

**title**

by

Michael Wathen

M.Sci., The University of Birmingham, 2012

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF  
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

The Faculty of Graduate Studies

(Computer Science)

THE UNIVERSITY OF BRITISH COLUMBIA

(Vancouver)

July 2014

© Michael Wathen 2014

# Abstract

# Preface

# Table of Contents

<b>Abstract</b>	ii
<b>Preface</b>	iii
<b>Table of Contents</b>	iv
<b>List of Tables</b>	vii
<b>List of Figures</b>	x
<b>1 Introduction</b>	1
1.1 A model problem in incompressible magnetohydrodynamics	1
1.2 Numerical solution	5
1.2.1 Finite Element methods	6
1.2.2 Solution techniques	8
1.3 Objectives and contributions	12
1.4 Outline	13
<b>2 Finite element discretisation</b>	14
2.1 Variational formulation	14
2.2 Mixed finite element discretisation	17

2.2.1	Matrix representation . . . . .	19
2.3	Picard iteration (P) . . . . .	21
2.4	Decoupled iterations . . . . .	23
2.4.1	Magnetic decoupling (MD) . . . . .	24
2.4.2	Complete decoupling (CD) . . . . .	25
2.5	Inhomogeneous boundary conditions and initial guess . . . . .	26
2.6	Summary . . . . .	28
<b>3</b>	<b>Preconditioning . . . . .</b>	<b>29</b>
3.1	Preconditioning the incompressible Navier-Stokes equations . . . . .	30
3.1.1	Pressure Convection-Diffusion (PCD) . . . . .	31
3.1.2	Least-Squares Commutator (LSC) . . . . .	32
3.1.3	PCD vs. LSC . . . . .	34
3.2	Preconditioning Maxwell's equations . . . . .	35
3.2.1	An ideal preconditioner . . . . .	35
3.2.2	A practical preconditioner . . . . .	36
3.3	Preconditioning the MHD equations . . . . .	36
3.3.1	Picard iteration . . . . .	37
3.3.2	Magnetic Decoupling . . . . .	38
3.3.3	Complete decoupling . . . . .	39
3.3.4	Summary . . . . .	41
<b>4</b>	<b>Numerical Results . . . . .</b>	<b>42</b>
4.1	Software . . . . .	42
4.2	Problem setup . . . . .	43
4.3	Numerical results: Navier-Stokes equations . . . . .	45

4.3.1	Convergence results for a 2D smooth solution . . . . .	45
4.3.2	Convergence results for a 3D smooth solution . . . . .	46
4.3.3	Preconditioning with LSC and PCD . . . . .	47
4.4	Numerical results: Maxwell's equations . . . . .	49
4.4.1	Convergence results for a 2D smooth solution . . . . .	49
4.4.2	Convergence results for a 3D smooth solution . . . . .	50
4.4.3	Preconditioning . . . . .	51
4.5	Numerical results: MHD problem . . . . .	52
4.5.1	Convergence results for a 2D smooth solution . . . . .	52
4.5.2	Convergence results for a 3D smooth solution . . . . .	54
4.5.3	Parameter tests . . . . .	56
4.6	Preconditioned MHD problem . . . . .	58
4.6.1	Picard Iteration . . . . .	59
4.6.2	Magnetic Decoupling . . . . .	61
4.6.3	Complete Decoupling . . . . .	64
<b>5</b>	<b>Conclusion and Future work . . . . .</b>	<b>66</b>
5.1	Conclusion . . . . .	66
5.2	Future work . . . . .	66
	<b>Bibliography . . . . .</b>	<b>67</b>

# List of Tables

3.1	Summary of coefficient and corresponding preconditioners for each iteration scheme . . . . .	41
4.1	Convergence for 2D Navier-Stokes smooth solution - $tol = 1e-10$	45
4.2	Convergence for 3D Navier-Stokes smooth solution - $tol = 1e-5$	46
4.3	Iteration table for LSC preconditioner for 2D example for various values of $\nu$ and $tol = 1e-10$ . . . . .	48
4.4	Iteration table for PCD preconditioner for 2D example for various values of $\nu$ and $tol = 1e-10$ . . . . .	48
4.5	Convergence for 2D Maxwell smooth solution - Magnetic field	49
4.6	Convergence for 2D Maxwell smooth solution - multiplier field	50
4.7	Convergence for 3D Maxwell smooth solution - magnetic field	50
4.8	Convergence for 3D Maxwell smooth solution - multiplier field	51
4.9	Iteration table for Maxwell preconditioner for 2D example - direct application of preconditioner . . . . .	51
4.10	Iteration table for Maxwell preconditioner for 3D example - direct application of preconditioner . . . . .	52
4.11	Convergence for 2D MHD smooth solution - $tol = 1e-8$ . . . .	53
4.12	Convergence for 2D MHD smooth solution - Magnetic field .	54

4.13	Convergence for 2D MDH smooth solution - multiplier field .	54
4.14	Convergence for 3D MHD smooth solution - $tol = 1e-8$ . . . .	55
4.15	Convergence for 3D MHD smooth solution - Magnetic field .	55
4.16	Convergence for 3D MDH smooth solution - multiplier field .	56
4.17	Number of non-linear iterations for various values of $\nu$ with $tol = 1e-5$ , $\kappa = 1$ and $\nu_m = 10$ . . . . .	57
4.18	Number of non-linear iterations for various values of $\kappa$ with $tol = 1e-5$ , $\nu = 1$ and $\nu_m = 10$ . . . . .	58
4.19	Number of non-linear and preconditioning iterations for various values of $\nu$ with $tol = 1e-5$ , $\kappa = 1$ and $\nu_m = 10$ . . . . .	59
4.20	Number of non-linear and preconditioning iterations for various values of $\kappa$ with $tol = 1e-5$ , $\nu = 1$ and $\nu_m = 10$ . . . . .	60
4.21	Number of non-linear and preconditioning iterations for various inner tolerances with outer tolerance $1e-6$ with $tol = 1e-5$ , $\nu = 1$ , $\kappa = 1$ and $\nu_m = 10$ . . . . .	61
4.22	Table number of non-linear iterations and number of iterations to solve the Navier-Stokes and Maxwells subproblem for the MD scheme with $tol = 1e-4$ , $\kappa = 1$ , $\nu = 1$ and $\nu_m = 10$ .	62
4.23	Table number of non-linear iterations and number of iterations to solve the Navier-Stokes and Maxwells subproblem for the MD scheme with $tol = 1e-4$ $\nu = 1$ and $\nu_m = 10$ . . . . .	62
4.24	Table number of non-linear iterations and number of iterations to solve the Navier-Stokes and Maxwells subproblem for the MD scheme with $tol = 1e-4$ , $\kappa = 1$ , $\nu = 1$ and $\nu_m = 10$ .	63



4.25	Table number of non-linear iterations and number of iterations to solve the Stokes and Maxwells subproblem for the CD scheme with $tol = 1e-4$ $\kappa = 1$ , and $\nu_m = 10$ . . . . .	64
4.26	Table number of non-linear iterations and number of iterations to solve the Stokes and Maxwells subproblem for the CD scheme with $tol = 1e-4$ $\nu = 1$ and $\nu_m = 10$ . . . . .	64
4.27	Table number of non-linear iterations and number of iterations to solve the Stokes and Maxwells subproblem for the CD scheme with $tol = 1e-4$ $\kappa = 1$ , $\nu = 1$ and $\nu_m = 100$ . . . .	65

# List of Figures

4.1	Level 3 grid on unit square domain . . . . .	44
-----	--	----

# Chapter 1

## Introduction

The primary topic of this thesis is to test a large scale implementation of an incompressible magnetohydrodynamics model. In this introduction chapter, we first will present a brief description of the problem studied. We then give a brief overview of Finite element methods and Krylov subspace methods for this problem. Finally, we outline the objectives and contributions of the thesis.

### 1.1 A model problem in incompressible magnetohydrodynamics

The area of incompressible magnetohydrodynamics (MHD) describes the behaviour of electrically conductive fluids (namely liquid metals, plasma, salt water, etc) in an electromagnetic field [13, 34, 42]. The model consist of a coupling between electromagnetism and fluid dynamics. The coupling effects are due to two fundamental physical properties. First, through the movement of the conductive material that induces a magnetic field which then modifies any existing electromagnetic field. Secondly, the magnetic and electric fields generate a mechanical force in the fluid known as the Lorentz

force. The Lorentz force accelerates the fluid, particles in the direction normal to both the electric and magnetic fields.

Incompressible MHD has a number of important applications within both technology and industry as well as Geophysical and Astrophysical applications. Some such applications are: electromagnetic pumping, aluminium electrolysis, the Earth's molten core and solar flares. For more applications see [42].

In this thesis we are principally interested in steady-state incompressible MHD. This means that the electrically conductive fluid is incompressible, i.e., the mass of the fluid is conserved, and the electric charge of the fluid cannot be ignored. The MHD model we consider consist of two coupled fundamental equations: the incompressible Navier-Stokes equations and Maxwell's equations. We will outline the derivation of a formulation of the full incompressible MHD system; for full details see [8].

The transient incompressible Navier-Stokes equations the govern incompressible fluid flow are given by:

$$\frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \nabla p = \mathbf{f} + \mathbf{f}_L \quad \text{in } \Omega \times (0, T), \quad (1.1a)$$

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

Here  $\mathbf{u}$  and  $p$  are the velocity and pressure of the fluid,  $\mathbf{f}$  denotes the body forces acting on the fluid and  $\mathbf{f}_L$  is the Lorentz force. The spatial domain is given by  $\Omega$  and end time is denoted by  $T$ . The parameter  $\nu$  is the kinematic viscosity. The mass conservation is given by (1.1b), see [19, Chapter 0] for the derivation of the incompressible Navier-Stokes equations

Maxwell's equations that govern electromagnetic effects are given by:

$$\text{Faraday's law:} \quad \frac{\partial \mathbf{b}}{\partial t} + \nabla \times \mathbf{e} = \mathbf{0}, \quad (1.2a)$$

$$\text{Coulomb's law:} \quad \nabla \cdot \mathbf{e} = \rho, \quad (1.2b)$$

$$\text{Ampère's law:} \quad -\frac{\partial(\delta \mathbf{e})}{\partial t} + \nabla \times \left(\frac{1}{\mu} \mathbf{b}\right) = \mathbf{j}, \quad (1.2c)$$

$$\text{Gauss's law:} \quad \nabla \cdot \mathbf{b} = 0. \quad (1.2d)$$

The fields in (1.2) are given by:  $\mathbf{b}$  the magnetic field,  $\mathbf{e}$  the electric field,  $\mathbf{j}$  the electric current density and  $\rho$  the electric charge density. The parameter  $\delta$  denotes the electric permittivity and  $\mu$  is the magnetic permeability, see [41, Chapter 1] for more details about Maxwell's equations.

Using the quasi-neutrality assumption in [8] (that is the positive and negative charges are equal in a given region), the displacement current  $\frac{\partial(\delta \mathbf{e})}{\partial t}$  can be neglected and the use of Ohm's law

$$\mathbf{j} = \theta(\mathbf{e} + \mathbf{u} \times \mathbf{b}),$$

it is possible to formulate (1.1) and (1.2) into the incompressible magneto-hydrodynamics system:

$$\frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \nabla p - \kappa (\nabla \times \mathbf{b}) \times \mathbf{b} = \mathbf{f}, \quad (1.3a)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (1.3b)$$

$$\frac{\partial \mathbf{b}}{\partial t} + \nu_m \nabla \times (\nabla \times \mathbf{b}) - \kappa \nabla \times (\mathbf{u} \times \mathbf{b}) = \mathbf{0}, \quad (1.3c)$$

$$\nabla \cdot \mathbf{b} = 0, \quad (1.3d)$$

with suitable initial conditions and boundary conditions. We note that  $\kappa$  is a non-dimensional parameter called the coupling number.

The solution to (1.3) depends on three non-dimensional parameters  $\nu = 1/\text{Re}$ ,  $\nu_m = 1/\text{Rm}$  and  $\kappa$ . The first parameter  $\text{Re}$  is the hydrodynamic Reynolds number, which indicates the balance between the inertial forces and the viscous forces. The parameter  $\text{Rm}$  is the magnetic Reynolds number, which measures the effect by which the magnetic field induces flow motion. The final parameter, the coupling number, is sometimes represented in terms of the Hartmann number, denoted by  $\text{Ha}$ , as

$$\kappa = \frac{\text{Ha}^2}{\text{Re Rm}}.$$

The Hartmann number represents the influence of the electromagnetic field on the flow. To find typical physical values for these parameter, we refer to [8, 34, 48].

In this thesis, we are interested in the steady-state ( $\frac{\partial}{\partial t} = 0$ ) MHD system of the form:

$$-\nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \nabla p - \kappa (\nabla \times \mathbf{b}) \times \mathbf{b} = \mathbf{f}, \quad (1.4a)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (1.4b)$$

$$\nu_m \nabla \times (\nabla \times \mathbf{b}) - \kappa \nabla \times (\mathbf{u} \times \mathbf{b}) = \mathbf{0}, \quad (1.4c)$$

$$\nabla \cdot \mathbf{b} = 0. \quad (1.4d)$$

In this thesis we will consider both two and three dimensional solutions to (1.4). The curl operator is well defined in three dimensions and the two

dimensional curl maybe defined as follows: given 2D vectors  $\mathbf{b} = (b_1, b_2)$ ,  $\mathbf{u} = (u_1, u_2)$  and the scalar function  $r$  then the curl and cross products are

$$\begin{aligned}\nabla \times \mathbf{b} &= \frac{\partial b_2}{\partial x} - \frac{\partial b_1}{\partial y}, \\ \nabla \times r &= \left( \frac{\partial r}{\partial y}, -\frac{\partial r}{\partial x} \right), \\ \mathbf{u} \times \mathbf{b} &= u_1 b_2 - u_2 b_1.\end{aligned}$$

Note that taking the curl of a 2D vector results in a scalar which is in the normal direction to the 2D field ( $z$ -component).

## 1.2 Numerical solution

The partial differential equation (PDE) system given in (1.4) requires a numerical approximation as in general an analytical solution is not possible. There are two main components in computing a numerical solution of a PDE:

1. Discretisation: take a continuous model and transfer it into a discrete model;
2. Solve: take the discretised model and solve for the unknowns.

In the sequel, we briefly describe these components in the context of the approach we take.

### 1.2.1 Finite Element methods

There are three principal discretisation methods: finite differences, finite volumes and finite elements.

There are several finite element methods for discretising the MHD problem (1.4). A common approach in the literature is to approximate the magnetic field using standard nodal  $H^1$ -conforming elements [8, 21, 25–27, 47]. Using such a formulation enables one to use the following vector calculus identity

$$-\Delta \mathbf{b} = \nabla \times (\nabla \times \mathbf{b}) - \nabla(\nabla \cdot \mathbf{b}). \quad (1.6)$$

Since  $\mathbf{b}$  is divergences-free then it is possible to apply an augmentation technique to replace the curl-curl operator with a vector Laplacian. This would then reduce one of the principal computational difficulties (the large null-space of the curl-curl operator (1.4c)). However, for non-convex domains (such as an L-shaped domain) the magnetic field will converge to a solution that is not correct around the singular point, see [11, 12]. Therefore, we would like to consider a mixed discretisation that captures singular solutions better. One such family of elements are  $H(\text{curl})$  conforming Nédélec elements [44]. This thesis will look at a mixed finite element method discretisation of (1.4).

