta} := \left[u^{n+\theta}, b^{n+\theta}, p^{n+\theta}, r^{n+\theta}\right]^T,$$

and the definition of A and g is straightforward from (4.6) and (4.5). At this point, we can consider semi-implicit or implicit algorithms, depending on the treatment of the linearization. An *implicit* formulation of the system at t^{n+1} reads as follows: given the values (u^n, b^n) from the previous step, choose an initial value $(u^{n+\theta,0}, b^{n+\theta,0})$, e.g. $(u^{n+\theta,0}, b^{n+\theta,0}) \leftarrow (u^n, b^n)$, set $m = 0$ and compute:

$$A(u^{n+\theta,m}, b^{n+\theta,m})x^{n+\theta,m+1} = g^{n+\theta}(u^{n+\theta,m}, b^{n+\theta,m}; u^n, b^n), \qquad m \leftarrow m + 1$$

where m is the nonlinear iteration counter, till an appropriate convergence criterium is satisfied.

Alternatively, we can consider a *semi-implicit* formulation: given the values (u^n, b^n), compute

$$A(u^{*,n+\theta}, b^{*,n+\theta})x^{n+\theta} = g^{n+\theta}(u^{*,n+\theta}, b^{*,n+\theta}; u^n, b^n), \tag{4.8}$$

where $u^{*,n+\theta}$ and $b^{*,n+\theta}$ are extrapolations of $u^{n+\theta}$ and $b^{n+\theta}$ respectively. In this case, we have to solve one linear system per time step value. However, the quality of the extrapolation affects the error in time of the method. For instance, a second order semi-implicit formulation can be obtained using the Crank-Nicholson time integrator, i.e. $\theta = \frac{1}{2}$, and a second order extrapolation,[4]namely $u^{*,n+\frac{1}{2}} = \frac{3}{2}u^n - \frac{1}{2}u^{n-1}$ and $b^{*,n+\frac{1}{2}} = \frac{3}{2}b^n - \frac{1}{2}b^{n-1}$.

Corollary 4.1. *The semi-implicit formulation (4.8) satisfies the energy inequality (4.7), for any choice of the linearization* $(u^{*,n+\theta}, b^{*,n+\theta})$ *and* $\theta \in [\frac{1}{2}, 1]$.

Proof. The proof of the semi-implicit formulation is identical to the one for the fully implicit (nonlinear) formulation, since the skew-symmetry of K_{skw}, C_u and C_b is still true after linearization, independently on the point with respect to which linearization is performed. □

Unfortunately, the resulting system matrix is fully coupled, in the sense that we have to solve for all the unknowns together. Furthermore, the double saddle-point structure of the system makes the monolithic matrix indefinite, which deteriorates the performance

[4]In order to keep second order accurate results, we can perform nonlinear iterations at $n = 1$, obtaining a second order accurate value for (u^1, b^1).

of direct and iterative solvers (see [67]). In what follows we work on a stabilized FE formulation that avoids the need to use inf-sup stable approximations and introduces convection stabilization. Further, we consider splitting methods that decouple the full monolithic system into smaller sub-problems that are positive-definite while keeping unconditional stability and energy bounds. We consider a two-step splitting: in the first step, the pressures p and r are segregated from the vectorial fields u and b, and in the second step the computation of the vectorial fields is also decoupled. We will see in the next sections that unconditionally stable segregation techniques require a particular type of stabilization, as the one introduced below.

4.4 Term-by-term stabilized finite element formulation

The crude Galerkin approximation of the full incompressible MHD problem (4.6) presents some drawbacks, namely:

- Galerkin FE formulations of saddle-point problems require FE spaces that satisfy the discrete inf-sup conditions in Assumption 4.1 (properties 2, 3 and 4). Since the inf-sup conditions for the fluid and magnetic sub-problems are different, it would lead to different interpolation spaces for different unknowns. It complicates the implementation issues, e.g. the database structure, the computation of the coupling terms and the graphs needed for the compressed storage of the system matrix.

- The lumping of mass matrices is not well-defined in general for this type of stable discretizations, making the segregated algorithms below less efficient.

- Galerkin formulations deteriorate as convection becomes dominant.

This situation has motivated the use of Lagrangian FE approximations with stabilized formulations, that allow to avoid the difficulties related to inf-sup conditions, introduce convection stabilization and make the mass lumping clean. The aim of this section is to analyze how stabilization can affect the design of segregated algorithms and design appropriate stabilized methods that allow segregation keeping unconditional stability.

We can consider a variational multiscale formulation of the problem, by using the framework in [83, 84]. This way, we add a term of the type $\sum_{k\in\mathcal{T}_h}\int_K(F-\partial_t\mathcal{M}(U)-\mathcal{L}(U))\cdot\boldsymbol{\tau}\mathcal{L}^T(V)\mathrm{d}\Omega$ to the LHS of (4.6), where F is the forcing term, $\mathcal{L}$ the spatial differential operator of the problem at hand and $\mathcal{M}$ the continuous mass operator. $\boldsymbol{\tau}$ is the matrix of stabilization parameters. Alternatively, when considering a Galerkin/Least-Squares (GLS) stabilization of the problem, we just replace $\mathcal{L}^T(V)$ by $-\mathcal{L}(V)$ in the definition of the stabilization term. This approach was initially considered in [53] for the full incompressible MHD problem. However, the discrete method was derived assuming extra regularity over b, i.e. $b\in H^1(\Omega)$. As a consequence, the corresponding formulation does not converge to non-smooth solutions $b\notin H^1(\Omega)$.

In order to capture singular solutions of the magnetic problem, we have recently proposed in Chapter 2 a consistent modification of $\mathcal{L}(u,p,b,r)$ in the stabilization term,

replacing

$$\begin{pmatrix} -\nu\Delta u + (w\cdot\nabla)u + \nabla p - (\nabla\times b)\times d \\ \nabla\cdot u \\ \lambda\nabla\times\nabla\times b - \nabla\times(u\times\varrho d) + \nabla r \\ \nabla\cdot b \end{pmatrix} \text{ by } \begin{pmatrix} -\nu\Delta u + (w\cdot\nabla)u + \nabla p - (\nabla\times b)\times d \\ \nabla\cdot u \\ \lambda\nabla\times\nabla\times b - \nabla\times(u\times\varrho d) \\ \nabla r \\ \nabla\cdot b \end{pmatrix},$$

where we have split the induction equation operator into two parts, and defined $\mathcal{M}(u,b) = \mathrm{diag}(1,0,\varrho 1,0,0)$ and $F = (f_u,0,f_b,0,0)$. The key point is the fact that the VMS term is still a residual-based term, since $r = 0$ but now we can consider a diagonal stabilization matrix $\boldsymbol{\tau} := \mathrm{diag}(\tau_u,\tau_p,\tau_{b1},\tau_{b2},\tau_r)$ in such a way that the discrete formulation mimics the correct functional setting, being capable to approximate singular solutions. We refer to Chapters 2 and 3 for a detailed motivation and analysis of this method. After the stabilization, some of the properties of the operators are not true anymore. In particular, $(C_b)^T \neq -C_u$, and $G^T \neq -D$. In any case, the method is unconditionally stable and convergent (see Chapter 3).

Unfortunately, as we can infer from the subsequent sections, unconditionally stable pressure segregation algorithms rely on the fact that $G = -D^T$. This property is used in the stability proof of these methods, in order to treat the pressure terms (see [8–10, 51, 75]). As a result, VMS (or GLS) stabilization of the full MHD (or any other problem with this structure) is not suitable for pressure segregation.

Therefore, in the following, we will design a stabilized formulation that is *block-diagonal*, i.e. the off-diagonal matrices are not modified with stabilization. This way, we keep the *block-skew-symmetric* form of the matrix, allowing to get unconditionally stable segregated algorithms. Obviously, the resulting method must provide the desired stability and keep optimal accuracy. All these requirements can be attained by using a term-by-term orthogonal subscales formulation (see [50]).

However, the use of the term-by-term formulation must be carried out with care. It has been proved in [14, Lemma 2] that orthogonal subscales stabilization of the Maxwell operator deals with weakened stability results. For this reason, we consider two types of terms, those that do require to introduce projections and those that do not. Since we have that

$$\nabla\cdot u = 0, \qquad \nabla\cdot b = 0, \qquad \nabla r = 0,$$

for the continuous solution, we can simply add the stabilization terms:[5]

$$(\tau_p\nabla\cdot u, \nabla\cdot v) + (\tau_r\nabla\cdot b, \nabla\cdot c) + (\tau_{b,2}\nabla r, \nabla s),$$

to the left-hand-side of the Galerkin formulation, without affecting the consistency and convergence of the method. For the rest of the needed stabilization terms, we consider:

$$(\tau_u\pi^{\perp}_{V_{uh}}(\nabla p), \nabla q) + (\tau_u\pi^{\perp}_{V_{uh}}(\bar{u}\cdot\nabla u), \bar{u}\cdot\nabla v) + (\tau_u\pi^{\perp}_{V_{uh}}((\nabla\times b)\times\varrho\bar{b}), (\nabla\times c)\times\varrho\bar{b})$$
$$+ (\tau_{b,1}\pi^{\perp}_{V_{bh}}(\nabla\times(u\times\varrho\bar{b})), \nabla\times(v\times\varrho\bar{b})).$$

Adding these terms, we lose strong consistency, but the orthogonal projection allows to keep optimal rates of convergence towards the exact solution, which is the important

[5] For the first two terms, we can also consider an orthogonal projection stabilization only, without affecting stability. On the contrary, the last term must provide full control on r (see [14]).

point (see [48]); this property is usually called *weak-consistency* and simply means that the *inconsistent* term vanishes as $h \searrow 0$ (see [30]). The first term is the one that provides pressure stability, whereas the other three terms are the convection-stabilization terms. Let us remark that the second term can be switched off for low Reynolds number flows ($\mathrm{Re}_f < 1$), whereas the last two terms can be switched off for low magnetic Reynolds numbers ($\mathrm{Re}_m < 1$). The choice of the stabilization parameters comes from the stability and convergence analysis (see Chapter 3), and have the following expressions within each element k at time step m:

$$\tau_u := (\alpha^m)^{-1}\left(1+\tfrac{\phi^m}{\sqrt{\alpha^m\gamma}}\right)^{-1}, \quad \tau_p := c_5\tfrac{h^2}{\tau_u}, \quad \tau_{b,1} := \gamma^{-1}\left(1+\tfrac{\phi^m}{\sqrt{\alpha^m\gamma}}\right)^{-1},$$
$$\tau_{b,2} := c_6\tfrac{L_0^2}{\lambda}, \qquad\qquad \tau_r := c_7\tfrac{h^2\lambda}{L_0^2},$$

with

$$\alpha^m := c_1\frac{\|u^m\|_{L^\infty(k)}}{h} + c_2\frac{\nu}{h^2}, \qquad \phi^m := c_3\frac{\varrho\|b^m\|_{L^\infty(k)}}{h}, \qquad \gamma := c_4\frac{\lambda}{h^2}.$$

$c_1,\dots,c_5$ are algorithmic constants and L_0 is a length scale of the problem. In order to state the stabilized problem in compact form, we can define the stabilization operators. The stabilization matrix $S := \mathrm{diag}(S_u, S_b, S_p, S_r)$ is composed by:

$$\begin{aligned}
S_u(\bar u,\bar b) :=& (\tau_u\nabla\cdot u, \nabla\cdot(\cdot)) + (\tau_u\pi^\perp_{V_{uh}}(\bar u\cdot\nabla u), \bar u\cdot\nabla(\cdot)) \\
&+ (\tau_{b,1}\pi^\perp_{V_{bh}}(\nabla\times(u\times\varrho\bar b)), \nabla\times((\cdot)\times\varrho\bar b)), \\
S_b(\bar b) :=& (\tau_r\nabla\cdot b, \nabla\cdot(\cdot)) + (\tau_u\pi^\perp_{V_{uh}}((\nabla\times b)\times\varrho\bar b), (\nabla\times(\cdot))\times\varrho\bar b), \\
S_p :=& (\tau_u\pi^\perp_{V_{uh}}(\nabla p), \nabla(\cdot)), \\
S_r :=& (\tau_{b,2}\nabla r, \nabla(\cdot)).
\end{aligned}$$

Let us stress the fact that all these operators are SPD.

Therefore, we can state the stabilized term-by-term formulation as: given $(u^n, b^n) \in V_{uh}\times V_{bh}$, find the solution $(u^{n+1}, p^{n+1}, b^{n+1}, r^{n+1}) \in V_{uh}\times Q_{ph}\times V_{bh}\times Q_{rh}$ of the problem:

$$\frac{1}{\delta t}M_u\delta u^{n+1} + K_u(\bar u^{n+\theta})u^{n+\theta} + C_u(\bar b^{n+\theta})b^{n+\theta} + G_u p^{n+1} = f_u^{n+\theta}, \tag{4.9a}$$
$$D_u u^{n+1} + S_p p^{n+1} = 0, \tag{4.9b}$$
$$\frac{\varrho}{\delta t}M_b\delta b^{n+1} + K_b b^{n+\theta} + C_b(b^{n+\theta})u^{n+\theta} + G_b r^{n+1} = f_b^{n+\theta}, \tag{4.9c}$$
$$D_b b^{n+1} + S_r r^{n+1} = 0, \tag{4.9d}$$

for $n = 0,\dots,N-1$, where we have modified the expression of the Galerkin operators as follows:

$$K_u(\bar u^{n+\theta}) \leftarrow K_u(\bar u^{n+\theta}) + S_u(\bar u^{n+\theta}, \bar b^{n+\theta}), \qquad K_b \leftarrow K_b + S_b(\bar b^{n+\theta}).$$

The stability of this algorithm comes from the following properties.

Theorem 4.3. *The discrete operators for the term-by-term stabilization technique satisfy the following properties*

1. *K_u satisfies $|v|_{K_u} \geq \alpha_u \|v\|_{V_u}$ for any $v \in V_{uh}$, with $\alpha_u > 0$,*
2. *For any $c \in V_{bh}$, there exists $\alpha_b > 0$ such that $\alpha_b \|c\|_{V_b} \leq \sup_{s \in Q_{rh}} \frac{\langle D_b c, s \rangle}{\|s\|_{Q_r}} + |c|_{K_b}$,*
3. *For any $q \in Q_{ph}$, there exists $\beta_u > 0$ such that $\beta_u \|q\|_{Q_p} \leq \sup_{v \in V_{uh}} \frac{\langle G_u(q), v \rangle}{\|v\|_{V_u}} + |q|_{S_p}$,*
4. *For any $s \in Q_{rh}$, there exists $\beta_b > 0$ such that $\beta_b \|s\|_{Q_r} \leq |s|_{S_r}$,*
5. *$G_u = -D_u^T$, $G_b = -D_b^T$ and $C_u(d) = -C_b(d)^T$, for any $d \in V_{bh}$,*

for FE spaces V_{uh}, V_{bh}, Q_{ph} and Q_{rh} constructed by Lagrangian FEs.

Proof. The first property comes from the fact that coercivity is inherited by any conforming FE space as well as the definition of the SPD operator S_u and the fact that we consider the skew-symmetric form of the convective operator.

The second result relies on the continuous inf-sup condition (4.4) for the continuous magnetic field. Since we assume a conforming FE formulation, $(c, 0) \in V_b \times Q_r$ for any $c \in V_{bh}$. We know from (4.4) that there exists $(\bar{c}, \bar{s}) \in V_b \times Q_r$ with unit norm, i.e. $\|\bar{c}\|_{V_b} + \|\bar{s}\|_{Q_r} = 1$, such that $c_{\text{Mxw}}(c, 0, \bar{c}, \bar{s}) \gtrsim \|c\|_{V_b}$. Now, we have:

$$c_{\text{Mxw}}(c, 0, \bar{c}, \bar{s}) = c_{\text{Mxw}}(c, 0, 0, \varepsilon_h(\bar{s})) + c_{\text{Mxw}}(c, 0, \bar{c}, \pi^{\text{SZ}}_{Q_{rh}}(\bar{s})),$$

where $\varepsilon_h(t) := t - \pi^{\text{SZ}}_{Q_{rh}}(t)$ and $\pi^{\text{SZ}}_{Q_{rh}}(v)$ is a continuous FE interpolant in $H^1(\Omega)$ of a function t (scalar or vector valued) with optimal interpolation properties that preserves null traces, e.g. the Scott-Zhang interpolant [33]. Now, we bound the first term as follows:

$$c_{\text{Mxw}}(c, 0, 0, \varepsilon_h(\bar{s})) = (c, \nabla \varepsilon_h(\bar{s})) \leq -\|\nabla \cdot c_h\| \|\varepsilon_h(\bar{s})\| \lesssim \|\tau_r^{\frac{1}{2}} \nabla \cdot c\|,$$

where we have used the typical error interpolation estimate (see [33]) and Poincaré inequality in order to get $\|\varepsilon_h(\bar{s})\|_{L^2(\Omega)} \lesssim h \|\bar{s}\|_{H^1(\Omega)} \lesssim \frac{h \lambda^{\frac{1}{2}}}{L_0} \|\bar{s}\|_{Q_r}$, and the definition of τ_r. On the other hand,

$$c_{\text{Mxw}}(c, 0, \bar{c}, \pi^{\text{SZ}}_{Q_{rh}}(\bar{s})) = (\lambda \nabla \times c, \nabla \times \bar{c}) + (\nabla \cdot c, \pi^{\text{SZ}}_{Q_{rh}}(\bar{s})) \leq \lambda^{\frac{1}{2}} \|\nabla \times c\| + \|\nabla \cdot c\|_{Q'_{rh}},$$

where we have used the $H^1(\Omega)$ continuity of the Scott-Zhang interpolant and the dual norm $\|f\|_{Q'_{rh}} := \sup_{s \in Q_{rh}} \frac{\langle f, s \rangle}{\|s\|_{Q_r}}$. Combining these results, we easily prove the second result.

With regard to the Lagrange multipliers, S_r is a PD operator that straightforwardly provides $H^1(\Omega)$ stability, and so, r stability does not rely on an inf-sup condition anymore. The stability of the fluid pressure is obtained from the continuous inf-sup condition (4.3). For any discrete pressure $p \in Q_{ph} \subset Q_p$, property 3 in Assumption 4.1 implies that there exists a $v_p \in V_u$ such that

$$\begin{aligned} \beta \|p\|_{Q_p} \|v_p\|_{V_u} &\leq (\nabla \cdot v_p, p) = (\nabla \cdot \varepsilon_h(v_p), p) + (\nabla \cdot \pi^{\text{SZ}}_{V_{uh}}(v_p), p) \\ &\leq \|\tau_u^{\frac{1}{2}} \nabla p\| \|v_p\|_{V_u} + (\nabla \cdot \pi^{\text{SZ}}_{V_{uh}}(v_p), p), \end{aligned}$$

where we have used the standard interpolation property stated above, as well as the definition of τ_u. Using the H^1-continuity of the Scott-Zhang projector, we easily prove the weakened discrete inf-sup condition.

Finally, the last property is inherited from the continuous problem, as commented above for the Galerkin discretization, since those operators have not been modified by the term-by-term stabilization. □

Remark 4.2. *The analysis of the term-by-term stabilization relies on* weakened *versions of the discrete inf-sup conditions that are true for a much larger set of FE spaces, e.g. equal order Lagrangian FE spaces. It is different from VMS formulations, that make the resulting method coercive. However, due to the additional coupling between unknowns that come from the stabilization terms, the analysis of the VMS method results to be more involved (see Theorem 2 in Chapter 2 for the incompressible MHD system and a VMS stabilization).*

Remark 4.3. *By the definition of V_{uh} and V_{bh}, any function belonging to these spaces satisfy the boundary conditions for the velocity and magnetic field respectively. However, we cannot consider the orthogonal projections onto these spaces because the consistency error would not be optimal. We have to use the orthogonal projections onto the spaces without any boundary condition. It has been proved in [52] that, even though it reduces the stability of the algorithm, the orthogonal subscales method is still stable.*

Remark 4.4. *In order to implement the projected terms, we consider the following approximation*

$$\pi^{\perp}_{X_h}(y^{n+1}) = y^{n+1} - \pi_{X_h}(y^{n+1}) = y^{n+1} - M^{-1}_{X_h} y^{n+1}$$

where M_{X_h} is the consistent mass matrix for an arbitrary FE space X_h. This way, the projection is computed once per time step and the corresponding CPU cost has been observed to be negligible, i.e. less than 1% of the solver time. Furthermore, it slightly improves stability (see [25]). Additionally, $M^{-1}_{X_h}$ can be replaced by its lumped version, reducing even more the computational cost of the projection.

4.5 Operator splitting techniques

In this section, we consider splitting operator techniques at the fully discrete level (see e.g. [9, 47]).

4.5.1 Level 1: pressure segregation

Let us group the unknowns into the vectorial field variables $y := (u, b)$ and the Lagrange multipliers or "pressures" $z := (p, r)$, in order to carry out the pressure segregation with compact notation. The first splitting technique that we apply over the full MHD system consists of the segregation of the y and z computation. In order to do that, we consider an extension of the classical pressure segregation technique, also called pressure correction or fractional step method, that is widely used in the numerical approximation of the incompressible Navier-Stokes equations (see [9, 42, 75, 116]). The MHD stabilized system (4.9) (or (4.6) when considering the non-stabilized formulation) can be rewritten as:

$$\frac{1}{\delta t} M \delta y^{n+1} + K(\bar{y}^{n+\theta}) y^{n+\theta} + G z^{n+1} = f^{n+\theta}, \qquad D y^{n+1} + S z^{n+1} = 0, \tag{4.10}$$

where we have used the notation

$$K(\bar{y}) = \begin{bmatrix} K_u(\bar{u}) & C_u(\bar{b}) \\ C_b(\bar{b}) & K_b \end{bmatrix},\ M = \begin{bmatrix} M_u & 0 \\ 0 & \varrho M_b \end{bmatrix},\ G = \begin{bmatrix} G_u & 0 \\ 0 & G_b \end{bmatrix},$$
$$D = \begin{bmatrix} D_u & 0 \\ 0 & D_b \end{bmatrix},\ S = \begin{bmatrix} S_p & 0 \\ 0 & S_r \end{bmatrix},\ f^{n+\theta} = \begin{bmatrix} f_u^{n+\theta} \\ f_b^{n+\theta} \end{bmatrix}. \tag{4.11}$$

We treat the Galerkin and stabilized formulations in a unified way. The final expression of the operators will depend on this choice. In particular, the S-matrices are null for the Galerkin FE approximation. Since $G = -D^T$, we can consider pressure segregation techniques, and obtain unconditionally stable segregated algorithms. We consider the equivalent statement of the problem, by introducing the intermediate unknown $\tilde{y}$:

$$\frac{M}{\delta t}(\tilde{y}^{n+1} - y^n) + K(\bar{y}^{n+\theta})y^{n+\theta} + Gz^{*,n+1} = f^{n+\theta}, \tag{4.12}$$
$$\frac{M}{\delta t}(y^{n+1} - \tilde{y}^{n+1}) + G(z^{n+1} - z^{*,n+1}) = 0, \quad Dy^{n+1} + Sz^{n+1} = 0,$$

where $z^{*,n+1}$ is an approximation of z^{n+1}. Now, we introduce the splitting error, by replacing $y^{n+\theta}$ by $\tilde{y}^{n+\theta}$ in (4.12), where $\tilde{y}^{n+\theta} := \theta\tilde{y}^{n+1} + (1-\theta)y^n$. This approximation perturbs the monolithic problem with an error term $e_y = \theta\delta t K(\bar{y}^{n+\theta})M^{-1}G(z^{n+1} - z^{*,n+1})$ in the momentum equation (4.12).[6] Furthermore, applying $DM^{-1}(\cdot)$ over the second equation, and invoking the mass conservation equation, we obtain:

$$\frac{M}{\delta t}(\tilde{y}^{n+1} - y^n) + K(\bar{y}^{n+\theta})\tilde{y}^{n+\theta} + Gz^{*,n+1} = f^{n+\theta}, \tag{4.13a}$$
$$D\tilde{y}^{n+1} + Sz^{n+1} + \delta t L_D(z^{n+1} - z^{*,n+1}) = 0, \tag{4.13b}$$
$$\frac{M}{\delta t}(y^{n+1} - \tilde{y}^{n+1}) + G(z^{n+1} - z^{*,n+1}) = 0, \tag{4.13c}$$

where $L_D := -DM^{-1}G$ is a discrete Laplacian matrix. From the definition of the matrices above, we easily see that

$$L_D = \begin{pmatrix} L_{Du} & 0 \\ 0 & L_{Db} \end{pmatrix}, \quad \text{where } L_{Du} := D_u M_u^{-1} G_u, \quad L_{Db} := \frac{1}{\varrho} D_b M_b^{-1} G_b. \tag{4.14}$$

So, the computation of p and r in (4.13b) is also decoupled.[7] In practice, $DM^{-1}G$ is usually replaced by more standard (and cheaper) discretizations of the Laplacian with homogeneous boundary conditions, i.e. $L_D \leftarrow \mathrm{diag}(L_p, \varrho^{-1}L_r)$ where $L_p p = (\nabla p, \nabla(\cdot))$ (idem for r). This approximation introduces an additional perturbation (with respect to the monolithic system) in the mass conservation equation $e_z := \delta t E_L(z^{n+1} - z^{*,n+1})$ which does not spoil the accuracy of the algorithm and improves stability; $E_L := L_D - DM^{-1}G$ is a semi-positive-definite matrix.

[6] Therefore, if $z^{n+1} - z^{*,n+1} \sim \mathcal{O}(\delta t^k)$, the error introduced by the splitting is of order $\mathcal{O}(\delta t^{k+1})$.

[7] We can infer that $z \cdot DM^{-1}Gz = \|\pi_{V_{yh}}(z)\|^2$ from the definition of the operators, where $V_{yh} \equiv V_{uh} \times V_{bh}$. This fact, together with the *stabilized* inf-sup condition for p and the coervity for r in Theorem 4.3 prove the well-posedness of the term-by-term stabilized problem. For the Galerkin formulation, it simply comes from the inf-sup conditions in 4.1.

Theorem 4.4. *The sequence of solutions of system (4.13) for $m = 0, \dots, N-1$ satisfies the energy inequality:*

$$\|u^{m+1}\|^2 + \varrho\|b^{m+1}\|^2 + \sum_{n=0}^{m} \delta t(\alpha_u \|\tilde{u}^{n+\theta}\|^2_{V_u} + \alpha_b \|\tilde{b}^{n+\theta}\|^2_{V_b}) + \sum_{n=0}^{m} \delta t(|p^{n+1}|^2_{S_p} + |r^{n+1}|^2_{S_r}) \leq \|u_0\|^2 + \varrho\|b_0\|^2 + \sum_{n=0}^{m} \delta t(\frac{1}{\alpha_u}\|f_u^{n+\theta}\|_{V_u'} + \frac{1}{\alpha_b}\|f_b^{n+\theta}\|_{V_b'}),$$

for $z^{*,n+1} = (p^n, 0)$, with $\theta = \frac{1}{2}$. Therefore, system (4.13) is unconditionally stable both for the Galerkin and term-by-term stabilization, and either $L_D := -DM^{-1}G$ or $L_D := \mathrm{diag}(L_p, \varrho^{-1}L_r)$.

Proof. Let us prove the result for the Crank-Nicolson algorithm; the analysis for the Backward-Euler time integration is simpler. The proof of this result is similar to the one for the incompressible Navier-Stokes equations in [47]; we just sketch the proof, introducing the slight differences between the two analyses. One particularity is the fact that, since $r = 0$ at the continuous level, we have considered $z^{*,n+1} = (p^n, 0)$ instead of $z^{*,n+1} = z^n$. Stability is obtained after testing (4.13a) against $2\delta t\tilde{y}^{n+\frac{1}{2}}$ and (4.13c) against $\delta t(y^{n+1} + \tilde{y}^{n+1})$. Adding up the resulting equations, we easily get:

$$|y^{n+1}|^2_M - |y^n|^2_M + \delta t|\tilde{y}^{n+\frac{1}{2}}|^2_K + \delta t\tilde{y}^{n+1} \cdot Gz^{n+1} + \delta t y^n \cdot Gz^{*,n+1} + \delta t y^{n+1} \cdot G(z^{n+1} - z^{*,n+1}) \lesssim \tilde{y}^{n+\frac{1}{2}} \cdot f^{n+\frac{1}{2}}. \tag{4.15}$$

Using Eq. (4.13b), and the fact that $D = -G^T$ we obtain:

$$\delta t\tilde{y}^{n+1} \cdot Gz^{n+1} = \delta t^2 z^{n+1} L_D(z^{n+1} - z^{*,n+1}) + \delta t|z|^2_S. \tag{4.16}$$

On the other hand, (4.13c) can be rewritten as:

$$y^{n+1} = \tilde{y}^{n+1} - \delta t M^{-1} G(z^{n+1} - z^{*,n+1}), \tag{4.17}$$

which used for steps $n+1$ and n easily leads to:[8]

$$\delta t y^{n+1} \cdot G(z^{n+1} - z^{*,n+1}) + \delta t y^n \cdot Gz^n = \delta t^2 |z^{n+1} - z^{*,n+1}|^2_{E_L} + \delta_t^2 z^n \cdot E_L(z^n - z^{*,n}).$$

Combining the last three equations, we obtain:

$$\begin{aligned} &\delta t\tilde{y}^{n+1} \cdot Gz^{n+1} + \delta t y^n \cdot Gz^{*,n+1} + \delta t y^{n+1} \cdot G(z^{n+1} - z^{*,n+1}) = \\ &\delta t|z|^2_S + \delta t^2 z^{n+1} \cdot L(z^{n+1} - z^{*,n+1}) + \delta t^2 |z^{n+1} - z^{*,n+1}|^2_{E_L} + \delta t^2 z^n \cdot E_L(z^n - z^{*,n}). \end{aligned} \tag{4.18}$$

After adding up for all the time step values (considering the special treatment for the first step) we easily check that the sum of the last three terms is a positive value. Then, we have

$$\sup_{m=0,\dots,N-1} |y^m|^2_M + \sum_{n=0}^{N-1} \delta t(|\tilde{y}^{n+1}|^2_K + |z^{n+1}|^2_S) \lesssim |y_0|_M + \sum_{n=0}^{N-1} \delta t\tilde{y}^{n+\frac{1}{2}} \cdot f^{n+\frac{1}{2}}. \tag{4.19}$$

[8] In order to treat the case $n = 0$ in the previous equation we have some options. The most straightforward is to consider $z^0 = (0, 0)$, but it introduces an $\mathcal{O}(\delta t)$ error at the first time step value. Alternatively, we can consider a monolithic solver for the first time step. In this case, the previous equation only applies for $n = 1, \dots, N-1$ and $y^1 \cdot Gz^1 = |p^1|_{S_p} + |r^1|_{S_r}$.

In order to have full control over $\|\tilde{b}^{n+\frac{1}{2}}\|_{V_b}$ we use the fact that $D_b\tilde{b}^{n+1} = -S_r r^{n+1} - \frac{\delta t}{\varrho} L_r r^{n+1} \in Q'_{rh}$. Property 2 in Theorem 4.3 leads to:

$$|\tilde{b}^{n+1}|_{K_b} + |r^{n+1}|_{S_r} \gtrsim \alpha_b \|b\|_{V_b},$$

where we have used $\frac{\delta t}{\varrho} \lesssim \frac{L_0^2}{\lambda}$, for a dimensionless constant. Using the expressions of y and z, combining the last two results and Poincare-Friedrich's inequality, we prove the theorem. □

Remark 4.5. *The VMS and GLS stabilized formulations perturb the G and D operators in such a way that $G \neq -D^T$. As a result, $DM^{-1}G$ is not positive definite. Further, taking $L_D = \mathrm{diag}(L_p, \varrho^{-1}L_r)$ is not enough to solve the problem, since the matrix $E_L := L_D - DM^{-1}G$ is indefinite in general. E_L must be a semi-positive-definite matrix in order to prove that the last two terms in (4.18) are larger or equal to zero, after adding up for all time step values. As far as we know, a standard pressure segregation algorithm cannot be applied to this type of stabilized formulations without affecting the stability properties of the method.*

In Algorithm 4.1 we have written the second order in time method resulting of a Crank-Nicholson time integration scheme and $z^{*,n+1} = (p^n, 0)$, in its semi-implicit version; the implicit formulation is obtained by performing nonlinear iterations over this algorithm. The first two equations (for u and b) are coupled, but those for p and r are decoupled. Summarizing, the splitting has converted a monolithic indefinite (double saddle-point) system into three definite (parabolic) problems. The problem for $(\tilde{u}, \tilde{b})$ is a convection-diffusion-reaction system, whereas two Dirichlet (Laplacian) problems must be solved for p and r separately. Further, in order to obtain the end-of-step values (u, b) we must solve two different systems with a mass matrix, which can be lumped. These systems are trivial and have a negligible CPU cost. Unfortunately, *lumping* is not obvious for the inf-sup stable FE interpolations needed when using a crude Galerkin approximation. This is another reason to favour the term-by-term stabilized formulation (4.9). It is important to note also that because $r = 0$ at the continuous level, its numerical value is very small (except for very coarse meshes). Therefore, it is possible to freeze the computation of r^{n+1} for several time steps without affecting the accuracy on the solution of b, thus reducing the total computational cost of the algorithm. See Section 4.6.2 for a numerical example of this fact.

4.5.2 Level 2: $u - b$ uncoupling

Step 1 in Algorithm 4.1 still requires to solve a convection-diffusion-reaction problem with system matrix $\frac{1}{\delta t}M + K$, that couples the u and b computations. However, we can observe that the K matrix that couples the u and b computations has a *block-skew-symmetric* form. Therefore, we can consider an additional splitting over this matrix, keeping unconditional stability, as we have done for the pressure. We start with a simple splitting, which is at most first order accurate in time. Then, we will show how to get more accurate methods. Using a first order product formula over the discrete evolutive

Algorithm 4.1: Second order segregated Galerkin formulation

Given the previous time step values $(u^n, u^{n-1}, b^n, b^{n-1})$, the linearized problem around $(\bar{u}, \bar{b})$ reads as: compute the solution $(u^{n+1}, p^{n+1}, b^{n+1}, r^{n+1}) \in V_{uh} \times Q_{ph} \times V_{bh} \times Q_{rh}$ as follows:

- Step 1: Compute $(\tilde{u}^{n+1}, \tilde{b}^{n+1}) \in V_{uh} \times V_{bh}$ such that

$$\begin{cases} \frac{1}{\delta t} M_u(\tilde{u}^{n+1} - u^n) + K_u(\bar{u}^{n+\frac{1}{2}})\tilde{u}^{n+\frac{1}{2}} + G_u p^n + C_u(\bar{b}^{n+\frac{1}{2}})\tilde{b}^{n+\frac{1}{2}} = f_u^{n+\frac{1}{2}}, \\ \frac{\varrho}{\delta t} M_b(\tilde{b}^{n+1} - b^n) + K_b \tilde{b}^{n+\frac{1}{2}} + C_b(\bar{b}^{n+\frac{1}{2}})\tilde{u}^{n+\frac{1}{2}} = f_b^{n+\frac{1}{2}}, \end{cases}$$

  where $\bar{u}^{n+\frac{1}{2}} = \frac{3}{2}u^n - \frac{1}{2}u^{n-1}$ and $\bar{b}^{n+\frac{1}{2}} = \frac{3}{2}b^n - \frac{1}{2}b^{n-1}$.

- Step 2: Compute $p^{n+1} \in Q_{ph}$ and $r^{n+1} \in Q_{rh}$ such that

$$\delta t L_p p^{n+1} = D_u \tilde{u}^{n+1} + \delta t L_p p^n, \qquad \frac{\delta t}{\varrho} L_r r^{n+1} = D_b \tilde{b}^{n+1}.$$

- Step 3: Compute $u^{n+1} \in V_{uh}$ and $b^{n+1} \in V_{bh}$ such that

$$\frac{1}{\delta t} M_u(u^{n+1} - \tilde{u}^{n+1}) + G_u(p^{n+1} - p^n) = 0, \qquad \frac{\varrho}{\delta t} M_b(b^{n+1} - \tilde{b}^{n+1}) + G_b r^{n+1} = 0$$

problem (see [3, 44]), Eq. (4.13a) can be re-written as:

$$\frac{1}{\delta t} M_u \tilde{u}^{n+1} + K_u(\bar{u}^{n+\theta})\tilde{u}^{n+\theta} + C_u(\bar{b}^{n+\theta})\tilde{b}^{n+\theta} = f_u^{n+\theta} + \frac{1}{\delta t} M u^n - G_u p^{*,n+1}, \tag{4.20a}$$

$$\frac{\varrho}{\delta t} M_b \tilde{b}^{n+1} + K_b \tilde{b}^{n+\theta} + C_b(\bar{b}^{n+\theta})\tilde{u}^{n+\theta} = f_b^{n+\theta} + \frac{\varrho}{\delta t} M b^n. \tag{4.20b}$$

At this point, let us introduce the auxiliary variable $\hat{u}$. We can re-write eq. (4.20a) as:

$$\frac{1}{\delta t} M \hat{u}^{n+1} + C_u(\bar{b}^{n+\theta})\tilde{b}^{n+\theta} = \frac{1}{\delta t} M u^n, \tag{4.21a}$$

$$\frac{1}{\delta t} M(\tilde{u}^{n+1} - \hat{u}^{n+1}) + K_u(\bar{u}^{n+\theta})\tilde{u}^{n+\theta} = f_u^{n+\theta} - G_u p^{*,n+1}. \tag{4.21b}$$

Now, we introduce the splitting error replacing $\tilde{u}^{n+\theta}$ in (4.20b) by $\hat{u}^{n+\theta}$:

$$\frac{\varrho}{\delta t} M \tilde{b}^{n+1} + K_b \tilde{b}^{n+\theta} + C_b(\bar{b}^{n+\theta})\hat{u}^{n+\theta} = f_b^{n+\theta} + \frac{\varrho}{\delta t} M b^n, \tag{4.22}$$

where $\hat{u}^{n+\theta} := \theta \hat{u}^{n+1} + (1-\theta)u^n$. This way, the splitting error introduced in the induction equation is

$$e_b = C_b(\tilde{u}^{n+\theta} - \hat{u}^{n+\theta}) = C_b(\tilde{u}^{n+\theta} - u^n + \delta t C_u(\bar{b}^{n+\theta})\tilde{b}^{n+\theta}) \sim \mathcal{O}(\delta t),$$

and so, the resulting method is at most first order accurate in time. This way, we can compute $(\hat{u}, \tilde{b})$ coupled by the system (4.21a)-(4.22), whereas $\tilde{u}$ can be obtained in a

subsequent step from (4.21b). Alternatively, we can eliminate $\hat{u}$ invoking (4.21a) in (4.22), getting:

$$\frac{\varrho}{\delta t}M\tilde{b}^{n+1} + (K_b - \delta t C_b(\bar{b}^{n+\theta})M_u^{-1}C_b(\bar{b}^{n+\theta})^T)\tilde{b}^{n+\theta} = f_b^{n+\theta} + \frac{\varrho}{\delta t}Mb^n - C_b(\bar{b}^{n+\theta})u^n. \tag{4.23}$$

Replacing (4.20) by (4.23)-(4.21b), we obtain Algorithm 4.2. We have considered Backward-Euler time integration, i.e. $\theta = 1$, since the method is at most first order accurate in time and the numerical dissipation introduced by this time integrator is needed in order to prove stability. In any case, we always use a second order segregation for the pressure, as it clearly improves the numerical results without additional cost. Alternatively, we can obtain a "dual" $u - b$ splitting by interchanging the roles of u and b above. We omit the details of the obtention of the method, which is stated in Algorithm 4.3.

Algorithm 4.2: Segregated and perturbed Galerkin FE method, first order accurate in time (perturbation over u)

Given the previous time step values $(u^n, u^{n-1}, b^n, b^{n-1})$, the linearized problem around $(\bar{u}, \bar{b})$ reads as: compute the solution $(u^{n+1}, p^{n+1}, b^{n+1}, r^{n+1}) \in V_{uh} \times Q_{ph} \times V_{bh} \times Q_{rh}$ as follows:

- Step 1: Compute $\tilde{b}^{n+1} \in V_{hb}$ such that

$$\frac{1}{\delta t}M_b(\tilde{b}^{n+1} - b^n) + K_b b^{n+1} + \delta t C_b(b^n)M_{\hat{u}}^{-1}C_b^T(b^n)\tilde{b}^{n+1} = f_b^{n+1} - C_b(b^n)u^n.$$

- Step 2: Compute $\tilde{u}^{n+1} \in V_{uh}$ and $r^{n+1} \in Q_{rh}$ that satisfy the uncoupled problems

$$\frac{1}{\delta t}M_u(\tilde{u}^{n+1} - u^n) + K_u(u^n)\tilde{u}^{n+1} + G_u p^n + C_u(b^n)\tilde{b}^{n+1} = f_u^{n+1},$$
$$\frac{\delta t}{\varrho}L_r r^{n+1} = D_b\tilde{b}^{n+1}.$$

- Step 3: Compute $p^{n+1} \in Q_{uh}$ and $b^{n+1} \in V_{bh}$ that satisfy the uncoupled problems

$$\delta t L_p p^{n+1} = D_u\tilde{u}^{n+1} + \delta t L_p p^n, \quad \text{and} \quad \frac{\varrho}{\delta t}M_b(b^{n+1} - \tilde{b}^{n+1}) + G_b r^{n+1} = 0.$$

- Step 4: Compute $u^{n+1} \in V_{uh}$ that satisfies $\frac{1}{\delta t}M_u(u^{n+1} - \tilde{u}^{n+1}) + G_u(p^{n+1} - p^*) = 0$.

The perturbed linearizations introduced above uncouple the computation of the two sub-problems. However, at the discrete level, the obtention of $\hat{u}^{n+1} = u^n - \delta t M_u^{-1}C_u(b^n, b^{n+1})$ (respectively, $\hat{b}$ for the "dual" formulation), requires to solve an additional linear system with the mass matrix M_u. In order to circumvent this extra cost, we can instead consider a different FE space for $\hat{u}$, denoted as $\hat{V}_{uh}$ such that $C_u(d)c \in \hat{V}_{uh}$ for any $c,\ d \in V_{bh}$, e.g. $\hat{V}_{uh} \equiv V_{uh} \oplus C_u(V_{bh}, V_{bh})$, where $C_u(V_{bh}, V_{bh})$ is the space of functions spanned by $C_u(d, c)$, for any $c,\ d \in V_{bh}$. This space is composed by discontinuous functions, but it is not a problem, since $\hat{u}$ is only required to be in $L^2(\Omega)$ in

order for the method to have sense. This way, the previous expression of $\hat{u}$ now reads as $u^n - \delta t C_u(b^n, b^{n+1})$[9] and the perturbation term is simply:

$$\left(C_b(d)M_{\hat{u}}^{-1}C_b^T(d)b, c\right) = \left((\nabla \times b) \times d, (\nabla \times c) \times d\right).$$

In fact, using this trick we recover (for the crude Galerkin approximation) the scheme in [3] for a more rudimentary formulation of the MHD problem, in terms of (u, p, b), that could not enforce b to be solenoidal and could not converge to singular solutions ($H^1(\Omega)$ regularity over b was assumed). Therein, the authors motivated the splitting between the fluid problem (u, p) and the magnetic problem for b using a first order product formula for the approximation of semi-groups in the sense of [44]; no pressure segregation was studied, since the pressure is not a historical variable.

Analogously for Algorithm 4.3, using the analogous trick for $\hat{b}$, i.e. $\hat{V}_{bh} \equiv V_{bh} \oplus C_b(V_{bh}, V_{uh})$, the perturbation term can be re-written as:

$$\left(C_u(d)M_{\hat{b}}^{-1}C_u^T(d)u, v\right) = \left(\nabla \times (u \times d), \nabla \times (v \times d)\right).$$

This is the way we have implemented the perturbation terms in the numerical experiments section for Algorithms 4.2 and 4.3.

Remark 4.6. *Instead of considering the pressure segregation first and the $u - b$ segregation later, we could perform the splitting in reverse order. In any case, the resulting method is identical.*

The resulting method is first order accurate in time. However, we can design second order algorithms (in time) using the velocity-correction splitting in [10] for the $u - b$ system. E.g. a second order splitting of the $u - b$ problem would read as:

$$\frac{1}{\delta t}M\hat{u}^{n+1} + C_u(\bar{b}^{n+\theta})\tilde{b}^{n+\theta} = f_u^{n+\theta} + \frac{1}{\delta t}Mu^n - K_u(\bar{u}^{n+\theta})u^{*,n+\theta} - G_u p^{*,n+1}, \tag{4.24a}$$

$$\frac{\varrho}{\delta t}M\tilde{b}^{n+1} + K_b\tilde{b}^{n+\theta} + C_b(\bar{b}^{n+\theta})\hat{u}^{n+\theta} = f_b^{n+\theta} + \frac{\varrho}{\delta t}Mb^n, \tag{4.24b}$$

$$\frac{1}{\delta t}M(\tilde{u}^{n+1} - \hat{u}^{n+1}) + K_u(\bar{u}^{n+\theta})(\tilde{u}^{n+\theta} - u^{*,n+\theta}) = 0. \tag{4.24c}$$

This splitting error is second order in time for a first order extrapolation $u^{*,n+\theta}$, e.g. taking the value from the previous time step u^n (see [47]). Since the K_u operator is a second order operator, and we cannot redefine the V_{uh} space as above, second order methods would require to work with $C_b M_u^{-1} C_b^T$, slightly complicating the implementation issues and CPU time. Analogously, we can consider the "dual" version of (4.23) performing the splitting with respect to b instead of u.

