APPLICATION OF DOMAIN DECOMPOSITION METHODS TO PROBLEMS IN TOPOLOGY OPTIMISATION

by

JAMES TURNER

A thesis submitted to
The University of Birmingham
for the degree of
DOCTOR OF PHILOSOPHY

School of Mathematics
College of Engineering and Physical Sciences
The University of Birmingham
October 2014

Abstract

Determination of the optimal layout of structures can be seen in everyday life, from nature to industry, with research dating back to the eighteenth century. The focus of this thesis involves investigation into the relatively modern field of topology optimisation, where the aim is to determine both the optimal shape and topology of structures. However, the inherent large-scale nature means that even problems defined using a relatively coarse finite element discretisation can be computationally demanding.

This thesis aims to describe alternative approaches allowing for the practical use of topology optimisation on a large scale. Commonly used solution methods will be compared and scrutinised, with observations used in the application of a novel substructuring domain decomposition method for the subsequent large-scale linear systems. Numerical and analytical investigations involving the governing equations of linear elasticity will lead to the development of three different algorithms for compliance minimisation problems in topology optimisation. Each algorithm will involve an appropriate preconditioning strategy incorporating a matrix representation of a discrete interpolation norm, with numerical results indicating mesh independent performance.

ACKNOWLEDGEMENTS

First and foremost I would like to thank both of my supervisors Professor Michal Kočvara and Doctor Daniel Loghin for their insight and guidance over the last four years. Without their advice, along with many other teachers who have helped me throughout my academic career, I certainly would not be in the position I am today.

I would also like to thank my family for their seemingly never-ending patience and support during both my undergraduate and postgraduate studies, as well as friends in and around the department. Particular thanks go to Azizah Alrashidi, Nina Embleton, Rodrigue Kammogne, Sudaba Mohammed, Thomas Reeve and Samia Riaz, along with everyone I have collaborated with in the department over the past eight years.

I am grateful to ANSYS for kind permission to reproduce a pictorial example within the conclusion of this thesis.

Finally, I wish to thank the School of Mathematics for the financial support and resources to be able to complete this degree. I also acknowledge conference and workshop related funding from not only the school (and Prof. Kočvara), but also from the Technical University of Denmark, the Society for Industrial and Applied Mathematics and also the Institute of Mathematics and its Applications. This support has been vital in allowing me to meet other researchers within my field, and also deliver work within this thesis to an international audience.

LITERARY CONTRIBUTIONS

Contributions in the last two chapters of this thesis have been presented within the following two papers, based on collaborations with both Professor Michal Kočvara and Doctor Daniel Loghin at the University of Birmingham.

- M. Kočvara, D. Loghin, and J. Turner. Constraint Interface Preconditioning for Topology Optimization Problems. *To appear in SIAM J. Sci. Statist. Comput. (SISC): Copper Mountain Special Section 2014*, Colorado, USA; 6-11 April 2014.

- J. Turner, M. Kočvara, and D. Loghin. Parallel Solution of the Linear Elasticity Problem with Applications in Topology Optimization. In *Proceedings of the 4th Annual BEAR PGR Conference, University of Birmingham, United Kingdom*, 2013.

Additional work beyond the scope of this thesis (co-authored as above) has also been published in

- J. Turner, M. Kočvara, and D. Loghin. A nonlinear domain decomposition technique for scalar elliptic PDEs. In *Proceedings of the 21st International Conference on Domain Decomposition Methods, INRIA Rennes-Bretagne-Atlantique, France*, 2013.

CONTENTS

LIST OF ALGORITHMS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

Symbol	**Description**
CG	Conjugate Gradient
FEM	Finite Element Mesh
FETI	Finite Element Tearing and Interconnect
FETI-DP	Finite Element Tearing and Interconnect Dual-Primal
FOM	Full Orthogonalization Method
FSAI	Factorised Sparse Approximate Inverse
GMRES	Generalised Minimum Residual
GPU	Graphics Processing Unit
ILU	Incomplete Lower-Upper
IP	Interior Point
KKT	Karush-Kuhn-Tucker
LICQ	Linear Independence Constraint Qualification
MBB	Messerschmitt-Bölkow-Blohm
MEMS	Micro Electro Mechanical Systems
MINRES	Minimum Residual
MMA	Method of Moving Asymptotes
OC	Optimality Criteria
ORTHORES	Orthogonal Residual
PCG	Preconditioned Conjugate Gradient
PDE	Partial Differential Equation
SIMP	Solid Isotropic Material with Penalisation

SLP	Sequential Linear Programming
VTS	Variable Thickness Sheet

CHAPTER 1

INTRODUCTION AND PRELIMINARIES

1.1 Introduction

The desire to determine the optimal layout of structures can be seen in works dating back to the eighteenth century, with one dimensional problems considered by both Euler [53] and Lagrange [101]. The focus of this thesis is on a relatively modern branch of structural optimisation termed topology optimisation which, for a given structure, aims to determine both the optimal shape and topology. It has been used to produce conceptual designs for a wide range of engineering applications, having been used in the design of materials [22, 107, 173], micro electro mechanical systems (MEMS) [119, 169], mechanisms [30, 168, 205] and other complex structural design problems.

Given an amount of material, an external load and boundary conditions (as well as potentially other support conditions), the essential aim of topology optimisation is to determine the optimal distribution of material subject to an appropriately defined objective function. In the design of structures, relevant criteria such as weight, stiffness, compliance, displacement and stress (amongst others) can all be involved within the definition of the objective function as well as other associated constraints, dependent on the task at hand.

The need for efficient use of material is prevalent within a number of scientific and engineering applications. For instance, within aircraft design one typically aims to produce cost effective components delivering savings in weight and improvements in efficiency without compromising overall performance. Such components include fuselage door stops and intercostals, as well as wing edge and box ribs and also types of wing trailing edge brackets. As such, a number of innovative designs have been obtained through the use of topology optimisation, and utilised (for instance) within the aerospace [63, 96, 97, 98, 157, 165] and automotive [35, 36, 40, 145] industries.

Unfortunately, an ever-present issue with topology optimisation is that a large number of design variables are required in the discrete formulation in order to maintain the quality of the contours in the final design. Solutions to the resulting nonlinear optimisation problem are typically obtained through the use of iterative optimisation techniques, requiring the solution to the governing equations of linear elasticity at each iterative step. As a result, even problems defined using a relatively coarse finite element discretisation can be computationally demanding. The ultimate aim of this thesis is to describe alternative approaches whereby topology optimization on a large scale becomes practical in a computational sense.

Attempts to alleviate such difficulties can involve the application of a faster finite element solver, or the use of efficient discretisation techniques [43]. Standard approaches based around fixed point iterations target the ill-conditioned equilibrium equations, in which the bulk of computational effort resides. In [197], MINRES coupled with recycling is explored based on the observation that the densities are only expected to undergo minor changes after a relatively small number of iterative steps. The ill-conditioning is dealt with through a preconditioning strategy involving both rescaling and an incomplete Cholesky decomposition.

This thesis will explore the application of domain decomposition to compliance minimisation problems in topology optimisation. Initially, this will involve a description of a parallel framework for the equations of linear elasticity, where solutions are sought through use of preconditioned GMRES. The preconditioner considered is based on a matrix representation of an appropriate fractional Sobolev norm (initially explored by Dryja

[47] and also Bramble et al. [26]), with analytical and also numerical assertions of results without dependence on the chosen mesh parameter. These findings are then used in the development of an appropriate preconditioning strategy for fixed point solution methods for topology optimisation.

The use of primal-dual Interior Point (IP) methods will also be considered within this thesis. Examples illustrating the application of such approaches for solving large scale topology optimization problems can be found in [12, 76, 114]. Here, a reformulation of the problem is considered in which the box constraints form part of the objective function through the use of logarithmic barrier terms. The KKT conditions from the resulting nonlinear equality constrained optimization problem are then solved using Newton's method. Evidently, for large scale problems, obtaining solutions to the resulting system of equations will become expensive and even prohibitive in certain cases. In [114], Maar and Schultz applied multigrid to the resulting system, and from their results were able to witness an approximately linear overall complexity with respect to the number of unknowns used in the problem for the presented solution method.

We instead consider use of domain decomposition within the resulting Newton system, with preconditioning approaches described in two cases. Initial investigation will involve reduction of the system through use of an appropriate Schur complement, with the resulting matrix vector system solved using GMRES coupled with preconditioning through consideration of a matrix representation of an appropriate fractional Sobolev norm. However, analysis of the Schur complement suggests the need to either enhance the preconditioner, or return to the unreduced system. Exploring the latter option leads to an expanded formulation, from which local Jacobian matrices related to topology optimisation problems posed on subdomains arise. Preconditioning in a similar manner to the reduced case leads to mesh independent results, with improvements noted when compared directly to the solution methods described for fixed point approaches and also for the reduced interior point formulation.

This thesis has been structured in a sequential manner in that it is designed to be read from the first chapter directly to the last. However, in order to provide a comprehensive description of the problem at hand, a fair amount of background material within matrix and functional analysis, optimisation, linear elasticity, structural optimisation, numerical methods for linear systems and also domain decomposition is described. Therefore, a reader with thorough understanding of one or more of these fields may choose to bypass Chapters 2, 3, and 5 and still be able to follow the remainder of the thesis, aside from the literature review contained at the end of Chapter 5. In particular, the notation defined within both this chapter and Chapter 5 is designed to be self-contained. All numerical results presented within this thesis were obtained using a Linux machine with an Intel® Core$^{\text{TM}}$ i7 CPU 870 @ 2.93 GHz with 8 cores.

Following on from this introduction, a number of relevant background definitions and theorems in the fields of both matrix and functional analysis, and also optimisation will be presented and will be used and referred to throughout the thesis. A description of the format and layout used within the remainder of the thesis will now be presented.

Chapter 2 provides an introduction to linear elasticity, with the classical formulation described along with the associated weak, discrete weak and finite element formulations.

Chapter 3 begins with an introduction to structural optimisation, along with the three principal branches, namely sizing, shape and topology optimisation. A mathematical formulation for compliance minimisation is then provided for the latter, along with a description of the variable thickness sheet problem and also model problems used for numerical investigations in later chapters.

Chapter 4 essentially describes both fixed point and also primal-dual interior point methods applied to compliance minimisation problems. Fixed point approaches have seen widespread use, and details on two of the most common approaches will be provided, namely the Optimality Criteria (OC) method and also the Method of Moving Asymptotes (MMA), along with numerical comparisons for each approach.

Chapter 5 provides a background of typical solution methods that look to solve linear systems based on the findings in the previous chapter. Both direct and iterative approaches are described, followed by an in-depth discussion into a particular hybrid approach, namely domain decomposition. A historical background is provided, followed by a literature review detailing existing work in the field of topology optimisation. Based on these findings, we propose to develop a preconditioning strategy based on the application of domain decomposition to the governing equations of linear elasticity.

This is the main focus of Chapter 6, involving the derivation of an iterative solution method that is able to provide results independently of the chosen mesh parameter. An analytical proof of concept for this algorithm will be shown using [5] along with numerical validation. Additionally, calculations indicating projected speedup in a parallel environment will also be provided. These investigations are then used as a platform for the description of solution methods involving the application of domain decomposition within topology optimisation. Aspects of the work within this chapter were presented at the 2013 BlueBEAR conference at the University of Birmingham [194].

Chapter 7 describes three approaches in principle, along with associated numerical results. The first of these approaches is based on the fixed point type solution method, with results suggesting the need to control the number of fixed point iterations in order to provide a scalable algorithm. We also describe solution methods for the reduced primal-dual Newton system, followed by an expanded system in an attempt to provide a more effective solution method. Investigations and numerical results for the expanded system described within this chapter were also presented in [95].

Finally, we compare each of the presented approaches, before providing concluding remarks along with areas for future investigation in Chapter 8.

1.2 Background Material

We now introduce definitions and background theorems that will be used throughout this thesis. We begin by denoting the respective positive and non-negative orthants of a field $\mathbb{F}$ as $\mathbb{F}_+$ and $\mathbb{F}_{0_+}$, with results for this work typically focussed on the case where $\mathbb{F} = \mathbb{R}$.

However, certain results that will be clearly indicated will involve $\mathbb{F} \in \{\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{C}\}$.

1.2.1 Matrix Analysis

We begin this section by describing the notion of symmetric positive definiteness as follows.

Definition 1.2.1. *(Symmetric Positive (Semi-) Definiteness) A symmetric matrix $A \in \mathbb{R}^{n\times n}$ is said to be symmetric positive definite if for all nonzero $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{x}^T A\mathbf{x} > 0$. If $\mathbf{x}^T A\mathbf{x} \geq 0$ for all nonzero $\mathbf{x} \in \mathbb{R}^n$, A is said to be symmetric positive semi-definite. The notation $A \succ 0$ will be used to denote symmetric positive definiteness, and $A \succeq 0$ for symmetric positive semi-definiteness, with the set of all symmetric positive definite $n \times n$ matrices denoted by $\mathbb{S}^n$.*

There are a number of conditions that effectively characterise symmetric positive definite matrices. In particular, Horn and Johnson [79] describe a number of related theorems and results. For this thesis, the following properties are of particular note

Lemma 1.2.1. *The following properties are equivalent to a symmetric matrix $A \in \mathbb{R}^{n\times n}$ being symmetric positive (semi-definite) definite:*

(i) A is symmetric positive (semi-) definite if and only if all of its eigenvalues are (non-negative) positive.

(ii) A is symmetric positive (semi-) definite if and only if there exists a nonsingular matrix $C \in \mathbb{R}^{n\times n}$ such that $C^T AC$ is symmetric positive (semi-) definite.

(iii) A is symmetric positive definite if and only if there exists a nonsingular lower triangular matrix G with positive diagonal entries such that $A = GG^T$. The matrix G in such a factorisation is referred to as the Cholesky factor of A.

(iv) For the matrix A presented in the form

$$A = \begin{pmatrix} A_1 & A_2 \\ A_2^T & A_3 \end{pmatrix}, \tag{1.2.1}$$

where each of A_1, A_2 and A_3 are block matrices, the following properties hold

(a) $A \succ 0$ if and only if $A_1 \succ 0$ and $A_3 - A_2^T A_1^{-1} A_2 \succ 0$.

(b) If $A_1 \succ 0$, then $A \succeq 0$ if and only if $A_3 - A_2^T A_1^{-1} A_2 \succeq 0$.

Proof. For each of the properties (i), (ii) and (iii), the reader is referred to [79, p. 398], [79, p. 399] and [79, pp. 406 – 407] respectively.

For the first part of (iv), note that (1.2.1) can be decomposed into the product of three matrices as follows

$$\begin{pmatrix} I & A_2^T A_1^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} A_1 & 0 \\ 0 & A_3 - A_2^T A_1^{-1} A_2 \end{pmatrix} \begin{pmatrix} I & A_2 A_1^{-1} \\ 0 & I \end{pmatrix}^T =: VDV^T, \tag{1.2.2}$$

where the matrix $S_{A_1} := A_3 - A_2^T A_1^{-1} A_2$ is referred to as the Schur complement of the block A_1 in A. Using the following observation

$$V^{-1} = \begin{pmatrix} I & -A_2 A_1^{-1} \\ 0 & I \end{pmatrix},$$

we may use the result in (ii) for the matrix D. This block diagonal matrix will be symmetric positive definite provided each diagonal block is. Therefore,

$$A_1 \succ 0 \qquad \text{and} \qquad A_3 - A_2^T A_1^{-1} A_2 \succ 0,$$

as required.

For the second part of (iv), we again use the decomposition as described in (1.2.2), and an application of (ii) for the matrix D to see that

$$A \succeq 0 \iff VDV^T \succeq 0 \iff D \succeq 0.$$

In this case, the block diagonal matrix D will only be symmetric positive semi-definite if each diagonal block is. We are already given that $A_1 \succ 0$, therefore it must hold that $A_3 - A_2^T A_1^{-1} A_2 \succeq 0$. Note the requirement of symmetric positive definiteness for the lead block to enable construction of the Schur complement. □

We also introduce the notion of a matrix pencil as follows

Definition 1.2.2. *A matrix pencil, of degree n, is defined via the following polynomial*

$$P(A_1, \ldots, A_n) := A_0 + \lambda A_1 + \lambda^2 A_2 + \ldots + \lambda^n A_n,$$

where each $A_i \in \mathbb{R}^{n\times n}$ for $i = 1, \ldots, n$ and $\lambda \in \mathbb{R}$. In particular, the linear matrix pencil, denoted $(A, B) := A - \lambda B$ will be considered in this thesis with A symmetric and $B \in \mathbb{S}^n$.

We close this section with the following lemma, essentially relating a symmetric matrix to its corresponding eigenvalues and eigenvectors through a similarity transformation.

Lemma 1.2.2. *(Spectral Theorem) Let $A \in \mathbb{R}^{n\times n}$ be a symmetric matrix with eigenvalues $\lambda_1, \ldots, \lambda_n$ and corresponding eigenvectors $\mathbf{v}_1, \ldots, \mathbf{v}_n$. Then, A may be written in terms of the following matrix product*

$$A := VDV^T,$$

referred to as the spectral decomposition of A, where $D := \operatorname{diag}(\lambda_1, \ldots, \lambda_n) \in \mathbb{R}^{n\times n}$, with $\mathbf{v}_1, \ldots, \mathbf{v}_n$ representing the columns of the matrix V.

Proof. Omitted, but may be found in [79, pp. 46 – 47]. □

1.2.2 Functional Analysis

This section contains results from Functional Analysis that will be used during the course of this thesis. In particular, the books [54] entitled *'Partial Differential Equations'* by Evans, [92] entitled *'Introductory Real Analysis'* by Kolmogorov and Fomin and also [104] entitled *'Functional Analysis'* by Lax are useful and recommended texts in order to supplement terminology presented throughout the thesis. We begin by defining the space of continuous functions as follows

$$C^0(\Omega) := \left\{ f \colon \Omega \to \mathbb{R} \;\middle|\; f \text{ continuous on } \Omega \right\}. \tag{1.2.3}$$

In the above, and throughout the duration of this section, the space $\Omega \subseteq \mathbb{R}^d$ is assumed to be both open and bounded with Lipschitz boundary[I] $\partial\Omega$ and closure $\overline{\Omega} := \Omega \cup \partial\Omega$, with a typical point in $\mathbb{R}^d$ denoted $\mathbf{x} = (x_1, x_2, \ldots, x_d)^T$. Through the introduction of the multi-index notation for $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_d)$, with length $|\alpha| := \alpha_1 + \cdots + \alpha_d$, where each $\alpha_i \in \mathbb{Z}_{0_+}$, higher-order derivatives can be expressed efficiently in the following manner

$$D^\alpha f = D_1^{\alpha_1} \ldots D_d^{\alpha_d} f = \frac{\partial^{|\alpha|} f}{\partial x_1^{\alpha_1} \ldots \partial x_d^{\alpha_d}}.$$

Using this notation along with (1.2.3), we define $C^k(\Omega)$ for $k \in \mathbb{Z}_{0_+}$ in the following way

$$C^k(\Omega) := \left\{ f \colon \Omega \to \mathbb{R} \;\middle|\; D^\alpha f \in C^0(\Omega),\ 0 \leq |\alpha| \leq k \right\}. \tag{1.2.4}$$

A function belonging to $C^k(\Omega)$ therefore possesses at least k continuous derivatives. As mentioned, $C^0(\Omega)$ defines the space of continuous functions. Using (1.2.4), $C^1(\Omega)$ can be seen to define the space of continuously differentiable functions, and $C^\infty(\Omega)$ the space of smooth functions possessing derivatives of all orders, namely

[I] Broadly speaking, a Lipschitz boundary is a boundary with sufficient regularity that locally amounts to the graph of a Lipschitz continuous function, where a function f is said to be Lipschitz continuous if there exists a non-negative constant c_L such that $|f(x) - f(y)| \leq c_L |x - y|$ for all $x, y \in \Omega$.

$$C^\infty(\Omega) := \bigcap_{k \in \mathbb{Z}_{0_+}} C^k(\Omega).$$

We now present the notion of L^p spaces, requiring an understanding of key concepts from measure theory. For further information, the interested reader should consult [54, pp. 684 – 688].

Definition 1.2.3. *(L^p Spaces) The space $L^p(\Omega, \mathcal{F}, \mu)$ is defined in the following way*

$$L^p(\Omega, \mathcal{F}, \mu) := \left\{ f(\mathbf{x}) \colon \Omega \to \mathbb{R} \;\middle|\; \int_\Omega |f(\mathbf{x})|^p \, d\mu(\mathbf{x}) < \infty \right\},$$

where $(\Omega, \mathcal{F}, \mu)$ denotes a σ-finite measurable space. Ω is the underlying space, $\mathcal{F}$ is the σ-algebra of measurable sets, and μ the measure.

Unless necessary, the notation will be simplified to $L^p(\Omega) := L^p(\Omega, \mathcal{F}, \mu)$. In the particular case of $p = 2$, namely:

$$L^2(\Omega) = \left\{ f(\mathbf{x}) \colon \Omega \to \mathbb{R} \;\middle|\; \int_\Omega |f(\mathbf{x})|^2 \, \mathrm{d}\mu(\mathbf{x}) < \infty \right\},$$

we define the Lebesgue space of square-integrable functions defined on Ω equipped with the inner product $\langle \cdot, \cdot \rangle_{L^2(\Omega)} \colon L^2(\Omega) \times L^2(\Omega) \to \mathbb{R}$ defined as follows

$$\langle f, g \rangle_{L^2(\Omega)} := \int_\Omega f(\mathbf{x}) g(\mathbf{x}) \, \mathrm{d}\mathbf{x}.$$

The corresponding norm for $L^2(\Omega)$ is

$$\|f\|^2_{L^2(\Omega)} := \langle f, f \rangle_{L^2(\Omega)} = \int_\Omega |f(\mathbf{x})|^2 \, \mathrm{d}\mathbf{x}.$$

For a function $f \colon \Omega \to \mathbb{R}$, we define its support by the set

$$\mathrm{supp}_\Omega(f) := \overline{\{ \mathbf{x} \in \Omega \mid f(\mathbf{x}) \neq 0 \}},$$

where f has a compact support if $\mathrm{supp}(f)$ is a strict subset of Ω. Based on this, we define the space of continuous functions with compact support as follows

$$C_0^k(\Omega) := \left\{ f \in C^k(\Omega) \;\middle|\; \mathrm{supp}_\Omega(f) \subset \Omega \right\}.$$

We also consider the space

$$L^1_{loc}(\Omega) := \left\{ f \in L^1(\Omega) \;\middle|\; f_{|_M} \in L^1(M)\ \forall M \subset \Omega,\ M \text{ compact} \right\},$$

corresponding to the set of locally integrable functions, namely the set of all Lebesgue measurable functions in Ω that are integrable on any compact subset of Ω.

Based on previous definitions, we are now in a position to introduce the concept of the weak derivative.

Definition 1.2.4. *(Weak Derivative) A function $f \in L^1_{loc}(\Omega)$ is referred to as the α^{th} weak derivative of $g \in L^1_{loc}(\Omega)$ provided*

$$\int_\Omega g(\mathbf{x}) D^\alpha \psi(\mathbf{x})\, d\mathbf{x} = (-1)^{|\alpha|} \int_\Omega f(\mathbf{x}) \psi(\mathbf{x})\, d\mathbf{x} \qquad \forall \psi \in C_0^{|\alpha|}(\Omega).$$

The weak derivative can be viewed as a special case of the more general distributional derivative, which will not be introduced here but can be found in a number of sources, including [178]. In comparison to the definition of the classical derivative, involving the limit of difference quotients, weak derivatives are only defined in terms of an integral. The Generalised Variational Lemma [176, p. 24] can be used to illustrate that weak derivatives are uniquely determined up to a set of measure zero, meaning that they remain unaltered under arbitrary changes to the function g on such a set.

Based on the definition, if g is continuously differentiable on Ω up to order k, then for each α such that $|\alpha| \leq k$, the classical partial derivative $D^\alpha f$ corresponds to the α^{th} weak derivative of g. Therefore, the notation $D^\alpha f$ is also used for the α^{th} weak derivative of f, where we note that $D^\alpha f = f$ in the case where $|\alpha| = 0$.

Using the definition of the weak derivative, we now introduce the notion of Sobolev spaces, denoted by $W^{k,p}(\Omega)$.

Definition 1.2.5. *(Sobolev Spaces) For a given $p \in [1, \infty]$ and $k \in \mathbb{N}_{0+}$, the Sobolev space of index k is defined by the set*

$$W^{k,p}(\Omega) := \left\{ f \in L^1_{loc}(\Omega) \;\middle|\; D^\alpha f \in L^p(\Omega), |\alpha| \leq k \right\}. \tag{1.2.5}$$

When coupled with the following norm

$$\|f\|_{W^{k,p}(\Omega)} := \|f\|_{k,p} = \left(\sum_{|\alpha| \leq k} \int_\Omega |D^\alpha f|^p \, d\mathbf{x} \right)^{1/p} \qquad p \in [1, \infty), \tag{1.2.6a}$$

$$\|f\|_{W^{k,\infty}(\Omega)} := \|f\|_{k,\infty} = \max_{|\alpha| \leq k} \left(\operatorname*{ess\,sup}_{\mathbf{x} \in \Omega} |D^\alpha f| \right) \qquad p = \infty, \tag{1.2.6b}$$

the spaces $W^{k,p}(\Omega)$ represent Banach spaces. Here, we note that the essential supremum (denoted ess sup) of a function amounts to the supremum of that function up to a set of measure zero. The space $C^k(\Omega)$ is dense[I] in $W^{k,p}(\Omega)$, with a famous result by Meyers and Serrin [120] showing that, for a Lipschitz domain Ω, the completion of the space

$$S^k := \left\{ f \in C^k(\Omega) \;\middle|\; \|f\|_{k,p} < \infty \right\},$$

is equivalent to $W^{k,p}(\Omega)$ in the case $p \in [1, \infty)$. Therefore, one can describe Sobolev spaces in terms of the closure of $C^k(\Omega)$ with respect to the corresponding norms in (1.2.6a).

The resulting space realised through the closure of $C_0^\infty(\Omega)$ with respect to the relevant norm described in (1.2.6) corresponds to a closed subspace of $W^{k,p}(\Omega)$, denoted $W_0^{k,p}(\Omega)$. In this thesis, the particular case of $p = 2$ will be of primary interest, where we denote $H^k(\Omega) := W^{k,2}(\Omega)$ and $H_0^k(\Omega) := W_0^{k,2}(\Omega)$. Here, the spaces $H^k(\Omega)$, $k \geq 0$ equipped with the inner product

$$\langle f, g \rangle_{H^k(\Omega)} = \sum_{|\alpha| \leq k} \langle D^\alpha f, D^\alpha g \rangle_{L^2(\Omega)},$$

[I] For details on the notion of denseness, the reader should consult [92, p. 48].

and associated norm

$$\|f\|_{k,\Omega}^2 = \sum_{|\alpha|\leq k} \|D^\alpha f(\mathbf{x})\|_{L^2(\Omega)}^2 \tag{1.2.7}$$

are Hilbert spaces, where $H^0(\Omega) = L^2(\Omega)$ by construction. We also describe the associated seminorm to $H^k(\Omega)$ using (1.2.7) in the following way

$$|f|_{k,\Omega}^2 := \sum_{|\alpha|=k} \|D^\alpha f(\mathbf{x})\|_{L^2(\Omega)}^2, \tag{1.2.8}$$

adhering to the usual properties of norms with the relaxation allowing $|f|_{k,\Omega} = 0$ for $f \neq 0$.

At this point, we describe the notion of norm equivalence in the following manner

Definition 1.2.6. *(Norm Equivalence) Two norms $\|\cdot\|_A$ and $\|\cdot\|_B$ on a vector space V are said to be equivalent if there exist constants $c_1, c_2 > 0$ such that*

$$c_1\|f\|_A \leq \|f\|_B \leq c_2\|f\|_A \qquad \forall f \in V.$$

In this case, we write $\|\cdot\|_A \sim \|\cdot\|_B$.

Using [64, p. 136], one can illustrate a norm equivalence between the seminorm described in (1.2.8) and the corresponding norm in (1.2.6). Such an equivalence is important in the treatment of boundary value problems, allowing for the seminorm to be used as the natural norm for certain Sobolev spaces.

Based on the definition of $H^k(\Omega)$, the primary focus of this thesis will involve the space where $k = 1$. In this case, the space $H^1(\Omega)$ may be presented as follows

$$H^1(\Omega) = \left\{ f \in L^2(\Omega) \;\middle|\; D^\alpha f \in L^2(\Omega),\ 1 \leq |\alpha| \leq d \right\}, \tag{1.2.9}$$

equipped with the norm

$$\|f\|_{1,\Omega}^2 = \int_\Omega |f|^2 \,\mathrm{d}\mathbf{x} + \int_\Omega |\nabla f|^2 \,\mathrm{d}\mathbf{x}.$$

Using (1.2.9), we can describe $H^1_D(\Omega)$ as

$$H^1_D(\Omega) := \left\{ f \in H^1(\Omega) \;\middle|\; f = 0 \text{ on } \partial\Omega_D \subset \partial\Omega \right\}. \tag{1.2.10}$$

The above space enforces homogeneous Dirichlet conditions on a subset $\partial\Omega_D$ of the boundary $\partial\Omega$. In the case where $\partial\Omega_D$ corresponds to the entire boundary of Ω, we refer to the space (1.2.10) as $H^1_0(\Omega)$. These spaces will be used extensively within this thesis.

In order to compare different Sobolev spaces, the notion of embedding is used, as outlined below.

Definition 1.2.7. *(Continuous Embedding) For two normed spaces X and Y such that $X \subset Y$, we say that X is continuously embedded in Y if there exists a positive constant c_E such that*

$$\|f\|_Y \leq c_E \|f\|_X \qquad \forall f \in X.$$

We denote such a relation by $X \hookrightarrow Y$.

There are a number of theoretical results related to embeddings as illustrated (for instance) in [2]. A particular result of concern to our work is based on the definitions of $H^1_0(\Omega)$, $H^1(\Omega)$ and $L^2(\Omega)$ along with the associated norms, where we have the following relation

$$H^1_0(\Omega) \hookrightarrow H^1(\Omega) \hookrightarrow L^2(\Omega).$$

We now consider an extension to the notion of a Sobolev space described to this point to describe spaces for a real index $\theta \in [0,1]$. Based on the presentation in [109], these are defined for the pair $[H^1(\Omega), L^2(\Omega)]$ as interpolation spaces of index $1-\theta$ as follows

$$H^\theta(\Omega) := \left[H^1(\Omega), L^2(\Omega)\right]_{1-\theta}, \tag{1.2.11}$$

equipped with the norm $\|\cdot\|_{s,\Omega}$ defined in the following manner

$$\|f\|_{\theta,\Omega}^2 := \int_\Omega |f(x)|^2 \,\mathrm{d}x + \int_\Omega \int_\Omega \frac{|f(x) - f(y)|^2}{|x-y|^{d+2\theta}} \,\mathrm{d}x\mathrm{d}y.$$

In the case of $\theta = 1/2$, we define the following interpolation space

$$H^{1/2}(\Omega) := \left[H^1(\Omega), L^2(\Omega)\right]_{1/2}.$$

Additionally, using the presentation in (1.2.10) we denote the interpolation space for the pair $[H_D^1(\Omega), L^2(\Omega)]$ as

$$H_{00}^{1/2}(\Omega) := \left[H_D^1(\Omega), H^0(\Omega)\right]_{1/2},$$

where we use the fact that $L^2(\Omega) = H^0(\Omega)$. The space $\left(H_{00}^{1/2}(\Omega)\right)^* \subset H^{-1/2}(\Omega)$ denotes the dual of the space $H_{00}^{1/2}(\Omega)$, where $H^{-1/2}(\Omega) = \left(H^{1/2}(\Omega)\right)^* = \left(H_0^{1/2}(\Omega)\right)^*$. Here, the space $H_0^{1/2}(\Omega)$ corresponds to the completion of $C_0^\infty(\Omega)$ in $H^{1/2}(\Omega)$, as illustrated (for instance) in [109, p. 60].

From the definition of $W^{k,p}(\Omega)$ in (1.2.5), functions belonging to Sobolev spaces can only be seen to be uniquely defined almost everywhere within the domain Ω as a consequence of their belonging to $L^p(\Omega)$. Now, since the measure of the boundary $\partial\Omega$ is zero in $\mathbb{R}^d$, it is clear that functions belonging to a Sobolev space are not well defined on the prescribed boundary. However, for a given Sobolev function one can describe its so-called trace on the boundary, corresponding to a well defined mapping even for Sobolev functions with piecewise discontinuities. In the case of a continuous function within the closure of Ω, the value of its trace can be seen to agree with its corresponding boundary value. In the case $p = 2$, we describe these ideas mathematically through the following lemma.

Lemma 1.2.3. *(Trace Operator) Suppose that $\Omega \subset \mathbb{R}^d$ is a Lipschitz domain. Then, there exists a compact and continuous linear operator $\gamma_0 \colon H^1(\Omega) \to H^{1/2}(\partial\Omega)$, where $\gamma_0 u = u_{|\partial\Omega}$ whenever $u \in H^1(\Omega) \cap C^0(\overline{\Omega})$.*

Proof. Omitted, but can be determined as a particular case of the proof in [54, pp. 258 –

259]. □

The operator γ_0 is known as the trace operator, with $\gamma_0 u$ referred to as the trace of $u \in H^1(\Omega)$. Due to the continuity of γ_0 (see [199, Theorem 8.7]), there exists a positive constant $c_{\gamma_0} := c_{\gamma_0}(\Omega)$ such that

$$\|\gamma_0 v\|_{1/2,\partial\Omega} \le c_{\gamma_0}(\Omega)\|v\|_{1,\Omega} \qquad \forall v \in H^1(\Omega). \tag{1.2.12}$$

Additionally, for the trace operator defined by the mapping $\gamma_0 : H^1_D(\Omega) \to H^{1/2}_{00}(\partial\Omega_D)$ a similar inequality can be shown to hold

$$\|\gamma_0 v\|_{1/2,\partial\Omega_D} \le c_{\gamma_0}(\Omega)\|v\|_{1,\Omega} \qquad \forall v \in H^1_D(\Omega), \tag{1.2.13}$$

where $\|\cdot\|_{1/2,\partial\Omega}$ and $\|\cdot\|_{1/2,\partial\Omega_D}$ denote the associated norms in each respective case. For the interested reader, an illustration of the action of the trace operator on the solution to a model linear elasticity problem can be seen in Section A.1.

We close this section with the notion of both Gâteaux and Fréchet differentiability as follows.

Definition 1.2.8. *A mapping $\mathcal{G}\colon U \subset V \to W$ from an open subset U of V to W is said to be $\mathcal{G}-$differentiable (Gâteaux differentiable) at a point $u \in U$ in the direction w if the limit*

$$\mathcal{G}'(u,w) := \lim_{h\to 0} \frac{\mathcal{G}(u+hw)-\mathcal{G}(u)}{h},$$

exists in $\mathbb{R}$. We refer to this unique limit as the Gâteaux derivative of $\mathcal{G}$ in the direction w.

Definition 1.2.9. *A mapping $\mathcal{H}\colon U \subset V \to W$from an open subset U of V to W is said to be $\mathcal{H}-$differentiable (Fréchet differentiable) at a point $u \in U$ provided that a continuous linear mapping $\mathcal{H}'_u\colon V \to W$ exists such that*

$$\lim_{\bar{w}\to 0} \frac{\|\mathcal{H}(u+\bar{w})-\mathcal{H}(u)-\mathcal{H}'_u\bar{w}\|_W}{\|\bar{w}\|_V} = 0.$$

The unique $\mathcal{H}'_u$ satisfying the above is referred to as the Fréchet derivative.

Remark 1.2.1. *The relationship between the Gâteaux and Fréchet differentials can be seen by setting $\bar{w} = hw$ in the above definition. As $h \to 0$, $\mathcal{H}'_u w = \mathcal{G}'(u, w)$ and provided that $\mathcal{G}'(u, w)$ can be written in the form $\mathcal{B}w$, we have that $\mathcal{H}'_u = \mathcal{B}$.*

1.2.3 Optimisation

This section contains results from optimisation that will be used throughout this thesis. In particular, the book [132] entitled *'Numerical Optimization - Second Edition'* by Nocedal and Wright is a useful read and will be used at several points in this section. We begin by introducing the notion of the gradient and Hessian of a function, followed by a brief introduction to convexity.