To enable us to use Nédélec elements for the magnetic field we use the mixed formulation proposed in [52]. This leads to the following governing



equations in a domain  $\Omega$ :

$$-\nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \nabla p - \kappa (\nabla \times \mathbf{b}) \times \mathbf{b} = \mathbf{f} \quad \text{in } \Omega, \quad (1.7a)$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega, \quad (1.7b)$$

$$\kappa \nu_m \nabla \times (\nabla \times \mathbf{b}) + \nabla r - \kappa \nabla \times (\mathbf{u} \times \mathbf{b}) = \mathbf{g} \quad \text{in } \Omega, \quad (1.7c)$$

$$\nabla \cdot \mathbf{b} = 0 \quad \text{in } \Omega, \quad (1.7d)$$

where we have introduced the Lagrange multiplier,  $r$ , in the form of  $\nabla r$  in (1.7c) and changed the coefficient of the curl-curl operator in (1.7c). Again,  $\mathbf{u}$  and  $p$  are the velocity and pressure of the fluids and  $\mathbf{b}$  is the magnetic field. The introduction of Lagrange multiplier,  $r$ , corresponds to the divergence-free constraint of the magnetic field. With the addition of the Lagrange multiplier,  $r$ , we may introduce a generic forcing term  $\mathbf{g}$  associated with the Maxwell part of (1.7).

The numerical tests that we will consider will have inhomogeneous Dirichlet boundary conditions for the fluid and magnetic fields and homogeneous Dirichlet boundary condition for the multiplier field,  $r$ , of the form:

$$\mathbf{u} = \mathbf{u}_D \quad \text{on } \partial\Omega, \quad (1.8a)$$

$$\mathbf{n} \times \mathbf{b} = \mathbf{n} \times \mathbf{b}_D \quad \text{on } \partial\Omega, \quad (1.8b)$$

$$r = 0 \quad \text{on } \partial\Omega, \quad (1.8c)$$

with  $\mathbf{n}$  as the unit outward normal to the boundary  $\partial\Omega$ . Notice, by taking

the divergence of equation (1.7c) we obtain Poissons equation (as  $\nabla \cdot \nabla \times \mathbf{b} = 0$ ):

$$\Delta r = \nabla \cdot \mathbf{g} \quad \text{in } \Omega, \quad r = 0 \quad \text{on } \partial\Omega.$$

In many physical applications  $\mathbf{g}$  is divergence-free, this therefore implies the multiplier,  $r$ , is zero. In general, its main purpose is to provide stability, see [18].

### 1.2.2 Solution techniques

PDE discretisation, such as the one we are considering here, usually leads to sparse linear systems. For problems of small dimension, a popular choice to solve such a system are sparse direct method. However, for systems of large dimension, it is often not possible to use direct methods due to computational memory restrictions, time constraints, etc. Therefore an iterative solution, based on matrix-vector products, is required. The state of the art method many consider is to use a preconditioned Krylov subspace method [51].

Consider iteratively solving the linear system

$$Ax = b,$$

where  $A$  is a real non-singular  $n \times n$  matrix with vectors  $x$  and  $b$  of the same dimension. Suppose our initial guess is given by  $x_0$ , usually in practice this is taken to be zero, then the initial residual is given by  $r_0 = b - Ax_0$ . Using

the initial residual  $r_0$ , the  $m$ -dimensional Krylov subspace is defined as:

$$\mathcal{K}_m(A; r_0) = \text{span}\{r_0, Ar_0, A^2r_0, \dots, A^{m-1}r_0\}.$$

A Krylov subspace method is based on finding a solution,  $x_m$ , in the Krylov subspace:

$$A^{-1}b \approx x_m = x_0 + y_m,$$

where  $y \in \mathcal{K}_m(A; r_0)$  satisfies some optimality criteria. One could characterise the three main components to Krylov subspace methods by:

1. Computing an orthogonal basis to the Krylov subspace;
2. Defining an Optimality criteria within the Krylov subspace;
3. Preconditioning approaches.

### Orthogonal basis

Consider a general real unsymmetric matrix  $A$ , then the process in which it is possible to construct an orthogonal basis to the Krylov subspace  $\mathcal{K}_m(A; r_0)$  is known as the Arnoldi process. The Arnoldi process is essentially the modified Gram-Schmidt process applied to the Krylov subspace. This process is done iteratively, so at each new iteration a new orthogonal vector is added to the basis. The Arnoldi process is written as:

$$AQ_m = Q_{m+1}H_{m+1,m}, \tag{1.9}$$

where  $Q_{m+1}$  is the matrix whos  $m + 1$  columns form an orthogonal basis to

the subspace  $\mathcal{K}_m(A; r_0)$ .  $Q_m$  has the same leading  $m$  columns as in  $Q_{m+1}$  and  $H_{m+1,m} \in \mathbb{R}^{(m+1) \times m}$  is an upper Hessenberg matrix. From (1.9) one can deduce the following form, where  $H_{m,m}$  is the matrix containing the first  $m$  rows of  $H_{m+1,m}$ :

$$Q_m^T A Q_m = H_{m,m}. \quad (1.10)$$

If  $A$  is symmetric then from (1.10)  $H_{m,m}$  must also be symmetric. This leads to the Lanczos method where  $H_{m,m}$  is tridiagonal and hence, we define it as  $T_{m,m}$ . Then the Lanczos process is as follows

$$A Q_m = Q_{m+1} T_{m+1,m},$$

where  $T_{m+1,m}$  has an extra row than  $T_{m,m}$ . See [51] for more details about both the Arnoldi and Lanczos processes.

### Optimality criteria

There are two of the popular choices how to define the optimal criterion for a Krylov solver. These are minimising the  $L^2$ -norm within the Krylov subspace (minimum residual approach) and ensuring that the residual is orthogonal to the Krylov subspace (Galerkin approach).

For the three Krylov subspace methods that will be used within this thesis, namely GMRES [50] for unsymmetric systems, MINRES [46] for symmetric and Conjugate Gradients [29] (CG) for symmetric positive definite systems, the optimal criteria are as follows:

- GMRES:  $\min \|b - Ax_m\|_2$  i.e., minimum residual approach

- MINRES:  $\min \|b - Ax_m\|_2$  i.e., minimum residual approach
- CG:  $\min \|x_m - x\|_A$ , the energy norm of the error i.e., Galerkin approach

For more details about these methods and other Krylov subspace methods we refer the reader to [51].

### Preconditioning

Preconditioning is an integral part of using any Krylov subspace method. The convergence of CG but also other methods depends on the spread of the eigenvalues. Preconditioning the linear system  $Ax = b$  amounts to solving the system  $M^{-1}Ax = M^{-1}b$ , where  $M$  is a preconditioner. There are certain properties that are required to create a good preconditioner:

1. the preconditioner  $M$  must approximate  $A$  so that the eigenvalues of  $M^{-1}A$  are clustered,
2. systems associated with  $M$  should be much easier to solve than equations with  $A$ .

Loosely speaking, one could classify preconditioners by two types. The first type is an operator-based preconditioner. By this we mean that given a certain problem, usually based on a PDE discretisation, it may be possible to design certain operators that lead to an optimal or nearly optimal preconditioner. These operators are derived from an understanding of the physical problem and discretisation. Often these operators are based on spectral equivalence:  $M^{-1}A$  eigenvalues are independent of discretisation

parameters or weakly dependent. The second type is more of a “black-box” method, where the preconditioner is purely based on the matrix coefficients. The preconditioning strategies employed in this thesis are mainly based on the first technique.

In the literature there have not been too many approaches to preconditioning the MHD equations given in either the non-multiplier form (1.4) or with the multiplier form (1.7). In the very recent work [47], an operator-based preconditioner for the non-multiplier MHD equations (1.4) has been proposed. The identity (1.6) is used to form the preconditioner. The preconditioners we employ are based on the Navier-Stokes preconditioners in [19, Chapter 8] and the Maxwell preconditioner in [24].

### 1.3 Objectives and contributions

The aim of this thesis is to provide a full scalable iterative solution the MHD model (1.7), (1.8). We establish such methods and solve discrete problems in excess of 20 million degrees of freedom using them. Our numerical results show good scalability with respect to the mesh size. We provided a test of the relevant non-dimensional parameters to show robustness of the preconditioners. Very little has been done in terms of solution methods for the MHD model (1.7), (1.8) as stated above. The scalable solvers established in this thesis could lead to more investigation of the physical problems they describe. The availability of these solvers should enable more use of such MHD models.

## 1.4 Outline

This thesis is made up of five chapters and is structured as follows. In Chapter 2, we introduce a mixed finite element approximation to the MHD system (1.7). The mixed approximation is based on standard nodal  $H^1$ - $L^2$ -conforming finite element approximation for the fluid and pressure fields together with a mixed Nédélec element approximation for the magnetic field and the multiplier. Using this approximation, we introduce three possible non-linear iteration schemes.

In Chapter 3, we present an overview of the preconditioning approaches for the individual subproblems separately, namely the incompressible Navier-Stokes and Maxwell's equations. We then apply these preconditioning techniques to propose preconditioning strategies for the three non-linear iteration schemes from Chapter 2.

In Chapter 4, numerical experiments are discussed in both 2 and 3 spatial dimensions. We show convergence results for the full discretised MHD system along with the incompressible Navier-Stokes and Maxwell subproblems in isolation. Along with the convergence results we numerically test the preconditioning approaches to the three non-linear iteration schemes, providing heuristic tests with respect to the dimensionless parameters ( $\nu$ ,  $\nu_m$  and  $\kappa$ ). These tests examine the robustness both the preconditioners and the iteration schemes.

Chapter 5 outlines the conclusions and possible areas for future work.

## Chapter 2

# Finite element discretisation

In this chapter we introduce a mixed finite element discretisation for a steady-state incompressible MHD problem that model electrically conductive fluids. Following the setting in [52], we use curl-conforming elements for the magnetic field and conforming continuous elements for the velocity field. The resulting discretisation is verified through a series of numerical experiments which appear later in Chapter 4. For simplicity, we only discuss in detail homogeneous Dirichlet boundary conditions, that is

$$\mathbf{u} = \mathbf{0} \quad \text{and} \quad \mathbf{n} \times \mathbf{b} = \mathbf{0}. \quad (2.1)$$

Inhomogeneous conditions as in (1.8) can be incorporated as discussed in Section 2.5.

### 2.1 Variational formulation

To express the problem (1.7), (1.8) in weak form we follow [52] and denote the  $L^2$ -inner product on  $L^2(\Omega)^d$  by  $(\cdot, \cdot)_\Omega$ , for  $d = 2, 3$ . We introduce the



standard Sobolev spaces

$$\mathbf{V} = H_0^1(\Omega)^d = \left\{ \mathbf{u} \in H^1(\Omega)^d : \mathbf{u} = \mathbf{0} \text{ on } \partial\Omega \right\},$$

$$Q = L_0^2(\Omega) = \{ p \in L^2(\Omega) : (p, 1)_\Omega = 0 \},$$

$$\mathbf{C} = H_0(\text{curl}; \Omega) = \left\{ \mathbf{b} \in L^2(\Omega)^d : \nabla \times \mathbf{b} \in L^2(\Omega)^{2d-3}, \mathbf{n} \times \mathbf{b} = \mathbf{0} \text{ on } \partial\Omega \right\},$$

$$S = H_0^1(\Omega) = \{ r \in H^1(\Omega) : r = 0 \text{ on } \partial\Omega \}.$$

We write  $\|\cdot\|_{L^2(\Omega)}$ ,  $\|\cdot\|_{H^1(\Omega)}$  and  $\|\cdot\|_{H(\text{curl}; \Omega)}$  for the associated natural norms. More precisely, for vector fields  $\mathbf{u}, \mathbf{b}$  and a scalar functions  $r$  the norms are defined as follows:

$$\begin{aligned} \|\mathbf{u}\|_{L^2(\Omega)} &= \left( \int_{\Omega} \mathbf{u} \cdot \mathbf{u} \, dx \right)^{\frac{1}{2}}, \\ \|\mathbf{u}\|_{H^1(\Omega)} &= \left( \|\mathbf{u}\|_{L^2(\Omega)}^2 + \|\nabla \mathbf{u}\|_{L^2(\Omega)}^2 \right)^{\frac{1}{2}}, \\ \|\mathbf{b}\|_{H(\text{curl}, \Omega)} &= \left( \|\mathbf{b}\|_{L^2(\Omega)}^2 + \|\nabla \times \mathbf{b}\|_{L^2(\Omega)}^2 \right)^{\frac{1}{2}}, \\ \|r\|_{L^2(\Omega)} &= \left( \int_{\Omega} r^2 \, dx \right)^{\frac{1}{2}}, \\ \|r\|_{H^1(\Omega)} &= \left( \|r\|_{L^2(\Omega)}^2 + \|\nabla r\|_{L^2(\Omega)}^2 \right)^{\frac{1}{2}}, \end{aligned}$$

where  $\|\nabla \mathbf{u}\|_{L^2(\Omega)}^2$  is the  $L^2$ -norm of the gradient

$$\nabla \mathbf{u} = \begin{pmatrix} \frac{\partial u_1}{\partial x} & \frac{\partial u_1}{\partial y} & \frac{\partial u_1}{\partial z} \\ \frac{\partial u_2}{\partial x} & \frac{\partial u_2}{\partial y} & \frac{\partial u_2}{\partial z} \\ \frac{\partial u_3}{\partial x} & \frac{\partial u_3}{\partial y} & \frac{\partial u_3}{\partial z} \end{pmatrix}.$$

The weak formulation of the incompressible MHD system (1.7), (1.8) consists in finding  $(\mathbf{u}, p, \mathbf{b}, r) \in \mathbf{V} \times Q \times \mathbf{C} \times S$  such that

$$A(\mathbf{u}, \mathbf{v}) + O(\mathbf{u}; \mathbf{u}, \mathbf{v}) + C(\mathbf{b}; \mathbf{v}, \mathbf{b}) + B(\mathbf{v}, p) = (\mathbf{f}, \mathbf{v})_\Omega, \quad (2.2a)$$

$$B(\mathbf{u}, q) = 0, \quad (2.2b)$$

$$M(\mathbf{b}, \mathbf{c}) - C(\mathbf{b}; \mathbf{u}, \mathbf{c}) + D(\mathbf{c}, r) = (\mathbf{g}, \mathbf{c})_\Omega, \quad (2.2c)$$

$$D(\mathbf{b}, s) = 0, \quad (2.2d)$$

for all  $(\mathbf{v}, q, \mathbf{c}, s) \in \mathbf{V} \times Q \times \mathbf{C} \times S$ . The individual variational forms are given by

$$\begin{aligned} A(\mathbf{u}, \mathbf{v}) &= \int_{\Omega} \nu \nabla \mathbf{u} : \nabla \mathbf{v} \, d\mathbf{x}, \\ O(\mathbf{w}; \mathbf{u}, \mathbf{v}) &= \int_{\Omega} (\mathbf{w} \cdot \nabla) \mathbf{u} \cdot \mathbf{v} \, d\mathbf{x}, \\ B(\mathbf{u}, q) &= - \int_{\Omega} (\nabla \cdot \mathbf{u}) q \, d\mathbf{x}, \\ M(\mathbf{b}, \mathbf{c}) &= \int_{\Omega} \kappa \nu_m (\nabla \times \mathbf{b}) \cdot (\nabla \times \mathbf{c}) \, d\mathbf{x}, \\ D(\mathbf{b}, s) &= \int_{\Omega} \mathbf{b} \cdot \nabla s \, d\mathbf{x}, \\ C(\mathbf{d}; \mathbf{v}, \mathbf{b}) &= \int_{\Omega} \kappa (\mathbf{v} \times \mathbf{d}) \cdot (\nabla \times \mathbf{b}) \, d\mathbf{x}, \end{aligned} \quad (2.3)$$

where  $\nabla \mathbf{u} : \nabla \mathbf{v}$  is defined as

$$\nabla \mathbf{u} : \nabla \mathbf{v} = \sum_{i,j=1}^d (\nabla \mathbf{u})_{ij} (\nabla \mathbf{v})_{ij}.$$

In [52] it has been shown that this formulation of the problem is discrete energy-stable and has a unique solution for small data (i.e. for small  $\nu$ ,  $\nu_m$ ,

$\kappa$  and forcing terms  $\mathbf{f}$  and  $\mathbf{g}$  with small  $L^2$ -norms).