Corollary 4.2. *Algorithms 4.2 and 4.3 for $m = 0, \ldots, N-1$ satisfy the energy inequality:*

$$\|u^{m+1}\|^2 + \varrho\|b^{m+1}\|^2 + \sum_{n=0}^{m} \delta t(\alpha_u\|\tilde{u}^{n+1}\|_{V_u}^2 + \alpha_b\|\tilde{b}^{n+1}\|_{V_b}^2)$$
$$+ \sum_{n=0}^{m} \delta t(|p^{n+1}|_{S_p}^2 + |r^{n+1}|_{S_r}^2) \leq \|u_0\|^2 + \varrho\|b_0\|^2 + \sum_{n=0}^{m} \delta t(\frac{1}{\alpha_u}\|f_u^{n+1}\|_{V_u'} + \frac{1}{\alpha_b}\|f_b^{n+1}\|_{V_b'}),$$

[9] The introduction of this space is only used at the theoretical level, in order to get rid of the additional linear system. It is never explicitly built in practice.

Algorithm 4.3: Segregated and perturbed Galerkin FE method, first order accurate in time (perturbation over b)

Given the previous time step values $(u^n, u^{n-1}, b^n, b^{n-1})$, the linearized problem around $(\bar{u}, \bar{b})$ reads as: compute the solution $(u^{n+1}, p^{n+1}, b^{n+1}, r^{n+1}) \in V_{uh} \times Q_{ph} \times V_{bh} \times Q_{rh}$ as follows:

- Step 1: Compute $\tilde{u}^{n+1} \in V_{uh}$ such that

$$\frac{1}{\delta t} M_u(\tilde{u}^{n+1} - u^n) + K_u(u^n)\tilde{u}^{n+1} + \delta t C_u(b^n) M_{\hat{b}}^{-1} C_u^T(b^n)\tilde{u}^{n+1} + G_u p^{n+1} = f_u^{n+1} - C_u(b^n)b^n.$$

- Step 2: Compute $\tilde{b}^{n+1} \in V_{bh}$ and $p^{n+1} \in Q_{ph}$ that satisfy the uncoupled problems

$$\frac{\varrho}{\delta t} M_b(\tilde{b}^{n+1} - b^n) + K_b \tilde{b}^{n+1} + C_b(b^n)\tilde{u}^{n+1} = f_b^{n+1}, \text{ and } \delta t L_p p^{n+1} = D_u \tilde{u}^{n+1} + \delta t L_p p^n.$$

- Step 3: Compute $r^{n+1} \in Q_{rh}$ and $u^{n+1} \in V_{uh}$ that satisfy the uncoupled problems

$$\frac{\delta t}{\varrho} L_r r^{n+1} = D_b \tilde{b}^{n+1}, \quad \text{and} \quad \frac{1}{\delta t} M_u(u^{n+1} - \tilde{u}^{n+1}) + G_u(p^{n+1} - p^*) = 0.$$

- Step 4: Compute $b^{n+1} \in V_{bh}$ that satisfies $\frac{\varrho}{\delta t} M_b(b^{n+1} - \tilde{b}^{n+1}) + G_b r^{n+1} = 0$.

for $z^{*,n+1} = (p^n, 0)$. Therefore, these algorithms are unconditionally stable both for the Galerkin and term-by-term stabilization, and either $L_D := -DM^{-1}G$ or $L_D := \mathrm{diag}(L_p, \varrho^{-1}L_r)$.

Proof. The treatment of the pressure terms due to pressure segregation is straightforward from Theorem 4.4 and the analysis in [47] for the Bacward-Euler time integrator. With respect to Theorem 4.4, we have perturbed the K term and the right-hand side f only. We can proceed as before, in order to get:

$$\begin{aligned} &\sup_{m=0,\dots,N-1} |y^m|_M^2 + \sum_{n=0}^{N-1} \delta t (|\tilde{y}^{n+1}|_K^2 + |z^{n+1}|_S^2) \\ &\qquad \lesssim |y_0|_M + \sum_{n=0}^{N-1} \delta t (\tilde{y}^{n+\frac{1}{2}} \cdot f^{n+1} - \tilde{b}^{n+1} \cdot C(\bar{b})\hat{u}^{n+1}). \end{aligned}$$

The split matrix K leads to:

$$|\tilde{y}^{n+1}|_K^2 = \nu \|\nabla \tilde{u}^{n+1}\|^2 + \lambda \|\nabla \times \tilde{b}^{n+1}\|^2 + \tilde{u}^{n+1} \cdot C_u(\bar{b})\tilde{b}^{n+1}.$$

Combining this term with the right-hand-side, we obtain:

$$\begin{aligned} &\tilde{u}^{n+1} \cdot C_u(\bar{b})\tilde{b}^{n+1} + \tilde{b}^{n+1} \cdot C_b(\bar{b})\hat{u}^{n+1} = \tilde{u}^{n+1} \cdot C_u(\bar{b})\tilde{b}^{n+1} + \hat{u}^{n+1} \cdot C_b(\bar{b})^T \tilde{b}^{n+1} \\ &= (\tilde{u}^{n+1} - u^n) \cdot C_u(\bar{b})\tilde{b}^{n+1} + C_b(\bar{b})^T \tilde{b}^{n+1} \cdot \delta t M_{\hat{u}}^{-1} C_b(\bar{b})^T \tilde{b}^{n+1} \\ &\geq \frac{\delta t}{2} |M_{\hat{u}}^{-1} C_b(\bar{b})^T b^{n+1}|_{M_{\hat{u}}} - \frac{1}{2\delta t} \beta_1(M_u; \tilde{u}^{n+1}, u^n), \end{aligned} \tag{4.25}$$

where we have used the fact that $V_{uh} \subset \hat{V}_{uh}$. Combined with the numerical dissipation of the Backward-Euler time integration and using Property 2 in Theorem 4.3 in order to get full control over b, it easily proves the theorem. Analogously, we prove that the dual formulation is stable. □

4.6 Numerical experiments

4.6.1 Convergence of a time-evolutive analytical solution

In this section, we aim to compare the behavior of the segregated algorithms presented in this work with respect to a monolithic approach. We have simulated a time-evolutive problem with analytical solution in the 2D unit square for the time interval $[0, 5]$ s. The computational domain has been discretized using a mesh of 3281 nodes and 6400 linear triangular elements. The expression of the analytical solution we have used is:

$$\begin{array}{ll} u_x = y\sin(\pi t/10)\exp(t/25), & b_x = y\sin(\pi t/10)\exp(t/25), \\ u_y = -x\sin(\pi t/10)\exp(t/25), & b_y = -x\sin(\pi t/10)\exp(t/25), \\ p = x + y, & r = 0. \end{array}$$

This analytical solution is obtained through the boundary conditions and the source terms f_u and f_b.

Table 4.1 shows the results we have obtained for the velocity u solving the problem for several time-step sizes δt. The values correspond to the L^2-norm of the velocity error for the monolithic scheme using Crank-Nicolson as time integrator (CN-MONOLITHIC) and three versions of operator splitting algorithms. We consider Algorithm 4.1, that segregates the *pressures* using a second order accurate splitting and Crank-Nicolson time integration in its semi-implicit (CN-SEGP-SI) and implicit version (CN-SEGP) and Algorithm 4.2 in semi-implicit form that segregates the four unknowns computation together with Backward-Euler time integration (BE-SEGPU). It can be seen that both the monolithic approach and the proposed segregation schemes verify the expected accuracies in time.

Similarly, Table 4.2 lists the error norms for the magnetic induction b when solving the problem with the same series of time-step sizes δt. The computed errors are the L^2-norm of the difference between the analytical (exact) solution and the numerical solution for the same previous algorithms, CN-MONOLITHIC, CN-SEGP, BE-SEGPU and CN-SEGP-SI. It is clear that the order of convergence of each one of the studied cases verifies the theoretical expectations. As expected, the first order segregated scheme that uncouples all the unknowns (BE-SEGPU) has a worse accuracy (the errors are higher).

4.6.2 Delay in the computation of r

In this subsection, we compare the effect of freezing the magnetic pseudo-pressure r for several steps when using the pressure segregation algorithms (SEGP) and the fully uncoupled (SEGPU) scheme. We have solved the MHD problem in a 2D square domain

δt	CN-MONOLITHIC	CN-SEGP	BE-SEGPU	CN-SEGP-SI
1.0	$2.13 \cdot 10^{-2}$ (-)	$2.61 \cdot 10^{-2}$ (-)	$7.21 \cdot 10^{-2}$ (-)	$2.78 \cdot 10^{-2}$ (-)
0.5	$6.55 \cdot 10^{-3}$ (1.70)	$7.35 \cdot 10^{-3}$ (1.83)	$9.74 \cdot 10^{-3}$ (2.89)	$7.09 \cdot 10^{-3}$ (1.97)
0.1	$2.53 \cdot 10^{-4}$ (2.02)	$3.12 \cdot 10^{-4}$ (1.96)	$6.38 \cdot 10^{-4}$ (1.69)	$3.11 \cdot 10^{-4}$ (1.94)
0.05	$6.31 \cdot 10^{-5}$ (2.00)	$7.90 \cdot 10^{-5}$ (1.98)	$3.32 \cdot 10^{-4}$ (0.94)	$8.04 \cdot 10^{-5}$ (1.95)
0.01	$2.49 \cdot 10^{-6}$ (2.01)	$3.25 \cdot 10^{-6}$ (1.98)	$6.82 \cdot 10^{-5}$ (0.98)	$3.25 \cdot 10^{-6}$ (1.99)

Table 4.1: Numerical errors and rate of convergence in brackets for the velocity u. Time-evolutive analytical solution.

δt	CN-MONOLITHIC	CN-SEGP	BE-SEGPU	CN-SEGP-SI
1.0	$2.23 \cdot 10^{-2}$ (-)	$2.21 \cdot 10^{-2}$ (-)	$4.35 \cdot 10^{-1}$ (-)	$2.36 \cdot 10^{-2}$ (-)
0.5	$6.29 \cdot 10^{-3}$ (1.83)	$6.20 \cdot 10^{-3}$ (1.83)	$3.00 \cdot 10^{-1}$ (0.54)	$5.84 \cdot 10^{-3}$ (2.01)
0.1	$2.43 \cdot 10^{-4}$ (2.02)	$2.43 \cdot 10^{-4}$ (2.01)	$8.39 \cdot 10^{-2}$ (0.79)	$2.41 \cdot 10^{-4}$ (1.98)
0.05	$6.06 \cdot 10^{-5}$ (2.00)	$6.13 \cdot 10^{-5}$ (1.99)	$4.39 \cdot 10^{-2}$ (0.93)	$6.12 \cdot 10^{-5}$ (1.98)
0.01	$2.42 \cdot 10^{-6}$ (2.00)	$2.66 \cdot 10^{-6}$ (1.95)	$9.08 \cdot 10^{-3}$ (0.98)	$2.66 \cdot 10^{-6}$ (1.95)

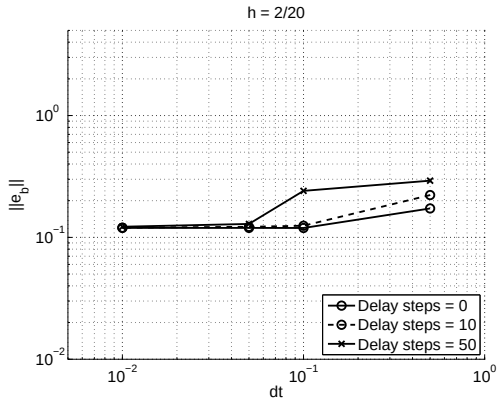
Table 4.2: Numerical errors and rate of convergence in brackets for the magnetic induction b. Time-evolutive analytical solution.

$[-1,1] \times [-1,1]$ imposing an analytical solution of the form:

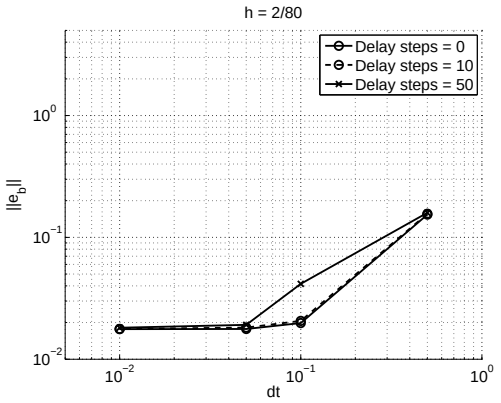
$$\begin{array}{ll} u_x = x^4 \sin(\pi t/10) \exp(t/25), & b_x = x^4 \sin(\pi t/10) \exp(t/25), \\ u_y = -4yx^3 \sin(\pi t/10) \exp(t/25), & b_y = -4yx^3 \sin(\pi t/10) \exp(t/25), \\ p = x^2 + y^2, & r = 0. \end{array}$$

We have solved the transient problem for the time interval $[0, 5]$ s using several time-steps, $\delta t = 0.50, 0.10, 0.05, 0.01$s. We have computed the L^2-norm of the error of the magnetic induction field in 2 meshes composed by 800 and 12800 linear triangular elements, respectively. Figure 4.1 displays the results we have obtained when using the following schemes: the implicit pressure segregation scheme together with the Crank-Nicolson time integrator (CN-SEGP), its semi-implicit counterpart (CN-SEGP-SI) and the fully uncoupled scheme (BE-SEGPU).

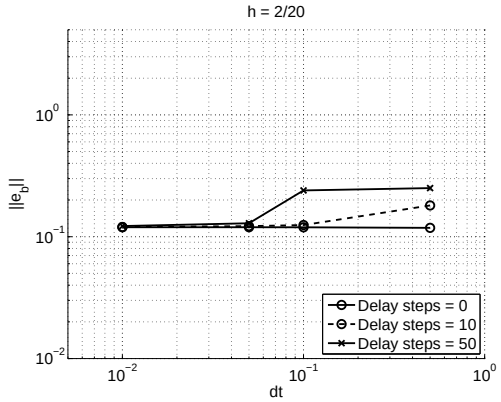
Regarding the two versions of the classical fractional step scheme (SEGP), it can be clearly seen that it is not needed to compute r at any time step, clearly reducing the total computational cost. For this particular test, the computed error is acceptable only computing r at every 10 time steps (denoted as 10 delay steps in the figures) when solving with a time-step size less or equal than 0.1 s and for 50 delay steps when using a time-step size less or equal than 0.05 s. Furthermore, it is also clear that the extra error we obtain when not updating r in any step reduces when the mesh is refined. On the other hand, the results obtained for the fully uncoupled segregated scheme (SEGPU) show that the error introduced when freezing the computation of r is lower than the error due to the uncoupling itself since the total error is not sensitive to the delaying steps.



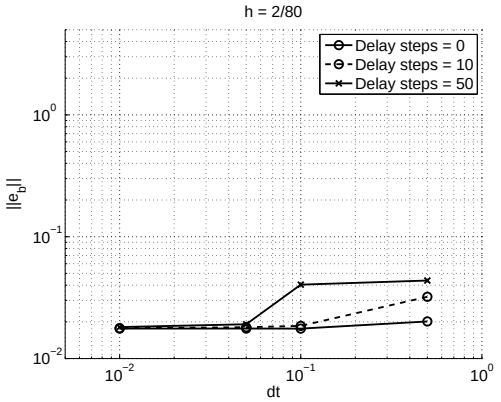
(a) CN-SEGP - 800 elements



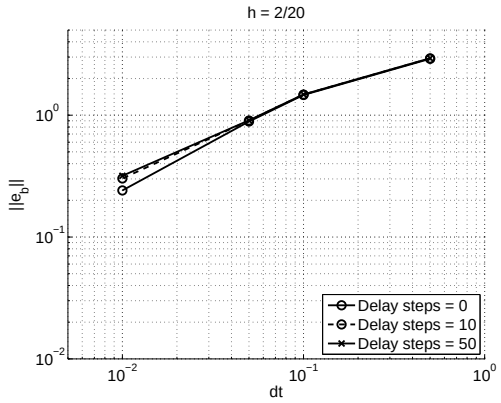
(b) CN-SEGP - 12800 elements



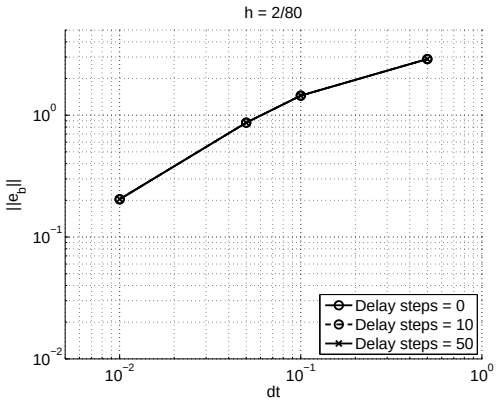
(c) CN-SEGP-SI - 800 elements



(d) CN-SEGP-SI - 12800 elements



(e) BE-SEGPU - 800 elements



(f) BE-SEGPU - 12800 elements

Figure 4.1: Delay in computation of r.

4.6.3 Flow around an obstacle in 2D

This subsection is devoted to the approximation of a classical problem in fluid mechanics, the flow around an obstacle. We have chosen this example for two reasons. First, we want to check that the developed numerical algorithms are able to reproduce the physics behind magnetohydrodynamics. Furthermore, this is a good problem to study the time evolutive properties of the several schemes presented in this work.

From the physical point of view, the presence of an external magnetic field modifies the flow of a conducting fluid. It is well-known from fluid mechanics that for high Reynolds numbers, a flow becomes turbulent with the generation of vortices. However, when the same conducting fluid flows under an external magnetic field, the vortex shedding disappears and the vortices are reduced for high enough magnetic fields, leading to laminar flows. This physical property of magnetohydrodynamics has been studied in this example.

The computational domain has been taken as $\Omega = [0, 31] \times [0, 11]$ with an internal obstacle defined by the square $[15, 16] \times [5, 6]$. This domain has been discretized using a mesh of 4610 nodes and 9000 linear triangular elements. The boundary conditions used in the simulations are $u = (1, 0)$ m/s on the upper and lower boundaries, $u = 0$ on the obstacle and periodic boundary conditions on the inlet (left boundary) and the outlet (right boundary) regarding the hydrodynamics unknowns. For the magnetic unknowns, the boundary conditions are an imposed normal component of the magnetic induction on the upper and lower boundaries, an imposed tangent component together with $r = 0$ on the obstacle and periodic boundary conditions on the inlet and the outlet.

We have solved the problem for a Reynolds number $\mathrm{Re} = 100$ and two Hartmann numbers $\mathrm{Ha} = 1, 100$ depending on the norm of the externally applied magnetic field. The simulations have been carried out beginning with an initial condition of $u = (1, 0)$ m/s on the whole domain Ω. The computation has been done till a final time $T = 250$ s for both cases. In each case the time-step size has been set as $\delta t = 0.10$ s. The results obtained when solving the problem using the monolithic method together with the Backward Euler time-integration scheme (BE-MONOLITHIC) are shown in Figure 4.2. It is clearly seen how the vortex shedding disappears when the magnetic field is increased and also how the vortices are drastically reduced.

The second goal of this example was to study the time-evolutive behavior of the several operator splitting schemes presented in this work. We have focused in the problem with $\mathrm{Ha} = 1$, which leads to an unsteady solution. In order to compare the solution obtained with each method, we have set as the initial solution to start the computation the solution of the monolithic scheme at $T = 250$ s that can be seen in Figure 4.2. The computations were run until a final time of $T = 60$ s for two time-step sizes, $\delta t = 1.0, 0.1$ s.

Figures 4.3(a) and 4.3(b) show the time-evolutive graph of the y-component of the velocity field at point $(18, 5.5)$ which is located after the obstacle, whereas Figures 4.3(c) and 4.3(d) show the x-component of the magnetic induction field at the same point. For a big time step size, $\delta t = 1.0$ s, the 2nd order classical fractional step scheme (CN-SEGP) has a good behavior at simulating the period of the solution. The semi-implicit version of the 2nd order segregation algorithm (CN-SEGP-SI) obtains a solution with a similar period than the reference solution but it is not able to capture properly the amplitude when the time advances. Finally, the fully uncoupled scheme (BE-SEGPU) leads to a

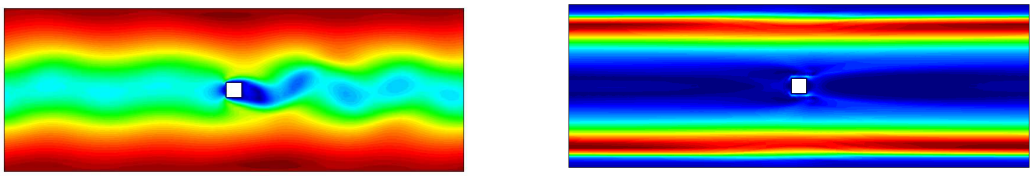


Figure 4.2: Velocity contours. From left to right: Ha $= 1, 100$.

much worse solution, both in terms of period and amplitude. However, when the time-step is reduced, $\delta t = 0.1$ s, the solution obtained with the operator splitting schemes improves drastically. Therefore, we can conclude that for a suitable time-step size, the use of fractional step methods is very appealing in the sense that the computational cost is much lower than monolithic schemes and the solution we obtain is accurate. These observations are the same as for the Navier-Stokes equations alone, and it is the reason of the wide popularity of fractional step schemes in computational fluid dynamics.

Furthermore, we were also interested in studying the convergence to a stationary solution for high Hartmann numbers. Figure 4.4 shows the point evolution graphs when solving the problem with Ha $= 10$. It is clearly seen that the several studied versions of segregation schemes (CN-SEGP, CN-SEGP-SI and BE-SEGPU) are able to obtain a good solution compared to the stationary solution of the monolithic scheme.

4.6.4 2D island coalescence problem

This subsection is devoted to the solution of a two-dimensional driven reconnection magnetic problem, the island coalescence problem. Fast magnetic reconnection is one of the main issues for understanding plasma physics. The island coalescence problem (IC) consists of magnetic islands embedded in a Harris current sheet, see [69, 71] for more details on the physics that lie behind it.

The initial equilibrium is defined by, see [40, 88, 112]

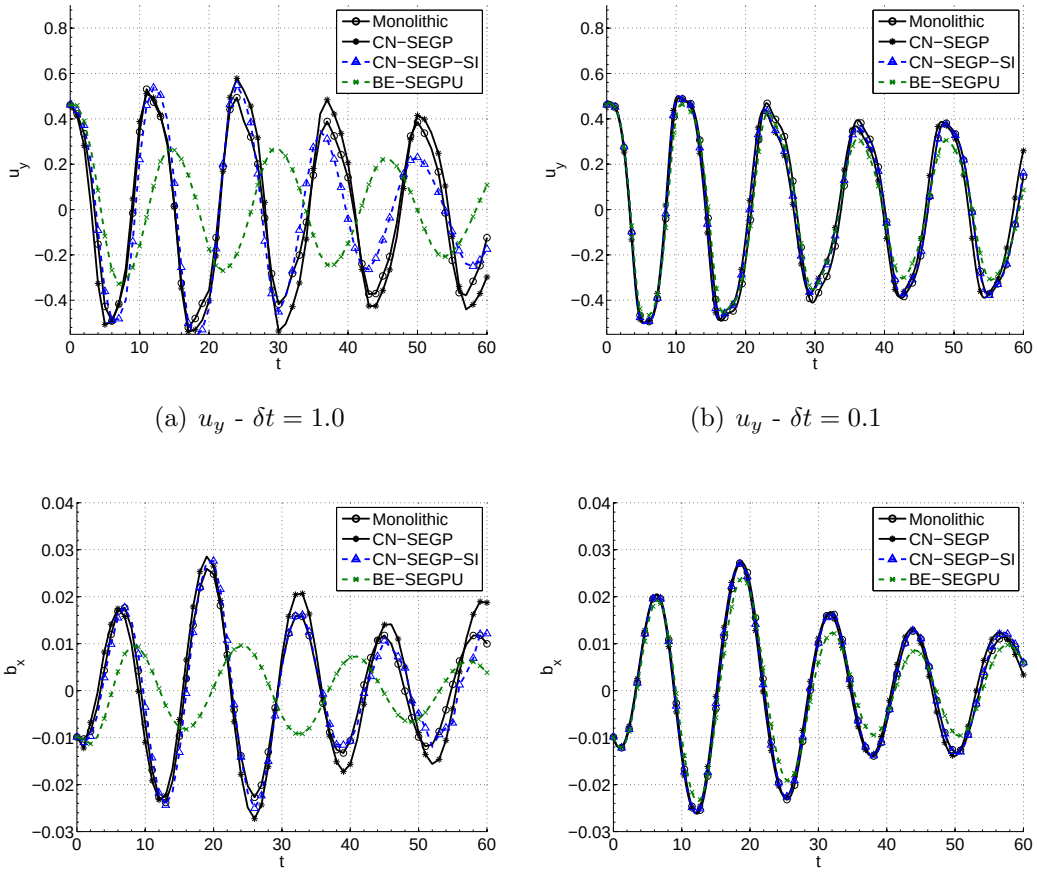
$$u(x,y,0) = 0, \qquad p(x,y,0) = P_0 + \frac{1-\epsilon^2}{2[\cosh(\frac{y}{\delta}) + \epsilon\cos(\frac{x}{\delta})]^2},$$

$$b_x(x,y,0) = \frac{\sinh(\frac{y}{\delta})}{\cosh(\frac{y}{\delta}) + \epsilon\cos(\frac{x}{\delta})}, \qquad b_y(x,y,0) = \frac{\epsilon\sin(\frac{x}{\delta})}{\cosh(\frac{y}{\delta}) + \epsilon\cos(\frac{x}{\delta})}.$$

where $\delta = 1/(2\pi)$, $\epsilon = 0.2$ and $P_0 = 1.0$. The problem dynamics are produced by the addition of an external electric field which plays the role of an external force in the magnetic subproblem,

$$f_{bx}(x,y) = \frac{2\eta(1-\epsilon^2)}{\delta^2}\frac{\sinh(\frac{y}{\delta})}{[\cosh(\frac{y}{\delta}) + \epsilon\cos(\frac{x}{\delta})]^3}, \quad f_{by}(x,y) = \frac{2\epsilon\eta(1-\epsilon^2)}{\delta^2}\frac{\sin(\frac{x}{\delta})}{[\cosh(\frac{y}{\delta}) + \epsilon\cos(\frac{x}{\delta})]^3}$$

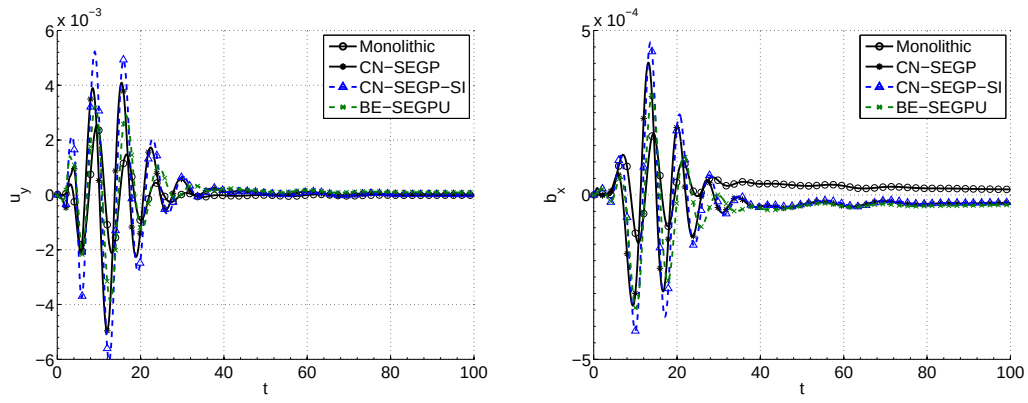
where $\eta = \frac{1}{\mu_m\sigma}$. The computational domain is the square $[-1,1]\times[-1,1]$ and it has been discretized using linear triangular elements. The boundary conditions for this problem correspond to zero tangential stress ($u_y = 0$) and perfect conducting wall ($b_y = 0$) for



(a) u_y - $\delta t = 1.0$ (b) u_y - $\delta t = 0.1$

(c) b_x - $\delta t = 1.0$ (d) b_x - $\delta t = 0.1$

Figure 4.3: Temporal evolution at point (18, 5.5). Hartmann number Ha = 1.



(a) u_y (b) b_x

Figure 4.4: Temporal evolution at point (18, 5.5). Hartmann number Ha = 10.

the top and bottom boundaries ($y = -1$ and $y = 1$) and periodic conditions in the left and right boundaries ($x = -1$ and $x = 1$). The physical properties of the problem have been set as $\rho = 1$, $\nu = 10^{-3}$, $\sigma = 1$ and $\mu_m = 10^3$. Therefore, the resistivity is $\eta = 10^{-3}$.

Since we have not been able to find an experimental solution of this problem to compare with a computational solution, we have solved the problem using the monolithic scheme together with the Crank-Nicolson method for time integration. We have used a mesh consisting of 40000 linear triangular elements and 20201 nodes. The pressure and magnetic induction fields obtained are shown at different times in Figure 4.5. This solution will be used as a reference solution to compare with the solutions obtained when using the several operator splitting schemes presented in this work.

Figure 4.6 includes a series of evolution graphs that compare the reference solution with the solutions obtained when solving the problem with the same segregation schemes studied in the previous numerical examples, that is CN-SEGP, CN-SEGP-SI and BE-SEGPU. We have chosen the evolutive solution of the pressure at the point $(0, 0)$ and the magnetic induction (euclidean) norm at point $(0.5, 0)$ because these are the points where the study of the dynamics of the problem is more interesting from a physical point of view. As expected, segregation algorithms tend to the monolithic solution as the time step size is reduced. Another reasonable result is the fact that semi-implicit versions of the algorithm start to show some local oscillations when the time step size is too large; it is due to we are linearizing at every time step size value with a solution from the previous time step size, which is very different to the current one, due to the large time step size. In any case, the oscillations are local and only happen when fast changes of the solution happen and the situation is solved by adequately reducing the time step size.

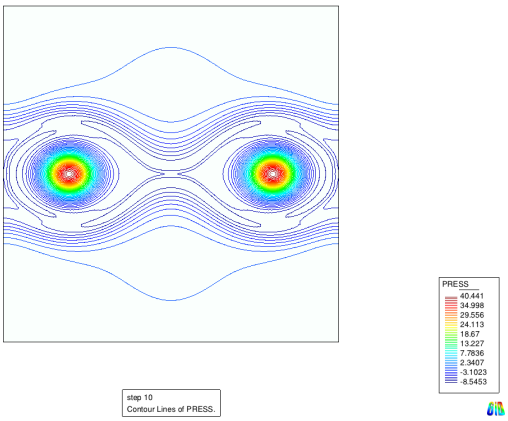
It is important to note that, as it is well known, segregation schemes need smaller time-step sizes in order to obtain solutions with good enough accuracy. This fact is even more crucial when using the total uncoupling scheme (BE-SEGPU).

4.7 Conclusions

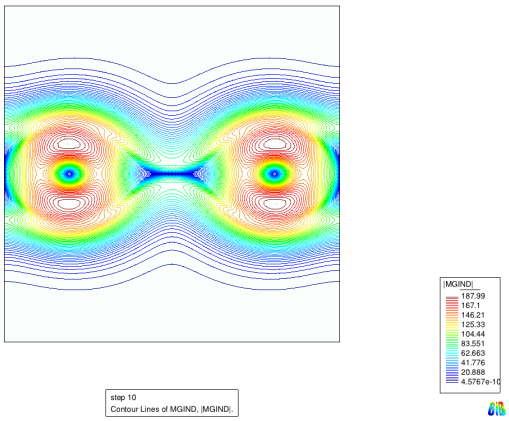
In this chapter, we have proposed and analyzed a term-by-term stabilized finite element formulation based on orthogonal subscales for the numerical approximation of the incompressible MHD system. The resulting scheme is well-posed in the correct functional setting, as the residual-based formulation in Chapter 2, but it also keeps the skew-symmetry of the off-diagonal blocks.

This last property is basic in order to design unconditionally stable operator splitting techniques. We have considered two different levels of splitting. First, we have carried out the (fluid and magnetic) pressure segregation, extending pressure correction algorithms for the incompressible Navier-Stokes problem to the problem at hand. In a second step, we have considered the splitting of the fluid and induction fields, using a simple but first order accurate approach that leads to two alternative algorithms. Further, we have pointed out how to improve the order of accuracy, by using the ideas of velocity-correction methods in [10]. In any case, since the velocity-induction field block has a skew-symmetric off-diagonal structure, we could also consider pressure correction type schemes, not considered in this chapter for the sake of conciseness.

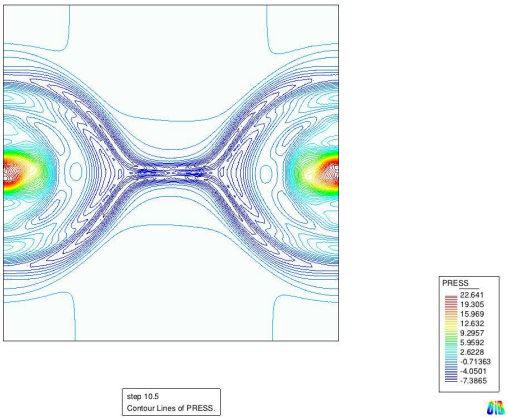
Numerical experiments have showed the performance of the different algorithms in



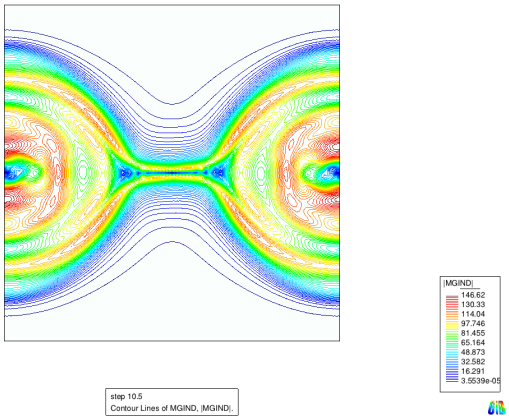
(a) Pressure at $t = 10.0$ s



(b) Magnetic induction norm at $t = 10.0$ s

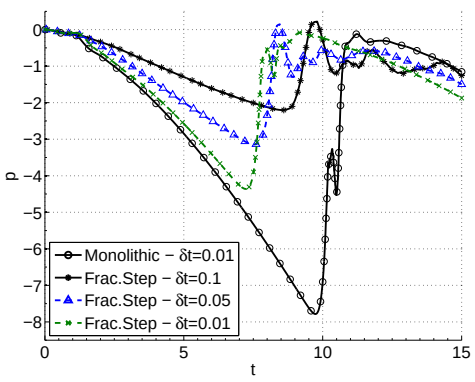


(c) Pressure at $t = 10.5$ s

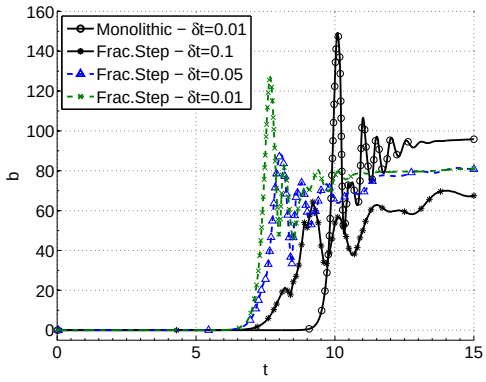


(d) Magnetic induction norm at $t = 10.5$ s

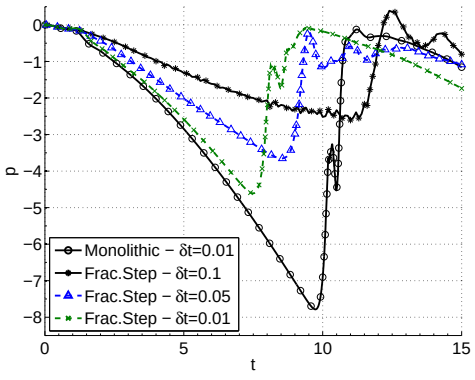
Figure 4.5: Contour lines of the pressure and magnetic induction fields at different times



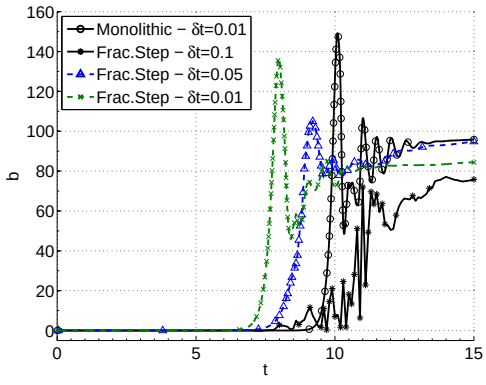
(a) CN-SEGP



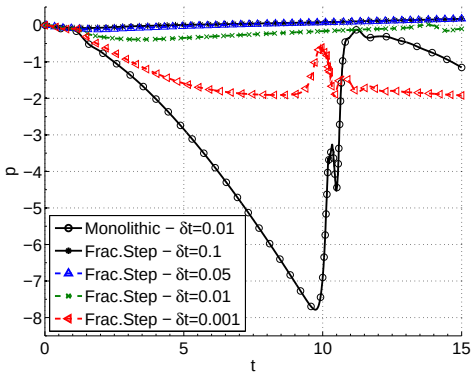
(b) CN-SEGP



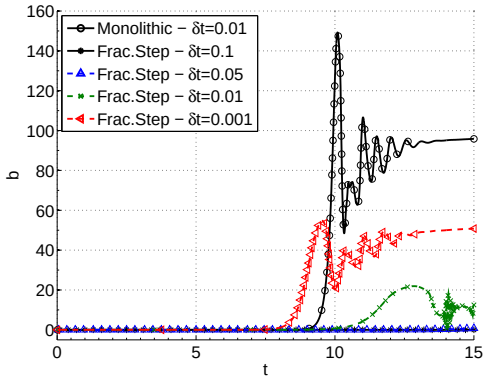
(c) CN-SEGP-SI



(d) CN-SEGP-SI



(e) BE-SEGPU



(f) BE-SEGPU

Figure 4.6: Temporal evolution, left: pressure at point (0,0); right: magnetic induction norm at point (0.5, 0)

their semi-implicit and implicit versions compared to the results of the fully coupled implicit scheme. As expected, segregated algorithms introduce an additional error that is reduced with the time step size. The second order pressure segregation used performs better than the vectorial fields uncoupling, since we have considered first order schemes. Second order velocity-induction segregation schemes should be used in order to improve accuracy.

All the schemes presented herein have been motivated at the algebraic level. It allows to straightforwardly use these schemes as preconditioners of Krylov solvers, instead of perturbed solvers (see [67]). The use of these schemes as preconditioners, which now are positive-definite matrices, as well as its application in parallel by scalable domain decomposition techniques is the main objective of subsequent work, see Chapter 6.

Chapter 5

Inductionless MHD problem

In this chapter, a stabilized formulation to solve the inductionless magnetohydrodynamic (MHD) problem using the finite element (FE) method is presented. The inductionless MHD problem couples the Navier-Stokes equations and a Darcy-type system for the electric potential via the Lorentz's force in the momentum equation of the Navier-Stokes equations and the currents generated by the moving fluid in Ohm's law. The key feature of the FE formulation resides in the design of the stabilization terms, which serve several purposes. First, the formulation is suitable for convection dominated flows. Second, there is no need to use interpolation spaces constrained to a compatibility condition in both sub-problems and therefore, equal-order interpolation spaces can be used for all the unknowns. Finally, this formulation leads to a coupled linear system; this monolithic approach is effective, since the coupling can be dealt by effective preconditioning and iterative solvers that allows to deal with high Hartmann numbers.

5.1 Introduction

The objective of this chapter is to present a finite element (FE) method for the approximation of the inductionless magnetohydrodynamic (MHD) problem which arises when the magnetic field induced by currents in the liquid metal is negligible compared to the external magnetic field, $\boldsymbol{B}$. The inductionless approximation to the MHD problem consists of the momentum, mass and charge conservation equations together with Ohm's law; the problem is written in terms of velocity $\boldsymbol{u}$, pressure p, current density $\boldsymbol{j}$ and electric potential ϕ. The structure of this system of partial differential equations corresponds to the Navier-Stokes equations coupled to a Darcy-type problem via the Lorentz's force and generated currents terms.

This set of equations can be used to model several industrial processes, such as MHD pumps based on conduction or induction principles, MHD generators, continuous casting of steel, crystal growth devices or test blanket modules (TBMs) in nuclear fusion reactors. TBMs will be one of the key components of ITER (International Termonuclear Experimental Reactor), that should demonstrate the scientific reliability of fusion (see www.iter.org for more details). Each of these breeding blankets is designed in a modular shape performing a triple function: 1) heat power extraction from the plasma, 2) tritium generation (breeding) and 3) shielding of the magnets from neutron and gamma radiation. The breeding material used is the eutectic lead-lithium liquid metal. In normal

regimes, this liquid metal flow can be modeled by the inductionless MHD equations. The aim to design effective TBMs and the lack of experimental data has increased the demand of numerical methods for this system of equations.

The FE approximation of the inductionless MHD system faces several difficulties. First, there is the classical and well-known problem of dealing with cases in which the first order derivatives, i.e. the convective term in the Navier-Stokes equations, dominate the second order ones, i.e. the viscous term. In this singular limit, the elliptic nature of the system of differential equations vanishes. This behavior may lead to oscillations when using crude Galerkin techniques. Second, there is the compatibility condition between the approximation spaces for the velocity and the pressure, but also for the current density and the electric potential. These conditions are expressed in a classical inf-sup form. Finally, the coupling between the hydrodynamic and the electromagnetic problems may lead to numerical difficulties when solving the resulting discrete system of equations. In the Navier-Stokes equations, the coupling comes from Lorentz's force, whereas in the magnetic problem the coupling appears in Ohm's law because the conducting fluid moves with velocity $\boldsymbol{u}$. The goal of this chapter is to design a stabilized FE method able to circumvent all these problems.

The stabilization technique presented in this chapter is developed in the variational multiscale framework introduced in [83]. It is based on a two-scale decomposition of the unknowns into a FE component and a subgrid scale or subscale that corresponds to the unknown component that can not be captured by the FE space. The key ingredient is the model for the subgrid scales. In this chapter, we have considered the subgrid scales as a projection of the residual of the FE approximation times a matrix of stabilization parameters. Among the several options for the projection and the structure of the matrix of stabilization parameters, the identity and a diagonal structure have been chosen, respectively. Up to this point, the only missing issue to close the formulation is the design of the stabilization parameters. Based on the stability and convergence analyses of the method, we have obtained an effective expression for them.

In the last years, the increasing demand of computational tools for the design on fusion reactor technology has increased the interest on computational MHD. However, the literature about the numerical approximation of the inductionless MHD equations is still quite scarce. There has been some recent research done in the finite difference and finite volume community. The finite volume method has been used to solve the inductionless MHD equations in simulations of the HCLL test blanket module in nuclear fusion reactors (see, e.g., [37,95] for examples in this field). In all cases, the methodology consists of first solving a Poisson equation for the electric potential (obtained by taking the divergence of the Ohm's law) and then, solving the Navier-Stokes equations adding the Lorentz force as a body force in the momentum equation. A crude fixed point iterative algorithm is used to converge to the coupled solution. Using this approach, a conservative finite volume scheme for incompressible MHD flows is proposed in [101,102]; the scheme further uncouples the computation of velocities and pressures via a pressure segregation scheme; see, e.g., [9, 51] for a detailed exposition of pressure segregation schemes and the quite poor performance of fixed point iterations over the $\boldsymbol{u}$-p resulting system. Since block-Jacobi and block-Gauss-Seidel preconditioned Richardson iterations converge in a quite poor fashion (when convergence is attained), more effective preconditioned solvers are mandatory for large scale simulations (see, e.g., [67]). This observation motivates

the monolithic approach proposed herein; the scheme we propose ends up with a linear system that couples the fluid and magnetic problems. The coupling can be transferred to the preconditioner in an effective manner and then an efficient and robust iterative solver (like GMRES) can be used, leading to optimal MHD solvers. In this chapter, we have considered incomplete LU factorizations of the monolithic system as preconditioners, together with a GMRES iterative solver.

There exist several articles applying the FE method to solve the full MHD equations in the general case of non-negligible induced magnetic field (see for instance [53, 72, 110–112]) but the authors are not aware of previous works dealing with the approximation of the inductionless MHD by the FE method. For this chapter, we have used the same methodology as for the full MHD problem in [54], treating the same issues with a similar strategy, even though the problems considered are significantly different from the point of view of the mathematical structure.

The chapter is organized as follows. The problem to be solved is stated in Section 5.2, both in its continuous and its variational form. Issues regarding the time integration and the linearization of the nonlinear term are discussed in Section 5.3, leading to a time discrete and linearized scheme. Next, the variational multiscale framework is applied to the inductionless MHD problem in Section 5.4. After proposing the stabilization method, it is fully analyzed regarding its stability, accuracy and convergence properties; it motivates an optimal expression of the stabilization parameters that takes into account the coupling. The final scheme proposed in this chapter is written in Section 5.5. Numerical experiments verifying the theoretical results are presented in Section 5.6 and finally, some conclusions are drawn in Section 5.7.