Definition 1.2.10. *(Gradient and Hessian) Let $f\colon \mathbb{R}^n \to \mathbb{R}$. At a given point of the domain, the gradient of f is defined as*

$$\nabla f(\mathbf{x}) := \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n}\right)^T.$$

Furthermore, suppose that $\mathbf{x} = (x_1, x_2, \dots, x_n)^T =: \left((\mathbf{y}^1)^T, \dots, (\mathbf{y}^m)^T\right)^T$ with $m < n$ such that $m_j := |\mathbf{y}^j|$ for $j = 1, \dots, m$. Then, for a given $\mathbf{y}^j$, we define

$$\nabla_{\mathbf{y}^j} f(\mathbf{x}) := \left(\frac{\partial f(\mathbf{x})}{\partial y^j_1}, \dots, \frac{\partial f(\mathbf{x})}{\partial y^j_{m_j}}\right)^T.$$

The Hessian of f is a symmetric $n \times n$ matrix of the form

$$\nabla^2 f(\mathbf{x}) := \left(\frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j}\right) \qquad i, j = 1, \dots, n.$$

Definition 1.2.11. *(Convexity) A set $\Omega \subset \mathbb{R}^d$ is called convex if for all $\mathbf{x}, \mathbf{y} \in \Omega$,*

$$\alpha\mathbf{x} + (1 - \alpha)\mathbf{y} \in \Omega \qquad \forall \alpha \in [0, 1].$$

A function $f\colon \mathbb{R}^n \to \mathbb{R}$ *is called convex if for all* $\mathbf{x}, \mathbf{y} \in \Omega$*,*

$$f(\alpha \mathbf{x} + (1-\alpha)\mathbf{y}) \leq \alpha f(\mathbf{x}) + (1-\alpha) f(\mathbf{y}) \qquad \forall \alpha \in [0,1].$$

If the function $-f$ *is convex, then* f *is said to be concave.*

Straightforward examples of convex functions include linear functions, namely f such that $f(\mathbf{x}) = a\mathbf{x} + b$, and also quadratic functions of the form $f(\mathbf{x}) = \mathbf{x}^T H \mathbf{x}$, with H a symmetric positive semidefinite matrix.

An optimisation problem may be described mathematically in the following manner.

Definition 1.2.12. *(Minimisation Problem) A general minimisation problem involves solving a problem of the following form:*

$$\textit{Find } \mathbf{x}^* = \ \textit{arg} \min_{\mathbf{x} \in \mathcal{S}} \ f(\mathbf{x}). \tag{1.2.14}$$

where the feasible set $\mathcal{S}$ *is defined as*

$$\mathcal{S} := \{\mathbf{x} \in \mathbb{R}^n \mid g_i(\mathbf{x}) = 0,\, h_j(\mathbf{x}) \geq 0,\, i \in \mathcal{E},\, j \in \mathcal{I}\}\,.$$

The functions f, g_i and h_j represent a mapping from elements of $\mathbb{R}^n$ to $\mathbb{R}$. The sets $\mathcal{E}$ and $\mathcal{I}$ represent index sets that specify equality and inequality constraints respectively, where we assume that the functions f, g_i, $h_j \in C^2(\mathbb{R}^n)$ to ensure the existence of necessary derivatives.

The above provides a definition for a minimisation problem, however a maximisation problem may also be considered based on replacement of min with max in (1.2.14). The function f as described in (1.2.14) is commonly referred to as the objective function.

We now distinguish two cases involving the makeup of $\mathcal{S}$, namely when $\mathcal{S}$ is equal to $\mathbb{R}^d$, and when $\mathcal{S}$ is a proper subset of $\mathbb{R}^d$. Unconstrained problems are said to arise in the former case (where $\mathcal{E} \cup \mathcal{I} = \emptyset$), and constrained problems in the latter (where $\mathcal{E} \cup \mathcal{I} \neq \emptyset$). Using Definition 1.2.12 an unconstrained minimisation problem will be labelled as (UP) whereas a constrained minimisation problem will be referred to as (CP).

A nonlinear problem emerges whenever nonlinear terms are involved in at least one of f, g_i or h_j. Problems of this type can be seen to arise naturally in a number of engineering and physical science applications, including design, operations and control problems. More recently, formulations involving problems in management and economic sciences have also been studied.

We now introduce some important theoretical concepts behind unconstrained problems, beginning with the concepts of a local and global solution to the problem (UP).

Definition 1.2.13. *(Local/Global Solution) We say that a point $\mathbf{x}^* \in \mathcal{S}$ is a local solution of (UP) if there exists a neighbourhood $U(\mathbf{x}^*)$ such that $f(\mathbf{x}) \geq f(\mathbf{x}^*)\ \forall \mathbf{x} \in \mathcal{S} \cap U(\mathbf{x}^*)$. A point $\mathbf{x}^* \in \mathcal{S}$ is referred to as an strict local solution of (UP) if there exists a neighbourhood $U(\mathbf{x}^*)$ such that $f(\mathbf{x}) > f(\mathbf{x}^*)\ \forall \mathbf{x} \in \mathcal{S} \cap U(\mathbf{x}^*) \setminus \{\mathbf{x}^*\}$. A point $\mathbf{x}^* \in \mathbb{R}^n$ is a global solution of (UP) if $f(\mathbf{x}) \geq f(\mathbf{x}^*)\ \forall \mathbf{x} \in \mathcal{S}$.*

Introducing Definition 1.2.13 enables the following theorem to be presented.

Theorem 1.2.1. *Let $f\colon \mathbb{R}^n \to \mathbb{R}$ be convex.*

- *Suppose that $\mathbf{x}^*$ is a local solution of (UP). Then, $\mathbf{x}^*$ is a global solution of (UP).*
- *In addition, if $f \in C^1(\mathbb{R}^n)$, any stationary point also corresponds to a global minimiser of f.*

Proof. Omitted, however can be found in [132, pp. 16 – 17]. □

This theorem is particularly useful, and illustrates the benefits of solving problems involving convex sets and functions. For constrained problems, we also require convexity of the feasible set $\mathcal{S}$. We now introduce both first and second order criteria in order to identify both local and global minima for unconstrained optimisation problems.

Theorem 1.2.2. *(1^{st} Order Optimality Conditions) Let $\mathbf{x}^*$ be a local solution of (UP) and let f be continuously differentiable in an open neighbourhood of $\mathbf{x}^*$. Then $\nabla f(\mathbf{x}^*) = 0$.*

Proof. Omitted, however can be found in [132, pp. 14 – 15], involving the consideration of a first order Taylor expansion about $\mathbf{x}^*$. □

It should be noted that $\mathbf{x}^*$ is referred to as a stationary point if $\nabla f(\mathbf{x}^*) = 0$. As a consequence of the above theorem, any local minimum must also be a stationary point.

Theorem 1.2.3. *(2^{nd} Order Optimality Conditions) Let the function $f\colon \mathbb{R}^n \to \mathbb{R}$ be (at least) twice differentiable at a point $\mathbf{x}^* \in \mathbb{R}^n$. Then,*

1. *(necessary) If $\mathbf{x}^*$ is a local solution to (UP), then $\nabla f(\mathbf{x}^*) = 0$ and $\nabla^2 f(\mathbf{x}^*)$ is symmetric positive semi-definite.*

2. *(sufficient) If $\nabla f(\mathbf{x}^*) = 0$ and $\nabla^2 f(\mathbf{x}^*)$ is symmetric positive definite, then $\mathbf{x}^*$ is a strict local solution to (UP).*

Proof.

1. Omitted, however can be found in [132, p. 15] involving the use of a second order Taylor expansion.

2. Omitted, however can be found in [132, p. 16] involving use of the generalised mean value theorem. □

We have now introduced both first and second order criteria enabling for the classification of stationary points for unconstrained minimisation problems. We now look to extend this theory to constrained problems.

In order to solve constrained optimisation problems of increasing complexity, we must transform the problem into a workable form so that tools from unconstrained optimisation can be used. We begin by introducing the so-called Lagrangian function.

Definition 1.2.14. *(Lagrangian) The function $\mathcal{L} : \mathbb{R}^n \times \mathbb{R}^{|\mathcal{E}|} \times \mathbb{R}^{|\mathcal{I}|} \to \mathbb{R}$ defined by:*

$$\mathcal{L}(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) := f(\mathbf{x}) - \sum_{i \in \mathcal{E}} \lambda_i g_i(\mathbf{x}) - \sum_{j \in \mathcal{I}} \mu_j h_j(\mathbf{x}).$$

is called the Lagrangian of the constrained minimisation problem (CP).

An explanation of the use of the Lagrangian function in constrained optimisation problems can be found in [132, pp. 308 – 315]. We now introduce the notion of active and inactive constraints.

Definition 1.2.15. *An inequality constraint $h_j(\mathbf{x}) \geq 0$, $j \in \mathcal{I}$ is called active in $\mathbf{x} \in \mathcal{S}$ if $h_j(\mathbf{x}) = 0$, and inactive in $\mathbf{x} \in \mathcal{S}$ if $h_j(\mathbf{x}) > 0$. A set $\mathcal{A}$ of active constraints is given by*

$$\mathcal{A}(\mathbf{x}) := \mathcal{E} \cup \mathcal{Q},$$

where the set $\mathcal{Q}$ is defined as

$$\mathcal{Q} := \{j \in \mathcal{I} \mid h_j(\mathbf{x}) = 0\}.$$

It is desirable for active constraints as described above to remain active under feasible perturbations of solutions to (1.2.14) in order to rule out particular points located (for instance) at cusps of the constraint boundary. Such a requirement is referred to as a constraint qualification, ensuring that linear approximations to active nonlinear constraints characterise all feasible perturbations about solutions to (1.2.14). The requirement as outlined is evidently satisfied in the case where the constraints are linear. It is also satisfied whenever the gradients of the active constraints are linearly independent, as described in the following definition

Definition 1.2.16. *(LICQ) The Linear Independence Constraint Qualification (LICQ) is said to hold at $x \in \mathcal{S}$ if the gradients of the active constraints $\{\nabla g_i(\mathbf{x}), \nabla h_j(\mathbf{x}) \mid i \in \mathcal{E},\ j \in \mathcal{Q}\}$ form a linearly independent set.*

In practise, insisting on linear independence can be seen as a fairly strong condition to impose on the gradients of the active constraints. Nevertheless, this is one of many constraint qualifications that have been proposed, and in fact potentially troublesome local minimisers may be ruled out for a number of problems by imposing notably weaker conditions than the LICQ.

Now, we may use the LICQ as described above to define the 1$^{\text{st}}$ order necessary optimality conditions for general constrained optimisation problems. This is a common approach used to solve numerous problems in nonlinear optimisation.

Theorem 1.2.4. *(1st Order Necessary Condition - Karush-Kuhn-Tucker (KKT)) Let* $\mathbf{x}^*$ *be a local solution of (CP) and assume that (LICQ) holds at* $\mathbf{x}^*$*. Then, there exists vectors* $\boldsymbol{\lambda}^*$, $\boldsymbol{\mu}^*$ *(with elements* λ_i^*, μ_j^*, $i \in \mathcal{E}$, $j \in \mathcal{I}$*) such that the following conditions hold:*

$$\text{(KKT)} \quad \begin{cases} \textit{Stationarity:} & \nabla_{\mathbf{x}} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = 0 & \\ \textit{Primal Feasibility:} & \begin{cases} g_i(\mathbf{x}^*) = 0 & \qquad \forall i \in \mathcal{E} \\ h_j(\mathbf{x}^*) \geq 0 & \qquad \forall j \in \mathcal{I} \end{cases} \\ \textit{Dual Feasibility:} & \mu_j^* \geq 0 & \forall j \in \mathcal{I} \\ \textit{Comp. Slackness:} & \mu_j^* h_j(\mathbf{x}^*) = 0 & \forall j \in \mathcal{I}. \end{cases}$$

Proof. Omitted, however a proof can be found in both [18] and [132, pp. 323 – 330] which involves the use of a generalization of Farkas Lemma and the Bolzano-Weierstrass property for compact sets. □

Required in the proof is the set of linearised feasible directions, which is defined as follows.

Definition 1.2.17. *For a feasible point* $\mathbf{x}$*, coupled with the set of active constraints* $\mathcal{A}(\mathbf{x})$*, the set of linearised feasible directions is defined as follows*

$$\mathcal{F}(\mathbf{x}) = \left\{ \mathbf{d} \in \mathbb{R}^n \;\middle|\; \mathbf{d}^T \nabla g_i(\mathbf{x}) = 0 \;\forall i \in \mathcal{E}, \mathbf{d}^T \nabla h_k(\mathbf{x}) \geq 0 \;\forall k \in \mathcal{A}(\mathbf{x}) \cap \mathcal{I} \right\}.$$

This will be used when introducing second order conditions. The first order conditions provide information on the relation of the derivatives of the objective function and the active constraints at a solution $\mathbf{x}^*$. When the KKT conditions are satisfied, moving along a vector $\mathbf{d} \in \mathcal{F}(\mathbf{x}^*)$ will affect the approximation of the objective function by either an increase through the action of $\mathbf{d}^T f(\mathbf{x}^*) > 0$, or by remaining the same due to $\mathbf{d}^T f(\mathbf{x}^*) = 0$. However, for directions $\mathbf{d} \in \mathcal{F}(\mathbf{x}^*)$ where $\mathbf{d}^T f(\mathbf{x}^*) = 0$, simply using the first order information does not indicate whether moving in this direction will increase or decrease the objective function. We therefore look to second order conditions, so that information relating to either the increase or decrease of f can be computed. It will be seen that the second order conditions effectively track the curvature of the Lagrangian function in the direction $\mathbf{d} \in \mathcal{F}(\mathbf{x}^*)$ where $\mathbf{d}^T f(\mathbf{x}^*) = 0$.

We now introduce the notion of a critical cone $\mathcal{C}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ as follows

Definition 1.2.18. *(Critical Cone) For the triple $(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ satisfying the conditions of Theorem 1.2.4, the critical cone is defined as follows*

$$\mathcal{C}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*) = \left\{ \mathbf{d} \in \mathcal{F}(\mathbf{x}^*) \;\middle|\; \mathbf{d}^T \nabla h_k(\mathbf{x}^*) = 0 \;\forall k \in \mathcal{A}(\mathbf{x}^*) \cap \mathcal{I} \;\text{ with }\; \mu_j^* > 0 \;\forall j \in \mathcal{I} \right\}.$$

The critical cone characterises the directions $\mathbf{d}$ whereby, for small changes in the value of the objective function, the active and equality constraints still hold. We are now in a position to introduce necessary second order criteria.

Theorem 1.2.5. *(2nd Order Necessary Conditions) Suppose that $(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ are a triple satisfying the conditions of Theorem 1.2.4, and that $\mathbf{x}^*$ satisfies the LICQ condition (Definition 1.2.16). Then*

$$\mathbf{d}^T \nabla^2_{\mathbf{xx}} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)\mathbf{d} \geq 0 \qquad \forall \mathbf{d} \in \mathcal{C}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*).$$

Proof. Omitted. However, the proof is presented in [132, pp. 332 – 333], involving similar ideas to the proof of the first order criteria for constrained optimisation seen previously. □

In addition to the above, the following theorem introduces sufficient conditions for $\mathbf{x}^*$ to be a strict local minimum.

Theorem 1.2.6. *(2nd Order Sufficient Conditions) Suppose that* $(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)$ *is a triple satisfying the conditions of Theorem 1.2.4. Additionally, suppose that*

$$\mathbf{d}^T \nabla^2_{\mathbf{xx}} \mathcal{L}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*)\mathbf{d} > 0 \qquad \forall \mathbf{d} \in \mathcal{C}(\mathbf{x}^*, \boldsymbol{\lambda}^*, \boldsymbol{\mu}^*),\ \mathbf{d} \neq \mathbf{0}.$$

Then, $\mathbf{x}^*$ *is a strict local minimum for (CP).*

Proof. The proof of this theorem is also omitted, however can be found in [132, pp. 333 – 335]. □

This completes the definitions and theory that will be in this thesis. Certain definitions and theorems may also be defined later, however their use will be restricted to the sections where they are introduced.

CHAPTER 2

LINEAR ELASTICITY

2.1 Introduction

This chapter begins with a brief introduction to elasticity, focussing on the deformation of solid objects as a result of prescribed loading conditions, along with the calculation of the resulting stresses and displacements. For the interested reader, a more in-depth presentation can be found in [127]. We begin by considering a solid elastic body occupying an open and connected domain $\Omega \subset \mathbb{R}^d$ with Lipschitz boundary $\partial\Omega = \partial\Omega_D \cup \partial\Omega_N$, with $d \in \{2, 3\}$ and clamping and traction imposed on $\partial\Omega_D$ and $\partial\Omega_N$ respectively as illustrated in Figure 2.1. Within Ω, we allow for the definition of areas of both fixed and void material, as represented by the black and white regions respectively within the domain. Under the application of both body forces $f : \Omega \to \mathbb{R}^d$ and boundary tractions $g : \partial\Omega_N \to \mathbb{R}^d$, the material is subject to deformation so that a given reference point $\mathbf{x}$ of the initial undeformed material is translated to the point $\mathbf{x}' = \mathbf{x} + u(\mathbf{x})$ of the deformed material, with $u := \left(u^{(c)}\right)_{c \in \mathbb{W}}$ denoting the displacement, where

$$\mathbb{W} = \begin{cases} \{x, y\} & \text{if } d = 2, \\ \{x, y, z\} & \text{if } d = 3. \end{cases}$$

A wide variety of contributing factors can be involved in the make-up of both f and g. Body forces act in a uniformly distributed manner across the whole domain with the

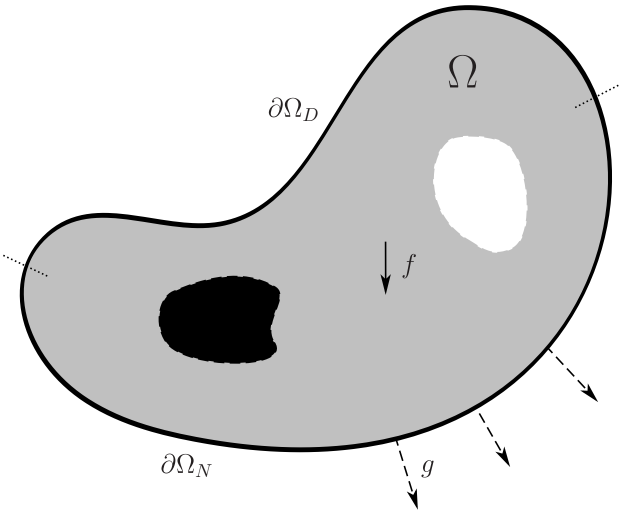


Figure 2.1: Pictorial example of a typical linear elasticity problem. The action of clamping corresponds to the application of homogeneous Dirichlet boundary conditions (namely $u = 0$) on $\partial\Omega_D$, with traction (denoted g) being imposed on the portion of the boundary denoted $\partial\Omega_N$.

exclusion of the boundary. Typical examples may involve the effects of gravitational pull, where $f = -\hat{p}g$ for some $\hat{p} \in \mathbb{R}$, or as a consequence of electromagnetic force, for instance. The boundary tractions correspond to contact forces acting on the surface of the domain. For instance, one can think of pressure within a fluid acting along a normal to a real fluid surface, with a force proportional to the surface area. Despite only acting at the boundary, the contact forces are interpreted as vector fields acting throughout the domain.

We now refer to the presentation in Appendix A.2 to describe the classical formulation of the problem. Under the assumption of positive measure [I] of dimension $d-1$, and also the application of homogeneous Dirichlet conditions for the clamped portion of the boundary, we look to find the displacement vector u and the stress tensor $\boldsymbol{\sigma}(u)$ such that

$$\begin{cases} \mathcal{L}u := -\nabla \cdot \boldsymbol{\sigma}(u) = f & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega_D, \\ \boldsymbol{\sigma}(u) \cdot \mathbf{n} = g & \text{on } \partial\Omega_N. \end{cases} \tag{2.1.1}$$

[I] A measure is non-negative by definition. This restriction simply rules out the case where the measure over the Dirichlet boundary is equal to zero. Such a requirement is necessary in order to illustrate well-posedness of the variational formulation, as described in the next section.

For this work, we consider materials for which a linear relation exists between the stress and strain tensors. Such a relation corresponds to Hooke's Law, described by the following equation

$$\boldsymbol{\sigma}(u) = E \colon \boldsymbol{\epsilon}(u), \tag{2.1.2}$$

where $\boldsymbol{\epsilon}(u)$ denotes the strain tensor as per Appendix A.2. The contraction operator $:$ is defined as

$$A \colon B := \sum_{i,j=1}^{d} A_{ij} B_{ij}.$$

In index notation, we write

$$\sigma_{ij}(u) = \sum_{k,l=1}^{d} E_{ijkl} \epsilon_{kl}(u),$$

where the fourth order tensor E represents the elasticity tensor satisfying standard conditions relating to symmetry, uniform boundedness and uniform positive definiteness (as presented, for instance, in [66]) as follows

$$E_{ijkl} = E_{jikl} = E_{klij} \qquad \forall i,j,k,l = 1,\ldots,d, \tag{2.1.3a}$$

$$E_{ijkl} \in L^{\infty}(\Omega) \qquad \forall i,j,k,l = 1,\ldots,d, \tag{2.1.3b}$$

$$\exists a > 0 \;\text{ such that }\; a|\zeta|^2 \leq \zeta \colon E \colon \zeta, \tag{2.1.3c}$$

with ζ denoting a symmetric second order tensor. In addition to the above, we also assume for the presentation in this chapter that the elasticity tensor is not only defined independently of the reference point $\mathbf{x}$, but also invariant under all Cartesian coordinate transformations (i.e.: rotational and directional independence). Such a material is referred to as *homogeneous* and *isotropic*, where the relation between the stress and strain tensor presented in (2.1.2) reduces to the following

$$\boldsymbol{\sigma}(u) = 2\mu\boldsymbol{\epsilon}(u) + \lambda \text{tr}\,(\boldsymbol{\epsilon}(u))I,$$

with I representing the identity matrix of appropriate size, $\text{tr}(\boldsymbol{\epsilon}(u))$ denoting the trace of the strain displacement matrix and both μ and λ corresponding to Lamé constants defined in the usual manner

$$\mu := \frac{E_Y}{2(1+\nu)}\,, \qquad\qquad \lambda := \frac{\nu E_Y}{(1+\nu)(1-2\nu)}\,,$$

where E_Y represents Young's modulus and $-1 < \nu < \frac{1}{2}$ Poisson's ratio.

Remark 2.1.1. *In general, the nature of the elasticity tensor is dependent on the constitutive properties of the material under consideration. For this work, the straight forward case of a homogeneous and isotropic material has been considered.*

2.2 Variational Formulation

We now consider the variational formulation of (2.1.1). We begin by defining the set of admissible displacements as

$$V_0 = \left\{ v \in \left[H^1(\Omega)\right]^d \;\middle|\; v = 0 \text{ on } \partial\Omega_D \right\} = \left[H^1_D(\Omega)\right]^d,$$

where $H^1(\Omega)$ and $H^1_D(\Omega)$ denote Sobolev spaces as defined in (1.2.9) and (1.2.10). By multiplying both sides of the equilibrium equation by $v \in V_0$ and integrating over Ω, we arrive at the following

$$-\int_\Omega (\nabla \cdot \boldsymbol{\sigma}(u))\, v \,\mathrm{d}\mathbf{x} = \int_\Omega f v \,\mathrm{d}\mathbf{x}. \tag{2.2.1}$$

Through the use of Green's first identity [68, p. 5],

$$-\int_\Omega (\nabla \cdot \boldsymbol{\sigma}(u))\, v \,\mathrm{d}\mathbf{x} = \int_\Omega \boldsymbol{\sigma}(u) \colon \boldsymbol{\epsilon}(v) \,\mathrm{d}\mathbf{x} - \int_{\partial\Omega} (\boldsymbol{\sigma}(u) \cdot \mathbf{n})\, v \,\mathrm{d}s, \tag{2.2.2}$$

(2.2.1) can now be written as

$$\int_\Omega \boldsymbol{\sigma}(u) \colon \boldsymbol{\epsilon}(v) \,\mathrm{d}\mathbf{x} - \int_{\partial\Omega} (\boldsymbol{\sigma}(u) \cdot \mathbf{n})\, v \,\mathrm{d}s = \int_\Omega f v \,\mathrm{d}\mathbf{x}. \tag{2.2.3}$$

Applying both Dirichlet and Neumann boundary conditions, (2.2.3) becomes

$$\int_\Omega \boldsymbol{\sigma}(u)\colon \boldsymbol{\epsilon}(v)\,\mathrm{d}\mathbf{x} = \int_{\partial\Omega_N} gv\,\mathrm{d}s + \int_\Omega fv\,\mathrm{d}\mathbf{x} =: F(v). \tag{2.2.4}$$

The quantity $F(v)$ represents the load linear form, or compliance, where $F(\cdot) : V_0 \to \mathbb{R}$. We also define $a_E(\cdot,\cdot) : V_0 \times V_0 \to \mathbb{R}$ as follows

$$\begin{aligned} a_E(u,v) &:= \int_\Omega \boldsymbol{\sigma}(u)\colon \boldsymbol{\epsilon}(v)\,\mathrm{d}\mathbf{x} = \int_\Omega \boldsymbol{\epsilon}(u)\colon E\colon \boldsymbol{\epsilon}(v)\,\mathrm{d}\mathbf{x} \\ &:= \int_\Omega \sum_{i,j,k,l=1}^{d} E_{ijkl}\epsilon_{ij}(u)\epsilon_{kl}(v)\,\mathrm{d}\mathbf{x}. \end{aligned} \tag{2.2.5}$$

This represents the energy bilinear form, namely the internal work of an elastic body at an equilibrium point u and for an arbitrary displacement v. Under this notation, we therefore look to determine $u \in V_0$ such that

$$a_E(u,v) = F(v) \qquad \forall v \in V_0. \tag{2.2.6}$$

The Lax-Milgram Lemma may be used to ensure the well-posedness of the variational formulation (2.2.6), which for the bilinear form given in (2.2.5) may be described in the following manner

Lemma 2.2.1. *(Lax-Milgram Lemma) Let $a_E(\cdot,\cdot) : V_0 \times V_0 \to \mathbb{R}$ denote a continuous, V_0-elliptic bilinear form. Then, for each $F \in V_0^*$, the variational equation (2.2.6) has a unique solution $u \in V_0$.*

Proof. The interested reader should consult [64, p. 145]. □

Based on the above, we require the bilinear form (2.2.5) to adhere to the following three properties

(i) (*Coercivity*) There exists a positive constant $c_1 \in \mathbb{R}$ dependent on both Ω and $\partial\Omega_D$ such that

$$c_1\|v\|_{1,\Omega}^2 \leq a_E(v,v) \qquad \forall v \in V_0.$$

(ii) (*Boundedness of bilinear form*) There exists a positive constant $c_2 \in \mathbb{R}$ such that

$$|a_E(u, v)| \leq c_2 \|u\|_{1,\Omega} \|v\|_{1,\Omega} \qquad \forall u, v \in V_0.$$

(iii) (*Boundedness of right hand side*) There exists a positive constant $c_3 \in \mathbb{R}$ such that

$$|F(v)| \leq c_3 \|v\|_{1,\Omega} \qquad \forall v \in V_0.$$

Proof of the first property is illustrated in [68, p. 120], which can be shown to hold as a consequence of (2.1.3), the assumption of positive measure on the clamped boundary, and the use of Korn's second inequality [127, pp. 79 – 85]. The boundedness of the bilinear form holds as a consequence of the assumption of boundedness for the components of the elasticity tensor, and the boundedness of the right hand side follows due in part to (1.2.12), as well as an appropriate selection of body force to ensure continuity of the linear functional (2.2.4) on V_0. Properties (i) to (iii) are sufficient for the application of the Lax-Milgram Lemma, and as such ensure existence and uniqueness of solutions $u \in V_0$ to (2.2.6).

2.3 Finite Element Discretisation

A common approach to solving the system of equations presented in (2.2.6) is to discretise the system through the use of the Galerkin finite element method. For the interested reader, more information on the finite element method, as well as other methods that look to approximate (2.2.6) may be found in [127, pp. 233 – 272]. We begin by considering a discretisation of the domain Ω into simplices[I] $T_h = \{\mathcal{C}\}$ with maximum diameter h. By letting $P_\alpha(\mathcal{C})$ denote the space of polynomials in d variables of degree α defined on the set $\{\mathcal{C}\} \subset \mathbb{R}^d$, we define the finite dimensional space of piecewise polynomial

[I] Simplices are used here in order to retain generality. However, for the remainder of this thesis we will consider discretisations involving either rectangular or parallelpiped elements, based on d.

functions with respect to T_h as $V_h = [\mathcal{V}_h]^d$, where

$$\mathcal{V}_h = \left\{ v \;\middle|\; v \in C^0(\Omega), v_{|_{\mathcal{C}}} \in P_\alpha(\mathcal{C}) \;\; \forall \mathcal{C} \in T_h,\, v_{|_{\partial\Omega_D}} = 0 \right\} \subset H_0^1\left(\Omega\right), \tag{2.3.1}$$

is introduced as a finite element subspace of V_0 spanned by the set of basis functions $\boldsymbol{\psi}_h = \left(\boldsymbol{\psi}_h^{(c)}\right)_{c\in\mathbb{W}}$, where $\boldsymbol{\psi}_h^{(c)} := \left(\psi_1, \psi_2, \ldots, \psi_n\right)^T$. The subscript h denotes a discretisation parameter with the property $n = \mathcal{O}(h^{-d})$ as $h \to 0$, with n representing the number of finite element nodes.

We now define the following finite element Galerkin isomorphism as:

$$u_h = \sum_{j=1}^{n} \boldsymbol{\psi}_j u_j, \tag{2.3.2}$$

where the vector of unknowns $\mathbf{u}_h := \left(\mathbf{u}_h^{(c)}\right)_{c\in\mathbb{W}} = \left(u_j^{(c)}\right)_{j=1,\ldots,n}$ is mapped to the corresponding finite element functional $u_h := \left(u_h^{(c)}\right)_{c\in\mathbb{W}} \in V_h$. Each displacement node in the finite element discretisation of Ω will be viewed in terms of d components of the form $\left(\mathbf{u}^{(c)}\right)_{c\in\mathbb{W}}$, hence the length of the vector $\mathbf{u}_h$ will be $\hat{n} := dn$. For instance, for a finite element discretisation of a two dimensional domain, the first degree of freedom will have displacement components u_1 and u_2, with u_1 corresponding to horizontal displacement and u_2 corresponding to vertical displacement of the first degree of freedom.

Now, the finite element Galerkin solution to (2.2.6) can be viewed as the solution to (2.2.6) on the subspace V_h. We therefore look to find $u_h \in V_h$ such that:

$$a_E(u_h, v_h) = F(v_h) \qquad \forall v_h \in V_h. \tag{2.3.3}$$

By selecting an appropriate basis $\boldsymbol{\psi}_h$, direct substitution of (2.3.2) into the above provides the following

$$\sum_{j=1}^{\hat{n}} u_j \int_\Omega \boldsymbol{\epsilon}(\boldsymbol{\psi}_j) \colon E \colon \boldsymbol{\epsilon}(\boldsymbol{\psi}_l)\, \mathrm{d}\mathbf{x} = \int_{\partial\Omega_N} g\boldsymbol{\psi}_l\, \mathrm{d}s + \int_\Omega f\boldsymbol{\psi}_l\, \mathrm{d}\mathbf{x}.$$

We therefore look to determine $\mathbf{u} \in \mathbb{R}^{\hat{n}}$ satisfying

$$K\mathbf{u} = \mathbf{f}, \tag{2.3.4}$$

where the matrix $K = (k_{jl}) \in \mathbb{R}^{\hat{n}\times\hat{n}}$, with

$$k_{jl} := \int_\Omega \boldsymbol{\epsilon}(\boldsymbol{\psi}_j) \colon E \colon \boldsymbol{\epsilon}(\boldsymbol{\psi}_l)\,\mathrm{d}\mathbf{x} \qquad j, l = 1, \ldots, \hat{n},$$

and $\mathbf{f}_h = (\bar{f}_l) \in \mathbb{R}^{\hat{n}}$, where

$$\bar{f}_l := \int_{\partial\Omega_N} g\boldsymbol{\psi}_l\,\mathrm{d}s + \int_\Omega f\boldsymbol{\psi}_l\,\mathrm{d}\mathbf{x} \qquad l = 1, \ldots, \hat{n}.$$

The subscript h has been dropped in the above, and will also be omitted in subsequent equations to aid presentation. In this system of equations, the expression $\mathbf{f} \in \mathbb{R}^{\hat{n}}$ represents the corresponding discretisation of the load linear form. K represents the finite element stiffness matrix for the elasticity equation, and $\mathbf{u}$ the $\hat{n}$-vector of nodal values of the solution to the elasticity equations (2.1.1). As mentioned at the beginning of the chapter, a more in-depth introduction into linear elasticity may be found in [127]. The next chapter builds on the presentation given here in order provide a mathematical description for compliance minimisation problems in topology optimisation.

CHAPTER 3

STRUCTURAL OPTIMISATION

3.1 Introduction

This chapter provides a description and derivation of the underlying problem on which much of the work within this thesis is based, namely the field of topology optimisation. This area can be viewed as part of an important branch of computational mechanics known as structural optimisation, which, loosely speaking, involves the assembly of materials in the best way possible in order to withstand sustained loads. The ambiguous term 'best' is defined based on measures of structural performance, potentially involving factors such as displacement, geometry, stress, weight, compliance and/or stiffness (amongst others). Based on the chosen measure, we may wish to increase or decrease the particular quantity in order to obtain improvements to our final design. Therefore, a structural optimisation problem involves the maximisation or minimisation of one of these factors as an objective function, coupled with appropriate constraints related to other design criteria.

In addition to the objective function, elements key to all structural optimisation problems include the design variables and the state variables. The design variables correspond to a function or vector that describes the design, and can be modified during optimisation. For this thesis, the design variables will be used to describe the geometry, and may relate to aspects of the design including the area of a bar, a sparse set of surface points or the thickness of a sheet, amongst others.

Based on a given design, the state variables represent a function or vector that describes the state, or response of the structure. For mechanical design problems, the state could, for instance, correspond to the stress, strain, displacement or force.

Evidence of work involving the optimisation of structures can be seen as early as the eighteenth century, where one-dimensional problems were examined by Euler and Lagrange. Both considered problems involving the design of elastic columns requiring optimal cross sectional areas, as shown in [53] and [101] respectively. Additionally, work by Euler [52, pp. 299 – 316] eventually led to the involute gear profile, which today has widespread application in modern industry.

By the turn of the twentieth century, a mathematical description for structures with minimum weight and prescribed stress constraints was provided by Michell [121]. In the work, a number of different design domains were considered, and the resulting structures (termed Michell structures) were later shown by Save and Prager in 1985 [163] to have minimum compliance for an associated structure with corresponding volume. Therefore, the structures could be seen to agree with global optima to compliance minimisation problems, subject to bounds on the amount of available material.

Based on the type of geometric feature represented by the design variables, the field of structural optimisation may be considered in terms of three branches, namely sizing, shape and topology optimisation. The subsequent sections aim to provide an overview of each.

3.2 Sizing Optimisation

A general sizing, or truss, optimisation problem involves the optimisation of a typical size of a structure. For example, the goal may involve finding the optimal thickness distribution of a sheet, or the optimal cross sectional areas in a truss structure (as previously mentioned). In such a problem, the optimal thickness distribution would minimise or maximise a physical quantity such as the stiffness, compliance or deflection, for instance, while ensuring that equilibrium and (possibly) other constraints on the state and design variables are fulfilled. Here, the thickness of the plate corresponds to the design variable and the deflection could, for example, correspond to the state variable. General sizing problems can be quite particular and lack real generality; both the domain of the design and the state variables are known and fixed. However their formulation and implementation is typically straight forward. A fair amount of development and study into the topology design of truss structures has been conducted already, with [16, pp. 221 – 259] describing a number of important contributions to the field, as well as the underlying mathematical model involved.