## 2.2 Mixed finite element discretisation

Consider the domain  $\Omega$  to be divided up into a regular and quasi-uniform mesh  $\mathcal{T}_h = \{K\}$  consisting of triangles ( $d = 2$ ) or tetrahedra ( $d = 3$ ) with mesh size  $h$ . Based on the function spaces defined in (2.2), our finite element approximation will be sought in the finite spaces given by:

$$\begin{aligned} \mathbf{V}_h &= \{ \mathbf{u} \in H^1(\Omega) : \mathbf{u}|_K \in \mathcal{P}_k(K)^d, K \in \mathcal{T}_h \}, \\ Q_h &= \{ p \in L^2(\Omega) \cap H^1(\Omega) : p|_K \in \mathcal{P}_{k-1}(K), K \in \mathcal{T}_h \}, \\ \mathbf{C}_h &= \{ \mathbf{b} \in H_0(\text{curl}; \Omega) : \mathbf{b}|_K \in \mathcal{P}_{k-1}(K)^d \oplus \mathbf{R}_k(K), K \in \mathcal{T}_h \}, \\ S_h &= \{ r \in H_0^1(\Omega) : r|_K \in \mathcal{P}_k(K), K \in \mathcal{T}_h \}, \end{aligned} \tag{2.4}$$

for  $k \geq 2$ . We define  $\mathcal{P}_k(K)$  as the space of polynomials of total degree at most  $k$  on  $K$  and  $\mathbf{R}_k(K)$  as the space of homogeneous vector polynomials of total degree  $k$  on  $K$  that are orthogonal to the position vector  $\mathbf{x}$ . Here we note that we are using  $\mathcal{P}_k/\mathcal{P}_{k-1}$  Taylor-Hood elements for the fluid unknowns  $(\mathbf{u}, p)$  [55]. For the magnetic variables  $(\mathbf{b}, r)$  we use the curl-conforming Nédélec elements of the first kind [44]. These choices of finite elements spaces  $\mathbf{V}_h$ ,  $\mathbf{C}_h$ ,  $Q_h$  and  $S_h$  imply that we have conforming subspaces to our Sobolev spaces  $\mathbf{V}$ ,  $\mathbf{C}$ ,  $Q$  and  $S$ , respectively. Then the finite element solution to (2.2) consists in finding  $(\mathbf{u}_h, p_h, \mathbf{b}_h, r_h) \in \mathbf{V}_h \times Q_h \times \mathbf{C}_h \times S_h$

such that

$$A(\mathbf{u}_h, \mathbf{v}) + \tilde{O}(\mathbf{u}_h; \mathbf{u}_h, \mathbf{v}) + C(\mathbf{b}_h; \mathbf{v}, \mathbf{b}_h) + B(\mathbf{v}, p_h) = (\mathbf{f}, \mathbf{v}), \quad (2.5a)$$

$$B(\mathbf{u}_h, q) = 0, \quad (2.5b)$$

$$M(\mathbf{b}_h, \mathbf{c}) - C(\mathbf{b}_h; \mathbf{u}_h, \mathbf{c}) + D(\mathbf{c}, r_h) = (\mathbf{g}, \mathbf{c}), \quad (2.5c)$$

$$D(\mathbf{b}_h, s) = 0, \quad (2.5d)$$

for all  $(\mathbf{v}, q, \mathbf{c}, s) \in \mathbf{V}_h \times Q_h \times \mathbf{C}_h \times S_h$ .

The forms  $A, M, B, D$  and  $C$  stay the same as on the continuous level. However, for the convection term  $\tilde{O}(\cdot; \cdot, \cdot)$  we to modify the form  $O(\mathbf{w}; \mathbf{u}, \mathbf{v})$  in a standard fashion to ensure the energy-stability property

$$\tilde{O}(\mathbf{w}; \mathbf{u}, \mathbf{u}) = 0, \quad \forall \mathbf{w}, \mathbf{u} \in \mathbf{V}_h. \quad (2.6)$$

To ensure this property we integrate by parts the convection form  $O(\mathbf{w}; \mathbf{u}, \mathbf{u})$  to obtain

$$\int_{\Omega} (\mathbf{w} \cdot \nabla) \mathbf{u} \cdot \mathbf{u} \, d\mathbf{x} = -\frac{1}{2} \int_{\Omega} \nabla \cdot \mathbf{w} \mathbf{u} \cdot \mathbf{u} \, d\mathbf{x} + \frac{1}{2} \int_{\partial\Omega} \mathbf{w} \cdot \mathbf{n} |\mathbf{u}|^2 \, ds,$$

recalling that  $\mathbf{n}$  is the unit outward normal on  $\partial\Omega$ . Therefore, we choose the modified convection form  $\tilde{O}(\mathbf{w}; \mathbf{u}, \mathbf{v})$  as

$$\tilde{O}(\mathbf{w}; \mathbf{u}, \mathbf{v}) = \int_{\Omega} (\mathbf{w} \cdot \nabla) \mathbf{u} \cdot \mathbf{v} \, d\mathbf{x} + \frac{1}{2} \int_{\Omega} \nabla \cdot \mathbf{w} \mathbf{u} \cdot \mathbf{v} \, d\mathbf{x} - \frac{1}{2} \int_{\partial\Omega} \mathbf{w} \cdot \mathbf{n} \mathbf{u} \cdot \mathbf{v} \, ds.$$

By construction, property (2.6) is now satisfied. Note also that for homoge-

neous boundary conditions as assumed in (2.1), the boundary integral term in  $\tilde{O}$  can be omitted.

Again in [52] it has been shown that this formulation of a MHD problem is discretely energy-stable and has a unique solution for small data. Also, optimal order error estimates in the mesh size  $h$  have been derived for small data using the stability property (2.6). Namely, for sufficiently smooth solutions, we have the error bound

$$\|\mathbf{u} - \mathbf{u}_h\|_{H^1(\Omega)} + \|\mathbf{b} - \mathbf{b}_h\|_{H(\text{curl};\Omega)} + \|p - p_h\|_{L^2(\Omega)} + \|r - r_h\|_{H^1(\Omega)} \leq Ch^k,$$

for a constant  $C > 0$  independent of the mesh size. In addition, the  $L^2$ -norm error for the velocity field are of order  $\mathcal{O}(h^{k+1})$  (as  $\mathbf{V}_h$  consists of a full polynomial space on each element). However, we cannot expect  $L^2$ -norm errors of order  $\mathcal{O}(h^{k+1})$  for the magnetic field (as  $\mathbf{C}_h$  does not consists of a full polynomial space on each element).

### 2.2.1 Matrix representation

This variational formulation (2.5) now can be converted into a matrix representation. To do this, we introduce the basis function for the finite elements spaces in (2.4):

$$\mathbf{V}_h = \text{span}\langle \boldsymbol{\psi}_j \rangle_{j=1}^{n_u}, \quad Q_h = \text{span}\langle \alpha_i \rangle_{i=1}^{m_u}, \quad (2.7)$$

$$\mathbf{C}_h = \text{span}\langle \boldsymbol{\phi}_j \rangle_{j=1}^{n_b}, \quad S_h = \text{span}\langle \beta_i \rangle_{i=1}^{m_b}. \quad (2.8)$$

The aim now is to find the coefficient vectors  $u = (u_1, \dots, u_{n_u}) \in \mathbb{R}^{n_u}$ ,  $p = (p_1, \dots, p_{m_u}) \in \mathbb{R}^{m_u}$ ,  $b = (b_1, \dots, b_{n_b}) \in \mathbb{R}^{n_b}$ , and  $r = (r_1, \dots, r_{m_b}) \in \mathbb{R}^{m_b}$  of the finite element functions  $(\mathbf{u}_h, p_h, \mathbf{b}_h, r_h)$ . As usual, this is done by writing the bilinear forms in (2.5) in terms of the following stiffness matrices and load vectors:

$$\begin{aligned}
A_{i,j} &= A(\boldsymbol{\psi}_j, \boldsymbol{\psi}_i), & 1 \leq i, j \leq n_u, \\
B_{i,j} &= B(\boldsymbol{\psi}_j, \alpha_i), & 1 \leq i \leq m_u, \ 1 \leq j \leq n_u, \\
D_{i,j} &= D(\boldsymbol{\phi}_j, \beta_i), & 1 \leq i \leq m_b, \ 1 \leq j \leq n_b, \\
M_{i,j} &= M(\boldsymbol{\phi}_j, \boldsymbol{\phi}_i), & 1 \leq i, j \leq n_b, \\
f_i &= (\mathbf{f}, \boldsymbol{\psi}_i)_\Omega, & 1 \leq i \leq n_u, \\
g_i &= (\mathbf{g}, \boldsymbol{\phi}_i)_\Omega, & 1 \leq i \leq n_b.
\end{aligned}$$

For the two non-linear forms,  $\tilde{O}$  and  $C$ , we define the corresponding stiffness matrices with respect to given finite element functions  $\mathbf{w} \in \mathbf{V}_h$  and  $\mathbf{d}_h \in \mathbf{C}_h$  in the first argument and their associated coefficient vectors  $w$  and  $d$  as

$$\begin{aligned}
O(w)_{i,j} &= \tilde{O}(\mathbf{w}; \boldsymbol{\psi}_j, \boldsymbol{\psi}_i), & 1 \leq i, j \leq n_u, \\
C(d)_{i,j} &= C(\mathbf{d}; \boldsymbol{\psi}_j, \boldsymbol{\phi}_i), & 1 \leq i \leq n_b, \ 1 \leq j \leq n_u.
\end{aligned}$$

Thus, the numerical solution to (1.7) consists in solving the non-linear

system

$$\begin{pmatrix} A + O(u) & B^T & C^T(b) & 0 \\ B & 0 & 0 & 0 \\ -C(b) & 0 & M & D^T \\ 0 & 0 & D & 0 \end{pmatrix} \begin{pmatrix} u \\ p \\ b \\ r \end{pmatrix} = \begin{pmatrix} f \\ 0 \\ g \\ 0 \end{pmatrix}. \quad (2.9)$$

where the vectors  $u \in \mathbb{R}^{n_u}$ ,  $p \in \mathbb{R}^{m_u}$ ,  $b \in \mathbb{R}^{n_b}$ , and  $r \in \mathbb{R}^{m_b}$  are the unknown coefficients of the finite element functions.

### 2.3 Picard iteration (P)

The discrete system (2.9) is non-linear, and therefore applying a non-linear solver to this problem is necessary. A common choice to deal with the non-linearity within the incompressible Navier-Stokes equations in isolation is to perform Oseen or Picard iterations [19]. This involves linearising around the current velocity and solving for updates.

We adapt this approach for the full MHD system as well. Given a current iterate  $(\mathbf{u}_h, p_h, \mathbf{b}_h, r_h)$  we solve for updates  $(\delta \mathbf{u}_h, \delta p_h, \delta \mathbf{b}_h, \delta r_h)$  and introduce the next iterate by setting:

$$\begin{aligned} \mathbf{u}_h &\rightarrow \mathbf{u}_h + \delta \mathbf{u}_h, & p_h &\rightarrow p_h + \delta p_h, \\ \mathbf{b}_h &\rightarrow \mathbf{b}_h + \delta \mathbf{b}_h, & r_h &\rightarrow r_h + \delta r_h. \end{aligned}$$

In variational form, the updates  $(\delta \mathbf{u}_h, \delta p_h, \delta \mathbf{b}_h, \delta r_h) \in \mathbf{V}_h \times Q_h \times \mathbf{C}_h \times S_h$

are found by solving the Picard system (P):

$$A(\delta \mathbf{u}_h, \mathbf{v}) + \tilde{O}(\mathbf{u}; \delta \mathbf{u}_h, \mathbf{v}) + C(\mathbf{b}_h; \mathbf{v}, \delta \mathbf{u}_h) + B(\mathbf{v}, \delta p_h) = R_u(\mathbf{u}_h, \mathbf{b}_h, p_h; \mathbf{v}),$$

$$B(\delta \mathbf{u}_h, q) = R_p(\mathbf{u}_h; q),$$

$$M(\delta \mathbf{b}_h, \mathbf{c}) + D(\mathbf{c}, \delta r_h) - C(\mathbf{b}_h; \delta \mathbf{u}_h, \mathbf{v}) = R_b(\mathbf{u}_h, \mathbf{b}_h, r_h; \mathbf{c}),$$

$$D(\delta \mathbf{b}_h, s) = R_r(\mathbf{b}_h; s),$$

for all  $(\mathbf{v}, q, \mathbf{c}, s) \in \mathbf{V}_h \times Q_h \times \mathbf{C}_h \times S_h$ . Note that the system is linearised around  $(\mathbf{u}_h, \mathbf{b}_h)$ . The right-hand side linear forms correspond to the residual at the current iteration  $(\mathbf{u}_h, p_h, \mathbf{b}_h, r_h)$  defined by:

$$R_u(\mathbf{u}_h, \mathbf{b}_h, p_h; \mathbf{v}) = (\mathbf{f}, \mathbf{v})_\Omega - A(\mathbf{u}_h, \mathbf{v}) - \tilde{O}(\mathbf{u}_h; \mathbf{u}_h, \mathbf{v}) - C(\mathbf{b}_h; \mathbf{v}, \mathbf{b}_h) - B(\mathbf{v}, p_h),$$

$$R_p(\mathbf{u}_h; q) = -B(\mathbf{u}_h, q),$$

$$R_b(\mathbf{u}_h, \mathbf{b}_h, r_h; \mathbf{c}) = (\mathbf{g}, \mathbf{c})_\Omega - M(\mathbf{b}_h, \mathbf{c}) + C(\mathbf{b}_h; \mathbf{u}_h, \mathbf{c}) - D(\mathbf{c}, r_h),$$

$$R_r(\mathbf{b}_h; s) = -D(\mathbf{b}_h, s),$$

for all  $(\mathbf{v}, q, \mathbf{c}, s) \in \mathbf{V}_h \times Q_h \times \mathbf{C}_h \times S_h$ .

In [52] it is shown that for small data the Picard iteration (P) will converge to the exact solution for any initial guess.

To formulate the variational form of the Picard iteration (P) in matrix form, let  $(u, p, b, r)$  be the coefficient vectors associated with  $(\mathbf{u}_h, p_h, \mathbf{b}_h, r_h)$  and  $(\delta u, \delta p, \delta b, \delta r)$  be the coefficient vectors of  $(\delta \mathbf{u}_h, \delta p_h, \delta \mathbf{b}_h, \delta r_h)$ . Then it can readily be seen that the Picard iteration (P) amounts to solving the matrix



system

$$\begin{pmatrix} A + O(u) & B^T & C(b)^T & 0 \\ B & 0 & 0 & 0 \\ -C(b) & 0 & M & D^T \\ 0 & 0 & D & 0 \end{pmatrix} \begin{pmatrix} \delta u \\ \delta p \\ \delta b \\ \delta r \end{pmatrix} = \begin{pmatrix} r_u \\ r_p \\ r_b \\ r_r \end{pmatrix}, \quad (2.10)$$

with

$$r_u = f - Au - O(u)u - C(b)^T b - B^T p,$$

$$r_p = -Bu,$$

$$r_b = g - Mu + C(b)b - D^T r,$$

$$r_r = -Db.$$

Here, the matrix  $A$  is symmetric positive definite (SPD),  $O(u)$  is non-symmetric and  $-C(b), C(b)^T$  appear in a skew symmetric fashion. We also note that  $M$  is symmetric positive semidefinite (SPSD) with nullity  $m_b$  corresponding to the discrete gradients, see [24].

## 2.4 Decoupled iterations

The full MHD system (1.7), (1.8) is a coupled system consisting of the incompressible Navier-Stokes and Maxwell's equations, coupled through the non-linear skew symmetric coupling term  $C(b)$ . In addition, the convection term  $O(u)$  is non-linear as well. These two terms make the numerical solution challenging. Therefore, if one or both of these terms is small then it

may be possible to iterate explicitly. In particular if the coupling term,  $C(b)$ , is small then we may completely decouple the system into a Navier-Stokes problem and a Maxwell problem. The two resulting decoupling schemes are what we call Magnetic and Complete Decoupling and are both described below.