5.2 Problem statement

5.2.1 Initial and boundary value problem

Let $\Omega \subset \mathbb{R}^d$ (d =2 or 3) be a domain where we want to solve the inductionless MHD problem during the time interval $[0, T]$. The unknowns of the problem are the fluid velocity $\boldsymbol{u} : \Omega \times (0,T) \longrightarrow \mathbb{R}^d$, the pressure $p : \Omega \times (0,T) \longrightarrow \mathbb{R}$, the current density $\boldsymbol{j} : \Omega \times (0,T) \longrightarrow \mathbb{R}^d$ and the electric potential $\phi : \Omega \times (0,T) \longrightarrow \mathbb{R}$, which are the solution of the system of partial differential equations:

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \nabla p - \frac{1}{\rho}(\boldsymbol{j} \times \boldsymbol{B}) = \boldsymbol{f}, \tag{5.1}$$

$$\nabla \cdot \boldsymbol{u} = 0, \tag{5.2}$$

$$\boldsymbol{j} + \sigma \nabla \phi - \sigma(\boldsymbol{u} \times \boldsymbol{B}) = 0, \tag{5.3}$$

$$\nabla \cdot \boldsymbol{j} = 0, \tag{5.4}$$

where ρ is the fluid density, $\boldsymbol{B}$ the external magnetic field, $\boldsymbol{f}$ the body forces of the flow motion and σ the electric conductivity. It is important to note that the pressure p we are working with here is the kinematic pressure (pressure divided by density).

Let us define two different partitions of the domain boundary $\Gamma = \partial\Omega$. The first one, for imposing the boundary conditions of the hydrodynamic unknowns, is divided into the part $\Gamma_{E,u}$ in which essential (Dirichlet) boundary conditions are enforced, and the rest

of the boundary $\Gamma_{N,u}$ where we impose natural (Neumann) boundary conditions. The other partition is used for the boundary conditions of the magnetic problem. It consists of the part of the boundary $\Gamma_{C,j}$ that corresponds to perfectly conducting walls and the part $\Gamma_{I,j}$ that corresponds to perfectly insulated walls. So, we have:

$$\Gamma = \Gamma_{E,u} \cup \Gamma_{N,u} = \Gamma_{C,j} \cup \Gamma_{I,j}, \qquad \text{and} \qquad \emptyset = \Gamma_{E,u} \cap \Gamma_{N,u} = \Gamma_{C,j} \cap \Gamma_{I,j}.$$

The boundary conditions for the velocity at the walls are the non-slip wall conditions, that is to say, $\boldsymbol{u} = \mathbf{0}$ on $\Gamma_{E,u}$. On the other hand, the free boundary conditions for the velocity are zero traction conditions,

$$-p\boldsymbol{n} + \nu \boldsymbol{n} \cdot \nabla \boldsymbol{u} = 0, \quad \text{on } \Gamma_{N,u}.$$

Two different kinds of boundary conditions have been considered for the magnetic equations. For insulating walls, the electric currents cannot cross the wall surface, which implies that the normal component of the density currents has to vanish, that is, $\boldsymbol{j} \cdot \mathbf{n} = 0$ on $\Gamma_{I,j}$. On the other hand, perfectly conducting walls do not apply any resistance to the current and therefore, the electric currents cross the wall surface in an orthogonal way. This means that the tangential component of the density current has to vanish on the boundary, i.e. $\boldsymbol{j} \times \mathbf{n} = \mathbf{0}$ on $\Gamma_{C,j}$. Note that, because $\boldsymbol{u} = \mathbf{0}$ on the wall boundary, the density current and the electric potential are related as $\boldsymbol{j} = -\sigma \nabla \phi$. Therefore, on a perfectly conducting wall it is verified that $\nabla \phi \times \mathbf{n} = 0$. This means that ϕ must be constant on the boundary. So, we can model conducting walls by the boundary condition $\phi = 0$ on $\Gamma_{C,j}$ without loss of generality.

Finally, an initial condition for the velocity field has to be considered, i.e. $\boldsymbol{u} = \boldsymbol{u_0}$ in Ω at instant $t = 0$.

5.2.2 Weak form

Let us introduce some notation. Let $\langle f, g \rangle_\omega := \int_\omega fg$, where f and g are two generic functions defined on a region ω such that the integral of their product is well defined. When $f, g \in L^2(\Omega)$, we will write $(f, g)_\omega := \langle f, g \rangle_\omega$. The norm in $L^2(\Omega)$ will be denoted by $\|f\| := (f, f)^{1/2}$.

Let $\boldsymbol{v}, q, \boldsymbol{k}$ and ψ be the test functions for $\boldsymbol{u}, p, \boldsymbol{j}$ and ϕ respectively. We consider them time-independent because time will be discretized using a finite difference scheme. To obtain the weak form of (5.1)-(5.4), the equations are multiplied by the corresponding test functions, integrated over the domain Ω and the second order terms are integrated by parts, resulting in the variational form

$$(\partial_t \boldsymbol{u}, \boldsymbol{v}) + \langle \boldsymbol{u} \cdot \nabla \boldsymbol{u}, \boldsymbol{v} \rangle + \nu (\nabla \boldsymbol{u}, \nabla \boldsymbol{v}) - (p, \nabla \cdot \boldsymbol{v}) - \frac{1}{\rho} \langle \boldsymbol{j} \times \boldsymbol{B}, \boldsymbol{v} \rangle = \langle \boldsymbol{f}, \boldsymbol{v} \rangle, \tag{5.5}$$

$$(q, \nabla \cdot \boldsymbol{u}) = 0, \tag{5.6}$$

$$(\boldsymbol{j}, \boldsymbol{k}) + \sigma (\nabla \phi, \boldsymbol{k}) - \sigma (\boldsymbol{u} \times \boldsymbol{B}, \boldsymbol{k}) = 0, \tag{5.7}$$

$$-(\nabla \psi, \boldsymbol{j}) = -\langle \psi, \boldsymbol{j} \cdot \boldsymbol{n} \rangle_\Gamma, \tag{5.8}$$

which must hold for all test functions $\boldsymbol{v}, q, \boldsymbol{k}$ and ψ in the functional spaces that will be defined next. Note that σ is assumed to be constant and that the boundary term

appearing from integration by parts in (5.8) is zero both in the case of conducting walls and in the case of insulating walls. Let us assume that $\boldsymbol{B} \in L^3(\Omega)$, in order for this system to be well-posed in the subsequent functional setting. The functional spaces considered in this work are

$$\begin{aligned}
V_u &= \{\boldsymbol{v} \in H^1(\Omega)^d \mid \boldsymbol{v} = \boldsymbol{0} \text{ on } \Gamma_{E,u}\},\\
V_p &= \{q \in L^2(\Omega) \mid \int_\Omega q = 0 \text{ if } \Gamma_{N,u} = \emptyset\},\\
V_j &= \{\boldsymbol{k} \in L^2(\Omega)^d \mid \boldsymbol{k} \cdot \mathbf{n} = 0 \text{ on } \Gamma_{I,j}\},\\
V_\phi &= \{\psi \in H^1(\Omega) \mid \psi = 0 \text{ on } \Gamma_{C,j}\}.
\end{aligned}$$

Remark 5.1. *It is important to note that the $\boldsymbol{j}$-ϕ system has the same structure as the Darcy problem. The formulation selected in this chapter corresponds to the primal version of the problem. However, there exists also the dual formulation which consists of considering a different functional setting of the problem: $\boldsymbol{j} \in H(\mathrm{div};\Omega)$ and $\phi \in L^2(\Omega)$, see [11] for a complete definition and stabilized FE analysis of these two formulations for Darcy's problem.*

Remark 5.2. *From (5.3), it follows that the trace of $\boldsymbol{j} \cdot \boldsymbol{n}$ is well defined if so is the trace of $\boldsymbol{n} \cdot \nabla\phi$ and $\boldsymbol{n} \cdot (\boldsymbol{u} \times \boldsymbol{B}) = (\boldsymbol{n} \times \boldsymbol{B}) \cdot \boldsymbol{u}$. The first term is well defined because $\phi \in H^1(\Omega)$. The second term is well-defined for $\boldsymbol{B} \in H(\mathrm{curl};\Omega)$ (e.g., for $\boldsymbol{B}$ a given datum solving the Maxwell equations), since $\boldsymbol{u} \in H^1(\Omega)^d$ has trace on $\Gamma_{I,j}$ (for almost all t).*

The multilinear forms appearing in the variational form of the problem are well defined and continuous for

$$\begin{aligned}
&\boldsymbol{u} \in L^2(0,T;V_u), \quad \boldsymbol{v} \in V_u,\\
&p \in \mathcal{D}'(0,T;V_p), \quad q \in V_p,\\
&\boldsymbol{j} \in \mathcal{D}'(0,T;V_j), \quad \boldsymbol{k} \in V_j,\\
&\phi \in \mathcal{D}'(0,T;V_\phi), \quad \psi \in V_\phi.
\end{aligned}$$

In these expressions, the Bochner space $L^2(0,T;X)$ denotes the set of mappings defined on $\Omega\times(0,T)$ such that their X-spatial norm is an $L^2(0,T)$ function. Similarly, $\mathcal{D}'(0,T;X)$ denotes the set of mappings for which their X-spatial norm is a distribution in time.

The variational form of the problem (5.5)-(5.8) can be written as a single variational equation of the form

$$M(\partial_t \boldsymbol{U}, \boldsymbol{V}) + A(\boldsymbol{U}, \boldsymbol{V}) = L(\boldsymbol{V}), \tag{5.9}$$

where

$$\begin{aligned}
\boldsymbol{U} &:= [\boldsymbol{u}, p, \boldsymbol{j}, \phi]^t, \quad \boldsymbol{V} := [\boldsymbol{v}, q, \boldsymbol{k}, \psi]^t,\\
A(\boldsymbol{U},\boldsymbol{V}) &:= \langle \boldsymbol{u}\cdot\nabla\boldsymbol{u}, \boldsymbol{v}\rangle + \nu(\nabla\boldsymbol{u},\nabla\boldsymbol{v}) - (p, \nabla\cdot\boldsymbol{v}) + (q, \nabla\cdot\boldsymbol{u}) - \frac{1}{\rho}\langle \boldsymbol{j}\times\boldsymbol{B}, \boldsymbol{v}\rangle\\
&\quad + \alpha_j\left[(\boldsymbol{j},\boldsymbol{k}) + \sigma\left(\nabla\phi,\boldsymbol{k}\right) - \sigma\left(\boldsymbol{u}\times\boldsymbol{B},\boldsymbol{k}\right)\right] + \alpha_\phi\left[-(\nabla\psi,\boldsymbol{j})\right],\\
L(\boldsymbol{V}) &:= \langle \boldsymbol{f}, \boldsymbol{v}\rangle,\\
M(\boldsymbol{U},\boldsymbol{V}) &:= (\boldsymbol{u},\boldsymbol{v}).
\end{aligned}$$

The scaling coeficients α_j and α_ϕ are introduced to make $A(\boldsymbol{U},\boldsymbol{U})$ dimensionally consistent. A possible choice of these coeficients is

$$\alpha_j = \frac{1}{\rho\sigma}\,, \quad \alpha_\phi = \frac{1}{\rho}.$$

5.3 Linearization, time discretization and spatial approximation

5.3.1 Linearization of the stationary inductionless MHD problem

The simplest way to linearize problem (5.9) is by a fixed point method, i.e. Picard's method. Let us assume there exists an estimate for the velocity at iteration k, $\boldsymbol{u}^k$. Then, the approximation of $A(\boldsymbol{U},\boldsymbol{V})$ at iteration k+1 using Picard's method can be written as

$$\begin{aligned} A^{k+1}(\boldsymbol{U}^{k+1},\boldsymbol{V}) = {} & \left\langle (\boldsymbol{u}^k\cdot\nabla)\boldsymbol{u}^{k+1},\boldsymbol{v}\right\rangle + \nu(\nabla\boldsymbol{u}^{k+1},\nabla\boldsymbol{v}) - (p^{k+1},\nabla\cdot\boldsymbol{v}) \\ & + (q,\nabla\cdot\boldsymbol{u}^{k+1}) - \frac{1}{\rho}\left\langle \boldsymbol{j}^{k+1}\times\boldsymbol{B},\boldsymbol{v}\right\rangle + \frac{1}{\rho\sigma}(\boldsymbol{j}^{k+1},\boldsymbol{k}) + \frac{1}{\rho}(\nabla\phi^{k+1},\boldsymbol{k}) \\ & - \frac{1}{\rho}(\boldsymbol{u}^{k+1}\times\boldsymbol{B},\boldsymbol{k}) - \frac{1}{\rho}(\nabla\psi,\boldsymbol{j}^{k+1}). \end{aligned}$$

Remark 5.3. *Note that the linearization proposed above is the only one that leads to a stable scheme that satisfies an energy bound. It comes from the fact that testing the linearized system with* $\mathbf{v} = \boldsymbol{u}^{k+1}$ *and* $\boldsymbol{k} = \boldsymbol{j}^{k+1}$*, the coupling terms cancel out:*

$$-\frac{1}{\rho}\left\langle \boldsymbol{j}^{k+1}\times\boldsymbol{B},\boldsymbol{u}^{k+1}\right\rangle - \frac{1}{\rho}(\boldsymbol{u}^{k+1}\times\boldsymbol{B},\boldsymbol{j}^{k+1}) = 0.$$

Analogously, we can easily check that a $\boldsymbol{u}-\phi$ *formulation in wich* ϕ *is computed using a Poisson problem and* $\boldsymbol{j}$ *is recovered as a postprocess cannot lead to a stable algorithm satisfying an energy inequality. This is one of the reasons that favor the choice of a* $\boldsymbol{u}-\boldsymbol{j}$ *formulation. It implies that the problem needs to be solved for* $\boldsymbol{u}^{k+1}$*,* p^{k+1}*,* $\boldsymbol{j}^{k+1}$ *and* ϕ^{k+1} *in a coupled way. Then, it is very convenient to have all the unknowns in terms of their nodal values, i.e. equal order Lagrangian FE approximations of all the components of the vectorial quantities and scalar quantities, which reinforces the choice of a monolithic approach to solve the problem.*

Therefore, calling $\boldsymbol{a}\equiv\boldsymbol{u}^k$, $\boldsymbol{u}\equiv\boldsymbol{u}^{k+1}$, $p\equiv p^{k+1}$, $\boldsymbol{j}\equiv\boldsymbol{j}^{k+1}$ and $\phi\equiv\phi^{k+1}$, the linearization of the stationary inductionless MHD scaled problem is

$$\begin{aligned} -\nu\Delta\boldsymbol{u} + \boldsymbol{a}\cdot\nabla\boldsymbol{u} + \nabla p - \frac{1}{\rho}(\boldsymbol{j}\times\boldsymbol{B}) &= \boldsymbol{f}, \\ \nabla\cdot\boldsymbol{u} &= 0, \\ \frac{1}{\rho\sigma}\boldsymbol{j} + \frac{1}{\rho}\nabla\phi - \frac{1}{\rho}(\boldsymbol{u}\times\boldsymbol{B}) &= 0, \\ \frac{1}{\rho}\nabla\cdot\boldsymbol{j} &= 0. \end{aligned}$$

The linearized counterpart of the variational form (5.9) is written as

$$M(\partial_t \boldsymbol{U}, \boldsymbol{V}) + A^{\text{lin}}(\boldsymbol{U}, \boldsymbol{V}) = L(\boldsymbol{V}), \tag{5.10}$$

where

$$A^{\text{lin}}(\boldsymbol{U}, \boldsymbol{V}) = \nu(\nabla \boldsymbol{u}, \nabla \boldsymbol{v}) + \langle \boldsymbol{a} \cdot \nabla \boldsymbol{u}, \boldsymbol{v} \rangle - (p, \nabla \cdot \boldsymbol{v}) + (q, \nabla \cdot \boldsymbol{u}) - \frac{1}{\rho} \langle \boldsymbol{j} \times \boldsymbol{B}, \boldsymbol{v} \rangle + \frac{1}{\rho\sigma}(\boldsymbol{j}, \boldsymbol{k}) + \frac{1}{\rho}(\nabla \phi, \boldsymbol{k}) - \frac{1}{\rho}(\boldsymbol{u} \times \boldsymbol{B}, \boldsymbol{k}) - \frac{1}{\rho}(\nabla \psi, \boldsymbol{j}).$$

5.3.2 Stability of the continuous and linearized problem

Consider the linearized stationary problem. Its variational form is : Find the solution $\boldsymbol{U} \in (V_u \times V_p \times V_j \times V_\phi)$ of the problem

$$A^{\text{lin}}(\boldsymbol{U}, \boldsymbol{V}) = L(\boldsymbol{V}) \quad \forall \boldsymbol{V} \in (V_u \times V_p \times V_j \times V_\phi). \tag{5.11}$$

Note that, since $\nabla \cdot \boldsymbol{a} = 0$ at the continuous level, A^{lin} satisfies the stability estimate

$$A^{\text{lin}}(\boldsymbol{U}, \boldsymbol{U}) = \nu \|\nabla \boldsymbol{u}\|^2 + \frac{1}{\rho\sigma} \|\boldsymbol{j}\|^2. \tag{5.12}$$

In order to be able to guarantee that the linearized problem is well posed, the inf-sup conditions between V_u and V_p and between V_j and V_ϕ have to be added to the stability estimate given by (5.12); we refer to [34, 68] for a detailed exposition of these concepts. For the inductionless MHD problem, the corresponding inf-sup conditions are

$$\inf_{q \in V_p} \sup_{\boldsymbol{v} \in V_u} \frac{(q, \nabla \cdot \boldsymbol{v})}{\|q\| \|\nabla \boldsymbol{v}\|} \geq \beta^* > 0, \qquad \inf_{\psi \in V_\phi} \sup_{\boldsymbol{k} \in V_j} \frac{(\nabla \psi, \boldsymbol{k})}{\|\nabla \psi\| \|\boldsymbol{k}\|} \geq \gamma^* > 0,$$

where β^* and γ^* are positive constants. Therefore, for each iteration k and given $\boldsymbol{u}^k$, there exists a unique solution $(\boldsymbol{u}^{k+1}, p^{k+1}, \boldsymbol{j}^{k+1}, \phi^{k+1})$ of the linearized problem (5.11).

5.3.3 Time discretization of the linearized scheme

Consider the variational problem given by (5.10) and a uniform partition of the time domain $[0, T]$ of size δt, the time step size. The method used in this work for the time integration is

$$M(\delta_t \boldsymbol{U}^n, \boldsymbol{V}) + A^{\text{lin}}(\boldsymbol{U}^{n+1}, \boldsymbol{V}) = L(\boldsymbol{V}),$$

where $\delta_t \boldsymbol{U}^n = \delta t^{-1}(\boldsymbol{U}^{n+1} - \boldsymbol{U}^n)$. This time discretization corresponds to the Backward-Euler method, which is a first-order method in time. Other time integration schemes could also be applied to obtain the final discrete problem, e.g. the second order Crank-Nicholson scheme. Anyway, the following discussion can straightforwardly be extended to other time integration schemes.

The time discrete and linearized scheme reads as:
For $n = 0, 1, 2, ..., T/\delta t$, given $\boldsymbol{u}^n$ find $\boldsymbol{u}^{n+1}$, p^{n+1}, $\boldsymbol{j}^{n+1}$ and ϕ^{n+1} as the converged solutions of the following iterative algorithm, initialized with the values at the previous time step n:

$$(\delta_t \boldsymbol{u}^{n,k+1}, \boldsymbol{v}) + \left\langle (\boldsymbol{u}^{n+1,k} \cdot \nabla)\boldsymbol{u}^{n+1,k+1}, \boldsymbol{v} \right\rangle + \nu(\nabla \boldsymbol{u}^{n+1,k+1}, \nabla \boldsymbol{v}) - (p^{n+1,k+1}, \nabla \cdot \boldsymbol{v}) - \frac{1}{\rho} \left\langle \boldsymbol{j}^{n+1,k+1} \times \boldsymbol{B}, \boldsymbol{v} \right\rangle = \left\langle \boldsymbol{f}^{n+1}, \boldsymbol{v} \right\rangle, \tag{5.13}$$

$$(q, \nabla \cdot \boldsymbol{u}^{n+1,k+1}) = 0, \tag{5.14}$$

$$\frac{1}{\rho\sigma}(\boldsymbol{j}^{n+1,k+1}, \boldsymbol{k}) + \frac{1}{\rho}(\nabla \phi^{n+1,k+1}, \boldsymbol{k}) - \frac{1}{\rho}(\boldsymbol{u}^{n+1,k+1} \times \boldsymbol{B}, \boldsymbol{k}) = 0, \tag{5.15}$$

$$-\frac{1}{\rho}(\nabla \psi, \boldsymbol{j}^{n+1,k+1}) = 0, \tag{5.16}$$

where $k > 0$ is the iteration counter. Therefore, considering $\boldsymbol{a} \equiv \boldsymbol{u}^{n+1,k}$, $\boldsymbol{u} \equiv \boldsymbol{u}^{n+1,k+1}$, $p \equiv p^{n+1,k+1}$, $\boldsymbol{j} \equiv \boldsymbol{j}^{n+1,k+1}$ and $\phi \equiv \phi^{n+1,k+1}$, the differential equations associated to (5.13)-(5.16) are

$$\begin{aligned} \delta_t \boldsymbol{u} - \nu \Delta \boldsymbol{u} + \boldsymbol{a} \cdot \nabla \boldsymbol{u} + \nabla p - \frac{1}{\rho}(\boldsymbol{j} \times \boldsymbol{B}) &= \boldsymbol{f}, \\ \nabla \cdot \boldsymbol{u} &= 0, \\ \frac{1}{\rho\sigma}\boldsymbol{j} + \frac{1}{\rho}\nabla \phi - \frac{1}{\rho}(\boldsymbol{u} \times \boldsymbol{B}) &= 0, \\ \frac{1}{\rho}\nabla \cdot \boldsymbol{j} &= 0. \end{aligned}$$

This problem can be written as the vector differential equation

$$\boldsymbol{M}\delta_t \boldsymbol{U} + \mathcal{L}(\boldsymbol{U}) = \boldsymbol{F} \quad \text{in } \Omega, \tag{5.17}$$

where $\boldsymbol{M} = \text{diag}(\boldsymbol{I}, 0, \boldsymbol{0}, 0)$, $\boldsymbol{I}$ being the $d \times d$ identity, $\delta_t \boldsymbol{U} = (\delta t)^{-1}(\boldsymbol{U} - \boldsymbol{U}^n)$, $\boldsymbol{F} = [\boldsymbol{f}, 0, \boldsymbol{0}, 0]^t$ a vector of $n_{\text{unk}} = 2d + 2$ components and the scaled operator $\mathcal{L}$ is given by

$$\mathcal{L}(\boldsymbol{U}) = \begin{bmatrix} -\nu \Delta \boldsymbol{u} + \boldsymbol{a} \cdot \nabla \boldsymbol{u} + \nabla p - \frac{1}{\rho}(\boldsymbol{j} \times \boldsymbol{B}) \\ \nabla \cdot \boldsymbol{u} \\ \frac{1}{\rho\sigma}\boldsymbol{j} + \frac{1}{\rho}\nabla \phi - \frac{1}{\rho}(\boldsymbol{u} \times \boldsymbol{B}) \\ \frac{1}{\rho}\nabla \cdot \boldsymbol{j} \end{bmatrix}. \tag{5.18}$$

The time discrete and linearized version of the variational form (5.9) can be written as

$$M(\delta_t \boldsymbol{U}, \boldsymbol{V}) + A^{\text{lin}}(\boldsymbol{U}, \boldsymbol{V}) = L(\boldsymbol{V}). \tag{5.19}$$

5.3.4 Space discretization and stability of the Galerkin approximation

The space discretization of problem (5.19) is obtained by means of the classical Galerkin FE approximation. Therefore, the problem can be stated as:
Given $\boldsymbol{U}_h^n$, find $\boldsymbol{U}_h \in (V_{u,h} \times V_{p,h} \times V_{j,h} \times V_{\phi,h})$ such that

$$M(\delta_t \boldsymbol{U}_h, \boldsymbol{V}_h) + A^{\text{lin}}(\boldsymbol{U}_h, \boldsymbol{V}_h) = L(\boldsymbol{V}_h) \quad \forall \boldsymbol{V}_h \in (V_{u,h} \times V_{p,h} \times V_{j,h} \times V_{\phi,h}), \tag{5.20}$$

where the FE spaces $V_{u,h}, V_{p,h}, V_{j,h}$ and $V_{\phi,h}$ are subspaces of their infinite dimensional counterparts V_u, V_p, V_j and V_ϕ (i.e., a conforming approximation is considered).

The Galerkin approximation of the inductionless MHD problem satisfies the stability estimate

$$A^{\text{lin}}(\boldsymbol{U}_h, \boldsymbol{U}_h) = \nu \|\nabla \boldsymbol{u}_h\|^2 + \frac{1}{\rho\sigma}\|\boldsymbol{j}_h\|^2. \tag{5.21}$$

Remark 5.4. *We have assumed here that $\nabla \cdot \boldsymbol{a} = 0$. This is not necessarily true at the discrete level, where $\boldsymbol{a} = \boldsymbol{u}_h^{n+1,k}$. Technically speaking, we should work with the skew-symmetric expression of the convective term, $(\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \frac{1}{2}(\nabla\cdot\boldsymbol{u})\boldsymbol{u}$. However, the results obtained in the numerical analysis of the stabilized formulation would be the same. Therefore, we will keep working with the assumption $\nabla\cdot\boldsymbol{a} = 0$ for simplicity. We refer to [15] for the technicalities associated to the use of the skew-symmetric form.*

The stability given by estimate (5.21) is not enough to guarantee that the discrete problem is well-posed. Thus, discrete inf-sup conditions between $V_{u,h}$ and $V_{p,h}$ and between $V_{j,h}$ and $V_{\phi,h}$ have to be satisfied. The corresponding discrete inf-sup conditions are

$$\inf_{q_h\in V_{p,h}} \sup_{\boldsymbol{v}_h\in V_{u,h}} \frac{(q_h, \nabla\cdot\boldsymbol{v}_h)}{\|q_h\|\|\nabla\boldsymbol{v}_h\|} \geq \beta^* > 0, \qquad \inf_{\psi_h\in V_{\phi,h}} \sup_{\boldsymbol{k}_h\in V_{j,h}} \frac{(\nabla\psi_h, \boldsymbol{k}_h)}{\|\nabla\psi_h\|\|\boldsymbol{k}_h\|} \geq \gamma^* > 0, \tag{5.22}$$

where β^* and γ^* are positive constants uniform with respect to the mesh size h and different from the constants appearing in the inf-sup conditions for the continuous problem.

The Galerkin FE approximation of this problem faces several well-known difficulties. First, oscillations may appear when dealing with problems where the first order derivatives dominate the second order derivatives in the Navier-Stokes equations. Second, the compatibility conditions being verified at the continuous level do not imply that the discrete versions will also be verified. It depends on the choice of the FE spaces $V_{u,h}, V_{p,h}, V_{j,h}$ and $V_{\phi,h}$. For instance, equal order approximation spaces $V_{u,h}$ and $V_{p,h}$ or $V_{j,h}$ and $V_{\phi,h}$ do not verify the discrete inf-sup conditions. Finally, when the coupling between the hydrodynamical and the electromagnetic problems is strong, the solution of the discrete system of equations may lead to numerical difficulties. The approach taken in this work to face these difficulties is the use of a stabilization method able to deal with all these drawbacks of the Galerkin FE approximation.

5.4 Stabilized formulation and numerical analysis

5.4.1 Stabilized FE approximation for the linearized problem

The basic idea of the stabilization method proposed in this work is based on the subgrid scale concept introduced in [83]. The following ideas are a summary of the approach described in [45]. The main idea is to split the continuous solution of the problem in two components, the FE solution and the *subscales* or *subgrid scales*, which are the part of the solution that cannot be captured by the discretization. In this situation, the problem is reduced to obtain a good approximation for the *subscales*.

There exist several subgrid scale (SGS) stabilization methods. The purpose of this chapter is to see how to apply a well established formulation to the inductionless MHD

problem. This can be obtained by approximating the subscales by the algebraic expression

$$\tilde{\boldsymbol{U}} \approx \boldsymbol{\tau}\tilde{P}[\boldsymbol{F} - \mathcal{L}(\boldsymbol{U}_h)], \tag{5.23}$$

where $\boldsymbol{\tau}$ is a $n_{\text{unk}} \times n_{\text{unk}}$ matrix of stabilization parameters, the expression of which is discussed below, and $\tilde{P}$ is the projection onto the space of subscales. The option taken in this work has been $\tilde{P} = I$, the identity, although it is also possible to take $\tilde{P} = P_h^{\perp}$, the projection orthogonal to the final element space; we refer to [49, 105] for a discussion about the benefits of this last approach. Herein, we have used $\tilde{P} = I$ for simplicity and because it is the most widely used option in the variational multiscale community. Anyway, the statement of the orthogonal subscales method is straightforward.

The discrete problem to be solved is: Find $\boldsymbol{U}_h \in (V_{u,h} \times V_{p,h} \times V_{j,h} \times V_{\phi,h})$ such that

$$A^{\text{lin}}_{\text{stab}}(\boldsymbol{U}_h, \boldsymbol{V}_h) = L_{\text{stab}}(\boldsymbol{V}_h) \quad \forall \boldsymbol{V}_h \in (V_{u,h} \times V_{p,h} \times V_{j,h} \times V_{\phi,h})$$

where

$$A^{\text{lin}}_{\text{stab}}(\boldsymbol{U}_h, \boldsymbol{V}_h) = A^{\text{lin}} - \langle \mathcal{L}^*(\boldsymbol{V}_h), \boldsymbol{\tau}\mathcal{L}(\boldsymbol{U}_h) \rangle_h \,, \tag{5.24}$$

$$L_{\text{stab}}(\boldsymbol{V}_h) = L(\boldsymbol{V}_h) - \langle \mathcal{L}^*(\boldsymbol{V}_h), \boldsymbol{\tau}\boldsymbol{F} \rangle_h \,, \tag{5.25}$$

and where the notation

$$\langle \cdot, \cdot \rangle_h := \sum_{e=1}^{n_{el}} \langle \cdot, \cdot \rangle_{\Omega^e} \,,$$

has been used. The adjoint operator of this problem $\mathcal{L}^*(\boldsymbol{V}_h)$ is given by

$$\mathcal{L}^*(\boldsymbol{V}_h) = \begin{bmatrix} -\nu\Delta\boldsymbol{v}_h - \boldsymbol{a}\cdot\nabla\boldsymbol{v}_h - \nabla q_h + \frac{1}{\rho}(\boldsymbol{k}_h \times \boldsymbol{B}) \\ -\nabla\cdot\boldsymbol{v}_h \\ \frac{1}{\rho\sigma}\boldsymbol{k}_h - \frac{1}{\rho}\nabla\psi_h + \frac{1}{\rho}(\boldsymbol{v}_h \times \boldsymbol{B}) \\ -\frac{1}{\rho}\nabla\cdot\boldsymbol{k}_h \end{bmatrix}. \tag{5.26}$$

The next step is to define an expression for the matrix of stabilization parameters $\boldsymbol{\tau}$. In the case we are considering, we will see in the following subsection that stability can be improved maintaining optimal accuracy by taking a diagonal expression for $\boldsymbol{\tau}$, with one scalar component for each equation. In the 3D case we have

$$\boldsymbol{\tau} = \text{diag}(\tau_1, \tau_1, \tau_1, \tau_2, \tau_3, \tau_3, \tau_3, \tau_4). \tag{5.27}$$

Using both expressions (5.26) and (5.27) in problem (5.24) – (5.25), the stabilized bilinear form is

$$\begin{aligned} A^{\text{lin}}_{\text{stab}}(\boldsymbol{U}_h, \boldsymbol{V}_h) &= A^{\text{lin}}(\boldsymbol{U}_h, \boldsymbol{V}_h) - \langle \mathcal{L}^*(\boldsymbol{V}_h), \boldsymbol{\tau}\mathcal{L}(\boldsymbol{U}_h) \rangle_h \\ &= A^{\text{lin}}(\boldsymbol{U}_h, \boldsymbol{V}_h) + \langle X_u(\boldsymbol{v}_h, q_h, \boldsymbol{k}_h) + \nu\Delta\boldsymbol{v}_h, \tau_1(X_u(\boldsymbol{u}_h, p_h, \boldsymbol{j}_h) - \nu\Delta\boldsymbol{u}_h) \rangle_h \\ &+ \langle \nabla\cdot\boldsymbol{v}_h, \tau_2(\nabla\cdot\boldsymbol{u}_h) \rangle_h + \left\langle X_j(\boldsymbol{v}_h, \psi_h) - \frac{1}{\rho\sigma}\boldsymbol{k}_h, \tau_3 \left(X_j(\boldsymbol{u}_h, \phi_h) + \frac{1}{\rho\sigma}\boldsymbol{j}_h \right) \right\rangle_h \\ &+ \left\langle \frac{1}{\rho}\nabla\cdot\boldsymbol{k}_h, \tau_4 \left(\frac{1}{\rho}\nabla\cdot\boldsymbol{j}_h \right) \right\rangle_h , \end{aligned} \tag{5.28}$$

where we have used the abbreviations

$$\begin{aligned} X_u(\boldsymbol{v}_h, q_h, \boldsymbol{k}_h) &:= \boldsymbol{a}\cdot\nabla\boldsymbol{v}_h + \nabla q_h - \frac{1}{\rho}(\boldsymbol{k}_h\times\boldsymbol{B}), \\ X_j(\boldsymbol{v}_h, \psi_h) &:= \frac{1}{\rho}\nabla\psi_h - \frac{1}{\rho}(\boldsymbol{v}_h\times\boldsymbol{B}). \end{aligned}$$

The right-hand-side of the stabilized problem is given by

$$\begin{aligned} L_{\text{stab}}(\boldsymbol{V}_h) =& L(\boldsymbol{V}_h) - \langle\mathcal{L}^*(\boldsymbol{V}_h), \boldsymbol{\tau F})\rangle_h \\ =& L(\boldsymbol{V}_h) + \langle X_u(\boldsymbol{v}_h, q_h, \boldsymbol{k}_h) + \nu\Delta\boldsymbol{v}_h, \tau_1\boldsymbol{f}\rangle_h . \end{aligned}$$

The definition of the stabilized FE method only misses the expression of the stabilization parameters. The expressions proposed in this work are

$$\begin{aligned} &\alpha := c_1\frac{a}{h} + c_2\frac{\nu}{h^2}, \quad \beta := c_3\frac{B}{\rho}, \quad \gamma := c_4\frac{1}{\rho\sigma}, \\ &\tau_1 = \alpha^{-1}\left(1 + \frac{1}{\sqrt{\alpha\gamma}}\beta\right)^{-1}, \quad \tau_2 = c_5\frac{h^2}{\tau_1}, \\ &\tau_3 = \gamma^{-1}\left(1 + \frac{1}{\sqrt{\alpha\gamma}}\beta\right)^{-1}, \quad \tau_4 = c_6\frac{\rho^2h^2}{\tau_3}. \end{aligned} \tag{5.29}$$

These expressions are evaluated element by element. Here, a is the maximum norm of the velocity field $\boldsymbol{a}$ computed in the element under consideration. Likewise, B is the maximum norm of the magnetic field $\boldsymbol{B}$ in the corresponding element, and h the element diameter.

The stabilization parameters have been developed for the steady problem. For the transient problem, we consider the stabilized formulation

$$M(\delta_t\boldsymbol{U}_h, \boldsymbol{V}_h) + A^{\text{lin}}_{\text{stab}}(\boldsymbol{U}_h, \boldsymbol{V}_h) = L_{\text{stab}}(\boldsymbol{V}_h) \quad \forall\boldsymbol{V}_h \in (V_{u,h}\times V_{p,h}\times V_{j,h}\times V_{\phi,h}),$$

instead of (5.20). Therefore, the stabilization parameters are the same as those of the steady problem and do not depend on the time step size.[1] Alternatively, in order to take into account the time behavior of the subscale, we could consider *dynamic* subscales (see [55]). For the sake of conciseness, we have not included this option here, but it is straightforward from the quasi-static formulation above and [55].

Note that if $-\mathcal{L}^*(\boldsymbol{V}_h)$ is replaced by $\mathcal{L}(\boldsymbol{V}_h)$ (which amounts for a change in two signs), a GLS formulation of the inductionless MHD problem is recovered [85].

5.4.2 Numerical analysis and justification of the stabilization parameters

In this subsection we proceed with the numerical analysis of the formulation introduced before that will justify the stabilization parameter expression (5.29). For the sake of

[1] This kind of stabilized transient formulation is the *quasi-static* subscale approach in [55]. Therein, we have justified why time step size dependent stabilization parameters should be avoided.

simplicity we assume that $\boldsymbol{a}$ and $\boldsymbol{B}$ are constant and that the FE meshes are quasi-uniform. Thus, we can consider a characteristic mesh size h in the definition of the stabilization parameters and therefore $\tau_i, i = 1, 2, 3, 4$ are constant. Moreover, for quasi-uniform meshes the following inverse estimates hold

$$\|\nabla v_h\| \leq \frac{C_{\text{inv}}}{h}\|v_h\|, \quad \|\nabla\nabla v_h\| \leq \frac{C_{\text{inv}}}{h}\|\nabla v_h\|, \tag{5.30}$$

for any function v_h in the FE space and for a certain constant C_{inv}.

The stability and convergence analysis will be made using the mesh-dependent norm

$$\begin{aligned}
|||\boldsymbol{U}_h|||^2 := &\nu\|\nabla\boldsymbol{u}_h\|^2 + \frac{1}{\rho\sigma}\|\boldsymbol{j}_h\|^2 + \tau_1\|\boldsymbol{a}\cdot\nabla\boldsymbol{u}_h + \nabla p_h - \frac{1}{\rho}(\boldsymbol{j}_h\times\boldsymbol{B})\|^2 \\
&+ \tau_2\|\nabla\cdot\boldsymbol{u}_h\|^2 + \tau_3\|\frac{1}{\rho}\nabla\phi_h - \frac{1}{\rho}(\boldsymbol{u}_h\times\boldsymbol{B})\|^2 + \tau_4\frac{1}{\rho^2}\|\nabla\cdot\boldsymbol{j}_h\|^2 \\
= &\nu\|\nabla\boldsymbol{u}_h\|^2 + \frac{1}{\rho\sigma}\|\boldsymbol{j}_h\|^2 + \tau_1\|X_u(\boldsymbol{u}_h, p_h, \boldsymbol{j}_h)\|^2 \\
&+ \tau_2\|\nabla\cdot\boldsymbol{u}_h\|^2 + \tau_3\|X_j(\boldsymbol{u}_h,\phi_h)\|^2 + \tau_4\frac{1}{\rho^2}\|\nabla\cdot\boldsymbol{j}_h\|^2.
\end{aligned} \tag{5.31}$$

From now on, C will denote a positive constant independent of the mesh discretization and the physical parameters, not necessarily the same at different stages.

Coercivity

Let us start by proving stability in the form of coercivity of the bilinear form (5.28):

$$\begin{aligned}
A^{\text{lin}}_{\text{stab}}(\boldsymbol{U}_h, \boldsymbol{U}_h) &= A^{\text{lin}}(\boldsymbol{U}_h, \boldsymbol{U}_h) - \langle\mathcal{L}^*(\boldsymbol{U}_h), \boldsymbol{\tau}\mathcal{L}(\boldsymbol{U}_h)\rangle_h \\
&= \nu\|\nabla\boldsymbol{u}_h\|^2 + \frac{1}{\rho\sigma}\|\boldsymbol{j}_h\|^2 + \tau_1\|X_u(\boldsymbol{u}_h, p_h, \boldsymbol{j}_h)\|^2 - \tau_1\nu^2\|\Delta\boldsymbol{u}_h\|^2 \\
&\quad + \tau_2\|\nabla\cdot\boldsymbol{u}_h\|^2 + \tau_3\|X_j(\boldsymbol{u}_h,\phi_h)\|^2 - \tau_3\frac{1}{\rho^2\sigma^2}\|\boldsymbol{j}_h\|^2 + \tau_4\frac{1}{\rho^2}\|\nabla\cdot\boldsymbol{j}_h\|^2.
\end{aligned}$$

Using the second inverse estimate in (5.30), a sufficient condition for $A^{\text{lin}}_{\text{stab}}$ to be coercive is

$$\nu - \tau_1\nu^2\frac{C^2_{\text{inv}}}{h^2} \geq \alpha\nu \Longleftrightarrow \tau_1 \leq (1-\alpha)\frac{1}{\nu}\frac{h^2}{C^2_{\text{inv}}}, \tag{5.32}$$

$$\frac{1}{\rho\sigma} - \tau_3\frac{1}{\rho^2\sigma^2} \geq \alpha\frac{1}{\rho\sigma} \Longleftrightarrow \tau_3 \leq (1-\alpha)\rho\sigma, \tag{5.33}$$

with $0 < \alpha < 1$. Conditions (5.32)-(5.33) imply

$$A^{\text{lin}}_{\text{stab}}(\boldsymbol{U}_h, \boldsymbol{U}_h) \geq C|||\boldsymbol{U}_h|||^2,$$

for a constant C independent of the discretization and of the physical parameters.

Optimal accuracy

The requirement that the stabilized formulation is optimally accurate will allow us to obtain new conditions on the stabilization parameters. These new conditions together with (5.32)-(5.33) from stability will lead to the final expression of the stabilization parameters.

For a function v, let $\pi_h(v)$ be its optimal FE approximation. We assume that the following estimates hold

$$\|v - \pi_h(v)\|_{H^i(\Omega)} \leq \varepsilon_i(v) := Ch^{k+1-i}|v|_{H^{k+1}(\Omega)}, \quad i = 0, 1, \tag{5.34}$$

where $\|v\|_{H^q(\Omega}$ is the $H^q(\Omega)$-norm of v, that is, the sum of the $L^2(\Omega)$-norm of the derivatives of v up to degree q, $|v|_{H^q(\Omega)}$ the corresponding semi-norm, and k the degree of the FE approximation.

We will prove next that the interpolation error function of the formulation is

$$E(h) := \tau_1^{-1/2}\varepsilon_0(\boldsymbol{u}) + \tau_2^{-1/2}\varepsilon_0(p) + \tau_3^{-1/2}\varepsilon_0(\boldsymbol{j}) + \tau_4^{-1/2}\varepsilon_0(\phi).$$

Let $\boldsymbol{U}$ be the solution of the continuous problem and $\pi_h(\boldsymbol{U})$ its optimal FE approximation. The accuracy estimate that will be needed to prove convergence is

$$A^{\text{lin}}_{\text{stab}}(\boldsymbol{U} - \pi_h(\boldsymbol{U}), \boldsymbol{V}_h) \leq CE(h)|||\boldsymbol{V}_h|||, \tag{5.35}$$

for any FE function $\boldsymbol{V}_h$.