3.3 Shape Optimisation

In contrast, a general shape optimisation problem aims to locate the optimal shape of the domain without changing the topology. Here, the focus of the optimisation process is the geometry, with the domain, or contour of a select part of the boundary being the design variable. The connectivity of the structure remains unaltered, meaning that no new boundaries are formed during the optimisation process. Therefore, solutions are typically obtained through the movement of boundaries from an initial trial configuration in order to provide improvements to the objective function value while satisfying relevant constraints on the design variables. When compared to sizing optimisation, shape optimisation affords greater generality. However, the formulation of typical problems is less straight forward. A detailed discussion into different approaches in shape optimisation can be found in [67].

3.4 Topology Optimisation

Topology optimisation of solid structures involves the treatment of both the shape and the connectivity of the domain as design variables. Here, features such as the shape and location of holes, as well as the connectivity of the structure are determined. The distinguishing feature separating this approach from shape optimisation involves the introduction of new boundaries, allowing for the consideration of a broader range of feasible solutions. As a result, a much wider range of structural domains can be considered when compared to both sizing and shape optimisation. The resulting topology can then be transferred as an initial guess for shape optimisation after undergoing post-processing and modification.

Subject to a given amount of material, boundary conditions and external loading, the aim of topology optimisation is to determine the optimal distribution of material in order to maximise structural stiffness (minimise compliance). The field was initially investigated with the intention of being applied to the design of mechanical structures [14]. Since then, it has become a popular choice for the systematic computation of innovative designs in a wide range of engineering disciplines, having being applied to the design of materials [22, 107, 173], Micro Electro Mechanical Systems (MEMS) [119, 161, 169], mechanisms [30, 168, 205] and other complex structural design problems including a reduction in weight of approximately 1 tonne per aircraft for the Airbus A380 [96]. A driving factor behind these developments is due in part to the flexible parameterisation of the design space, which allows optimisation algorithms to explore it efficiently.

In the context of production and manufacturing, the use of topology optimisation offers the chance to study a proposed design at the concept level of the design process. Such a proposal can then be fine tuned based on relevant criteria to aid production and improve performance. This process is beneficial, as it replaces time consuming and potentially costly design iterations. Therefore, topology optimisation can not only be seen to reduce design development time and overall cost, but also provide improvement to the overall performance of final designs. Furthermore, the integration of topology optimisation into existing computational codes is relatively straightforward, and additional sensitivity analysis evaluations, namely the change in the objective function and/or constraints on the state variables with respect to the design variables, are relatively simple and efficient to compute.

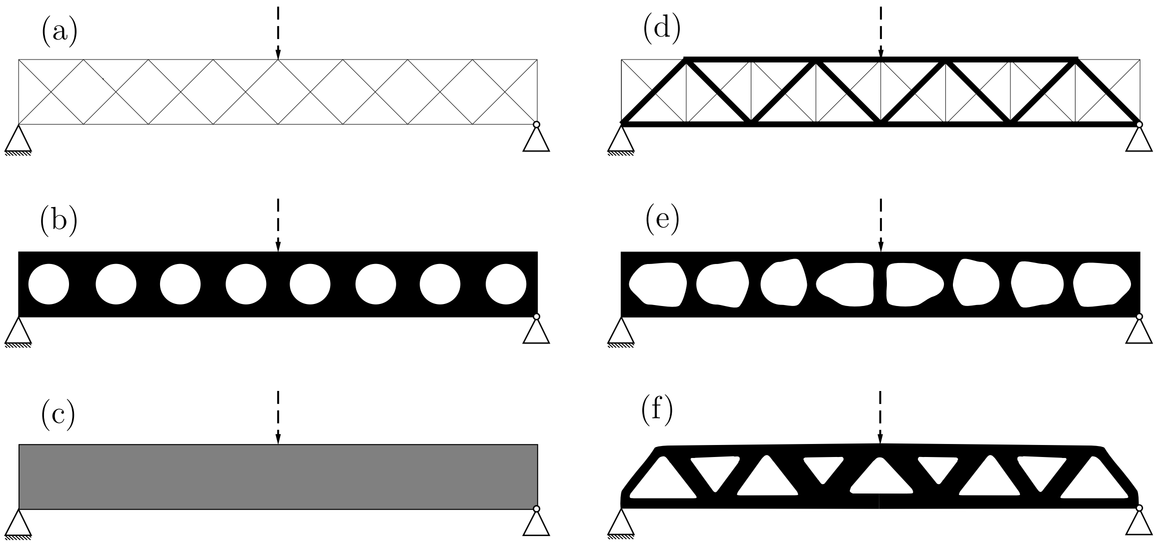


Figure 3.1: Illustrations of structural optimisation for a Messerschmitt-Bölkow-Blohm (MBB) beam. Each of (a), (b) and (c) represent initial designs for sizing, shape and topology optimisation, respectively. Illustrated on the right by (d), (e) and (f) are the corresponding optimal designs based on the loads and support data. In each of the above, the roller support illustrated in the right corner prohibits vertical movement, whereas the fixed support in the left corner prohibits both horizontal and vertical movement.

One of the major computational breakthroughs in the field came in 1988, with the application of the so-called microstructure/homogenisation approach to structural optimisation by Bendsøe and Kikuchi [14]. The theory behind this approach involves limiting arguments, aiming to target oscillatory coefficients within the underlying Partial Differential Equations (PDEs). Through an asymptotic expansion and the assumption of periodicity, this is achieved by considering alternative differential equations involving damped coefficients in such a way that the solution to the adjusted system bears reasonable representation to that of the original.

Extensions of this work to a broader range of problems were later considered by Suzuki and Kikuchi in 1991 [181]. Further use of the homogenisation approach can also be found in a number of other sources including [114, 188], as well as an industrial application by Larsen, Sigmund and Bouwstra [103] involving the design and fabrication of material structures and compliant mechanisms with negative Poisson's ratio. Homogenisation approaches have more recently given way to alternative methods, however a review of the application of homogenisation to structural optimisation, with particular focus on topology optimisation can be found in [69, 70, 71].

More recent work has involved an alternative interpolation strategy initially presented in 1989, where Bendsøe [13] introduced the *artificial density approach*, which would later became known by the more familiar moniker Solid Isotropic Material with Penalisation (SIMP) approach. The method here involves the approximation of a discrete solution by a continuous counterpart, coupled with the use of an appropriate penalisation power in order to steer areas of intermediate density towards either extreme. Similar approaches were proposed by Zhou and Rozvany [208] in 1991 and Mlejnek [123] in 1992.

Initially, this approach did not see widespread use, due to issues including mesh dependence, and also the physical interpretation of intermediate material (hence the term *artificial* above). However, a paper produced in 1999 by Bendsøe and Sigmund [15] on material interpolation schemes provided physical justification for the proposed penalisation approach through the use of the inverse homogenisation method. This will not be discussed here but can be found, for instance, in [167]. Since then, the SIMP approach has received a relatively general acceptance within the structural optimisation community, forming the basis of a number of publications during the last fifteen years.

The ease at which this approach can be implemented computationally is illustrated to good effect by Sigmund in [170], where an open source 99-line code is provided for use in MATLAB. Further developments have seen the number of lines required shrink further [4]. The use of SIMP interpolation will be discussed further in Section 3.5; for the interested reader, Rozvany [149] provides an account of various works contributing to the development of the SIMP approach for structural optimisation, including a comparison with other solution strategies not discussed in this work. Alternative interpolation methods have been considered by a number of authors, with examples found in [140, 179, 185, 186].

A recent monograph produced by Bendsøe and Sigmund [16] for the field of topology optimisation provides an insight into relevant findings and applications, and will be referred to throughout the course of this thesis. The next section will present and describe the mathematical background behind minimum compliance problems in topology optimisation.

3.5 Mathematical Derivation

3.5.1 Weak Formulation

The mathematical derivation of minimum compliance problems in topology optimisation bears similarities to the presentation of linear elasticity discussed in Section 2. Here, we also begin with a domain as depicted in Figure 2.1, where the aim is to establish the optimal layout of material. We look to determine the optimal elasticity tensor

$$\hat{E}_{ijkl}(\mathbf{x}) : \Omega \to \mathbb{R} \qquad (i,j,k,l) \in \{1,\dots,d\}^4,$$

for a given set of acceptable elasticity tensors $\hat{E}^{\mathrm{ad}}$ satisfying standard conditions as described in both (2.1.3a) and (2.1.3b). We also impose uniform positive semi-definiteness, which can be viewed as a relaxation of (2.1.3c) to encompass a non-negative (as opposed to positive) constant.

Using Section 2.2, the variational formulation of the minimum compliance (or maximum global stiffness) problem can be described in the following manner

$$\begin{aligned} &\min_{u,E} && F(u) \quad (= a_E(u,u)) && && \text{(surface traction)} && (3.5.1\text{a})\\ &\text{subject to:} && a_E(u,v) = F(v) && \forall v \in V_0, && \text{(linear elasticity)} && (3.5.1\text{b})\\ & && \hat{E}(\mathbf{x}) \in \hat{E}^{\mathrm{ad}}. && && && \end{aligned}$$

In the above, the objective function arises from setting the compliance $F(u)$ equal to the energy $a_{\hat{E}}(u,u)$. As in the case of linear elasticity, $F(\cdot) : V_0 \to \mathbb{R}$ represents the load linear form, or compliance, and is presented in (2.2.4). The bilinear form $a_{\hat{E}}(\cdot,\cdot) : V_0 \times V_0 \to \mathbb{R}$ is as given in (2.2.5). The relaxation of (2.1.3c) as described above permits regions of void material within a design. Consequently, the problem as formulated is ill-posed - the effects of which will be discussed in Section 3.5.3. In the case where (2.1.3c) is explicitly imposed, both existence and uniqueness of solutions can be guaranteed by proving the so-called V_0-ellipticity of the bilinear form $a_{\hat{E}}(\cdot,\cdot)$ as a consequence of the Lax-Milgram Lemma presented in Lemma 2.2.1. This is achieved in exactly the same manner as in the case of linear elasticity, and the discussion towards the end of Section 2.2 can be applied here.

3.5.2 Finite Element Formulation

A common approach to solving the system of equations (3.5.1) is to discretise the system using finite elements. We therefore consider a finite element approximation to the variational equation (3.5.1b) via a Galerkin scheme in a similar manner to the presentation in Section 2.3. It is standard to assume for two dimensional problems in topology optimisation that the set Ω is discretised using four-noded rectangles (for three dimensions, the discretisation would typically involve eight-noded parallelepipeds). Therefore, we assume that the elasticity tensor is discretised into m components E_e, $e = 1, \ldots, m$ using piecewise constant basis functions. We now consider the resulting finite dimensional nonlinear programming problem of the form:

$$\begin{aligned} &\min_{\mathbf{u},E} && \mathbf{f}^T\mathbf{u} \\ &\text{subject to:} && K(E)\mathbf{u} = \mathbf{f}, \\ & && E(\mathbf{x}) \in E^{\text{ad}}. \end{aligned} \tag{3.5.2}$$

In this system of equations, the expression $\mathbf{f} \in \mathbb{R}^{\hat{n}}$ represents the corresponding discretisation of the load linear form, where the set E^{ad} corresponds to a discretisation of the admissible set $\hat{E}^{\text{ad}}$. The matrix $K(E)$ represents the finite element stiffness matrix for the elasticity equations, where $K(E) := \sum_{e=1}^{m} K_e(E_e)$, $K_e(E_e) \in \mathbb{R}^{(\hat{n},\hat{n})}$ with K_e denoting the global level element stiffness matrix. Finally, $\mathbf{u}$ denotes the $\hat{n}$-vector of nodal values of the solution to the elasticity equations (3.5.1b).

3.5.3 Solid Isotropic Material with Penalisation (SIMP)

Based on the underlying topology, different choices for the set $\hat{E}^{\mathrm{ad}}$ may be considered. In particular, we distinguish between elements of the topology as follows

- Solid: Filled with a single material.
- Void: No material.
- Porous: As single material and void.
- Composite: Multiple materials with no void region.
- Composite-Porous: Multiple materials with void region.

For our work, we consider a fairly straightforward description of a topology consisting of regions of either void[I] or a single solid isotropic material. Therefore, we look to describe the set of stiffness tensors where solid material exists for a given subset $\Omega^{0,1}$ of Ω. Our geometric representation of a structure can then be interpreted in a similar manner to the binary rendering of an image, with areas of black corresponding to solid regions and those of white corresponding to void. In this instance, we may express the limit of available material as $\int_{\Omega^{0,1}} 1 \,\mathrm{d}\Omega \leq V_{\mathrm{vol}}$, with V_{vol} denoting the volume, or amount of material at our disposal. The aim is therefore to obtain an optimal subset of material points $\Omega^{0,1}$. One way to view this mathematically is to describe the set of admissible stiffness tensors $\hat{E}^{\mathrm{ad}}$ in the following way

$$\hat{E}^{\mathrm{ad}} := \left\{ \hat{E}(\mathbf{x}) \;\middle|\; \hat{E}(\mathbf{x}) = q(\mathbf{x})\bar{E} \right\}, \tag{3.5.3a}$$

$$\int_{\Omega} q(\mathbf{x}) \,\mathrm{d}\Omega = \mathrm{Vol}(\Omega^{0,1}) \leq V_{\mathrm{vol}}, \qquad q(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in \Omega^{0,1}, \\ 0 & \text{if } \mathbf{x} \in \Omega \backslash \Omega^{0,1}. \end{cases} \tag{3.5.3b}$$

[I] Permitted within our formulation as a consequence of the relaxation described in Section 3.5.1.

This representation is similar to the presentation in [16, p. 5]. Here, $\Omega^{0,1}$ defines the set of stiffness tensors that retain the material properties of the initial design. The constant elasticity tensor $\bar{E}$ represents the material properties for a given homogenous and isotropic material satisfying the criteria presented in (2.1.3). By defining $\hat{E}^{\mathrm{ad}}$ in this manner, we guarantee that every $\hat{E}(\mathbf{x}) \in L^{\infty}(\Omega)$. Through the definition of the discrete indicator function q (corresponding to plate stiffness), it is clear that the weak formulation (3.5.1) with $\hat{E}^{\mathrm{ad}}$ defined as per (3.5.3) corresponds to a distributed, discrete valued design problem, commonly referred to as a $0-1$ problem. Nevertheless, the weak formulation (3.5.1) coupled with (3.5.3) results in an ill-posed problem [89, 90, 91], with issues in general including mesh dependence and nonexistence of solutions. In order to work around this issue, one could consider solving the associated discretised problem coupled with heuristic rules. Whilst this approach can be achieved for a relatively cheap computational cost, a precise definition of the problem posed at the continuous level is unclear.

Alternatively, modifications to the problem described through a combination of the weak formulation (3.5.1) coupled with (3.5.3) can be considered in order to ensure existence of solutions, along with the development of solution methods for the resulting discretised problem. Typical modifications involve relaxation of the original problem, and thus an enlargement of the feasible set. Approaches aimed at solving the formulation of the problem as presented until now involve the approximation of the discrete indicator function q by a continuous function ρ (corresponding to the density) defined on $[0, 1]$, allowing for the use of gradient based optimisation algorithms. For instance, the homogenisation approach within topology optimisation mentioned previously involves the consideration of material properties at two different levels. At the microscopic level, composites consisting of either full ($q(\mathbf{x}) = 1$) or void ($q(\mathbf{x}) = 0$) material are considered, whereas at the macroscopic level a continuous function is used along with an appropriate elasticity tensor. The global behaviour of the structure can be realised in the limit as $\boldsymbol{\epsilon}$ tends to zero, with $\boldsymbol{\epsilon}$ denoting the width of a macroscopic element.

From the formulation of the problem as presented up until this point, a relaxation of the discrete function q described in (3.5.3) suggests that the stiffness matrix presented in (3.5.2) can be seen to depend continuously on the density of material, and can thus viewed in the same manner as a sizing problem on a fixed domain. The obvious issue with such an approach is that the resulting solution will inevitably contain areas of intermediate density, and therefore some form of penalisation should be used in order to distribute such areas towards either solid or void regions. This is achieved by raising the continuous function ρ to a power $\mu > 1$, where the relationship between the stiffness (s) and thickness (ρ) of the plate corresponding to $s(\mathbf{x}) = \rho(\mathbf{x})^\mu$ is as illustrated in Figure 3.2 for different values of μ. As mentioned previously in Section 3.4, this approach corresponds to the SIMP interpolation scheme introduced by Bendsøe [13], which can be described mathematically using (3.5.3) in the following way

$$\hat{E}(\mathbf{x}) = \rho(\mathbf{x})^\mu \bar{E}, \quad \mu \geq 1, \quad \rho \in \hat{I}, \tag{3.5.4a}$$

$$\hat{I} := \left\{ \rho \in L^\infty(\Omega) \;\middle|\; 0 < \rho_{low} \leq \rho(\mathbf{x}) \leq \rho_{high} \text{ a.e in } \Omega, \int_\Omega \rho(\mathbf{x}) \, \mathrm{d}\Omega \leq V_{\text{vol}} \right\}. \tag{3.5.4b}$$

The set (3.5.4b) is similar to the set of permissible designs as printed in [24], where it is assumed that the parameters involved with the set $\hat{I}$ are chosen in such a way that the set is nonempty. In the above system, the volume of the structure is evaluated as per (3.5.4b), and so ρ can be viewed as the density of material. In this case, the density of material $\rho(\mathbf{x})$ is the design function, interpolating between the material properties of approximately 0 and $\bar{E}_{ijkl}$.

The reason for the approximation of zero material is due to the introduction of a lower bound on the density of material, namely ρ_{low}, in order to preserve the non-singularity of the stiffness matrix. Such an approach is referred to in the literature as *weak material approximation*. The choice of lower bound was suggested in [16, p. 10] to be 10^{-3}, however the topic remained ambiguous at the time as it was typically found that resulting solutions to compliance minimisation problems were not overly sensitive to the quantity. Therefore

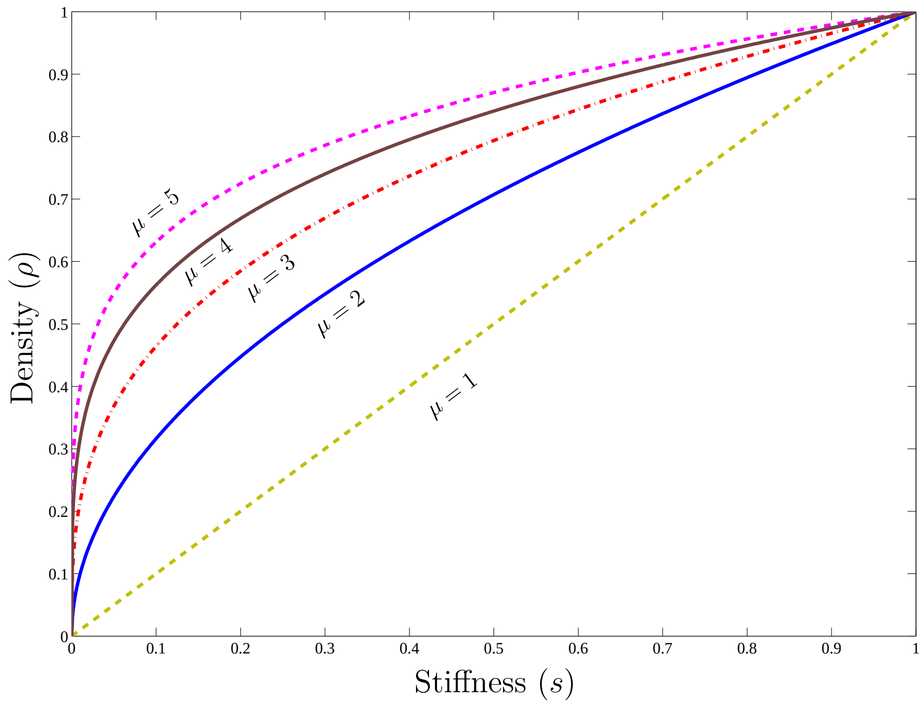


Figure 3.2: Illustration of the effects of penalisation.

no further work was conducted into the effects of different values for ρ_{low}. However, a more recent analytical study into the impact of this lower bound on the resulting solution has been carried out by Berggren and Kasolis in [17], where an appropriate preconditioning strategy is proposed in such a way that the limit of the resulting preconditioned stiffness matrix as the lower bound tends to zero can be seen to result in a stiffness matrix corresponding to an elastically extended finite element problem, encompassing potential areas of void material within the domain.

In addition to computational efficiency, the simplicity and robustness of this approach make it desirable when compared to other solution strategies. However, the use of a strict penalisation power (namely $\mu > 1$) fails to deal with the aforementioned ill-posedness within the original formulation. Consequently, solutions may or may not exist, and where a solution is found, it can be seen to depend on the size of the penalisation power with well known issues such as chequerboarding common. This not only presents a serious theoretical drawback, but also carries the implication that computational results become sensitive to the mesh parameter h, leading to mesh dependent designs.

Work by Stolpe and Svanberg [180] affirms these issues. Here, a continuation approach is considered based on gradual increments of the penalisation power, using convexity in the absence of penalisation (namely $\mu = 1$) in order to provide a suitable initial design. However, examples were provided illustrating cases where solutions could be obtained involving areas of intermediate density, independently of the chosen penalty parameter. Reitz in [147] was able to provide globally optimal discrete solutions to minimum compliance problems under certain conditions for sufficiently large μ, provided that the volume constraint remained consistent with such a design. Work by Martinez [118] also displays existence of solutions under suitable assumptions regarding the penalty function.

In order to correct this problem, a global or local restriction must be placed on the density. Using [16, p. 30], these restrictions fall into three categories. We can either reduce the parameter space for the designs, add constraints to the optimisation problem or apply filters in the optimisation process. An account of some of the common restrictions used can be found in [16, pp. 31 – 47].

3.5.4 Variational Formulation - SIMP Approach

We now return to the variational formulation as defined in (3.5.1a) - (3.5.1b). We look to minimise the compliance subject to the volume of the optimal body, V_{vol} being a known quantity. The weak formulation is then transformed to:

$$\begin{aligned}
&\min_{u,\rho} && F(u) \quad (= a(u,u)) && && \text{(surface traction)} && (3.5.5a)\\
&\text{subject to:} && a_\rho(u,v) = F(v) && \forall v \in V_0, && \text{(linear elasticity)} && (3.5.5b)\\
& && \int_\Omega \rho(\mathbf{x})\,\mathrm{d}\Omega \le V_{\text{vol}}, && && \text{(volume constraint)} && (3.5.5c)\\
& && 0 < \rho_{low} \le \rho(\mathbf{x}) \le \rho_{high} && \forall \mathbf{x} \in \Omega. && && (3.5.5d)
\end{aligned}$$

In the above,

$$a_\rho(u,v) := \int_\Omega \rho(\mathbf{x})^\mu \left(\boldsymbol{\epsilon}(u) : \bar{E} : \boldsymbol{\epsilon}(v)\right) \mathrm{d}\mathbf{x}, \qquad (3.5.6)$$

where $u = u(\rho)$.

3.5.5 Finite Element Formulation - SIMP Approach

From this formulation, we now consider a Galerkin finite element discretisation in a similar manner to the presentation for the system (3.5.1a) to (3.5.1b). In this situation, the density function ρ is piecewise constant, with ρ_e representing the density for rectangle e (or cuboid e in three dimensions). Therefore, we arrive at the following finite dimensional nonlinear programming problem of the form:

$$\begin{aligned} &\min_{\mathbf{u},\boldsymbol{\rho}} && \mathbf{f}^T\mathbf{u} && (3.5.7)\\ &\text{subject to:} && K(\boldsymbol{\rho})\mathbf{u} = \mathbf{f}, \\ &&& \sum_{e\in D} \rho_e \leq V_{\text{vol}}, \\ &&& 0 < \underline{\rho} \leq \rho_e \leq \overline{\rho} && \forall e \in D. \end{aligned}$$

In this system of equations, we assume that the vector of coefficients $\boldsymbol{\rho}$ is piecewise constant in Ω, so that $\rho(\mathbf{x}) = \rho_e$ for each $e \in D = \{1, 2, \ldots, m\}$, with upper and lower limits denoted ρ_{high} and ρ_{low} respectively. The $\hat{n}$-vector of nodal values $\mathbf{u} = \mathbf{u}(\boldsymbol{\rho})$ denotes the solution to the elasticity equations (3.5.5b). As before, the expression $\mathbf{f} \in \mathbb{R}^{\hat{n}}$ represents the corresponding discretisation of the load linear form and the summation is a discretisation of the integral present in the volume constraint. The matrix $K(\boldsymbol{\rho}) \in \mathbb{R}^{(\hat{n},\hat{n})}$ represents the finite element stiffness matrix for the elasticity equations which is now dependent on the density of material, where $K(\boldsymbol{\rho}) = \sum_{e=1}^{m} \rho_e^{\mu} K_e$. An illustration of this can be seen by using the definition of $a_\rho(u, v)$ provided in (3.5.6), where we assume $\boldsymbol{\rho}$ to be piecewise constant to leave

$$a_\rho(u, v) := \sum_{e=1}^{m} \rho_e^{\mu} \int_{\Omega_e} \boldsymbol{\epsilon}(u) : \bar{E} : \boldsymbol{\epsilon}(v)\, \mathrm{d}\mathbf{x} =: \sum_{e=1}^{m} \rho_e^{\mu} a_e(u, v),$$

with Ω_e denoting the restriction of Ω to the e^{th} simplex.

3.5.6 The Variable Thickness Sheet (VTS) Problem

By setting $\mu = 1$ in the SIMP description (3.5.4a), we arrive at the well known variable thickness sheet design problem (VTS) as discussed in [16, pp. 54 – 57]. Here, we consider the situation where the density of each element is allowed to attain values between $\underline{\rho}$ and $\overline{\rho}$, as originally presented by Rossow and Taylor in 1973 [148]. The density $\boldsymbol{\rho}$ in this case can then be viewed as the thickness of a sheet.

Using the formulation of the minimum compliance problem presented in (3.5.7), the VTS problem can be expressed by the following nonlinear optimisation problem

$$
\begin{aligned}
\min_{\mathbf{u},\boldsymbol{\rho}} \quad & \mathbf{f}^T\mathbf{u} \\
\text{subject to:} \quad & K(\boldsymbol{\rho})\mathbf{u} = \mathbf{f}, \\
& \sum_{e \in D} \rho_e \leq V_{\text{vol}}, \\
& K(\boldsymbol{\rho}) = \sum_{e=1}^{m} \rho_e K_e \\
& 0 \leq \underline{\rho} \leq \rho_e \leq \overline{\rho} \qquad \forall e \in D,
\end{aligned}
\tag{3.5.8}
$$

where the variables $\mathbf{f}$, $\mathbf{u}$, $\boldsymbol{\rho}$, K, $\underline{\rho}$ and $\overline{\rho}$ are defined in an analogous fashion to (3.5.7). It should be noted that the VTS design problem bears similarities to truss design problems. This is due to the fact that both formulations involve the structural linear dependence of both the stiffness and volume on the design variables. One can therefore use algorithms that have been developed for truss topology design problems in order to consider the solution to (3.5.8). A discussion of effective solution methods and algorithms for truss topology design problems is presented in [16, pp. 221 – 259], where (by construction) it is not a requirement to enforce a strict positive lower bound on the discretised density values. This property is noteworthy due to the fact that the solution is not forced towards a discrete $0-1$ final design, allowing for the optimal layout of the sheet to be determined without being concerned about whether the lower bound on the density has been selected appropriately.

Using observations from the objective function presented in (3.5.1a), the compliance $\mathbf{f}^T\mathbf{u}$ can be set equal to the energy. This is obtained by substituting the first constraint into the objective function to give $\mathbf{u}^T K(\boldsymbol{\rho})\mathbf{u}$. In fact, one can use the equilibrium equations of the discretised system (3.5.7) to eliminate $\mathbf{u}$ by writing $\mathbf{u} = K^{-1}(\boldsymbol{\rho})\mathbf{f}$ provided that a strict positive lower bound is enforced. We then arrive at the so-called nested formulation as follows

$$\begin{aligned} &\min_{\rho} && \|\mathbf{f}\|^2_{K^{-1}(\boldsymbol{\rho})} \\ &\text{subject to:} && \sum_{e\in D} \rho_e \leq V_{\text{vol}}, \\ & && 0 < \underline{\rho} \leq \rho_e \leq \overline{\rho} \qquad\qquad \forall e \in D, \end{aligned}$$

where we use the fact that $\|\mathbf{f}\|^2_{K^{-1}(\boldsymbol{\rho})} = \mathbf{f}^T K^{-1}(\boldsymbol{\rho})\mathbf{f}$. The reasoning behind the reformulation of (3.5.8) in this manner is that the objective function is now convex[I]. Adding to this the fact that the set $\hat{I}$ can be formulated in such a way that, for a suitable choice of ρ_{low}, the discretised form of the set $\hat{I}$, namely:

$$I = \left\{ e \in D \;\middle|\; 0 < \underline{\rho} \leq \ \rho_e \leq \overline{\rho}\ , \sum_{e=1}^{m} \rho_e \leq V_{\text{vol}} \right\},$$

is nonempty, convex and compact, we find that (3.5.8) can be written as a nonlinear convex optimisation problem. We therefore expect local and global optima to coincide (see Theorem 1.2.1). In particular, existence of solutions can be shown by using the fact that the stiffness is linear with respect to the density ρ in the case $\mu = 1$. For the interested reader, the proof is illustrated in [16, pp. 272 – 274]. Therefore, by using [172], we expect optimal designs to be determined independently of the chosen mesh parameter

[I] This is a consequence of the positive definiteness of the matrix $K^{-1}(\boldsymbol{\rho})$.

without the need to introduce restriction methods such as a perimeter control, as required in the more general $\mu > 1$ setting.

The variable thickness sheet problem will be the focus of this thesis, however for the moment work will be concentrated on $\mu \geq 1$ in the SIMP description in order to establish results for general cases. Now that the basic foundations and background of topology optimisation have been presented, we now describe three model problems that will appear during this thesis, followed by a consideration of typical solution methods used to solve such problems. Based on this presentation, suggestions for potential areas of improvement will lead to the development of solution methods that aim to provide results in a more effective manner.

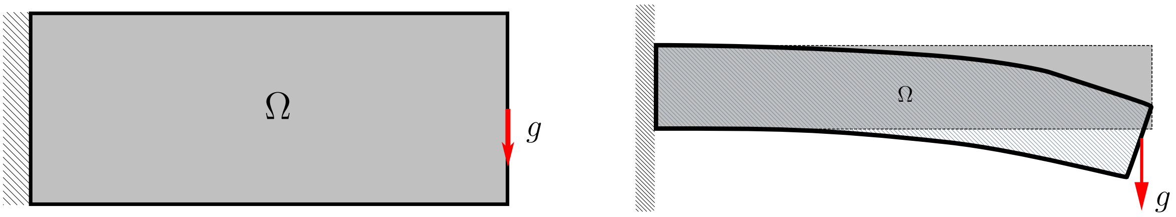
3.6 Model Problems

Within this section, an overview of three model problems will be provided and used to produce numerical results throughout the course of this thesis. Displacement profiles are illustrated for each model problem based on zero body force and boundary traction $\mathbf{g} = \mathbf{1}$.

3.6.1 Cantilever Beams

Cantilever beams occur in a wide range of areas including construction and industry, and can be found, for example, in the design of bridges and buildings. Such problems are generally straight forward to construct, and can be discretised using a regular finite element mesh. Therefore, they are an ideal candidate to consider as a set of test problems.

By construction, cantilevers are assumed to be fixed at one end with a load (or loads) acting at some point (or points) in the domain, which are resisted by moment and shear stress. Loads can be applied in multiple ways, with the most straightforward being a single load at the opposing end to the fixed edge. However, they can also be applied at an intermediate point, or can be spread across the beam, possibly unevenly and with differing



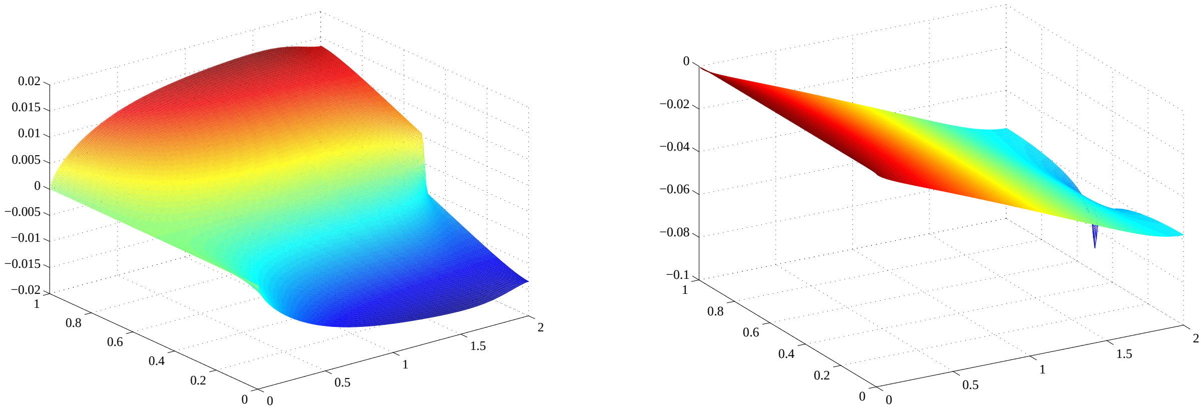
(a) Layout of the cantilever beam problem

(b) Deflection of the beam

Figure 3.3: Illustration of a typical cantilever beam, with clamped left hand edge and force applied on the right as depicted in (a). The corresponding deflection, typically occurring along the entire length of the beam, is illustrated in (b).

force. In general, a fixed support is achieved by the application of homogeneous Dirichlet conditions as illustrated in Figure 3.3(a), however extensions can also be considered. For instance, the use of Robin boundary conditions allows for the modelling of an elastic spring. For the purpose of this thesis, the problem illustrated in Figure 3.3(a) posed on a domain of size $(0, 2) \times (0, 1)$ will be considered within this thesis.

Illustrated in Figure 3.3(b) is the typical deflection of the beam, which corresponds to the degree at which the structural element is displaced under a load. The resulting displacement profile in both horizontal and vertical directions is displayed in Figure 3.4. Typically, the deflection of the beam can be quite involved, however under the assumption that relatively small deflections occur in a suitable neighbourhood of the beam, it is



(a) Horizontal displacement

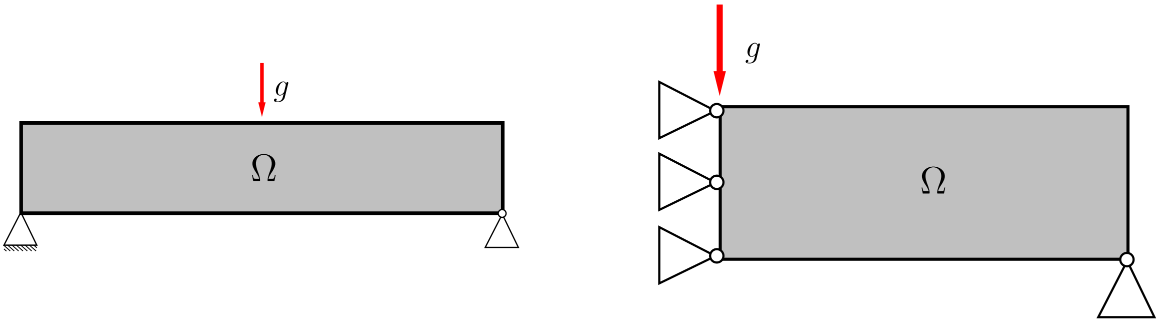
(b) Vertical displacement

Figure 3.4: Displacement profiles for the optimal layout of the cantilever beam problem. The vertical axis describes the displacement, with the remaining axes representing the respective coordinates of the domain.

possible to approximate the deflection using a series of circles.

A mathematical presentation will not be discussed here, however for the interested reader [156], provides a useful insight and mathematical derivation for the example problem discussed.