#### 2.4.1 Magnetic decoupling (MD)

Consider the first situation where there is weak coupling within the system, that is when  $C(b)$  is small. Then it may be possible to drop these terms to completely decouple the system into the two subproblems, the Navier-Stokes and Maxwell's equations. We will call this Magnetic Decoupling. Then (2.10) amounts to

$$\begin{pmatrix} A + O(u) & B^T & 0 & 0 \\ B & 0 & 0 & 0 \\ 0 & 0 & M & D^T \\ 0 & 0 & D & 0 \end{pmatrix} \begin{pmatrix} \delta u \\ \delta p \\ \delta b \\ \delta r \end{pmatrix} = \begin{pmatrix} r_u \\ r_p \\ r_b \\ r_r \end{pmatrix}, \quad (2.11)$$

with

$$r_u = f - Au - O(u)u - C(b)^T b - B^T p,$$

$$r_p = -Bu,$$

$$r_b = g - Mu + C(b)b - D^T r,$$

$$r_r = -Db.$$

From (2.11) we can see that the system is now completely decoupled. This enable us to solve each individual subproblem separately and possibly in parallel.

### 2.4.2 Complete decoupling (CD)

For the second decoupling scheme, we again consider there to be weak coupling of the system but we also consider that the fluid equations are diffusion-dominated and hence can exclude the convection terms. This amounts to

$$\begin{pmatrix} A & B^T & 0 & 0 \\ B & 0 & 0 & 0 \\ 0 & 0 & M & D^T \\ 0 & 0 & D & 0 \end{pmatrix} \begin{pmatrix} \delta u \\ \delta p \\ \delta b \\ \delta r \end{pmatrix} = \begin{pmatrix} r_u \\ r_p \\ r_b \\ r_r \end{pmatrix}, \quad (2.12)$$

with

$$r_u = f - Au - O(u)u - C(b)^T b - B^T p,$$

$$r_p = -Bu,$$

$$r_b = g - Mu + C(b)b - D^T r,$$

$$r_r = -Db.$$

This is the simplest technique as it removes all non-linear terms and hence leaves the linear Stokes problem in the upper  $(1, 1)$  block matrix.

## 2.5 Inhomogeneous boundary conditions and initial guess

In this section, we described the formulation of the MHD system (1.7), (2.1), that is with homogeneous Dirichlet boundary conditions. In general, the problems we numerically test have inhomogeneous Dirichlet boundary conditions (1.8). When considering inhomogeneous Dirichlet boundary conditions, we consider solving (2.10), (2.11) and (2.12) for the solution updates with homogeneous Dirichlet boundary conditions. This therefore means that we must incorporate the inhomogeneous Dirichlet boundary conditions within the initial guess.

To start the non-linear iteration schemes given in Section 2.4 we require an initial guess. To form the initial guess, we solve the decoupled Stokes problem (with (1.8))

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

then the non-symmetric Maxwell problem (with (1.8))

$$\begin{pmatrix} M - C & D^T \\ D & 0 \end{pmatrix} \begin{pmatrix} b \\ r \end{pmatrix} = \begin{pmatrix} g \\ 0 \end{pmatrix}.$$

Here the term  $C$  corresponds the the coupling term using  $u$  (the initial guess for the velocity field). We use the coupling term within the non-symmetric Maxwell problem because we have the initial guess for the velocity field and

incorporating this into initial guess for the magnetic field and multiplier which will hopefully increase the accuracy.

When iteratively solving these two systems, the convergence tolerance which we use is important. Consider the fluid equations, solving the Stokes equations for the initial guess with inhomogeneous boundary conditions. If the discrete Stokes problem is solved approximately to some tolerance, then we approximately solve the boundary equations of the form:

$$1 \times u_B = u_D,$$

where  $u_B$  is the coefficients of the solution on the boundary and  $u_D$  is the given boundary data. That is that the initial guess only starts with boundary values of the same order as the initial solve. The outcome is that we must solve for the initial guesses very accurately in order that correct boundary conditions are applied. A similar consideration applies to the non-symmetric Maxwell problem.

If the subsequent solution updates (which have homogeneous Dirichlet boundary conditions) are solved accurately, the inaccuracy in the boundary values will remain. For similar reasons, low accuracy of the solution updates will lead to perturbation in the boundary conditions and thus in the whole solution.

## 2.6 Summary

In this chapter we reviewed a mixed finite element approximation to the full MHD system given in (1.7) and (1.8). We followed the mixed approach outlined in [52] and expressed the MHD system in the matrix form (2.10). Using the Picard iteration (2.10) we introduced two possible decoupling schemes which maybe simpler to solve for depending on the parameters ( $\kappa$ ,  $\nu$  and  $\nu_m$ ). However, it may not be always possible to decouple the system hence we will consider three possibilities denoted by (P), (MD) and (CD). The next chapter will discuss possible preconditioning approaches to these systems.

In the sequel, we shall omit the dependence of  $O(u)$  and  $C(b)$  on  $b$  and  $u$ , respectively, and simply write  $O$  and  $C$ .

## Chapter 3

# Preconditioning

The linear system (2.10) is typically sparse and of large dimensions, hence to efficiently solve for it we use a preconditioned iterative approach as proposed in [35]. We start by reviewing some preconditioning strategies for the incompressible Navier-Stokes and Maxwell subproblems in isolation. From these techniques we will then introduce and numerically test preconditioners for the full MHD system.

Let us introduce the a number of matrices in addition to the ones already defined in Chapter 2, which are used for preconditioning: the velocity mass matrix  $Q$ , pressure mass matrix  $Q_p$ , pressure convection diffusion operator  $F_p$ , pressure Laplacian  $A_p$ , scalar Laplacian  $L$  and the mass matrix  $X$  by

$$Q_{i,j} = \int_{\Omega} \boldsymbol{\psi}_j \cdot \boldsymbol{\psi}_i d\mathbf{x}, \quad 1 \leq i, j \leq n_u, \quad (3.1a)$$

$$(Q_p)_{i,j} = \int_{\Omega} \alpha_j \alpha_i dx, \quad 1 \leq i, j \leq m_u, \quad (3.1b)$$

$$(F_p)_{i,j} = \nu \int_{\Omega} \nabla \alpha_j \cdot \nabla \alpha_i + (\mathbf{w} \cdot \nabla \alpha_j) \alpha_i dx, \quad 1 \leq i, j \leq m_u, \quad (3.1c)$$

$$(A_p)_{i,j} = \int_{\Omega} \nabla \alpha_j \cdot \nabla \alpha_i dx, \quad 1 \leq i, j \leq m_u, \quad (3.1d)$$

$$L_{i,j} = \int_{\Omega} \nabla \beta_j \cdot \nabla \beta_i d\mathbf{x}, \quad 1 \leq i, j \leq m_b \quad (3.1e)$$

$$X_{i,j} = \int_{\Omega} \boldsymbol{\psi}_j \cdot \boldsymbol{\psi}_i d\mathbf{x}, \quad 1 \leq i, j \leq n_b. \quad (3.1f)$$

### 3.1 Preconditioning the incompressible

#### Navier-Stokes equations

Consider the steady state incompressible Navier-Stokes equations in isolation. Let

$$\mathcal{K}_{\text{NS}} = \begin{pmatrix} F & B^T \\ B & 0 \end{pmatrix}, \quad (3.2)$$

be the discretised and linearised Navier-Stokes subproblem where  $F = A + O$  has been defined in (2.3). Due to the convection term,  $O$ , this system is non-symmetric and we will use GMRES to solve this subproblem [50]. A common approach for solving a saddle point system like this is to use a block diagonal or block triangular preconditioner of the form

$$\mathcal{M}_{\text{tNS}} = \begin{pmatrix} F & B^T \\ 0 & -S \end{pmatrix} \quad \text{or} \quad \mathcal{M}_{\text{dNS}} = \begin{pmatrix} F & 0 \\ 0 & S \end{pmatrix}, \quad (3.3)$$

where  $S \approx BF^{-1}B^T$  i.e.,  $S$  approximates the Schur complement. If  $S$  is *precisely* the Schur complement then it has been proved in [43] that the preconditioned matrix has exactly 2 eigenvalues when using the block triangular preconditioner (i.e.,  $\mathcal{M}_{\text{tNS}}^{-1}\mathcal{K}_{\text{NS}}$ ) or 3 distinct eigenvalues in the block diagonal case (i.e.,  $\mathcal{M}_{\text{dNS}}^{-1}\mathcal{K}_{\text{NS}}$ ). This therefore means that for a suitable Krylov subspace method such as the minimum residual iterative scheme as GMRES will converge in exactly 2 iterations or 3 iterations.

In practice it is too expensive to form and solve for the Schur complement, hence, a good approximation to it is needed. Two well known pre-



conditioners for the incompressible Navier-Stokes equations are the Least Squares Commutator (LSC) and the Pressure Convection-Diffusion (PCD) preconditioners. A description of both can be found in [19] and we will just outline the procedure how these can be applied on the discrete level.

### 3.1.1 Pressure Convection-Diffusion (PCD)

In [19, Chap. 8] the discrete commutator of the convection-diffusion operator associated with the gradient operation is introduced and given by

$$\epsilon_h = (Q^{-1}F)(Q^{-1}B^T) - (Q^{-1}B^T)(Q_p^{-1}F_p). \quad (3.4)$$

where  $Q$ ,  $Q_p$  and  $F_p$  are defined in (3.1). Assuming that the commutator is small then pre- and post-multiplying (3.4) by  $BF^{-1}Q$  and  $F_p^{-1}Q_p$ , respectively, let us separate out the Schur complement to give

$$BF^{-1}B^T \approx BQ^{-1}B^TF_p^{-1}Q_p. \quad (3.5)$$

Our discretisation is inf-sup stable which means that there is spectral equivalence between  $BQ^{-1}B^T$  and the pressure Laplacian,  $A_p$ ; see [19, Section 5.5.1]. Hence, the Schur complement can be approximated by:

$$S_{\text{PCD}} = A_pF_p^{-1}Q_p.$$

Applying the PCD preconditioner to the full Navier-Stokes system involves

solving the system

$$\begin{pmatrix} F & B^T \\ 0 & -A_p F_p^{-1} Q_p \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix}$$

at each Krylov iteration. This can be solved efficiently by splitting it into the following two steps

1. Solve for  $y$ :  $y = -Q_p^{-1} F_p A_p^{-1} b$ ;
2. Solve for  $x$ :  $x = F^{-1}(a - B^T y)$ .

This means that we have one pressure Poisson solve ( $A_p^{-1}$ ), one mass matrix solve ( $Q_p^{-1}$ ) and one convection-diffusion solve ( $F^{-1}$ ) at each Krylov iteration. It is possible to solve these iteratively using multigrid and iterative methods. However, to test the preconditioning for the non-linear iteration schemes we will use direct solver.

### 3.1.2 Least-Squares Commutator (LSC)

The disadvantage with using the PCD preconditioner is that it requires construction of the operators  $A_p$ ,  $Q_p$  and  $F_p$ . The second approach to approximate the Schur complement is LSC which uses the matrix coefficients of (3.2) to form the preconditioner

As for the derivation of the PCD preconditioner we start off with the discrete commutator of the convection-diffusion operator

$$\epsilon_h = (Q^{-1} F)(Q^{-1} B^T) - (Q^{-1} B^T)(Q_p^{-1} F_p).$$

Suppose that the  $Q$ -norm is defined by  $\|v\|_Q = (Qv, v)^{1/2}$ . Then this time we minimise  $\epsilon_h$  over the columns of  $[F_p]_j$  in the  $Q$ -norm to try to find an expression for  $F_p$ . The minimisation is given by

$$\min \| [Q^{-1}FQ^{-1}B^T]_j - Q^{-1}B^TQ_p^{-1}[F_p]_j \|_Q.$$

Solving this optimisation problem, as shown in [19], is done by solving the normal equations

$$Q_p^{-1}BQ^{-1}B^TQ_p^{-1}F_p = Q_p^{-1}BQ^{-1}FQ^{-1}B^T.$$

This yields the following expression for  $F_p$ :

$$F_p = Q_p(BQ^{-1}B^T)^{-1}(BQ^{-1}FQ^{-1}B^T).$$

By substituting this into expression (3.5) we obtain the LSC approximation to the Schur complement:

$$S = BF^{-1}B^T \approx S_{\text{LSC}} = (BQ^{-1}B^T)(BQ^{-1}FQ^{-1}B^T)^{-1}(BQ^{-1}B^T).$$

Therefore, applying the LSC preconditioner to the full Navier-Stokes system  $\mathcal{K}_{\text{NS}}$  in (3.2) involves solving for the matrix

$$\begin{pmatrix} F & B^T \\ 0 & -S_{\text{LSC}} \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a \\ b \end{pmatrix}$$

at each Krylov iteration. Again, this can be split up into the following two

steps:

1. Solve for  $y$ :  $y = -(BQ^{-1}B^T)^{-1}(BQ^{-1}FQ^{-1}B^T)(BQ^{-1}B^T)^{-1}b$ ;
2. Solve for  $x$ :  $x = F^{-1}(a - B^Ty)$ .

Hence, we have two pressure Poisson solves  $((BQ^{-1}B^T)^{-1})$  and one Convection-Diffusion solve  $(F^{-1})$  at each Krylov iteration. In practice, we take the diagonal or lumped diagonal of  $Q$  to form  $BQ^{-1}B^T$ . These solves, as with the PCD preconditioner, will be done directly.

### 3.1.3 PCD vs. LSC

The main advantage of solving the commutator using the least-squares approach is that the the matrices that define the preconditioner are available from the original system. Whereas, for PCD we require the construction of the matrices  $A_p$ ,  $Q_p$  and  $F_p$ . Therefore, LSC is slightly more computationally efficient to form. However, to apply the LSC preconditioner we require two pressure Poisson solves where as PCD only requires ones.

We will consider experiments with both preconditioners for the incompressible Navier-Stokes problem in isolation to determine which seems more effective. This preconditioner will then be applied within the solver for the full MHD system.

## 3.2 Preconditioning Maxwell's equations

Next, consider the Maxwell subproblem

$$\mathcal{K}_{\text{MX}} = \begin{pmatrix} M & D^T \\ D & 0 \end{pmatrix}. \quad (3.6)$$

As we did for the Navier-Stokes subproblem in Section 3.1, we apply a block preconditioning strategy for  $\mathcal{K}_{\text{MX}}$  in (3.6). On the other hand, (3.6) is symmetric and hence we will focus on SPD block diagonal preconditioners.

### 3.2.1 An ideal preconditioner

The  $(1,1)$  block of  $\mathcal{K}_{\text{MX}}$  is the curl-curl operator, hence, the matrix  $M$  is singular, where the null space is the gradients of dimension  $m_b$ . Therefore the usual Schur complement does not exist as the matrix  $M$  cannot be inverted. To overcome this difficulty, we employ the approach in [22, 23] based on augmentation and use preconditioners of the form:

$$\begin{pmatrix} M + D^T W^{-1} D & 0 \\ 0 & W \end{pmatrix},$$

where  $W$  is a symmetric positive definite matrix.

It has been shown in [24] that an appropriate choice of  $W$  is the scalar Laplacian,  $L$ , defined in (3.1). This leads to the ideal preconditioner:

$$\mathcal{M}_{\text{iMX}} = \begin{pmatrix} M + D^T L^{-1} D & 0 \\ 0 & L \end{pmatrix}. \quad (3.7)$$

Applying (3.7) as the preconditioner yields exactly two eigenvalues, 1 and  $-1$  of algebraic multiplicities of  $n_b$  and  $m_b$ , respectively. Therefore using this matrix as a preconditioner means that MINRES will converge in two iterations, in the absence of roundoff errors [46]. However, forming the matrix  $M + D^T L^{-1} D$  is costly, hence,  $\mathcal{M}_{\text{iMX}}$  is impractical for large systems.