Let us prove this by showing that both the Galerkin and the stabilization terms in $A^{\text{lin}}_{\text{stab}}$ satisfy estimate (5.35) for sufficiently smooth solutions of the continuous problem. Integrating by parts some terms in the Galerkin contribution we obtain

$$\begin{aligned}
A^{\text{lin}}(\boldsymbol{U} - \pi_h(\boldsymbol{U}), \boldsymbol{V}_h) &= \nu(\nabla(\boldsymbol{u} - \pi_h(\boldsymbol{u})), \nabla\boldsymbol{v}_h) - (\boldsymbol{u} - \pi_h(\boldsymbol{u}), \boldsymbol{a}\cdot\nabla\boldsymbol{v}_h) - (\boldsymbol{u} - \pi_h(\boldsymbol{u}), \nabla q_h) \\
&\quad - (p - \pi_h(p), \nabla\cdot\boldsymbol{v}_h) + \frac{1}{\rho}(\boldsymbol{u} - \pi_h(\boldsymbol{u}), \boldsymbol{k}_h \times \boldsymbol{B}) \\
&\quad + \frac{1}{\rho}(\boldsymbol{j} - \pi_h(\boldsymbol{j}), \boldsymbol{v}_h \times \boldsymbol{B}) + \frac{1}{\rho\sigma}(\boldsymbol{j} - \pi_h(\boldsymbol{j}), \boldsymbol{k}_h) \\
&\quad - \frac{1}{\rho}(\boldsymbol{j} - \pi_h(\boldsymbol{j}), \nabla\psi_h) - \frac{1}{\rho}(\phi - \pi_h(\phi), \nabla\cdot\boldsymbol{k}_h) \\
&\leq C\Big(\varepsilon_0(\boldsymbol{u})\tau_1^{-1/2}\tau_1^{1/2}\|X_u(\boldsymbol{v}_h, q_h, \boldsymbol{k}_h)\| + \nu^{1/2}\varepsilon_1(\boldsymbol{u})\nu^{1/2}\|\nabla\boldsymbol{v}_h\| \\
&\quad + \varepsilon_0(p)\tau_2^{-1/2}\tau_2^{1/2}\|\nabla\cdot\boldsymbol{v}_h\| \\
&\quad + \varepsilon_0(\boldsymbol{j})\tau_3^{-1/2}\tau_3^{1/2}\left[\|X_j(\boldsymbol{v}_h, \psi_h)\| + \frac{1}{\rho\sigma}\|\boldsymbol{k}_h\|\right] \\
&\quad + \varepsilon_0(\phi)\tau_4^{-1/2}\tau_4^{1/2}\frac{1}{\rho}\|\nabla\cdot\boldsymbol{k}_h\|\Big).
\end{aligned} \tag{5.36}$$

Conditions (5.32)-(5.33) and the expression of the interpolation errors imply

$$\nu^{1/2}\varepsilon_1(\boldsymbol{u}) \leq C\varepsilon_0(\boldsymbol{u})\tau_1^{-1/2}, \quad \frac{1}{(\rho\sigma)^{1/2}}\|\boldsymbol{k}_h\| \leq \tau_3^{-1/2}\|\boldsymbol{k}_h\|,$$

and therefore from (5.36) it follows that the Galerkin contribution to $A^{\rm lin}_{\rm stab}(\boldsymbol{U} - \pi_h(\boldsymbol{U}), \boldsymbol{V}_h)$ can be bounded as indicated in (5.35). It remains to prove that also the stabilization terms can be bounded the same way:

$$
\begin{aligned}
&- \langle \mathcal{L}^*(\boldsymbol{V}_h), \boldsymbol{\tau}\mathcal{L}(\boldsymbol{U} - \pi_h(\boldsymbol{U}))\rangle_h \\
&\quad = \langle X_u(\boldsymbol{v}_h, q_h, \boldsymbol{k}_h) + \nu\Delta\boldsymbol{v}_h, \tau_1(X_u(\boldsymbol{u} - \pi_h(\boldsymbol{u}), p - \pi_h(p), \boldsymbol{j} - \pi_h(\boldsymbol{j})) - \nu\Delta(\boldsymbol{u} - \pi_h(\boldsymbol{u})))\rangle_h \\
&\qquad + \langle \nabla\cdot\boldsymbol{v}_h, \tau_2\nabla\cdot(\boldsymbol{u} - \pi_h(\boldsymbol{u}))\rangle_h + \left\langle \frac{1}{\rho}\nabla\cdot\boldsymbol{k}_h, \tau_4\frac{1}{\rho}\nabla\cdot(\boldsymbol{j} - \pi_h(\boldsymbol{j}))\right\rangle_h \\
&\qquad + \left\langle X_j(\boldsymbol{v}_h, \psi_h) - \frac{1}{\rho\sigma}\boldsymbol{k}_h, \tau_3(X_j(\boldsymbol{u} - \pi_h(\boldsymbol{u}), \phi - \pi_h(\phi)) + \frac{1}{\rho\sigma}(\boldsymbol{j} - \pi_h(\boldsymbol{j})))\right\rangle_h \\
&\quad \le C\Big(\tau_1^{1/2}\|X_u(\boldsymbol{u} - \pi_h(\boldsymbol{u}), p - \pi_h(p), \boldsymbol{j} - \pi_h(\boldsymbol{j}))\| + \tau_1^{1/2}\nu\|\Delta(\boldsymbol{u} - \pi_h(\boldsymbol{u}))\|\Big) \\
&\qquad\quad \times \Big(|||\boldsymbol{V}_h||| + \tau_1^{1/2}\nu\|\Delta\boldsymbol{v}_h\|\Big) \\
&\qquad + C\tau_2^{1/2}\varepsilon_1(\boldsymbol{u})|||\boldsymbol{V}_h||| + C\tau_4^{1/2}\frac{1}{\rho}\varepsilon_1(\boldsymbol{j})|||\boldsymbol{V}_h||| \\
&\qquad + C\Big(\tau_3^{1/2}\|X_j(\boldsymbol{u} - \pi_h(\boldsymbol{u}), \phi - \pi_h(\phi))\| + \tau_3^{1/2}\frac{1}{\rho\sigma}(\boldsymbol{j} - \pi_h(\boldsymbol{j}))\Big) \times \Big(|||\boldsymbol{V}_h||| + \tau_3^{1/2}\frac{1}{\rho\sigma}\|\boldsymbol{k}_h\|\Big).
\end{aligned}
\tag{5.37}
$$

Using again conditions (5.32)-(5.33) and the inverse estimates (5.30) we have

$$
\begin{aligned}
\tau_1^{1/2}\nu\|\Delta\boldsymbol{v}_h\| &\le C\tau_1^{1/2}\nu^{1/2}\frac{C_{\rm inv}}{h}\|\nabla\boldsymbol{v}_h\| \le C|||\boldsymbol{V}_h|||, \\
\tau_3^{1/2}\frac{1}{\rho\sigma}\|\boldsymbol{k}_h\| &\le C(\rho\sigma)^{1/2}\frac{1}{\rho\sigma}\|\boldsymbol{k}_h\| \le C|||\boldsymbol{V}_h|||.
\end{aligned}
$$

Therefore, we get from (5.37) that

$$
\begin{aligned}
&- \langle \mathcal{L}^*(\boldsymbol{V}_h), \boldsymbol{\tau}\mathcal{L}(\boldsymbol{U} - \pi_h(\boldsymbol{U}))\rangle_h \\
&\quad \le C|||\boldsymbol{V}_h|||\left[\tau_1^{1/2}\Big(\frac{\nu}{h^2}\varepsilon_0(\boldsymbol{u}) + \frac{a}{h}\varepsilon_0(\boldsymbol{u}) + \frac{1}{h}\varepsilon_0(p) + \frac{B}{\rho}\varepsilon_0(\boldsymbol{j})\Big)\right. \\
&\qquad \left. + \tau_2^{1/2}\Big(\frac{1}{h}\varepsilon_0(\boldsymbol{u})\Big) + \tau_3^{1/2}\Big(\frac{1}{\rho\sigma}\varepsilon_0(\boldsymbol{j}) + \frac{1}{\rho h}\varepsilon_0(\phi) + \frac{B}{\rho}\varepsilon_0(\boldsymbol{u})\Big) + \tau_4^{1/2}\Big(\frac{1}{\rho h}\varepsilon_0(\boldsymbol{j})\Big)\right] \\
&\quad \le C|||\boldsymbol{V}_h|||\left[\varepsilon_0(\boldsymbol{u})\Big[\tau_1^{1/2}\Big(\frac{\nu}{h^2} + \frac{a}{h}\Big) + \tau_2^{1/2}\frac{1}{h} + \tau_3^{1/2}\frac{B}{\rho}\Big]\right. \\
&\qquad\qquad + \varepsilon_0(p)\Big[\tau_1^{1/2}\frac{1}{h}\Big] \\
&\qquad\qquad + \varepsilon_0(\boldsymbol{j})\Big[\tau_1^{1/2}\frac{B}{\rho} + \tau_3^{1/2}\frac{1}{\rho\sigma} + \tau_4^{1/2}\frac{1}{\rho h}\Big] \\
&\qquad\qquad \left. + \varepsilon_0(\phi)\Big[\tau_3^{1/2}\frac{1}{\rho h}\Big]\right].
\end{aligned}
$$

Using the definition (5.29) of the stabilization parameters it is easily checked that these terms can also be bounded as indicated in (5.35).

Remark 5.5. *The last step provides the crucial design condition for the stabilization parameters. Expressions* (5.29) *result from solving*

$$\tau_1^{1/2}\Big(\frac{\nu}{h^2}+\frac{a}{h}\Big)+\tau_2^{1/2}\frac{1}{h}+\tau_3^{1/2}\frac{B}{\rho}\sim\tau_1^{-1/2}, \tag{5.38}$$

$$\tau_1^{1/2}\frac{1}{h}\sim\tau_2^{-1/2}, \tag{5.39}$$

$$\tau_1^{1/2}\frac{B}{\rho}+\tau_3^{1/2}\frac{1}{\rho\sigma}+\tau_4^{1/2}\frac{1}{\rho h}\sim\tau_3^{-1/2}, \tag{5.40}$$

$$\tau_3^{1/2}\frac{1}{\rho h}\sim\tau_4^{-1/2}, \tag{5.41}$$

where $\sim$ *stands for equality up to constants that do not depend on the physical variables nor the mesh discretization.*

Convergence

The properties of stability and optimal accuracy, in the sense of (5.35) allow us to show that the method is optimally convergent. From the orthogonality property $A^{\text{lin}}_{\text{stab}}(\boldsymbol{U}-\boldsymbol{U}_h,\boldsymbol{V}_h)=0$ for any FE function $\boldsymbol{V}_h$, a consequence of the consistency of the method, we have that

$$\begin{aligned}C|||\pi_h(\boldsymbol{U})-\boldsymbol{U}_h|||^2 &\le A^{\text{lin}}_{\text{stab}}(\pi_h(\boldsymbol{U})-\boldsymbol{U}_h,\pi_h(\boldsymbol{U})-\boldsymbol{U}_h)\\ &\le A^{\text{lin}}_{\text{stab}}(\pi_h(\boldsymbol{U})-\boldsymbol{U},\pi_h(\boldsymbol{U})-\boldsymbol{U}_h)+A^{\text{lin}}_{\text{stab}}(\boldsymbol{U}-\boldsymbol{U}_h,\pi_h(\boldsymbol{U})-\boldsymbol{U}_h)\\ &\le CE(h)|||\pi_h(\boldsymbol{U})-\boldsymbol{U}_h|||,\end{aligned}$$

and so $|||\pi_h(\boldsymbol{U})-\boldsymbol{U}_h|||\le CE(h)$. If we apply the triangle inequality, we get

$$\begin{aligned}|||\boldsymbol{U}-\boldsymbol{U}_h||| &\le |||\boldsymbol{U}-\pi_h(\boldsymbol{U})|||+|||\pi_h(\boldsymbol{U})-\boldsymbol{U}_h|||\\ &\le |||\boldsymbol{U}-\pi_h(\boldsymbol{U})|||+CE(h).\end{aligned}$$

It is trivial to check that $|||\boldsymbol{U}-\pi_h(\boldsymbol{U})|||\le CE(h)$ using the expression of the norm (5.31), the interpolation estimates (5.34) and the stabilization parameters (5.29). Therefore,

$$|||\boldsymbol{U}-\boldsymbol{U}_h|||\le CE(h).$$

The fact that this error estimate is exactly the same as the estimate for the interpolation error $|||\boldsymbol{U}-\pi_h(\boldsymbol{U})|||\le CE(h)$ justifies why it can be considered optimal.

5.5 Final numerical scheme

The final numerical scheme proposed to solve the inductionless MHD problem results from applying the stabilized FE approximation described in Section 5.4.1 to the time discrete and linearized problem (5.13)-(5.16). Therefore, the final algorithm reads:

For $n = 0, 1, 2, ..., T/\delta t$ and given $\boldsymbol{u}^n$, find $\boldsymbol{u}^{n+1}$, p^{n+1}, $\boldsymbol{j}^{n+1}$ and ϕ^{n+1} as the converged solutions of the following iterative algorithm:

$$
\begin{aligned}
&(\delta_t \boldsymbol{u}_h^{n,k+1}, \boldsymbol{v}_h) + \left\langle (\boldsymbol{u}_h^{n+1,k} \cdot \nabla)\boldsymbol{u}_h^{n+1,k+1}, \boldsymbol{v}_h \right\rangle + \nu(\nabla \boldsymbol{u}_h^{n+1,k+1}, \nabla \boldsymbol{v}_h) - (p_h^{n+1,k+1}, \nabla \cdot \boldsymbol{v}_h) \\
&\quad - \frac{1}{\rho}\left\langle \boldsymbol{j}_h^{n+1,k+1} \times \boldsymbol{B}, \boldsymbol{v}_h \right\rangle + \left\langle \boldsymbol{u}_h^{n+1,k} \cdot \nabla \boldsymbol{v}_h + \nu \Delta \boldsymbol{v}_h, \tau_1^{n+1,k} \boldsymbol{R}_{h,u}^{n+1,k+1} \right\rangle_h \\
&\quad + \left\langle \nabla \cdot \boldsymbol{v}_h, \tau_2^{n+1,k} R_{h,p}^{n+1,k+1} \right\rangle_h - \left\langle \frac{1}{\rho}(\boldsymbol{v}_h \times \boldsymbol{B}), \tau_3^{n+1,k} \boldsymbol{R}_{h,j}^{n+1,k+1} \right\rangle_h = \left\langle \boldsymbol{f}^{n+1}, \boldsymbol{v}_h \right\rangle, \\
&(q_h, \nabla \cdot \boldsymbol{u}_h^{n+1,k+1}) + \left\langle \nabla q_h, \tau_1^{n+1,k} \boldsymbol{R}_{h,u}^{n+1,k+1} \right\rangle_h = 0, \\
&\frac{1}{\rho\sigma}(\boldsymbol{j}_h^{n+1,k+1}, \boldsymbol{k}_h) + \frac{1}{\rho}(\nabla \phi_h^{n+1,k+1}, \boldsymbol{k}_h) - \frac{1}{\rho}(\boldsymbol{u}_h^{n+1,k+1} \times \boldsymbol{B}, \boldsymbol{k}_h) \\
&\quad - \left\langle \frac{1}{\rho}(\boldsymbol{k}_h \times \boldsymbol{B}), \tau_1^{n+1,k} \boldsymbol{R}_{h,u}^{n+1,k+1} \right\rangle_h - \left\langle \frac{1}{\rho\sigma}\boldsymbol{k}_h, \tau_3^{n+1,k} \boldsymbol{R}_{h,j}^{n+1,k+1} \right\rangle_h \\
&\quad + \left\langle \frac{1}{\rho}\nabla \cdot \boldsymbol{k}_h, \tau_4^{n+1,k} R_{h,\phi}^{n+1,k+1} \right\rangle_h = 0, \\
&-\frac{1}{\rho}(\nabla \psi_h, \boldsymbol{j}_h^{n+1,k+1}) + \left\langle \frac{1}{\rho}\nabla \psi_h, \tau_3^{n+1,k} \boldsymbol{R}_{h,j}^{n+1,k+1} \right\rangle_h = 0,
\end{aligned}
$$

where the expression of the residuals is,

$$
\begin{aligned}
\boldsymbol{R}_{h,u} &:= \delta_t \boldsymbol{u}_h + \boldsymbol{a} \cdot \nabla \boldsymbol{u}_h - \nu \Delta \boldsymbol{u}_h + \nabla p_h - \frac{1}{\rho}(\boldsymbol{j}_h \times \boldsymbol{B}) - \boldsymbol{f}, \\
R_{h,p} &:= \nabla \cdot \boldsymbol{u}_h, \\
\boldsymbol{R}_{h,j} &:= \frac{1}{\rho\sigma}\boldsymbol{j}_h + \frac{1}{\rho}\nabla \phi_h - \frac{1}{\rho}(\boldsymbol{u}_h \times \boldsymbol{B}), \\
R_{h,\phi} &:= \frac{1}{\rho}\nabla \cdot \boldsymbol{j}_h,
\end{aligned}
$$

with $\boldsymbol{a} = \boldsymbol{u}_h^{n+1,k}$. The superscript in the residuals and the stabilization parameters denotes the unknown with which they are evaluated.

5.6 Numerical experimentation

5.6.1 Comparison between monolithic solvers and uncoupling schemes

There exist several strategies to solve the linear system of equations resulting from the final numerical scheme written in Section 5.5. On one hand, the problem can be stated in a monolithic way, leading to a linear system of equations that includes all the problem unknowns. We can state the problem in an algebraic setting as:

$$
\begin{pmatrix} A_{uu} & A_{uj} \\ A_{ju} & A_{jj} \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ \mathbf{j} \end{pmatrix} = \begin{pmatrix} \mathbf{f}_u \\ \mathbf{f}_j \end{pmatrix}
$$

where the arrays **u** and **j** include the fluid and electromagnetic unknowns respectively. Using this splitting of the unknowns, we have written the system matrix and force vector in a block fashion. So, the block matrices A_{uj} and A_{ju} represent the coupling terms. In a compact form, the problem can simply be written as $A\mathbf{x} = \mathbf{f}$.

The coupled linear system can be solved with our preferred solver and preconditioner. A flexible and quite robust preconditioner P consists of an incomplete LU (ILU) factorization of the system matrix, in one of its multiple versions (see, e.g., [109]). So, e.g. the left-preconditioned system reads as $P^{-1}A\mathbf{x} = P^{-1}\mathbf{f}$. Since we are dealing with a non-symmetric matrix A, the GMRES Krylov iterative solver is a good choice. Therefore, the coupling between subproblems is transferred to an effective solver and the off-diagonal coupling matrices are also present in the preconditioner P. Other effective preconditioners for saddle-point problems are block preconditioners based on Schur complement approximations together with multigrid iterations; we refer to [67] for a detailed discussion in the frame of Stokes and Navier-Stokes problems.

On the other hand, there exists also the option to consider a segregated approach to the problem, i.e. sending the coupling terms to the right hand side and considering separated fluid and electromagnetic solvers. In this case, the coupling is performed via external iterations. This approach is nothing but a block-matrix splitting technique with stationary iterations. Let us consider the splitting $A = P - R$, where P is the preconditioner and R the residual matrix. Stationary (Richardson) iterations read as

$$P\mathbf{x}^{k+1} = R\mathbf{x}^k + \mathbf{f}, \qquad \text{or equivalently} \qquad \mathbf{x}^{k+1} = \mathbf{x}^k + P^{-1}(\mathbf{f} - A\mathbf{x}^k).$$

Two typical preconditioners that decouple fluid and electromagnetic computations at the preconditioner level are the block-Jacobi (bJ) and block-Gauss-Seidel (bGS) preconditioners:

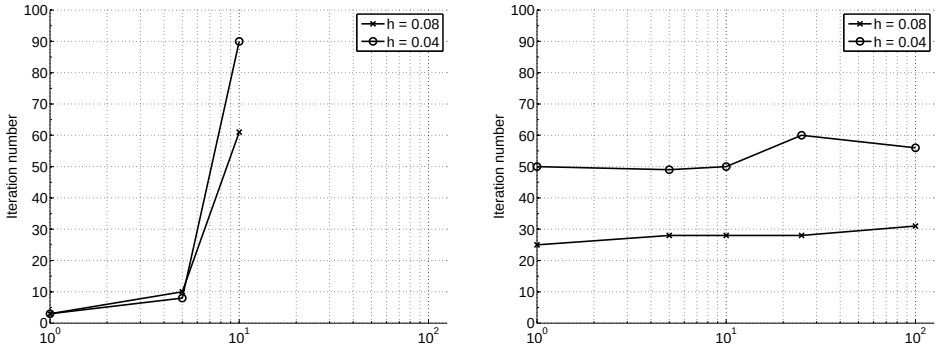
$$P_{\text{bJ}} = \begin{pmatrix} A_{uu} & 0 \\ 0 & A_{jj} \end{pmatrix} \qquad \text{and} \qquad P_{\text{bGS}} = \begin{pmatrix} A_{uu} & 0 \\ A_{ju} & A_{jj} \end{pmatrix}.$$

This *segregated* approach has two weak points: the preconditioner is independent from the coupling terms and the coupling iterations do not involve any orthogonalization (minimization) procedure. So, the convergence of the method is expected to deteriorate as the coupling becomes more important. It is well-known in other settings that this methodology is ill-posed for strongly coupled problems. As long as the coupling terms increase, the convergence becomes slower or it simply diverges (see e.g. [20, 39, 115] for detailed discussions in the fluid-structure framework).

These two different approaches to solve the coupled problem have been compared for the Hunt's example; see Section 5.6.3 for a complete definition of the problem. In this study, two meshes, the coarsest one consisting of 2028 nodes and 7500 linear tetrahedral elements and the finest one with 7803 nodes and 30000 linear tetrahedral elements, have been used. The problem has been solved for different values of the Hartmann number Ha = 1, 5, 10, 25, 100, where

$$\text{Ha} = BL\sqrt{\frac{\sigma}{\rho\nu}},$$

L being a characteristic length of the problem and B the norm of the externally applied magnetic field. Larger values of Ha mean stronger coupling effects.



(a) Richardson-bGS (b) ILUTP-GMRES

Figure 5.1: Number of iterations depending on the Hartmann number Ha.

The method selected to solve the problem in a monolithic way is the GMRES method, preconditioning the matrix of the system using an ILU factorization; the built-in MATLAB implementation of both schemes has been used. In particular, we have used the ILUTP factorization, setting the drop tolerance to 10^{-4}. The GMRES residual tolerance has been set to 10^{-8}. For the Richardson iterations, we have considered the bGS preconditioner; the stopping criteria is based on the magnitude of the residual, with a tolerance of 10^{-8}.

Figure 5.1 shows the number of iterations needed to achieve a converged solution in terms of the corresponding tolerances for both approaches and both discretizations. It is very clear in Figure 5.1(a) that the Richardson-bGS approach is very sensitive to the magnitude of the coupling. When the coupling effects are low, for Ha=1, 5, the Richardson method converges quickly to the coupled solution. However, when the coupling is stronger, for Ha=10, the number of iterations is much larger; the algorithm is not able to converge for Ha= 25, 100. On the other hand, the monolithic approach to solve the coupled problem has a much better behavior. The ILUTP-GMRES method is insensitive to the Ha number. Let us remark that the number of iterations presented for the Richardson-bGS method corresponds to external (coupling) iterations; we are not including the number of internal Krylov iterations needed for the evaluation of every subproblem. On the contrary, ILUTP-GMRES only include one iteration counter, and so, the iterations showed in Figure 5.1(b) are the only iterations to be performed.

Despite these results, previous approaches to the inductionless MHD problem systematically used the $\boldsymbol{u}-\phi$ formulation which uncouples the hydrodynamic and magnetic problems and solves the electric potential via a Poisson problem. Therefore, this solving strategy involves Richardson-bGS or Richardson-bJ iterations, probably together with relaxation or line search techniques (see [37, 95, 101, 102]).[2] This approach is effective for low Ha, but inappropriate for large Ha numbers, as those encountered in TBMs simulations. As a result, TBM simulations cannot be properly addressed when using $\boldsymbol{u}-\phi$

[2] Alternatively, this approach can be casted in a transient framework, in which the coupling is treated explicitly; in this situation, the convergence problem related to strong coupling is passed to the time step size. Stable time marching schemes with explicit coupling will require time step sizes that go to zero as Ha increases.

formulations. These results justify our approach to the inductionless MHD problem. Since real applications in fusion reaction technology involve Ha numbers of the order of $10^3 - 10^4$, a monolithic approach should be clearly favoured. Furthermore, as far as we know, there are no compatible finite element formulations for both sub-problems (Stokes and Darcy type sub-systems), that is to say, elements that satisfy both inf-sup conditions (5.22) such that the bilinear forms associated to the primal variable are coercive in the kernel of the finite element subspaces for Stokes' and Darcy's problem (see [11] for a detailed discussion). So, our stabilized formulation is appealing, in the sense that it allows equal interpolation for the different unknowns (simplifying data-base structures, coupling terms implementation and reducing CPU cost) and the use of effective solvers for high Ha numbers.

5.6.2 Shercliff's case

The first numerical experiment that has been carried out is the simulation of the Shercliff's case. It corresponds to a fully developed flow in a channel with square section where both the Hartmann walls, which are the walls orthogonal to the external magnetic field direction, and the side walls, which are the walls parallel to the external magnetic field, are considered electrically insulating. The fluid flows with unidirectional velocity in the z-direction driven by a constant pressure gradient. The channel is exposed to an external magnetic field applied in the y-direction. This problem has an analytical solution in form of Fourier series that was developed by J. A. Shercliff [113]. A more appropriate version of this solution for the implementation in a computer can be found in [102] [3]. The formulae used in this work to compute the analytical solution and compare with the numerical approximation are explained in Annex 5.A.

This problem has a 2D behavior that has been simulated setting as the computational domain a slice of the channel of width 1/100 times the section sides. The boundary conditions at the inflow and outflow sections have been set as periodic boundary conditions to enforce the situation of fully developed flow. Therefore, the constant pressure gradient that drives the liquid has to be set as an external body force. Its value can be computed as (see [98] for details)

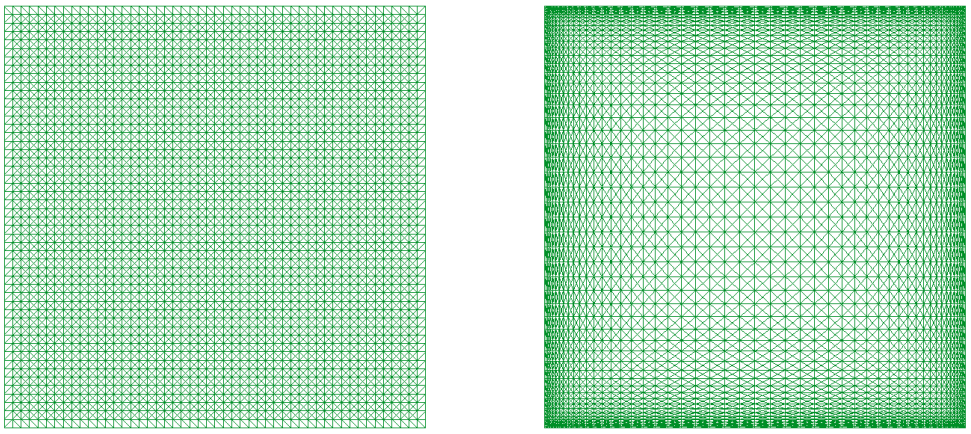
$$\frac{\mathrm{d}p}{\mathrm{d}z} = \frac{KL^3}{\rho\nu^2 \mathrm{Re}}$$

with

$$K = \frac{\mathrm{Ha}}{1 - 0.825\mathrm{Ha}^{-1/2} - \mathrm{Ha}^{-1}}, \qquad \mathrm{Re} = \frac{UL}{\nu},$$

where U is a characteristic velocity of the fluid. Every physical property of the problem, that is, density, viscosity and electrical conductivity has been set equal to one. In this way, the Hartmann number Ha is equal to the norm of the external magnetic field. Several meshes have been used to perform the computations. The coarsest consists of 2028 nodes and 7500 tetrahedral elements whereas the finest consists of 121203 nodes and 480000 tetrahedral elements. Furthermore, two different configurations of meshes have been considered, a uniformly structured one and a structured one but concentrating the

[3] There are some typographical errors in two of the formulae in [102].



(a) Uniformly structured mesh

(b) Structured mesh but concentrating elements near the boundaries

Figure 5.2: Mesh configurations.

elements near the boundaries. Figure 5.2 shows the two different configurations for a mesh of 30000 elements.

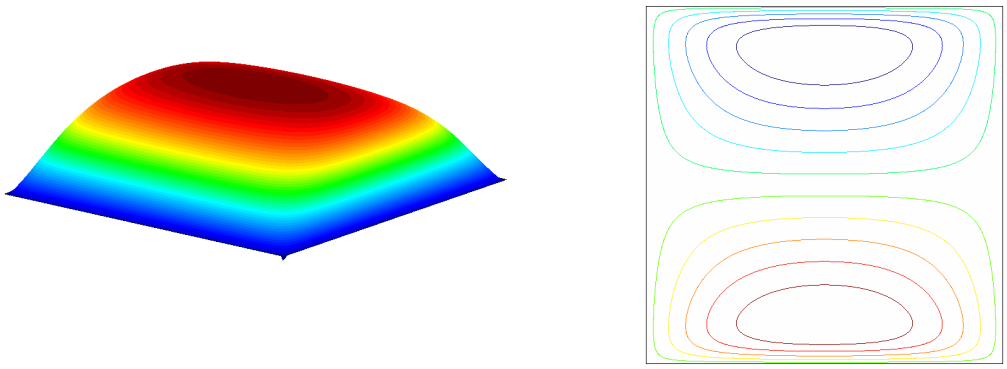
The first simulation has been performed for Ha=10 and Re=10. Figure 5.3 shows the velocity field and the current paths obtained using a mesh of 30603 nodes and 120000 tetrahedral elements.

The second simulation is for a test problem with Ha=100 and Re=10. In this case, the uniformly structured meshes do not lead to a proper solution because the Hartmann layer is much thinner than the mesh size h. Therefore, this case has been solved with meshes concentrating elements near the boundaries. The results for a mesh of 30603 nodes and 120000 tetrahedral elements are shown in Figure 5.4.

Figure 5.5 shows the convergence study of both Ha=10 and Ha=100 cases depending on the mesh size h in a logarithmic scale. Note that the mesh size for the meshes with element concentration is not constant. Therefore, the results have been plotted related to an equivalent mesh size h^* which corresponds to the same number of degrees of freedom than a uniformly structured mesh. The values shown in this study correspond to the L^2-norm of the error in the velocity $||\boldsymbol{e}_u||$, the velocity gradient $||\nabla \boldsymbol{e}_u||$, the current density $||\boldsymbol{e}_j||$ and the divergence of the current density $||\nabla \cdot \boldsymbol{e}_j||$. It can be clearly seen that in both cases, Ha=10 and Ha=100, the convergence rates are very good for every computed error. Actually, the convergence rates for the velocity gradient and the divergence of the current density are higher than the theoretical value; similar behavior has been found in [11] for the Darcy problem.

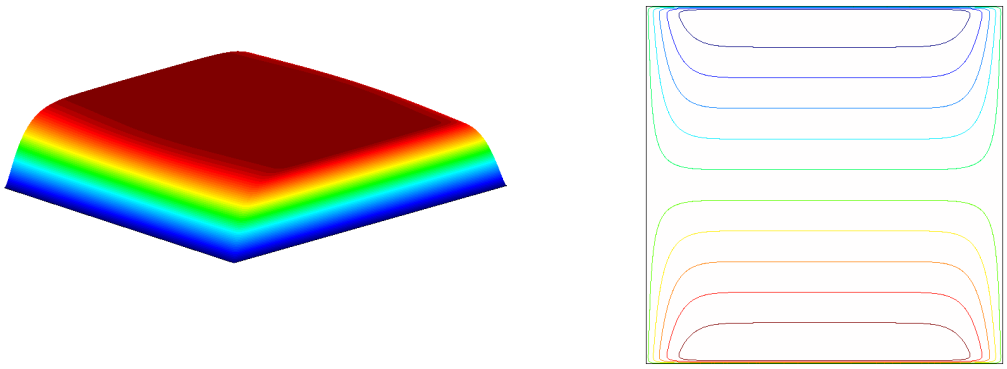
5.6.3 Hunt's case

The next test problem is Hunt's case. It corresponds to a fully developed flow in a channel with square section where the Hartmann walls are perfectly conducting and the side walls are electrically insulated. Similarly to Shercliff's case, this problem has an analytical solution (see Annex 5.B).



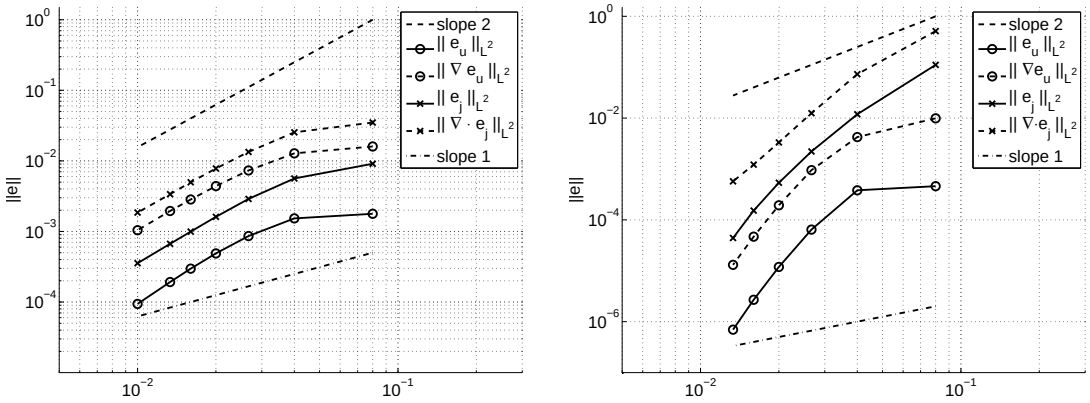
(a) Velocity field (b) Current paths

Figure 5.3: Shercliff's case : Ha = 10, Re = 10.



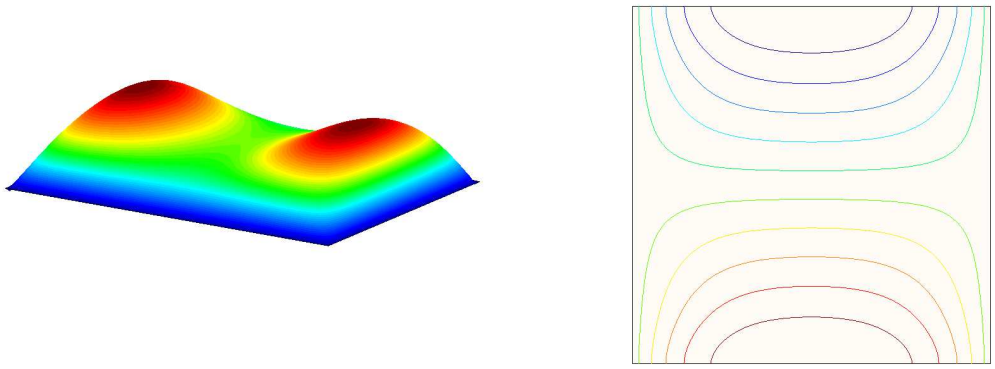
(a) Velocity field (b) Current paths

Figure 5.4: Shercliff's case : Ha = 100, Re = 10.



(a) Mesh in Fig. 5.2(a) with Ha=10 (b) Mesh in Fig. 5.2(b) with Ha=100

Figure 5.5: Shercliff's case convergence rates.



(a) Velocity field (b) Current paths

Figure 5.6: Hunt's case : Ha = 10, Re = 10.

This problem has a similar 2D behavior to the one for Shercliff's case. We have used the same computational domain, that is, a slice of channel of width 1/100 times the section sides with periodic conditions at the inflow and outflow sections. Therefore, the constant pressure gradient that drives the flow has to be set as an external body force. Its value can be computed with a slightly different formula from Shercliff's case as (see [98] for details)

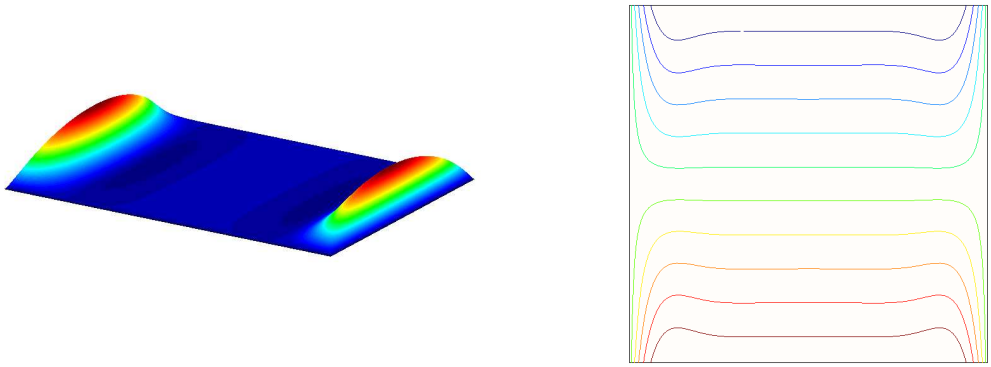
$$\frac{\mathrm{d}p}{\mathrm{d}z} = \frac{KL^3}{\rho\nu^2\mathrm{Re}}, \quad \text{where} \quad K = \frac{\mathrm{Ha}}{1 - 0.95598\mathrm{Ha}^{-1/2} - \mathrm{Ha}^{-1}}.$$

Every physical property involved in the calculation has been set equal to one. Therefore, the Hartmann number is computed directly as the norm of the external magnetic field. The meshes used to solve this problem and obtain the convergence rates are the same meshes that were used in the previous case.

The same two simulations as in the Shercliff's case have been performed. The first one is a fluid with Ha=10 and Re=10. Figure 5.6 shows the velocity field and the current paths solution of this problem when using a structured mesh of 30603 nodes and 120000 tetrahedral elements.

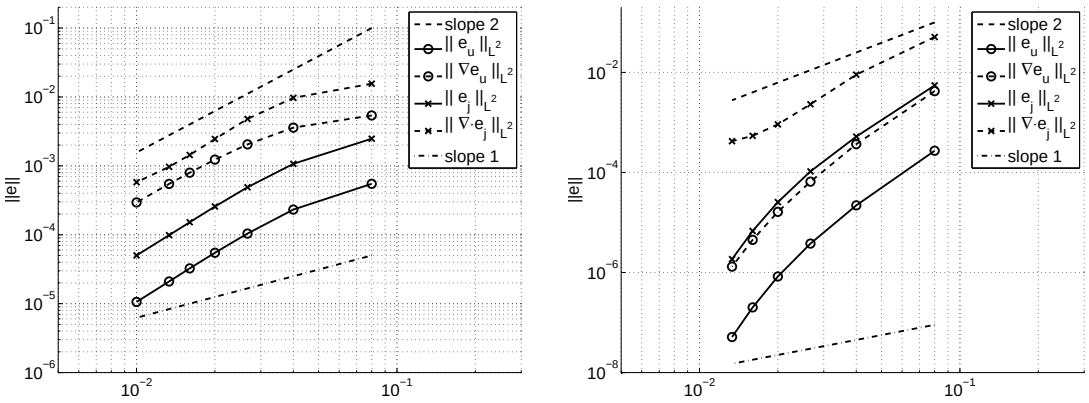
The second simulation corresponds to a fluid flowing with Ha=100 and Re=10. Figure 5.7 shows the velocity distribution and the current paths obtained with a mesh of 30603 nodes and 120000 tetrahedral elements but concentrating the elements near the boundaries to capture the Hartmann layers.

Figure 5.8 shows the convergence rates obtained for both Ha=10 and Ha=100 cases in a logarithmic scale. Again, for the meshes with element concentration, an equivalent mesh size h^* has been used. The quantities shown are: the L^2-norm of the error in the velocity $||\boldsymbol{e}_u||$, the velocity gradient $||\nabla\boldsymbol{e}_u||$, the current density $||\boldsymbol{e}_j||$ and the divergence of the current density $||\nabla\cdot\boldsymbol{e}_j||$. Again, the results show that in both cases the convergence rates are very good. Furthermore, the errors in the velocity gradient and the divergence of the current density also present a superconvergent behavior in relation to the theoretical expected value.



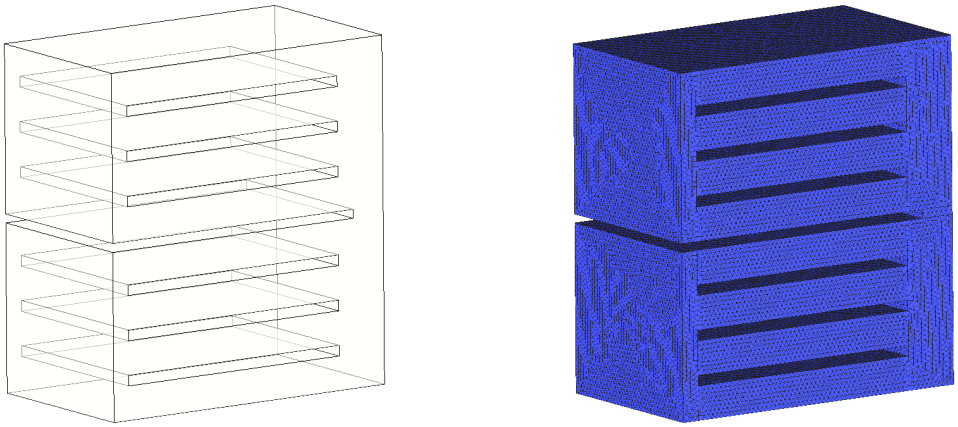
(a) Velocity field (b) Current paths

Figure 5.7: Hunt's case : Ha = 100, Re = 10.



(a) Mesh in Fig. 5.2(a) with Ha=10 (b) Mesh in Fig. 5.2(b) with Ha=100

Figure 5.8: Hunt's case convergence rates.



(a) HCLL blanket geometry

(b) Computational mesh

Figure 5.9: HCLL blanket configuration.

5.6.4 HCLL test blanket

The helium cooled lead lithium (HCLL) blanket is a liquid metal blanket concept developed in the framework of the European breeding blanket programme for a DEMO reactor to be tested in ITER (see the web site www.iter.org). Figure 5.9(a) shows the geometry considered as computational domain, see [95], [96] for details. It consists of a U-shaped channel which measures 360 mm in its longitudinal direction (x-axis). The total height is 390 mm (z-axis) divided into two subchannels of 190 mm and a transition zone of 10 mm. The section width (y-axis) is 206.5 mm. In every one of the subchannels, there are 3 cooling plates whose dimensions are $280 \times 206.5 \times 12$ mm.

Figure 5.9(b) shows the mesh generated to perform the calculations. It consists of 266,072 nodes and 1,417,435 linear tetrahedral elements. This mesh leads to 2,128,576 degrees of freedom.

The physical properties of the eutectic Pb-17Li fluid have been considered to be constant. The adopted values in this work are: fluid density $\rho = 9.2 \times 10^3$ kg/m^3, fluid viscosity $\nu = 1.4 \times 10^{-7}$ m^2/s and fluid electrical conductivity $\sigma = 7.4 \times 10^3$ 1/Ωm (see [37] and [62] for more details). The external magnetic field applied to the fluid has a value of 10 T and has a direction in the y-axis, $\boldsymbol{B} = (0, 10, 0)$T. Considering that the characteristic magnetic length is half the length of the side walls, $L = 0.103$ m, the Hartmann number associated to this flow is Ha=2470.

The hydrodynamic boundary conditions have been set as $\boldsymbol{u} = \boldsymbol{0}$ at the walls, both the external walls and the cooling plates, $\boldsymbol{u} = (0.001, 0, 0)$m/s at the inlet, which corresponds to the bottom subchannel, and free condition at the outlet, the top subchannel.

On the other hand, the magnetic boundary conditions have been set as perfectly insulating material in the exterior walls, that is $\boldsymbol{j} \cdot \boldsymbol{n} = 0$, and perfectly conducting material in the cooling plates, which corresponds to $\boldsymbol{j} \times \boldsymbol{n} = \boldsymbol{0}$.

The solution to this problem converges to a stationary solution. In Figure 5.10 there have been plotted the solutions in the plane $y = 0.103$ m. Those graphics show the longitudinal behavior of the flow in the x-direction. The velocity field shows clearly that

the distribution of the cooling plates in the top subchannel is not optimal because almost the entire flow takes place in the top part of the subchannel whereas in the bottom part the fluid has velocity equal to zero. Furthermore, the high values of the velocity near the top part of the top subchannel results on higher values of the current density in the same zone, instead of the distribution that could be expected, similar to the Shercliff's case solution, which actually is the solution in the bottom subchannel.

Figure 5.11(a) shows the streamlines of the velocity field. It is clearly seen how the fluid entering the blanket from the inlet surface goes to the outlet through only the top 2 subchannels, leaving the bottom 2 subchannels of the upper module with almost zero velocity. On the other hand, Figure 5.11(b) displays the current density streamlines in section $x = 0.150$ m. The streamlines in the bottom module reproduce almost perfectly the streamlines of Shercliff's case where both the Hartmann and side walls are perfectly insulating. However, the top module behavior is different. The velocity field concentrating in the top 2 subchannels produces a different distribution of the current density field.

5.7 Conclusions

In this chapter, a numerical formulation to solve the inductionless MHD equations that consists of a stabilized FE method has been presented. Its design is based on the variational multiscale framework which is derived from a splitting of the unknown into two parts, a FE component and a subscale that corresponds to the part of the unknown that cannot be captured by the discretization. The crucial point in this approach resides in the subscale approximation.

The most important aspects of this formulation are that it allows to use equal interpolation for all the unknowns without having to satisfy the compatibility conditions. Furthermore, it is stable and optimally convergent in a norm that is meaningful for every value of the physical parameters of the fluid.

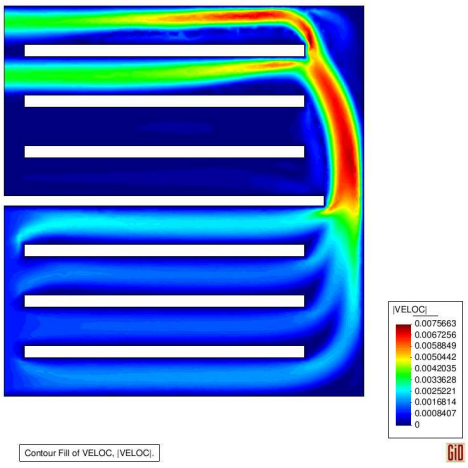
Another key point of this formulation is the monolithic approach for solving the problem instead of the possibility of uncoupling the global problem by solving a Laplacian equation for the electric potential. This latter option needs a block iteration algorithm to converge to the coupled solution but there exists no guarantee that it will converge to the solution nor the number of iterations needed in case it converges.

The approximation of the subscales leads to the introduction of some stabilization parameters that need to be proposed. An interesting point of this work is that these parameters have been designed based on the stability and convergence analysis of the method.

The time integration and linearization of the problem considered here is the simplest possible which leads to a method easy to implement but without losing any robustness and convergence properties. The numerical experimentation presented in this chapter validates these statements and the theoretical development of the method.