3.6.2 MBB Beam

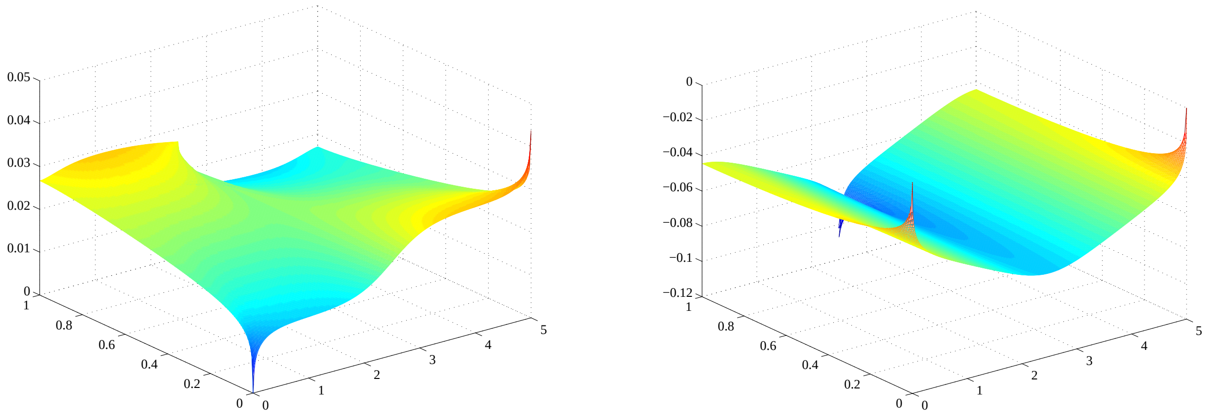


(a) Original domain

(b) Reduced domain

Figure 3.5: Illustration of MBB beam problem. The original problem is displayed in (a), with a roller support in the left corner and a fixed support in the right corner. Downward traction is applied in the centre of the top edge. Through symmetry, the problem may be reduced to the domain illustrated in (b), with the original solution recovered via reflection in the left edge.

The Messerschmitt-Bölkow-Blohm (MBB) beam also represents a commonly used example within the field of topology optimisation. For this work, a rectangular domain of size $(0, 5) \times (0, 1)$ is considered. Clamping is applied in the left corner, with a roller support in the right corner and a force at the top of the rectangular domain as illustrated in Figure 3.5(a). The resulting displacement profile in both horizontal and vertical directions is displayed in Figure 3.6. Symmetry within the domain may be exploited, leading to a reduced design domain as depicted in Figure 3.5(b), with the original solution recovered through reflection along the left edge of the optimal layout to the reduced problem. For this work, we choose not to exploit symmetrical patterns within our model problems in order to test the performance of our solution methods on the full problem.



(a) Horizontal displacement

(b) Vertical displacement

Figure 3.6: Displacement profiles for the optimal layout of the MBB beam problem. The vertical axis describes the displacement, with the remaining axes representing the respective coordinates of the domain.

3.6.3 Rotating Plate

In this example, a square of side $(0, 1) \times (0, 1)$ is considered, with a square hole of width 1/4 located in the center. Clamping is enforced on each of the four faces of the hole by imposing homogeneous Dirichlet boundary conditions, with traction applied on each edge of the outer boundary as depicted in Figure 3.7 so that the domain is subject to clockwise rotation as illustrated in Figure 3.8. Such a configuration can be used in order

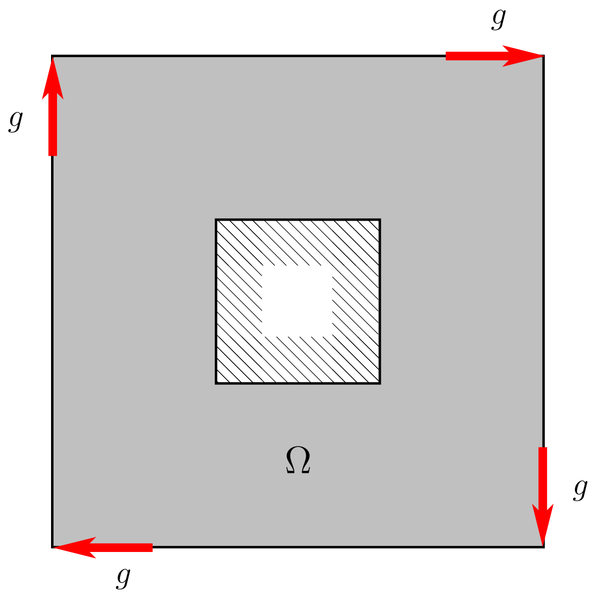


Figure 3.7: Illustration of the domain for the rotating plate example.

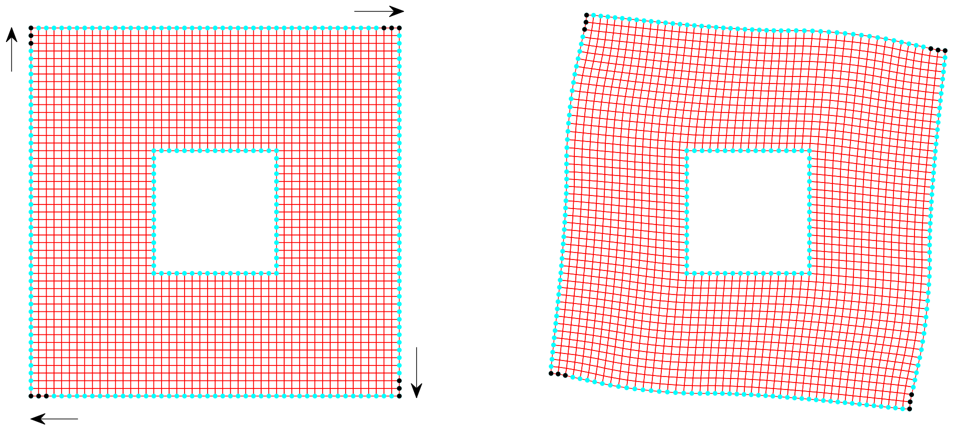
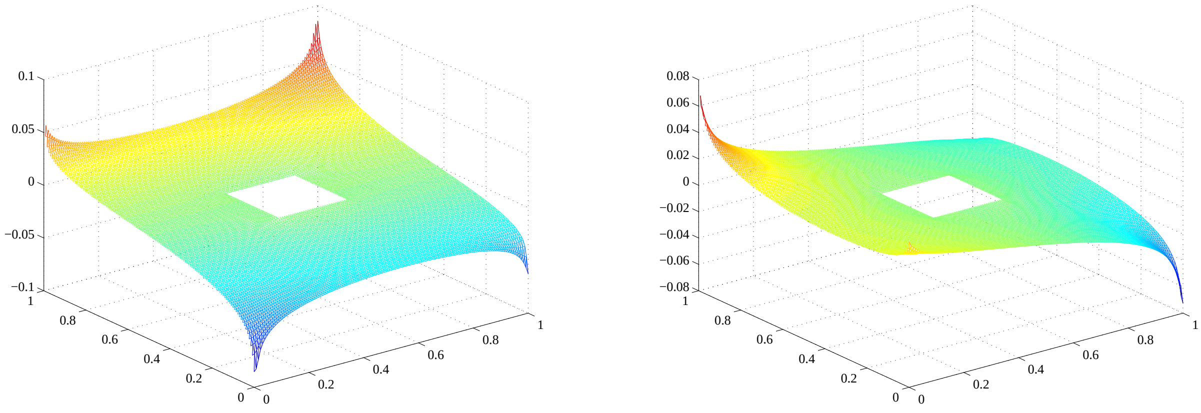


Figure 3.8: Finite element mesh for this example, with the deflection illustrated on the right. The red dots indicate nodal elements, with blue dots describing the boundary and black dots signifying the application of traction.

to produce optimal designs for wheel rims fitted to automotive vehicles such as cars and motorbikes. The displacement of the final design in both horizontal and vertical directions is illustrated in Figure 3.9.



(a) Horizontal displacement

(b) Vertical displacement

Figure 3.9: Displacement profiles for the optimal layout of the rotating plate problem. The vertical axis describes the displacement, with the remaining axes representing the respective coordinates of the domain.

CHAPTER 4

REVIEW OF SOLUTION METHODS FOR TOPOLOGY OPTIMISATION

This chapter will describe a number of approaches used to obtain solutions to minimum compliance problems as formulated in (3.5.8).

4.1 Derivative Free Approaches

Solution methods that obtain optima without the use of derivative information of the involved functions come under the heading of derivative free approaches. In 1961, Hooke and Jeeves [73] described the notion of direct search, namely the sequential generation and examination of trial solutions through an appropriately defined strategy. Whilst classical works related to direct search methods provided no assertions of either termination or convergence to stationary points, more recent works [189, 190] have addressed such issues, as well as proving other related (and useful) properties of such approaches. These results, along with advantages including flexibility, ease of use and simplicity mean that these classical methods still remain popular today.

The first simplex-based optimisation algorithm was proposed by Spendley, Hext and Himsworth in 1962 [177], followed in 1965 by the Nelder-Mead algorithm [128]. Based on an initial set of points forming a simplex, this approach determines the worst current point (of the simplex) at each iteration, and attempts to replace the point through the introduction of a new vertex that leads to the creation of a new simplex.

The Hooke-Jeeves and Nelder-Mead algorithms saw extensive use during the 1960s and 1970s. Both of these approaches can be viewed as deterministic in the sense that they do not require use of random search steps. Stochastic algorithms, which do require use of random search steps, came to the fore during the 1970s and 1980s with a vast number of works published involving such approaches, more so than deterministic algorithms. Theoretical results for deterministic approaches only began to come to light in the 1990s, forming the backbone of research into derivative free approaches over the last two decades. Nevertheless, these methods are unsuitable for problems of the form (3.5.8), due to the expected large-scale nature of the problems we expect to encounter. In particular, one must perform objective function and constraint evaluations for every candidate solution, and so such approaches can become computationally prohibitive, even for relatively small-scale problems.

A paper describing the benefits and drawbacks of derivative free approaches for topology optimisation was published recently by Sigmund [171], with the conclusion that such approaches are unsuitable for problems in topology optimisation due to their inherent large-scale nature. Additionally, for certain problems a dependence on the mesh parameter was noted even for relatively coarse finite element meshes. We therefore consider approaches in this thesis involving the use of derivative information.

4.2 Derivative Based Approaches

We begin this section by considering the Lagrangian function as described in Definition 1.2.14. In the case of (3.5.5), we arrive at the following

$$\begin{aligned}\hat{\mathcal{L}} = \hat{\mathcal{L}}(u, \rho, \hat{u}, \hat{\lambda}, \hat{\kappa}, \hat{\delta}) := F(u) - (a_{\rho}(u, \hat{u}) - F(\hat{u})) - \hat{\lambda}\left(V_{\mathrm{vol}} - \int_{\Omega} \rho(\mathbf{x})\,\mathrm{d}\Omega\right) - \\ \int_{\Omega} \hat{\kappa}(\mathbf{x})(\rho(\mathbf{x}) - \rho_{low})\,\mathrm{d}\Omega - \int_{\Omega} \hat{\delta}(\mathbf{x})(\rho_{high} - \rho(\mathbf{x}))\,\mathrm{d}\Omega.\end{aligned}$$

In the above, the variable $\hat{u}$ corresponds to the Lagrange multiplier for the equilibrium constraint (3.5.5b). Each of $\hat{\lambda}$, $\hat{\kappa}$ and $\hat{\delta}$ denote Lagrange multipliers for the inequality constraints (3.5.5c) and (3.5.5d), where we consider (3.5.5d) in the following manner

$$\rho_{low} \leq \rho(\mathbf{x}) \leq \rho_{high} \implies \begin{cases} \rho(\mathbf{x}) - \rho_{low} \geq 0, \\ \rho_{high} - \rho(\mathbf{x}) \geq 0. \end{cases}$$

Through consideration of the 1^{st} order necessary optimality conditions as given in Theorem 1.2.4, we have the following

$$\text{Stationarity:} \quad \begin{cases} \hat{\mathcal{L}}'_u = -\, a_\rho(\hat{u}, v) + F(v) = 0, \\ \hat{\mathcal{L}}'_\rho = -\, \boldsymbol{\epsilon}(u) \colon \dfrac{\partial E}{\partial \rho} \colon \boldsymbol{\epsilon}(u) + \hat{\lambda} - \hat{\kappa} + \hat{\delta} = 0, \end{cases} \tag{4.2.1a}$$

$$\text{Primal Feasibility:} \quad \begin{cases} -\, a_\rho(u, v) + F(v) = 0 \implies \hat{u} = v, \\ \hat{r}(\rho) := V_{\text{vol}} - \displaystyle\int_\Omega \rho(\mathbf{x}) \,\mathrm{d}\Omega \geq 0, \\ \rho(\mathbf{x}) - \rho_{low} \geq 0, \\ \rho_{high} - \rho(\mathbf{x}) \geq 0, \end{cases} \tag{4.2.1b}$$

$$\text{Dual Feasibility:} \quad \begin{cases} \hat{\lambda} \geq 0, \\ \hat{\kappa} \geq 0, \\ \hat{\delta} \geq 0, \end{cases} \tag{4.2.1c}$$

$$\text{Comp. Slackness:} \quad \begin{cases} \hat{\lambda} \left(V_{\text{vol}} - \displaystyle\int_\Omega \rho(\mathbf{x}) \,\mathrm{d}\Omega \right) = 0, \\ \hat{\kappa} \left(\rho(\mathbf{x}) - \rho_{low} \right) = 0, \\ \hat{\delta} \left(\rho_{high} - \rho(\mathbf{x}) \right) = 0. \end{cases} \tag{4.2.1d}$$

Using the complementarity slackness conditions (4.2.1d), it is clear that for areas of intermediate density, the second stationarity constraint will reduce to

$$\hat{\lambda} = \mu \rho(\mathbf{x})^{\mu - 1} \boldsymbol{\epsilon}(u) \colon \bar{E} \colon \boldsymbol{\epsilon}(u) > 0, \tag{4.2.2}$$

whenever $\boldsymbol{\epsilon}(u)$ is non-zero, suggesting that the right hand side, bearing resemblance to the strain energy density[I], is constant and equal the Lagrange multiplier $\hat{\lambda}$ in such regions. Here, it is noted that due to the positivity of $\hat{\lambda}$ above, the volume constraint is required to be satisfied with equality. This observation will be used in the following sections, where we consider different approaches for dealing with the nonlinearity present in (4.2.1).

4.3 Optimality Criteria (OC) Method

4.3.1 Introduction and Mathematical Description

As an initial attempt to deal with the nonlinearity present in both the equilibrium equations and also (4.2.2), we multiply both sides of (4.2.2) by $\rho(\mathbf{x})$ and consider linearisation through a fixed point iterative approach as follows

$$a_{\rho^{\{k_{it}\}}}(u^{\{k_{it}\}}, v) = F(v), \tag{4.3.1a}$$

$$\rho(\mathbf{x})^{\{k_{\text{it}}+1\}} = \left(\hat{\lambda}^{\{k_{\text{it}}\}}\right)^{-1} \mu \left(\rho(\mathbf{x})^{\{k_{\text{it}}\}}\right)^{\mu} \boldsymbol{\epsilon}\left(u^{\{k_{\text{it}}\}}\right) : \bar{E} \colon \boldsymbol{\epsilon}\left(u^{\{k_{\text{it}}\}}\right). \tag{4.3.1b}$$

Based on a given density $\rho^{\{k_{it}\}}$ subject to both upper and lower bounds, the displacement $u^{\{k_{it}\}}$ may be determined through (4.3.1a). Using this displacement value, we then look to update the density through (4.3.1b) subject to calculation of $\hat{\lambda}$. Theorem 1.2.4 asserts that the calculated value of $\hat{\lambda}$ in (4.3.1b) will correspond to the Lagrange multiplier for a KKT point when $\hat{r}(\rho) = 0$. Using the observation of positivity in (4.2.2), the value of $\hat{\lambda}$ may be determined in a straightforward manner through use of a bisection method for the linear function $\hat{r}(\rho)$. By defining

$$\hat{b}^{\{k_{\text{it}}\}}_{\rho^{\{k_{\text{it}}\}}} := \left(\hat{\lambda}^{\{k_{\text{it}}\}}\right)^{-1} \mu \left(\rho(\mathbf{x})^{\{k_{\text{it}}\}}\right)^{\mu-1} \boldsymbol{\epsilon}\left(u^{\{k_{\text{it}}\}}\right) : \bar{E} \colon \boldsymbol{\epsilon}\left(u^{\{k_{\text{it}}\}}\right), \tag{4.3.2}$$

[I] The strain energy density amounts to the strain energy (namely the integrand from (2.2.5)) per unit volume.

we present the optimality criteria method in Algorithm 4.1 through a heuristic fixed point update approach involving the addition (removal) of material from areas where the strain energy density is less (greater) than $\hat{\lambda}$ in a similar manner to the presentation in [16, p. 10]. The variables η and ζ in the algorithm represent a tuning parameter and move limit, respectively, appropriately selected through experimentation, with [16, p. 11] suggesting to take $\eta = 0.5$ and $\zeta = 0.2$ - local optima are found whenever $\hat{b}_\rho = 1$ for acceptable densities. Other quantities, including initial values for both $\overline{\lambda}$ and $\underline{\lambda}$, were set using [170] as a guide.

Included within the algorithm is the possibility to filter the design sensitivities based on the penalisation power μ, where the design sensitivities correspond to how sensitive the objective function will be under changes to the design variables. This area will not

Algorithm 4.1 *OPTIMALITY CRITERIA METHOD (WEAK)*

```
1. $k = 0$, $C = 1$. Set $\rho := \rho(\mathbf{x})$ based on volume and box constraints (e.g: $\rho := \rho_{low}$).
2. While $C > \mathcal{T}$, Do
    (a) Solve $a_\rho(u, v) = F(v)$ for $u$, $\rho_I := \rho$.
    (b) $\overline{\lambda} = 10000$, $\underline{\lambda} = 0$.
    (c) While $\overline{\lambda} - \underline{\lambda} > \mathcal{T}_{\hat{\lambda}}$
        i. $\hat{\lambda} := (\overline{\lambda} + \underline{\lambda})/2$.
        ii. $\rho := \begin{cases} \max\{(1-\zeta)\rho_I, \rho_{low}\}, & \text{if } \rho_I \hat{b}_{\rho_I}^{\eta} \le \max\{(1-\zeta)\rho_I, \rho_{low}\}, \\ \min\{(1+\zeta)\rho_I, \rho_{high}\}, & \text{if } \min\{(1+\zeta)\rho_I, \rho_{high}\} \le \rho_I \hat{b}_{\rho_I}^{\eta}, \\ \rho_I \hat{b}_{\rho_I}^{\eta}, & \text{otherwise.} \end{cases}$
        iii. If $\hat{r}(\rho) > 0$, $\underline{\lambda} := \hat{\lambda}$, else If $\hat{r}(\rho) < 0$, $\overline{\lambda} := \hat{\lambda}$,
    (d) Filtering Procedure (dependent on $\mu$) - see [16, pp. 35 – 36].
    (e) $C := \|\rho - \rho_I\|_\infty$, $k_{it} = k_{it} + 1$.
```

be discussed within this thesis, however an explanation of different mesh independent filtering strategies can be found in [16, pp. 28 – 47]. The reason for not discussing this component of the program further is that for the majority of this thesis, much of the work will be restricted to the variable thickness sheet problem. As mentioned in Section 3.5.6, the variable thickness sheet problem is computationally tractable due to the existence of solutions as a consequence of convexity. We therefore do not require a mesh independency filter in order to force solutions towards a black-white rendering in this instance.

The heuristic density update within the described algorithm can be implemented in a relatively straightforward fashion under suitable codes for the generation of finite element meshes. Therefore, the optimality criteria method can be used and applied without a great deal of programming knowledge - in particular (as previously mentioned in Section 3.4), a 99-line open source code has been developed by Sigmund in [170] incorporating the entire algorithm as outlined including an appropriate filtering strategy.

We now consider the above derivation of the optimality criteria algorithm in discretised form. In this case, the Lagrangian for the problem (3.5.7) is:

$$\begin{aligned}\mathcal{L} = \mathcal{L}(\mathbf{u}, \boldsymbol{\rho}, \mathbf{v}, \lambda, \boldsymbol{\kappa}, \boldsymbol{\delta}) := \mathbf{f}^T\mathbf{u} - \mathbf{v}^T\left(K(\boldsymbol{\rho})\mathbf{u} - \mathbf{f}\right) - \lambda\left(V_{\text{vol}} - \sum_{e\in D}\rho_e\right)& \\ - \boldsymbol{\kappa}^T(\boldsymbol{\rho} - \underline{\rho}\,\mathbf{1}_m) - \boldsymbol{\delta}^T(\overline{\rho}\,\mathbf{1}_m - \boldsymbol{\rho}).&\end{aligned} \tag{4.3.3}$$

with $\mathbf{v} \in \mathbb{R}^n$, $\lambda \in \mathbb{R}_{0_+}$, and $\boldsymbol{\kappa}, \boldsymbol{\delta} \in \mathbb{R}^m_{0_+}$, where $\mathbf{1}_m \in \mathbb{R}^m$ is used to denote a vector of ones. The KKT conditions can then be written down as follows

$$\text{Stationarity:}\quad \begin{cases} \nabla_{\mathbf{u}}\mathcal{L} = -\,K(\boldsymbol{\rho})\mathbf{v} + \mathbf{f} = \mathbf{0}, \\ \nabla_{\boldsymbol{\rho}}\mathcal{L} = -\,\mathbf{z}_{\mathbf{v},\mu} + \lambda\mathbf{1}_m - M\mathbf{1}_m + Q\mathbf{1}_m = \mathbf{0}, \end{cases}$$

$$\text{Primal Feasibility:}\quad \begin{cases} -\,K(\boldsymbol{\rho})\mathbf{u} + \mathbf{f} = \mathbf{0} \implies \mathbf{v} = \mathbf{u}, \\ r(\boldsymbol{\rho}) := V_{\text{vol}} - \sum_{e\in D}\rho_e \geq 0, \\ \boldsymbol{\rho}(\mathbf{x}) - \underline{\rho}\,\mathbf{1}_m \geq \mathbf{0}, \\ \overline{\rho}\,\mathbf{1}_m - \boldsymbol{\rho}(\mathbf{x}) \geq \mathbf{0}, \end{cases}$$

$$\text{Dual Feasibility:} \quad \begin{cases} \lambda \geq 0, \\ \boldsymbol{\kappa} \geq \mathbf{0}, \\ \boldsymbol{\delta} \geq \mathbf{0}, \end{cases}$$

$$\text{Comp. Slackness:} \quad \begin{cases} \lambda \left(V_{\text{vol}} - \sum_{e \in D} \rho_e \right) = 0, \\ \boldsymbol{\kappa}^T \left(\boldsymbol{\rho}(\mathbf{x}) - \underline{\rho} \mathbf{1}_m \right) = \mathbf{0}, \\ \boldsymbol{\delta}^T \left(\overline{\rho} \mathbf{1}_m - \boldsymbol{\rho}(\mathbf{x}) \right) = \mathbf{0}. \end{cases}$$

In the above,

$$\mathbf{z}_{\mathbf{v},\mu}(\boldsymbol{\rho}) = (z_1, \ldots, z_m)^T := \left(\mu \rho_1^{\mu-1} \mathbf{v}^T K_1 \mathbf{u}, \ldots, \mu \rho_m^{\mu-1} \mathbf{v}^T K_m \mathbf{u} \right)^T, \tag{4.3.5}$$

with $M := \operatorname{diag}(\kappa_e)$ and $Q := \operatorname{diag}(\delta_e)$ where $e = 1, \ldots, m$. Based on the manipulation of the optimality conditions to the weak formulation, we define the finite element counterpart to (4.3.2) in the following way

$$\mathbf{b}_{\boldsymbol{\rho}^{\{k_{\text{it}}\}}}^{\{k_{\text{it}}\}} := \left(\lambda^{\{k_{\text{it}}\}} \right)^{-1} \mathbf{z}_{\mathbf{u},\mu}^{\{k_{\text{it}}\}} (\boldsymbol{\rho}^{\{k_{\text{it}}\}}) = \left(\lambda^{\{k_{\text{it}}\}} \right)^{-1} (z_1, \ldots, z_m)_{\{k_{\text{it}}\}}^T,$$

enabling for the presentation of the optimality criteria method using finite elements as per Algorithm 4.2. As mentioned previously, the filtering procedure for the design sensitivities will not be discussed within this thesis. However, it is worth mentioning in the case of compliance design that the design sensitivities involve the the derivative of the objective function to (3.5.8) with respect to $\boldsymbol{\rho}$. Through implicit differentiation of the equilibrium equations, we see that

$$\frac{\partial}{\partial \boldsymbol{\rho}} [K(\boldsymbol{\rho}) \mathbf{u}] = \frac{\partial}{\partial \boldsymbol{\rho}} [\mathbf{f}] \implies \frac{\partial K(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}} \mathbf{u} = -K(\boldsymbol{\rho}) \frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}}.$$

By multiplying both sides by $\mathbf{u}^T$, we see that

$$\mathbf{u}^T \frac{\partial K(\boldsymbol{\rho})}{\partial \boldsymbol{\rho}} \mathbf{u} = -\mathbf{u}^T K(\boldsymbol{\rho}) \frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = -\mathbf{z}_{\mathbf{u},\mu}(\boldsymbol{\rho}),$$

Algorithm 4.2 *OPTIMALITY CRITERIA METHOD (FINITE ELEMENT)*

```
1. $k = 0,\ C = 1,\ \boldsymbol{\rho}(\mathbf{x}) := (V_{vol}/m) \cdot \mathbf{1}_m$.
2. While $C > \mathcal{T}$, Do
    (a) $\mathbf{u} = K(\boldsymbol{\rho})^{-1}\mathbf{f}$.
    (b) $\overline{\lambda} = 10000,\ \underline{\lambda} = 0,\ \boldsymbol{\rho}_I := \boldsymbol{\rho},\ P_I := \operatorname{diag}(\boldsymbol{\rho}_I)$.
    (c) While $\overline{\lambda} - \underline{\lambda} > \mathcal{T}_\lambda$
        i. $\lambda := (\overline{\lambda} + \underline{\lambda})/2$.
        ii. $\boldsymbol{\rho} := \begin{cases} \max\{(1-\zeta)\boldsymbol{\rho}_I, \underline{\rho}\}, & \textit{if } P_I \mathbf{b}^{\eta}_{\boldsymbol{\rho}_I} \leq \max\{(1-\zeta)\boldsymbol{\rho}_I, \underline{\rho}\}, \\ \min\{(1+\zeta)\boldsymbol{\rho}_I, \overline{\rho}\}, & \textit{if } \min\{(1+\zeta)\boldsymbol{\rho}_I, \overline{\rho}\} \leq P_I \mathbf{b}^{\eta}_{\boldsymbol{\rho}_I}, \\ P_I \mathbf{b}^{\eta}_{\boldsymbol{\rho}_I}, & \textit{otherwise.} \end{cases}$
        iii. If $r(\boldsymbol{\rho}) > 0,\ \underline{\lambda} := \lambda$, else If $r(\boldsymbol{\rho}) < 0,\ \overline{\lambda} := \lambda$,
    (d) Filtering Procedure (dependent on $\mu$) - see [16, pp. 35 – 36].
    (e) $C = \|\boldsymbol{\rho} - \boldsymbol{\rho}_I\|_\infty,\ k_{it} = k_{it} + 1$.
```

and due to the symmetry of the stiffness matrix

$$K(\boldsymbol{\rho})\mathbf{u} = \mathbf{f} \implies \mathbf{u}^T K(\boldsymbol{\rho}) = \mathbf{f}^T.$$

Therefore

$$\mathbf{f}^T \frac{\partial \mathbf{u}}{\partial \boldsymbol{\rho}} = -\mathbf{z}_{\mathbf{u},\mu}(\boldsymbol{\rho}) \implies \frac{\partial \left[\mathbf{f}^T \mathbf{u}\right]}{\partial \boldsymbol{\rho}} = -\mathbf{z}_{\mathbf{u},\mu}(\boldsymbol{\rho}).$$

Numerical results for the OC approach will be described towards the end of the next section, where we provide a description of another fixed point solution method widely used within the topology optimisation community.

4.4 The Method of Moving Asymptotes (MMA)

4.4.1 Introduction and Mathematical Description

The method of moving asymptotes belongs to a class of optimisation approaches known as conservative convex separable approximation methods. When coupled with upper and lower limits for the vector of variables, these methods are designed for nonlinear problems of the form (1.2.14) involving the replacement of both the nonlinear objective and constraint functions with separable convex approximations, as well as the replacement of the feasible set S with a convex subset $\widetilde{S}$ forming an approximating subproblem around the current iterate. Due to convexity and separability, either a dual method or an interior point approach (amongst others) may be used to obtain the optimal solution to the subproblem. The resulting solution is checked for suitability against the original objective and constraint functions and if necessary, new subproblems are formed involving increasingly conservative approximating functions yielding an updated solution within a finite number of steps.

Designed for structural optimisation problems, the method of moving asymptotes was initially described by Svanberg in [182] for solving nonlinear problems of the following form

$$\begin{array}{lll} \min\limits_{\bar{\mathbf{x}} \in \mathbb{R}^{\bar{n}}} & G_0(\bar{\mathbf{x}}) & \\ \text{subject to:} & G_i(\bar{\mathbf{x}}) \leq \mathbf{0} & i = 1, \ldots, \bar{m}, \\ & \bar{x}_L \leq \bar{x}_j \leq \bar{x}_U & j = 1, \ldots, \bar{n}. \end{array}$$

The method involves the approximation of both the objective and constraint functions in the following manner

$$G_i(\bar{\mathbf{x}}) \approx G_i(\tilde{\mathbf{x}}) + \sum_{j=1}^{\bar{n}} \left(\frac{p_{ij}}{U_j - \bar{x}_j} + \frac{q_{ij}}{\bar{x}_j - L_j} \right) \qquad i = 0, \ldots, \bar{m},$$

where the matrices P and Q defined component-wise as per [16, p. 19] as follows

$$\text{If } \nabla_{\tilde{x}_j} G_i\left(\tilde{\mathbf{x}}\right) > 0 \text{ then } p_{ij} = \left(U_j - \tilde{x}_j\right)^2 \nabla_{\tilde{x}_j} G_i\left(\tilde{\mathbf{x}}\right), \text{ and } q_{ij} = 0, \qquad i = 0, \dots, \bar{m},$$

$$\text{If } \nabla_{\tilde{x}_j} G_i\left(\tilde{\mathbf{x}}\right) < 0 \text{ then } p_{ij} = 0, \text{ and } q_{ij} = -\left(\tilde{x}_j - L_j\right)^2 \nabla_{\tilde{x}_j} G_i\left(\tilde{\mathbf{x}}\right), \qquad i = 0, \dots, \bar{m},$$

with $j = 1, \dots, \bar{n}$. The variables $L_j < \bar{x}_L$ and $U_j > \bar{x}_U$ represent vertical asymptotes for the approximating convex functions, and are updated at each iterative step based on previous values - hence the name of the method. Whilst the initial method presented in 1987 was found to work well in practice, global convergence was not assured, and examples where solutions could not be determined were also noted. Global convergence was described in a later paper published in 1995 [183]. Nevertheless, the resulting algorithm was found to be too slow for general use. A third paper, published in 2002, was not only able to assure global convergence, but also improve on the previous paper to provide an algorithm suitable for practical application.

In terms of compliance design, the method of moving asymptotes is used within a fixed point iteration in a similar manner to the OC method. By using the positivity of $\mathbf{z}_{\mathbf{u},\mu}(\boldsymbol{\rho})$[I] in (4.3.5), updating the density at each iterative step k_{it} through the approximation of the compliance gives rise to the following subproblem

$$\begin{aligned} \min_{\boldsymbol{\rho}} \quad & \left\{ \mathbf{u}^T K\left(\boldsymbol{\rho}^{\{k_{\text{it}}\}}\right)\mathbf{u} - \sum_{e\in D} \frac{\left(\rho_e^{\{k_{\text{it}}\}} - L_e\right)^2}{\rho_e - L_e} \mathbf{z}_{\mathbf{u},\mu}\left(\boldsymbol{\rho}^{\{k_{\text{it}}\}}\right) \right\} \\ \text{subject to:} \quad & \sum_{e\in D} \rho_e \le V_{\text{vol}}, \\ & 0 < \underline{\rho} \le \rho_e \le \overline{\rho} \qquad \forall e \in D. \end{aligned} \tag{4.4.1}$$

Based on appropriately calculated L_j and U_j, the convex separable subproblem (4.4.1) may be solved through use of a dual method. Precise details can be found in [182], however for (4.4.1) this essentially involves adjustment to the Lagrange multiplier for the

[I] The positivity of $\mathbf{z}_{\mathbf{u},\mu}(\boldsymbol{\rho})$ implies that the matrix P defined above is the zero matrix.

Algorithm 4.3 *METHOD OF MOVING ASYMPTOTES (FINITE ELEMENT)*

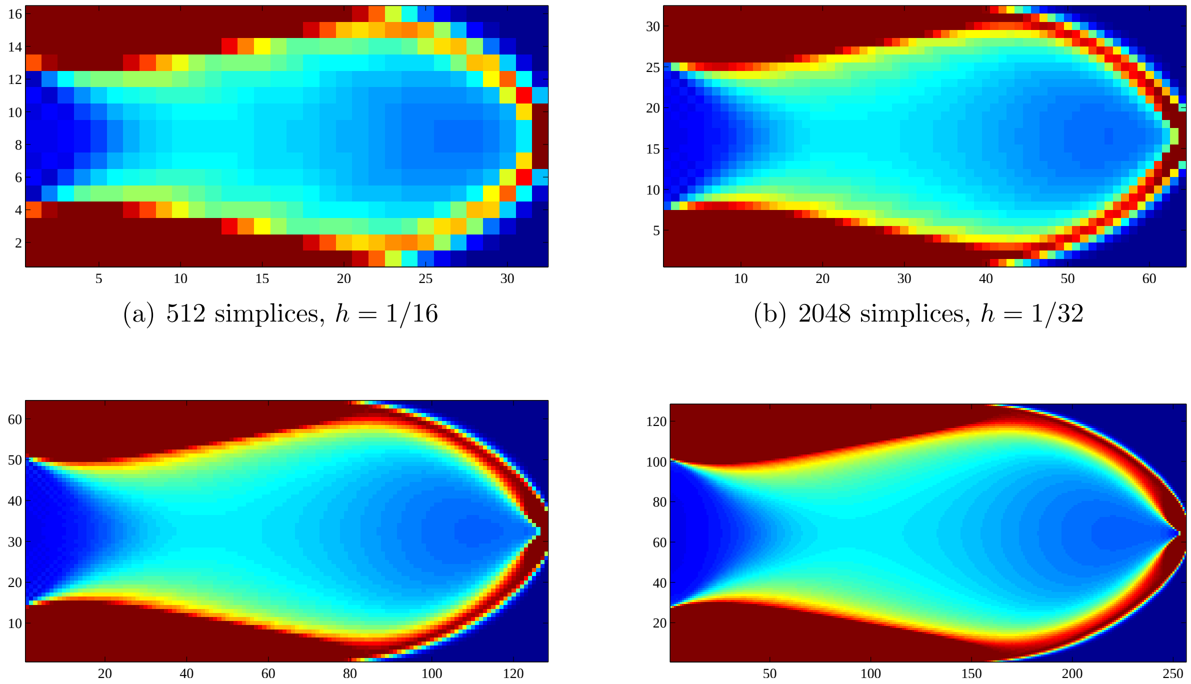
```
1. $k_{it} = 0$, $C = 1$, $s_{tun} := 0.7$, $\boldsymbol{\rho}(\mathbf{x}) := (V_{vol}/m)\,\mathbf{1}_m$, $\boldsymbol{\rho}_I := \boldsymbol{\rho}$, set $\mathbf{L}$ and $\mathbf{U}$ such that
   $\mathbf{L} < \underline{\rho}\,\mathbf{1}_m$ and $\mathbf{U} > \overline{\rho}\,\mathbf{1}_m$.
2. While $C > \mathcal{T}$, Do
    (a) $\mathbf{u} := K(\boldsymbol{\rho})^{-1}\mathbf{f}$, $\boldsymbol{\rho}_{II} := \boldsymbol{\rho}_I$, $\mathbf{L}_I := \mathbf{L}$, $\mathbf{U}_I := \mathbf{U}$.
    (b) If $k_{it} > 0$
        i. $\boldsymbol{\rho}_I := \boldsymbol{\rho}$.
    (c) Appropriate calculation of $\mathbf{L}$ and $\mathbf{U}$ (see [184]), for instance
            $\mathbf{L} := \boldsymbol{\rho}_I - s_{tun}\left(\boldsymbol{\rho}_{II} - \boldsymbol{L}_I\right),$
            $\mathbf{U} := \boldsymbol{\rho}_I + s_{tun}\left(\boldsymbol{U}_I - \boldsymbol{\rho}_{II}\right).$
    (d) Update $\boldsymbol{\rho}$ by solving (4.4.1) through a dual method - see [182] for full details.
    (e) Filtering Procedure (dependent on $\mu$) - see [16, pp. 35 – 36].
    (f) $C = \|\boldsymbol{\rho} - \boldsymbol{\rho}_I\|_\infty$, $k_{it} = k_{it} + 1$.
```

volume constraint in order to obtain an appropriate update for $\boldsymbol{\rho}$ in a not too dissimilar fashion to the approach used within the OC method seen previously. More precise details are provided in [16, p. 20] and also [182], however the above allows for the description of a fixed point solution method using MMA for the density update, as outlined in Algorithm 4.3. In the algorithm, $0 < s_{\text{tun}} < 1$ denotes a tuning parameter in the update of the moving asymptotes, chosen based on [182].