### 3.2.2 A practical preconditioner

A good approximation for  $M + D^T L^{-1} D$  is required to make the ideal preconditioner,  $\mathcal{M}_{\text{iMX}}$ , suitable in practise. It has been shown in [24] that  $M + D^T L^{-1} D$  is spectrally equivalent to  $M + X$  where  $X$  is the mass matrix on the magnetic space defined in (3.1). Using this approximation yields the practical preconditioner

$$\mathcal{M}_{\text{MX}} = \begin{pmatrix} N & 0 \\ 0 & L \end{pmatrix}, \quad (3.8)$$

where  $N = M + X$  is a shifted curl-curl operator. A scalable multigrid solver has been developed in [30] which involves two Laplacian solves on the scalar space. However, the construction of this multigrid solver is very involved and hence we will consider direct solves for this preconditioner but the implementation is a possible area of future work.

## 3.3 Preconditioning the MHD equations

In Section 2.3 and 2.4 we introduced three iteration schemes, namely Picard iteration (P), Magnetic Decoupling (MD) and Complete Decoupling (CD).

Using the results from sections 3.2 and 3.1 we will discuss the preconditioning approaches that we imply for these iteration schemes.

### 3.3.1 Picard iteration

In Sections 3.1 and 3.2 we looked briefly at the preconditioning strategies for the Navier-Stokes and Maxwell's equations. Using these techniques we will look at possible scalable preconditioners for the full MHD problem,

$$\mathcal{K}_{\text{MH}} = \begin{pmatrix} A + O & B^T & C^T & 0 \\ B & 0 & 0 & 0 \\ -C & 0 & M & D^T \\ 0 & 0 & D & 0 \end{pmatrix}. \quad (3.9)$$

Using the Navier-Stokes and Maxwell subproblem preconditioners (3.3) and (3.8) respectively, then we propose the following preconditioner for  $\mathcal{K}_{\text{MH}}$

$$\mathcal{M}_{\text{MH}} = \begin{pmatrix} F & B^T & C^T & 0 \\ 0 & -\hat{S} & 0 & 0 \\ -C & 0 & N & 0 \\ 0 & 0 & 0 & L \end{pmatrix}, \quad (3.10)$$

where  $\hat{S}$  is either the LSC or PCD approximation to the fluid flow Schur complement. The preconditioned matrix,  $\mathcal{M}_{\text{MH}}^{-1}\mathcal{K}_{\text{MH}}$ , has eigenvalues  $\lambda = 1$  with algebraic multiplicity of at least  $n_u + n_b$  and an eigenvalue  $\lambda = 1$  with algebraic multiplicity of at least  $m_b$  see [35]. Due to the coupling terms,  $C$ , the application of this preconditioner is hard. To overcome this,

we propose to invert  $\mathcal{M}_{\text{MH}}$  by means of an inner preconditioner. The inner preconditioner is given by

$$\mathcal{M}_{\text{innerMH}} = \begin{pmatrix} F & B^T & 0 & 0 \\ 0 & -\hat{S} & 0 & 0 \\ 0 & 0 & N & 0 \\ 0 & 0 & 0 & L \end{pmatrix}. \quad (3.11)$$

Here the preconditioner matrix,  $\mathcal{M}_{\text{innerMH}}^{-1}\mathcal{M}_{\text{MH}}$ , has an eigenvalue  $\lambda = 1$  with algebraic multiplicity of at least  $m_u + n_u + 3m_b - n_b$  see [35].

### 3.3.2 Magnetic Decoupling

From Section 2.4.1 the the matrix to be preconditioned is as follows:

$$\mathcal{K}_{\text{MD}} = \left( \begin{array}{cc|cc} F & B^T & 0 & 0 \\ B & 0 & 0 & 0 \\ \hline 0 & 0 & M & D^T \\ 0 & 0 & D & 0 \end{array} \right). \quad (3.12)$$

Recall that removing the coupling terms completely decouples the system. This therefore enables us to use the preconditioners for each of the subproblems separately and in parallel. Using the subproblem preconditioners (3.8)



and (3.3) then the optimal preconditioner for  $\mathcal{K}_{\text{MD}}$  is

$$\mathcal{M}_{\text{MD}} = \left( \begin{array}{cc|cc} F & B^T & 0 & 0 \\ 0 & -\hat{S} & 0 & 0 \\ \hline 0 & 0 & N & 0 \\ 0 & 0 & 0 & L \end{array} \right). \quad (3.13)$$

### 3.3.3 Complete decoupling

From Section 2.4.2 the the matrix to be preconditioned is as follows:

$$\mathcal{K}_{\text{CD}} = \left( \begin{array}{cc|cc} A & B^T & 0 & 0 \\ B & 0 & 0 & 0 \\ \hline 0 & 0 & M & D^T \\ 0 & 0 & D & 0 \end{array} \right). \quad (3.14)$$

First we consider how to deal with the upper (2,2) block matrix which corresponds to the discrete Stokes equations

$$\mathcal{K}_{\text{S}} = \left( \begin{array}{cc} A & B^T \\ B & 0 \end{array} \right).$$

As with the incompressible Navier-Stokes subproblem the idea for the Stokes preconditioner is again to approximate the Schur complement. The Schur complement associated with the Stokes system is

$$S_{\text{S}} = BA^{-1}B^T,$$

recall that the matrix  $A$  is defined with the viscosity  $\nu$  in section 2.1. It was shown in [53, 54] that the scaled pressure mass matrix,  $\frac{1}{\nu}W$  defined in (3.1), is spectrally equivalent to the Schur complement (which is also a consequence of the inf-sup stability condition). Therefore a possible scalable Stokes preconditioner is

$$\begin{pmatrix} A & 0 \\ 0 & \frac{1}{\nu}W \end{pmatrix}.$$

Using (3.15) together with the Maxwell subproblem preconditioner (3.8) gives the preconditioner

$$\mathcal{M}_{\text{CD}} = \left( \begin{array}{cc|cc} A & 0 & 0 & 0 \\ 0 & \frac{1}{\nu}W & 0 & 0 \\ \hline 0 & 0 & N & 0 \\ 0 & 0 & 0 & L \end{array} \right). \quad (3.15)$$

The biggest advantage of this decoupling approach is that the matrix system is now symmetric. This means that the appropriate choice for the Krylov subspace method is MINRES for each subproblem.

### 3.3.4 Summary

In summary, we proposed three preconditioning approaches to the iteration scheme proposed in Chapter 2. Table 3.1 references the coefficient matrices together with the associated preconditioner. Note that for the Picard iteration

Iteration scheme	Coefficient matrix	(Outer) preconditioner
(P)	$\mathcal{K}_{\text{MH}}$ in (3.9)	$\mathcal{M}_{\text{MH}}$ in (3.10)
(MD)	$\mathcal{K}_{\text{MD}}$ in (3.12)	$\mathcal{M}_{\text{MD}}$ in (3.13)
(CD)	$\mathcal{K}_{\text{CD}}$ in (3.14)	$\mathcal{M}_{\text{CD}}$ in (3.15)

Table 3.1: Summary of coefficient and corresponding preconditioners for each iteration scheme

(P) we employ an inner preconditioner  $\mathcal{M}_{\text{innerMH}}$  in (3.11) to solve systems corresponding with (3.10).

## Chapter 4

# Numerical Results

In this chapter we present a series of convergence and preconditioning experiments. The principal aim is to check the scalability performance of the preconditioned iterative methods for the MHD problem and the two decoupling schemes proposed in Chapter 3. All numerical experiments have been carried out in the Python programming language.

### 4.1 Software

Due to the complex nature of the MHD problem, a number of different libraries have been used both to discretise and then to solve the resulting systems.

The finite element software that was used to discretise (1.7), (1.8) is **FEniCS** [56]. The core libraries used within **FEniCS** are the problem-solving interface **DOLFIN** [39, 40], the compiler for finite element variational forms **FFC** [33, 38, 45], the finite element tabulator **FIAT** [31, 32], the just-in-time compiler **Instant**, the code generator **UFC** [2, 3] and the form language **UFL** [1, 4].

Along with **FEniCS** we have used a number of stand-alone linear algebra libraries. The main one that has been used is **PETSc4PY** which is the **python**

interface for PETSc [9, 10]. PETSc has been used mainly for the iterative solvers as well as the blockwise preconditioning setup. The following packages were used to solve the precondition system: HYPRE [20] as a multigrid solver and the sparse direct solvers UMFPACK [14–17], PASTIX [28], SuperLU [36, 37] and MUMPS [5–7].

## 4.2 Problem setup

When numerically solving a set of linear or non-linear equations the problem needs to be initialised: the mesh sequence, non-linear iteration stopping criteria and initial guess setup **needs** to be defined. In this small section we will briefly cover these aspects.

### Mesh sequence

To test the performance of the preconditioners from chapter 3 and to validate the code produced by FEniCS a sequence of meshes are required. To verify that the code produces the correct convergence rates, we consider levels,  $l$ , of uniformly refined grids. The levels define the number of edges between the nodes along the boundary edge to be  $2^l$ . For example, consider the third level ( $l = 3$ ) **hence**, the grid generated is the  $8 \times 8$  grid given in figure 4.1. The first column of all convergence and iteration tables below will show the grid level.

### **Stopping criteria**

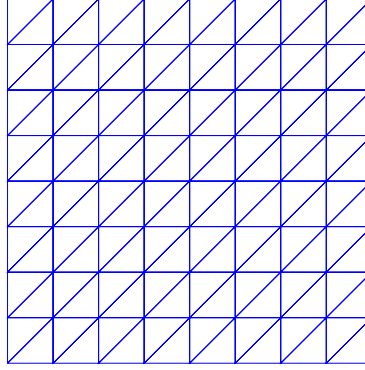


Figure 4.1: Level 3 grid on unit square domain

Recall that both the incompressible Navier-Stokes equations as well as the full MHD problem are a set of non-linear equations. Section 2.3 outlines the process in which we linearise the problem and then solve for the updates. The stopping criteria that we enforce for the Navier-Stokes and MHD problems are:

- NS:  $\|\delta u\|_p + \|\delta p\|_p < tol$ ,
- MHD:  $\|\delta u\|_p + \|\delta p\|_p + \|\delta b\|_p + \|\delta r\|_p < tol$ ,

where  $p = \infty$ . The tolerance is stated for each example run within this thesis.

### 4.3 Numerical results: Navier-Stokes equations

Before considering the full discretised MHD problem, we require that the Navier-Stokes subproblem perform as expected in terms of the error estimates and preconditioner scalability. To check the validity of the FEniCS code, we introduce two smooth test problems, the first in 2D and the second in 3D.

#### 4.3.1 Convergence results for a 2D smooth solution

Consider the domain  $\Omega = (0, 1)^2$  with boundary  $\partial\Omega$ . Taking the kinematic viscosity to be  $\nu = 1$ , we choose the source terms  $\mathbf{f}$  and inhomogeneous Dirichlet boundary conditions from the analytical solution

$$\begin{aligned}\mathbf{u}(x, y) &= \begin{pmatrix} \sin(x) \exp(x + y) + \cos(y) \exp(x + y) \\ -\sin(y) \exp(x + y) \end{pmatrix}, \\ p(x, y) &= x^3 \sin(y) + \exp(x + y).\end{aligned}$$

Running the FEniCS code which produces the Table 4.1 of results:

$l$	Dofs $\mathbf{u}_h/p_h$	$\ \mathbf{u} - \mathbf{u}_h\ _{L^2(\Omega)}$	order	$\ \mathbf{u} - \mathbf{u}_h\ _{H^1(\Omega)}$	order	$\ p - p_h\ _{L^2(\Omega)}$	order
3	578/81	2.2100e-04	-	1.5645e-02	-	7.2758e-03	-
4	2,178/289	2.7598e-05	3.00	3.9133e-03	2.00	1.8190e-03	2.00
5	8,450/1,089	3.4465e-06	3.00	9.7822e-04	2.00	4.5025e-04	1.99
6	33,282/4,225	4.3072e-07	3.00	2.4455e-04	2.00	1.1341e-04	1.99
7	132,098/16,641	5.3836e-08	3.00	6.1137e-05	2.00	2.8639e-05	1.99
8	526,338/66,049	6.7291e-09	3.00	1.5284e-05	2.00	7.0146e-06	2.03

Table 4.1: Convergence for 2D Navier-Stokes smooth solution -  $10^{-10}$

For the lowest-order ( $k = 2$ ) Taylor-Hood elements the expected conver-

gence rates for the velocity field in the  $L^2$ - and  $H^1$ -norm errors are third and second order and for the pressure the  $L^2$ -norm error is second order. From table 4.1 we can see that we obtain third order in  $L^2$ -norm error, second order for  $H^1$ -norm errors in the velocity fields and second order in  $L^2$ -norm errors for the pressures. This is precisely what we expect from Taylor-Hood elements.

### 4.3.2 Convergence results for a 3D smooth solution

The three-dimensional set up is very similar as with the 2D case. This time the domain is the unit cube, i.e.  $\Omega = (0, 1)^3$  with boundary  $\partial\Omega$ . As before the kinematic viscosity is  $\nu = 1$ , then the source term  $\mathbf{f}$  and inhomogeneous Dirichlet boundary conditions are calculated from the analytical solution

$$\begin{aligned} \mathbf{u}(x, y, z) &= \begin{pmatrix} -\exp(x + y + z) \sin(y) + \exp(x + y + z) \sin(z) \\ \exp(x + y + z) \sin(x) - \exp(x + y + z) \sin(z) \\ -\exp(x + y + z) \sin(x) + \exp(x + y + z) \sin(y) \end{pmatrix}, \\ p(x, y, z) &= \exp(x + y + z) + \sin(y). \end{aligned}$$

Running the 3D code produces Table 4.2 which shows the errors and convergence rates.

$l$	Dofs $\mathbf{u}_h/p_h$	$\ \mathbf{u} - \mathbf{u}_h\ _{L^2(\Omega)}$	order	$\ \mathbf{u} - \mathbf{u}_h\ _{H^1(\Omega)}$	order	$\ p - p_h\ _{L^2(\Omega)}$	order
1	375/27	2.6211e-02	-	4.5804e-01	-	1.7725e+00	-
2	2,187/125	3.2997e-03	2.99	1.1547e-01	1.99	2.8602e-01	2.63
3	14,739/729	4.1267e-04	3.00	2.8944e-02	2.00	4.0587e-02	2.82
4	107,811/4,913	5.1565e-05	3.00	7.2416e-03	2.00	6.4794e-03	2.65
5	823,875/35,937	6.4443e-06	3.00	1.8108e-03	2.00	1.2724e-03	2.35

Table 4.2: Convergence for 3D Navier-Stokes smooth solution -  $\text{tol} = 1\text{e-}5$



From the table we can see that the convergence rates are the same as the 2D test solution in the previous section for the velocity field. However, we observe that for the pressure field we get a slightly higher than expected order, namely about 2.5. We have seen this trend for multiple different examples that have been run but never have we seen an  $L^2$ -norm error order lower than 2. We therefore accept this convergence behaviour.

### 4.3.3 Preconditioning with LSC and PCD

Section 3.1 outlines the two main preconditioning techniques for the Navier-Stokes equations. In order to have a scalable preconditioner for the full MHD system we need to check the performance of both LSC and PCD for the Navier-Stokes subproblem. We use GMRES as the Krylov subspace solver (with a convergence tolerance of  $1e-5$ ) and the application of both the preconditioners will be done directly. The tables show average number of iterations which means we calculate the average the number of GMRES iterations at each non-linear iteration.

The outline of how to form and apply the LSC preconditioner can be found in Section 3.1.2.

Table 4.3 shows the average number of GMRES iterations for various values of  $\nu$  and grid levels. We observe that as the grid level increases the average number of GMRES iterations increases. This is a relatively unknown problem when using Taylor-Hood elements. Therefore we consider using PCD as the Navier-Stokes subproblem preconditioner instead.