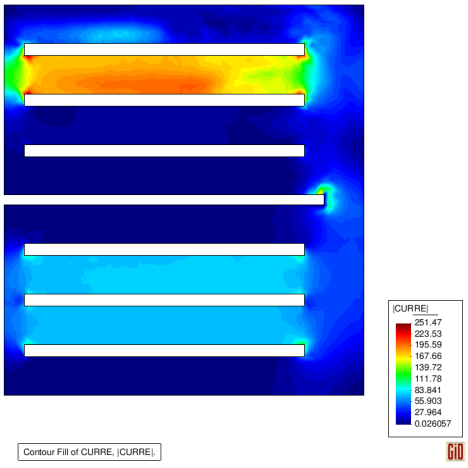
5.A Shercliff's analytical solution

Let the side walls be of length $2a$, the Hartmann walls of length $2b$ and $l = b/a$. The Hartmann walls are considered to have arbitrary conductivity with $d_B = (t_w\sigma_w)/(a\sigma)$,

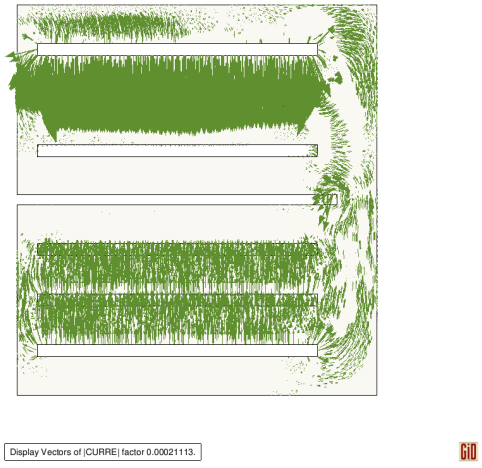


(a) Velocity norm

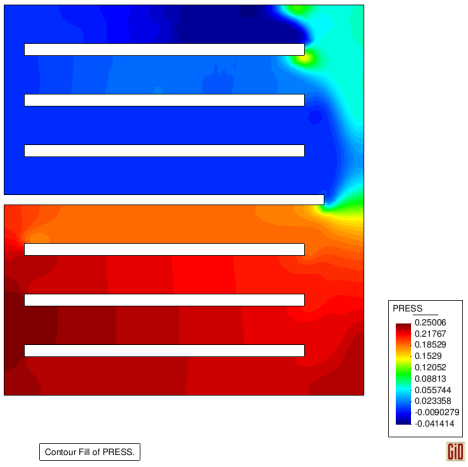
(b) Velocity field



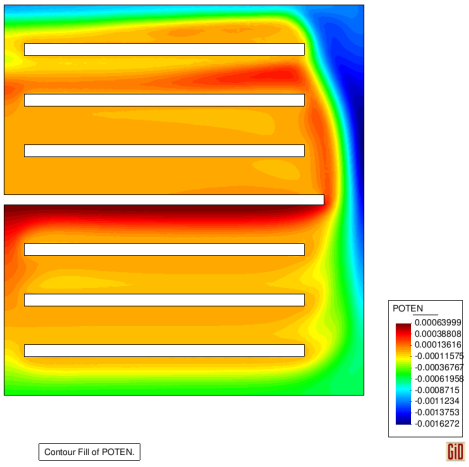
(c) Current norm



(d) Current field

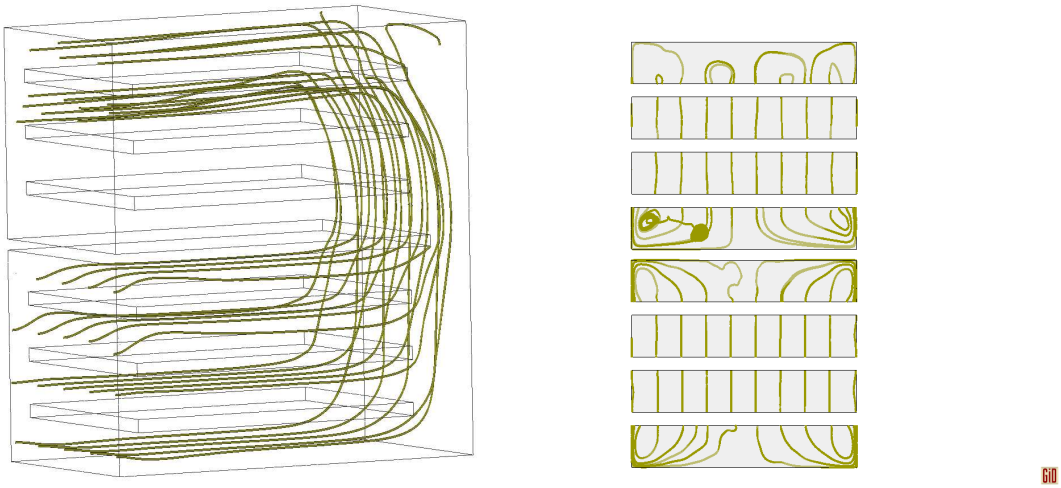


(e) Pressure



(f) Electric potential

Figure 5.10: Results in section $y = 0.103$ m.



(a) Velocity streamlines

(b) Current density streamlines at section $x = 0.150$ m

Figure 5.11: Velocity field and current density field streamlines

where σ_w is the conductivity of the wall, t_w its thickness and σ the conductivity of the fluid. The analytical solution was given by Hunt [86] as a Fourier series in $\xi = x/a \in [-l, l]$ and $\eta = y/a \in [-1, 1]$. The z-component of the velocity is written as

$$u_z = \frac{V}{\mu}\left(-\frac{\partial p}{\partial z}\right)a^2, \quad \text{where } V = \sum_{k=0}^{\infty}\frac{2(-1)^k\cos(\alpha_k\xi)}{l\alpha_k^3}(1 - V2 - V3), \tag{5.42}$$

for

$$V2 = \frac{\left(d_B r_{2k} + \frac{1-\exp(-2r_{2k})}{1+\exp(-2r_{2k})}\right)\frac{\exp(-r_{1k}(1-\eta))+\exp(-r_{1k}(1+\eta))}{2}}{\frac{1+\exp(-2r_{1k})}{2}d_B N + \frac{1-\exp(-2(r_{1k}+r_{2k}))}{1+\exp(-2r_{2k})}},$$

$$V3 = \frac{\left(d_B r_{1k} + \frac{1-\exp(-2r_{1k})}{1+\exp(-2r_{1k})}\right)\frac{\exp(-r_{2k}(1-\eta))+\exp(-r_{2k}(1+\eta))}{2}}{\frac{1+\exp(-2r_{2k})}{2}d_B N + \frac{1-\exp(-2(r_{1k}+r_{2k}))}{1+\exp(-2r_{1k})}},$$

and

$$N = (\mathrm{Ha}^2 + 4\alpha_k^2)^{1/2}, \qquad r_{1k}, r_{2k} = \frac{1}{2}\left(\pm\mathrm{Ha} + (\mathrm{Ha}^2 + 4\alpha_k^2)^{1/2}\right), \qquad \alpha_k = \left(k + \frac{1}{2}\right)\frac{\pi}{l}.$$

On the other hand, the current density components j_x and j_y are $j_x = \frac{\partial H_z}{\partial y}$ and $j_y = -\frac{\partial H_z}{\partial x}$ for

$$H_z = \frac{H}{\mu^{1/2}}\left(-\frac{\partial p}{\partial z}\right)a^2\sigma^{1/2} \quad \text{where } H = \sum_{k=0}^{\infty}\frac{2(-1)^k\cos(\alpha_k\xi)}{l\alpha_k^3}(H2 - H3), \tag{5.43}$$

for

$$H2 = \frac{\left(d_B r_{2k} + \frac{1-\exp(-2r_{2k})}{1+\exp(-2r_{2k})}\right) \frac{\exp(-r_{1k}(1-\eta))-\exp(-r_{1k}(1+\eta))}{2}}{\frac{1+\exp(-2r_{1k})}{2} d_B N + \frac{1-\exp(-2(r_{1k}+r_{2k}))}{1+\exp(-2r_{2k})}},$$

$$H3 = \frac{\left(d_B r_{1k} + \frac{1-\exp(-2r_{1k})}{1+\exp(-2r_{1k})}\right) \frac{\exp(-r_{2k}(1-\eta))-\exp(-r_{2k}(1+\eta))}{2}}{\frac{1+\exp(-2r_{2k})}{2} d_B N + \frac{1-\exp(-2(r_{1k}+r_{2k}))}{1+\exp(-2r_{1k})}}.$$

V_z, j_x and j_y are precisely the analytical solution of the problem. Note that in the Shercliff's case the Hartmann walls are perfectly insulating, and therefore $d_B = 0$ in the above formulae. Note also that the formulae in [86] have been written in terms of exponential functions to allow its computation in a computer. The original formulae in terms of hyperbolic functions is not suitable for computing at high values of the Hartmann number.

5.B Hunt's analytical solution

Hunt's problem has an analytical solution in the form of Fourier series that can be found in an article from J.C.R. Hunt [86]. The analytical solution is computed using the formulae (5.42)-(5.43). In this case, the Hartmann walls are perfectly conducting and therefore $d_B \to \infty$. Thus, the modifications in the formulae (5.42)-(5.43) for Hunt's case consist of taking the limit $d_B \to \infty$ in the Fourier series:

$$V2 = \frac{r_{2k}}{N} \cdot \frac{\exp(-r_{1k}(1-\eta)) + \exp(-r_{1k}(1+\eta))}{1+\exp(-2r_{1k})},$$

$$V3 = \frac{r_{1k}}{N} \cdot \frac{\exp(-r_{2k}(1-\eta)) + \exp(-r_{2k}(1+\eta))}{1+\exp(-2r_{2k})},$$

$$H2 = \frac{r_{2k}}{N} \cdot \frac{\exp(-r_{1k}(1-\eta)) - \exp(-r_{1k}(1+\eta))}{1+\exp(-2r_{1k})},$$

$$H3 = \frac{r_{1k}}{N} \cdot \frac{\exp(-r_{2k}(1-\eta)) - \exp(-r_{2k}(1+\eta))}{1+\exp(-2r_{2k})}.$$

Chapter 6

Recursive block preconditioners for the thermal inductionless MHD

The thermally coupled incompressible inductionless magnetohydrodynamics (MHD) problem models the flow of an electrically charged fluid under the influence of an external electromagnetic field with thermal coupling. This system of partial differential equations is strongly coupled and highly nonlinear for real cases of interest. Therefore, fully implicit time integration schemes are very desirable in order to capture the different physical scales of the problem at hand. However, solving the multiphysics linear systems of equations resulting from such algorithms is a very challenging task which requires efficient and scalable preconditioners. In this chapter, a new family of recursive block LU preconditioners is designed and tested for solving the thermally coupled inductionless MHD equations. These preconditioners are obtained after splitting the fully coupled matrix into one-physics problems for every variable (velocity, pressure, current density, electric potential and temperature) that can be optimally solved, e.g., using preconditioned domain decomposition algorithms. The main idea is to arrange the original matrix into an (arbitrary) 2×2 block matrix, and consider a LU preconditioner obtained by approximating the corresponding Schur complement. For every one of the diagonal blocks in the LU preconditioner, if it involves more than one type of unknown, we proceed the same way in a recursive fashion. This approach is stated in an abstract way, and can be straightforwardly applied to other multiphysics problems. Further, we precisely explain a flexible and general software design for the code implementation of this type of preconditioners.

6.1 Introduction

The thermally coupled incompressible inductionless magnetohydrodynamics (MHD) model describes the dynamics of an electrically conducting fluid under an external electromagnetic field with thermal coupling where the magnetic field induced by the currents is negligible with respect to the externally applied one. This system of partial differential equations can be applied to simulate a wide range of applications, such as MHD pumps, steel casting processes, crystal growth devices or breeding blankets in nuclear fusion reactors. The Galerkin finite element (FE) approximation of this problem faces

several well-known drawbacks. First, convective dominated flows may lead to oscillations because the system loses its elliptic nature. Second, a strong coupling between the two saddle-point subproblems, the hydrodynamic and the magnetic ones, may introduce numerical instabilities when solving the resulting linear systems of equations. Finally, there is the need to satisfy the classical inf-sup conditions between the approximation spaces for the velocity and the pressure and also for the current density and the electric potential in order to have a well-posed problem. There exist several options to circumvent these difficulties being stabilization methods one of the most widely used. In this work, we consider two stabilization techniques based on the variational multiscale ideas in [46, 49, 83].

The discretization of realistic problems with very fine finite element meshes leads to linear systems of equations to be solved with a number of degrees of freedom in the range of $10^6 - 10^9$. For the solution of such huge systems, an efficient and scalable preconditioner is required. There exist several approaches for preconditioning this type of problems. One approach that has been extensively used for preconditioning large-scale multi-physics problems is the algebraic multigrid (AMG) algorithm [91, 112]. This technique is very efficient for Laplacian-type problems but suffers for indefinite and nonsymmetric problems. Another interesting approach consists of an approximate block LU factorization of the system matrix. This type of preconditioners have been widely studied for the incompressible Navier-Stokes equations in the fluid mechanics community [59, 66, 67]. Recently, this approach has been applied to the full resistive MHD equations, see Chapter 4 (using the preconditioner as a solver in an operator splitting fashion), and to the 2D incompressible (reduced) resistive MHD formulation [60]. The crucial aspect in these approximate block preconditioners relies in an efficient approximation of the Schur complement that allows the uncoupling between the several physical variables of the problem at the preconditioner level.

These approximate block factorization ideas have been used in this work to design new block recursive LU preconditioners for the inductionless MHD and the thermally coupled inductionless MHD problems. As an example, the inductionless MHD system involves four different unknowns (velocity, pressure, current density and electric potential) and leads to a 4×4 block system matrix (one block per unknown). Our approach consists of arranging the multiphysics 4×4 block matrix as a 2×2 block matrix (grouping unknowns) where in turn every block is a 2×2 block matrix. Then, we perform an incomplete LU factorization of the 2×2 system block matrix where we consider some (cheap) approximation of the resulting Schur complement and possibly the rest of diagonal blocks. Recursively, since the diagonal blocks are in fact 2×2 block matrices, we approximate these matrices the same way, i.e., using an incomplete LU approximation. As a result, the only blocks that have to be *inverted* are one-variable (one-physics) problems. The key point for these preconditioners to be efficient relies in obtaining a good approximation of the Schur complement for all 2×2 systems. In order to define the Schur complement approximations, we have extended ideas from the techniques used for the incompressible Navier-Stokes equations to the inductionless MHD system. Moreover, a study of the exact Schur complement behavior and the effect of cancelling different terms in it has allowed us to propose an improved version of the Pressure-Convection-Diffusion (PCD) preconditioner (see [67]) where we introduce additional stabilization terms. The application of the MHD preconditioner to the thermally coupled problem is

straightforward, since the coupling is in one direction only.

The contributions of the chapter are the following. On one side, we propose new stabilized formulations based on term-by-term projection stabilization for the inductionless MHD problem, and extend this formulation and the one in Chapter 5 to the thermally coupled case. The most particular feature of our approach is the explicit introduction of the current density as an additional unknown of the problem. We note that the typical approach is to decouple fluid and electromagnetic problems [37, 95, 101, 102]. Then, the electromagnetic problem is solved in terms of the electric potential only and the current density and Lorentz force are computed. Next, the fluid problem is solved with the previously computed Lorentz force. This approach treats the multiphysics coupling explicitly (for transient problems), and, in the best case, it is only coupled via fixed point iterations (if it converges). However, for large Hartmann numbers (in fusion reaction breeding blanket simulations it is in the order of $10^4 - 10^5$) this approach can only work for extremely small time step values for transient problems and it cannot be used for steady problems. In any case, the resulting linear systems in our approach still require efficient preconditioning for high Hartmann numbers. Thus, we propose an abstract setting to design preconditioners for multiphysics problems, based on a recursive use of block factorization. This general framework is applied to the (thermally coupled) inductionless MHD problem, considering different preconditioners based on approximations of the resulting Schur complement matrices. The efficiency of these preconditioners is assessed via a complete set of numerical experiments. Finally, we give details about an abstract and flexible implementation of block recursive preconditioning. The combination of our FE formulations, with an explicit treatment of the current density, and the recursive LU preconditioners we propose, allow us to solve realistic breeding blanket simulations with very high Hartmann numbers.

The chapter is organized as follows. Section 6.2 states the problem in both its strong and weak form for the dimensionless version of the inductionless MHD equations and the thermally coupled inductionless MHD problem. In Section 6.3, the stabilization methods used in this work are presented and the block structure of the resulting linear system of equations is highlighted. In Section 6.4, the underlying recursion in the design of block preconditioners is defined together with a review of classical block preconditioners from fluid mechanics. Moreover, two different recursive block preconditioners are developed for the inductionless MHD problem depending on the grouping of the physical variables. For each preconditioner, the approximation of the Schur complement is chosen. An improved version of PCD block preconditioners is also derived in Section 6.4 with an experimental justification of its design. To close Section 6.4, the recursive block preconditioners derived for the inductionless MHD problem are extended for the thermally coupled problem in a straightforward manner. Some numerical experiments to test the properties of the block preconditioners presented in this work are carried out in Section 6.5. Section 6.6 deals with some software design and implementation aspects that are crucial to manage the block preconditioners recursion. Finally, some conclusions are drawn in Section 6.7.

6.2 Continuous MHD problem

6.2.1 Isothermal inductionless MHD

The incompressible inductionless mangnetohydrodynamics (MHD) system of partial differential equations consists of the incompressible Navier-Stokes problem coupled with Ohm's law and the electric charge conservation equations via the Lorentz force. It reads as: find a velocity field $u(x,t)$, a pressure $p(x,t)$, a current density field $j(x,t)$ and an electric potential $\phi(x,t)$ such that

$$\partial_t u + (u\cdot\nabla)u - \nu\Delta u + \nabla p - \frac{1}{\rho}(j\times B) = f, \tag{6.1}$$

$$\nabla\cdot u = 0, \tag{6.2}$$

$$j + \sigma\nabla\phi - \sigma(u\times B) = 0, \tag{6.3}$$

$$\nabla\cdot j = 0, \tag{6.4}$$

in $(x,t)\in\Omega\times(0,T)$, where $\Omega\subset\mathbb{R}^d$ is the open bounded domain filled by the fluid and d is the spatial dimension. The partial time derivative is denoted by ∂_t and ν, ρ, σ stand for the viscosity, density and electric conductivity of the fluid, respectively. Finally, f corresponds to the body forces of the flow motion and B to the external magnetic field.

Consider a partition of the domain boundary $\Gamma=\partial\Omega$ into two parts $\Gamma=\Gamma_{E,u}\cup\Gamma_{N,u}$. The boundary conditions for the velocity field are no-slip wall conditions, $u=0$ on $\Gamma_{E,u}$, and zero traction conditions, $-pn+\nu n\cdot\nabla u=0$ on $\Gamma_{N,u}$. Let us define a different partition of the domain boundary for imposing the electromagnetic boundary conditions, $\Gamma=\Gamma_{C,j}\cup\Gamma_{I,j}$, where $\Gamma_{C,j}$ corresponds to perfectly conducting walls and $\Gamma_{I,j}$ to perfectly insulating walls. Perfectly conducting walls do not apply any resistance to the current which implies that the electric currents cross the wall surface orthogonally. This condition is imposed by $j\times n=0$ on $\Gamma_{C,j}$. Taking into account that $u=0$ on $\Gamma_{C,j}$, this condition is equivalent to impose $\phi=0$ on $\Gamma_{C,j}$. On the other hand, insulating walls do not allow the electric currents to cross them, which means that the normal component of the current density field has to vanish, that is, $j\cdot n=0$ on $\Gamma_{I,j}$. Lastly, an initial condition for the velocity field has to be considered, i.e., $u=u_0$ in Ω for $t=0$.

It is convenient for solving real problems with extreme physical properties (as the test blanket module case in Section 6.5.3) to work with the dimensionless form of the incompressible inductionless MHD system. Let us redefine the problem in (6.1)-(6.4) by (see [98])

$$\partial_t u + (u\cdot\nabla)u - \frac{1}{\mathrm{Re}}\Delta u + \nabla p - \mathrm{N}(j\times B) = f, \tag{6.5}$$

$$\nabla\cdot u = 0, \tag{6.6}$$

$$j + \nabla\phi - (u\times B) = 0, \tag{6.7}$$

$$\nabla\cdot j = 0, \tag{6.8}$$

where the variables and operators have been scaled as

$$\begin{aligned} u &\to u\tfrac{1}{u_0}, & \nabla &\to \nabla L, \\ t &\to t\tfrac{u_0}{L}, & B &\to B\tfrac{1}{B_0}, \\ p &\to p\tfrac{1}{\rho u_0^2}, & f &\to f\tfrac{L}{u_0^2}, \\ j &\to j\tfrac{1}{\sigma u_0 B_0}, & \phi &\to \phi\tfrac{1}{u_0 B_0}. \end{aligned}$$

Here, L is a characteristic length of the domain and u_0, B_0 are the characteristic scales for the velocity and the external magnetic fields, respectively. Let us also introduce the dimensionless numbers in system (6.5)-(6.8)

$$\text{Reynolds number, Re} = \frac{u_0 L}{\nu},$$

$$\text{Interaction parameter, N} = \frac{\sigma L B_0^2}{\rho u_0}.$$

There is also another dimensionless number that governs system (6.5)-(6.8) behavior, the Hartmann number $\text{Ha} = \sqrt{\text{ReN}}$, that gives the ratio between electromagnetic and viscous forces.

Let us introduce some standard notation here. We denote by $L_0^2(\Omega)$ the set of functions in $L^2(\Omega)$ with zero mean value and by $H^1_{0,\Gamma}(\Omega)^d$ the functions belonging to $H^1(\Omega)$ that vanish on $\Gamma \subset \partial\Omega$. Let us now consider the following functional spaces: $V_u = H^1_{0,\Gamma_{E,u}}(\Omega)^d$, $Q_u = L_0^2(\Omega)$, $V_j = L^2(\Omega)$ and $Q_\phi = H^1_{0,\Gamma_{C,j}}(\Omega)$. Assuming a smooth external magnetic field B, the weak form of system (6.5)-(6.8) reads as follows: find $(u, p, j, \phi) \in V_u \times Q_p \times V_j \times Q_\phi$ such that

$$F_u(u)u + C_u j + G_u p = f_u \quad \text{in} \quad V_u', \qquad D_u u = 0 \quad \text{in} \quad Q_p', \tag{6.9}$$

$$F_j j + C_j u + G_j \phi = 0 \quad \text{in} \quad V_j', \qquad D_j j = 0 \quad \text{in} \quad Q_\phi', \tag{6.10}$$

for almost every $t \in (0, T)$. We omit the discussion about the regularity in time of the unknowns for simplicity. The definition of these operators comes from system (6.5)-(6.8) and the notion of weak derivatives as follows.

The semi-linear fluid operator $F_u : V_u \times V_u \to V_u'$ is defined as $F_u(w)v := M_u \partial_t v + K_u(w)v$. The fluid mass matrix $M_u : V_u \to V_u'$ is defined as $M_u v := (v, \cdot)$. The semi-linear operator $K_u : V_u \times V_u \to V_u'$ includes the viscous and convective terms and reads as $K_u(w)v := \frac{1}{\text{Re}}(\nabla v, \nabla(\cdot)) + \langle (w \cdot \nabla)v, \cdot \rangle$. $G_u : Q_p \to V_u'$ denotes the (integrated by parts) pressure gradient, i.e., $G_u q := -(q, \nabla \cdot (\cdot))$, and $D_u = -G_u^T$ is the velocity divergence operator; the superscript T indicates the transpose operator. The linear Lorentz force coupling operator $C_u : V_j \to V_u'$ is defined as $C_u k := -\text{N}\langle \cdot, k \times B \rangle$. With regard to the magnetic subproblem, we define the mass matrix operator $F_j : V_j \to V_j'$ as $F_j k := (k, \cdot)$, $C_j := -C_u^T$, the electric potential gradient operator $G_j \psi := (\cdot, \nabla\psi)$ and the corresponding divergence operator $D_j = -G_j^T$. We note that the conditions on $\Gamma_{E,u}$ and $\Gamma_{C,j}$ are strongly enforced whereas those on $\Gamma_{N,u}$ and $\Gamma_{I,j}$ are weakly enforced (see Chapter 5 for a more detailed explanation).

6.2.2 Thermally coupled problem

The strong form of the thermally coupled incompressible inductionless MHD problem is obtained from equations (6.5)-(6.8) and Boussinesq approximation for the thermal

coupling. It reads as: find a velocity field $u(x,t)$, a pressure $p(x,t)$, a current density field $j(x,t)$, an electric potential $\phi(x,t)$ and a temperature $\theta(x,t)$ such that,

$$\partial_t u + (u\cdot\nabla)u - \frac{1}{\mathrm{Re}}\Delta u + \nabla p - \mathrm{N}(j\times B) + \frac{\mathrm{Gr}}{\mathrm{Re}^2}\theta = f + \frac{\mathrm{Gr}}{\mathrm{Re}^2}\theta_{ref}, \tag{6.11}$$

$$\nabla\cdot u = 0, \tag{6.12}$$

$$j + \nabla\phi - (u\times B) = 0, \tag{6.13}$$

$$\nabla\cdot j = 0, \tag{6.14}$$

$$\partial_t\theta + (u\cdot\nabla)\theta - \frac{1}{\mathrm{Pe}}\Delta\theta = Q, \tag{6.15}$$

where the temperature has been scaled as $\theta \to \theta\frac{1}{\Delta\theta}$. The additional dimensionless numbers in (6.11)-(6.15) are defined as,

$$\text{Grashof number, } \mathrm{Gr} = \frac{g\beta\Delta\theta L^3}{\nu^2},$$
$$\text{Péclet number, } \mathrm{Pe} = \frac{\rho c_p u_0 L}{k_t},$$

where g is the norm of the gravity field, β the thermal expansion coefficient, $\Delta\theta$ a temperature increment, θ_{ref} a reference temperature, Q a heat source, c_p the specific heat at constant pressure, k_t the thermal conductivity and $\kappa = \frac{k_t}{\rho c_p}$ the thermal diffusivity. Moreover, the Prandtl number is a dimensionless number that relates the viscous and thermal diffusivities, $\mathrm{Pr} = \frac{\nu}{\kappa}$.

The boundary conditions to be imposed for problem (6.11)-(6.15) correspond to the conditions stated in Section 6.2.1 for the MHD variables plus boundary conditions for the temperature. Consider a partition of the domain boundary such as $\Gamma = \Gamma_{E,\theta}\cup\Gamma_{N,\theta}$. The Dirichlet condition on $\Gamma_{E,\theta}$ implies a fixed temperature $\theta = \theta_D$ whereas the condition on $\Gamma_{N,\theta}$ corresponds to imposing a heat flux on the boundary, i.e., $\frac{k_t}{\rho c_p} n\cdot\nabla\theta = q$. Finally, an initial condition for the temperature field, $\theta = \theta_0$ in Ω for $t = 0$, has to be considered. We assume that $Q = q = \theta_D = 0$ for simplicity.

We define the temperature functional space $V_\theta = H^1_{0,\Gamma_{E,\theta}}(\Omega)$. The weak form of problem (6.11)-(6.15) can be stated as: find $(u,p,j,\phi,\theta)\in V_u\times Q_p\times V_j\times Q_\phi\times V_\theta$ such that

$$F_u(u)u + C_u j + G_u p + H_u\theta = f_u \quad \text{in} \quad V_u', \qquad D_u u = 0 \quad \text{in} \quad Q_p', \tag{6.16}$$

$$F_j j + C_j u + G_j\phi = 0 \quad \text{in} \quad V_j', \qquad D_j j = 0 \quad \text{in} \quad Q_\phi', \tag{6.17}$$

$$F_\theta(u)\theta = 0 \quad \text{in} \quad V_\theta'. \tag{6.18}$$

where the operators related to the inductionless MHD problem are defined in Section 6.2.1. The semi-linear thermal problem operator $F_\theta : V_u\times V_\theta \to V_\theta'$ reads as $F_\theta(w)\varphi := (\partial_t\varphi,\cdot) + \frac{1}{\mathrm{Pe}}(\nabla\varphi,\nabla\cdot) + \langle(w\cdot\nabla)\varphi,\cdot\rangle$, whereas the buoyancy term operator $H_u : V_\theta \to V_u'$ is $H_u\theta := \frac{\mathrm{Gr}}{\mathrm{Re}^2}(\theta,\cdot)$.

6.3 Stabilized finite element formulation

6.3.1 Isothermal inductionless MHD

The variational problem (6.9)-(6.10) is linearized, discretized in time and spatially approximated following the same procedure as in Chapter 5. The linearization method chosen is Picard method, the time derivatives are discretized using the trapezoidal rule and the spatial approximation is obtained with the standard Galerkin method. Given, e.g., the operator M_u, we denote its finite element restriction as $M_{uh} : V_{uh} \to V'_{uh}$. This way, we define the FE restriction of all the operators in Section 6.2.1. However, in order to simplify notation, we will omit the h subindex for the discrete operators. Thus, the discrete and linearized form can be stated as

$$F_u(a_h^{n+\alpha})u_h^{n+\alpha} + C_u j_h^{n+1} + G_u p_h^{n+1} = f_u^{n+\alpha} \quad \text{in} \quad V'_{u,h}, \quad D_u u_h^{n+\alpha} = 0 \quad \text{in} \quad Q'_{ph}, \tag{6.19}$$

$$F_j j_h^{n+1} + C_j u_h^{n+\alpha} + G_j \phi_h^{n+1} = 0 \quad \text{in} \quad V'_{jh}, \quad D_j j_h^{n+1} = 0 \quad \text{in} \quad Q'_{\phi h}, \tag{6.20}$$

where $a_h^{n+\alpha} = u_h^{n+\alpha,k}$ being k the previous iteration of the nonlinear Picard iterative loop and the discrete FE spaces V_{uh}, Q_{ph}, V_{jh} and $Q_{\phi h}$ are subspaces of their infinite dimensional counterparts V_u, Q_p, V_j and Q_ϕ. The right-hand-side (RHS) term $f_u^{n+\alpha}$ includes the time derivative term $\frac{1}{\delta t} M_u u_h^n$.

The Galerkin approximation of this problem is known to have many drawbacks. First, the discrete problem is well-posed only if the discrete inf-sup conditions for $V_{uh} \times Q_{ph}$ and $V_{jh} \times Q_{\phi h}$ are satisfied:

$$\inf_{q_h \in Q_{ph}} \sup_{v_h \in V_{uh}} \frac{(q_h, \nabla \cdot v_h)}{\|q_h\| \|\nabla v_h\|} \geq \beta^* > 0, \qquad \inf_{\psi_h \in Q_{\phi h}} \sup_{k_h \in V_{jh}} \frac{(\nabla \psi_h, k_h)}{\|\nabla \psi_h\| \|k_h\|} \geq \gamma^* > 0,$$

where β^* and γ^* are positive constants independent of the mesh size h. Depending on how the finite element spaces V_{uh}, Q_{ph}, V_{jh} and $Q_{\phi h}$ are chosen, these conditions might not be satisfied. For instance, equal order spaces do not fulfill the discrete inf-sup conditions. Moreover, when solving problems where the first order derivatives dominate the second order ones in the Navier-Stokes equations, that is, convection dominated cases, oscillations may appear in the solution. Finally, a strong coupling between the hydrodynamic and electromagnetic problems may lead to numerical instabilities.

The solution adopted in this work to avoid these drawbacks consists of stabilization methods. The basic idea under a stabilization method is to add certain terms to the variational form of the problem that allow one to circumvent the previously mentioned difficulties associated to the Galerkin approximation of the problem without spoiling accuracy. Two different stabilization methods are used in this work.

The first one is the algebraic sub-grid scale method (ASGS) following the subgrid scale concept introduced in [83]. We can consider a variational multiscale formulation of the problem, by using the framework in [83,84]. Let us denote the finite element partition as $\mathcal{T}_h$; $K \in \mathcal{T}_h$ is a FE. This way, we add a term of the type $\sum_{k \in \mathcal{T}_h} \int_K (F - \partial_t \mathcal{M}(U) - \mathcal{L}(U)) \cdot \boldsymbol{\tau} \mathcal{L}^T(V) \mathrm{d}x$ to the left-hand-side (LHS) of (6.19)-(6.20), where F is the forcing term, $\mathcal{L}$ the steady-state spatial differential operator of the problem at hand and $\mathcal{M}$ the continuous mass operator. $\boldsymbol{\tau}$ is the matrix of stabilization parameters. Alternatively, when considering a Galerkin/Least-Squares (GLS) stabilization of the problem, we just

replace $\mathcal{L}^T(V)$ by $-\mathcal{L}(V)$ in the definition of the stabilization term. The application of this method to the incompressible inductionless MHD problem is deeply explained in Chapter 5. We include this method in Algorithm 6.1 for the sake of completeness. The symbol Δ_h stands for the broken Laplacian, i.e., $(\Delta_h v_h, \cdot) = \sum_{K\in\mathcal{T}_h} \int_K \Delta v_h(\cdot)\mathrm{d}x$.

Algorithm 6.1: ASGS stabilization for the inductionless MHD problem

Given u_h^n at the previous time step value, find u_h^{n+1}, j_h^{n+1}, p_h^{n+1} and ϕ_h^{n+1} such that

$$
\begin{aligned}
&\left(\delta_t u_h^{n+1,k+1}, v_h\right) + \left\langle (u_h^{n+1,k}\cdot\nabla)u_h^{n+1,k+1}, v_h\right\rangle + \tfrac{1}{\mathrm{Re}}\left(\nabla u_h^{n+1,k+1}, \nabla v_h\right) - \left(p_h^{n+1,k+1}, \nabla\cdot v_h\right)\\
&\quad -\mathrm{N}\left\langle j_h^{n+1,k+1}\times B, v_h\right\rangle + \left\langle (u_h^{n+1,k}\cdot\nabla)v_h + \tfrac{1}{\mathrm{Re}}\Delta_h v_h, \tau_1^{n+1,k}\boldsymbol{R}_{h,u}^{n+1,k+1}\right\rangle\\
&\quad + \left(\nabla\cdot v_h, \tau_2^{n+1,k}R_{h,p}^{n+1,k+1}\right) - \left\langle (v_h\times B), \tau_3^{n+1,k}\boldsymbol{R}_{h,j}^{n+1,k+1}\right\rangle = \left(f^{n+1}, v_h\right),\\
&\left(\nabla\cdot u_h^{n+1,k+1}, q_h\right) + \left\langle \tau_1^{n+1,k}\boldsymbol{R}_{h,u}^{n+1,k+1}, \nabla q_h\right\rangle = 0,\\
&\left(j_h^{n+1,k+1}, k_h\right) + \left(\nabla\phi_h^{n+1,k+1}, k_h\right) - \left\langle u_h^{n+1,k+1}\times B, k_h\right\rangle\\
&\quad -\mathrm{N}\left\langle k_h\times B, \tau_1^{n+1,k}\boldsymbol{R}_{h,u}^{n+1,k+1}\right\rangle - \left\langle k_h, \tau_3^{n+1,k}\boldsymbol{R}_{h,j}^{n+1,k+1}\right\rangle\\
&\quad + \left(\nabla\cdot k_h, \tau_4^{n+1,k}R_{h,\phi}^{n+1,k+1}\right) = 0,\\
&-\left(j_h^{n+1,k+1}, \nabla\psi_h\right) + \left\langle \nabla\psi_h, \tau_3^{n+1,k}\boldsymbol{R}_{h,j}^{n+1,k+1}\right\rangle = 0,
\end{aligned}
$$

where the residuals are:

$$
\begin{aligned}
\boldsymbol{R}_{h,u} &:= \delta_t u_h + (a\cdot\nabla)u_h - \tfrac{1}{\mathrm{Re}}\Delta_h u_h + \nabla p_h - \mathrm{N}(j_h\times B) - f,\\
R_{h,p} &:= \nabla\cdot u_h,\\
\boldsymbol{R}_{h,j} &:= j_h + \nabla\phi_h - (u_h\times B),\\
R_{h,\phi} &:= \nabla\cdot j_h.
\end{aligned}
$$

The stabilization parameters have the following expressions within each element K:

$$
\begin{aligned}
&\alpha := c_1\tfrac{a}{h} + c_2\tfrac{1}{h^2\mathrm{Re}}, \quad \beta := c_3\mathrm{N}B, \quad \gamma := c_4,\\
&\tau_1 = \alpha^{-1}\left(1 + \tfrac{1}{\sqrt{\alpha\gamma}}\beta\right)^{-1}, \quad \tau_2 = c_5\tfrac{h^2}{\tau_1},\\
&\tau_3 = \gamma^{-1}\left(1 + \tfrac{1}{\sqrt{\alpha\gamma}}\beta\right)^{-1}, \quad \tau_4 = c_6\tfrac{h^2}{\tau_3},
\end{aligned}
$$

where $c_1, \ldots, c_6$ are algorithmic constants.

The second stabilization method developed for this work is the orthogonal sub-scale stabilization method (OSS) introduced in [46, 49]. We consider a term-by-term formulation, where we only introduce as stabilization terms the quantities we want to stabilize (first order derivative terms) scaled with properly chosen stabilization parameters. These terms alone would destroy the convergence properties of the resulting method. In order to have optimal convergence, we subtract to the quantities to be stabilized a proper projection onto the FE space. The OSS stabilization terms to be added to the Galerkin formulation read as (before linearization)

$$
\begin{aligned}
&(\tau_1\pi_{hu}^{\perp}((u_h\cdot\nabla)u_h), \pi_{hu}^{\perp}((u_h\cdot\nabla)v_h)) + (\tau_1\pi_{hu}^{\perp}(\nabla p_h), \pi_{hu}^{\perp}(\nabla q_h))\\
&+ (\tau_1\pi_{hu}^{\perp}(j_h\times B), \pi_{hu}^{\perp}(k_h\times B)) + (\tau_2\nabla\cdot u_h, \nabla\cdot v_h) + (\tau_3\pi_{hj}^{\perp}(\nabla\phi), \pi_{hj}^{\perp}(\nabla\psi))\\
&+ (\tau_3\pi_{hj}^{\perp}(u_h\times B), \pi_{hj}^{\perp}(v_h\times B)) + (\tau_4\nabla\cdot j_h, \nabla\cdot k_h),
\end{aligned}
$$

where $\pi_{hu}^{\perp}(\cdot) = \mathrm{Id}(\cdot) - \pi_{hu}(\cdot)$; Id denotes the identity and $\pi_{hu} : L^2(\Omega) \rightarrow V_{uh,0}$ corresponds to a projector onto the FE space without any boundary condition. $\pi_{hj}^{\perp}(\cdot)$ is defined analogously with respect to $V_{jh,0}$. We note that the projections are not required for the divergence terms, since both u and j are solenoidal. Different choices for the projector have been proposed so far, e.g., the orthogonal subscales (OSS) formulation considers the L^2 projector and a local nodal Scott-Zhang projector is used in [7]; other local projection stabilization methods can be found in [21, 94]. In this work, we consider the OSS formulation. In practice, the projection in the OSS method is treated explicitly, e.g.,

$$(\tau_1 \pi_{hu}^{\perp}(\nabla p_h), \pi_{hu}^{\perp}(\nabla q_h)) = (\tau_1 \nabla p_h, \nabla q_h) - (\tau_1 \pi_{hu}(\nabla p_h), \nabla q_h),$$

where the last term is treated explicitly, using the value from the previous nonlinear iteration (idem for the rest of terms). We note that the approximation comes from the fact that τ_1 is not constant in general (otherwise the previous re-statement will be exact). In order to make this relation hold for non-constant τ, we can use a τ-weighted L^2 projector (see [49]). With all these ingredients, we end up with the OSS formulation included in Algorithm 6.2.

Both ASGS and OSS methods can be stated as

$$F_u(a_h^{n+\alpha}) u_h^{n+\alpha} + C_u j_h^{n+1} + G_u p_h^{n+1} + T_u \phi_h^{n+1} = f_u^{n+\alpha}, \tag{6.21}$$

$$D_u u_h^{n+\alpha} + C_p p_h^{n+1} + T_p j_h^{n+1} = f_p^{n+1}, \tag{6.22}$$

$$F_j j_h^{n+1} + C_j u_h^{n+\alpha} + G_j \phi_h^{n+1} + T_j p_h^{n+1} = f_j^{n+1}, \tag{6.23}$$

$$D_j j_h^{n+1} + C_\phi \phi_h^{n+1} + T_\phi u_h^{n+1} = f_\phi^{n+1}, \tag{6.24}$$

where F_u, C_u, F_j and C_j have been properly modified (with respect to the original Galerkin formulation) in order to include the corresponding stabilization terms in Algorithms 6.1 or 6.2. Note that for the ASGS formulation, the right-hand-side terms f_p, f_j and f_ϕ are zero (for OSS, they include the projection treated explicitly) whereas for the OSS formulation, the operators T_u, T_p, T_j and T_ϕ are zero because there are no stabilization terms that couple u-ϕ and j-p. The discrete and stabilized formulation (6.21)-(6.24) results in a linear system of equations to be solved. This system of equations has a 4×4 block structure, where we consider one block per unknown:

$$\begin{bmatrix} F_u & G_u & C_u & T_u \\ D_u & C_p & T_p & 0 \\ C_j & T_j & F_j & G_j \\ T_\phi & 0 & D_j & C_\phi \end{bmatrix} \begin{bmatrix} u \\ p \\ j \\ \phi \end{bmatrix} = \begin{bmatrix} f_u \\ f_p \\ f_j \\ f_\phi \end{bmatrix}. \tag{6.25}$$

We further note that the OSS algorithm does not modify the off-diagonal terms, i.e., G_u, C_u, D_u, G_j, C_j, and D_j, keeping the block skew-symmetric nature of the Galerkin matrix. However, using ASGS these off-diagonal terms are perturbed in such a way that this property is lost. It has important effects as segregated algorithms are not unconditionally stable for non skew-symmetric matrices (see Chapter 4). Finally, note that for the sake of conciseness, Algorithms 6.1 and 6.2 have been written using the Backward Euler method ($\alpha = 1$) for time integration.

Algorithm 6.2: OSS stabilization for the inductionless MHD problem

Given u_h^n at the previous time step value, find u_h^{n+1}, j_h^{n+1}, p_h^{n+1} and ϕ_h^{n+1} such that

$$
\begin{aligned}
&\left(\delta_t u_h^{n+1,k+1}, v_h\right) + \left\langle (u_h^{n+1,k}\cdot\nabla)u_h^{n+1,k+1}, v_h\right\rangle + \tfrac{1}{\mathrm{Re}}\left(\nabla u_h^{n+1,k+1}, \nabla v_h\right) - \left(p_h^{n+1,k+1}, \nabla\cdot v_h\right)\\
&\quad -\mathrm{N}\left\langle j_h^{n+1,k+1}\times B, v_h\right\rangle + \left\langle (u_h^{n+1,k}\cdot\nabla)u_h^{n+1,k+1}, \tau_1^{n+1,k}(u_h^{n+1,k}\cdot\nabla)v_h\right\rangle\\
&\quad + \left\langle u_h^{n+1,k+1}\times B, \tau_3^{n+1,k}(v_h\times B)\right\rangle + \left(\nabla\cdot u_h^{n+1,k+1}, \tau_2^{n+1,k}\nabla\cdot v_h\right)\\
&\quad = \left(f^{n+1}, v_h\right) + \left\langle x_{1,h}^{n+1,k}, \tau_1^{n+1,k}(u_h^{n+1,k}\cdot\nabla)v_h\right\rangle - \left\langle y_{2,h}^{n+1,k}, \tau_3^{n+1,k}(v_h\times B)\right\rangle,\\
&\left(\nabla\cdot u_h^{n+1,k+1}, q_h\right) + \left(\nabla p_h^{n+1,k+1}, \tau_1^{n+1,k}\nabla q_h\right)\\
&\quad = \left\langle x_{2,h}^{n+1,k}, \tau_1^{n+1,k}\nabla q_h\right\rangle,\\
&\left(j_h^{n+1,k+1}, k_h\right) + \left(\nabla\phi_h^{n+1,k+1}, k_h\right) - \left\langle u_h^{n+1,k+1}\times B, k_h\right\rangle\\
&\quad +\mathrm{N}^2\left\langle j_h^{n+1,k+1}\times B, \tau_1^{n+1,k}(k_h\times B)\right\rangle + \left(\nabla\cdot j_h^{n+1,k+1}, \tau_4^{n+1,k}\nabla\cdot k_h\right)\\
&\quad = -\mathrm{N}\left\langle x_{3,h}^{n+1,k}, \tau_1^{n+1,k}(k_h\times B)\right\rangle,\\
&-\left(j_h^{n+1,k+1}, \nabla\psi_h\right) + \left(\nabla\phi_h^{n+1,k+1}, \tau_3^{n+1,k}\nabla\psi_h\right)\\
&\quad = \left(y_{1,h}^{n+1,k}, \tau_3^{n+1,k}\nabla\psi_h\right),
\end{aligned}
$$

where the projections are computed from:

$$
\begin{aligned}
\left(x_{1,h}^{n+1,k}, v_h\right) &= \left\langle (u_h^{n+1,k}\cdot\nabla)u_h^{n+1,k}, v_h\right\rangle,\\
\left(x_{2,h}^{n+1,k}, v_h\right) &= \left(\nabla p_h^{n+1,k}, v_h\right),\\
\left(x_{3,h}^{n+1,k}, v_h\right) &= -\mathrm{N}\left\langle j_h^{n+1,k}\times B, v_h\right\rangle,\\
\left(y_{1,h}^{n+1,k}, k_h\right) &= \left(\nabla\phi_h^{n+1,k}, k_h\right),\\
\left(y_{2,h}^{n+1,k}, k_h\right) &= -\left\langle u_h^{n+1,k}\times B, k_h\right\rangle.
\end{aligned}
$$

The stabilization parameters have the following expressions within each element K:

$$
\begin{aligned}
&\alpha := c_1\tfrac{a}{h} + c_2\tfrac{1}{h^2\mathrm{Re}}, \quad \beta := c_3\mathrm{N}B, \quad \gamma := c_4,\\
&\tau_1 = \alpha^{-1}\left(1+\tfrac{1}{\sqrt{\alpha\gamma}}\beta\right)^{-1}, \quad \tau_2 = c_5\tfrac{h^2}{\tau_1},\\
&\tau_3 = \gamma^{-1}\left(1+\tfrac{1}{\sqrt{\alpha\gamma}}\beta\right)^{-1}, \quad \tau_4 = c_6\tfrac{h^2}{\tau_3},
\end{aligned}
$$

where $c_1,\ldots,c_6$ are algorithmic constants.