4.4.2 Comparison of Results - OC and MMA

In terms of a direct comparison between both the MMA and the OC methods, it is described in [23] that both methods essentially involve the same type of computations. We now present numerical results for both methods based on a typical topology optimisation problem. In Table 4.1, the number of fixed point iterations are presented for the determination of the cantilever beam problem described in Section 3.6.1. Results are provided for both Algorithm 4.2 and also Algorithm 4.3 introduced in the previous section for differing penalisation powers.

For the experimentation, initial values based on those provided by Sigmund in the open source 99-line code were considered. In particular, $V_{\text{vol}} = m/2$, $\overline{\rho} = 1$, $\underline{\rho} = 10^{-3}$ and the tolerance $\mathcal{T} = 10^{-4}$. In the results, we observe that both algorithms require substantially more iterations when determining effective approximations to black-white designs (corresponding to $\mu = 3$) in comparison to the solution of the VTS problem (namely $\mu = 1$). A pictorial description in both cases may be seen in both Figures 4.1



(a) 512 simplices, $h = 1/16$

(b) 2048 simplices, $h = 1/32$

(c) 8192 simplices, $h = 1/64$

(d) 32768 simplices, $h = 1/128$

Figure 4.1: Illustration of optimal solution layouts for differing numbers of simplices in the case $\mu = 1$.

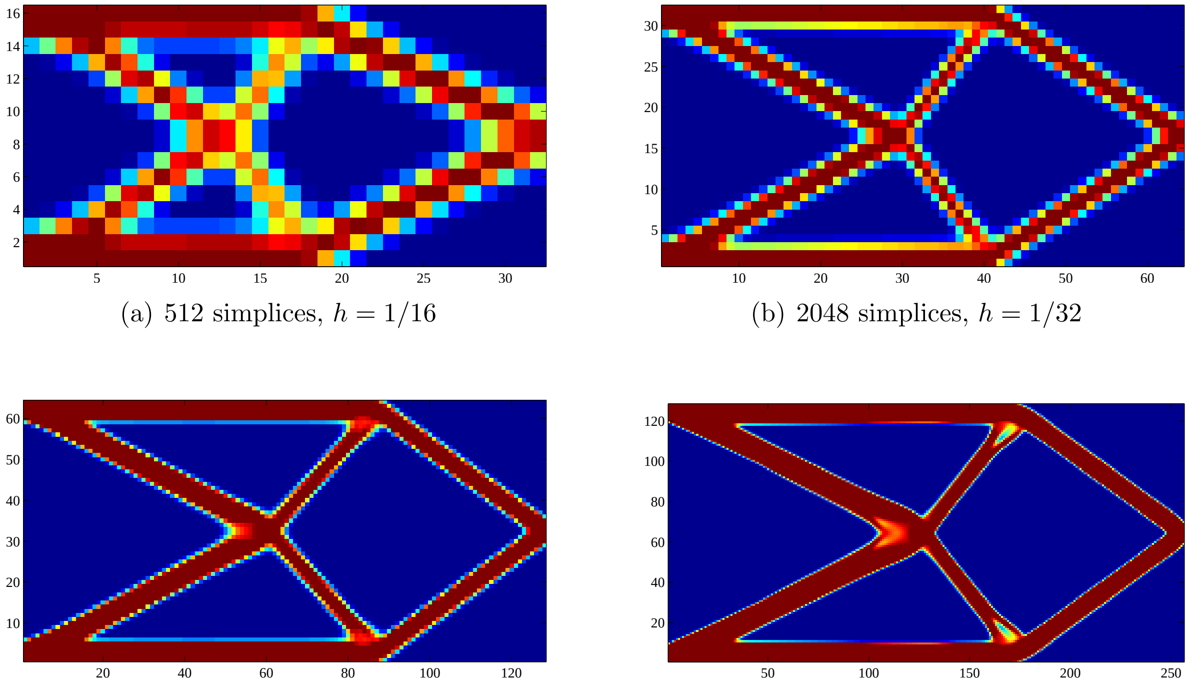
	$h =$	$1/16$	$1/32$	$1/64$	$1/128$	$1/256$
OC	$\mu = 1$	18	30	47	63	82
	$\mu = 3$	36	75	132	184	291
MMA	$\mu = 1$	31	52	83	105	169
	$\mu = 3$	71	82	146	169	274

Table 4.1: Total number of fixed point iterations for both the OC and MMA approaches for differing penalisation powers.

and 4.2, with the respective density plot for each illustration in Figure 4.1 provided in Figure 4.3.

From the tables, one can also see that solutions are typically obtained through the OC approach in a fewer number of iterations when compared directly to the results for the MMA approach. Whilst this is generally the case for compliance design, it is noted in [16, p. 20] that the MMA approach offers greater flexibility than the OC approach, with impressive convergence properties displayed for more complex problems.

Generally, fixed point approaches such as the OC and MMA described in both this



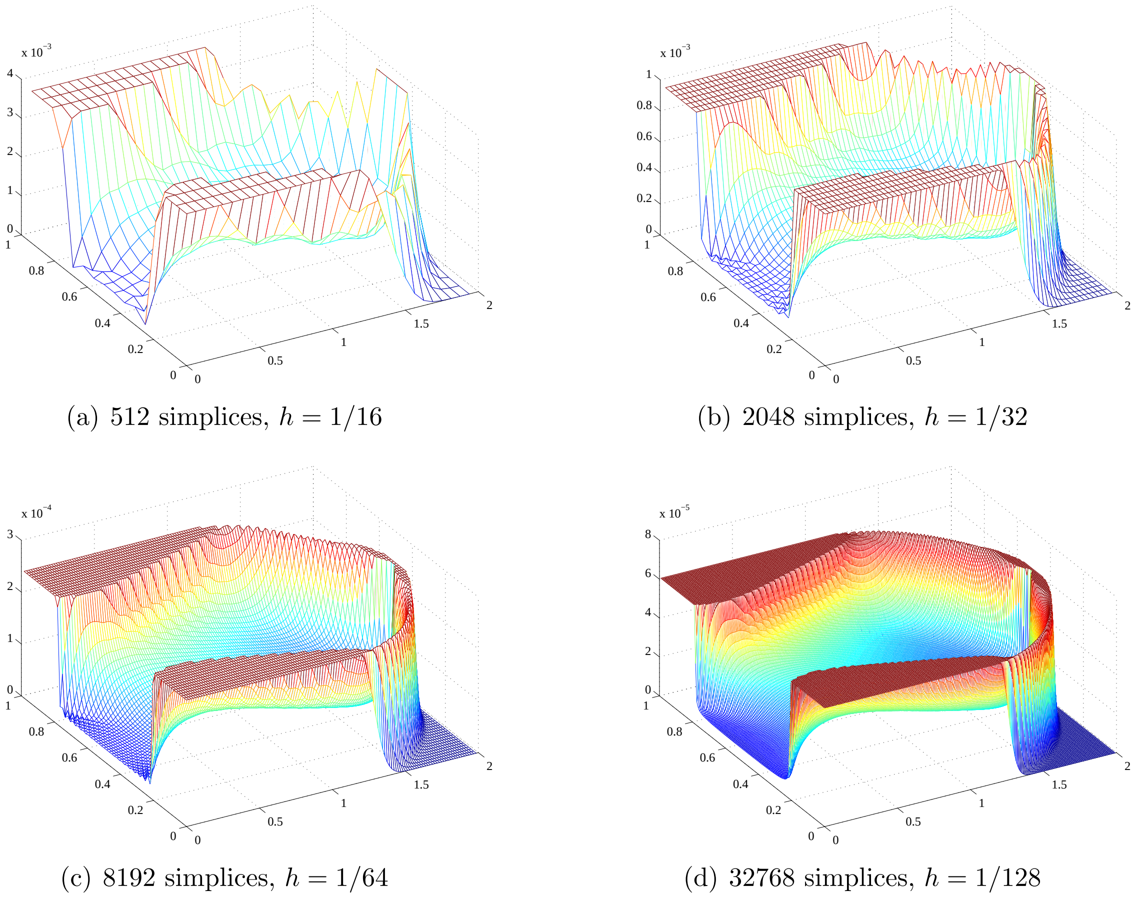
(a) 512 simplices, $h = 1/16$

(b) 2048 simplices, $h = 1/32$

(c) 8192 simplices, $h = 1/64$

(d) 32768 simplices, $h = 1/128$

Figure 4.2: Illustration of optimal solution layouts for differing numbers of simplices in the case $\mu = 3$.



(a) 512 simplices, $h = 1/16$

(b) 2048 simplices, $h = 1/32$

(c) 8192 simplices, $h = 1/64$

(d) 32768 simplices, $h = 1/128$

Figure 4.3: Density plot illustrations for each of the plots shown in Figure 4.1.

and the previous section are the most commonly used solution strategies for topology optimisation. The fixed point approaches described in Algorithms 4.2 and 4.3 involve separate treatment of both the design objective and the equilibrium equations. As has been seen, for an initial given design the stiffness matrix is assembled and used to solve the equilibrium equations for the displacement $\mathbf{u}$. This $\mathbf{u}$ is then used to obtain an appropriate update to the design variables, which is then checked for suitability based on previous values. If an appropriate solution has yet to be found, the process is repeated.

However, we would like to be able to deal with all of the constraints and variables at the same time. These so-called *monolithic* methods are now subject to considerable attention within the PDE constrained optimisation community. An important feature within these methods is that the equilibrium equations are embedded within the optimisation routine, allowing for the simultaneous treatment of all constraints and variables within the problem. Examples highlighting the benefits, and in particular the savings in computational time for such methods, can be found in a number of sources, including [19, 20, 76].

For this thesis, primal-dual Newton methods will be considered with particular focus on interior point approaches. For the interested reader, examples illustrating the application of such approaches for solving large scale topology optimisation problems can be found in [12, 76, 114].

4.5 Interior Point Approach

4.5.1 Introduction and Mathematical Description

As mentioned at the end of the previous section, our aim is to consider approaches that deal with all of the constraints present in (3.5.8) at the same time. For our problem, we will consider an interior point approach which, as originally described by Fiacco and McCormick in [57], involves the determination of solutions to a specified sequence of unconstrained minimisation problems. These types of approaches are used to solve both convex linear and nonlinear optimisation problems iteratively by considering updates confined to the feasible region (cf. [31, 50, 201]). As well as obtaining solutions to certain problems in a polynomial time (cf. [129]), these methods have been used to determine solutions to previously intractable problems, meaning that they are useful from both a theoretical as well as a practical view point.

In terms of the historical background of the interior point approach, Karmarkar [84] is credited with the invention through work published in 1984. This is due to a number of controversial copyright patents introduced in 1988 attempting to protect a code that had been developed at the time. However, a fair amount of research into this area had already been considered by a number of authors prior to this date. In particular, the idea of supplementing the objective function with a penalty term to penalise movements close

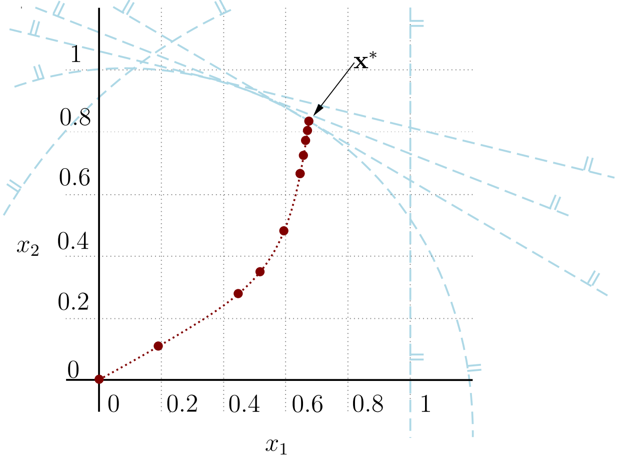


Figure 4.4: Typical example of a two dimensional nonlinear constrained programming problem solved using a primal-dual interior point method. Red dots indicate a typical iterative history of such an approach - confined only to the feasible region. In terms of the pictoral example above, this is assumed to be the positive orthant for both x_1 and x_2, where the blue dashed lines correspond to inequality constraints.

to the boundary was initially presented in 1955 by Frisch [60]. Later in 1961, a slightly different penalty approach was given by Carrol in [34], however much of the credit for the theoretical and computational development of the interior point method can be put down to both Fiacco and McCormick mentioned earlier, with particular note made to the 1968 monograph [57]. Over the years, collaborations between both mathematicians have produced numerous theoretical results relating to interior point approaches, including results describing the convergence properties as well as suggestions for numerical implementation. For the interested reader, a more detailed account on the history and development of interior point methods is discussed by Shanno in [65, pp. 55 – 64]. We begin by rewriting the formulation (3.5.8) slightly to incorporate the inequality constraints within the objective function. This will be achieved through the use of logarithmic barrier terms to present the problem in the following way

$$\begin{aligned} \min_{\mathbf{u},\boldsymbol{\rho}} \quad & \frac{1}{2}\mathbf{f}^T\mathbf{u} - r\sum_{e\in D}\log(\rho_e - \underline{\rho}) - s\sum_{e\in D}\log(\overline{\rho} - \rho_e) \qquad (4.5.1) \\ \text{subject to:} \quad & K(\boldsymbol{\rho})\mathbf{u} = \mathbf{f}, \\ & \sum_{e\in D}\rho_e = V_{\text{vol}}. \end{aligned}$$

Starting from a feasible point lying strictly between the upper and lower limits on the density, the basic aim of such an approach is to construct a barrier that prevents the density from reaching either extreme. However, for typical topology optimisation problems, it is to be expected that the final design will contain areas of either maximum or minimum density. This then presents a problem in that the logarithmic barriers in these areas will increase without bound. To alleviate this issue, the non-negative constants r and s are used in order to balance the relevant contributions from both the original objective function and the objective function with the augmented barrier terms.

By writing down the Lagrangian to the problem (4.5.1)

$$\begin{aligned}\mathcal{L}_{IP}^{(r,s)}\left(\mathbf{u},\boldsymbol{\rho},\mathbf{v},\lambda\right) := \frac{1}{2}\mathbf{f}^T\mathbf{u} - r\sum_{e\in D}\log(\rho_e - \underline{\rho}) - s\sum_{e\in D}\log(\overline{\rho} - \rho_e) \\ -\mathbf{v}^T\left(K(\boldsymbol{\rho})\mathbf{u} - \mathbf{f}\right) - \lambda\left(\sum_{e\in D}\rho_e - V_{vol}\right),\end{aligned}$$

we are able to view solutions to (4.5.1) as solutions to the following saddle point problem

$$\min_{\mathbf{u},\boldsymbol{\rho}}\ \max_{\mathbf{v},\lambda}\ \mathcal{L}_{IP}^{(r,s)}\left(\mathbf{u},\boldsymbol{\rho},\mathbf{v},\lambda\right),$$

with solution $\left(\mathbf{u}^{(r,s)},\boldsymbol{\rho}^{(r,s)}\right)$, where both $\mathbf{v}$ and λ represent Lagrange multipliers. Therefore, we have transformed our constrained minimisation problem into an unconstrained saddle point problem. Through consideration of the barrier trajectory set (or central path[I]) $\left\{\left(\mathbf{u}^{(r,s)},\boldsymbol{\rho}^{(r,s)}\right)\ \middle|\ r,s>0\right\}$, it is shown in [58] that

$$\left(\mathbf{u}^{(r,s)},\boldsymbol{\rho}^{(r,s)}\right)\rightarrow\left(\mathbf{u}^*,\boldsymbol{\rho}^*\right)\qquad\left(r,s\rightarrow 0\right), \tag{4.5.2}$$

with an asterick denoting the optimal solution. This result indicates that one can expect solutions to the family of subproblems (4.5.1) to match that of the original problem (3.5.8) in the limit as the barrier parameters are reduced to zero.

[I] For convex optimisation problems, the central path amounts to the curve described by solutions $\left(\mathbf{u}^{(r,s)},\boldsymbol{\rho}^{(r,s)}\right)$ to (4.5.1) for varying r and s.

The first order necessary optimality conditions to (4.5.1) can be written down in the following manner

$$\text{Stationarity: } \begin{cases} \nabla_{\mathbf{u}} \mathcal{L}_{IP}^{(r,s)} = -\,K(\boldsymbol{\rho})\mathbf{v} + \dfrac{1}{2}\mathbf{f} = \mathbf{0}, \\ \nabla_{\boldsymbol{\rho}} \mathcal{L}_{IP}^{(r,s)} = -\,\mathbf{z}_{\mathbf{v},1} - \lambda\,\mathbf{1}_m - rX^{-1}\mathbf{1}_m + s\widetilde{X}^{-1}\mathbf{1}_m = \mathbf{0}, \end{cases}$$

$$\text{Primal Feas.: } \begin{cases} -\,K(\boldsymbol{\rho})\mathbf{u} + \mathbf{f} = \mathbf{0} \implies \mathbf{v} = \dfrac{1}{2}\mathbf{u}, \\ V_{\text{vol}} - \displaystyle\sum_{e \in D} \rho_e = 0. \end{cases}$$

In the above, $X := \operatorname{diag}(\boldsymbol{\rho} - \underline{\rho}\,\mathbf{1}_m)$ and $\widetilde{X} := \operatorname{diag}(\overline{\rho}\,\mathbf{1}_m - \boldsymbol{\rho})$, where $\mathbf{z}_{\mathbf{v},1}$ is as defined in (4.3.5). It is important to note that the condition number of the Hessian of the Lagrangian may pose an issue for densities close to either $\underline{\rho}$ or $\overline{\rho}$ as both r and s tend to zero. To alleviate this issue, auxiliary non-negative variables $\hat{\boldsymbol{\kappa}}$ and $\hat{\boldsymbol{\delta}}$ are introduced in the following way

$$\hat{\boldsymbol{\kappa}} = \hat{\boldsymbol{\kappa}}^{(r,s)} := rX^{-1}\mathbf{1}_m, \qquad \text{and} \qquad \hat{\boldsymbol{\delta}} = \hat{\boldsymbol{\delta}}^{(r,s)} := s\widetilde{X}^{-1}\mathbf{1}_m. \tag{4.5.4}$$

This process is commonly referred to as perturbed complementarity, since (4.5.2) shows that the resulting values $\hat{\boldsymbol{\kappa}}$ and $\hat{\boldsymbol{\delta}}$ converge to the associated Lagrange multipliers $\boldsymbol{\kappa}$ and $\boldsymbol{\delta}$ for the box constraints in (3.5.8), as seen in (4.3.3). Therefore, as a consequence of (4.5.2),

$$\left(\boldsymbol{v}^{(r,s)}, \lambda^{(r,s)}\right) \to \left(\boldsymbol{v}^{*}, \lambda^{*}\right), \qquad \left(rX^{-1}\mathbf{1}_m, s\widetilde{X}^{-1}\mathbf{1}_m\right) \to \left(\boldsymbol{\kappa}^{*}, \boldsymbol{\delta}^{*}\right). \tag{4.5.5}$$

Additionally, using (4.5.4) we see that

$$MX\mathbf{1}_m = r\mathbf{1}_m, \qquad \text{and} \qquad Q\widetilde{X}\mathbf{1}_m = s\mathbf{1}_m, \tag{4.5.6}$$

where $M := \operatorname{diag}(\boldsymbol{\kappa})$ and $Q := \operatorname{diag}(\boldsymbol{\delta})$. Through this substitution, and the elimination of the Lagrange multiplier $\mathbf{v}$, the first order optimality conditions can be written as

$$\mathcal{R}_1 = \begin{pmatrix} \mathcal{R}_1^{(1)} \\ \mathcal{R}_1^{(2)} \\ \mathcal{R}_1^{(3)} \\ \mathcal{R}_1^{(4)} \\ \mathcal{R}_1^{(5)} \end{pmatrix} := \begin{pmatrix} \mathbf{f} - K(\boldsymbol{\rho})\mathbf{u} \\ V_{\text{vol}} - \displaystyle\sum_{e \in D} \rho_e \\ -\frac{1}{2}\mathbf{z}_{\mathbf{u},1} - \lambda\,\mathbf{1}_m - \boldsymbol{\kappa} + \boldsymbol{\delta} \\ r\mathbf{1}_m - MX\mathbf{1}_m \\ s\mathbf{1}_m - Q\widetilde{X}\mathbf{1}_m \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ 0 \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \tag{4.5.7}$$

where it should be noted that an appropriate ordering of the constraints has been considered for reasons that will become apparent shortly. These conditions can be viewed as the primal-dual conditions for the problem (3.5.8), with this approach often being referred to as the primal-dual method.

4.5.2 Newton Systems for KKT Conditions

By setting $\bar{\mathbf{x}}_1 = \left(\mathbf{u}^T, \lambda, \boldsymbol{\rho}^T, \boldsymbol{\kappa}^T, \boldsymbol{\delta}^T\right)^T$, we apply Newton's method to the resulting nonlinear optimality conditions (4.5.7) to arrive at the following matrix-vector system

$$J_1\left(\bar{\mathbf{x}}_1^{\{k_{\text{it}}-1\}}\right) \Delta\bar{\mathbf{x}}_1^{\{k_{\text{it}}\}} = \mathcal{R}_1\left(\bar{\mathbf{x}}_1^{\{k_{\text{it}}-1\}}\right), \tag{4.5.8}$$

where the Jacobian matrix $J_1\left(\bar{\mathbf{x}}_1\right)$ corresponds to

$$J_1(\bar{\mathbf{x}}_1) = \begin{bmatrix} K(\boldsymbol{\rho}) & 0_{n\times 1} & B(\mathbf{u}) & 0_{n\times m} & 0_{n\times m} \\ 0_{1\times n} & 0 & \mathbf{1}_m^T & 0_{1\times m} & 0_{1\times m} \\ B(\mathbf{u})^T & \mathbf{1}_m & 0_m & I_m & -I_m \\ 0_{m\times n} & 0_{m\times 1} & M & X & 0_m \\ 0_{m\times n} & 0_{m\times 1} & -Q & 0_m & \widetilde{X} \end{bmatrix}, \tag{4.5.9}$$

with $B(\mathbf{u}) := [K_1\mathbf{u}, K_2\mathbf{u}, \ldots, K_m\mathbf{u}] \in \mathbb{R}^{n\times m}$. One can show that the equations presented in (4.5.8) can be seen as identical to those arising from the application of a sequential quadratic programming algorithm applied to (4.5.1). Despite being both nonsymmetric and indefinite, the condition number of the sparse Jacobian matrix J_1 is expected to be bounded under reduction of the barrier parameters r and s with the proviso of non-degeneracy[I] for the constraining blocks. Therefore, it is important to consider appropriate strategies for obtaining an accurate update through (4.5.8). Work by Forsgren, Gill and Shinnerl [59] uses the diagonal structure of both M and Q to transform J_1 into a symmetric matrix. Another possibility is to consider appropriate techniques to condense the matrix J_1, possibly through the elimination of variables or through use of the Schur complement to the system. Whilst this approach inherently leads to ill-conditioning, it is discussed by Wright in [200] that this does not necessarily lead to any adverse effects, and so should not be simply overlooked. Using (4.5.9), it is clear that the last two block rows of the 5×5 block matrix J_1 only involve diagonal matrices, and thus by writing

$$\begin{aligned}
\Delta\boldsymbol{\kappa} &= X^{-1}\left(\mathcal{R}_1^{(4)} - M\Delta\boldsymbol{\rho}\right), \\
\Delta\boldsymbol{\delta} &= \widetilde{X}^{-1}\left(\mathcal{R}_1^{(5)} + Q\Delta\boldsymbol{\rho}\right),
\end{aligned}$$

we are able to reduce the system (4.5.8) to a 3×3 block system of the form

$$J_2\left(\bar{\mathbf{x}}_2^{\{k_{\text{it}}-1\}}\right)\Delta\bar{\mathbf{x}}_2^{\{k_{\text{it}}\}} := J_2\left(\bar{\mathbf{x}}_2^{\{k_{\text{it}}-1\}}\right)\begin{pmatrix}\Delta\mathbf{u}^{\{k_{\text{it}}\}} \\ \Delta\lambda^{\{k_{\text{it}}\}} \\ \Delta\boldsymbol{\rho}^{\{k_{\text{it}}\}}\end{pmatrix} = \begin{pmatrix}\mathcal{R}_1^{(1)} \\ \mathcal{R}_1^{(2)} \\ \mathcal{R}_2^{(3)}\end{pmatrix} =: \mathcal{R}_2. \tag{4.5.10}$$

Here, the matrix $J_2\left(\bar{\mathbf{x}}_2\right)$ and the residual $\mathcal{R}_2^{(3)}$ are as follows

$$J_2(\bar{\mathbf{x}}_2) = \left[\begin{array}{cc:c} K(\boldsymbol{\rho}) & 0_{n\times 1} & B(\mathbf{u}) \\ 0_{1\times n} & 0 & \mathbf{1}_m^T \\ \hdashline B(\mathbf{u})^T & \mathbf{1}_m & -\left(X^{-1}M + \widetilde{X}^{-1}Q\right) \end{array}\right] =: \begin{bmatrix} J_2^A(\bar{\mathbf{x}}_2) & J_2^B(\bar{\mathbf{x}}_2) \\ J_2^C(\bar{\mathbf{x}}_2) & J_2^D(\bar{\mathbf{x}}_2) \end{bmatrix},$$

$$\mathcal{R}_2^{(3)} = \mathcal{R}_1^{(3)} - X^{-1}\mathcal{R}_1^{(4)} + \widetilde{X}^{-1}\mathcal{R}_1^{(5)}.$$

[I] Non-degeneracy essentially ensures that the matrices M (Q) and X $(\widetilde{X})$ do not simultaneously tend to zero as $r \to 0$ $(s \to 0)$. This condition is assured under the LICQ (see Definition 1.2.16). For the matrix (4.5.9), we require that the constraining block $\left[B(\mathbf{u})^T, \mathbf{1}_m\right]$ has full column rank.

The system can be reduced further still by considering the Schur complement of the block $J_2^D(\bar{\mathbf{x}}_2)$ for the matrix $J_2(\bar{\mathbf{x}}_2)$. This results in a 2×2 block system of the form

$$J_3\left(\bar{\mathbf{x}}_3^{\{k_{\text{it}}-1\}}\right)\Delta\bar{\mathbf{x}}_3^{\{k_{\text{it}}\}} := J_3\left(\bar{\mathbf{x}}^{\{k_{\text{it}}-1\}}\right)\begin{pmatrix}\Delta\mathbf{u}^{\{k_{\text{it}}\}}\\ \Delta\lambda^{\{k_{\text{it}}\}}\end{pmatrix} = \begin{pmatrix}\mathcal{R}_3^{(1)}\\ \mathcal{R}_3^{(2)}\end{pmatrix} =: \mathcal{R}_3, \tag{4.5.11}$$

where the block matrix $J_3\left(\bar{\mathbf{x}}\right)$ and the residual $\mathcal{R}_3$ correspond to

$$J_3(\bar{\mathbf{x}}) = \begin{bmatrix}K(\boldsymbol{\rho}) & 0\\ 0 & 0\end{bmatrix} + \begin{bmatrix}B(\mathbf{u})\\ \mathbf{1}_m^T\end{bmatrix}\left[X^{-1}M + \widetilde{X}^{-1}Q\right]^{-1}\begin{bmatrix}B(\mathbf{u})^T & \mathbf{1}_m\end{bmatrix} =: \begin{bmatrix}J_3^A(\bar{\mathbf{x}}) & J_3^B(\bar{\mathbf{x}})\\ J_3^C(\bar{\mathbf{x}}) & J_3^D(\bar{\mathbf{x}})\end{bmatrix},$$

$$\begin{pmatrix}\mathcal{R}_3^{(1)}\\ \mathcal{R}_3^{(2)}\end{pmatrix} = \begin{pmatrix}\mathcal{R}_1^{(1)}\\ \mathcal{R}_1^{(2)}\end{pmatrix} + \begin{bmatrix}B(\mathbf{u})\\ \mathbf{1}_m^T\end{bmatrix}\left[X^{-1}M + \widetilde{X}^{-1}Q\right]^{-1}\mathcal{R}_2^{(3)}. \tag{4.5.12}$$

The remaining part of the solution, $\Delta\boldsymbol{\rho}$, is then computed through

$$\Delta\boldsymbol{\rho} = D\mathcal{R}_{\boldsymbol{\rho}} := \left[X^{-1}M + \widetilde{X}^{-1}Q\right]^{-1}\left(\mathcal{R}_2^{(3)} - B(\mathbf{u})^T\mathcal{R}_1^{(1)} - \mathcal{R}_1^{(2)}\mathbf{1}_m\right). \tag{4.5.13}$$

4.5.3 Step Size Rule

We propose a step size rule that will effectively involve finding α_L such that $\rho_e + \Delta\rho_e \geq \underline{\rho}$ for e such that $\Delta\rho_e < 0$ and α_U such that $\rho_e + \Delta\rho_e \leq \overline{\rho}$ for e such that $\Delta\rho_e > 0$. Therefore, we consider obtaining α_L and α_U as follows

$$\alpha_L = 0.9 \cdot \min_{e:\Delta\rho_e<0}\left\{\frac{\underline{\rho} - \rho_e}{\Delta\rho_e}\right\}, \tag{4.5.14a}$$

$$\alpha_U = 0.9 \cdot \min_{e:\Delta\rho_e>0}\left\{\frac{\overline{\rho} - \rho_e}{\Delta\rho_e}\right\}. \tag{4.5.14b}$$

The constant 0.9 represents an appropriate shortening of the Newton step to the interior of the feasible region. Now, an α will be selected from both of (4.5.14a) and (4.5.14b). In the event that both α_L and α_U are greater than 1, the step will be shortened appropriately by defining

$$\alpha_{LS} := \min\left\{\alpha_L, \alpha_U, 1\right\}.$$

Therefore, the update at each Newton step is modified so that at each k_{it}, the step size parameter is taken into account:

$$\bar{\boldsymbol{x}}_c^{\{k_{\text{it}}\}} = \bar{\boldsymbol{x}}_c^{\{k_{\text{it}}-1\}} + \alpha_{LS}\Delta\bar{\boldsymbol{x}}_c^{\{k_{\text{it}}\}} \qquad c \in \{1,2,3\}. \tag{4.5.15}$$

It should be noted that this step size technique is relatively simple and straightforward to both implement and use. A more sophisticated line search procedure is described in [83] and could potentially be used here. However, it will be illustrated later that the step size rule described above was able to yield desirable results.

4.5.4 Primal-Dual Newton Method

Presented in Algorithm 4.4 is the resulting algorithm for the primal-dual Newton method derived in the previous section as applied to either the full system (4.5.8), or the reduced systems described in both (4.5.10) and (4.5.11) (corresponding to c equal to 1, 2 and 3 respectively). Within the algorithm, we see that the both outer and inner iterations are described. At each outer iteration, the barrier parameters are reduced by damping coefficients $\beta_r, \beta_s \in (0,1)$, until a suitable tolerance $\mathcal{T}$ is reached, essentially corresponding to (4.5.5).

At each inner iteration, we look to solve the equality constrained barrier problem described in (4.5.1) using Newton's method for the current barrier parameters until an inner tolerance $\mathcal{T}_N$ is attained. The criteria for the inner tolerance corresponds to a term bearing resemblance to the inverse energy norm of the residual based on the relevant Jacobian matrix (as opposed to the stiffness matrix). For clarity, the subscript $\bar{\mathbf{x}}$ is used to denote assembly of the relevant matrices with respect to the current iterate.

It should also be mentioned here that a more sophisticated implementation of Algorithm 4.4 can be found in [83]. In particular, the method presented contains an adaptive choice for the penalty parameters r and s. However, it will be illustrated in the next section that the current penalisation of both penalty parameters is sufficient to yield satisfactory results.

Algorithm 4.4 *PRIMAL-DUAL NEWTON METHOD (c)*

```
1. $k = 0,\ l = 0,\ r = 1,\ s = 1,\ IE = 1.$

2. $\mathbf{u} = K^{-1}\left(\left(V_{vol}/m\right)\mathbf{1}_m\right)\mathbf{f},\ \boldsymbol{\rho} = \left(V_{vol}/m\right)\mathbf{1}_m,\ \lambda = 1,\ \boldsymbol{\kappa} = \mathbf{1}_m,\ \boldsymbol{\delta} = \mathbf{1}_m.$

3. $\bar{\mathbf{x}} := \bar{\mathbf{x}}_c,\ J := J_c,\ \mathcal{R} := \mathcal{R}_c.$

4. $\bar{\mathbf{x}}_1 = \left(\mathbf{u}^T, \lambda, \boldsymbol{\rho}^T, \boldsymbol{\kappa}^T, \boldsymbol{\delta}^T\right)^T.$

5. While $max(r, s) > \mathcal{T}$, Do

    (a) While $IE > \mathcal{T}_N$ Do

        i. $\Delta\bar{\mathbf{x}} := J^{-1}\left(\bar{\mathbf{x}}\right)\mathcal{R}\left(\bar{\mathbf{x}}\right).$

        ii. If $c = 3$ Do

            A. $\Delta\boldsymbol{\rho} := D\mathcal{R}_{\boldsymbol{\rho}}.$

        iii. If $c > 1$ Do

            A. $\Delta\boldsymbol{\kappa} := X^{-1}\left(\mathcal{R}_1^{(4)} - M\Delta\boldsymbol{\rho}\right)_{\bar{\mathbf{x}}}.$
            B. $\Delta\boldsymbol{\delta} := \widetilde{X}^{-1}\left(\mathcal{R}_1^{(5)} + Q\Delta\boldsymbol{\rho}\right)_{\bar{\mathbf{x}}}.$

        iv. $\bar{\mathbf{x}}_1 := \bar{\mathbf{x}}_1 + \alpha_{LS}\Delta\bar{\mathbf{x}}_1.$

        v. $k_{it} := k_{it} + 1.$

        vi. $IE := \mathcal{R}\left(\bar{\mathbf{x}}\right)^T\Delta\bar{\mathbf{x}}.$

    (b) $r := \beta_r r, s := \beta_s s.$

    (c) $l_{it} := l_{it} + 1.$
```

4.5.5 Numerical Results

Table 4.2 provides an illustration of the total number of iterations required to achieve optimal designs of both the full and reduced formulations (corresponding to $c = 1$ and $c = 3$ respectively) for the cantilever beam problem.