As with the LSC preconditioner an outline of the PCD preconditioner can be found in Section 3.1.1. Applying this preconditioner to the incompressible

l	Dofs	Average iterations			
	$\mathbf{u}_h/p_h$	$\nu = 10$	$\nu = 1$	$\nu = 0.1$	$\nu = 0.01$
1	50/9	5	5	6	8
2	162/25	10	10	10	21
3	578/81	16	16	15	30
4	2,178/289	25	24	24	31
5	8,450/1,089	48	47	46	39
6	33,282/4,225	103	86	77	63
7	132,098/16,641	190	190	141	131

Table 4.3: Iteration table for LSC preconditioner for 2D example for various values of  $\nu$  and  $tol = 1e-10$

Navier-Stokes problem in isolation gives Table 4.4 of results.

Running the 2D example gives Table 4.4.

l	Dofs	Average iterations			
	$\mathbf{u}_h/p_h$	$\nu = 10$	$\nu = 1$	$\nu = 0.1$	$\nu = 0.01$
5	8450/1089	17	15	18	5053
6	33,282/4,225	18	15	18	144
7	132,098/16,641	18	15	18	41
8	526,338/66,049	18	15	18	40
9	2,101,250/263,169	18	15	18	40

Table 4.4: Iteration table for PCD preconditioner for 2D example for various values of  $\nu$  and  $tol = 1e-10$

From Table 4.4 we can see that as the mesh level increases then the average number of GMRES iteration numbers stay roughly constant. This is what we expect and require for the Navier-Stokes subproblem preconditioner. Also, the table shows iteration number for multiple different values of the kinematic viscosity  $\nu$ . We also not that as  $\nu$  decreases from 1 to 0.01 the number of average iterations increase slightly but not vastly.

## 4.4 Numerical results: Maxwell's equations

As for the Navier-Stokes subproblem (Section 4.3), we consider a 2D and 3D test solution to validate the FEniCS code for the Maxwell subproblem.

### 4.4.1 Convergence results for a 2D smooth solution

Again, consider the domain  $\Omega = (0,1)^2$  with boundary  $\partial\Omega$ . Now choose the source terms  $\mathbf{g}$  and inhomogeneous Dirichlet boundary conditions that satisfy the analytical solution

$$\begin{aligned}\mathbf{b}(x, y) &= \begin{pmatrix} \exp(x+y) \cos(x) \\ \exp(x+y) \sin(x) - \exp(x+y) \cos(x) \end{pmatrix}, \\ r(x, y) &= \sin(2\pi x) \sin(2\pi y).\end{aligned}$$

l	Dofs $\mathbf{b}_h/r_h$	$\ \mathbf{b} - \mathbf{b}_h\ _{L^2(\Omega)}$	order	$\ \mathbf{b} - \mathbf{b}_h\ _{H(\text{curl}, \Omega)}$	order
1	48/25	9.3833e-02	-	1.0696e-01	-
2	176/81	2.3350e-02	2.01	2.7088e-02	1.98
3	672/289	5.8372e-03	2.00	6.4220e-03	2.08
4	2,624/1,089	1.4597e-03	2.00	1.4586e-03	2.14
5	10,368/4,225	3.6497e-04	2.00	3.5066e-04	2.06
6	41,216/16,641	9.1246e-05	2.00	8.6688e-05	2.02
7	164,352/66,049	2.2812e-05	2.00	2.1609e-05	2.00

Table 4.5: Convergence for 2D Maxwell smooth solution - Magnetic field

For Maxwell equations we use second order ( $k = 2$ ) Nédélec elements of the first kind [44] and polynomial elements of the same order for the magnetic and multiplier fields respectively. Therefore, we expect second order convergence in  $\|\mathbf{b} - \mathbf{b}_h\|_{L^2(\Omega)}$ ,  $\|\mathbf{b} - \mathbf{b}_h\|_{H(\text{curl}, \Omega)}$  and  $\|r - r_h\|_{H^1(\Omega)}$  and third order in  $\|r - r_h\|_{L^2(\Omega)}$ . Tables 4.5 and 4.6 show that we do obtain the

l	Dofs $\mathbf{b}_h/r_h$	$\ r - r_h\ _{L^2(\Omega)}$	order	$\ r - r_h\ _{H^1(\Omega)}$	order
1	48/25	2.7761e-01	-	2.8932e+00	-
2	176/81	4.3540e-02	2.67	9.3299e-01	1.63
3	672/289	4.8633e-03	3.16	2.5904e-01	1.85
4	2,624/1,089	5.6724e-04	3.10	6.6810e-02	1.96
5	10,368/4,225	6.9363e-05	3.03	1.6841e-02	1.99
6	41,216/16,641	8.6203e-06	3.01	4.2193e-03	2.00
7	164,352/66,049	1.0760e-06	3.00	1.0554e-03	2.00

Table 4.6: Convergence for 2D Maxwell smooth solution - multiplier field

expected rates of convergence for both the magnetic and multiplier fields.

#### 4.4.2 Convergence results for a 3D smooth solution

As for the 3D Navier-Stokes problem the domain is the unit cube, i.e.  $\Omega = (0, 1)^3$  with boundary  $\partial\Omega$ . Using the analytical solution

$$\mathbf{b}(x, y, z) = \begin{pmatrix} -\exp(x + y + z) \sin(y) + \exp(x + y + z) \sin(z) \\ \exp(x + y + z) \sin(x) - \exp(x + y + z) \sin(z) \\ -\exp(x + y + z) \sin(x) + \exp(x + y + z) \sin(y) \end{pmatrix},$$

$$r(x, y, z) = \sin(2\pi x) \sin(2\pi y) \sin(2\pi z),$$

the source term  $\mathbf{g}$  and inhomogeneous Dirichlet boundary conditions are defined.

l	Dofs $\mathbf{b}_h/r_h$	$\ \mathbf{b} - \mathbf{b}_h\ _{L^2(\Omega)}$	order	$\ \mathbf{b} - \mathbf{b}_h\ _{H(curl, \Omega)}$	order
1	436/125	5.3312e-02	0.00	3.5258e-01	0.00
2	2,936/729	1.4192e-02	1.91	8.9944e-02	1.97
3	21,424/4,913	3.5801e-03	1.99	2.2690e-02	1.99
4	163,424/35,937	8.9697e-04	2.00	5.7585e-03	1.98

Table 4.7: Convergence for 3D Maxwell smooth solution - magnetic field

1	Dofs $\mathbf{b}_h/r_h$	$\ r - r_h\ _{L^2(\Omega)}$	order	$\ r - r_h\ _{H^1(\Omega)}$	order
1	436/125	2.2689e-01	0.00	2.8923e+00	0.00
2	2,936/729	5.9452e-02	1.93	1.1782e+00	1.30
3	21,424/4,913	6.9068e-03	3.11	3.3901e-01	1.80
4	163,424/35,937	7.5913e-04	3.19	9.0082e-02	1.91

Table 4.8: Convergence for 3D Maxwell smooth solution - multiplier field

Tables 4.7 and 4.8 show the same orders of convergence as in the 2D case from Section 4.4.1.

#### 4.4.3 Preconditioning

Section 3.2 introduces the preconditioning strategy for Maxwell's equations that will be employed in this thesis. Two- and three-dimensional results are presented in this section.

#### 2D example

Table 4.9 shows the performance of using the preconditioner  $\mathcal{M}_{\text{MX}}$  with a MINRES tolerance of 1e-6 and a direct application of the preconditioner.


1	Dofs $\mathbf{b}_h/r_h$	Number of iterations			
		$\nu_m = 10$	$\nu_m = 100$	$\nu_m = 1000$	$\nu_m = 10000$
5	10,368/4,225	5	4	6	6
6	41,216/16,641	5	6	6	6
7	164,352/66,049	5	6	6	6
8	656,384/263,169	5	6	6	8
9	2,623,488/1,050,625	4	6	6	8
10	10,489,856/4,198,401	4	6	8	10

Table 4.9: Iteration table for Maxwell preconditioner for 2D example - direct application of preconditioner


From Table 4.9 we can see that as the number of MINRES iterations

stays about constant as the mesh level,  $l$ , increases. Note that as the magnetic viscosity ( $\nu_m$ ) increases then the number of iterations remains roughly the same.

### 3D example



Running the 3D code produces Table 4.10  of iterations. Again as with the

1	Dofs $\mathbf{b}_h/r_h$	Number of iterations			
		$\nu_m = 10$	$\nu_m = 100$	$\nu_m = 1000$	$\nu_m = 10000$
2	2,936/729	4	4	6	5
3	21,424/49,13	4	4	6	5
4	163,424/35,937	4	6	6	5
5	1,276,096/274,625	4	6	6	5

Table 4.10: Iteration  table for Maxwell preconditioner for 3D example - direct application of preconditioner

2D results, the number of MINRES iterations stays roughly constant as the mesh level,  $l$ , increases. We also note that the number of iterations seems to be independent of the magnetic viscosity,  $\nu_m$ .

## 4.5 Numerical results: MHD problem

Sections 4.3 and 4.4  shows results for the Navier-Stokes and  Maxwells equations in isolation. The next step is to incorporate these two subproblems into the full MHD system.

### 4.5.1 Convergence results for a 2D smooth solution

To validate the code, we consider the following 2D test problem. For this problem, let the domain be the unit square  $\Omega = (0, 1)^2$  with purely Dirichlet

boundary conditions on  $\partial\Omega$ . Let  $\nu = \kappa = 1$ ,  $\nu_m = 1e4$  then the source terms  $\mathbf{f}$ ,  $\mathbf{g}$  and inhomogeneous Dirichlet boundary conditions are defined from the analytical solution:

$$\begin{aligned}\mathbf{u}(x, y) &= \begin{pmatrix} xy \exp(x + y) + x \exp(x + y) \\ -xy \exp(x + y) - y \exp(x + y) \end{pmatrix}, \\ p(x, y) &= \exp(y) \sin(x), \\ \mathbf{b}(x, y) &= \begin{pmatrix} \exp(x + y) \cos(x) \\ \exp(x + y) \sin(x) - \exp(x + y) \cos(x) \end{pmatrix}, \\ r(x, y) &= x \sin(2\pi x) \sin(2\pi y).\end{aligned}$$

The asymptotic rates of convergence are given in tables 4.11 to 4.13.

$l$	Dofs $\mathbf{u}_h/p_h$	$\ \mathbf{u} - \mathbf{u}_h\ _{L^2(\Omega)}$	order	$\ \mathbf{u} - \mathbf{u}_h\ _{H^1(\Omega)}$	order	$\ p - p_h\ _{L^2(\Omega)}$	order
1	50/9	8.1347e-02	0.00	1.2748e+00	0.00	2.8866e-01	0.00
2	162/25	1.0566e-02	2.94	3.3018e-01	1.95	3.7372e-02	2.95
3	578/81	1.3354e-03	2.98	8.3246e-02	1.99	4.8723e-03	2.94
4	2,178/289	1.6744e-04	3.00	2.0840e-02	2.00	7.2809e-04	2.74
5	8,450/1,089	2.0998e-05	3.00	5.2110e-03	2.00	1.5307e-04	2.25
6	33,282/4,225	2.6528e-06	2.98	1.3028e-03	2.00	3.7059e-05	2.05
7	132,098/16,641	3.4518e-07	2.94	3.2570e-04	2.00	9.2136e-06	2.01
8	526,338/66,049	4.9471e-08	2.80	8.1425e-05	2.00	2.3009e-06	2.00

Table 4.11: Convergence for 2D MHD smooth solution -  $tol = 1e-8$

The results shown in tables 4.11 to 4.13 agree with the optimal rates for  $\|\mathbf{u} - \mathbf{u}_h\|_{L^2(\Omega)}$ ,  $\|\mathbf{u} - \mathbf{u}_h\|_{H^1(\Omega)}$ ,  $\|\mathbf{b} - \mathbf{b}_h\|_{L^2(\Omega)}$ ,  $\|\mathbf{b} - \mathbf{b}_h\|_{H(curl, \Omega)}$ ,  $\|r - r_h\|_{L^2(\Omega)}$  and  $\|r - r_h\|_{H^1(\Omega)}$ . The pressure field exhibits a little higher than expected rate of convergence for  $l < 5$  but settles down to second order for the higher levels.

l	Dofs $\mathbf{b}_h/r_h$	$\ \mathbf{b} - \mathbf{b}_h\ _{L^2(\Omega)}$	order	$\ \mathbf{b} - \mathbf{b}_h\ _{H(curl,\Omega)}$	order
1	48/25	1.1206e-01	0.00	1.7807e-01	0.00
2	176/81	2.8439e-02	1.98	4.5429e-02	1.97
3	672/289	7.1388e-03	1.99	1.1414e-02	1.99
4	2,624/1,089	1.7867e-03	2.00	2.8571e-03	2.00
5	103,68/4,225	4.4679e-04	2.00	7.1450e-04	2.00
6	41,216/16,641	1.1171e-04	2.00	1.7864e-04	2.00
7	164,352/66,049	2.7927e-05	2.00	4.4661e-05	2.00
8	656,384/263,169	6.9818e-06	2.00	1.1165e-05	2.00

Table 4.12: Convergence for 2D MHD smooth solution - Magnetic field

l	Dofs $\mathbf{b}_h/r_h$	$\ r - r_h\ _{L^2(\Omega)}$	order	$\ r - r_h\ _{H^1(\Omega)}$	order
1	48/25	1.8438e-01	0.00	1.9449e+00	0.00
2	176/81	2.8307e-02	2.70	6.0694e-01	1.68
3	672/289	3.1214e-03	3.18	1.6510e-01	1.88
4	2,624/1,089	3.6035e-04	3.11	4.2417e-02	1.96
5	103,68/4,225	4.3890e-05	3.04	1.0683e-02	1.99
6	41,216/16,641	5.4485e-06	3.01	2.6757e-03	2.00
7	164,352/66,049	6.7988e-07	3.00	6.6923e-04	2.00
8	656,384/263,169	8.4948e-08	3.00	1.6733e-04	2.00

Table 4.13: Convergence for 2D MDH smooth solution - multiplier field

#### 4.5.2 Convergence results for a 3D smooth solution

As for the three-dimensional examples we consider a test problem discretised on a unit cube domain with Dirichlet boundary conditions. Let  $\nu = \kappa = 1$ ,



$\nu_m = 1e4$  and the analytical solution as:

$$\mathbf{u}(x, y, z) = \begin{pmatrix} -xy \exp(x + y + z) + xz \exp(x + y + z) \\ xy \exp(x + y + z) - yz \exp(x + y + z) \\ -xz \exp(x + y + z) + yz \exp(x + y + z) \end{pmatrix},$$

$$p(x, y, z) = \exp(x + y + z) \sin(y),$$

$$\mathbf{b}(x, y, z) = \begin{pmatrix} -\exp(x + y + z) \sin(y) + \exp(x + y + z) \sin(z) \\ xy \exp(x + y + z) - yz \exp(x + y + z) \\ -\exp(x + y + z) \sin(x) + \exp(x + y + z) \sin(y) \end{pmatrix},$$

$$r(x, y, z) = \sin(2\pi x) \sin(2\pi y) \sin(2\pi z),$$

then the source terms  $\mathbf{f}$  and  $\mathbf{g}$  are defined.