6.3.2 Thermally coupled problem

The weak form of the thermally coupled inductionless MHD problem (6.16)-(6.18) is linearized and discretized in both time and space following the same ideas exposed in Section 6.3.1. The only additional nonlinear term corresponds to the convective term in the temperature equation. This term is also linearized using Picard method, which leads

to the discrete and linearized form,

$$F_u(a_h^{n+\alpha})u_h^{n+\alpha} + C_u j_h^{n+1} + G_u p_h^{n+1} + H_u \theta_h^{n+\alpha} = f_u^{n+\alpha} \quad \text{in} \quad V'_{uh}, \tag{6.26}$$

$$D_u u_h^{n+\alpha} = 0 \quad \text{in} \quad Q'_{ph}, \tag{6.27}$$

$$F_j j_h^{n+1} + C_j u_h^{n+\alpha} + G_j \phi_h^{n+1} = 0 \quad \text{in} \quad V'_{jh}, \tag{6.28}$$

$$D_j j_h^{n+1} = 0 \quad \text{in} \quad Q'_{\phi h}, \tag{6.29}$$

$$F_\theta(a_h^{n+\alpha})\theta_h^{n+\alpha} = 0 \quad \text{in} \quad V'_{\theta h}, \tag{6.30}$$

where the discrete FE space $V_{\theta h}$ is a subspace of V_θ.

The Galerkin approximation (6.26)-(6.30) of the problem adds another source of instability, i.e., the presence of the convective term in the temperature equation. This term may introduce oscillations in the solution for convection dominated cases. Thus, we include a SUPG-type stabilization for the thermal problem in the ASGS formulation for the thermally coupled MHD system, i.e.,

$$(\tau_5(u_h \cdot \nabla)\varphi_h, \delta_t\theta_h + (a \cdot \nabla)\theta_h - \frac{1}{\mathrm{Pe}}\Delta_h\theta_h - Q).$$

The thermal coupling in system (6.26)-(6.30), after Picard linearization, is in one direction only; the thermal subproblem is independent of the fluid subproblem. Using a full ASGS formulation, this very interesting property would be lost. The final system with the definition of the stabilization parameters is included in Algorithm 6.3.

On the other hand, we can also use the OSS technique explained in the previous section to the thermally coupled system (6.26)-(6.30). In this case, we simply need to add the term

$$(\tau_5\pi^\perp_{h\theta}((u_h \cdot \nabla)\phi_h), \pi^\perp_{h\theta}((u_h \cdot \nabla)\psi_h)),$$

where $\pi^\perp_{h\theta}$ is the L^2-projection onto the FE space without boundary conditions $V_{\theta h,0}$. The resulting OSS algorithm, after time integration and linearization, is stated in Algorithm 6.4.

The addition of the new variational forms due to the thermal coupling and its associated stabilization terms can be stated compactly as the following 5×5 block linear system of equations,

$$\begin{bmatrix} F_u & G_u & C_u & T_u & H_u \\ D_u & C_p & T_p & 0 & H_p \\ C_j & T_j & F_j & G_j & H_j \\ T_\phi & 0 & D_j & C_\phi & 0 \\ 0 & 0 & 0 & 0 & F_\theta \end{bmatrix} \begin{bmatrix} u \\ p \\ j \\ \phi \\ \theta \end{bmatrix} = \begin{bmatrix} f_u \\ f_p \\ f_j \\ f_\phi \\ f_\theta \end{bmatrix}, \tag{6.31}$$

where the matrices have been modified accordingly to include the stabilization terms. Note that the operators H_p and H_j are zero for the OSS stabilized formulation because there are not coupling terms between p-θ and j-θ.

Algorithm 6.3: ASGS stabilization for the thermally coupled inductionless MHD problem

Given u_h^n at the previous time step value, find u_h^{n+1}, j_h^{n+1}, p_h^{n+1} and ϕ_h^{n+1} such that

$$
\begin{aligned}
&\left(\delta_t u_h^{n+1,k+1}, v_h\right) + \left\langle (u_h^{n+1,k}\cdot\nabla)u_h^{n+1,k+1}, v_h\right\rangle + \tfrac{1}{\mathrm{Re}}\left(\nabla u_h^{n+1,k+1}, \nabla v_h\right)\\
&\quad - \left(p_h^{n+1,k+1}, \nabla\cdot v_h\right) - \mathrm{N}\left\langle j_h^{n+1,k+1}\times B, v_h\right\rangle + \tfrac{\mathrm{Gr}}{\mathrm{Re}^2}\left(\theta_h^{n+1,k+1}, v_h\right)\\
&\quad + \left\langle (u_h^{n+1,k}\cdot\nabla)v_h + \tfrac{1}{\mathrm{Re}}\Delta_h v_h, \tau_1^{n+1,k}\boldsymbol{R}_{h,u}^{n+1,k+1}\right\rangle + \left(\nabla\cdot v_h, \tau_2^{n+1,k}R_{h,p}^{n+1,k+1}\right)\\
&\quad - \left\langle (v_h\times B), \tau_3^{n+1,k}\boldsymbol{R}_{h,j}^{n+1,k+1}\right\rangle = \left(f^{n+1}, v_h\right) + \tfrac{\mathrm{Gr}}{\mathrm{Re}^2}\left(\theta_{ref}, v_h\right),\\
&\left(\nabla\cdot u_h^{n+1,k+1}, q_h\right) + \left\langle \tau_1^{n+1,k}\boldsymbol{R}_{h,u}^{n+1,k+1}, \nabla q_h\right\rangle = 0,\\
&\left(j_h^{n+1,k+1}, k_h\right) + \left(\nabla\phi_h^{n+1,k+1}, k_h\right) - \left\langle u_h^{n+1,k+1}\times B, k_h\right\rangle\\
&\quad - \mathrm{N}\left\langle k_h\times B, \tau_1^{n+1,k}\boldsymbol{R}_{h,u}^{n+1,k+1}\right\rangle - \left\langle k_h, \tau_3^{n+1,k}\boldsymbol{R}_{h,j}^{n+1,k+1}\right\rangle\\
&\quad + \left(\nabla\cdot k_h, \tau_4^{n+1,k}R_{h,\phi}^{n+1,k+1}\right) = 0,\\
&-\left(j_h^{n+1,k+1}, \nabla\psi_h\right) + \left\langle \nabla\psi_h, \tau_3^{n+1,k}\boldsymbol{R}_{h,j}^{n+1,k+1}\right\rangle = 0,\\
&\left(\delta_t\theta_h^{n+1,k+1}, \varphi_h\right) + \left\langle (u_h^{n+1,k}\cdot\nabla)\theta_h^{n+1,k+1}, \varphi_h\right\rangle + \tfrac{1}{\mathrm{Pe}}\left(\nabla\theta_h^{n+1,k+1}, \nabla\varphi_h\right)\\
&\quad + \left\langle (u_h^{n+1,k}\cdot\nabla)\varphi_h, \tau_5^{n+1,k}\boldsymbol{R}_{h,\theta}^{n+1,k+1}\right\rangle = \left(Q^{n+1}, \varphi_h\right).
\end{aligned}
$$

where the residuals are:

$$
\begin{aligned}
\boldsymbol{R}_{h,u} &:= \delta_t u_h + (a\cdot\nabla)u_h - \tfrac{1}{\mathrm{Re}}\Delta_h u_h + \nabla p_h - \mathrm{N}(j_h\times B) + \tfrac{\mathrm{Gr}}{\mathrm{Re}^2}\theta_h - f,\\
R_{h,p} &:= \nabla\cdot u_h,\\
\boldsymbol{R}_{h,j} &:= j_h + \nabla\phi_h - (u_h\times B),\\
R_{h,\phi} &:= \nabla\cdot j_h,\\
R_{h,\theta} &:= \delta_t\theta_h + (a\cdot\nabla)\theta_h - \tfrac{1}{\mathrm{Pe}}\Delta_h\theta_h - Q.
\end{aligned}
$$

The stabilization parameters have the following expressions within each element K:

$$
\begin{aligned}
&\alpha := c_1\tfrac{a}{h} + c_2\tfrac{1}{h^2\mathrm{Re}}, \quad \beta := c_3\mathrm{N}B, \quad \gamma := c_4,\\
&\tau_1 = \alpha^{-1}\left(1 + \tfrac{1}{\sqrt{\alpha\gamma}}\beta\right)^{-1}, \quad \tau_2 = c_5\tfrac{h^2}{\tau_1},\\
&\tau_3 = \gamma^{-1}\left(1 + \tfrac{1}{\sqrt{\alpha\gamma}}\beta\right)^{-1}, \quad \tau_4 = c_6\tfrac{h^2}{\tau_3},\\
&\tau_5 = \left(c_7\tfrac{a}{h} + c_8\tfrac{1}{h^2\mathrm{Pe}}\right)^{-1},
\end{aligned}
$$

where $c_1, \ldots, c_8$ are algorithmic constants.

6.4 Block recursive preconditioners for the inductionless MHD problem

In this section, we design block preconditioners for multiphysics problems based on a recursive use of inexact block *LU* factorization. This strategy is first presented in a general (abstract) form. The key ingredient of this formulation is the approximation of the Schur complements. Next, we list typical Schur complement approximations for the Navier-Stokes equations. Finally, we apply the abstract setting for the (thermally coupled) inductionless MHD problem.

Algorithm 6.4: OSS stabilization for the thermally coupled inductionless MHD problem

Given u_h^n at the previous time step value, find u_h^{n+1}, j_h^{n+1}, p_h^{n+1} and ϕ_h^{n+1} such that

$$
\begin{aligned}
&\left(\delta_t u_h^{n+1,k+1}, v_h\right) + \left\langle (u_h^{n+1,k}\cdot\nabla)u_h^{n+1,k+1}, v_h\right\rangle + \tfrac{1}{\mathrm{Re}}\left(\nabla u_h^{n+1,k+1}, \nabla v_h\right) - \left(p_h^{n+1,k+1}, \nabla\cdot v_h\right)\\
&\quad -\mathrm{N}\left\langle j_h^{n+1,k+1}\times B, v_h\right\rangle + \tfrac{\mathrm{Gr}}{\mathrm{Re}^2}\left(\theta_h^{n+1,k+1}, v_h\right) + \left\langle (u_h^{n+1,k}\cdot\nabla)u_h, \tau_1^{n+1,k}(u_h^{n+1,k}\cdot\nabla)v_h\right\rangle\\
&\quad + \left\langle u_h^{n+1,k+1}\times B, \tau_3^{n+1,k}(v_h\times B)\right\rangle + \left(\nabla\cdot u_h^{n+1,k+1}, \tau_2^{n+1,k}\nabla\cdot v_h\right)\\
&\quad = \left(f^{n+1}, v_h\right) + \left\langle x_{1,h}^{n+1,k}, \tau_1^{n+1,k}(u_h^{n+1,k}\cdot\nabla)v_h\right\rangle - \left\langle y_{2,h}^{n+1,k}, \tau_3^{n+1,k}(v_h\times B)\right\rangle,\\
&\left(\nabla\cdot u_h^{n+1,k+1}, q_h\right) + \left(\nabla p_h^{n+1,k+1}, \tau_1^{n+1,k}\nabla q_h\right)\\
&\quad = \left(x_{2,h}^{n+1,k}, \tau_1^{n+1,k}\nabla q_h\right),\\
&\left(j_h^{n+1,k+1}, k_h\right) + \left(\nabla\phi_h^{n+1,k+1}, k_h\right) - \left\langle u_h^{n+1,k+1}\times B, k_h\right\rangle\\
&\quad +\mathrm{N}^2\left\langle j_h^{n+1,k+1}\times B, \tau_1^{n+1,k}(k_h\times B)\right\rangle + \left(\nabla\cdot j_h^{n+1,k+1}, \tau_4^{n+1,k}\nabla\cdot k_h\right)\\
&\quad = -\mathrm{N}\left\langle x_{3,h}^{n+1,k}, \tau_1^{n+1,k}(k_h\times B)\right\rangle,\\
&-\left(j_h^{n+1,k+1}, \nabla\psi_h\right) + \left(\nabla\phi_h^{n+1,k+1}, \tau_3^{n+1,k}\nabla\psi_h\right)\\
&\quad = \left(y_{1,h}^{n+1,k}, \tau_3^{n+1,k}\nabla\psi_h\right),\\
&\left(\delta_t\theta_h^{n+1,k+1}, \varphi_h\right) + \left\langle (u_h^{n+1,k}\cdot\nabla)\theta_h^{n+1,k+1}, \varphi_h\right\rangle + \tfrac{1}{\mathrm{Pe}}\left(\nabla\theta_h^{n+1,k+1}, \nabla\varphi_h\right)\\
&\quad + \left\langle (u_h^{n+1,k}\cdot\nabla)\theta_h, \tau_5^{n+1,k}(u_h^{n+1,k}\cdot\nabla)\varphi_h\right\rangle\\
&\quad = \left(Q^{n+1}, \varphi_h\right) + \left\langle z_{1,h}^{n+1,k}, \tau_5^{n+1,k}(u_h^{n+1,k}\cdot\nabla)\varphi_h\right\rangle,
\end{aligned}
$$

where the projections are computed from:

$$
\begin{aligned}
\left(x_{1,h}^{n+1,k}, v_h\right) &= \left\langle (u_h^{n+1,k}\cdot\nabla)u_h^{n+1,k}, v_h\right\rangle,\\
\left(x_{2,h}^{n+1,k}, v_h\right) &= \left(\nabla p_h^{n+1,k}, v_h\right),\\
\left(x_{3,h}^{n+1,k}, v_h\right) &= -\mathrm{N}\left\langle j_h^{n+1,k}\times B, v_h\right\rangle,\\
\left(y_{1,h}^{n+1,k}, k_h\right) &= \left(\nabla\phi_h^{n+1,k}, k_h\right),\\
\left(y_{2,h}^{n+1,k}, k_h\right) &= -\left\langle u_h^{n+1,k}\times B, k_h\right\rangle,\\
\left(z_{1,h}^{n+1,k}, \varphi_h\right) &= \left\langle (u_h^{n+1,k}\cdot\nabla)\theta_h^{n+1,k}, \varphi_h\right\rangle.
\end{aligned}
$$

The stabilization parameters have the following expressions within each element K:

$$
\begin{aligned}
&\alpha := c_1\tfrac{a}{h} + c_2\tfrac{1}{h^2\mathrm{Re}}, \quad \beta := c_3\mathrm{N}B, \quad \gamma := c_4,\\
&\tau_1 = \alpha^{-1}\left(1 + \tfrac{1}{\sqrt{\alpha\gamma}}\beta\right)^{-1}, \quad \tau_2 = c_5\tfrac{h^2}{\tau_1},\\
&\tau_3 = \gamma^{-1}\left(1 + \tfrac{1}{\sqrt{\alpha\gamma}}\beta\right)^{-1}, \quad \tau_4 = c_6\tfrac{h^2}{\tau_3},\\
&\tau_5 = \left(c_7\tfrac{a}{h} + c_8\tfrac{1}{h^2\mathrm{Pe}}\right)^{-1},
\end{aligned}
$$

where $c_1, \ldots, c_8$ are algorithmic constants.

6.4.1 Abstract block recursive factorization

Let us consider a generic system

$$\begin{bmatrix} A_{11} & \cdots & A_{1n_{\text{unk}}} \\ \vdots & \ddots & \vdots \\ A_{n_{\text{unk}}1} & \cdots & A_{n_{\text{unk}}n_{\text{unk}}} \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_{n_{\text{unk}}} \end{bmatrix} = \begin{bmatrix} b_1 \\ \vdots \\ b_{n_{\text{unk}}} \end{bmatrix}$$

arising from the discretization of a multiphysics problem that involves n_{unk} physical variables. Our target is to design an efficient preconditioner for the system matrix A such that it only involves the solution of one-variable (one-physics) problems, for which we can find efficient preconditioners of domain decomposition or algebraic multigrid type. In order to do this, we rely on the incomplete block factorization of a 2×2 block matrix. Obviously, the original multiphysics problem can be arranged as a 2×2 system matrix by splitting (and reordering) the n_{unk} variables into two different ordered sets. After this, the original problem is denoted as:

$$\begin{bmatrix} F & G \\ D & C \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix}.$$

We have denoted the block matrices and vectors in the arranged 2×2 block system using the typical notation for the incompressible Navier-Stokes equations. (We do not assume that $D = -G^T$ since it is not true in general, e.g., when solving transient incompressible flows with SUPG-type stabilization techniques.) For saddle-point problems, y usually is a Lagrange multiplier, e.g., the pressure. At this point, we can consider an *exact* block LU factorization of the 2×2 block matrix:

$$A = \begin{bmatrix} F & G \\ D & C \end{bmatrix} = \begin{bmatrix} F & 0 \\ D & S \end{bmatrix} \begin{bmatrix} I & F^{-1}G \\ 0 & I \end{bmatrix},$$

where $S = C - DF^{-1}G$ is the Schur complement matrix with respect to y. This matrix cannot be easily handled. In order to obtain an inexact factorization, the key aspect is the design of a good approximation for the Schur complement matrix. We denote this approximation by $S_\sharp$. We can further consider an approximation of F, which we denote by $F_\sharp$, even though it is not essential in many cases. With these two ingredients, namely $S_\sharp$ and $F_\sharp$, we can consider different preconditioners $P(A)$ for A:

$$D: \qquad P(A) = \begin{bmatrix} F_\sharp & 0 \\ 0 & S_\sharp \end{bmatrix}^{-1} = \begin{bmatrix} F_\sharp^{-1} & 0 \\ 0 & S_\sharp^{-1} \end{bmatrix}, \tag{6.32}$$

$$U: \qquad P(A) = \begin{bmatrix} F_\sharp & G \\ 0 & S_\sharp \end{bmatrix}^{-1} = \begin{bmatrix} F_\sharp^{-1} & -F_\sharp^{-1}GS_\sharp^{-1} \\ 0 & S_\sharp^{-1} \end{bmatrix}, \tag{6.33}$$

$$LU: \quad P(A) = \begin{bmatrix} I & F_\sharp^{-1}G \\ 0 & I \end{bmatrix}^{-1} \begin{bmatrix} F & 0 \\ D & S_\sharp \end{bmatrix}^{-1} = \begin{bmatrix} I & -F_\sharp^{-1}G \\ 0 & I \end{bmatrix} \begin{bmatrix} F^{-1} & 0 \\ -S_\sharp^{-1}DF^{-1} & S_\sharp^{-1} \end{bmatrix}. \tag{6.34}$$

Since we aim at solving the global problem using a preconditioned Krylov iterative solver, we only require to perform matrix-vector multiplications for both A and the preconditioner $P(A)$. The preconditioner $P(A)$ is defined by the inverses of the matrices on the diagonal, i.e., $F_\sharp^{-1}$, $S_\sharp^{-1}$, and possibly F^{-1}. For practical problems, the computation of the action of $F_\sharp^{-1}$ or $S_\sharp^{-1}$ on a vector is not viable via sparse direct solvers, specially in 3D, due to their high memory and computational demands. We use the following notation: given a matrix H, the approximate action of H^{-1} over a vector computed by a Krylov iterative solver preconditioned with $P(H)$ up to a given tolerance tol_H is denoted by $\texttt{precond\_Krylov}(H, P(H), \mathrm{tol}_H)$. Thus, in practical implementations, we replace $F_\sharp^{-1}$ by $\texttt{precond\_Krylov}(F_\sharp, P(F_\sharp), \mathrm{tol})$ in the definition (6.32),(6.33) or (6.34) of the preconditioner (analogously for $S_\sharp^{-1}$ and possibly F^{-1}).

In this setting, with a particular definition of the block matrix and its approximations, we can recover most of the Schur complement preconditioners in the literature (see Section 6.4.2). For one-physics problems with a saddle-point structure (e.g., the incompressible (Navier)-Stokes equations, electromagnetics with Lorentz gauge, mixed form of Laplacian-type problems, Darcy's law for flow in porous media) matrices $F_\sharp$ and $S_\sharp$ involve one-variable problems (for the field and the Lagrange multiplier respectively) but this is not the general case for multiphysics problems. However, if, e.g., $F_\sharp$ involves two or more variables, we can perform again an incomplete block factorization of this matrix, i.e., approximate $F_\sharp^{-1}$ by $P(F_\sharp)$ with one of the definitions in (6.32),(6.33) or (6.34) (analogously for $S_\sharp^{-1}$ and F^{-1}). This process can be applied recursively till all diagonal block matrices (to be *inverted*) only involve one variable. We state in Algorithm 6.5 the definition of the *recursive block LU preconditioner*. (We have assumed in this algorithm that the system has been ordered in such a way that the Schur complement is always defined for the second block unknown.) Let us note that it is not required to end the process when the diagonal system matrices are one-variable matrices, as soon as we have at our disposal an efficient preconditioner for a particular multi-variable matrix.

Algorithm 6.5: $P = \texttt{LU\_block\_precond}(A)$

1: Define a 2×2 block partition of the system matrix (into subsets of physical variables):

$$A = \begin{bmatrix} F & G \\ D & C \end{bmatrix}$$

2: Define the approximations $S_\sharp \approx C - DF^{-1}G$ and $F_\sharp \approx F$
3: **for** $H = \{S_\sharp,\ F_\sharp$ *(and possibly* F*)* $\}$ **do**
4: | **if** H *involves more than one physical variable* **then**
5: | | Replace H^{-1} by its LU approximation, i.e., $H^{-1} \leftarrow \texttt{LU\_block\_precond}(H)$
6: | **else**
7: | | Define an effective preconditioner $P(H)$ (e.g., using DDM, multigrid...)
8: | | Define the tolerance tol_H and replace $H^{-1} \leftarrow \texttt{precond\_Krylov}(H, P(H), \mathrm{tol}_H)$
9: Define $P(A)$ as in (6.32), (6.33) or (6.34)

As an alternative, we can also consider only the LU factorization at the first level, and

solve every multi-variable diagonal block using a Krylov iterative solver preconditioned with an incomplete inexact LU factorization. Again, we can proceed in a recursive way. As a result, this procedure will involve as many nested iterative loops as levels of recursion. The resulting algorithm is presented in Algorithm 6.6. Both approaches will be considered in the numerical experiments section.

Algorithm 6.6: $P = \texttt{LU\_approximation}(A)$

1: Define a 2×2 block partition of the system matrix (into subsets of physical variables):

$$A = \begin{bmatrix} F & G \\ D & C \end{bmatrix}$$

2: Define the approximations $S_\sharp \approx C - DF^{-1}G$ and $F_\sharp \approx F$
3: **for** $H = \{S_\sharp,\ F_\sharp$ *(and possibly* F*)* $\}$ **do**
4: | **if** H *involves more than one variable* **then**
5: | | Define the LU factorization of H: $P(H) \leftarrow \texttt{LU\_approximation}(H)$
6: | | Define the tolerance tol_H and replace $H^{-1} \leftarrow \texttt{precond\_Krylov}(H, P(H), \text{tol}_H)$
7: | **else**
8: | | Define an effective preconditioner $P(H)$ (e.g., using DDM, multigrid...)
9: | | Define the tolerance tol_H and replace $H^{-1} \leftarrow \texttt{precond\_Krylov}(H, P(H), \text{tol}_H)$
10: Define $P(A)$ as in (6.32), (6.33) or (6.34)

For a particular multiphysics problem the key ingredient to be defined is an efficient and robust approximation of $S_\sharp$ for every incomplete LU factorization.

6.4.2 Incompressible Navier-Stokes preconditioners

Let us review some of the classical block preconditioners for solving the incompressible Navier-Stokes problem. Consider that the linear system of equations to be solved after discretization and stabilization of the Navier-Stokes equations is written as

$$\begin{bmatrix} F & G \\ D & C \end{bmatrix} \begin{bmatrix} u \\ p \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix}.$$

First, let us state the Uzawa method as a U-preconditioner (6.33) for solving the stationary Stokes problem. Considering that the block matrix $F = K$, i.e., it only contains diffusive terms, the exact Schur complement for the pressure is $S = C - DK^{-1}G$. However, at the continuous level, $DK^{-1}G = \nabla^T \Delta_0^{-1} \nabla$, where Δ_0^{-1} denotes the inverse of the Laplace problem with homogeneous Dirichlet boundary conditions on $\Gamma_{E,u}$. We consider the approximation $\nabla^T \Delta_0^{-1} \nabla \approx 1$; it is exact for periodic boundary conditions. As a result, we approximate $DK^{-1}G \approx \text{Re}\, M_p$, where M_p is a mass matrix for the pressure. This way, we can write the Uzawa block preconditioner as,

$$P_{\text{Uzw}}(A) = \begin{bmatrix} F_\sharp & G \\ 0 & S_\sharp \end{bmatrix}^{-1}, \text{ where } \begin{array}{l} F_\sharp^{-1} = F^{-1} \\ S_\sharp^{-1} = C^{-1} + \frac{1}{\text{Re}}\, M_p^{-1} \end{array}. \tag{6.35}$$

Note that the expression of the Schur complement approximation also involves the stabilization matrix inverse.

Cahouet and Chabard extended these ideas to the transient Stokes problem in [38]. In this case, the block matrix $F = \frac{1}{\delta t}M + K$ contains the temporal and diffusive terms. If we take into account that $-D(\frac{1}{\delta t}M)^{-1}G$ is spectrally equivalent to a Laplacian matrix, i.e., $-D(\frac{1}{\delta t}M)^{-1}G \approx \delta t\, L$ (L denotes the typical FE discretization for $-\Delta$), the Cahouet-Chabard (CC) block preconditioner is written as,

$$P_{CC}(A) = \begin{bmatrix} F_\sharp & G \\ 0 & S_\sharp \end{bmatrix}^{-1}, \text{ where } \begin{array}{l} F_\sharp^{-1} = F^{-1} \\ S_\sharp^{-1} = (C + \delta t L)^{-1} + \frac{1}{\mathrm{Re}}\, M_p^{-1} \end{array} . \tag{6.36}$$

In order to introduce the convective term into the preconditioner, the pressure convection-diffusion (PCD) preconditioner was developed in [66,67]. This preconditioner is based on approximating the original Schur complement by a commutation of operators, viz. $\nabla^T \mathcal{L}^{-1} \nabla \approx \nabla^T \nabla \mathcal{L}_p^{-1}$, where $\mathcal{L}(\cdot) = \frac{1}{\delta t}(\cdot) + a \cdot \nabla(\cdot) - \frac{1}{\mathrm{Re}}\Delta(\cdot)$ is a CDR operator and $\mathcal{L}_p$ a pressure CDR operator. If we apply this approximation to the discrete level, it leads to $DF^{-1}G \approx D(M^{-1}F)^{-1}M^{-1}G \approx DM^{-1}G(M_p^{-1}F_p)^{-1} \approx L_p F_p^{-1} M_p$, where F_p is the matrix obtained after discretization of the pressure CDR operator $\mathcal{L}_p$. Therefore, the expression of the PCD preconditioner is,

$$P_{PCD}(A) = \begin{bmatrix} F_\sharp & G \\ 0 & S_\sharp \end{bmatrix}^{-1}, \text{ where } \begin{array}{l} F_\sharp^{-1} = F^{-1} \\ S_\sharp^{-1} = M_p^{-1} F_p L_p^{-1} \end{array} . \tag{6.37}$$

Finally, other classical and well-known algorithms for solving the transient Navier-Stokes equations are the pressure segregation (PC) methods, also known as fractional step schemes. They were first developed independently by Chorin [42,43] and Temam [116]. Basically, they consist of two steps. First, an intermediate velocity that does not verify the incompressibility condition is obtained from the momentum equation. Then, an end-of-step velocity is computed taking into account the pressure gradient and the velocity divergence terms. These schemes assume that the convective and diffusive terms are negligible with respect to the temporal evolutionary term and therefore $F \approx \frac{1}{\delta t}M$ and $-DF^{-1}G \approx -D(\frac{1}{\delta t}M)^{-1}G \approx \delta t\, L$, which is a reasonable assumption for δt small. They can be implemented as a LU-preconditioner

$$P_{PC}(A) = \begin{bmatrix} I & F_\sharp^{-1}G \\ 0 & I \end{bmatrix}^{-1} \begin{bmatrix} F & 0 \\ D & S_\sharp \end{bmatrix}^{-1}, \text{ where } \begin{array}{l} F_\sharp^{-1} = \delta t M^{-1} \\ S_\sharp^{-1} = (C + \delta t L)^{-1} \end{array} . \tag{6.38}$$

6.4.3 Incompressible inductionless MHD preconditioners

In this section, we consider two different preconditioners for the inductionless MHD problem, based on the recursive block LU factorization introduced above. The first preconditioner is based on an initial factorization of the system matrix into fluid and magnetic subproblems. The second preconditioner segregates at the first level field variables (velocity and current) from Lagrange multiplier-type variables (pressure and electric potential).

Fluid-magnetic subproblem factorization (FMS preconditioner)

Let us consider the 4×4 block system (6.25) and reorder it in such a way that the electromagnetic variables are written first and then the fluid unknowns:

$$A = \left[\begin{array}{cc|cc} F_j & G_j & C_j & T_j \\ D_j & C_\phi & T_\phi & 0 \\ \hline C_u & T_u & F_u & G_u \\ T_p & 0 & D_u & C_p \end{array}\right] \left[\begin{array}{c} j \\ \phi \\ \hline u \\ p \end{array}\right] = \left[\begin{array}{c} f_j \\ f_\phi \\ \hline f_u \\ f_p \end{array}\right]. \tag{6.39}$$

Following Algorithms 6.5 and 6.6, we arrange the 4×4 block matrix A from (6.39) as a 2×2 block system grouping together the electromagnetic unknowns, j-ϕ, on one hand and the fluid unknowns, u-p, on the other hand. Moreover, we choose the preconditioner P_{FMS} as a U-preconditioner (6.33),

$$\left[\begin{array}{cc} F_{j\phi} & C_{j\phi} \\ C_{up} & F_{up} \end{array}\right] \left[\begin{array}{c} x \\ y \end{array}\right] = \left[\begin{array}{c} f \\ g \end{array}\right], \quad P_{FMS}(A) = \left[\begin{array}{cc} F_{j\phi} & C_{j\phi} \\ 0 & S_{up\sharp} \end{array}\right]^{-1}, \tag{6.40}$$

where $x = [j, \phi]^T$, $y = [u, p]^T$, $f = [f_j, f_\phi]^T$, $g = [f_u, f_p]^T$ and matrices $F_{j\phi}$, F_{up}, $C_{j\phi}$ and C_{up} are the corresponding 2×2 block matrices. The key aspect to derive the preconditioner is the definition of the approximation for $S_{up\sharp} = F_{up} - C_{up}F_{j\phi}^{-1}C_{j\phi}$. We have chosen $S_{up\sharp} = F_{uj}$. Let us justify this (simple) choice for the Schur complement approximation. It comes from the analysis of the continuous problem (6.5)-(6.8). We consider the electromagnetic subproblem (6.7)-(6.8). From (6.8), we have that $\phi = \Delta_0^{-1}(\nabla \cdot (u \times B))$, where Δ_0^{-1} is the inverse of the Laplacian with homogeneous boundary conditions on $\Gamma_{C,j}$. Invoking this expression of ϕ in (6.7) we obtain:

$$j = -\nabla\Delta_0^{-1}(\nabla \cdot (u \times B)) + (u \times B).$$

Using this expression in the momentum equation (6.5), we get

$$\partial_t u + (u \cdot \nabla)u - \frac{1}{\mathrm{Re}}\Delta u + \nabla p + \mathrm{N}((\nabla\Delta^{-1}\nabla \cdot (u \times B) - (u \times B)) \times B) = f.$$

Now, we note that the coupling term from the magnetic problem to the fluid problem, i.e., $((\nabla\Delta^{-1}\nabla \cdot (u \times B)) - (u \times B))$, vanishes when we consider the same approximation as for the Uzawa algorithm, $\nabla\Delta^{-1}\nabla\cdot \approx I$. This derivation suggests that the subproblems for the Navier-Stokes (NSI) and Darcy-type (DCY) equations can be solved uncoupled at the preconditioner level. This is certainly an approximation, but this kind of approximation has been proved to be optimal for the Stokes problem [67], leading to robust preconditioners. No approximation of $F_{j\phi}$ is taken at this level.

The following step in the preconditioner definition consists of adding a second level of recursion with the approximation of the inverses of block matrices $F_{j\phi}$ and F_{up}. For instance, let us replace them by a U-preconditioner, $F_{j\phi}^{-1} \leftarrow$ `LU_block_precond`$(F_{j\phi})$ and $F_{up}^{-1} \leftarrow$ `LU_block_precond`(F_{up}),

$$P(F_{j\phi}) = \left[\begin{array}{cc} F_j & G_j \\ 0 & S_{\phi\sharp} \end{array}\right]^{-1}, \qquad P(F_{up}) = \left[\begin{array}{cc} F_u & G_u \\ 0 & S_{p\sharp} \end{array}\right]^{-1}.$$

The last missing ingredient is to define the approximation of $S_{\phi\sharp}$ and $S_{p\sharp}$. For $S_{p\sharp}$, we can consider any of the methods in Section 6.4.2. On the other hand, since $S_\phi = C_\phi - D_\phi F_j^{-1} G_j$ and F_j is a mass matrix for the Galerkin approximation, we can naturally use the same approximation as for PC or PCD problems. Summarizing, we consider the following options:

$$\text{Uzawa}: \left\{ \begin{array}{l} S_{p\sharp}^{-1} = C_p^{-1} + \frac{1}{\mathrm{Re}} M_p^{-1} \\ S_{\phi\sharp}^{-1} = (C_\phi + L_\phi)^{-1} \end{array} \right. , \tag{6.41}$$

$$CC: \left\{ \begin{array}{l} S_{p\sharp}^{-1} = (C_p + \delta t L_p)^{-1} + \frac{1}{\mathrm{Re}} M_p^{-1} \\ S_{\phi\sharp}^{-1} = (C_\phi + L_\phi)^{-1} \end{array} \right. , \tag{6.42}$$

$$PCD: \left\{ \begin{array}{l} S_{p\sharp}^{-1} = M_p^{-1} F_p L_p^{-1} \\ S_{\phi\sharp}^{-1} = M_\phi^{-1} F_\phi L_\phi^{-1} \end{array} \right. \text{where} \quad \begin{array}{l} F_p(p,q) = \frac{1}{\delta t}(p,q) + ((u\cdot\nabla)p, q) + \frac{1}{\mathrm{Re}}(\nabla p, \nabla q) \\ F_\phi(\phi,\psi) = (\phi,\psi) \end{array} , \tag{6.43}$$

$$PC: \left\{ \begin{array}{l} S_{p\sharp}^{-1} = (C_p + \delta t L_p)^{-1} \\ S_{\phi\sharp}^{-1} = (C_\phi + L_\phi)^{-1} \end{array} \right. . \tag{6.44}$$

Note that for using the PC method, we have to define $P_{FMS}(A)$ in (6.40) as a *LU*-preconditioner (6.34). We do not write the details here for the sake of conciseness. Finally, we can write the 4×4 block preconditioner $P_{FMS}(A)$ for (6.39) as

$$P_{FMS}(A) = \left[\begin{array}{cc|cc} F_j & G_j & C_j & T_j \\ 0 & S_\phi & T_\phi & 0 \\ \hline 0 & 0 & F_u & G_u \\ 0 & 0 & 0 & S_p \end{array} \right]^{-1} . \tag{6.45}$$

Summarizing, we have defined a recursive block preconditioner P_{FMS} that allows us to decouple the computation of a multiphysics problem such as the inductionless MHD problem into one-physics problems for every physical variable (velocity, pressure, current density and electric potential) at the preconditioner level. Figure 6.1 displays a tree diagram to highlight how the coupled problem is uncoupled into one-physics problems along the two levels of recursion. First, the preconditioner splits the magnetic unknowns from the fluid ones whereas the second level allows us to decouple the computation of j and ϕ on one hand, and the computation of u and p on the other hand. At every splitting, we create two new problems, namely the F and S problems, and we have defined how to approximate every one of these matrices.

Field-Lagrange multiplier factorization (FLM preconditioner)

Let us consider now a different reordering of system (6.25) where the vectorial fields u and j are written first and the Lagrange multipliers p and ϕ next,

$$\left[\begin{array}{cc|cc} F_u & C_u & G_u & T_u \\ C_j & F_j & T_j & G_j \\ \hline D_u & T_p & C_p & 0 \\ T_\phi & D_j & 0 & C_\phi \end{array} \right] \left[\begin{array}{c} u \\ j \\ \hline p \\ \phi \end{array} \right] = \left[\begin{array}{c} f_u \\ f_j \\ \hline f_p \\ f_\phi \end{array} \right] . \tag{6.46}$$

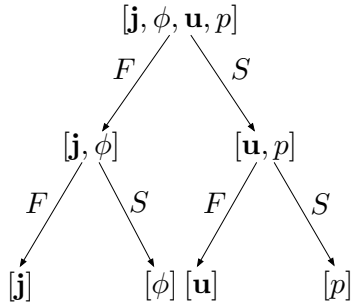


Figure 6.1: Schematic representation of the tree-like structure of the FMS block preconditioner.

Let us write the 4×4 block matrix A from (6.46) as a 2×2 block system splitting the vectorial fields from the Lagrange multipliers, that is, grouping together u-j and p-ϕ,

$$\begin{bmatrix} F_{uj} & G_{uj} \\ D_{uj} & C_{p\phi} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix}, \tag{6.47}$$

where now $x = [u, j]^T$, $y = [p, \phi]^T$, $f = [f_u, f_j]^T$, $g = [f_p, f_\phi]^T$ and matrices F_{uj}, G_{uj}, D_{uj} and $C_{p\phi}$ are the corresponding 2×2 block matrices. Let us now define a U-preconditioner for the 2×2 block system (6.47). The required ingredients are a good approximation for the Schur complement $S_{p\phi} = C_{p\phi} - D_{p\phi} F_{uj}^{-1} G_{uj}$ and possibly for F_{uj}^{-1}. For the Schur complement matrix, we consider the following approximations. First, we neglect the off-diagonal (coupling) blocks in F_{uj}^{-1}, $D_{p\phi}$ and G_{uj} (we note that for $D_{p\phi}$ and G_{uj} the off-diagonal blocks are zero for Galerkin and OSS approximations). This way, we only need to approximate the fluid Schur complement $S_p = C_p - D_p F_u^{-1} G_u$ and the magnetic Schur complement $S_\phi = C_\phi - D_\phi F_j^{-1} G_j$. As approximation of S_p and S_ϕ, we can use any of the preconditioners presented in (6.41)-(6.44).

However, the 2×2 block matrix F_{uj} still couples the computation of u and j. At this point, we can add a second level of recursion and approximate the inverse of F_{uj} by, e.g., its U-factorization, $F_{uj}^{-1} \leftarrow$ `LU_block_precond`(F_{uj}). Following the same ideas presented in Chapter 4 where the matrix F_u is approximated by $\delta t^{-1} M_u$, the Schur complement $S_j = F_j - \delta t C_j F_u^{-1} C_u$ can be approximated by the term

$$S_j \approx F_j - \delta t C_j M_u^{-1} C_u \approx F_j + R_j, \quad \text{where} \quad R_j = \delta t \mathrm{N}^2(j \times B, k \times B).$$

Finally, let us write the P_{FLM} preconditioner expression,

$$P_{FLM}(A) = \left[\begin{array}{cc|cc} F_u & C_u & G_u & T_u \\ 0 & S_j & T_j & G_j \\ \hline 0 & 0 & S_p & 0 \\ 0 & 0 & 0 & S_\phi \end{array}\right]^{-1}. \tag{6.48}$$

This preconditioner decouples the computation of the four physical variables into one-physics problems in a recursive way. The first level of recursion decouples the vectorial fields, velocity and current density, from the Lagrange multipliers, pressure and electric

potential. In the second recursive level, the computation of u and j is also decoupled. Note that the Lagrange multipliers p and ϕ are decoupled because of the diagonal structure of the Schur complement approximation for $S_{p\phi}$ considered above. Figure 6.2 shows these two levels of recursive uncoupling in a tree structure.

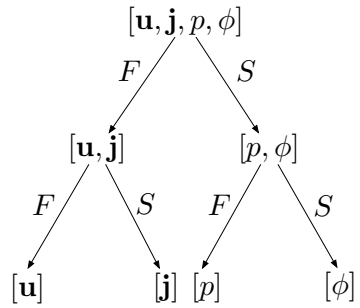


Figure 6.2: Schematic representation of the tree-like structure of FLM block preconditioner.

Note that the same derivation can be made for system (6.46) if the position of u and j is interchanged. We do not write the details for the sake of conciseness but it is important to highlight that the approximation of F_{uj} implies in this case the approximation of the Schur complement with respect to u instead of j, i.e.,

$$P(F_{uj}) = \begin{bmatrix} F_j & C_j \\ 0 & S_u \end{bmatrix}^{-1} \text{ where } S_u \approx F_u - C_u M_j^{-1} C_j \approx F_u + R_u \text{ and } R_u = (u \times B, v \times B).$$

6.4.4 New stabilized PCD preconditioners for inductionless MHD

In this section, we motivate the FLM preconditioner, via some numerical experiments. Let us consider system (6.46) preconditioned with the U-preconditioner with no approximations, i.e., using the *exact* Schur complement $S_{p\phi}$ and F_{uj}. Note that this is not affordable for a real simulation but for a very small problem, it will give us an insight on the importance of every term that is included in the Schur complement. The exact expression of the Schur complement for the OSS formulation presented in Algorithm 6.2 is, (recall that blocks T_* in (6.46) are zero for OSS stabilization),

$$S_\sharp = C_{p\phi} - D_{uj} F_{uj}^{-1} G_{uj} = \begin{bmatrix} C_p & 0 \\ 0 & C_\phi \end{bmatrix} - \begin{bmatrix} D_u & 0 \\ 0 & D_j \end{bmatrix} \begin{bmatrix} F_u & C_u \\ C_j & F_j \end{bmatrix}^{-1} \begin{bmatrix} G_u & 0 \\ 0 & G_j \end{bmatrix}. \tag{6.49}$$

Table 6.1 presents the number of iterations needed for solving the inductionless MHD problem using an iterative solver like GMRES. We have solved the test problem for two Hartmann numbers, Ha $= 10, 1000$. To assess the importance of the terms in F_{uj}^{-1} in the Schur complement, two different versions of F_{uj}^{-1} in (6.49) have been tested. First, we want to evaluate the importance of the stabilization terms at the preconditioner. In order to do so, we have considered the exact Schur complement, denoted as "Stab. Coupled" in Table

6.1 and the case in which we do eliminate the stabilization terms at the preconditioner, denoted as "Galerkin Coupled." Next, we want to evaluate the importance of the coupling terms, eliminating at the preconditioner the coupling terms; this case is denoted as "Stab. Uncoupled". Finally, we consider the case without stabilization and coupling terms, denoted as "Galerkin Uncoupled".

We have solved the 3D inductionless MHD cavity flow problem (see Section 6.5.2 for a detailed description) using a very coarse mesh of $8 \times 8 \times 8$ linear hexahedral elements and the numerical method with OSS stabilization presented in Algorithm 6.2. The results obtained when solving the linear system of equations from the first nonlinear iteration for the four possible combinations are reported in Table 6.1. These results indicate that the coupling blocks C_u and C_j do not have an important effect on the Schur complement definition. This fact is the numerical evidence that has motivated the FLM preconditioners above.

	Stab. Coupled	Galerkin Coupled	Stab. Uncoupled	Galerkin Uncoupled
Ha=10	2	5	7	7
Ha=1000	2	31	8	36

Table 6.1: Number of iterations when solving the exact Schur complement.

On the other hand, the results in Table 6.1 suggest that the stabilization terms are crucial for a robust Schur complement, especially when the Hartmann number increases. It has motivated us to introduce the stabilization terms in the pressure operator for the Schur complement approximation defined in (6.45) for the PCD preconditioner in order to improve its efficiency. Therefore, we propose the following *stabilized* pressure CDR operators

$$\hat{F}_p(p,q) = \frac{1}{\delta t}(p,q) + ((u\cdot\nabla)p,q) + \frac{1}{\mathrm{Re}}(\nabla p,\nabla q) + (\tau_1(u\cdot\nabla)p,(u\cdot\nabla)q) + (\tau_3 p|B|,q|B|), \tag{6.50}$$

$$\hat{F}_\phi(\phi,\psi) = (\phi,\psi) + \mathrm{N}^2(\tau_1\phi|B|,\psi|B|), \tag{6.51}$$

instead of (6.43). It is important to note that the stabilization terms with vectorial products for u and j have been approximated by scalar multiplications for the scalar variables p and ϕ where $|B|$ is the norm of the external magnetic field B. The improvement associated to this modification will be shown with numerical tests in Section 6.5.2. Note also that this new definition of the Schur complement approximation for the Lagrange multipliers p and ϕ can be used to improve the FMS block preconditioner in (6.45).

6.4.5 Thermally coupled inductionless MHD preconditioners

The design of recursive block preconditioners for solving the thermally coupled inductionless MHD equations follows the ideas presented in Section 6.4.1, since the system matrix (6.31) already has a U structure; the coupling between θ and the rest of MHD unknowns is in one direction only. Therefore, the definition of, e.g., a U-preconditioner

for (6.31) reads as

$$P(A) = \begin{bmatrix} F_{upj\phi\sharp}^{-1} & -F_{upj\phi\sharp}^{-1} C_{upj\phi} F_{\theta}^{-1} \\ 0 & F_{\theta}^{-1} \end{bmatrix}, \tag{6.52}$$

where $F_{upj\phi\sharp}^{-1}$ can be the FMS preconditioner (6.45) or the FLM preconditioner (6.48).