Both outer and inner iteration numbers are provided describing the total number of primal-dual as well as Newton steps, with the bracketed numbers indicating the total number of Newton steps (k_{it} - inner iterations) required. Data provided was similar to that described in Section 4.4.2, with the main difference being the tolerance criteria to account for the relevant variables updated at each Newton step - for completeness, $\mathcal{T}_N = 10^{-7}$ was used. The damping parameters considered were $\beta_r, \beta_s = 1/10$, with the outer stopping criteria $\mathcal{T}$ set as 10^{-8} in order to gauge both approaches in the limit as the barrier parameters were reduced to zero. The table also provides estimates of the condition numbers for the relevant Jacobian matrices, taken after satisfaction of the outer tolerance, as well as the condition number for the associated stiffness matrices at optimality. Here, the ill-conditioning of the reduced approach is plain to see, with figures roughly corresponding to the squares of those viewed in the case $c = 1$. In particular, the figures represent a substantial increase on those displayed for the associated stiffness matrix.

Within all of the (finite element) algorithms presented within this chapter, there is a

	$h =$	$1/16$	$1/32$	$1/64$	$1/128$	$1/256$
Its	$c = 1$	9 (33)	9 (33)	9 (29)	9 (26)	9 (24)
	$c = 3$	9 (41)	9 (40)	9 (35)	9 (32)	9 (30)
Cond (est)	$c = 1$	2.6×10^4	3.0×10^6	6.3×10^7	8.6×10^8	7.3×10^{10}
	$c = 3$	4.1×10^8	3.5×10^{11}	2.2×10^{14}	1.9×10^{17}	8.8×10^{19}
$\kappa_2(K)$		7.2×10^3	3.4×10^4	1.4×10^5	9.2×10^5	7.9×10^6

Table 4.2: Comparison between number of iterations required for both full ($c = 1$) and reduced ($c = 3$) problems, as well as condition numbers for the relevant Jacobian matrices and also the associated stiffness matrices.

persistent need to determine solutions to linear systems of equations of the form $A\mathbf{x} = \mathbf{b}$. For the OC and the MMA approaches, the matrix A involved in the finite element analysis was symmetric positive definite. However, in the case of the IP approach, the full Jacobian matrix was found to be both non-symmetric and indefinite. Despite symmetry in the reduced block 2×2 case, the resulting matrix was also indefinite. The notably high condition numbers recorded in Table 4.2 not only suggest that our Jacobian matrix for the reduced problem approaches singularity at optimality, but also that solutions obtained using a particularly fine finite element mesh may well consist of inaccuracies. Therefore, we would like to consider appropriate strategies for the determination of solutions to such systems in order to arrive at computationally efficient algorithms for the solution of large scale topology optimisation problems.

CHAPTER 5

REVIEW OF SOLUTION METHODS FOR LINEAR SYSTEMS

As per the title, the aim of this chapter is to provide an overview of solution methods for linear systems of equations. Lettering used within this chapter is designed to be self-contained, and should not be confused with existing definitions presented elsewhere within the thesis.

5.1 Introduction

A number of strategies have been considered for solving linear systems of the form

$$A\mathbf{x} = \mathbf{b}, \tag{5.1.1}$$

where $A \in \mathbb{R}^{n\times n}$, $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}^n$. Systems of this type are commonplace in numerous mathematical modelling situations, typically arising from the discretisation of either integral or differential equations.

Provided that A is nonsingular, the linear system (5.1.1) will have a unique solution. In the case of singularity, the system (5.1.1) may have either an infinite number of solutions, or no solutions at all. For this thesis, the systems that will be encountered will involve non-singular system matrices, for which we describe and outline appropriate solution techniques throughout the course of this chapter.

Numerical methods for obtaining solutions to (5.1.1) can be classed as either direct methods or iterative methods. Hybrid methods, involving a combination of both direct and iterative solution methods, can also be considered. Examples include multigrid and domain decomposition methods, as well as the factorisation of problems for construction of an appropriate preconditioning strategy, and also block iterative methods, involving the use of direct solution methods on subblocks.

5.2 Direct Methods

In the absence of rounding errors, direct solution methods are able to provide an exact solution to (5.1.1) within a finite number of steps. In the case where the system matrix in (5.1.1) is dense, direct inversion involves $\mathcal{O}\left(n^3\right)$ operations in order to determine the solution. Typical direct approaches involve the factorisation of A into matrices X and Y such that $A = XY$, with the idea that both X and Y will have a certain structure which can be exploited. Then, one could think of obtaining the solution to (5.1.1) in the following manner

- Factorise A into $A = XY$,
- Solve $X\mathbf{y} = \mathbf{b}$,
- Solve $Y\mathbf{x} = \mathbf{y}$.

The most commonly used method is Gaussian elimination, involving the transformation of the system matrix A into an upper triangular matrix through successive multiplications by appropriate lower triangular matrices. In this case, the matrix A can be seen to be factorised into the product of a lower triangular matrix L and an upper triangular matrix U such that $A = LU$, often coupled with partial (or full) pivoting for stability. The solution is then obtained initially through forward substitution involving the lower triangular matrix L followed by backward substitution involving the upper triangular matrix U, both having a computational cost of approximately $\mathcal{O}\left(n^2\right)$ operations. The computational cost of Gaussian elimination can be shown to be of $\mathcal{O}\left(\frac{2}{3}n^3\right)$.

A comparative approach that avoids the need for pivoting involves writing A in terms of an orthogonal matrix Q and an upper triangular matrix R such that $A = QR$, known as the QR factorisation of A. This is commonly achieved through the use of either Householder reflections, Gram-Schmidt orthogonalisation or Givens rotations, to allow for the solution to the system (5.1.1) to be determined through the solution to a system of the form (5.1.1) involving an orthogonal matrix Q, followed by backward substitution using the matrix R. Despite avoiding the need for pivoting, this solution method involves approximately twice the number of operations of Gaussian elimination, namely $\mathcal{O}\left(\frac{4}{3}n^3\right)$.

In the case where the system matrix A is symmetric positive definite, a Cholesky factorisation involving a lower triangular matrix L of the form $A = LL^T$ can be determined with a computational cost of $\mathcal{O}\left(\frac{1}{3}n^3\right)$, roughly half the cost of the LU factorisation. Nevertheless, the requirement of $\mathcal{O}\left(n\right)$ steps with $\mathcal{O}\left(n^2\right)$ work involved in each leads to methods with a flop count of $\mathcal{O}\left(n^3\right)$. In the case where the matrix A is dense, this will be not only expensive but also prohibitive in certain cases; in particular, for large systems of equations. Additionally, direct approaches cannot be truncated part way through the solution process in order to provide an approximate solution - they must be run to completion. In the case where the system matrix A is symmetric positive definite, a Cholesky factorisation involving a lower triangular matrix L of the form $A = LL^T$ can be determined with a computational cost of $\mathcal{O}\left(\frac{1}{3}n^3\right)$, roughly half the cost of the LU factorisation. Nevertheless, the requirement of $\mathcal{O}\left(n\right)$ steps with $\mathcal{O}\left(n^2\right)$ work involved in each leads to methods with a flop count of $\mathcal{O}\left(n^3\right)$. In the case where the matrix A is dense, this will be not only expensive but also prohibitive in certain cases; in particular, for large systems of equations. Additionally, direct approaches cannot be truncated part way through the solution process in order to provide an approximate solution - they must be run to completion. In the case where the system matrix A is symmetric positive definite, a Cholesky factorisation involving a lower triangular matrix L of the form $A = LL^T$ can be determined with a computational cost of $\mathcal{O}\left(\frac{1}{3}n^3\right)$, roughly half the cost of the LU factorisation. Nevertheless, the requirement of $\mathcal{O}\left(n\right)$ steps with $\mathcal{O}\left(n^2\right)$ work involved in each leads to methods with a flop count of $\mathcal{O}\left(n^3\right)$. In the case where the matrix A is dense, this will be not only expensive but also prohibitive in certain cases; in particular, for large systems of equations. Additionally, direct approaches cannot be truncated part way through the solution process in order to provide an approximate solution - they must be run to completion. In the case where the system matrix A is symmetric positive definite, a Cholesky factorisation involving a lower triangular matrix L of the form $A = LL^T$ can be determined with a computational cost of $\mathcal{O}\left(\frac{1}{3}n^3\right)$, roughly half the cost of the LU factorisation. Nevertheless, the requirement of $\mathcal{O}\left(n\right)$ steps with $\mathcal{O}\left(n^2\right)$ work involved in each leads to methods with a flop count of $\mathcal{O}\left(n^3\right)$. In the case where the matrix A is dense, this will be not only expensive but also prohibitive in certain cases; in particular, for large systems of equations. Additionally, direct approaches cannot be truncated part way through the solution process in order to provide an approximate solution - they must be run to completion. In the case where the system matrix A is symmetric positive definite, a Cholesky factorisation involving a lower triangular matrix L of the form $A = LL^T$ can be determined with a computational cost of $\mathcal{O}\left(\frac{1}{3}n^3\right)$, roughly half the cost of the LU factorisation. Nevertheless, the requirement of $\mathcal{O}\left(n\right)$ steps with $\mathcal{O}\left(n^2\right)$ work involved in each leads to methods with a flop count of $\mathcal{O}\left(n^3\right)$. In the case where the matrix A is dense, this will be not only expensive but also prohibitive in certain cases; in particular, for large systems of equations. Additionally, direct approaches cannot be truncated part way through the solution process in order to provide an approximate solution - they must be run to completion. In the case where the system matrix A is symmetric positive definite, a Cholesky factorisation involving a lower triangular matrix L of the form $A = LL^T$ can be determined with a computational cost of $\mathcal{O}\left(\frac{1}{3}n^3\right)$, roughly half the cost of the LU factorisation. Nevertheless, the requirement of $\mathcal{O}\left(n\right)$ steps with $\mathcal{O}\left(n^2\right)$ work involved in each leads to methods with a flop count of $\mathcal{O}\left(n^3\right)$. In the case where the matrix A is dense, this will be not only expensive but also prohibitive in certain cases; in particular, for

large systems of equations. Additionally, direct approaches cannot be truncated part way through the solution process in order to provide an approximate solution - they must be run to completion. In the case where the system matrix A is symmetric positive definite, a Cholesky factorisation involving a lower triangular matrix L of the form $A = LL^T$ can be determined with a computational cost of $\mathcal{O}\left(\frac{1}{3}n^3\right)$, roughly half the cost of the LU factorisation. Nevertheless, the requirement of $\mathcal{O}\left(n\right)$ steps with $\mathcal{O}\left(n^2\right)$ work involved in each leads to methods with a flop count of $\mathcal{O}\left(n^3\right)$. In the case where the matrix A is dense, this will be not only expensive but also prohibitive in certain cases; in particular, for large systems of equations. Additionally, direct approaches cannot be truncated part way through the solution process in order to provide an approximate solution - they must be run to completion.
In the case where the system matrix A is symmetric positive definite, a Cholesky factorisation involving a lower triangular matrix L of the form $A = LL^T$ can be determined with a computational cost of $\mathcal{O}\left(\frac{1}{3}n^3\right)$, roughly half the cost of the LU factorisation. Nevertheless, the requirement of $\mathcal{O}\left(n\right)$ steps with $\mathcal{O}\left(n^2\right)$ work involved in each leads to methods with a flop count of $\mathcal{O}\left(n^3\right)$. In the case where the matrix A is dense, this will be not only expensive but also prohibitive in certain cases; in particular, for large systems of equations. Additionally, direct approaches cannot be truncated part way through the solution process in order to provide an approximate solution - they must be run to completion.

These drawbacks suggest inefficiencies with direct approaches that lead to obvious investigations into ways to improve and remove such issues. Typically, approaches can be seen to involve the exploitation of the structure or patterns involved within A in order to devise a more efficient solution method. Whilst certain matrices may have no obvious patterns or structure to utilise, large matrices of computational interest typically arise as a result of the discretisation of differential or integral equations. Such matrices generally involve arguably the most obvious and exploitable structure, namely sparsity. The precise definition of sparsity is somewhat ambiguous, however a sparse matrix is generally referred to as a matrix involving a small number of non-zero entries, with potential computational advantages gained through the exploitation of the large number of zero entries. For instance, one can imagine a system of the form (5.1.1) involving circa 10^7 entries arriving as a direct result of a finite element discretisation of a partial differential equation with only 20 non-zero elements in each row. Despite the relatively small number of non-zero elements within the matrix, use of any of the direct approaches mentioned up to this point would be computationally demanding. However, efficient direct methods that are able to exploit sparsity have been developed, with work dating back to the 1960s. Examples include both nested dissection and also minimum degree reordering. A more in-depth look at direct methods for sparse linear systems can be found in [48], with two of the authors (Duff and Reid) viewed as contributing towards numerous early developments within the field.

5.3 Iterative Methods

Iterative approaches seek to approximate the solution to (5.1.1) by generating a sequence of iterates $\left\{\mathbf{x}^{\{k_{\text{it}}\}}\right\}_{k_{\text{it}}\in\mathbb{N}}$ such that the sequence will tend to the solution within a finite (and reasonable) number of iterative steps. Instead of accessing elements or blocks of K, iterative approaches only involve the system matrix in the context of matrix-vector multiplication. These methods are able to exploit the sparsity structure of K, and usually require relatively low storage requirements, meaning that they are advantageous for larger problems (particularly in three dimensions) where the use of direct methods can become prohibitive. Unlike direct methods, iterative methods may be stopped at a given iterate to provide an approximate solution to (5.1.1) possibly to satisfy a relatively coarse tolerance, as required in certain engineering applications. Here, a flop count of $\mathcal{O}\left(n^3\right)$ represents a worst case scenario, with ideal iterative methods involving $\mathcal{O}\left(1\right)$ steps (in place of $\mathcal{O}\left(n\right)$), with $\mathcal{O}\left(n\right)$ work per step (in place of $\mathcal{O}\left(n^2\right)$), to reduce the total flop count to just $\mathcal{O}\left(n\right)$. However, in general such rapid speedup should not be expected; a more realistic improvement corresponds to a reduction in the flop count from $\mathcal{O}\left(n^3\right)$ to

$\mathcal{O}\left(n^2\right)$.

The majority of iterative solvers can be viewed as either stationary or non-stationary iterative methods. Stationary iterative methods attempt to solve linear systems by constructing an iterative sequence of the form

$$\mathbf{x}^{\{k_{\text{it}}\}} = B\mathbf{x}^{\{k_{\text{it}}-1\}} + \mathbf{c},$$

based on an initial $\mathbf{x}^{\{0\}}$, where neither the matrix B nor the vector $\mathbf{c}$ carry a dependence on k_{it}. Examples of this type include the Jacobi, Gauss-Seidel and Successive Over-Relaxation (SOR) approaches. Whilst these methods are straight forward to construct, use and analyse, convergence can only be guaranteed for matrices adhering to specific criteria. For the interested reader, further information can be found in [206, pp. 63 – 105], where a more rigorous explanation of the above approaches is presented, including convergence analysis.

In comparison, non-stationary methods involve information that continually changes throughout the iterative history of the algorithm. Popular approaches of this type include projection methods, typically involving the orthogonalisation of the residual to (5.1.1) against vectors from an appropriate subspace. In particular, iterates based on projection onto lower dimensional Krylov subspaces of the form

$$\mathcal{K}_m(A, \mathbf{b}) = \text{span}\left\{\mathbf{b}, A\mathbf{b}, A^2\mathbf{b}, \dots, A^{m-1}\mathbf{b}\right\} \qquad (m \in \mathbb{N}, m \leq n), \tag{5.3.1}$$

are commonly studied, and will be considered in this thesis.

5.4 Krylov Subspace Methods

The most common iterative solvers used today for large scale linear systems are Krylov subspace methods. The parentage of such methods can be traced back to 1931 [100], where the mathematician and naval engineer Alexei Krylov sought the solution to eigenvalue problems through the determination of coefficients to the characteristic polynomial of A [25], namely

$$p(A) := c_0 + c_1 A + c_2 A^2 + \dots c_n A^n.$$

The paper describes a method for the efficient computation of the minimum polynomial of a given matrix A, namely the monic[I] polynomial of least degree such that $p(A) = 0$. Later in the 1950s the first Krylov subspace methods were developed, with notable contributions by Arnoldi [7], Hestenes and Stiefel [72] and Lanczos [102], independently of the work produced by Krylov. However, it was not until 1959 that the notion of a *Krylov sequence* was used [80], with Parlett [136] roughly two decades later describing the notion of a *Krylov subspace*, as presented in (5.3.1).

Krylov methods are now among the top ten most important classes of numerical methods [45], and have seen widespread use in many applications of scientific computing where large sparse systems of equations are common. A relatively brief mathematical introduction will now be presented, however for the interested reader, a detailed introduction and derivation of a number of such methods for solving linear equations can be found in [153, pp. 157 – 258].

A Krylov method for solving systems of the form (5.1.1) can be viewed as follows

$$\text{Find } \ \mathbf{x}^{\{m\}} \in \mathcal{K}_m \ \text{ such that } \ \langle \mathbf{w}, \mathbf{r}^{\{0\}} - A\mathbf{x}^{\{m\}} \rangle = 0 \ \text{ for all } \mathbf{w} \in \mathcal{U}_m,$$

where $\mathbf{r}^{\{m\}} := \mathbf{b} - A\mathbf{x}^{\{m\}}$. The specific choice of the subspace $\mathcal{U}_m$ gives rise to different Krylov subspace methods. The reason for projecting onto lower dimensional Krylov subspaces of the form (5.3.1) in this manner can be seen through the characteristic polynomial of A. Through application of the Cayley-Hamilton theorem[II], we have the following

$$d_0 I + d_1 A + \dots + d_n A^n = 0_{n\times n} \implies A^{-1} = \frac{-1}{d_0}\left(d_1 I + d_2 A + \dots + d_n A^{n-1}\right).$$

[I] A monic polynomial is a polynomial such that the coefficient related to the degree of highest order is equal to 1.

[II] The Cayley-Hamilton theorem states that every square matrix A satisfies its own characteristic equation, namely that $p(A) = 0$.

Therefore, when a linear system of the form (5.1.1) is considered, the solution can be formed via

$$\mathbf{x} = p(A)\mathbf{b}, \tag{5.4.1}$$

where the degree of of the polynomial p will not exceed $n-1$. As a result, obtaining a solution to (5.1.1) can be viewed in terms of the calculation of the minimum polynomial to (5.4.1), namely the monic polynomial of minimum degree whereby $p(A)\mathbf{b} = 0$. If this value is denoted m_1, then the subspace $\mathcal{K}_{m_1}$ is invariant under A, with $\mathcal{K}_m = \mathcal{K}_{m_1}$ for all $m \geq m_1$, suggesting the size of the subspace necessary in order to solve (5.1.1). However, the ill-conditioned basis $\mathcal{K}_m$ can lead to computational issues, and one typically looks to locate an orthogonal or bi-orthogonal alternative.

5.4.1 Arnoldi Process

The Arnoldi iterative process involves the factorisation of A in terms of an orthonormal matrix V and an upper Hessenberg matrix H as follows

$$A = VHV^T. \tag{5.4.2}$$

A description of the algorithm can be found in a number of sources, including [192]. In terms of the determination of eigenvalues, the above factorisation can then be coupled with the Rayleigh-Ritz procedure [151, p. 98] to provide a number of eigenpairs of A - a process known as the Arnoldi algorithm. In the context of Krylov subspace methods, the Arnoldi process in essence involves the construction of orthogonal columns of $V := \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m\}$ via the Gram-Schmidt procedure on the Krylov space $\mathcal{K}_m\left(K, \mathbf{r}^{\{0\}}\right)$. Similarly to Gram-Schmidt, the Arnoldi process can not only be implemented computationally within a relatively small number of lines, but can also be terminated early to yield a partial reduction to upper Hessenberg form. At the m^{th} iterative step of the Arnoldi process, we have the factorisation

$$H_m = V_m^T A V_m, \qquad AV_m = V_{m+1}\bar{H}_m, \tag{5.4.3}$$

where the columns of V_m represent an orthonormal basis of $\mathcal{K}_m$. The matrix $\bar{H}_m$ denotes the upper $m \times m$ Hessenberg matrix H_m augmented with an additional $(m+1)^{\text{th}}$ row containing only one non-zero entry, namely $h_{m+1,m}$. When $m = n$ (as per (5.4.2)), the eigenvalues of A and H will coincide. Despite the fact that the matrix is only required within the context of matrix-vector multiplications, a full factorisation can become computationally prohibitive in terms of both application and storage. Nevertheless, for $m < n$, the spectrum of H_m will be contained within the spectrum of A, and so the factorisation described in (5.4.3) can be used to determine certain eigenvalue, eigenvector pairs, with a restarted version possible in the case where specific eigenvalues are either desired, or not required.

The Arnoldi process will used again in the next section, however we close this section by mentioning that in the case where the matrix A is symmetric, the matrix H described in (5.4.2) reduces to a tridiagonal matrix T, with the resulting algorithm known as the Lanczos process [102].

As mentioned, different Krylov subspace methods arise based on the choice of the subspace $\mathcal{U}_m$. Two particular choices will be of interest in this work, with a short description given of each.

- Case 1: $\mathcal{U}_m = \mathcal{K}_m$. In the situation that the matrix A is symmetric positive definite, an inner product (and associated norm) involving A may be defined. Consequently, optimality properties relating to the norm of the error, namely $\|\mathbf{e}^{\{m\}}\|_A := \|\mathbf{x}^* - \mathbf{x}^{\{m\}}\|_A$, are considered, with the term *error projection methods* applied to such algorithms. The most well known method adhering to the above is the Conjugate Gradient (CG) method [72]. Other methods falling under this category include the Orthogonal Residual (ORTHORES) method [207], as well as the Full Orthogonalization Method (FOM) [150]. However, optimality properties in the case where A is non-symmetric cannot be described in general. Whilst one can consider the solution to the normal equations, namely

$$A^T A\mathbf{x} = A^T\mathbf{b},$$

the consequence is a potential loss of sparsity, as well as an amplified condition number. An algorithm similar to ORTHORES has been developed [8], however in general methods developed for non-symmetric matrices can be seen to fall into the next class of candidates for the subspace $\mathcal{U}_m$.

- Case 2: $\mathcal{U}_m = A\mathcal{K}_m$. In this case, the approximate solution to (5.1.1) can be seen to minimise the norm of the residual $\mathbf{r}^{\{m\}}$ over the affine space $\mathbf{x}^{\{0\}} + \mathcal{K}_m$ (see [154]). A number of methods have been developed in this case for non symmetric matrices (for instance, see [9, 51]), however the main focus for this work will be on the Generalised Minimum Residual Method [155].

5.4.2 GMRES

The Generalised Minimum RESidual (GMRES) method was presented initially by Saad and Schultz in 1986 [155] as a generalisation of the MINimum RESidual (MINRES) algorithm developed by Paige and Saunders [133] for non-symmetric systems of the form (5.1.1). The basic idea behind the algorithm is essentially outlined in the title, and can be described in terms of the following least squares problem: Find $\mathbf{x}^{\{m\}} \in \mathcal{K}_m$ in order to minimise the norm of the residual $\mathbf{r}^{\{m\}}$. As formulated, the problem has dimension $n \times m$, however by writing

$$\mathbf{x}^{\{m\}} = \mathbf{x}^{\{0\}} + V_m\mathbf{y}^{\{m\}}, \tag{5.4.4}$$

where the columns of $V_m \in \mathbb{R}^{n\times m}$ represent a basis of $\mathcal{K}_m$, it can be seen that

$$\|\mathbf{r}^{\{m\}}\|_2 = \|\mathbf{b} - A\mathbf{x}^{\{m\}}\|_2 = \|\mathbf{b} - A\mathbf{x}^{\{0\}} - AV_m\mathbf{y}^{\{m\}}\|_2 = \|\mathbf{r}^{\{0\}} - AV_m\mathbf{y}^{\{m\}}\|_2.$$

Through the use of the Arnoldi process, we use the relation (5.4.3) so that

$$\|\mathbf{r}^{\{m\}}\|_2 = \|\mathbf{r}^{\{0\}} - V_{m+1}\bar{H}_m\mathbf{y}^{\{m\}}\|_2.$$

Since $\mathbf{v}_1 := \frac{\mathbf{r}^{\{0\}}}{\|\mathbf{r}^{\{0\}}\|_2}$, we may write $\mathbf{r}^{\{0\}} = \beta V_{m+1}\mathbf{1}_{m+1}$ with $\beta := \|\mathbf{r}^{\{0\}}\|_2$ to give

$$\|\mathbf{r}^{\{m\}}\|_2 = \|\beta V_{m+1}\mathbf{1}_{m+1} - V_{m+1}\bar{H}_m\mathbf{y}^{\{m\}}\|_2 = \|\beta\mathbf{1}_{m+1} - \bar{H}_m\mathbf{y}^{\{m\}}\|_2,$$

using the orthonormality of V_{m+1}. Therefore, the problem has been reduced in size to $(m+1) \times m$. The problem is solved for $\mathbf{y}^{\{m\}}$ using a QR factorisation of the upper Hessenberg matrix $\bar{H}_m$. Initially, this may appear to provide an issue for particularly large m, however by starting at $m = 1$ and considering sequential increments, the QR factorisation of $\bar{H}_m := Q_mR_m$ may be obtained from that of $\bar{H}_{m-1}$ through the use of a single Given's rotation.

Now, the solution to the least squares problem may be viewed as

$$\|\mathbf{r}^{\{m\}}\|_2 = \|\mathbf{z}^{\{m\}} - R_m\mathbf{y}^{\{m\}}\|,$$

where $\mathbf{z}^{\{m\}} := \beta Q_m\mathbf{1}_{m+1}$. As a consequence of the structure of $\bar{H}_m$, the last row of the upper triangular matrix R_m is zero, and so the vector $\mathbf{y}^{\{m\}}$ is obtained through backward substitution with the final entry of $\mathbf{z}^{\{m\}}$ removed, leading to the solution $\mathbf{x}^{\{m\}}$ via (5.4.4).

Direct comparison of GMRES to alternative methods for non-symmetric matrices suggests that the method provides computational savings both in terms of the overall flop count, and also in storage requirements [155]. Additionally, GMRES will not break down unless $h_{m+1,m} = 0$, which is referred to as a *lucky* break down since this suggests that the system (5.1.1) can be solved exactly in a subspace of dimension m.

5.4.3 Drawbacks with Krylov Methods

In the particular case of GMRES, it is clear to see that when $m = n$, $\mathcal{K}_n = \mathbb{R}^n$ and thus, in exact arithmetic, GMRES can be seen to solve (5.1.1) in at most n iterations. However, at this point the operational complexity would amount to direct matrix inversion ($\mathcal{O}(n^3)$), and so the expectation is that the degree of the minimum polynomial of $\mathbf{r}^{\{0\}}$ is significantly less than n. Therefore, approaches should be considered whereby the value of m does not increase to a point at which the algorithm becomes unfeasible to work with. In particular, the storage requirements for the orthonormal basis V_m formed through the Arnoldi process should be taken into account, since this may provide a computational bottleneck for large values of m.

One approach is to consider restarting the Krylov method after a fixed number of iterations with an updated initial guess. Unfortunately in the particular case of GMRES, whilst the optimality property $h_{m+1,m} = 0$ holds as in the original algorithm, relatively straightforward examples can be constructed highlighting the fact that convergence cannot be guaranteed (see, for instance, [155]).

Alternatively, one could look at the reasons as to why certain Krylov methods may require a substantial number of iterations to achieve convergence for specific problems. It is well understood that the convergence rate of any iterative method is generally governed by the distribution of its eigenvalues. For instance, in the case where A is symmetric positive definite with clustered eigenvalues away from the origin, one can expect the conjugate gradient method to determine the solution to systems of the form (5.1.1) in a relatively small number of iterations dependent on the distance of the cluster from the origin [192]. Similarly, in [192, pp. 271 – 274] two examples are provided in the case of GMRES showing both excellent and poor rates of convergence linked to the location of eigenvalues. Therefore, appropriate strategies should be considered in order to improve the spectral properties of the system matrix.

An efficient way to modify the distribution of eigenvalues, and thus aid the performance of an iterative solver, is to consider an appropriate preconditioning strategy for (5.1.1).

This will be the focus of discussion in the next section.

5.4.4 Preconditioning

Based on the observations in the previous section, we now look to describe a suitable preconditioning strategy for (5.1.1). Evidently, an appropriate choice of preconditioning matrix $P \in \mathbb{R}^{n\times n}$ should not only be non-singular, but should also be efficient in terms of computational complexity with regard to inversion, storage requirements and application. In particular, systems of the form (5.1.1) with P as the system matrix should be able to be solved relatively efficiently, as matrix-vector systems of this form will be involved at least once within each step of an iterative algorithm. As mentioned in the previous section, we also seek to improve the spectral properties of the resulting matrix vector system, suggesting that a choice of P should be close to A in some sense to be described.

We begin by first considering three different approaches in order to precondition systems of the form (5.1.1)

$$\begin{cases} P^{-1}A\mathbf{x} = P^{-1}\mathbf{b} & & \text{(Left)} \\ AP^{-1}\tilde{\mathbf{x}} = \mathbf{b} & (\tilde{\mathbf{x}} := P\mathbf{x}), & \text{(Right)} \\ P_1^{-1}AP_2^{-1}\mathbf{x} = P_1^{-1}\mathbf{b} & (\tilde{\mathbf{x}} := P_2\mathbf{x}, P := P_1P_2). & \text{(Split)} \end{cases} \tag{5.4.5}$$

The split case can be viewed in terms of a combination of the other two cases, and is typically used to preserve symmetry.

A description and implementation of each strategy outlined in (5.4.5) applied to both the CG and GMRES approaches can be found in [153, pp. 275 – 293]. Here, it is noted in [153, p. 285] that each of the preconditioned systems described in (5.4.5) have the same spectrum. With regard to GMRES, both the left and split preconditioning strategies consider minimisation with respect to the preconditioned residual, namely

$$\mathbf{r}^{\{m\}} = P^{-1}\left(\mathbf{b} - A\mathbf{x}^{\{m\}}\right),$$

$$\mathbf{r}^{\{m\}} = P_1^{-1}\left(\mathbf{b} - A\mathbf{x}^{\{m\}}\right),$$

respectively. The residual to the original matrix vector system is in both cases unavailable as a by-product of the method, with recovery only possible through direct multiplication by the preconditioner. This may present an issue whenever a stopping tolerance has been prescribed based on the residual to the original problem, potentially leading to unexpected early or delayed termination of the algorithm. As an adaptive stopping tolerance will be used within this work, explicit access to the original residual value is preferable.

In comparison, the resulting algorithm arising from the application of a preconditioner from the right, and also the strategy that will be used within this thesis, considers minimisation over the residual of the original system. This value is calculated explicitly at each iterative step, and essentially represents the main difference in this approach from the others described in (5.4.5).

The application of a right preconditioning strategy also allows for the consideration of a flexible variant, whereby the preconditioner is allowed to change throughout the course of the algorithm [152]. The reason as to why this will be beneficial for the methods within this thesis will become evident within future chapters.

We now present the following result related to GMRES coupled with right preconditioning.

Lemma 5.4.1. *Suppose that $A \in \mathbb{R}^{\hat{n}\times\hat{n}}$ is symmetric positive definite, and $R, P \in \mathbb{R}^{\hat{n}\times\hat{n}}$ are nonsingular matrices such that*

$$\xi_1 \leq \frac{\langle \mathbf{x}, RP^{-1}\mathbf{x}\rangle_A}{\langle \mathbf{x}, \mathbf{x}\rangle_A}, \qquad \frac{\langle RP^{-1}\mathbf{x}, RP^{-1}\mathbf{x}\rangle_A}{\langle \mathbf{x}, \mathbf{x}\rangle_A} \leq \xi_2,$$

where the constants $\xi_1, \xi_2 \in \mathbb{R}_+$. Under these bounds, the GMRES algorithm in the A-inner product delivers, after m iterations, a residual $\mathbf{r}^{\{m\}}$ satisfying

$$\frac{\|\mathbf{r}^{\{m\}}\|_A}{\|\mathbf{r}^{\{0\}}\|_A} \leq \left(1 - \frac{\xi_1^2}{\xi_2^2}\right)^{m/2}.$$

Proof. Omitted, but can be found, for instance, in [51, 153]. □

The following result is also included, as described by Ipsen in [82].

Proposition 5.4.1. *Suppose that $A, P \in \mathbb{R}^{\hat{n}\times\hat{n}}$ are written as*

$$A = \begin{pmatrix} A_1 & A_2 \\ A_3 & A_4 \end{pmatrix}, \qquad P = \begin{pmatrix} A_1 & A_2 \\ 0 & S \end{pmatrix},$$

with A_1, A_2, A_3 and A_4 denoting appropriate matrix blocks. and S the Schur complement of A_1 in A. The product AP^{-1} corresponds to

$$AP^{-1} = \begin{pmatrix} I & 0 \\ A_3A_1^{-1} & I \end{pmatrix},$$

where the minimum polynomial of AP^{-1} is $p(\lambda) = (\lambda - 1)^d$. Based on this preconditioning strategy, an iterative method such as GMRES is expected to converge in at most d iterations.

The result outlined in Proposition 5.4.1 represents an optimal[I] preconditioner for the matrix A in the sense of requiring the fewest number of iterations in order to determine a solution to (5.4.5) whilst exploiting the block structure of the matrix. The inverse of the proposed preconditioner can be written and applied as follows

$$P^{-1} = \begin{pmatrix} A_1^{-1} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & -A_2 \\ 0 & I \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & S^{-1} \end{pmatrix}, \tag{5.4.6}$$

suggesting the need to calculate the inverse to both the matrix A_1 and also the Schur complement. This issue will be discussed in greater detail within the next chapter, however inversion and storage of the dense Schur complement matrix represents a potential cause for concern. With this in mind, an approximation to P, namely

[I] Evidently, an optimal choice of preconditioner amounts to direct inversion of the matrix A, with convergence realised in a single iteration. Nevertheless, the block matrix approach described in Proposition 5.4.1 allows for practical application, as well as exploitation of certain blocks. This latter point will be seen to be of particular importance in the two chapters to follow.

$$\tilde{P} = \begin{pmatrix} A_1 & A_2 \\ 0 & \widetilde{S} \end{pmatrix},$$

should be considered, with $\widetilde{S}$ representing an approximation to the Schur complement that is not only practical to invert, but can also be seen to provide an appropriate preconditioning strategy for the problem at hand.

For this work, the preconditioner $\tilde{P}$ will be based on a non-overlapping decomposition of the domain into subdomains, as described in the next section.

5.5 Domain Decomposition

5.5.1 Introduction

The aim of this chapter is to provide an appropriate description of the application of domain decomposition to to the solution methods described in Chapter 4. The notion of domain decomposition can be thought of in terms of the division of a problem into a set of smaller problems posed on certain regions of the original domain. The inception of domain decomposition within the mathematical formulation of the problem can occur at either the continuous level, applied directly to the underlying partial differential equation, or at a later stage where the underlying partial differential equation is approximated through the use of an appropriate discretisation method. In the latter case, the algebraic system resulting (for instance) from a finite element discretisation of the domain is often substantial in size. Underlying mathematical formulations for problems in engineering, for example, typically involve the solution to an algebraic system arising from the discretisation of a partial differential equation in three dimensions involving millions of degrees of freedom. Therefore, the development of efficient computational codes within the field of continuum mechanics is desired, with particular mention given to problems in elasticity and fluid dynamics, for which associated models can be seen to govern resulting equations for a wide range of problems.

As described in Section 5.2, the use of direct methods for such problems can present computational issues, not only in terms of complexity but also with regard to storage, even for methods that are able to exploit sparsity. Whilst iterative methods represent a viable alternative, a substantial number of iterates for large scale problems may be needed in the absence of an appropriate preconditioning strategy. The use of domain decomposition allows for the splitting of systems of the form (5.1.1) into smaller problems posed on subspaces so that instead of solving a single problem with a large number of degrees of freedom, a number of substantially smaller problems of the form (5.1.1) may be solved to provide local solutions which, when collected together appropriately, should provide the solution to the original problem. Solving in this manner can be seen to provide a number of advantages. For instance, in the presence of heterogeneous operators, such an approach can be seen to provide benefits due to the changing nature of the underlying partial differential equation over the course of the domain. Additionally, for problems posed on complex geometries, domain decomposition allows for the domain to be divided into a number of subdomains, with resulting subproblems posed on more manageable domains.