$l$	Dofs $\mathbf{u}_h/p_h$	$\ \mathbf{u} - \mathbf{u}_h\ _{L^2(\Omega)}$	order	$\ \mathbf{u} - \mathbf{u}_h\ _{H^1(\Omega)}$	order	$\ p - p_h\ _{L^2(\Omega)}$	order
1	375/27	4.2196e-03	-	7.8282e-02	-	5.0614e-02	-
2	2187/125	5.2733e-04	3.00	1.9495e-02	2.01	9.0306e-03	2.49
3	14739/729	6.5749e-05	3.00	4.8664e-03	2.00	1.9035e-03	2.25
4	107811/4913	8.2092e-06	3.00	1.2161e-03	2.00	4.5311e-04	2.07

Table 4.14: Convergence for 3D MHD smooth solution -  $tol = 1e-8$

1	Dofs $\mathbf{b}_h/r_h$	$\ \mathbf{b} - \mathbf{b}_h\ _{L^2(\Omega)}$	order	$\ \mathbf{b} - \mathbf{b}_h\ _{H(curl, \Omega)}$	order
1	436/125	3.9931e-03	-	2.4817e-02	-
2	2936/729	1.1211e-03	1.83	7.4101e-03	1.74
3	21424/4913	2.9642e-04	1.92	1.9439e-03	1.93
4	163424/35937	7.5494e-05	1.97	4.9180e-04	1.98

Table 4.15: Convergence for 3D MHD smooth solution - Magnetic field

From tables 4.14 to 4.16 it can be seen that we obtain the same convergence orders as with the 2D case.

l	Dofs $\mathbf{b}_h/r_h$	$\ r - r_h\ _{L^2(\Omega)}$	order	$\ r - r_h\ _{H^1(\Omega)}$	order
1	436/125	7.4750e-04	-	1.0113e-02	-
2	2936/729	9.2457e-05	3.02	2.9368e-03	1.78
3	21424/4913	1.1182e-05	3.05	7.7197e-04	1.93
4	163424/35937	1.3829e-06	3.02	1.9597e-04	1.98

Table 4.16: Convergence for 3D MDH smooth solution - multiplier field

### 4.5.3 Parameter tests

There are two different decoupling schemes considered in thesis. The effectiveness of these approaches is likely to be limited by the parameter set up of the problem, i.e. the fluid viscosity ( $\nu$ ), magnetic viscosity ( $\nu_m$ ) and the coupling number ( $\kappa$ ). In this section we will look at how these two schemes perform with respect to the full Picard iteration (P) when fixing two parameters and varying the second. We will only look at varying  $\kappa$  and  $\nu$  since  $\nu_m$  appears within all three schemes we will not consider varying this parameter. In the tables we denote (P) as the full MHD system given in (2.10), (MD) is the magnetic decoupling scheme (2.11) and (CD) is complete decoupling (2.12). Also, if the non-linear iterations do not converge then this is denoted by - in the table.



#### Viscosity test

The first test we will consider is to vary the fluid viscosity results are represented in Table 4.17. The numerical problem set up here is that the non-linear tolerance is  $tol = 1e-5$ , coupling number  $\kappa = 1$  and magnetic viscosity  $\nu_m = 10$ .



As the fluid viscosity ( $\nu$ ) decreases then the equations that determine

l	DoF	$\nu = 1$			$\nu = 0.1$			$\nu = 0.01$			$\nu = 0.001$		
		(P)	(MD)	(CD)	(P)	(MD)	(CD)	(P)	(MD)	(CD)	(P)	(MD)	(CD)
3	1,620	5	7	15	8	10	-	12	14	-	40	40	-
4	6,180	5	7	15	8	10	-	12	14	-	40	40	-
5	24,132	5	7	15	8	10	-	12	14	-	19	19	-
6	95,364	5	7	15	8	10	-	12	14	-	17	19	-
7	379,140	5	7	15	8	10	-	12	14	-	17	19	-
8	1,511,940	5	7	15	8	10	-	12	14	-	17	19	-

Table 4.17: Number of non-linear iterations for various values of  $\nu$  with  $tol = 1e-5$ ,  $\kappa = 1$  and  $\nu_m = 10$ .

the fluid flow become more **convect**  **dominated**. Thus we would expect that the complete decoupling scheme might break down for small  $\nu$  **as for** this decoupling scheme we remove the convection term. This is precisely **what we what we** see from the results in Table 4.17. **However**, the full MHD problem and magnetic decoupling cases perform very similarly in terms of the number of Picard iterations it takes to converge to the solution for  $\kappa = 1$ . For  $\nu = 0.0001$  all schemes break down and do not converge. This **maybe** due to the **stability** of the convection discretisation. 

### Coupling number test

The other parameter test that we **exam**  is to vary the coupling **ing**  term. This test shows how increasing the coupling term effects the number of non-linear iterations for each of the three schemes. We expect that the full Picard (P) scheme would perform best **with** both the magnetic and complete decoupling schemes **breaking** down as  $\kappa$  increases. Table 4.18 shows that this is what happens. Notice that the complete decoupling scheme completely breaks down for  $\kappa > 100$  whereas the number of non-linear iterations for magnetic

l	DoF	$\kappa = 0.1$			$\kappa = 1$			$\kappa = 10$			$\kappa = 100$			$\kappa = 1000$		
		(P)	(MD)	(CD)	(P)	(MD)	(CD)	(P)	(MD)	(CD)	(P)	(MD)	(CD)	(P)	(MD)	(CD)
3	1,620	5	5	13	5	5	13	6	7	17	7	13	26	7	93	-
4	6,180	5	5	13	5	5	15	6	7	18	7	13	26	7	93	-
5	24,132	5	5	13	5	5	15	6	7	18	7	13	26	7	93	-
6	95,364	5	5	14	5	5	15	6	7	18	7	13	26	7	93	-
7	379,140	5	5	14	5	5	15	6	7	18	7	13	26	7	93	-
8	1,511,940	5	5	14	5	5	15	6	7	18	7	13	26	7	93	-

Table 4.18: Number of non-linear iterations for various values of  $\kappa$  with  $tol = 1e-5$ ,  $\nu = 1$  and  $\nu_m = 10$ .

decoupling stays roughly constant for each mesh level but is larger than for the full Picard iterations.

## 4.6 Preconditioned MHD problem

Sections 3.3 introduced the three preconditioning techniques that will be applied to each of the decoupling schemes and the full MHD system. In this section we will look at the performances of these strategies with respect to time and iteration count for various tolerances.

In the tables we denote  $Its_{NL}$  as the number of non-linear iterations (used in (P), (MD) and (CD)),  $Its_O$  average number of outer GMRES iterations (used in (P)),  $Its_I$  average number of inner iterations (used in (P)),  $Its_{NS}$  average number of iterations to solve the Navier-Stokes problem (used in (MD)),  $Its_S$  average number of iterations to solve the Stokes problem (used in (CD)) and  $Its_M$  average number of iterations to solve the Maxwell problem (used in (MD) and (CD)). The numerical test presented in this section will be done in 2D unless specifically stated.

### 4.6.1 Picard Iteration

In this section, we will present preconditioning results for the full Picard iteration given in Section 3.3.1. Results will be shown both in two and three-dimensions. We start of with a parameter test similar to Section 4.5.3 then show a few results when changing the outer and inner Krylov solve tolerances.

#### Preconditioning parameter test

The first preconditioning parameter test we consider is to see how the fluids viscosity effects the preconditioner performance. For the preconditioner for the Picard iteration we imply the inner-outer preconditioner from Section 3.3.1. For the outer Krylov solver we use flexible GMRES [49] with a convergence tolerance of 1e-6 and the inner Krylov solver we use GMRES with a tolerance of 1e-6. The iteration results are presented in Table 4.19.

l	DoF	$\nu = 0.1$			$\nu = 1$			$\nu = 10$		
		Its <sub>NL</sub>	Its <sub>O</sub>	Its <sub>I</sub>	Its <sub>NL</sub>	Its <sub>O</sub>	Its <sub>I</sub>	Its <sub>NL</sub>	Its <sub>O</sub>	Its <sub>I</sub>
4	6,180	-	-	-	6	18.8	14.7	5	11.2	12.6
5	24,132	-	-	-	6	27.3	15.6	6	9.8	12.7
6	95,364	13	91.5	63.4	6	20.2	15.3	6	9.7	11.3
7	379,140	10	74.6	31.4	7	16.3	14.3	7	14.6	15.9
8	1,511,940	12	82.3	37.8	8	24.3	14.9	7	15.4	15.7

Table 4.19: Number of non-linear and preconditioning iterations for various values of  $\nu$  with  $tol = 1e-5$ ,  $\kappa = 1$  and  $\nu_m = 10$ .

From Table 4.19 we can see that for smaller  $\nu$  (the fluids viscosity) the preconditioner works harder (more iterations). This is to be expected as a small  $\nu$  corresponds to a convection dominated Navier-Stokes subproblem.

The table shows that as the mesh level increases the number of inner and outer iterations stays roughly constant. This is what we hoped to see. We also note that for levels 4 and 5 the iterations do not seem to converge for  $\nu = 0.1$ .

The second test we consider is to vary the coupling number. Again, we use the outer convergence tolerance of  $1e-6$  and inner tolerance  $1e-6$  with the same Krylov solvers. The iteration results are shown in Table 4.20 below.

l	DoF	$\kappa = 0.1$			$\kappa = 1$			$\kappa = 10$			$\kappa = 100$		
		Its <sub>NL</sub>	Its <sub>O</sub>	Its <sub>I</sub>	Its <sub>NL</sub>	Its <sub>O</sub>	Its <sub>I</sub>	Its <sub>NL</sub>	Its <sub>O</sub>	Its <sub>I</sub>	Its <sub>NL</sub>	Its <sub>O</sub>	Its <sub>I</sub>
4	6,180	5	22.2	15.4	6	18.8	14.7	7	30.7	14.1	9	61.4	24.4
5	24,132	5	23.8	11.4	6	27.3	15.2	8	43.8	24.0	10	80.3	37.9
6	95,364	5	28.0	17.4	6	20.2	15.3	7	41.1	15.4	9	74.6	31.1
7	379,140	5	18.6	14.6	7	16.3	14.2	7	37.4	16.4	14	73.9	34.7
8	1,511,940	5	20.4	15.2	8	24.3	14.9	7	39.6	18.4	11	75.4	33.3

Table 4.20: Number of non-linear and preconditioning iterations for various values of  $\kappa$  with  $tol = 1e-5$ ,  $\nu = 1$  and  $\nu_m = 10$ .

Table 4.20 shows that as the mesh level increases then the number of outer and inner GMRES iterations stays roughly constant. We note that for larger  $\kappa$  the number of outer iterations increases. We also see the trend with the inner iterations but only really between  $\kappa = 10$  and  $100$ .

### Varying solver tolerances

We will now present some results for different solver tolerances. Here we will fix the tolerance for the outer FGMRES solver as  $1e-6$  and vary the inner tolerance. The results are presented in Table 4.21.

From Table 4.21 we can see that as we decrease the inner tolerance of the GMRES iteration then the number of average inner iterations decreases.

l	DoF	Inner tol 1e-6			Inner tol 1e-4			Inner tol 1e-2		
		Its <sub>NL</sub>	Its <sub>O</sub>	Its <sub>I</sub>	Its <sub>NL</sub>	Its <sub>O</sub>	Its <sub>I</sub>	Its <sub>NL</sub>	Its <sub>O</sub>	Its <sub>I</sub>
3	1,620	6	22.2	20.2	6	18.3	9.3	6	45.1	19.6
4	6,180	6	18.8	14.7	6	19.7	9.8	6	30.6	2.7
5	24,132	6	27.3	15.2	6	22.7	10.8	6	33.7	3.8
6	95,364	6	20.2	15.3	6	21.7	12.5	6	46.8	4.3
7	379,140	7	16.3	14.3	7	17.1	10.3	6	28.6	3.3
8	1511940	8	24.3	14.8	7	19.6	10.1	6	26.0	3.2

Table 4.21: Number of non-linear and preconditioning iterations for various inner tolerances with outer tolerance 1e-6 with  $tol = 1e-5$ ,  $\nu = 1$ ,  $\kappa = 1$  and  $\nu_m = 10$ .



This is to be expected. We also note that setting the inner tolerance to 1e-4 gives roughly the same average outer iterations as with a tolerance of 1e-6. When setting the inner tolerance to 1e-2 we see a slight increase in the average outer iterations but we still see scalability with respect to the mesh level.

### Preconditioned 3D results

#### 4.6.2 Magnetic Decoupling

This section will look at preconditioning results for the Magnetic Decoupling scheme outlined in Section 3.3.2. As with the Picard iterations results we will consider a preconditioner parameter test to test the robustness of this decoupling scheme. All tests considered here will have a GMRES stopping tolerance of 1e-6 for the Navier-Stokes subproblem and a MINRES stopping tolerance of 1e-6 for the Maxwell subproblem.

### Preconditioning parameter test

We will again start of by seeing how the fluids viscosity effects the preconditioner. The results are shown in Table 4.22.

l	DoF	$\nu = 0.01$			$\nu = 0.1$			$\nu = 1$			$\nu = 10$		
		Its <sub>NL</sub>	Its <sub>NS</sub>	Its <sub>M</sub>	Its <sub>NL</sub>	Its <sub>NS</sub>	Its <sub>M</sub>	Its <sub>NL</sub>	Its <sub>NS</sub>	Its <sub>M</sub>	Its <sub>NL</sub>	Its <sub>NS</sub>	Its <sub>M</sub>
5	24,132	13	360.3	3.3	8	24.1	3.3	5	22.0	3.4	4	22.0	3.5
6	95,364	13	105.2	3.3	8	25.1	3.4	5	21.2	3.4	4	20.8	3.5
7	379,140	13	71.5	3.4	8	25.3	3.3	5	21.2	3.4	4	21.0	3.5
8	1,511,940	13	65.3	3.4	8	25.9	3.4	5	21.4	3.4	4	21.3	3.5

Table 4.22: **Table** number of non-linear iterations and number of iterations to solve the Navier-Stokes and Maxwells subproblem for the MD scheme with  $tol = 1e-4$ ,  $\kappa = 1$ ,  $\nu = 1$  and  $\nu_m = 10$ .

From Table 4.22 we can see that for  $\nu = 0.01$  the non-linear iteration scheme converges slowly to the solution for the lower mesh levels ( $l \leq 6$ ). A possible reason for this is that the PCD preconditioner for the Navier-Stokes equations does not always converge for small viscosities and small meshes. However, we see that for  $\nu \geq 0.1$  both the Navier-Stokes and **Maxwells** subproblems exhibit iterations independent of the mesh level.

l	DoF	$\kappa = 0.1$			$\kappa = 1$			$\kappa = 10$			$\kappa = 100$		
		Its <sub>NL</sub>	Its <sub>NS</sub>	Its <sub>M</sub>	Its <sub>NL</sub>	Its <sub>NS</sub>	Its <sub>M</sub>	Its <sub>NL</sub>	Its <sub>NS</sub>	Its <sub>M</sub>	Its <sub>NL</sub>	Its <sub>NS</sub>	Its <sub>M</sub>
5	24,132	4	22.1	4.5	5	22.0	3.4	10	21.4	2.3	-	21.7	2.2
6	95,364	4	21.5	4.5	5	21.2	3.4	10	21.1	2.3	-	21.5	2.3
7	379,140	4	21.5	4.8	5	21.2	3.4	10	21.1	2.4	-	21.6	2.2
8	1,511,940	4	21.5	4.8	5	21.4	3.4	10	21.1	2.4	-	21.7	2.2

Table 4.23: **Table** number of non-linear iterations and number of iterations to solve the Navier-Stokes and Maxwells subproblem for the MD scheme with  $tol = 1e-4$   $\nu = 1$  and  $\nu_m = 10$ .

Table 4.23 shows the **iterations results of** the Magnetic Decoupling



scheme when the coupling number,  $\kappa$ , is varied. One can see that as  $\kappa$  increases the number of non-linear iterations increases. In fact, for  $\kappa = 100$  the scheme does not converge. If we recall Table 4.18 we note that for direct solves  $\kappa = 100$  does converge to the solution. Therefore, it seems that the Krylov tolerance of  $1e-6$  is not tight enough for (MD) to converge. The tables also shows that Navier-Stokes and Maxwells subproblems exhibit iterations independent of the mesh level and coupling number.