6.5 Numerical experiments

This section is devoted to numerically test the behavior of the block preconditioners exposed in previous sections. On one hand, the 3D lid-driven magnetohydrodynamic cavity flow has been used to test and evaluate the properties of the several block preconditioners designed previously. On the other hand, a simulation of a real application such as a Test Blanket Module (TBM) for nuclear fusion reactors has been carried out in order to check the preconditioner behavior when solving a very challenging problem due to its extreme physical working conditions, i.e., a very high Hartmann number. Finally, a flow into a vertical enclosure subject to a temperature gradient has been simulated to test the block preconditioners derived for the thermally coupled inductionless MHD equations.

6.5.1 Experimental framework

The block recursive *LU* preconditioners subject of study were implemented and tested within FEMPAR. FEMPAR is an in-house, parallel hybrid OpenMP/MPI, object-oriented (OO) framework, developed in Fortran90/95, for the massively parallel stabilized FE simulation of multiphysics problems governed by systems of PDEs. FEMPAR provides the tools to drive all the steps required in a typical massively parallel FE multiphysics simulation. These steps comprise the partition (via multilevel graph partitioning) of the underlying unstructured computational mesh into submeshes for distributed-memory computation, the definition of a multi-physics coupled problem and its FE time and space discretization, the nonlinear solution of the problem, the parallel assembly of the underlying blocked large and sparse linear system, the definition of block preconditioners for its preconditioned Krylov subspace solution, and the use of optimal parallel solvers for each block problem at hand. As an optimal parallel solver, among others, FEMPAR provides highly scalable distributed-memory implementations of the Balancing Domain Decomposition by Constraints (BDDC) preconditioner [18, 19, 63].

All experiments reported in the sequel were obtained on a large-scale multicore-based distributed-memory machine, Marenostrum III, located at the Barcelona Supercomputing Center. The Marenostrum III is a FDR10 Infiniband interconnected cluster with 36 IBM System x iDataPlex racks devoted to computations. Each rack is composed of 84 IBM dx360 M4 compute nodes, each equipped with two Intel Xeon E5-2670 EightCore processors running at 2.6 GHz (16 computational cores in total) and 32 GBytes of DDR3 memory (2 GBytes per core), and runs a full-featured Linux OS (SuSe distribution 11 SP2). The codes were compiled using Intel Fortran compiler (13.0.1) with recommended optimization flags and we used OpenMPI (1.5.4) tools and libraries for message-passing. The codes were linked against the BLAS/LAPACK and PARDISO available on the Intel MKL library (version 11.0, update 1). All floating-point calculations were performed in IEEE double precision.

6.5.2 Three-dimensional (3D) MHD cavity flow

The experiments in this subsection deal with the three-dimensional (3D) lid-driven magnetohydrodynamic cavity flow. The computational domain is a unit cube $[0,1]^3$ discretized by a series of uniform meshes composed of linear hexahedral elements with 2^n elements by dimension, where $n = 3, ..., 7$. These meshes were uniformly partitioned into/distributed over (proportionally) increasing number of subdomains/computational cores.

The velocity boundary conditions for this problem consist of a moving upper lid in x-direction, $u = (1,0,0)$ on $(0,1) \times (0,1) \times \{1\}$, and a no-slip condition $u = (0,0,0)$ elsewhere on the boundary. The pressure is fixed to zero in one point, in order to fix the mean value. The boundary condition for the current density is $j \cdot n = 0$ on the whole boundary. The external magnetic field B is chosen to be orthogonal to the moving lid and therefore, only its z-component is nonzero. Note that it can be written in terms of the Reynolds and Hartmann numbers as

$$B = (0,0,B_z) \text{ with } B_z = \frac{\mathrm{Ha}}{\sqrt{\mathrm{Re}}}.$$

For the underlying preconditioned iterative solvers, the iteration is stopped whenever the residual r_k at a given iteration k satisfies $\|r_k\|_2 \leq \mathrm{atol} + \mathrm{rtol}\|r_0\|_2$, with atol and rtol being, respectively, the absolute and relative residual tolerances. Unless specified, $\mathrm{atol} = 0.0$ and $\mathrm{rtol} = 10^{-8}$ for both the external and internal iterative solution processes in block recursive preconditioners, i.e., we use Algorithm 6.6. In particular, FGMRES will be used as the iterative solver for the topmost and intermediate levels, and BDDC preconditioning for the bottommost level (one-physics problems).

The following subsections 6.5.2 and 6.5.2 will compare the behavior of the FLM preconditioner (6.48) depending on the approximation of the Schur complement with respect to the Lagrange multipliers, p and ϕ, defined in (6.41)-(6.44). Therefore, the several versions of the FLM preconditioner tested in the following subsections will be called after the approximation of the Schur complement for p and ϕ, i.e., Uzawa, Cahouet-Chabard (CC), Pressure Convection-Diffusion (PCD) and Pressure Correction (PC).

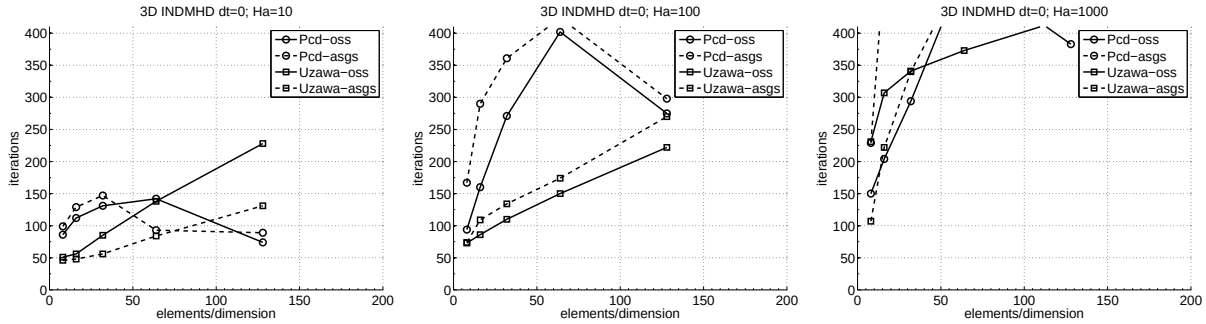
The following numerical experiments will also study the effect of the stabilization technique, ASGS or OSS, in the system matrix. It is important to stress that, for the following tests, we have only included the terms that are strictly needed for stabilization purposes in the OSS formulation. Therefore, terms $(\tau_2 \nabla \cdot u_h, \nabla \cdot v_h)$ and $(\tau_4 \nabla \cdot j_h, \nabla \cdot k_h)$ have not been used in Algorithm 6.2.

Comparison between Schur complement approximations for the FLM preconditioner (stationary case)

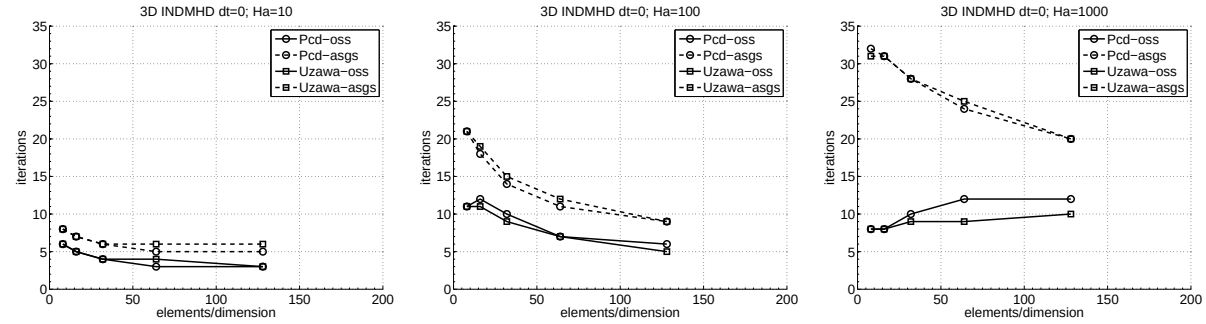
This subsection compares the efficiency of two block preconditioners for the stationary problem. Moreover, another goal of this study is to assess the stabilization method influence in the preconditioner behavior. The block preconditioners used are Pressure Convection-Diffusion (PCD) and Uzawa, together with the two stabilization techniques, ASGS and OSS, see Algorithms 6.1 and 6.2.

The 3D cavity problem has been solved with a Reynolds number $\mathrm{Re} = 10$ and for three Hartmann numbers $\mathrm{Ha} = 10, 100, 1000$. Figure 6.3 shows the number of iterations needed

for the linear solver to converge. In the top row, the number of iterations corresponds to the external solver whereas the bottom row displays the iterations to converge the inner block u-j. On one hand, the Uzawa preconditioner does not optimally converge with h; the number of external iterations increases when the mesh is refined for every Hartmann number and for both stabilization methods. On the other hand, the PCD preconditioner has a much better behavior for small Hartmann numbers, Ha=10. For Larger Hartmann numbers, such as Ha $= 100, 1000$, the external solver requires more iterations to converge. However, as the mesh is refined, the number of iterations is reduced because for smaller h the diffusive term is more important than the convective one. Moreover, the plots in the bottom row in Figure 6.3 show that the inner solver for the coupling u-j has a much better behavior. The iterations do not increase when reducing h, or even they are reduced for the PCD preconditioner. There exists a slight increase when increasing the Hartmann number because the coupling between the fluid and magnetic subproblems becomes stronger.



(a) External iterations (FLM preconditioner). Left: Ha=10, Center: Ha=100, Right: Ha=1000.



(b) Internal iterations u-j (FLM preconditioner). Left: Ha=10, Center: Ha=100, Right: Ha=1000.

Figure 6.3: 3D stationary magnetohydrodynamic cavity problem.

Comparison between Schur complement approximations for the FLM preconditioner (transient case)

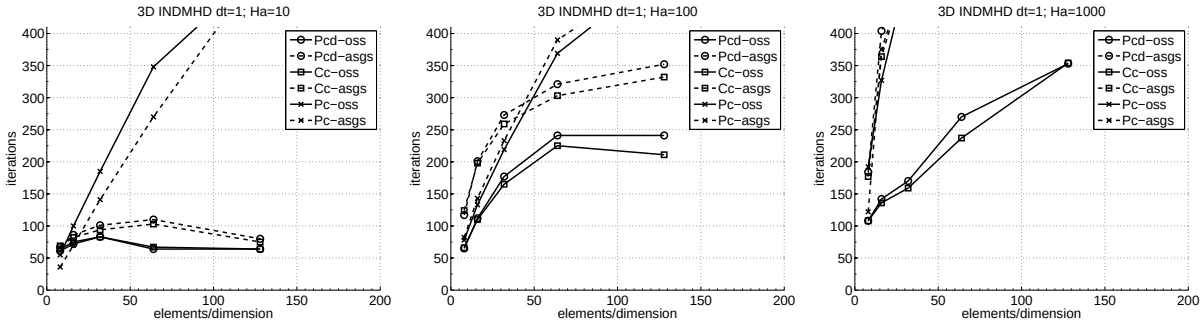
This subsection compares the number of iterations needed to solve the transient magnetohydrodynamic cavity problem in three dimensions (3D) using different block preconditioners for the inductionless MHD equations. Three block preconditioners have been

used, namely Pressure Convection-Diffusion (PCD), Cahouet-Chabard (CC), and Pressure Correction (PC). Similarly to the previous subsection, two different stabilization formulations have been solved, ASGS and OSS methods.

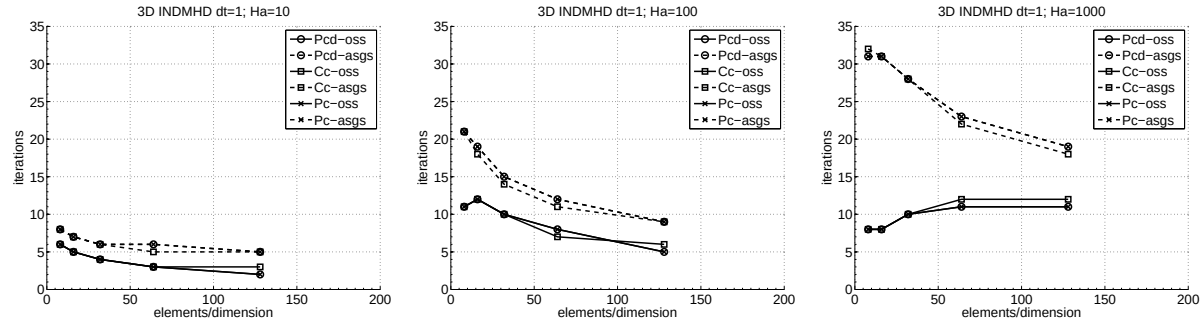
Figures 6.4 and 6.5 display the results obtained for two time step sizes $\delta t = 1.0$s and $\delta t = 0.01$s, respectively. The top row shows the plots for the number of external iterations needed to solve the linear system for three Hartmann numbers, Ha $= 10, 100, 1000$ from left to right. Similarly, the bottom row contains the results for the internal block u-j solver, also for Ha $= 10, 100, 1000$ from left to right.

In general, the number of external iterations required to solve OSS linear systems is smaller than that required for ASGS ones for both choices of δt, especially for the highest Hartmann number, i.e., Ha $= 1000$. Regarding preconditioner efficiency, the PCD and CC preconditioners have a very similar good behavior, both for the number of iterations and the convergence with h, except for the highest Hartmann number Ha $= 1000$ where the number of iterations increases when reducing h. The PC preconditioner only works reasonably well for small δt but it shows a degradation with h.

On the other hand, the internal block u-j has a better behavior when reducing h, independently of δt. However, there is a mild increase in the number of iterations for larger Hartmann numbers.

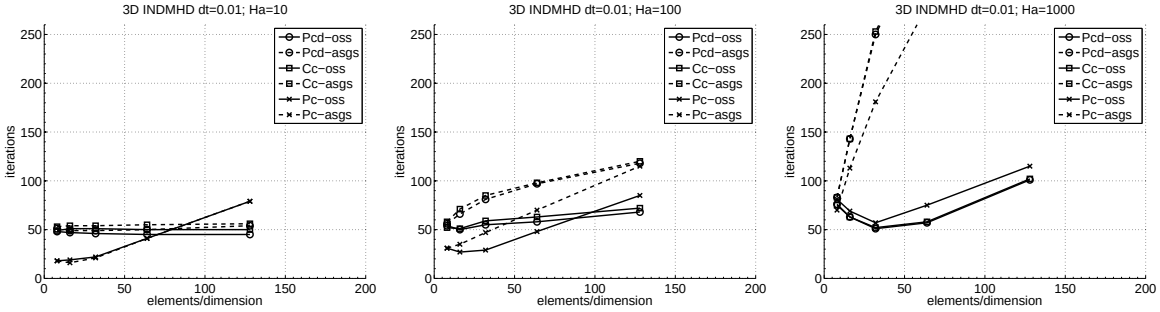


(a) External iterations. Left: Ha=10, Center: Ha=100, Right: Ha=1000.

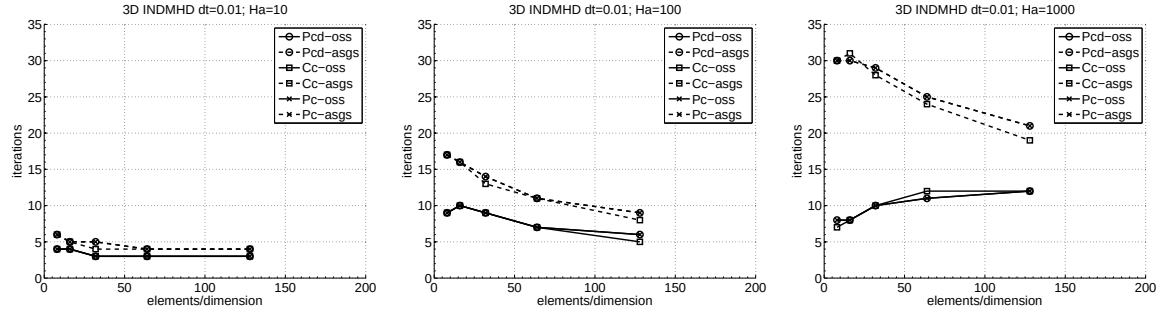


(b) Internal iterations u-j. Left: Ha=10, Center: Ha=100, Right: Ha=1000.

Figure 6.4: 3D transient magnetohydrodynamic cavity problem with $\delta t = 1.0$s (FLM preconditioner).



(a) External iterations. Left: Ha=10, Center: Ha=100, Right: Ha=1000.



(b) Internal iterations u-j. Left: Ha=10, Center: Ha=100, Right: Ha=1000.

Figure 6.5: 3D transient magnetohydrodynamic cavity problem with $\delta t = 0.01$s (FLM preconditioner).

Improved PCD Schur complement approximation for the FLM preconditioner

In Section 6.4.4 we have proposed an improvement of the original PCD preconditioner by introducing some stabilization terms in the Schur complement approximation for both the pressure and the electric potential. This subsection studies its properties solving the transient 3D magnetohydrodynamic cavity flow problem with $\delta t = 0.1, 0.01, 0.001$s for the FLM preconditioner version.

Figure 6.6 shows the number of external iterations for solving the linear system of equations using the stabilized PCD preconditioner. It is clear that the preconditioner behavior with respect to h improves when the time step size is reduced. However, although the number of iterations is also reduced when reducing the time step size δt for Ha = 100, the behavior with respect to h is not as good as the rest of Hartmann numbers tested. This behavior is similar to the original PCD preconditioner, as shown in the previous subsection. However, the addition of the stabilization terms into the Schur complement approximation is crucial for improving the preconditioner efficiency and reducing the number of iterations needed to solve the system. Figure 6.7 shows the number of iterations versus the elements per dimension for both the original PCD and the stabilized PCD preconditioners for 3D uniform meshes with number of elements per dimension from 2^3 to 2^8. The subplot in the left displays the results for a small Hartmann number Ha = 10 where both preconditioners have almost the same behavior. However, for larger Hartmann numbers Ha = $100, 1000$, the center and right subplots show that the stabilized PCD preconditioner reduces the number of iterations needed to solve the

problem and also speeds up the convergence to an asymptotic state. Although the behavior of the stabilized PCD is better than the original PCD, for a Hartmann number of Ha=1000 and $\delta t = 0.01$s the number of iterations increases when reducing h in meshes composed by up to 2^8 elements per dimension.

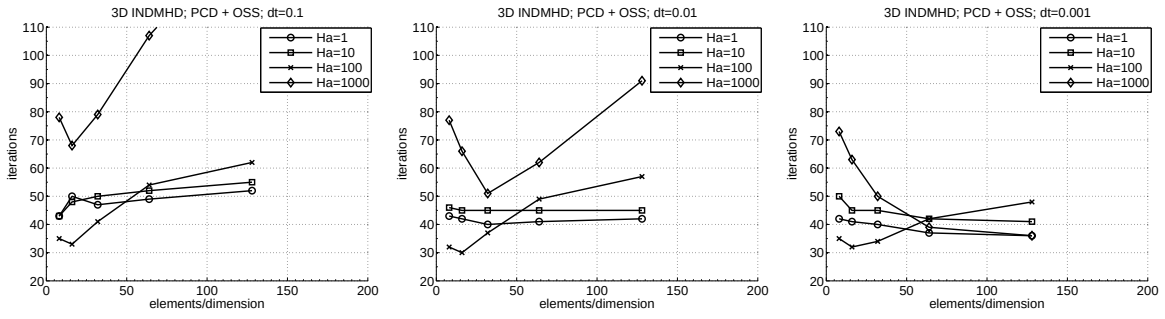


Figure 6.6: Stabilized PCD FLM iterations. Left: $\delta t = 0.1$s, Center: $\delta t = 0.01$s, Right: $\delta t = 0.001$s.

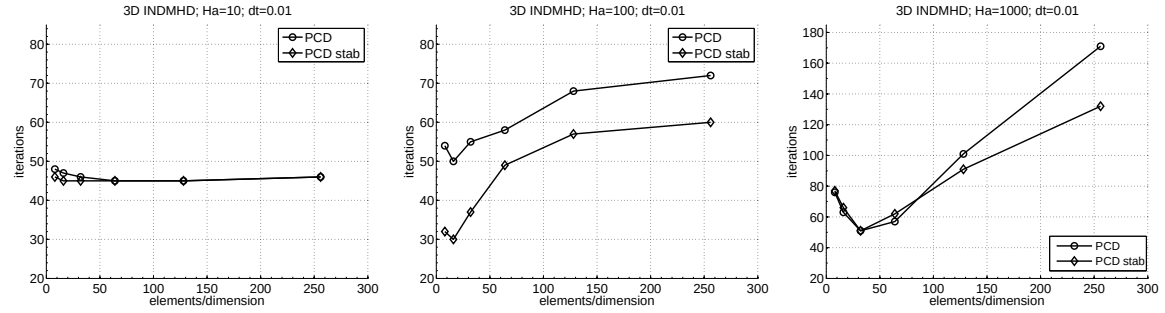


Figure 6.7: Comparison between PCD and stab. PCD FLM. Left: Ha=10, Center: Ha=100, Right: Ha=1000.

Improved PCD Schur complement approximation for the FMS preconditioner

The improvement of the Schur complement approximation for the Lagrange multipliers for the PCD block preconditioner proposed in Section 6.4.4 has also been applied together with the FMS preconditioner defined in Section 6.4.3. It involves the internal blocks splitting between subproblems, that is, between the fluid (NSI) and magnetic (DCY) subproblems. This section deals with the numerical tests done using this approach.

Figure 6.8 shows the number of external iterations needed to solve the linear system of equations for three different time step sizes, namely $\delta t = 0.1, 0.01, 0.001$s. The plots display the results obtained for four Hartmann numbers $\mathrm{Ha} = 1, 10, 100, 1000$. It is clear that this preconditioner has a very good behavior and it is optimally convergent with the mesh size h. Further, the results are quite insensitive to δt and the Hartmann number.

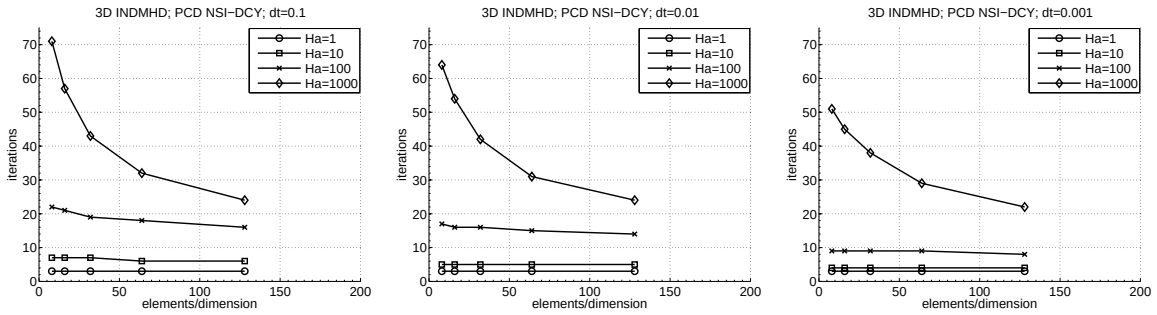


Figure 6.8: Stabilized PCD FMS iterations. Left: $\delta t = 0.1$s, Center: $\delta t = 0.01$s, Right: $\delta t = 0.001$s.

Effect of the internal blocks precision over the external solver

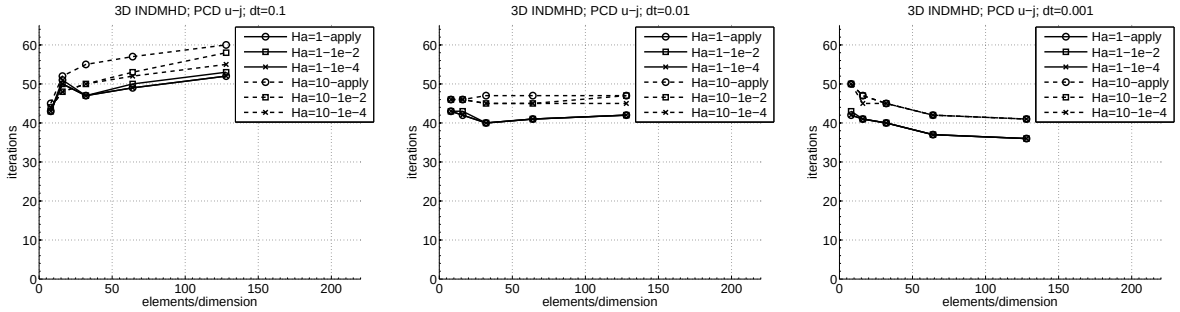
This study aims to understand which is the effect of the precision when solving the internal blocks, *u-j* (FLM) or by subproblems (FMS), over the external iterations for the stabilized PCD block preconditioners. We seek a reduction of the total computation time by relaxing the internal tolerance (and, consequently, the time spent by the internal solver), despite the potential increase in the number of external iterations.

On one hand, Figure 6.9 shows the results obtained using the PCD FLM preconditioner for $\delta t = 0.1, 0.01, 0.001$s and Hartmann numbers Ha $= 1, 10, 100, 1000$. Each combination has been solved with three different internal block approaches, just applying once the internal preconditioner, i.e., using the block recursive preconditioner as stated in Algorithm 6.5, or iterating (Algorithm 6.6) with rtol $= 10^{-2}, 10^{-4}$. The results indicate that for small Hartmann numbers, the number of external iterations is insensitive to the internal blocks precision. However, when the Hartmann number is larger, just applying the internal preconditioner leads to an important increase in the number of iterations, although they are reduced when reducing the mesh size h. Therefore, these results allow a relaxation in the solution of the internal blocks which greatly reduces the total computation time, as it will be shown later on this subsection.

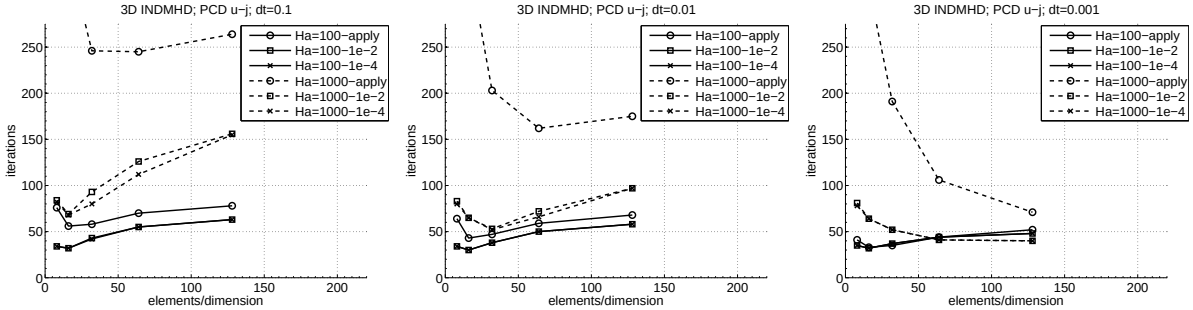
On the other hand, the same study has been done for the PCD FMS preconditioner. The results are presented in Figure 6.10. In this case, the number of external iterations is much more sensitive to just applying the internal preconditioner or iterating until convergence of the internal blocks (NSI and DCY). It is very clear that iterating the internal blocks reduces drastically the number of external iterations for every combination of time step size δt and Hartmann number Ha.

However, the number of external iterations is not conclusive when deciding which is the best option to solve the problem. This study has to be completed with time measurements of the external iterative solver (FGMRES) for both preconditioners. It has been done for the finest mesh composed by $128 \times 128 \times 128$ elements, uniformly partitioned into/distributed over $8 \times 8 \times 8 = 512$ subdomains/cores. The computational times for the PCD FLM preconditioner are shown in Table 6.2 whereas the results for the PCD FMS version are presented in Table 6.3.

The results for the PCD FLM indicate that the fastest option is to iterate the internal blocks with a relative tolerance of 10^{-2}, even though the number of external iterations



(a) Ha=1, 10. Left: $\delta t = 0.1$s, Center: $\delta t = 0.01$s, Right: $\delta t = 0.001$s.



(b) Ha=100, 1000. Left: $\delta t = 0.1$s, Center: $\delta t = 0.01$s, Right: $\delta t = 0.001$s.

Figure 6.9: External iterations for the PCD FLM block preconditioner.

is very similar for just applying once the internal preconditioner. The internal solver for the u-j block requires very few iterations to converge. Despite this moderate additional internal solver cost (compared to a single application of the internal preconditioner), the (slight) reduction of the number of external iterations pays off for reducing the total computation time. Note that the block of the Lagrange multipliers p and ϕ is block diagonal and therefore there is no need to iterate it.

On the other hand, the best option for solving the magnetohydrodynamic cavity flow problem using as preconditioner the PCD FMS is just applying once the internal preconditioners, for both subproblems NSI and DCY. In this case, the internal solver for both subproblems requires a high number of iterations to converge, so that it more than pays off a single application of the internal preconditioner (despite the significant increase in the number of external iterations). However, for the largest Hartmann number considered in this study, Ha = 1000, we expect iterating the internal subproblems to be the method of choice for finer meshes, given the dramatic increase of the number of external iterations with h for a single application of the internal preconditioner.

6.5.3 Simulation of a Test Blanket Module (TBM) for nuclear fusion reactors

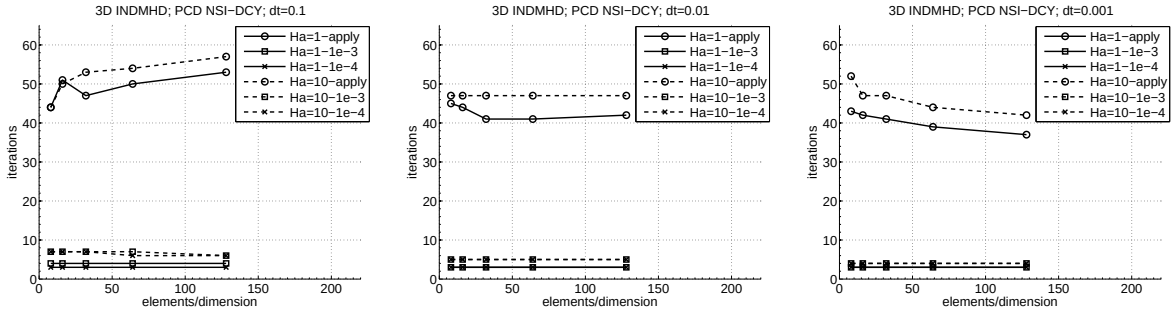
In recent years, a great worldwide effort has been put on the design and development of new nuclear fusion reactors, materializing in ITER, the International Thermonuclear

$h = \frac{1}{128}$		Apply M^{-1}	Iter. tol=10^{-2}	Iter. tol=10^{-4}
δt=0.001	Ha=1	38.04	25.81	55.83
	Ha=10	40.99	31.13	68.61
	Ha=100	53.16	58.12	77.66
	Ha=1000	78.34	55.07	69.16
δt=0.01	Ha=1	51.43	34.28	81.57
	Ha=10	53.37	39.52	86.87
	Ha=100	81.91	81.25	123.16
	Ha=1000	206.99	153.72	202.35
δt=0.1	Ha=1	67.77	47.20	109.38
	Ha=10	73.05	59.59	116.42
	Ha=100	103.33	91.44	141.97
	Ha=1000	357.03	257.45	377.55

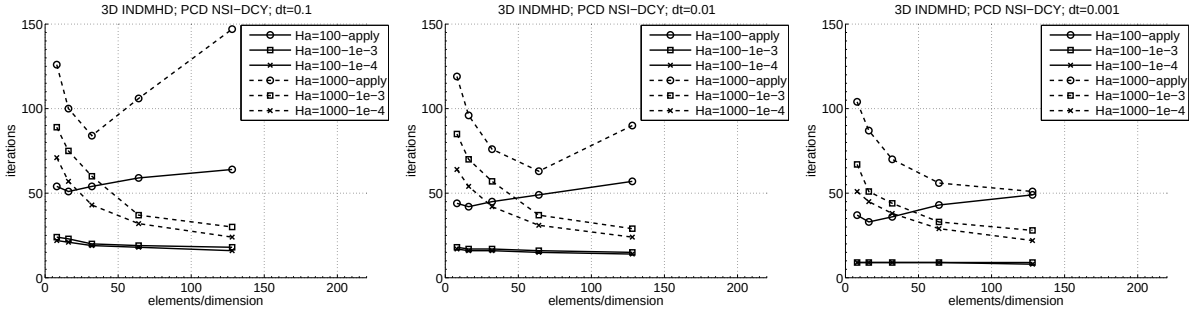
Table 6.2: Computational time (s) for the PCD FLM preconditioner.

$h = \frac{1}{128}$		Apply M^{-1}	Iter. tol=10^{-3}	Iter. tol=10^{-4}
δt=0.001	Ha=1	44.45	27.04	44.66
	Ha=10	50.72	36.21	61.80
	Ha=100	60.01	74.52	110.64
	Ha=1000	63.75	281.99	361.56
δt=0.01	Ha=1	60.05	41.80	85.82
	Ha=10	63.92	72.76	124.58
	Ha=100	81.46	159.81	268.28
	Ha=1000	134.11	384.04	613.33
δt=0.1	Ha=1	76.44	97.88	114.23
	Ha=10	87.56	104.86	184.19
	Ha=100	99.93	273.65	366.13
	Ha=1000	228.79	632.55	1057.98

Table 6.3: Computational time (s) for the PCD FMS preconditioner.



(a) Ha=1, 10. Left: $\delta t = 0.1$s, Center: $\delta t = 0.01$s, Right: $\delta t = 0.001$s.



(b) Ha=100, 1000. Left: $\delta t = 0.1$s, Center: $\delta t = 0.01$s, Right: $\delta t = 0.001$s.

Figure 6.10: External iterations for the PCD FMS block preconditioner.

Experimental Reactor, that is currently being constructed in Cadarache (France).[1] ITER is thought to be an experimental laboratory to test the very complex and new technology needed for energy generation of future plants like DEMO. One of the main components to be studied in ITER are the Breeding Blankets (BB). These devices have crucial functions for the fusion reactor to work properly, i.e., heat power extraction from the plasma, tritium generation and shielding of the magnets from neutron and gamma radiation. In the frame of the Spanish Breeding Blanket Technology Programme TECNOFUS (see www.tecnofus.net), a dual-coolant liquid metal blanket has been designed. The liquid metal flow inside the blanket channels can be modelled through the incompressible inductionless MHD equations because its magnetic Reynolds number is very low (see [98]).

The simulation of the Tecnofus TBM is a very challenging task due to its extreme physical conditions. The fluid is a liquid metal, the alloy Pb-15.7Li, which has a density of $\rho = 9660\,\mathrm{kg/m^3}$, a viscosity of $\nu = 1.3 \cdot 10^{-7}\,\mathrm{m^2/s}$ and an electric conductivity of $\sigma = 751280\,(\mathrm{Ohm} \cdot \mathrm{m})^{-1}$. The dimensionless numbers corresponding to these physical properties are a Reynolds number of $\mathrm{Re} = 4.55 \cdot 10^6$ and a Hartmann number of $\mathrm{Ha} = 5.14 \cdot 10^4$. Note that the external magnetic field magnitude is $B = 10\,\mathrm{T}$, the velocity magnitude at the channel core is designed as $U = 0.2\,\mathrm{m/s}$ and the characteristic length of the channel section is $L = 0.305\,\mathrm{m}$.

Two meshes have been used to solve this problem. The first one, MESH-12.5M, is composed by 2,509,705 nodes and 12,395,008 linear tetrahedral elements. The second mesh, MESH-100M, is obtained after an uniform refinement of the first one where every

[1] See `http://www.iter.org` for details.

	MESH-12.5M	MESH-100M
1st time step	404	366
	304	229
	205	147
	121	111
	74	75
	39	41
2nd time step	248	239
	196	156
	85	70
3rd time step	189	185
	169	123
	53	50

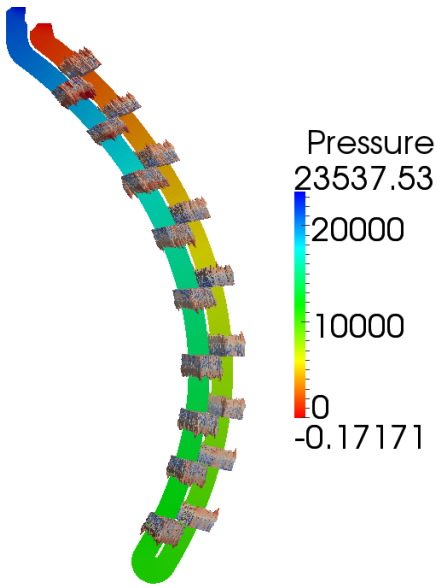
Table 6.4: Number of solver iterations for the Tecnofus TBM.

tetrahedra is divided into 8 smaller tetrahedral elements. This way, the second mesh has 20,017,537 nodes and 99,160,064 elements. Using an automatic mesh partitioner [87], MESH-12.5M and MESH-100M were partitioned into/distributed over 512 and 4096 subdomains/cores, respectively. The boundary conditions have been set to no-slip conditions at the walls ($u = 0$) and a fixed velocity at the inlet ($u = (0, 0, -4.0)\,\mathrm{m/s}$) for the velocity field and perfectly conducting walls ($\phi = 0$) for the electromagnetic variables. Furthermore, the tolerance for the nonlinear iterations has been set to nltol $= 10^{-3}$. The iterative solver (FGMRES) tolerances are rtol $= 10^{-6}$ for the relative part and the absolute one is chosen as atol $=$ nlres $\cdot$ nltol/10, where nlres is the norm of the system residual and nltol is the tolerance of the nonlinear iterative loop. This way, the iterative solver converges to a residual an order of magnitude lower than what is imposed for the nonlinear iterations to converge.

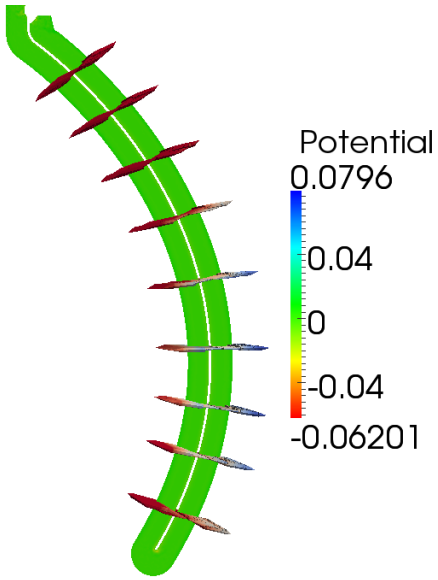
The computation has been carried out using the block recursive preconditioner PCD FMS explained in Section 6.4.3 with just applying once the internal recursive preconditioners for the subproblems, Navier-Stokes (NSI) and electromagnetic problem (DCY), which has proven to be the fastest preconditioner for high Hartmann numbers, see Section 6.5.2. The time step size has been chosen to $\delta t = 0.1\,\mathrm{s}$ for MESH-12.5M and $\delta t = 0.025\,\mathrm{s}$ for MESH-100M. This way, the Courant-Friedrichs-Lewy (CFL) number for MESH-100M is twice the CFL number of MESH-12.5M because the mesh size h is reduced by a factor 8 from MESH-12.5M to MESH-100M. Let us recall that the CFL number is defined as $\mathrm{CFL} = U\frac{\delta t}{h}$ where U is a characteristic velocity of the problem. Table 6.4 shows the number of solver iterations needed in every nonlinear iteration for the first three time steps of the simulation for both meshes. The results show that, even for a CFL number twice larger, the number of solver iterations does not increase, actually they slightly decrease, when refining the mesh and reducing the mesh size h by a factor of 8.

Finally, the transient computation converges to a stationary solution that is shown

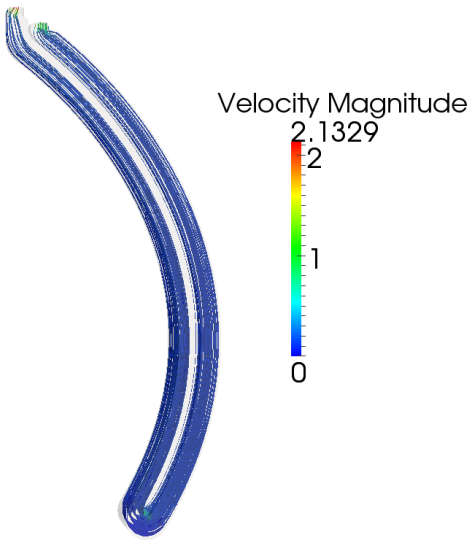
in Figure 6.11 for MESH-12.5M. In the top row, Figure 6.11(a) displays the pressure field in a vertical section and the velocity field at several cross sections of the channels whereas Figure 6.11(b) shows the electric potential field in the same vertical section as the previous plot and the current density field in cross sections of the channels. Figure 6.11(c) plots the velocity streamlines along the TBM with a zoom of such streamlines around the turn in Figure 6.11(d).



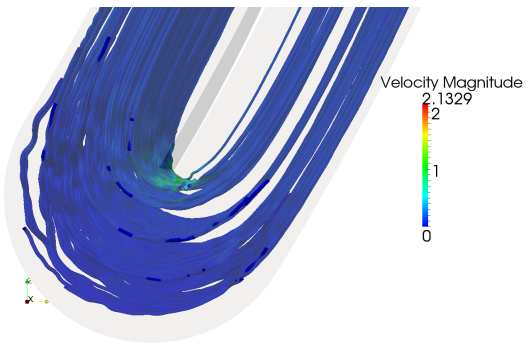
(a) Velocity and Pressure fields



(b) Current Density and Electric Potential fields



(c) Velocity streamlines



(d) Zoom of velocity streamlines around the turn

Figure 6.11: Simulation results for the Tecnofus TBM.

6.5.4 Thermally coupled inductionless MHD flow in a vertical enclosure

This subsection deals with the simulation of the thermally coupled inductionless MHD problem in a vertical enclosure with square section. The computational domain is the volume $[0,1]\times[0,1]\times[0,7.5]$ and it has been discretized with a structured uniform mesh of lineal hexahedral elements containing 48, 48 and 360 elements in x, y and z direction, respectively. This mesh was uniformly partitioned into/distributed over $4\times4\times10=160$ subdomains/cores. Following the indications in [5], the boundary conditions have been set as,

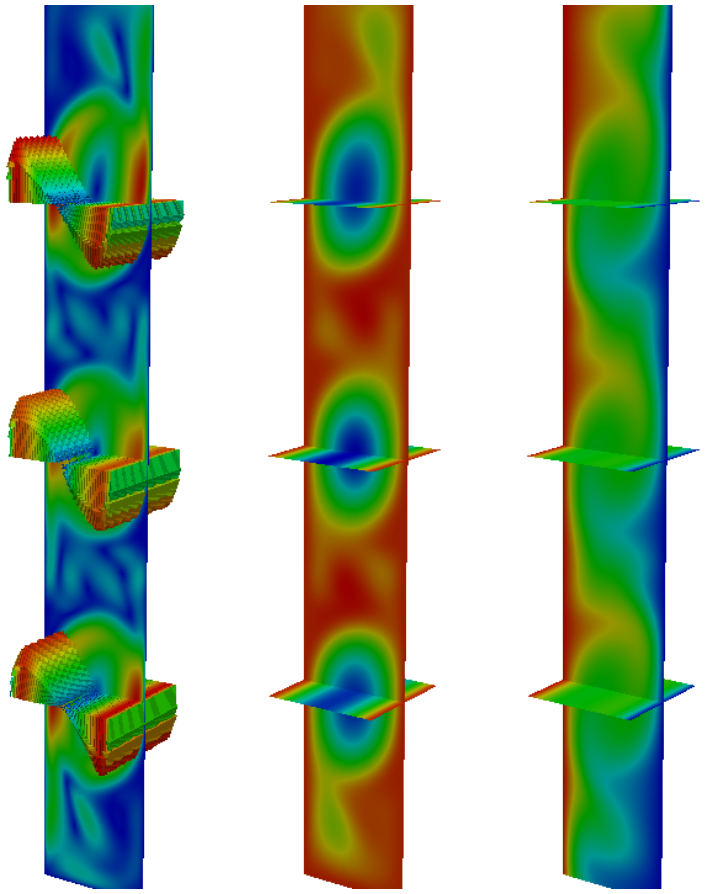
$$
\begin{aligned}
&u_x = u_y = u_z = 0 \text{ at } x = 0,1 \text{ and } z = 0,1,\\
&u_y = 0 \text{ at } y = 0,1,\\
&j_x = 0 \text{ at } x = 0,1,\\
&j_y = \frac{\Omega}{\mathrm{Ha}} \text{ at } y = 0,\\
&j_y = -\frac{\Omega}{\mathrm{Ha}} \text{ at } y = 1,\\
&j_z = 0 \text{ at } z = 0,1,\\
&\theta = 0.5 \text{ at } x = 0 \text{ and } \theta = -0.5 \text{ at } x = 1.
\end{aligned}
$$

The gravity field is applied in the z direction and the external magnetic field is horizontal and perpendicular to the temperature gradient $B=(0,B,0)$ where the magnitude B is computed from the Hartmann number. The dimensionless numbers that govern the flow are taken as a Grashof number of $\mathrm{Gr}=4\cdot10^6$, a Prandtl number of $\mathrm{Pr}=0.025$ and two different Hartmann numbers of $\mathrm{Ha}=100,500$.