The application of domain decomposition tends itself towards the framework of parallel computing, whereby smaller problems may be solved concurrently on multiple computer resources. Computer systems and hardware are subject to continual improvement. Nevertheless, components such as chip boards have a finite space, and so physical improvements are only possible up to a particular point. The notion of computing in parallel can therefore be viewed as a design compromise involving the use of multiple computer resources in tandem to address the aforementioned issue. In comparison to solving problems in serial, appropriate use of parallel computing can lead to reduction in the overall elapsed time to a point where an optimal complexity may be realised. Additionally, parallel computing provides an avenue for the determination of solutions to problems that may be computationally intractable in serial, suggesting that problems may be modelled using a

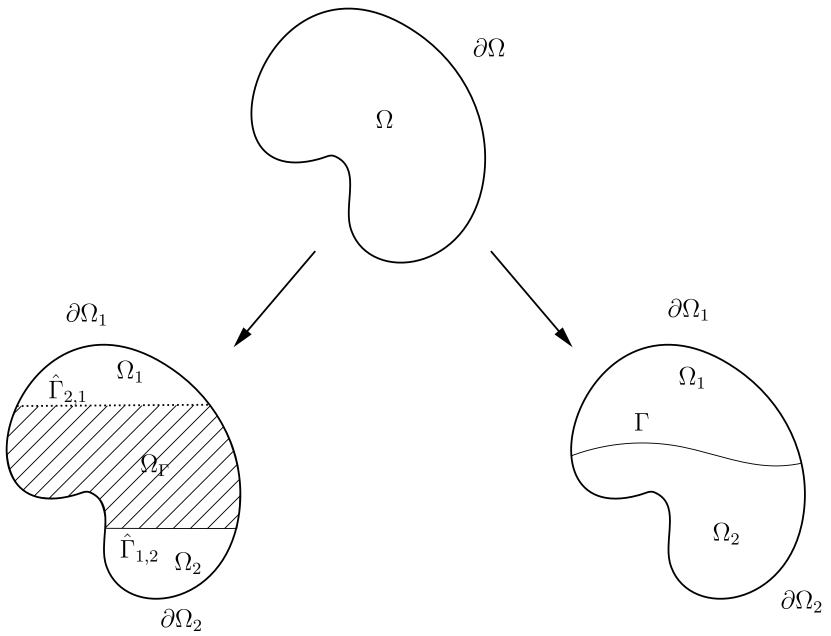


Figure 5.1: Two typical examples of both an overlapping (left) and non-overlapping (right) decomposition of the domain Ω.

significantly increased number of degrees of freedom than previously possible.

A domain may be decomposed in either an overlapping or non-overlapping nature, with an illustrative example in the case of two subdomains provided in Figure 5.1. On the left, the domain Ω is decomposed into Ω_1 and Ω_2 such that $\Omega_1 \cup \Omega_2 = \Omega$ and $\Omega_1 \cap \Omega_2 = \Omega_\Gamma$, where Ω_Γ can be interpreted as an interfacial domain. The boundary to Ω_k is denoted $\partial\Omega_k$, where $\hat{\Gamma}_{1,2} = \partial\Omega_1 \cap \Omega_2$ denotes the interface between Ω_1 and Ω_2, and $\hat{\Gamma}_{2,1} = \partial\Omega_2 \cap \Omega_1$ the interface between Ω_2 and Ω_1. In comparison, the figure on the right illustrates a typical non-overlapping domain, decomposed into Ω_1 and Ω_2 with $\Omega_1 \cup \Omega_2 = \Omega$ and $\partial\Omega_1 \cup \partial\Omega_2 = \partial\Omega$ as in the overlapping case. Additionally, we enforce the constraint $\Omega_1 \cap \Omega_2 = \emptyset$, with the interface between domains Ω_1 and Ω_2 being denoted as Γ. The division described above into two subdomains is included as a convenient example in order to provide an appropriate illustration, however the generalisation to N subdomains may be achieved analogously.

The next two sections will aim to provide a brief introduction and historical background to both overlapping and nonoverlapping decompositions. In particular, the books by Toselli and Widlund [191], Smith, Bjorstad and Gropp [174], and Quarteroni and Valli [143] offer a much deeper insight into theoretical and practical applications of domain decomposition, albeit from mathematically different points of view. The presentation in the first two aforementioned references focuses on a more algebraic and algorithmic point of view, whereas the latter reference discusses topics from a mainly analytic point of view. In addition to the above references, articles by Xu [202], LeTallec [105], Chan and Mathew [37] and Xu and Zhou [203] provide further details on the subject for the interested reader.

Remark 5.5.1. *The definition of the subspaces usually encompasses geometrical properties within the localised problem (subdomain). For instance, the subdomain boundary, the resultant interface between subdomains, vertices and faces would all be included.*

5.5.2 Overlapping Methods

In 1869, Schwarz [166] released an important paper which is now believed to contain the first description of an algorithm designed to utilise domain decomposition. In this paper, an investigation into the existence of harmonic functions for domains Ω with nonsmooth boundaries $\partial\Omega$ was considered. In particular, Schwarz considered problems involving Laplace's equation coupled with Dirichlet boundary conditions as follows

$$\begin{cases} -\Delta \bar{u} = 0 & \text{in } \ \Omega, \\ \bar{u} = \bar{g} & \text{on } \partial\Omega. \end{cases} \tag{5.5.1}$$

The domain $\Omega := \Omega_1 \cup \Omega_2$ was constructed in an overlapping fashion as per Figure 5.1 where existence of harmonic functions was assumed in both Ω_1 and Ω_2. A replica of the original drawing provided by Schwarz is illustrated in Figure 5.2 (namely an overlapping circle and rectangle). The iterative approach proposed by Schwarz involved successive distribution of overlapping data, as described for the linear elasticity problem in Algorithm 5.1 based on an initial guess on $\hat{\Gamma}_{12}$. A notable characteristic of elliptic partial differential

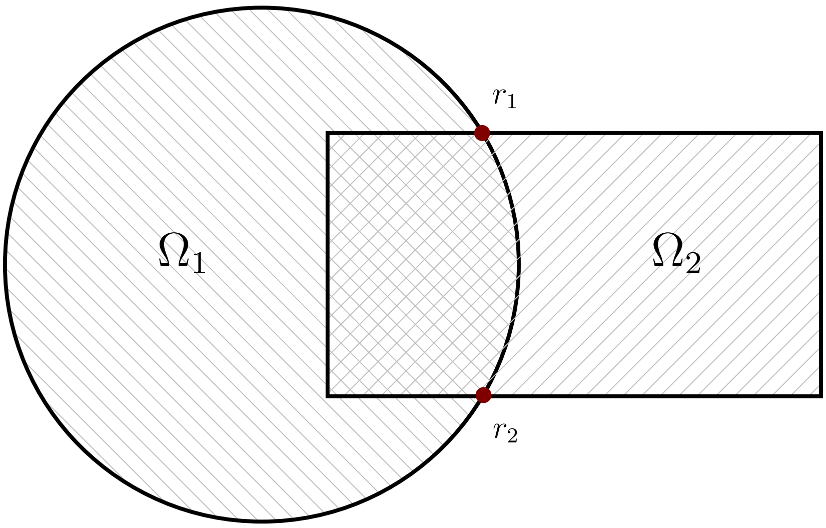


Figure 5.2: Replica of the original sketch by Schwarz in [166] for two 'simple' domains, namely a circle and a rectangle.

equations (such as the linear elasticity problem) is that the solution at each point will depend upon global conditions. In Algorithm 5.1, data is transferred from one subdomain to another in the region of overlap. Convergence analysis was provided by Schwarz in the same paper [166] for problems of the form (5.5.1) using this observation, where the rate of convergence was shown to be directly proportional to the measure of the overlap region Ω_Γ. The proof involved the use of the maximum principle, and was presented in an informal fashion by considering the operation of a vacuum pump; an explanation of this is provided by Gander in [61, p. 3]. Nevertheless, certain issues were noted in the proof that would require further explanation, such as the ambiguity of an arbitrarily small region of overlap between subdomains at the points r_1 and r_2 in Figure 5.2. In this situation, the maximum principle should be used with some care. Additionally, an article by Nevanlinna [131] is cited by Korneev and Langer [94] in relation to convergence analysis of the Schwarz overlapping method - again using the maximum principle.

The variational setting of the Schwarz overlapping method described in Algorithm 5.1 was first presented by Sobolev in 1936 [175] in order to solve linear elasticity problems by utilizing an alternating minimisation procedure in both subdomains. By writing the

Algorithm 5.1 *SCHWARZ METHOD FOR THE LINEAR ELASTICITY PROBLEM*

```
1. $u^B := u_{|\hat{\Gamma}_{12}}$. Solve
   $\begin{cases} \mathcal{L}u = f & \text{in } \Omega_1, \\ u = 0 & \text{on } \partial\Omega_1 \cap \partial\Omega_D, \\ \boldsymbol{\sigma}(u) \cdot \mathbf{n}_1 = g & \text{on } \partial\Omega_1 \cap \partial\Omega_N, \\ u = u^B & \text{on } \hat{\Gamma}_{12}. \end{cases}$
2. $u^B := u_{|\hat{\Gamma}_{21}}$. Solve
   $\begin{cases} \mathcal{L}u = f & \text{in } \Omega_2, \\ u = 0 & \text{on } \partial\Omega_2 \cap \partial\Omega_D, \\ \boldsymbol{\sigma}(u) \cdot \mathbf{n}_2 = g & \text{on } \partial\Omega_2 \cap \partial\Omega_N, \\ u = u^B & \text{on } \hat{\Gamma}_{21}. \end{cases}$
3. Check for convergence. If not satisfied, return to Step 1.
```

space $V_0 = V_1 \oplus V_2$, with

$$V_k := \left[H^1_{\partial\Omega_D \cap \partial\Omega_k}(\Omega_k)\right]^d,$$

we define local bilinear forms $a_k(\cdot,\cdot)\colon V_k \times V_k \to \mathbb{R}$ as follows

$$a_k(u,v) := \int_{\Omega_k} \boldsymbol{\sigma}(u) : \boldsymbol{\epsilon}(v)\,\mathrm{d}\mathbf{x}.$$

Coupled with an appropriate initial guess, the above spaces and local bilinear forms allow for the description of the variational counterpart to the Schwarz overlapping method in two subdomains as outlined in Algorithm 5.2.

Sobolev was able to prove that the solution to this algorithm corresponds to the solution arising from the variational formulation of the original problem, namely (2.2.6). The main advantage of Sobolev's result was that it was now possible to apply the Schwarz

Algorithm 5.2 *SCHWARZ METHOD (VARIATIONAL FORM) - LINEAR ELASTICITY PROBLEM*

1. $w := u$. *Find* $u \in V_1$ *such that for all* $v \in V_1$

$$a_1(u,v) = \int_{\Omega_1} f v dx + \int_{\partial\Omega_1 \cap \partial\Omega_N} g v ds - a_1(w,v).$$

2. $w = w + u$. *Find* $u \in V_2$ *such that for all* $v \in V_2$

$$a_2(u,v) = \int_{\Omega_2} f v dx + \int_{\partial\Omega_2 \cap \partial\Omega_N} g v ds - a_2(w,v).$$

3. $u := u + w$. *Check for convergence. If not satisfied, return to Step 1.*

method to problems without enforcing the restriction of a maximum principle on the operator. Furthermore, the analysis could now be extended to include numerous other mathematical models, as well as other methods derived from the calculus of variables.

Fifteen years later in 1951, Mikhlin [122] was able to illustrate uniform convergence for every closed subdomain $\Omega_k \subset \Omega$. Since then a number of different studies have been carried out, in particular by Morgenstern [124] in 1956, Babuška [10] and Browder [29] in 1958, and more recently Lions [110, 111, 112], with the latter author also providing a proof for the convergence of the overlapping Schwarz method. The proof utilised the variational formulation of the problem previously provided by Sobolev, and was illustrated in an efficient way using projection operators [110]. Importantly, the proof by Lions was described without the requirement that the underlying operator satisfy the maximum principle, allowing the proof to be extended to an overlapping Schwarz method involving N subdomains as opposed to just two. Through this extension, one is then able to solve a number of subproblems in parallel. There are numerous ways to define such a method, often based on the nature of the decomposition, and hence why the discussion above is restricted to cases involving just two subdomains for ease of presentation. However, one should take care to ensure that, in an overlapping Schwarz method, the boundary data of subdomains is correctly updated at each step so that the newest data is used by subdomains when obtaining solutions.

A key factor in the Schwarz overlapping methods discussed up until now is the need for an overlap region. As mentioned previously, Schwarz discussed how the rate of convergence depended upon the size of the overlap region. However, increasing the size of the overlap region will evidently incur extra computational costs. Therefore, it is natural to consider the case where there is no overlapping region between subdomains, with such a formulation naturally lending itself towards parallelism. In this situation however, nodal values lying on the interface cannot be correctly updated in Algorithm 5.1, ultimately resulting in a non-differentiable interfacial solution. Therefore, alternative approaches are sought to alleviate this issue. Dinh et al [44], Bjørstad and Widlund [21], as well as others including Lions [112] have proposed a number of methods in the case where a decomposition is constructed with no region of overlap. The next section aims to present an introduction to such approaches.

5.5.3 Nonoverlapping Methods

The idea of a non-overlapping decomposition of the domain can be seen to originate from the field of mechanical engineering, where approaches were considered in order to analyse complex structures using finite elements. Kron [99] introduced the concept of *diakoptics* through the study of electrical networks, which roughly translates (from Greek) as the method of tearing, involving the splitting of a problem into subproblems. These subproblems would then be solved independently, with the solutions to each being returned to form the solution to the original problem. Kron was able to illustrate instances where the total number of operations taken to solve all of the subproblems and the corresponding connecting interface problem was actually fewer than the number of operations taken to solve the original problem.

The notion of *substructuring* was introduced by Przemieniecki [142] in 1963. Przemieniecki considered the structural analysis of aircraft, where it was found that not enough memory was available in order to store the full structural analysis of the object. Therefore, Przemieniecki considered dividing the aircraft into sections in such a way that the subproblems generated by the division could fit into the available memory. The noticeable difference between the studies by Przemieniecki and Kron is that Przemieniecki attempts to solve potentially unsolvable problems by considering subproblems that can be solved, however Kron is concerned with obtaining the solution to a large problem faster by reducing it to smaller problems. Analysis using the former approach is commonly referred to in literature as numerical scalability, whereas analysis using the latter approach is classified by the term parallel scalability.

In order to describe the substructuring approach, we consider a division of the domain Ω into N nonoverlapping subdomains Ω_k with local boundaries $\partial\Omega_k$. The resulting skeletal interface Γ is formed via $\Gamma = \cup_{k=1}^{N}\Gamma_k$, where $\Gamma_k := \partial\Omega_k \backslash \partial\Omega$. The index $I := \bar{\Omega}\backslash\Gamma$ denotes the set of interior nodes, allowing matrix-vector systems of the form (5.1.1) to be written as

$$A\mathbf{x} = \begin{pmatrix} A_{II} & A_{I\Gamma} \\ A_{\Gamma I} & A_{\Gamma\Gamma} \end{pmatrix} \begin{pmatrix} \mathbf{x}_I \\ \mathbf{x}_\Gamma \end{pmatrix} = \begin{pmatrix} \mathbf{b}_I \\ \mathbf{b}_\Gamma \end{pmatrix} = \mathbf{b},$$

where $A_{II} := \bigoplus_{k=1}^{N} A_{kk}$ and $\mathbf{x}_I := \mathbf{x}_{|\Omega_I}$. Through Gaussian elimination, the solution $\mathbf{x} := \left(\mathbf{x}_I^{(1)}, \mathbf{0}\right)^T + \left(\mathbf{x}_I^{(2)}, \mathbf{x}_\Gamma\right)^T$ may be obtained through the following $2N+1$ subproblems

$$\begin{cases} A_{kk}\mathbf{x}_k^{(1)} = \mathbf{b}_k & k = 1, \ldots, N, \\ \quad S\mathbf{x}_\Gamma = \mathbf{b}_\Gamma - \sum_{k=1}^{N} A_{\Gamma k}\mathbf{x}_k^{(1)}, & \\ A_{kk}\mathbf{x}_k^{(2)} = -A_{k\Gamma}\mathbf{x}_\Gamma & k = 1, \ldots, N. \end{cases} \tag{5.5.2}$$

The size of each of the $2N$ systems presented in the first and third lines of (5.5.2) decrease in size with an increasing number of subdomains. However, an issue with the substructuring approach is that as the number of subdomains are increased, there will be an increase in the number of nodal elements lying on the interface and thus the formation of the Schur complement $S := A_{\Gamma\Gamma} - A_{\Gamma I}A_{II}^{-1}A_{I\Gamma}$ can be seen to present computational issues, both in terms of time and storage.

In order to allieviate this concern, iterative approaches in place of direct matrix inversion may be considered in order to provide a suitable approximation to the solution of (5.5.2). A number of such approaches were developed during the 1970s, with the motivation that the Schur complement is only required in the context of matrix-vector multiplication, avoiding the need to explicitly form S. These iterative substructuring methods form the basis of modern nonoverlapping domain decomposition methods, typically coupled with an appropriate preconditioning strategy. The Schur complement matrix can be seen to correspond to the matrix representation of the so-called Steklov-Poincaré operator within the chosen basis. Investigation into this operator was initially considered by Lebedev and Agoshkov in 1983 [106] for certain elliptic partial differential equations subject to boundary conditions (including linear elasticity and Stokes equations, amongst others). This has since been followed by various works by a number of different authors, since an understanding of the underlying Steklov-Poincaré operator allows for the derivation of an appropriate preconditioning strategy.

With regard to the Schur complement matrix, a result described (for instance) by Brenner [27] applicable to elliptic partial differential equations subject to boundary conditions shows that the condition number of the Schur complement matrix corresponds to

$$\kappa_2(S) = \frac{\lambda_{\text{max}}(S)}{\lambda_{\text{min}}(S)} = \mathcal{O}\left(H^{-1}h^{-1}\right),$$

with H denoting the diameter of the largest subdomain. Since the mesh size h will be noticeably less than H, the condition number of the Schur complement will be substantially lower than the condition number of the original system matrix A, namely $\kappa_2(A) = \mathcal{O}\left(h^{-2}\right)$. Nevertheless, it is clear that the condition number of S carries a dependence on both parameters h and H, and therefore a preconditioner P is sought that aims to improve (or even remove) the dependence on one or both quantities. As mentioned in Section 5.4.3, the performance of iterative solution methods will (in general) be linked to the distribution of the eigenvalues of the system matrix at hand. Therefore, we look to define a symmetric positive definite matrix P as a preconditioner such that

$$\underline{\gamma} \leq \frac{\lambda^T S \lambda}{\lambda^T P \lambda} \leq \overline{\gamma},$$

with both $\underline{\gamma}$ and $\overline{\gamma}$ representing positive constants such that the dependence of $\kappa_2\left(SP^{-1}\right)$ on at least one of the parameters h or H is significantly reduced. Additionally, the preconditioner should computationally efficient to both apply and store, and should not be overbearing to the overall complexity of the method. Finally, the preconditioner should carry an element of parallelism so that the computational savings obtained through solving the local $2N$ subproblems are not lost on the resulting interface problem. A wide range of different preconditioning strategies can be found in the literature - for this thesis, work will concentrate on associated norms to fractional Sobolev spaces.

The application of domain decomposition to problems in topology optimisation is still in its infancy, with only relatively sparse literature accessible at the time of writing. The next section aims to compare and contrast current findings in order to provide an overview of existing studies.

5.5.4 Literature Review - Applications in Topology Optimisation

At present, there are a number of references that consider the application of domain decomposition to problems in topology optimisation. In [23], Borrvall and Petersson use domain decomposition in order to obtain solutions to three dimensional topology optimisation problems in parallel. The paper provides a useful insight into the effectiveness of the use of a Preconditioned Conjugate Gradient (PCG) solver for solving topology optimisation problems in parallel. Their work involved consideration of the VTS problem (3.5.8), coupled with an appropriate penalty term in the objective function of the nested formulation in order to force the solution towards a $0-1$ final design. The considered solution method is based on fixed point iterations using MMA for the density update, in a similar manner to the presentation of Algorithm 4.3 in Chapter 4. A regularised intermediate density control method was employed in order to ensure both existence and uniqueness of solutions. Parallel implementations are provided for not only the solution to the equilibrium equations[I], but also for the optimality procedure. The optimisation process is performed in parallel through exploitation of convexity due to linearisation, with the solution obtained through consideration of the associated dual problem. For the solution to the equilibrium equations, a parallel variant of the CG algorithm is considered involving a block Gauss-Seidel type approach based on an overlapping decomposition of the domain coupled with a Jacobi, or diagonal, preconditioner. The numerical implementation involved calculation of an initial density value using the VTS formulation in the absence of penalisation in order to provide a suitable initial guess for the penalised problem (as in [24]). A number of test cases were considered, including designs for a cantilever beam, crank and coat hanger. However, despite the fact that promising results were produced free from numerical anomalies, certain issues were noted. The regularised penalty was seen to struggle with small filter radius values[II], and the penalty coefficient was found to depend on the problem at hand. Calculation of the correct penalty coefficient required numerical experimentation, which added to the overall complexity of the presented algorithm. The paper also mentions that, even though both the optimisation procedure and the equilibrium equations were computed in parallel, the main bulk of the computational effort (approximately 97%) was found to be contained in the determination of the solution to the equilibrium equations.

The application of PCG with Jacobi preconditioning has also been considered in [55, 87, 115, 195]. In [115], two dimensional topology optimisation problems were solved by Mahdavi et. al. The OC solution method, as outlined in Algorithm 4.2, was the solution method of choice coupled with strict penalisation to enforce $0-1$ final designs (i.e. $\mu = 3$ in (3.5.4a)), as well as an appropriate filtering procedure for the design

[I] As described by the first constraint in (3.5.7).

[II] As per Chapter 3, the notion of filtering is only discussed briefly within this thesis. For a more in-depth presentation, the interested reader should consult [16, pp. 35 – 36].

sensitivities for existence and uniqueness. Parallel implementations for both the finite element analysis and the calculation of sensitivities are considered. No attempt was made to parallelise either the optimality criteria update or the filtering procedure for the design sensitivities, as it is reported (along with [23]) that approximately 97% of the computational complexity involved within the presented method is contained within these two areas. Furthermore, the fact that filtering is not local by nature suggests potential difficulties regarding implementation.

A description of a parallel filtering procedure was however achieved by Vemaganti and Lawrence in [195]. Here, unlike other papers, Hilbert space-filling curves were used to decompose the domain. What is interesting in this study is that alternative approaches to the PCG method with Jacobi preconditioning are considered and compared. For relatively small problems, it was found that the PCG method with Jacobi preconditioning provided satisfactory results. However, if a preconditioner based on an Incomplete LU factorization (ILU) was considered instead, the ill-conditioning of the global stiffness matrix was dealt with more effectively. Despite this, both preconditioners exhibited poor parallel efficiencies. Instead, a non-overlapping approach was found to deliver the strongest results, whereby the matrix-vector system describing the equilibrium equations written as per (5.5.2) was solved, with a direct solver used for the solution to individual subdomain problems, and PCG with Jacobi preconditioning for the resulting interface problem. This approach was much more efficient when considered in parallel, and was also able to deal with the ill-conditioning of the global stiffness matrix in a manner comparable to the ILU preconditioner. This paper also illustrated the effects of loosening the PCG tolerance on the convergence of the solution. It was found that it was possible to loosen the tolerance to an extent, however one should take care as errors can be accumulated in successive OC iterations in cases where the tolerance is too tight.

A substructuring approach was also considered by Evgrafov et. al. in [55] involving the use of a Finite Element Tearing and Interconnecting Dual Primal (FETI-DP) solver [56], specifically designed for the determination of solutions to large-scale systems arising from high resolution two or three dimensional finite element models. Whilst the focus of existing studies can be seen to centre on attempting to solve fixed size problems faster by increasing the number of subdomains (parallel scalability), this study draws focus to the concept of numerical scalability. Arguably, due to the inherent large-scale nature of the problems within the field, the purpose of the integration of parallel computing for the solution of topology optimisation problems may not be to arrive at solutions faster, but instead to determine solutions to problems that cannot be computed using serial computing alone - in particular, problems posed on complex three-dimensional domains. A method can be seen to be numerically scalable provided that there is only linear growth in the computational complexity with respect to the size of the problem. For domain decomposition methods, it is common knowledge within the community that numerical scalability cannot be achieved without consideration of a coarse space component [88, 198]. This study looks to exploit the numerical scalability of FETI that can be shown to hold for a variety of elliptic PDEs ([88, 117]), with a number of two and three dimensional examples considered including an MBB beam and also a compliant mechanism. However, it was found that this bound did not translate well for problems in topology optimisation due to the increasing heterogeneity of the subdomains, leading to large differences between the norms of the primal and dual residuals within the algorithm. Modifications to the preconditioning strategy in order to incorporate the heterogeneity were considered. Nevertheless, despite improvements to the conditioning obtained through the use of Jacobi-type preconditioning, only minor improvements were noted when used with FETI-DP.

More recently, a parallel topology optimisation framework was proposed in a paper by Aage and Lazarov [1]. As in [23], the solution method used here is similar to the presentation provided in Algorithm 4.3, with motivation from recent studies into the parallelisation and subsequent use of GPUs for optimisation problems in both heat conduction [196] and also linear elasticity [164]. The paper provides parallel implementations for the solution to both the state equations and also for the density update through consideration of the dual to (4.4.1). The domain was decomposed using METIS [85], a set of programs used (in this work) to partition the finite element mesh into non-regular subdomains. Optimisation problems in both solid and fluid mechanics were solved, with promising results displaying parallel scalability. In particular, the determination of the solution to the state equations involved the use of either preconditioned MINRES or PCG dependent on the problem at hand, with the preconditioner based on a Factorised Sparse Approximate Inverse (FSAI) [93]. Whilst the derived method was not seen to be numerically scalable, a decrease in the overall computational time was noted for problems solved using an increasing number of CPUs, providing a basis for further development.

In addition to the aforementioned references, studies have also been considered whereby parallel computing has been considered in slightly different circumstances with regard to topology optimisation. In [87], Kim, Kim and Kim use parallel computing in order to deal with topology optimisation for eigenvalue problems. The solution method proposed was similar to what was seen in [23], involving the communication of common degrees of freedom between processors. Both the numerical analysis and the optimisation procedure were written in parallel in such a way that the communication required between subdomains was kept to a minimum. Their work nicely illustrated the benefits of parallel computing through application to large-scale structural design problems that, in general, would be computationally demanding.

In general, literature on topology optimisation seems to centre around problems with a standard minimum compliance/maximum stiffness formulation. However, it can be argued that there are certain disadvantages with this approach. For instance, final designs can be infeasible in practise due to the fact that no constraints are imposed on either the stresses or the displacements. Also, minimum compliance problems can be said to be ill-posed due to the oscillatory nature of the solution as the discretised mesh is refined further and further. In [41], a minimum weight topology optimisation problem with added stress constraints is considered, as opposed to the more general minimum compliance/maximum stiffness problem. While such a formulation may be more realistic from a practical point of view, one of the drawbacks with this approach in general is that rather than just having to deal with a single linear inequality, one now has to consider a large number of highly non-linear constraints as illustrated in [38, 39, 49, 108, 126, 134, 135, 141, 204], for example. As a result, there is an increase in the computational effort devoted to both the structural analysis and, in particular, the sensitivity analysis. Therefore, [41] considers the application of parallel computing to this particular formulation. The study gives an optimisation methodology for solving such problems, and considers areas of the procedure that are computationally demanding. Both the first order sensitivity analysis of the stress constraints and the search direction by means of Sequential Linear Programming (SLP) algorithms are parallelised. The first of these was found to be particularly efficient in parallel, since the derivatives of each local stress constraint can be computed on different processors. The parallelisation of the SLP algorithm, on the other hand, involved some interprocessor communication. This meant that as the problem was divided into an increasing number of subdomains, the computational time that was spent dealing with interprocessor communication increased. This can be attributed to Amdahl's law, which, for fixed size problems, states that as we increase the number of processors, the speed-up of the algorithm approaches a theoretical limit due to the increasing time taken in interprocessor communication as well as other non-parallel tasks. Despite this, however, parallel computing focussed on these two particular areas of the optimisation methodology proved to be useful and notable speed-up figures were obtained.

In [116], Makrizi et.al. considered a modified weak formulation of the standard minimum compliance problem to include domain decomposition. In this case, the domain is divided into two subdomains and a penalty term is included in the formulation in order to ensure continuity of the displacements across the interface. While there is proof of the theoretical concepts, no computational results are supplied and it is a little unclear how to extend this formulation in order to consider a further division of the domain.

The use of interior point methods for problems in structural optimisation involving the simultaneous update of both design and state variables has been considered previously by both Hoppe and Petrova [75, 76], Hoppe, Linsenmann and Petrova [74], and also Hoppe, Petrova and Schultz [78]. Their work involved a preconditioning strategy based on a null-space formulation. The latter three authors also considered the application of domain decomposition within structural optimisation in order to solve optimisation problems in electromagnetics [77], however no numerical results are provided in the study.

Based on the current literature, the bulk of the computational effort within fixed point type solution methods for topology optimisation can be seen to reside within the equilibrium equations. At each fixed point step, we therefore require an effective solution method for the equations of linear elasticity, forming the basis of investigation in the next chapter. The linearity present within the associated PDE allows for decoupling, and by decomposing the domain into N nonoverlapping subdomains, the original problem can be viewed in terms of $2N + 1$ subproblems. The task will then be to derive an appropriate preconditioning strategy for the resulting interface problem in order to arrive at a numerically scalable algorithm. Theoretical as well as numerical justification will be required in order to implement the method within either the OC or the MMA solution methods described in Algorithms 4.2 and 4.3.

Further investigation will involve analysis of the Jacobian matrices arising in the primal-dual interior point methods described in Algorithm 4.4. The presence of the stiffness matrix in both the original and reduced system matrices suggests potential application of findings for the case of linear elasticity. However, adaptations may be necessary in order to incorporate other constraining blocks arising within the relevant system matrix.

CHAPTER 6

NONOVERLAPPING DOMAIN DECOMPOSITION FOR LINEAR ELASTICITY

6.1 Mathematical Derivation

6.1.1 Classical Formulation

Following on from the literature review at the foot of the previous chapter, we now look to apply domain decomposition to the linear elasticity problem as described in Chapter 2. To do this, we divide the domain Ω into N nonoverlapping subdomains Ω_k with local boundaries $\partial\Omega_k$ and outer unit normals $\mathbf{n}_k$. We denote by Γ the resulting skeletal interface $\Gamma = \cup_{k=1}^{N}\Gamma_k$ where $\Gamma_k := \partial\Omega_k \backslash \partial\Omega$, and by $I := \bar{\Omega}\backslash\Gamma$ the set of interior nodes as per the presentation provided in Section 5.5.3. Assuming the restriction of $u_k := u_{|\Omega_k}$ to components of the skeletal interface $\lambda_k := u_{|\Gamma_k}$ is known, the linear elasticity problem (2.1.1) may be described in terms of the following set of subproblems

$$\begin{cases} \mathcal{L}u_k = f_k & \text{in } \Omega_k, \\ u_k = 0 & \text{on } \partial\Omega_D \cap \partial\Omega_k =: \mathcal{D}_k, \\ \boldsymbol{\sigma}(u_k) \cdot \mathbf{n}_k = g_k & \text{on } \partial\Omega_N \cap \partial\Omega_k =: \mathcal{N}_k, \\ u_k = \lambda_k & \text{on } \Gamma_k, \end{cases}$$

where $k = 1, \dots, N$. By writing the displacements variables as $u = u^{(1)} + u^{(2)}$, we exploit the linearity of the stress tensor to form two sets of N subproblems as described below

$$\begin{cases} \mathcal{L}u_k^{(1)} = f_k & \text{in } \Omega_k, \\ u_k^{(1)} = 0 & \text{on } \partial\Omega_D \cap \partial\Omega_k, \\ \boldsymbol{\sigma}(u_k^{(1)}) \cdot \mathbf{n}_k = g_k & \text{on } \partial\Omega_N \cap \partial\Omega_k, \\ u_k^{(1)} = 0 & \text{on } \Gamma_k. \end{cases} \tag{6.1.1a}$$

$$\begin{cases} \mathcal{L}u_k^{(2)} = 0 & \text{in } \Omega_k, \\ u_k^{(2)} = 0 & \text{on } \partial\Omega_D \cap \partial\Omega_k, \\ \boldsymbol{\sigma}(u_k^{(2)}) \cdot \mathbf{n}_k = 0 & \text{on } \partial\Omega_N \cap \partial\Omega_k, \\ u_k^{(2)} = \lambda_k & \text{on } \Gamma_k. \end{cases} \tag{6.1.1b}$$

Evidently, the governing equations of linear elasticity may be split in a number of different ways in order to encompass a decomposition of the domain. Along with the potential to solve in parallel, the well-posedness of the subproblems presented in (6.1.1) suggests the splitting as described is both a meaningful as well as useful approach to pursue.

Owing to the construction of the second set of problems (6.1.1b), we define matrix extension operators H_k that map interface data to relevant subdomains via $u_k^{(2)} = H_k\lambda_k$. The associated weak solution to each of the subproblems defined in (6.1.1b) can then be shown to adhere to the following elliptic regularity result (as described, for instance, in [3])

$$\|u_k^{(2)}\|_{1,\Omega_k} = \|H_k\lambda_k\|_{1,\Omega_k} \leq c_{\text{REG}}\|\lambda_k\|_{1/2,\Gamma_k}, \tag{6.1.2}$$

with c_{REG} denoting a constant. The first set of N subproblems around $u^{(1)}$ involve the determination of solutions to problems posed locally on subdomains. Factored into each subproblem is the associated right hand side as well as potential Neumann data for the subdomains, where the intersection between the respective subdomain boundary and global Neumann boundary is nonempty. The second lot of N subproblems can be seen to involve the solution to linear elasticity problems defined on subdomains with zero right hand side and homogeneous boundary conditions applied on boundaries of subdomains bordering $\partial\Omega$.