1	DoF	AV solve Time	Total picard time	Its <sub>NL</sub>	Its <sub>NS</sub>	Its <sub>M</sub>
5	24,132	0.5	7.3	5	22.0	3.4
6	95,364	2.5	30.4	5	21.2	3.4
7	379,140	13.1	134.7	5	21.2	3.4
8	1,511,940	69.8	627.3	5	21.4	3.4
9	6,038,532	407.7	3159.7	5	21.6	3.2
10	24,135,684	3022.1	19668.3	5	21.6	3.4

Table 4.24: Table number of non-linear iterations and number of iterations to solve the Navier-Stokes and Maxwells subproblem for the MD scheme with  $tol = 1e-4$ ,  $\kappa = 1$ ,  $\nu = 1$  and  $\nu_m = 10$ .

### Preconditioned 3D results

DoF	AV solve Time	Total picard time		Its <sub>NL</sub>	Its <sub>NS</sub>	Its <sub>M</sub>
1	963	0.04	2.7	5	23.2	3.4
2	5977	0.20	17.2	5	34.2	3.0
3	41805	4.86	151.1	5	34.2	3.4
4	312,085	242.1	2222.8	5	32.4	3.4

### 4.6.3 Complete Decoupling

Finally, we consider preconditioning the Complete Decoupling scheme presented in Section 3.3.3. As the (P) and (MD) we will consider varying both the fluids viscosity as well as the coupling number. To solve the Stokes block of (3.14) we will use MINRES, as Stokes equations are symmetric. Results are shown 1e-6 as the convergence tolerance for both the Stokes and Maxwell solvers.

#### Preconditioning parameter test

l	DoF	$\nu = 1$			$\nu = 10$		
		Its <sub>NL</sub>	Its <sub>S</sub>	Its <sub>M</sub>	Its <sub>NL</sub>	Its <sub>S</sub>	Its <sub>M</sub>
5	24,132	11	29.0	3.3	4	29.0	3.5
6	95,364	11	28.5	3.4	4	29.0	3.5
7	379,140	11	27.5	3.4	4	29.0	3.5
8	1,511,940	11	28.3	3.5	4	29.0	3.5

Table 4.25: Table number of non-linear iterations and number of iterations to solve the Stokes and Maxwells subproblem for the CD scheme with  $tol = 1e-4$   $\kappa = 1$ , and  $\nu_m = 10$ .

l	DoF	$\kappa = 0.1$			$\kappa = 1$			$\kappa = 10$			$\kappa = 100$		
		Its <sub>NL</sub>	Its <sub>S</sub>	Its <sub>M</sub>	Its <sub>NL</sub>	Its <sub>S</sub>	Its <sub>M</sub>	Its <sub>NL</sub>	Its <sub>S</sub>	Its <sub>M</sub>	Its <sub>NL</sub>	Its <sub>S</sub>	Its <sub>M</sub>
5	24,132	9	29.0	4.4	11	29.0	3.3	18	29.0	2.2	-	-	-
6	95,364	9	29.0	4.6	11	28.5	3.4	18	29.0	2.3	-	-	-
7	379,140	8	28.5	4.5	11	27.6	3.4	18	29.0	2.3	-	-	-
8	1,511,940	8	28.3	4.6	11	28.3	3.5	18	29.0	2.4	-	-	-

Table 4.26: Table number of non-linear iterations and number of iterations to solve the Stokes and Maxwells subproblem for the CD scheme with  $tol = 1e-4$   $\nu = 1$  and  $\nu_m = 10$ .

l	DoF	AV solve Time	Total picard time	Its <sub>NL</sub>	Its <sub>S</sub>	Its <sub>M</sub>
5	24,132	0.4	7.7	11	29.0	3.3
6	95,364	2.6	38.8	11	28.5	3.4
7	379,140	13.0	181.7	11	27.5	3.4
8	1,511,940	66.5	888.0	11	28.3	3.5
9	6,038,532	358.0	4565.4	11	28.3	3.4

Table 4.27: Table number of non-linear iterations and number of iterations to solve the Stokes and Maxwells subproblem for the CD scheme with  $tol = 1e-4$   $\kappa = 1$ ,  $\nu = 1$  and  $\nu_m = 100$ .

### Preconditioned 3D results

l	DoF	AV solve Time	Total picard time	Its <sub>NL</sub>	Its <sub>S</sub>	Its <sub>M</sub>
1	963	0.03	1.8	6	30.0	3.5
2	5,977	0.20	9.9	6	45.7	3.2
3	41,805	4.32	89.2	6	43.0	2.8
4	312,085	214.3	1786.5	6	42.3	2.8
5	2,410,533	26954.4	165671.1	6	41.3	2.8

## Chapter 5

# Conclusion and Future work

### 5.1 Conclusion

### 5.2 Future work

- Scalable inner solvers:
- Parallelisation of the code:
- Test other non-linear solvers:

# Bibliography

- [1] Martin S. Alnæs. *UFL: a Finite Element Form Language*, chapter 17. Springer, 2012.
- [2] Martin S. Alnæs, Anders Logg, and Kent-Andre Mardal. *UFC: a Finite Element Code Generation Interface*, chapter 16. Springer, 2012.
- [3] Martin S. Alnæs, Anders Logg, Kent-Andre Mardal, Ola Skavhaug, and Hans Petter Langtangen. Unified Framework for Finite Element Assembly. *International Journal of Computational Science and Engineering*, 4(4):231–244, 2009.
- [4] Martin S. Alnæs, Anders Logg, Kristian B. Ølgaard, Marie E. Rognes, and Garth N. Wells. Unified Form Language: A domain-specific language for weak formulations of partial differential equations. *ACM Transactions on Mathematical Software*, To appear, 2013.
- [5] Patrick R Amestoy, Iain S Duff, and J-Y L’Excellent. Multifrontal parallel distributed symmetric and unsymmetric solvers. *Computer methods in applied mechanics and engineering*, 184(2):501–520, 2000.
- [6] Patrick R Amestoy, Iain S Duff, Jean-Yves L’Excellent, and Jacko Koster. A fully asynchronous multifrontal solver using distributed dy-

- dynamic scheduling. *SIAM Journal on Matrix Analysis and Applications*, 23(1):15–41, 2001.
- [7] Patrick R Amestoy, Abdou Guermouche, Jean-Yves LExcellent, and Stéphane Pralet. Hybrid scheduling for the parallel solution of linear systems. *Parallel computing*, 32(2):136–156, 2006.
- [8] F Armero and JC 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):41–90, 1996.
- [9] Satish BalaB, Mark F. Adams, Jed Brown, Peter Brune, Kris Buschelman, Victor Eijkhout, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Lois Curfman McInnes, Karl Rupp, Barry F. Smith, and Hong Zhang. PETSc Web page. <http://www.mcs.anl.gov/petsc>, 2014.
- [10] Satish Balay, Mark F. Adams, Jed Brown, Peter Brune, Kris Buschelman, Victor Eijkhout, William D. Gropp, Dinesh Kaushik, Matthew G. Knepley, Lois Curfman McInnes, Karl Rupp, Barry F. Smith, and Hong Zhang. PETSc Users Manual. Technical Report ANL-95/11 - Revision 3.4, Argonne National Laboratory, 2013.
- [11] Ramon Codina and Noel Hernández-Silva. Stabilized finite element approximation of the stationary magneto-hydrodynamics equations. *Computational Mechanics*, 38(4-5):344–355, 2006.
- [12] Martin Costabel and Monique Dauge. Singularities of Electromagnetic

- Fields¶ in Polyhedral Domains. *Archive for Rational Mechanics and Analysis*, 151(3):221–276, 2000.
- [13] P. A. Davidson. *An introduction to magnetohydrodynamics*. Cambridge Texts in Applied Mathematics. Cambridge University Press, Cambridge, 2001.
- [14] Timothy A. Davis. A Column Pre-ordering Strategy for the Unsymmetric-pattern Multifrontal Method. *ACM Trans. Math. Softw.*, 30(2):165–195, June 2004.
- [15] Timothy A. Davis. Algorithm 832: UMFPACK V4.3—an Unsymmetric-pattern Multifrontal Method. *ACM Trans. Math. Softw.*, 30(2):196–199, June 2004.
- [16] Timothy A Davis and Iain S Duff. An unsymmetric-pattern multifrontal method for sparse LU factorization. *SIAM Journal on Matrix Analysis and Applications*, 18(1):140–158, 1997.
- [17] Timothy A. Davis and Iain S. Duff. A Combined Unifrontal/Multifrontal Method for Unsymmetric Sparse Matrices. *ACM Trans. Math. Softw.*, 25(1):1–20, March 1999.
- [18] Leszek Demkowicz and L Vardapetyan. Modeling of electromagnetic absorption/scattering problems using  $h_p/h_h$ -adaptive finite elements. *Computer Methods in Applied Mechanics and Engineering*, 152(1):103–124, 1998.
- [19] Howard C Elman, David J Silvester, and Andrew J Wathen. *Finite El-*

*ements and Fast Iterative Solvers: with Applications in Incompressible Fluid Dynamics: with Applications in Incompressible Fluid Dynamics.* Oxford University Press, 2005.

- [20] RobertD. Falgout and UlrikeMeier Yang. hypre: A library of high performance preconditioners. In PeterM.A. Sloot, AlfonsG. Hoekstra, C.J.Kenneth Tan, and JackJ. Dongarra, editors, *Computational Science ICCS 2002*, volume 2331 of *Lecture Notes in Computer Science*, pages 632–641. Springer Berlin Heidelberg, 2002.
- [21] J-F Gerbeau. A stabilized finite element method for the incompressible magnetohydrodynamic equations. *Numerische Mathematik*, 87(1):83–111, 2000.
- [22] Gene H Golub and Chen Greif. On solving block-structured indefinite linear systems. *SIAM Journal on Scientific Computing*, 24(6):2076–2092, 2003.
- [23] Chen Greif and D Schötzau. Preconditioners for saddle point linear systems with highly singular  $(1, 1)$  blocks. *ETNA, Special Volume on Saddle Point Problems*, 22:114–121, 2006.
- [24] Chen Greif and Dominik Schötzau. Preconditioners for the discretized time-harmonic Maxwell equations in mixed form. *Numerical Linear Algebra with Applications*, 14(4):281–297, 2007.
- [25] J-L Guermond, Raphael Laguerre, Jacques L  orat, and Caroline Nore. Nonlinear magnetohydrodynamics in axisymmetric heterogeneous do-



- mains using a Fourier/finite element technique and an interior penalty method. *Journal of Computational Physics*, 228(8):2739–2757, 2009.
- [26] JL Guermond and PD Mineev. Mixed finite element approximation of an MHD problem involving conducting and insulating regions: the 3D case. *Numerical Methods for Partial Differential Equations*, 19(6):709–731, 2003.
- [27] Max D Gunzburger, Amnon J Meir, and Janet S Peterson. On the existence, uniqueness, and finite element approximation of solutions of the equations of stationary, incompressible magnetohydrodynamics. *Mathematics of Computation*, 56(194):523–563, 1991.
- [28] Pascal Hénon, Pierre Ramet, and Jean Roman. Pastix: a high-performance parallel direct solver for sparse symmetric positive definite systems. *Parallel Computing*, 28(2):301–321, 2002.
- [29] Magnus Rudolph Hestenes and Eduard Stiefel. *Methods of conjugate gradients for solving linear systems*, volume 49. NBS, 1952.
- [30] Ralf Hiptmair and Jinchao Xu. Nodal auxiliary space preconditioning in  $H(\text{curl})$  and  $H(\text{div})$  spaces. *SIAM Journal on Numerical Analysis*, 45(6):2483–2509, 2007.
- [31] Robert C. Kirby. Algorithm 839: FIAT, a New Paradigm for Computing Finite Element Basis Functions. *ACM Transactions on Mathematical Software*, 30(4):502–516, 2004.

- [32] Robert C. Kirby. *FIAT: Numerical Construction of Finite Element Basis Functions*, chapter 13. Springer, 2012.
- [33] Robert C. Kirby and Anders Logg. A Compiler for Variational Forms. *ACM Transactions on Mathematical Software*, 32(3), 2006.
- [34] Claude Le Bris, Tony Lelièvre, et al. *Mathematical methods for the magnetohydrodynamics of liquid metals*. Oxford University Press, 2006.
- [35] Dan Li. Numerical solution of the time-harmonic Maxwell equations and incompressible magnetohydrodynamics problems. *University of British Columbia*, 2010.
- [36] Xiaoye S. Li. An overview of SuperLU: Algorithms, implementation, and user interface. *ACM Transactions on Mathematical Software*, 31(3):302–325, September 2005.
- [37] X.S. Li, J.W. Demmel, J.R. Gilbert, iL. Grigori, M. Shao, and I. Yamazaki. SuperLU Users’ Guide. Technical Report LBNL-44289, Lawrence Berkeley National Laboratory, September 1999. <http://crd.lbl.gov/~xiaoye/SuperLU/>. Last update: August 2011.
- [38] Anders Logg, Kristian B. Ølgaard, Marie E. Rognes, and Garth N. Wells. *FFC: the FEniCS Form Compiler*, chapter 11. Springer, 2012.
- [39] Anders Logg and Garth N. Wells. DOLFIN: Automated Finite Element Computing. *ACM Transactions on Mathematical Software*, 37(2), 2010.
- [40] Anders Logg, Garth N. Wells, and Johan Hake. *DOLFIN: a C++/Python Finite Element Library*, chapter 10. Springer, 2012.

- [41] Peter Monk. *Finite element methods for Maxwell's equations*. Oxford University Press, 2003.
- [42] Ulrich Müller and Leo Bühler. *Magnetofluidynamics in channels and containers*. Springer, 2001.
- [43] Malcolm F Murphy, Gene H Golub, and Andrew J Wathen. A note on preconditioning for indefinite linear systems. *SIAM Journal on Scientific Computing*, 21(6):1969–1972, 2000.
- [44] Jean-Claude Nédélec. Mixed finite elements in 3. *Numerische Mathematik*, 35(3):315–341, 1980.
- [45] Kristian B. Ølgaard and Garth N. Wells. Optimisations for Quadrature Representations of Finite Element Tensors Through Automated Code Generation. *ACM Transactions on Mathematical Software*, 37, 2010.
- [46] Christopher C Paige and Michael A Saunders. Solution of sparse indefinite systems of linear equations. *SIAM Journal on Numerical Analysis*, 12(4):617–629, 1975.
- [47] Edward G Phillips, Howard C Elman, Eric C Cyr, John N Shadid, and Roger P Pawlowski. A Block Preconditioner for an Exact Penalty Formulation for Stationary MHD. 2014.
- [48] Paul Harry Roberts. An introduction to magnetohydrodynamics. *An Introduction to Magnetohydrodynamics, by PH Roberts. Textbook published by Longmans, Green and Co ltd, London, 1967, 1, 1967.*

- [49] Youcef Saad. A flexible inner-outer preconditioned GMRES algorithm. *SIAM Journal on Scientific Computing*, 14(2):461–469, 1993.
- [50] Youcef Saad and Martin H Schultz. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, 7(3):856–869, 1986.
- [51] Yousef Saad. *Iterative methods for sparse linear systems*. Siam, 2003.
- [52] Dominik Schötzau. Mixed finite element methods for stationary incompressible magneto–hydrodynamics. *Numerische Mathematik*, 96(4):771–800, 2004.
- [53] David Silvester and Andrew Wathen. Fast iterative solution of stabilised Stokes systems. Part I: Using simple diagonal preconditioners. *SIAM Journal on Numerical Analysis*, 30(3):630–649, 1993.
- [54] David Silvester and Andrew Wathen. Fast iterative solution of stabilised Stokes systems part II: using general block preconditioners. *SIAM Journal on Numerical Analysis*, 31(5):1352–1367, 1994.
- [55] C Taylor and P Hood. A numerical solution of the Navier-Stokes equations using the finite element technique. *Computers & Fluids*, 1(1):73–100, 1973.
- [56] Garth Wells, Kent-Andre Mardal, and Anders Logg. *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book*. Springer, 2012.