The linear systems of equations to be solved for every time step and nonlinear iteration have been approximately solved with the preconditioned FGMRES iterative scheme using as preconditioner the recursive block preconditioner $P(A)$ defined in (6.52) using the block preconditioner P_{FMS} from Section 6.4.3 to approximate the inductionless MHD block. Figure 6.12 shows the results for a Hartmann number of $\mathrm{Ha}=100$. Figure 6.12(a) displays the velocity field whereas the electric potential and the temperature fields are shown in Figures 6.12(b) and 6.12(c), respectively. Similarly, the results for the simulation with a Hartmann number of $\mathrm{Ha}=500$ are plotted in Figures 6.13(a), 6.13(b) and 6.13(c) for the velocity, the electric potential and the temperature fields. In Figure 6.13 we observe that three vortices appear for low Hartmann numbers, whereas for larger Hartmann numbers the solution presents only one vortex.

6.6 Software design and implementation

In this section we describe key design guidelines of the software that provides the tools for the code implementation of block recursive preconditioners within FEMPAR. While being applied to the particular context of our simulation software, we expect these guidelines to be very useful for practitioners willing to implement block recursive preconditioning within their computer simulation codes. Some requirements for this software are:



(a) Velocity (b) Electric Potential (c) Temperature

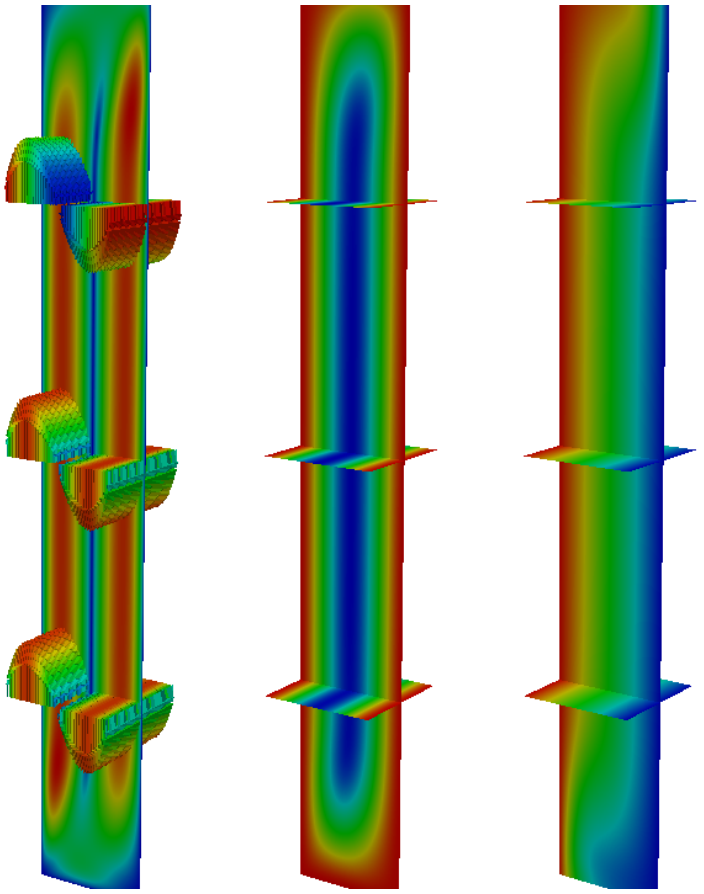
Figure 6.12: Simulation results for Ha=100.

1. It must be *abstract* and *flexible* enough so that it accommodates a large bunch of different block preconditioning strategies in a unified framework, while it is easy to extend with new functionalities without the need of modifying existing code.

2. It must support *recursion* so that the action of the inverses of the diagonals blocks in an approximate block factorization (as might be required e.g., for step 5 of Algorithm 6.5 at any inner level of the preconditioner hierarchy) can be in turn computed recursively using an approximate block factorization.

3. It must be built on top of the preconditioned iterative solvers available in FEMPAR [17–19] for the computation of the action of the inverse of a matrix into a vector (as required, e.g., for step 5 of Algorithm 6.5 in the bottommost level of the preconditioner hierarchy), so that the large bunch code available in FEMPAR for such purpose can be *reused*.

We found the abstractions, design principles and mechanisms provided by the Object-Oriented (OO) software development approach [108] to be particularly useful (if not essential) to meet the aforementioned requirements. Figure 6.14 illustrates the OO design



(a) Velocity (b) Electric Potential (c) Temperature

Figure 6.13: Simulation results for Ha=500.

of the software for block recursive preconditioning as a standard Unified Modeling Language (UML) class diagram. This diagram shows a static, structural view of the software being designed, focusing on its main elements: classes and their relationship. A class is represented as a rectangle containing two compartments, with the top and bottom ones showing the class's name and its methods (also called operations), respectively. Two types of relationships are depicted in Figure 6.14: *realization* and *aggregation*.[2] We refer the reader to [28] for a comprehensive treatment of UML class diagrams.

To keep the presentation simple, we omitted from Figure 6.14 the definition of classes (and their relationship) for vectors and sparse matrices. In particular, the Vector class represents a single (discrete) scalar or vector field, e.g., u or p, while Blk_vector repre-

[2] A realization is a relationship among an abstract class and a class (an implementor) that realizes (i.e., implements) the abstract methods provided by the abstract class. Realizations are represented as a solid line among the abstract class and its implementor, with an unfilled triangle pointing to the abstract class; abstract classes and methods have their name depicted *in italics*. On the other hand, in an aggregation, one class (the whole) is a collection or container of another class (the part). An aggregation is depicted as a solid line connecting the whole and the part, with an unfilled diamond on the whole side, and the multiplicity of the aggregation in the part side, i.e., how many instances of the part class are contained in one instance of the whole class.

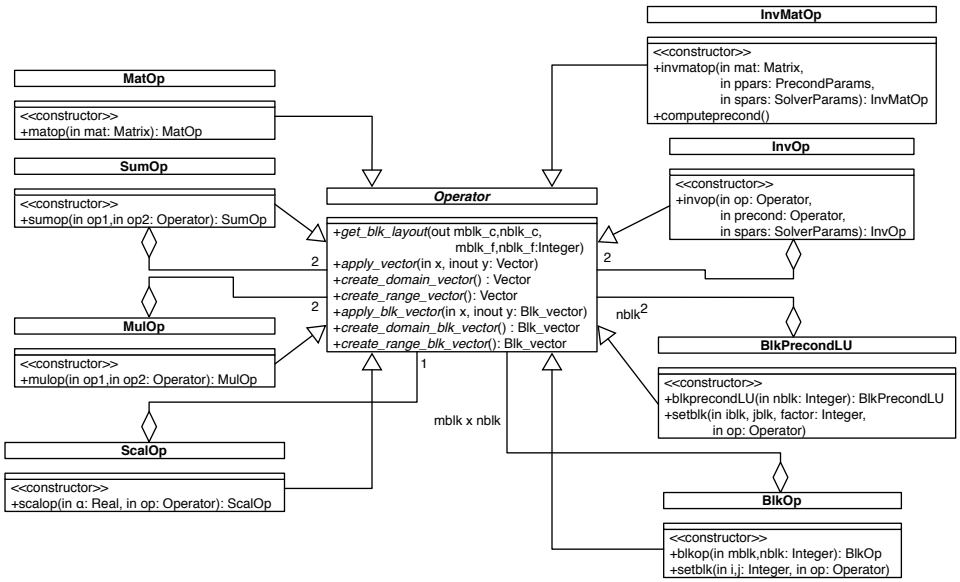


Figure 6.14: UML class diagram representing the OO design of the software that accommodates block recursive preconditioning within FEMPAR.

sents a block aggregation of several (discrete) scalar or vectorial fields, e.g., $[u, p]$, $[j, \phi]$, or $[j, \phi, u, p]$. In UML terminology, an instance of the Blk_vector class is *composed of* several instances of the Vector class [28]. For reasons made clear below, it is essential for the overall framework that an instance of the Vector class can be either created from scratch or from another already created instance. In the former case, new storage space is allocated for the new instance, while in the latter case the new instance shares the storage space with the instance from which it is created. In other words, the former instance is a *view* of the latter instance. A Blk_vector instance can be therefore composed of instances of the Vector class which are *views* of instances previously created. On the other hand, the Matrix class encapsulates a single sparse matrix. The set of methods of these three classes (i.e., Vector, Blk_vector, and Matrix) provide the basic functionality for the implementation of Krylov subspace methods, no matter e.g., how they are stored, nor laid out in a distributed-memory environment.

Central to the design depicted in Figure 6.14 is the *Operator* abstract class, which represents any linear mapping among vector spaces in its strict mathematical sense. Its application to an instance x of the Vector or Blk_vector classes is returned as an instance y by the *apply_vector* or *apply_blk_vector* abstract methods, respectively. This is the minimal functionality that implementor classes have to realize (i.e., implement) to act as a coefficient matrix or preconditioner in preconditioned iterative solvers. The rest of abstract methods of *Operator* are required by implementor classes to implement the *apply_vector* and *apply_blk_vector* abstract methods. The former methods are better grasped by sketching the realization of the latter ones in the implementor classes, which is considered next.

The most basic form of an *Operator* is a matrix, represented by the MatOp implementor class. An instance of the MatOp class is created from an instance of the Matrix class, so that the realization of the *Operator* abstract methods *apply_vector* or *apply_blk_vector*

is naively performed as an invocation of the corresponding methods in the Matrix class. A more involved realization of an *Operator* is the InvMatOp class, which represents the (possible approximate) action of the inverse of a matrix computed by means of a preconditioned iterative solver. This class is the common entry point to all preconditioners and iterative solvers available in FEMPAR (see requirement #3 at the beginning of the section). An InvMatOp instance is created from an instance of the Matrix class, and a set of preconditioner and iterative solver parameters. In this work, we make intensive use of the BDDC preconditioner, for which FEMPAR provides highly efficient distributed-memory implementations [18, 19]. Preconditioner parameters for the BDDC method are e.g., the type of continuity constraints enforced, the corner-detection mechanism that ensures the invertibility of the local Neumann and global coarse-grid problems, or the strategy used to deal with the coarse-grid problem (e.g., serialized or overlapped with fine-grid duties [18]). On the other hand, FEMPAR provides templated implementations of fixed-point (e.g., the Richardson method) and Krylov subspace methods (e.g., PCG and GMRES for symmetric positive definite and general unsymmetric linear systems, respectively, and FGMRES to support variable preconditioning). The former methods are used for the computation of rough approximations of the action of the inverse of a matrix into a vector, which will be shown to be useful for some block recursive preconditioners in Section 6.5. Iterative solver parameters are essentially those that control the convergence criteria (e.g., absolute and relative residual tolerance, maximum number of iterations, etc.), although there are solver-specific parameters, such as the orthogonalization method and the number of iterations for each restart of GMRES.

The implementor classes SumOp, MulOp, and ScalOp provide the basic building blocks to construct new linear mappings from existing ones as follows: Given two linear mappings A and B, the SumOp and MulOp classes represent the $A + B$ and AB linear mappings, respectively, while given A and a scalar α, the ScalOp represents αA. An instance of each of these classes is built from its *Operators*, as represented by the aggregation relationship among these three classes and the *Operator* abstract class. The reader should now notice that any implementor class (e.g., MatOp or InvMatOp) can appear in place of an *Operator* (this is precisely the potential behind the *realization* relationship), so that *by means of the SumOp, MulOp and ScalOp implementor classes one may build any linear mapping that involves a combination of these three operations*. For example, to build $M_p^{-1}F_pL_p^{-1}$ in (6.37) using our software design, one first construct two instances of InvMatOp for M_p^{-1} and L_p^{-1}, respectively, and one instance of MatOp for F_p. Then, an instance of the MulOp class is created from F_p and L_p^{-1} to build $F_pL_p^{-1}$, and finally, another MulOp instance is created to build $M_p^{-1}F_pL_p^{-1}$ from a InvMatOp instance (i.e., M_p^{-1}) and a MulOp instance (i.e., $F_pL_p^{-1}$).

The *create_domain_vector* (*create_domain_blk_vector*) and *create_range_vector* (*create_range_blk_vector*) abstract methods play a major role for the realization of the *apply_vector* (*apply_blk_vector*) by the SumOp, MulOp and ScalOp classes. Let us consider for e.g., the implementation of $y := ABx$, where x and y are instances of the Vector class (this is performed by the *apply_vector* method realized by the MulOp class). This operation can in turn be decomposed into $w := Bx$, and $y := Aw$, where w is a workspace instance of the Vector class. For the product to be well-defined, w must be compatible with the range space of B, or equivalently, with the domain space of A. For example, low-level details such as the size, storage or distributed-memory layout must match. The

create_range_vector and *create_domain_vector* abstract methods provide a new Vector instance w compatible with the range and domain spaces of the *Operator*, respectively. The implementation of these abstract methods for the MatOp, and InvMatOp classes is the one that ultimately determines the size, storage or distributed-memory layout for w (that in turn are extracted from the Matrix class, that internally encapsulates all these low-level details). The rest of implementor classes, such as SumOp, MulOp, and ScalOp, just implement *create_range_vector* and *create_domain_vector* by means of a call to the corresponding method on its leftmost and rightmost *Operator*, respectively.

The BlkOp, BlkPrecondLU, and InvOP classes are the ones that provide our software design with the ability to construct block recursive preconditioners (see requirement #2 at the beginning of the section). The BlkOp implementor class represents a linear mapping which is blocked into $mblk \times nblk$ blocks. A BlkOp instance is built from as many *Operators* as blocks, as represented by the aggregation relationship among BlkOp and the *Operator* abstract method. For example, the coefficient matrix in (6.25) can be built as a single 4×4 BlkOp instance, with each block being a MatOp. However, as the BlkOp instance in turn realizes the *Operator* abstract class, this coefficient matrix can be alternatively built with a three-level tree-like structure. In the bottommost level, one first builds a MatOp per each of the 4×4 blocks. In an intermediate level, four 2×2 BlkOp instances are then built, each of them corresponding to the four 2×2 blocks delimited by the partitioning lines in (6.25). Finally, at the topmost level, another 2×2 BlkOp instance is built from the four BlkOp instances in the intermediate level. The realization of the *apply_blk_vector* by the BlkOp class just performs the operation $y := Ax$, where x, y are instances of the Blk_vector class. On the other hand, the BlkPrecondLU represents an approximate block LU factorization (see $P(A)$ in (6.34)).[3] Its realization of the *apply_blk_vector* performs the operation "Solve $(LU)y = r$", where L and U are lower and upper block triangular factors, respectively, built from $nblk \times nblk$ *Operators* each, and r, y are Blk_vector instances. Finally, the InvOP class realizes *apply_blk_vector* as the action of the (possible approximate) inverse of an *Operator* on a vector, using a preconditioned iterative method with a prescribed preconditioner provided as an *Operator*.

The *get_blk_layout* abstract method is essential for the realization of the *apply_blk_vector* by the BlkOp and BlkPrecondLU classes. Given an *Operator*, this abstract method returns in $mblk_c$ and $nblk_c$ the number of row and column blocks in the coarsest-grain block partitioning of the *Operator*, while $mblk_f$ and $nblk_f$ provide those in the finest-grain one. For instance, in the example of the previous paragraph, it returns $mblk_c = nblk_c = 2$ and $mlbk_f = nblk_f = 4$ for the BlkOp instance in the topmost level of the hierarchy, $mblk_c = nblk_c = mblk_f = nblk_f = 2$ for any of the four BlkOp instances in the intermediate level, and $mblk_c = nblk_c = mblk_f = nblk_f = 1$ for any of the MatOp instances in the bottommost level of the hierarchy. Let us now consider the *apply_blk_vector* in the topmost BlkOp instance, let us call it A. On entry, this method expects x and y to be partitioned into as many blocks as those present in the finest-grain partitioning of the *Operator*, i.e., $nblk_f = 4$ and $mblk_f = 4$, respectively, in this case. In preparation to the call of the *apply_blk_vector* on a given intermediate BlkOp instance, say A_{ij}, the *apply_blk_vector* in A creates a pair of temporary Blk_vector instances, say x_j

[3] One can similarly define BlkPrecondD and BlkPrecondU in Figure 6.14 in order to represent the D-preconditioner and U-preconditioner, respectively, in (6.32) and (6.33), although they are omitted from the figure for simplicity.

and y_i, that are built from those blocks of x and y corresponding to A_{ij}. In other words, the number of blocks of x_j and y_i is given by $nblk_f$ and $mblk_f$ resulting from a call to *get_blk_layout* on A_{ij}, while, x_j (y_i) starts from the block of x (y) with identifier given by the sum of those $nblk_f$'s ($mblk_f$'s) resulting from $j-1$ ($i-1$) calls to *get_blk_layout* on A_{ik} (A_{kj}), with $k = 1, 2, \ldots, j-1$ ($k = 1, 2, \ldots, i-1$). Notice that the blocks of x_j and y_i are created as *views* (see above for the notion of a view) of the corresponding blocks in x and j, so that this mechanism allows the blocks of x and y received on entry to the root of the hierarchy to flow top-bottom thorough the hierarchic deployment of the *apply_blk_vector* method.

The reader might have already observed that this software design accommodates, e.g., the two approximate block recursive preconditioners discussed in Section 6.4.3. In the topmost level, one creates a 2×2 BlkPrecondU instance, with the two leading diagonal blocks $F_\sharp^{-1}$ and $S_\sharp^{-1}$ defined as InvOp instances, and the upper off-diagonal block G as a BlockOp instance. In the intermediate level, $F_\sharp^{-1}$ is in turn built as an InvOp instance from a 2×2 BlockOp instance (i.e., built from the blocks of $F_\sharp$ as MatOp instances), and a further 2×2 BlkPrecondU to be used as a preconditioner for the preconditioned iterative computation of the action of $F_\sharp^{-1}$ on a vector (see step 6 of Algorithm 6.6).

Finally, for those developers reluctant to pure OO languages, such as C++ or Fortran2003, let us stress that FEMPAR is a Fortran90/95 code. We were able to implement the design in Figure 6.14 within FEMPAR, with no loss of functionality, using the techniques discussed in [1] for generic programming and run-time polymorphism emulation in Fortran90/95.

6.7 Conclusions

In this chapter, we have extended block preconditioning techniques used in computational fluid dynamics to the (thermally coupled) incompressible inductionless magnetohydrodynamics problem. Our approach considers the explicit introduction of the current density as an additional unknown of the problem, in order to end up with a formulation that will be suitable for problems involving large Hartmann numbers, e.g., breeding blanket simulations.

We propose an abstract setting to design preconditioners for multiphysics problems, based on a recursive use of block factorization, that allows us to decouple the computation of every physical variable in a multiphysics problem at the preconditioner level. This idea has been applied to our target problem, (thermally coupled) inductionless MHD problem, (where the unknowns are the velocity, pressure, current density and electric potential) but can also be applied to other problems like resistive MHD, see Chapter 2, or liquid crystal problems [16]. We consider different preconditioners based on approximations of the resulting Schur complement matrices. The robustness of these preconditioners relies on good approximations of the Schur complement matrices that appear in the recursive factorization process. An study of the inductionless MHD system has motivated a first preconditioner that initially decouples fluid and magnetic problems. Next, we propose a method that instead decouples vector fields and Lagrange multipliers at the first level. The assumptions undertaken in this last case have been justified via numerical evidences. As a result of this work, we have also observed the importance to consider

stabilization terms in the PCD preconditioners. The recursive preconditioners for both subproblems allow us to obtain block preconditioners with good properties with respect to the mesh size h and the Hartmann number. We give details about an abstract and flexible implementation of block recursive preconditioning.

A detailed set of numerical tests has been performed to assess the properties of the different methods proposed herein. The combination of our FE formulations, with an explicit treatment of the current density, and the recursive LU preconditioners we propose, finally allow us to solve realistic breeding blanket simulations with very high Hartmann numbers.

Chapter 7

Conclusions

The main conclusions and contributions from this thesis are summarized in this chapter. Furthermore, some open lines of research for the future are also listed.

7.1 Achievements and contributions

This work has focused on developing numerical techniques for solving the incompressible magnetohydrodynamics (MHD) problem based on stabilized finite element methods. The contributions from each chapter are explained in the following list:

- In Chapter 2, a stabilized finite element method for solving the incompressible resistive MHD equations has been proposed and developed. The main goal of the stabilization method has been to circumvent the need to satisfy the discrete inf-sup conditions which allows us to use any finite element spaces, even equal interpolation spaces. Furthermore, the stabilization terms also avoid the numerical oscillations when first order derivatives dominate second order ones. However, the most important feature of the proposed method is that it allows to converge to singular solutions even when using a continuous approximation for the magnetic induction field. This property is achieved because of the splitting of the residual of the equation for the magnetic field into two parts, separating the gradient of the magnetic pseudo-pressure from the remaining terms. The resulting formulation mimics the correct functional setting of the continuous problem.

- The stabilized finite element method proposed in Chapter 2 has been extensively analyzed in Chapter 3. A detailed stability and convergence analysis of the formulation has been derived to assess the method properties, both in terms of stability and convergence. Moreover, as a result of the analysis, the need to use a certain type of meshes with macro-element structure has been identified. This particular structure can be easily obtained after a modification of any original mesh, both for triangles or quadrilaterals in two dimensions and for tetrahedra or hexahedra in three dimensions.

- Chapter 4 has dealt with the development of efficient algorithms for solving the resistive MHD problem. In this chapter, the stabilized finite element formulation

has been based on orthogonal subscales because apart from maintaining the features from the method proposed in Chapter 2, it also keeps the skew-symmetry of the off-diagonal blocks of the system matrix. This property is crucial for designing unconditionally stable operator splitting schemes. Furthermore, another very important feature of the proposed algorithms is that they consist of two levels of splitting. The first level segregates the computation of the Lagrange multipliers, the pressure and the magnetic pseudo-pressure from the vectorial fields. Then, the second level splits the computation of the velocity and the magnetic induction fields.

- In Chapter 5, a simplified set of equations for MHD has been studied, the inductionless MHD problem which couples the Navier-Stokes equations from fluid mechanics with a Darcy-type problem for the magnetic part that consists of Ohm's law and the electric charge conservation equation. This PDE system can be applied for solving problems where the magnetic field induced by the moving fluid is negligible with respect to the externally applied magnetic field. The condition for this to happen is that the magnetic Reynolds number has to be small. A stabilized finite element method has been designed based on the variational multiscale paradigm to deal with the drawbacks from a crude Galerkin approximation. The stabilization terms allow us to use equal order interpolation finite element spaces, to avoid unstabilities appearing when the convective terms dominate the diffusive ones and to reduce numerical complications when the coupling between the two subproblems is important. Another important aspect of this work is that the stabilization parameters have been designed based on the stability and convergence analysis of the method.

- Chapter 6 has been devoted to the design of new block recursive LU preconditioners for the thermally coupled inductionless MHD problem. The fully-coupled monolithic approach for solving the inductionless MHD problem developed in Chapter 5 leads to the solution of huge linear systems of equations that are typically ill-conditioned. The use of Krylov iterative solvers, such as GMRES, implies the need to design efficient and scalable preconditioners to accelerate the convergence of the iterative solver. The chosen approach in this work has been block LU preconditioners that allow the splitting of the fully-coupled multi-physics problem into a series of smaller and easier to solve one-physics problems at the precondtioner level. There are two major features of the proposed block preconditioners. On one hand, recursivity. To be able to implement the multi-level splitting techniques for uncoupling the computation of the physical variables, the block preconditioner has to be recursive, in the sense that for every iteration of the external solver, there is the need to internally solve other blocks. This is not possible without adding recursivity to the code implementation. On the other hand, a study of the exact solution of the Schur complement for the Lagrange multipliers, the pressure and the electric potential, has allowed us to identify the key terms to design a good approximation of it. This way, a new stabilized version of the PCD preconditioner has been introduced that improves the behavior of the original one.

7.2 Future lines of research

There are some open lines of research for future work:

- To design new recursive block LU preconditioners for the incompressible resistive MHD problem. The ideas behind the block preconditioners explained in Chapter 6 together with the operator splitting schemes developed in Chapter 4 will be used to develop new block preconditioners. Moreover, a detailed set of experiments has to be performed in order to assess the block preconditioners behavior.

- To design stabilization methods for the thermally coupled resistive MHD problem using the Boussinesq's assumption. It is very important to take into account the thermal coupling into the resistive MHD problem when simulating certain industrial processes as steel casting or some devices for nuclear fusion reactors. Therefore, the design of algorithms to approximate the thermally coupled resistive MHD system is a must. Moreover, the simulation of real processes needs the use of efficient and scalable preconditioners. In this sense, the implementation of block recursive preconditioners for this problem is also a very important task.

- To improve the scalability of preconditioned domain decomposition algorithms, such as Balancing Domain Decomposition by Constraints (BDDC), for solving the internal blocks when applying the block LU preconditioners. Specifically, these algorithms have to be improved when solving convective-dominated problems, such as the blocks dealing with the velocity and current density fields for the induction-less MHD problem or the velocity and magnetic induction fields for the resistive MHD equations.

- To implement thin-wall boundary conditions to properly simulate TBM's where the solid walls have finite conductivity. For these walls, currents leaving the fluid enter the wall, turn in the wall into a tangential direction and create in the wall a distribution of wall potential. This phenomena can be modelled using a thin-wall condition for the electric potential. Such boundary condition is widely used to simulate the interaction of the fluid with the solid walls.

- To simulate turbulence in MHD flows. In recent years, variational multiscale methods (VMS) have been used to model turbulent incompressible flows. These methods arise as a numerical stabilization techniques, but also model successfully the classical flow turbulence since they introduce numerical dissipation efficiently. The main idea relies on the splitting of the continuous unknown into a resolvable (finite element) component and a subgrid or subscale component that models the finer scales of the problem that cannot be captured by the finite element mesh. This framework will be extended to model turbulent MHD flows.

Bibliography

[1] J. E. Akin. *Object-oriented programming via Fortran 90/95*. Cambridge University Press, Cambridge; New York, 2003.

[2] C. Amrouche, C. Bernardi, M. Dauge, and V. Girault. Vector potentials in three-dimensional non-smooth domains. *Mathematical Methods in the Applied Sciences*, 21(9):823–864, 1998.

[3] F. Armero and J. C. Simo. Formulation of a new class of fractional-step methods for the incompressible MHD equations that retains the long-term dissipativity of the continuum dynamical system. *Fields Institute Communications*, 10:1–23, 1996.

[4] F. Armero and J. C. Simo. Long-term dissipativity of time-stepping algorithms for an abstract evolution equation with applications to the incompressible MHD and Navier-Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 131(1-2):41 – 90, 1996.

[5] G. Authié, T. Tagawa, and R. Moreau. Buoyant flow in long vertical enclosures in the presence of a strong horizontal magnetic field. Part 2. Finite enclosures. *European Journal of Mechanics - B/Fluids*, 22(3):203–220, 2003.

[6] S. H. Aydın, A. I. Nesliturk, and M. Tezer-Sezgin. Two-level finite element method with a stabilizing subgrid for the incompressible MHD equations. *International Journal for Numerical Methods in Fluids*, 62(2):188–210, 2010.

[7] S. Badia. On stabilized finite element methods based on the Scott–Zhang projector. Circumventing the inf–sup condition for the Stokes problem. *Computer Methods in Applied Mechanics and Engineering*, 247–248(0):65–72, 2012.

[8] 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*, 107(4):533–557, 2007.

[9] S. Badia and R. Codina. Algebraic pressure segregation methods for the incompressible Navier-Stokes equations. *Archives of Computational Methods in Engineering*, 15:343–369, 2008.

[10] S. Badia and R. Codina. Pressure segregation methods based on a discrete pressure Poisson equation. An algebraic approach. *International Journal for Numerical Methods in Fluids*, 56(4):351–382, 2008.

[11] S. Badia and R. Codina. Unified stabilized finite element formulations for the Stokes and the Darcy problems. *SIAM Journal on Numerical Analysis*, 47(3):1971–2000, 2009.

[12] S. Badia and R. Codina. A combined nodal continuous–discontinuous finite element formulation for the maxwell problem. *Applied Mathematics and Computation*, 218(8):4276–4294, 2011.

[13] S. Badia and R. Codina. A nodal-based finite element approximation of the maxwell problem suitable for singular solutions. *SIAM Journal on Numerical Analysis*, 50(2):398–417, 2012.

[14] S. Badia and R. Codina. Stokes, Maxwell and Darcy: A single finite element approximation for three model problems. *Applied Numerical Mathematics*, 62:246–263, 2012.

[15] S. Badia, R. Codina, and J. Gutiérrez-Santacreu. Long-term stability estimates and existence of a global attractor in a finite element approximation of the Navier-Stokes equations with numerical sub-grid scale modeling. *SIAM Journal on Numerical Analysis*, 48(3):1013–1037, 2010.

[16] S. Badia, F. Guillén-González, and J. V. Gutiérrez-Santacreu. Finite element approximation of nematic liquid crystal flows using a saddle-point structure. *Journal of Computational Physics*, 230(4):1686–1706, 2011.

[17] S. Badia, A. F. Martín, and J. Principe. Enhanced balancing Neumann-Neumann preconditioning in computational fluid and solid mechanics. *International Journal for Numerical Methods in Engineering*, 2013. In press.

[18] S. Badia, A. F. Martín, and J. Principe. A highly scalable parallel implementation of balancing domain decomposition by constraints. 2013. Submitted.

[19] S. Badia, A. F. Martín, and J. Principe. Implementation and scalability analysis of balancing domain decomposition methods. *Archives of Computational Methods in Engineering*, 20(3):239–262, 2013.

[20] S. Badia, A. Quaini, and A. Quarteroni. Modular vs. non-modular preconditioners for fluid-structure systems with large added-mass effect. *Computer Methods in Applied Mechanics and Engineering*, 197(49-50):4216–4232, 2008.

[21] R. Becker and M. Braack. A finite element pressure gradient stabilization for the Stokes equations based on local projections. *Calcolo*, 38(4):173–199, 2001.

[22] M. A. Belenli, S. Kaya, L. G. Rebholz, and N. E. Wilson. A subgrid stabilization finite element method for incompressible magnetohydrodynamics. *International Journal of Computer Mathematics*, pages 1–18, in press.

[23] N. Ben Salah, A. Soulaimani, and W. Habashi. A finite element method for magnetohydrodynamics. *Computer Methods in Applied Mechanics and Engineering*, 190(43-44):5867 – 5892, 2001.

[24] N. Ben Salah, A. Soulaimani, W. Habashi, and M. Fortin. A conservative stabilized finite element method for the magneto-hydrodynamic equations. *International Journal for Numerical Methods in Fluids*, 29(5):535–554, 1999.

[25] 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.

[26] D. Boffi. Finite element approximation of eigenvalue problems. *Acta Numerica*, 19:1–120, 2010.

[27] A. Bonito and J. Guermond. Approximation of the eigenvalue problem for the time harmonic Maxwell system by continuous Lagrange finite elements. *Mathematics of Computation*, 80:1887–1910, 2011.

[28] G. Booch, J. Rumbaugh, and I. Jacobson. *The unified modeling language user guide*. Addison-Wesley, Upper Saddle River, NJ, 2005.

[29] L. Boulton and M. Strauss. Eigenvalue enclosures and convergence for the linearized MHD operator. *BIT Numerical Mathematics*, 52(4):801–825, 2012.

[30] 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*, 43(6):2544, 2005.

[31] J. H. Bramble, T. V. Kolev, and J. E. Pasciak. The approximation of the Maxwell eigenvalue problem using a least-squares method. *Mathematics of Computation*, 74(252):1575–1598, 2005.

[32] J. H. Bramble and J. E. Pasciak. A new approximation technique for div-curl systems. *Mathematics of Computation*, 73(248):1739–1762, 2004.

[33] S. Brenner and L. Scott. *The Mathematical Theory of Finite Element Methods*. Springer–Verlag, 1994.

[34] F. Brezzi and M. Fortin. *Mixed and Hybrid Finite Element Methods*. Springer Verlag, 1991.

[35] A. Brooks and T. 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.

[36] A. Buffa, P. C. Jr., and E. Jamelot. Solving electromagnetic eigenvalue problems in polyhedral domains with nodal finite elements. *Numerische Mathematik*, 113(4):497–518, 2009.

[37] L. Bühler. Liquid metal magnetohydrodynamics for fusion blankets. In R. M. S. Molokov and H. Moffat, editors, *Magnetohydrodynamics. Historical evolution and Trends*, pages 171–194. Springer, 2007.

[38] J. Cahouet and J. Chabard. Some fast 3D finite element solvers for the generalized stokes problem. *International Journal for Numerical Methods in Fluids*, 8(8):869–895, 1988.

[39] P. Causin, J. 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, 2005.

[40] L. Chacón. An optimal, parallel, fully implicit Newton–Krylov solver for three-dimensional viscoresistive magnetohydrodynamics. *Physics of Plasmas*, 15(5):056103, 2008.

[41] L. Chacón. Scalable parallel implicit solvers for 3D magnetohydrodynamics. *J. Phys.: Conf. Ser.*, 125:012041, 2008.

[42] A. Chorin. A numerical method for solving incompressible viscous problems. *Journal of Computational Physics*, 2:12–26, 1967.

[43] A. J. Chorin. Numerical solution of the navier-stokes equations. *Mathematics of Computation*, 22(104):745–762, 1968.

[44] A. J. Chorin, T. J. R. Hughes, M. F. McCracken, and J. E. Marsden. Product formulas and numerical algorithms. *Communications on Pure and Applied Mathematics*, 31(2):205–256, 1978.

[45] R. Codina. On stabilized finite element methods for linear systems of convection-diffusion-reaction equations. *Computer Methods in Applied Mechanics and Engineering*, 188:61–82, 2000.

[46] R. Codina. Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 190(13–14):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. Analysis of a stabilized finite element approximation of the Oseen equations using orthogonal subscales. *Applied Numerical Mathematics*, 58:264–283, 2008.

[51] 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*, 47:2900–2918, 2006.

[52] 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.

[53] R. Codina and N. Hernández. Stabilized finite element approximation of the stationary MHD equations. *Computational Mechanics*, 38:344–355, 2006.

[54] R. Codina and N. Hernández. Approximation of the thermally coupled MHD problem using a stabilized finite element method. *Journal of Computational Physics*, 230:1281–1303, 2011.

[55] R. Codina, J. Principe, O. Guasch, and S. Badia. Time dependent subscales in the stabilized finite element approximation of incompressible flow problems. *Computer Methods in Applied Mechanics and Engineering*, 196:2413–2430, 2007.

[56] M. Costabel. A coercive bilinear form for Maxwell's equations. *Journal of Mathematical Analysis and Applications*, 157(2):527–541, 1991.

[57] M. Costabel and M. Dauge. Singularities of electromagnetic fields in polyhedral domains. *Archives for Rational Mechanics and Analysis*, 151(3):221–276, 2000.

[58] M. Costabel and M. Dauge. Weighted regularization of Maxwell equations in polyhedral domains. *Numerische Mathematik*, 93(2):239–277, 2002.

[59] E. C. Cyr, J. N. Shadid, and R. S. Tuminaro. Stabilization and scalable block preconditioning for the Navier–Stokes equations. *Journal of Computational Physics*, 231(2):345–363, 2012.

[60] E. C. Cyr, J. N. Shadid, R. S. Tuminaro, R. P. Pawlowski, and L. Chacón. A new approximate block factorization preconditioner for two-dimensional incompressible (reduced) resistive MHD. *SIAM Journal on Scientific Computing*, 35(3):B701–B730, 2013.

[61] P. Davidson. *An introduction to magnetohydrodynamics*. Cambridge University Press, 2001.

[62] E. M. de les Valls, L. Sedano, L. Batet, I. Ricapito, A. Aiello, O. Gastaldi, and F. Gabriel. Lead-lithium eutectic material database for nuclear fusion technology. *Journal of Nuclear Materials*, 376:353–357, 2008.

[63] C. R. Dohrmann. A preconditioner for substructuring based on constrained energy minimization. *SIAM Journal on Scientific Computing*, 25(1):246, 2003.

[64] H.-Y. Duan, F. Jia, P. Lin, and R. C. E. Tan. The local L^2 projected C^0 finite element method for Maxwell problem. *SIAM Journal on Numerical Analysis*, 47(2):1274–1303, 2009.

[65] H. Elman. Preconditioners for saddle point problems arising in computational fluid dynamics. *Applied Numerical Mathematics*, 43:75–89, 2002.

[66] H. Elman, V. Howle, J. Shadid, R. Shuttleworth, and R. Tuminaro. A taxonomy and comparison of parallel block multi-level preconditioners for the incompressible Navier–Stokes equations. *Journal of Computational Physics*, 227(3):1790–1808, 2008.

[67] H. Elman, D. Silvester, and A. Wathen. *Finite Elements and Fast Iterative Solvers*. Oxford Science Publications, 2005.

[68] A. Ern and J. Guermond. *Theory and Practice of Finite Elements*. Springer Verlag, 2004.

[69] V. Fadeev, I. Kvabtskhava, and N. Komarov. Self-focusing of local plasma currents. *Nuclear Fusion*, 5:202–209, 1965.

[70] C. Farhat, M. Lesoinne, P. LeTallec, K. Pierson, and D. Rixen. FETI-DP: a dual-primal unified FETI method–part I: A faster alternative to the two- level FETI method. *International Journal for Numerical Methods in Engineering*, 50(7):1523–1544, 2001.

[71] J. Finn and P. Kaw. Self-focusing of local plasma currents. *Physics of Fluids*, 20:72–78, 1977.

[72] J. Gerbeau. A stabilized finite element method for the incompressible magnetohydrodynamic equations. *Numerische Mathematik*, 87:83–111, 2000.

[73] J. Gerbeau, C. L. Bris, and T. Lelièvre. *Mathematical Methods for the Magnetohydrodynamic Equations*. Oxford U, 2006.

[74] C. Greif, D. Li, D. Schotzau, and X. Wei. A mixed finite element method with exactly divergence-free velocities for incompressible magnetohydrodynamics. *Computer Methods in Applied Mechanics and Engineering*, 199(45–48):2840–2855, 2010.

[75] J. Guermond, P. Minev, and J. Shen. An overview of projection methods for incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 195(44-47):6011–6045, 2006.

[76] M. Gunzburger, A. Meir, and J. Peterson. On the existence, uniqueness, and finite element approximation of solutions of the equations of stationary, incompressible magnetohydrodynamics. *Math. Comp.*, 56:523–563, 1991.

[77] U. Hasler, A. Schneebeli, and D. Schötzau. Mixed finite element approximation of incompressible mhd problems based on weighted regularization. *Applied Numerical Mathematics*, 51:19 – 45, 2004.

[78] C. Hazard and M. Lenoir. On the solution of time-harmonic scattering problems for Maxwell's equations. *SIAM Journal on Mathematical Analysis*, 27(6):1597–1630, 1996.

[79] M. A. Heroux, R. A. Bartlett, V. E. Howle, R. J. Hoekstra, J. J. Hu, T. G. Kolda, R. B. Lehoucq, K. R. Long, R. P. Pawlowski, E. T. Phipps, A. G. Salinger, H. K. Thornquist, R. S. Tuminaro, J. M. Willenbring, A. Williams, and K. S. Stanley. An overview of the Trilinos project. *ACM Transactions on Mathematical Software*, 31(3):397–423, 2005.

[80] M. A. Heroux and J. M. Willenbring. Trilinos users guide. Technical Report SAND2003-2952, Sandia National Laboratories, 2003.

[81] P. Houston, D. Schötzau, and X. Wei. A mixed DG method for linearized incompressible magnetohydrodynamics. *Journal of Scientific Computing*, 40:281–314, 2009.

[82] P.-W. Hsieh and S.-Y. Yang. A bubble-stabilized least-squares finite element method for steady MHD duct flow problems at high hartmann numbers. *Journal of Computational Physics*, 228:8301–8320, 2009.

[83] T. 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.

[84] T. Hughes, G. Feijóo, L. Mazzei, and J. Quincy. The variational multiscale method–a paradigm for computational mechanics. *Computer Methods in Applied Mechanics and Engineering*, 166:3–24, 1998.

[85] T. Hughes, L. Franca, and G. 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.

[86] J. Hunt. Magnetohydrodynamic flow in rectangular ducts. *Journal of Fluid Mechanics*, 21:577–590, 1965.

[87] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.*, 20(1):359–392, 1998.

[88] D. Knoll and L. Chacon. Coalescence of magnetic islands, sloshing and the pressure problem. *Physics of Plasmas*, 13:032307, 2006.

[89] A. Kritz and D. Keyes. Fusion simulation project workshop report. *Journal of Fusion Energy*, 28:1–59, 2009.

[90] J. Lavers and L. Kadar. Application of electromagnetic forces to reduce tundish nozzle clogging. *Applied Mathematical Modelling*, 28(1):29–45, 2004.

[91] P. T. Lin, J. N. Shadid, R. S. Tuminaro, M. Sala, G. L. Hennigan, and R. P. Pawlowski. A parallel fully coupled algebraic multilevel preconditioner applied to multiphysics PDE applications: Drift-diffusion, flow/transport/reaction, resistive MHD. *International Journal for Numerical Methods in Fluids*, 64(10-12):1148–1179, 2010.

[92] J. Mandel. Balancing domain decomposition. *Communications in Numerical Methods in Engineering*, 9(3):233–241, 1993.

[93] J. Mandel and C. R. Dohrmann. Convergence of a balancing domain decomposition by constraints and energy minimization. *Numerical Linear Algebra with Applications*, 10(7):639–659, 2003.

[94] G. Matthies, P. Skrzypacz, and L. Tobiska. A unified convergence analysis for local projection stabilisations applied to the Oseen problem. *ESAIM: Mathematical Modelling and Numerical Analysis*, 41(4):713–742, 2007.

[95] C. Mistrangelo. Magnetohydrodynamic flow in a mock-up of a HCLL blanket. Part I. Numerical analysis. Technical Report FZKA 7312, Forschungszentrum Karlsruhe, 2008.

[96] C. Mistrangelo and L. Bühler. Influence of helium cooling channels on magnetohydrodynamic flows in the HCLL blanket. *Fusion Engineering and Design*, 84:1323–1328, 2009.

[97] P. Monk. *Finite Element Methods for Maxwell's Equations*. Oxford University Press, 2003.

[98] U. Müller and L. Bühler. *Magnetofluiddynamics in Channels and Containers*. Springer, 2001.

[99] J. C. Nedelec. Mixed finite elements in $\mathbb{R}^3$. *Numer. Meth.*, 35:119–136, 1980.

[100] J. C. Nedelec. A new family of mixed finite elements in $\mathbb{R}^3$. *Numer. Meth.*, 50:57–81, 1986.

[101] M.-J. Ni, R. Munipalli, P. Huang, N. Morley, and M. Abdou. A current density conservative scheme for incompressible MHD flows at a low magnetic Reynolds number. Part I: On a rectangular collocated grid system. *Journal of Computational Physics*, 227:174–204, 2007.

[102] M.-J. Ni, R. Munipalli, P. Huang, N. Morley, and M. Abdou. A current density conservative scheme for incompressible MHD flows at a low magnetic Reynolds number. Part II: On an arbitrary collocated mesh. *Journal of Computational Physics*, 227:205–228, 2007.

[103] L. F. Pavarino and O. B. Widlund. Balancing Neumann-Neumann methods for incompressible Stokes equations. *Communications on Pure and Applied Mathematics*, 55(3):302–335, 2002.

[104] M. J. D. Powell and M. A. Sabin. Piecewise quadratic approximations on triangles. *ACM Trans. Math. Softw.*, 3(4):316–325, 1977.

[105] J. Principe, R. Codina, and F. Henke. The dissipative structure of variational multiscale methods for incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 199:791–801, 2010.

[106] A. Prohl. Convergent finite element discretizations of the nonstationary incompressible magnetohydrodynamics system. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(06):1065–1087, 2008.

[107] R. Rieben, D. White, B. Wallin, and J. Solberg. An arbitrary Lagrangian-Eulerian discretization of MHD on 3D unstructured grids. *Journal of Computational Physics*, 226(1):534–570, 2007.

[108] D. Rouson, J. Xia, and X. Xu. *Scientific software design: the object-oriented way.* Cambridge University Press, New York, 2011.

[109] Y. Saad. *Iterative methods for sparse linear systems.* PWS Publishing, Boston, MA, 1996.

[110] N. B. Salah, A. Soulaimani, and W. Habashi. A finite element method for magnetohydrodynamics. *Computer Methods in Applied Mechanics and Engineering*, 190:5867–5892, 2001.

[111] D. Schötzau. Mixed finite element methods for stationary incompressible magnetohydrodynamics. *Numerische Mathematik*, 96:771–800, 2004.

[112] J. Shadid, R. Pawlowski, J. Banks, L. Chacón, P. Lin, and R. Tuminaro. Towards a scalable fully-implicit fully-coupled resistive MHD formulation with stabilized FE methods. *Journal of Computational Physics*, 229:7649–7671, 2010.

[113] J. Shercliff. Steady motion of conducting fluids in pipes under transverse magnetic fields. *Proceedings of Cambridge Philosophical Society*, 49:126–144, 1953.

[114] T. Sorokina and A. Worsey. A multivariate Powell–Sabin interpolant. *Advances in Computational Mathematics*, 29(1):71–89, 2008.

[115] P. L. Tallec and J. Mouro. Fluid structure interaction with large structural displacements. *Computer Methods in Applied Mechanics and Engineering*, 190:3039–3067, 2001.

[116] 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.

[117] R. Temam. *Navier-Stokes equations.* North-Holland, 1984.

[118] A. Toselli and O. Widlund. *Domain decomposition methods–algorithms and theory.* Springer Verlag, 2005.

[119] J. A. Zukas. *Introduction to Hydrocodes.* Elsevier, 2004.