The task is now to construct an appropriate system that can be used to solve for λ (where $\lambda_{|\Gamma_k} = \lambda_k$), and ultimately decompose the problem into three steps involving a total of $2N+1$ subproblems. Through multiplication of the first equation in (6.1.1a) by $v \in [H^1_D(\Omega)]^d$ followed by integration over the relevant subdomain and the application of the specified boundary conditions, we look to determine $u_k^{(1)} \in V_k := \left[H^1_{\mathcal{D}_k}(\Omega_k)\right]^d$ such that for all $v \in [H^1_D(\Omega)]^d$

$$a_k(u_k^{(1)}, v_k) = \int_{\Gamma_k} \left(\boldsymbol{\sigma}(u_k^{(1)}) \cdot \mathbf{n}_k\right) v_k \,\mathrm{d}s + \int_{\mathcal{N}_k} g_k v_k \,\mathrm{d}s + \int_{\Omega_k} f_k v_k \,\mathrm{d}\mathbf{x}, \tag{6.1.3}$$

where $v_k := v_{|\Omega_k}$ and the local bilinear counterpart to (2.2.5), $a_k(\cdot,\cdot)\colon V_k \times V_k \to \mathbb{R}^d$, corresponds to

$$a_k(u,v) := \int_{\Omega_k} \boldsymbol{\sigma}\,(u) : \boldsymbol{\epsilon}(v)\,\mathrm{d}\mathbf{x}.$$

Analogously, for each k we look to determine $u_k^{(2)} \in [H^1(\Omega_k)]^d$ such that for all $v \in [H^1_D(\Omega)]^d$

$$a_k(u_k^{(2)}, v_k) = \int_{\Gamma_k} \left(\boldsymbol{\sigma}(u_k^{(2)}) \cdot \mathbf{n}_k\right) v_k \,\mathrm{d}s. \tag{6.1.4}$$

In order to generate an appropriate equation for λ, we sum together the equations provided in (6.1.3) and (6.1.4) for $k = 1, \dots, N$, and compare the result to the original weak form provided in (2.2.6). In order for the respective formulation to match the original problem, we require

$$\sum_{k=1}^{N} \left[\int_{\Gamma_k} \left(\boldsymbol{\sigma}(u_k^{(2)}) \cdot \mathbf{n}_k\right) v_k \,\mathrm{d}s\right] = -\sum_{k=1}^{N} \left[\int_{\Gamma_k} \left(\boldsymbol{\sigma}(u_k^{(1)}) \cdot \mathbf{n}_k\right) v_k \,\mathrm{d}s\right], \tag{6.1.5}$$

namely the solution to a system strictly defined on the interface Γ. By solving this system, the combination of both (6.1.3) and (6.1.4) provides the solution to the problem (2.2.6). Through substitution of the previously defined matrix extension operators H_k into (6.1.5), we arrive at the following

$$\sum_{k=1}^{N}\left[\int_{\Gamma_k}\left(\boldsymbol{\sigma}\left(H_k\lambda_k\right)\cdot\mathbf{n}_k\right)v_k\,\mathrm{d}s\right]=-\sum_{k=1}^{N}\left[\int_{\Gamma_k}\left(\boldsymbol{\sigma}(u_k^{(1)})\cdot\mathbf{n}_k\right)v_k\,\mathrm{d}s\right],$$

which corresponds to solving the system

$$\sum_{k=1}^{N}\boldsymbol{\sigma}\left(H_k\lambda\right)\cdot\mathbf{n}_k=-\sum_{k=1}^{N}\boldsymbol{\sigma}(u_k^{(1)})\cdot\mathbf{n}_k,$$

on Γ. This is referred to as the Steklov-Poincaré equation, where the so called Steklov-Poincaré pseudo differential operator $\mathfrak{S}:\Lambda_\theta\to\Lambda_\theta^*$ can be defined in the following manner

$$\langle\mathfrak{S}\lambda,\mu\rangle:=\sum_{k=1}^{N}\left[\int_{\Gamma_k}\left(\boldsymbol{\sigma}\left(H_k\lambda_k\right)\cdot\mathbf{n}_k\right)\mu_k\,\mathrm{d}s\right]=:\sum_{k=1}^{N}\langle\mathfrak{S}_k\lambda_k,\mu_k\rangle,\tag{6.1.6}$$

where $\lambda,\mu\in\Lambda_\theta$ and $\mu_{|\Gamma_k}=\mu_k$. The space Λ_θ is chosen to be a suitable fractional Sobolev space of index θ based on the boundary conditions of the problem, dependent on the intersection of Γ with $\partial\Omega$. For further details, the interested reader should consult [144, 191]. For this work, the space $\Lambda_\theta=\left[H_0^\theta(\Gamma)\right]^d$ is chosen to represent Λ_θ, however for problems constructed in a different manner (e.g. mixed boundary conditions), Λ_θ could either be defined as $\left[H_{00}^\theta(\Gamma)\right]^d$ or $\left[H^\theta(\Gamma)\right]^d$. The underlying spaces involved in the problem as posed up to this point suggest that the natural choice of the index θ is $1/2$ due to the nature of the trace operator for the spaces under consideration. Whilst certain results to come will be displayed in this particular case only, the numerical results section will suggest alternative values of θ based on the nature of the decomposition, hence the reason for retaining generality.

Using the definition of $\mathfrak{S}$ provided in (6.1.6), the problem (2.1.1) can be written as a sequence of three decoupled problems with corresponding boundary conditions assigned to subdomains and an interface problem defined on Γ as follows

$$\begin{cases} \mathcal{L}u_k^{(1)} = f_k & \text{in } \Omega_k, \\ u_k^{(1)} = 0 & \text{on } \partial\Omega_D \cap \partial\Omega_k, \\ \boldsymbol{\sigma}(u_k^{(1)}) \cdot \mathbf{n}_k = g_k & \text{on } \partial\Omega_N \cap \partial\Omega_k, \\ u_k^{(1)} = 0 & \text{on } \Gamma_k. \end{cases} \tag{6.1.7a}$$

$$\begin{cases} \mathfrak{S}\lambda = -\sum_{i=1}^{N} \boldsymbol{\sigma}(u_k^{(1)}) \cdot \mathbf{n}_k & \text{on } \Gamma. \end{cases} \tag{6.1.7b}$$

$$\begin{cases} \mathcal{L}u_k^{(2)} = 0 & \text{in } \Omega_k, \\ u_k^{(2)} = 0 & \text{on } \partial\Omega_D \cap \partial\Omega_k, \\ \boldsymbol{\sigma}(u_k^{(2)}) \cdot \mathbf{n}_k = 0 & \text{on } \partial\Omega_N \cap \partial\Omega_k, \\ u_k^{(2)} = \lambda_k & \text{on } \Gamma_k. \end{cases} \tag{6.1.7c}$$

6.1.2 Weak Formulation

In order to present the weak and discrete weak formulations to the decomposed problem described in (6.1.7), we consider a change of variables $\tilde{u}_k^{(2)} := u_k^{(2)} - z_k$ where $z_k = \lambda_k$ on Γ_k in order to describe (6.1.7c) in the following way

$$\begin{cases} \mathcal{L}\tilde{u}_k^{(2)} = -\mathcal{L}z_k & \text{in } \Omega_k, \\ \tilde{u}_k^{(2)} = 0 & \text{on } \partial\Omega_D \cap \partial\Omega_k, \\ \boldsymbol{\sigma}(\tilde{u}_k^{(2)}) \cdot \mathbf{n}_k = 0 & \text{on } \partial\Omega_N \cap \partial\Omega_k, \\ \tilde{u}_k^{(2)} = 0 & \text{on } \Gamma_k. \end{cases}$$

Through the use of Green's first identity (see (2.2.2)), the equations of equilibrium in (2.1.1) and also extension operators E_k that map data from the interface to the interior of subdomains, we may present the associated weak form of the right hand side to (6.1.7b) in the following manner

$$-\int_{\Gamma_k} \left(\boldsymbol{\sigma}(u_k^{(1)}) \cdot \mathbf{n}_k\right) v_k \, \mathrm{d}s = \int_{\Omega_k} f_k v_k \, \mathrm{d}\mathbf{x} - \int_{\Omega_k} \boldsymbol{\sigma}(u_k^{(1)}) \colon \boldsymbol{\epsilon}(v_k) \, \mathrm{d}\mathbf{x}$$

$$\begin{aligned} &= \int_{\Omega_k} f_k E_k \mu_k \,\mathrm{d}\mathbf{x} - \int_{\Omega_k} \boldsymbol{\sigma}(u_k^{(1)}) \colon \boldsymbol{\epsilon}(E_k \mu_k) \,\mathrm{d}\mathbf{x} \\ &= F_k(E_k\mu_k) - a_k(u_k^{(1)}, E_k\mu_k), \end{aligned}$$

where $v_k := E_k \mu_k$ and $F_k(\cdot) : V_k \to \mathbb{R}$ corresponds to

$$F_k(v_k) := \int_{\Omega_k} f_k v_k \,\mathrm{d}\mathbf{x}.$$

This allows for the presentation of the associated weak formulation to (6.1.1) as follows

$$\begin{cases} \text{Find } u_k^{(1)} \in V_k \text{ such that for all } v_k \in V_k \\ \qquad\qquad a_k(u_k^{(1)}, v_k) = F_k(v_k). \end{cases}$$

$$\begin{cases} \text{Find } \lambda \in \Lambda_\theta \text{ such that for all } \mu \in \Lambda_\theta \\ \qquad\qquad \langle \mathfrak{S}\lambda, \mu \rangle = \displaystyle\sum_{k=1}^{N} \Big[F_k(E_k\mu_k) - a_k(u_k^{(1)}, E_k\mu_k) \Big]. \end{cases}$$

$$\begin{cases} \text{Find } \tilde{u}_k^{(2)} \in V_k \text{ such that for all } v_k \in V_k \\ \qquad\qquad a_k(\tilde{u}_k^{(2)}, v_k) = -a_k(z_k, v_k). \end{cases}$$

We now present the following result, asserting that $\mathfrak{S}$ in the case $\theta = 1/2$ corresponds to a bounded positive operator on the space $\Lambda_{1/2}$.

Lemma 6.1.1. *Let $\mathfrak{S}$ be defined by (6.1.6). Then, there exist constants α_1, α_2 such that for all $\lambda, \mu \in \Lambda_{1/2}$*

$$\alpha_1 \|\lambda\|_{1/2,\Gamma} \leq \langle \mathfrak{S}\lambda, \lambda \rangle, \qquad \langle \mathfrak{S}\lambda, \mu \rangle \leq \alpha_2 \|\lambda\|_{1/2,\Gamma} \|\mu\|_{1/2,\Gamma}.$$

Proof. Let both $v_k = E_k\lambda_k$ and $w_k = E_k\mu_k$, so that $v_k, w_k \in \left(H^1_{\mathcal{D}_k \cup \mathcal{N}_k}(\Omega_k)\right)^d$. Using Green's first identity (see (2.2.2)), we have that

$$\langle \mathfrak{S}\lambda_k, \lambda_k \rangle = \int_{\Gamma_k} \left(\boldsymbol{\sigma}\left(H_k\lambda_k\right) \cdot \mathbf{n}_k \right) \lambda_k \,\mathrm{d}s$$

$$= \int_{\Omega_k} \boldsymbol{\sigma}(H_k\lambda_k) \colon \boldsymbol{\epsilon}(E_k\lambda_k)\,\mathrm{d}\mathbf{x} + \int_{\Omega_k} \left(\nabla \cdot \boldsymbol{\sigma}(H_k\lambda_k)\right) E_k\lambda_k\,\mathrm{d}\mathbf{x}.$$

By using the splitting of the problem along with the definition of the extension operators H_k, we have that

$$\int_{\Omega_k} \left(\nabla \cdot \boldsymbol{\sigma}(H_k\lambda_k)\right) E_k\lambda_k\,\mathrm{d}\mathbf{x} = 0. \tag{6.1.9}$$

Therefore

$$\langle \mathfrak{S}\lambda_k, \lambda_k\rangle = \int_{\Omega_k} \boldsymbol{\sigma}(H_k\lambda_k) \colon \boldsymbol{\epsilon}(E_k\lambda_k)\,\mathrm{d}\mathbf{x} = a_k(v_k, v_k) \geq c_1 \|v_k\|^2_{1,\Omega_k},$$

due to coercivity. As $\gamma_0 v_k = \lambda_k$ and $\gamma_0 w_k = \mu_k$, the trace inequalities (1.2.12) and (1.2.13) imply that, for $k = 1, \ldots, N$

$$c_1 \|v_k\|^2_{1,\Omega_k} \geq \alpha_1 \|\lambda_k\|^2_{1/2,\Gamma_k},$$

and hence the first result.

For the second inequality, we use (6.1.9) along with the boundedness of the bilinear form to write

$$\begin{aligned}
\langle \mathfrak{S}\lambda_k, \mu_k\rangle &= \int_{\Gamma_k} \left(\boldsymbol{\sigma}\left(H_k\lambda_k\right) \cdot \mathbf{n}_k\right) \mu_k\,\mathrm{d}s \\
&= \int_{\Omega_k} \boldsymbol{\sigma}(H_k\lambda_k) \colon \boldsymbol{\epsilon}(E_k\mu_k)\,\mathrm{d}\mathbf{x} + \int_{\Omega_k} \left(\nabla \cdot \boldsymbol{\sigma}(H_k\lambda_k)\right) E_k\mu_k\,\mathrm{d}\mathbf{x} \\
&= \int_{\Omega_k} \boldsymbol{\sigma}(H_k\lambda_k) \colon \boldsymbol{\epsilon}(E_k\mu_k)\,\mathrm{d}\mathbf{x} = a_k(v_k, w_k) \\
&\leq c_2 \|v_k\|_{1,\Omega} \|w_k\|_{1,\Omega_k} \leq \alpha_2 \|\lambda_k\|_{1/2,\Gamma_k} \|\mu_k\|_{1/2,\Gamma_k},
\end{aligned}$$

with the final inequality holding due to the elliptic regularity result (6.1.2). □

Remark 6.1.1. *The proof described above is an adaptation to the linear elasticity problem from a similar result presented in [5].*

6.1.3 Discrete Weak Formulation

We now consider a discretisation of our decomposed domain Ω, where the space V_h presented in (2.3.1) is divided into local copies $V_{k,h} = [\mathcal{V}_{k,h}]^d \subset V_h$, with

$$\mathcal{V}_{k,h} := \left\{ v \mid v \in C^0(\Omega_k), v_{|\mathcal{C}} \in P_\alpha(\mathcal{C}) \;\; \forall \mathcal{C} \in T_h, v_{|\mathcal{D}_k} = 0 \right\}. \tag{6.1.10}$$

By defining sets

$$\mathcal{N}_k := \left\{ j \in D \,\middle|\, \operatorname{supp}_{\mathcal{V}_{k,h}}(\psi_j) \neq \emptyset \right\} \qquad k = 1, \dots, N,$$

such that $\mathcal{N}_k$ represents the index set corresponding to specific basis functions with support on $V_{k,h}$, and $\mathcal{N}_\Gamma$ the index set for the remaining basis functions representing nodes defined on the interface, we have that

$$\mathcal{V}_{k,h} = \operatorname{span}\{\psi_i,\, i \in \mathcal{N}_k\} \subset H_0^1(\Omega_k) \qquad k = 1, \dots, N.$$

We also define $V_{\Gamma,h} = [\mathcal{V}_{\Gamma,h}]^d \subset V_h$ such that $V_{I,h} \oplus V_{\Gamma,h} = V_h$, where

$$\mathcal{V}_{\Gamma,h} = \operatorname{span}\{\psi_i,\, i \in \mathcal{N}_\Gamma\}.$$

Using the discrete weak formulation presented in (2.3.3), and through the definition of the space $S_h = [\mathcal{S}_h]^d$ such that

$$\mathcal{S}_h := \operatorname{span}\{\gamma_0(\Gamma)\psi_i,\, i \in \mathcal{N}_\Gamma\},$$

we are in a position to describe the discrete weak counterpart to (6.1.1) in the following way

$$\begin{cases} \text{Find } u_{k,h}^{(1)} \in V_{k,h} \text{ such that for all } v_{k,h} \in V_{k,h} \\ \qquad\qquad a_k(u_{k,h}^{(1)}, v_{k,h}) = F_k(v_{k,h}). \end{cases} \tag{6.1.11a}$$

$$\begin{cases} \text{Find } \lambda_h \in S_h \text{ such that for all } \mu_h \in \ S_h \\ s\left(\lambda_h, \mu_h\right) := \langle \mathfrak{S}\lambda_h, \mu_h \rangle = \displaystyle\sum_{k=1}^{N} \left[F_k(E_k\mu_{k,h}) - a_k(u_{k,h}^{(1)}, E_k\mu_{k,h}) \right]. \end{cases} \tag{6.1.11b}$$

$$\begin{cases} \text{Find } \tilde{u}_{k,h}^{(2)} \in V_{k,h} \text{ such that for all } v_{k,h} \in V_{k,h} \\ \qquad\qquad a_k(\tilde{u}_{k,h}^{(2)}, v_{k,h}) = -a_k\left(z_{k,h}, v_{k,h}\right). \end{cases} \tag{6.1.11c}$$

The discrete equivalent to Lemma 6.1.1 is presented below, showing that the bilinear form $s(\cdot,\cdot)\colon S_h \times S_h \to \mathbb{R}$ adheres to both coercivity and continuity bounds.

Lemma 6.1.2. *Let $s(\cdot,\cdot)$ be defined by (6.1.11b). Then, there exist constants α_1, α_2 such that for all $\lambda, \mu \in S_h \subset \Lambda_{1/2}$*

$$\alpha_1 \|\lambda_h\|_{1/2,\Gamma} \leq s\left(\lambda_h, \lambda_h\right), \qquad s\left(\lambda_h, \mu_h\right) \leq \alpha_2 \|\lambda_h\|_{1/2,\Gamma} \|\mu_h\|_{1/2,\Gamma}.$$

Proof. The proof follows a very similar presentation to that given in Lemma 6.1.1. □

6.1.4 Matrix Formulation

By writing $n = n_I + n_\Gamma$, with

$$n_I := |\mathcal{N}_I| = \left| \bigcup_{k=1}^{N} \mathcal{N}_k \right| =: \sum_{k=1}^{N} n_k \quad \text{and} \quad n_\Gamma := |\mathcal{N}_\Gamma|,$$

we define the arrays

$$K_{\mathfrak{ab}} := \begin{pmatrix} K_{\mathfrak{ab}}^{(1,1)} & \dots & K_{\mathfrak{ab}}^{(1,d)} \\ \vdots & \ddots & \vdots \\ K_{\mathfrak{ab}}^{(d,1)} & \dots & K_{\mathfrak{ab}}^{(d,d)} \end{pmatrix} \qquad \mathfrak{a}, \mathfrak{b} \in \{1, \dots, N, \Gamma\},$$

where $K^{(c,d)}_{\mathfrak{a}\mathfrak{b}} \in \mathbb{R}^{n_{\mathfrak{a}} \times n_{\mathfrak{b}}}$ for $\mathfrak{a}, \mathfrak{b} \in \{1, \dots, N, \Gamma\}$ such that

$$\begin{aligned}
\left(K^{(c,d)}_{kk}\right) &= a_k\left(\psi_i, \psi_j\right) && i, j \in \mathcal{N}_k,\\
\left(K^{(c,d)}_{\Gamma\Gamma}\right) &= a\left(\psi_i, \psi_j\right) && i, j \in \mathcal{N}_\Gamma,\\
\left(K^{(c,d)}_{k\Gamma}\right) &= a_k\left(\psi_i, \psi_j\right) && i \in \mathcal{N}_k,\ j \in \mathcal{N}_\Gamma,\\
\left(K^{(c,d)}_{\Gamma k}\right) &= a_k\left(\psi_i, \psi_j\right) && i \in \mathcal{N}_\Gamma,\ j \in \mathcal{N}_k,
\end{aligned}$$

for $c, d \in \mathbb{W}$, allowing for the presentation of the system described in (2.3.4) as follows

$$K\mathbf{u} = \begin{pmatrix} K_{II} & K_{I\Gamma} \\ K_{\Gamma I} & K_{\Gamma\Gamma} \end{pmatrix} \begin{pmatrix} \mathbf{u}_I \\ \mathbf{u}_\Gamma \end{pmatrix} = \begin{pmatrix} \mathbf{f}_I \\ \mathbf{f}_\Gamma \end{pmatrix} = \mathbf{f}. \tag{6.1.13}$$

In comparison, the corresponding matrix formulations of each of the discrete weak formulations to the problem presented in (6.1.1) can be written as

$$\begin{cases} K_{II}\mathbf{u}_I^{(1)} = \mathbf{f}_I, \\ \quad S\mathbf{u}_\Gamma = \mathbf{f}_\Gamma - K_{\Gamma I}\mathbf{u}_I^{(1)}, \\ K_{II}\mathbf{u}_I^{(2)} = -K_{I\Gamma}\mathbf{u}_\Gamma, \end{cases} \tag{6.1.14}$$

where $\mathbf{u} = \left(\mathbf{u}_I^{(1)}, \mathbf{0}\right)^T + \left(\mathbf{u}_I^{(2)}, \mathbf{u}_\Gamma\right)^T$. The matrix representation of the bilinear form $s(\cdot, \cdot)$ with respect to the basis of S_h corresponds to S, which effectively amounts to the Schur complement of the matrix K_{II} in K (see Section 1.2.1), namely

$$S = K_{\Gamma\Gamma} - K_{\Gamma I}K_{II}^{-1}K_{I\Gamma}. \tag{6.1.15}$$

Therefore, we may view the discretisation of the decoupled problem (6.1.1) as a Schur complement approach to the discretisation of the original system (2.3.4). Since

$$K_{II} = \bigoplus_{k=1}^{N} K_{kk},$$

the systems described in (6.1.14) may be written in terms of $2N+1$ systems as follows

$$\begin{cases} K_{kk}\mathbf{u}_k^{(1)} = \mathbf{f}_k & k = 1,\ldots,N, \\ S\mathbf{u}_\Gamma = \mathbf{f}_\Gamma - \displaystyle\sum_{k=1}^{N} K_{\Gamma k}\mathbf{u}_k^{(1)}, & \\ K_{kk}\mathbf{u}_k^{(2)} = -K_{k\Gamma}\mathbf{u}_\Gamma & k = 1,\ldots,N. \end{cases} \tag{6.1.16}$$

We therefore look to solve (6.1.13) by exploiting the potential for parallelisation present in (6.1.16).

6.2 Preconditioning

6.2.1 Observations

Using the presentation in Section 5.4.4, our preconditioner for the matrix-vector system described in (6.1.13) will have the following form

$$\tilde{P} = \begin{pmatrix} K_{II} & K_{I\Gamma} \\ 0 & \widetilde{S} \end{pmatrix}, \tag{6.2.1}$$

with $\widetilde{S}$ representing an approximation to the interface Schur complement S. Based on $\tilde{P}$ written in this manner, the inverse can be seen to be formed as the product of the following three matrices

$$\tilde{P}^{-1} = \begin{pmatrix} K_{II}^{-1} & 0 \\ 0 & I_{\Gamma\Gamma} \end{pmatrix} \begin{pmatrix} I_{II} & -K_{I\Gamma} \\ 0 & I_{\Gamma\Gamma} \end{pmatrix} \begin{pmatrix} I_{II} & 0 \\ 0 & \widetilde{S}^{-1} \end{pmatrix}. \tag{6.2.2}$$

The operations around the application of P^{-1} would initially involve the block inversion of K_{II} on subdomains. Next, a boundary-to-domain update would be applied through $K_{I\Gamma}$, before applying the action of $\widetilde{S}$ on the skeleton problem corresponding to the internal boundary. With the exception of $\widetilde{S}$, the potential for parallelism within both (6.2.1) and

(6.2.2) is evident. Therefore, the task is to seek an appropriate representation to $\widetilde{S}$ so that the preconditioner can be assembled, applied and stored in an efficient manner.

The form of $\widetilde{S}$ used within this work will be based on the spectral equivalence between the continuous and discrete norm topologies associated with the spaces Λ_θ and S_h respectively. In order to describe the relevant norms further, we first introduce the notion of an interpolation space.

6.2.2 Interpolation Spaces

Viewed from an analytical point of view, an interpolation space can be viewed as a space lying between two other spaces. Such spaces can often be found in the mathematical description of various modelling applications, with examples found in domain decomposition methods [28, 46, 144], elasticity [62], advection-diffusion problems [158, 159, 160] and image processing [125, 130, 187], to name a few. Therefore, by quantifying the corresponding discrete spaces, alternative numerical approaches and algorithms may be developed. This was considered in work by Arioli and Loghin [6], who were (in particular) able to derive discrete norm representations for projections of fractional Sobolev spaces onto suitable finite dimensional subspaces. Examples involving standard pairs typically arising in the formulation of elliptic PDEs were presented, with one example involving Poisson's equation particularly relevant to this work. The study goes on to provide matrix representations for the discrete norms written in terms of powers of Grammian matrix products associated with the bases of the relevant finite element spaces under consideration.

Based on the presentation in [6], we now refer to [109] in order to introduce the notion of interpolation between abstract Hilbert spaces. We begin by defining Hilbert spaces[I] X and Y such that $X := \mathcal{X}^d$ and $Y := \mathcal{Y}^d$, where the subset X of Y is both dense and continuously embedded within Y. Let $\langle \cdot, \cdot \rangle_X$, $\langle \cdot, \cdot \rangle_Y$ represent the associated inner products, and $\| \cdot \|_X$, $\| \cdot \|_Y$ the respective norms. Now, by the Riesz representation theory

[I] We also require both $\mathcal{X}$ and $\mathcal{Y}$ to be Hilbert spaces, but the associated inner products will not be required in the presentation to follow.

(illustrated in [146]), there exists a positive self-adjoint operator $\mathcal{P} : X \to Y$ such that

$$\langle u, v\rangle_X = \langle u, \mathcal{P}v\rangle_Y \qquad u, v \in X. \tag{6.2.3}$$

Through consideration of Lemma 1.2.2, we use the spectral decomposition of the operator $\mathcal{P}$ in order to define an operator

$$\mathcal{E} := \mathcal{P}^{1/2} : X \to Y,$$

which is also self adjoint in Y with domain $D(\mathcal{E})$. The corresponding norm of X is equivalent to the graph norm $\|\cdot\|_{\mathcal{E}}$

$$\|u\|_X \sim \|u\|_{\mathcal{E}} := \left(\|u\|_Y^2 + \|\mathcal{E}u\|_Y^2\right)^{1/2}.$$

In practice, use of the spectral decomposition of $\mathcal{E}$ allows for the definition of any real power $\mathcal{E}^\theta$ of $\mathcal{E}$. Therefore, by considering $\theta \in [0, 1]$, we define $\|\cdot\|_\theta$ to be the scale of graph norms

$$\|u\|_\theta := \left(\|u\|_Y^2 + \|\mathcal{E}^{1-\theta}u\|_Y^2\right)^{1/2}.$$

Now, as in [109, p. 255], it can be shown that the domain of $\mathcal{E}^{1-\theta}$ equipped with the inner product

$$\langle u, v\rangle_\theta = \langle u, v\rangle_Y + \langle u, \mathcal{E}^{1-\theta}v\rangle_Y,$$

represents a Hilbert space which, for the pair $[X, Y]$, corresponds to an interpolation space of index θ (denoted $[X, Y]_\theta$) as follows

$$[X, Y]_\theta := D(\mathcal{E}^{1-\theta}) \qquad \theta \in [0, 1].$$

From this definition, it is clear to see that $[X, Y]_0 = X$ and $[X, Y]_1 = Y$[I]. Furthermore,

[I] When $\theta = 0$, we have $[X, Y]_0 = D(\mathcal{E}) = X$. Similarly, when $\theta = 1$ we have $[X, Y]_1 = D(0) = Y$

for $0 < \theta_1 < \theta_2 < 1$, we note that

$$X \subset [X, Y]_{\theta_1} \subset [X, Y]_{\theta_2} \subset Y.$$

Now, by defining $T(\bar{X}; \bar{Y})$ as the space of continuous linear operators from $\bar{X}$ to $\bar{Y}$, we may present the following important interpolation theorem

Theorem 6.2.1. *Let X, Y be defined as above and let $\bar{X}, \bar{Y}$ be Hilbert spaces satisfying similar properties. Let π be a continuous linear operator mapping X to $\bar{X}$, and Y to $\bar{Y}$, ie:*

$$\pi \in T(X; \bar{X}) \cap T(Y; \bar{Y}).$$

Then, for all $\theta \in (0, 1)$,

$$\pi \in T\left([X, Y]_\theta; [\bar{X}, \bar{Y}]_\theta\right).$$

Proof. Omitted, however can be found in [109, p. 27]. □

The next section will focus on generating the scale of interpolation spaces from finite dimensional subspaces of X and Y, so that discrete norms for such spaces can be derived.

6.2.3 Finite Dimensional Interpolation Spaces

We now consider the case where $[\mathcal{X}_h]^d = X_h \subset X$, $[\mathcal{Y}_h]^d = Y_h \subset Y$ represent finite dimensional subspaces of X and Y, respectively. Suppose that the corresponding dimension of both $\mathcal{X}_h$ and $\mathcal{Y}_h$ is n. Then, both $[\mathcal{X}_h]^d$ and $[\mathcal{Y}_h]^d$ will have dimension $\hat{n} = dn$. Both X_h and Y_h correspond to Hilbert spaces equipped with the respective inner products $\langle \cdot, \cdot \rangle_X$, $\langle \cdot, \cdot \rangle_Y$. As in (6.2.3), we use the Riesz representation theory to define the self-adjoint operators $\mathcal{P}_h, \mathcal{E}_h : X_h \to Y_h$ in the following manner

$$\langle u_h, v_h \rangle_X = \langle u_h, \mathcal{P}_h v_h \rangle_Y \qquad u_h, v_h \in X_h, \tag{6.2.4}$$

where $\mathcal{E}_h = \mathcal{P}_h^{1/2}$. Additionally, we also define the discrete interpolation spaces as

$$[X_h, Y_h]_\theta := D(\mathcal{E}_h^{1-\theta}),$$

and also the scale of discrete norms

$$\|u\|_{\theta,h} := \left(\|u_h\|_Y^2 + \|\mathcal{E}_h^{1-\theta} u_h\|_Y^2\right)^{1/2}.$$

The following lemma asserts spectral equivalence between the continuous norm $\|\cdot\|_\theta$ and its discrete counterpart $\|\cdot\|_{\theta,h}$, using a result provided in [6].

Lemma 6.2.1. *For the pairs of Hilbert spaces (X, Y) and (X_h, Y_h), denote by $\|\cdot\|_\theta$ and $\|\cdot\|_{\theta,h}$ norms on the respective interpolation spaces $[X, Y]_\theta$ and $[X_h, Y_h]_\theta$. Define $\mathcal{I}_h \in \mathcal{L}(X; X_h) \cap \mathcal{L}(Y; Y_h)$ such that*

$$\|\mathcal{I}_h u\|_{X_h} \leq c_1 \|u\|_X \quad \forall u \in X, \qquad \|\mathcal{I}_h u\|_{Y_h} \leq c_2 \|u\|_Y \quad \forall u \in Y,$$

with both c_1 and c_2 real constants independent of $\hat{n}$. Under the assumption that $I_h u = u_h$ for all $u_h \in X_h$, both the norms $\|\cdot\|_\theta$ and $\|\cdot\|_{\theta,h}$ are equivalent on $[X_h, Y_h]_\theta$ for $\theta \in (0,1)$, with constants independent of $\hat{n}$.

Proof. Omitted, but can be found in [6]. □

Using the above, we would like to consider matrices $H_\theta \in \mathbb{S}^{\hat{n}}$ (for $\theta \in [0,1]$) which induce norms equivalent to $\|\cdot\|_{\theta,h}$ independent of both parameters d and n. We now define H_X and H_Y to be the Grammian matrices associated with the respective inner products $\langle\cdot,\cdot\rangle_X$ and $\langle\cdot,\cdot\rangle_Y$. Under a suitable basis for $\mathcal{X}_h$, the Hermitian matrices H_X, H_Y can be written as block diagonal matrices of size $d \times d$ as follows

$$H_X = \bigoplus_1^d G_X, \qquad H_Y = \bigoplus_1^d G_Y, \tag{6.2.5}$$

where

$$(G_X)_{ij} = \langle \phi_i, \phi_j \rangle_X, \quad (G_Y)_{ij} = \langle \phi_i, \phi_j \rangle_Y, \quad 1 \leq i, j \leq n.$$

Here, $\{\phi_i\}_{1\leq i\leq n}$ represents a basis for $\mathcal{X}_h$ such that

$$\|u_h\|_X = \|\mathbf{u}\|_{H_X}, \qquad \|u_h\|_Y = \|\mathbf{u}\|_{H_Y},$$

where $\mathbf{u}$ denotes the vector of coefficients for u_h with respect to a direct sum basis for $\mathcal{X}_h^d$. We now observe that the Riesz representation illustrated in (6.2.4) can be written as $\mathbf{u}^T H_X \mathbf{v} = \mathbf{u}^T H_Y Q \mathbf{v}$, with $Q := H_Y^{-1} H_X$ representing a product of two matrices from $\mathbb{S}^{\hat{n}}$. Due to the block representation of both H_X and H_Y given in (6.2.5), we can therefore consider the generalised eigenvalue decomposition of Q such that

$$G_X V = G_Y V D, \qquad \text{where} \qquad G_Y = V^T V, \tag{6.2.6}$$

with the eigenvalues for the matrix pencil (G_X, G_Y) contained within the diagonal matrix D. Following the presentation in [6], the matrix representation of the norm $\|\cdot\|_{\theta,h}$ can be written as

$$H_{\theta,h} = H_Y + H_Y Q^{1-\theta} = H_Y + H_Y \left(H_Y^{-1} H_X\right)^{1-\theta}, \tag{6.2.7}$$

which, again using [6], can be shown to be spectrally equivalent to the reduced matrix H_θ, defined as follows

$$H_{\theta,h} \sim H_\theta = H_Y Q^{1-\theta} = H_Y \left(H_Y^{-1} H_X\right)^{1-\theta}. \tag{6.2.8}$$

6.2.4 Interface Interpolation

Here, we recall the splitting of the discrete space V_h such that $V_h = V_{I,h} \oplus V_{\Gamma,h}$, with both $V_{I,h}$ and $V_{\Gamma,h}$ as described in Section 6.1.3. Let ∇_Γ represent the tangential gradient of a scalar function $v(\mathbf{x})$ such that

$$\nabla_\Gamma v(\mathbf{x}) = \nabla v(\mathbf{x}) - \mathbf{n}\left(\mathbf{n} \cdot \nabla v(\mathbf{x})\right),$$

corresponding to the projection of the gradient of v onto the plane tangent to Γ at the point $\mathbf{x} \in \Gamma$. This allows for a representation of the space $H^1(\Gamma)$ in the following way

$$H^1(\Gamma) = \left\{ v \in L^2(\Gamma) \;\middle|\; \int_\Gamma |\nabla_\Gamma v|^2 \, ds(\Gamma) < \infty \right\}.$$

By introducing $\partial\Gamma_D := \Gamma \cap \partial\Omega_D$ to be the set of points for which the Dirchlet boundary intersects the interface, the space $H^1_{\partial\Gamma_D}(\Gamma)$ can then be presented as

$$H^1_{\partial\Gamma_D}(\Gamma) = \left\{ v \in H^1(\Gamma) \;\middle|\; v_{|\partial\Gamma_D} = 0 \right\},$$

provided that $\partial\Gamma_D$ is non-empty. Now, using the presentation in Section 6.2.2, we define $\mathcal{X} = H^1_{\partial\Gamma}(\Gamma)$, $\mathcal{Y} = L^2(\Gamma)$ and $\mathcal{X}_h = \mathcal{Y}_h = \mathcal{S}_h$, where the Hilbert spaces $\left(S_h, \|\cdot\|_{(H^1_{\partial\Gamma}(\Gamma))^d}\right)$ and $\left(S_h, \|\cdot\|_{(L^2(\Gamma))^d}\right)$ are subsets of X and Y respectively.

By using (1.2.11), the fractional Sobolev space $H^\theta_{00}(\Gamma)$ will correspond to an interpolation space of index $1-\theta$ for the pair $[H^1_{\partial\Gamma}(\Gamma), L_2(\Gamma)]$ as follows

$$H^\theta_{00}(\Gamma) := \left[H^1_{\partial\Gamma_D}(\Gamma), L_2(\Gamma)\right]_{1-\theta}.$$

We define H_X and H_Y to be the Grammian matrices associated with the respective inner products $\langle\cdot,\cdot\rangle_X$ and $\langle\cdot,\cdot\rangle_Y$. Under the aforementioned basis, the Hermitian matrices H_X, H_Y can be expressed as block diagonal matrices of size $d \times d$ by writing

$$H_X = \bigoplus_1^d L_1, \qquad H_Y = \bigoplus_1^d L_0,$$

where each L_k is defined as

$$(L_k)_{i,j} := (\psi_i, \psi_j)_{H^k_{\partial\Gamma}(\Gamma)}.$$

Therefore, L_0 and L_1 can be written in the following way

$$(L_0)_{i,j} = (\psi_i, \psi_j)_{L^2(\Gamma)}, \qquad (L_1)_{i,j} = (\nabla_\Gamma\psi_i, \nabla_\Gamma\psi_j)_{L^2(\Gamma)}. \tag{6.2.9}$$

In the literature, the matrices $L_0\,(=M)$ and $L_1\,(=L)$ are referred to as the interface discrete Mass and Dirichlet Laplacian matrices. Now, using (6.2.7), a norm for the interpolation space $[X_h, Y_h]_\theta$ can be presented as

$$H_\theta = \bigoplus_{1}^{d} \left[M + M\left(M^{-1}L\right)^{1-\theta} \right], \tag{6.2.10}$$

which can be shown to be spectrally equivalent to

$$\widehat{H}_\theta = \bigoplus_{1}^{d} \left[M\left(M^{-1}L\right)^{1-\theta} \right], \tag{6.2.11}$$

based on (6.2.8). In the particular case of $\theta = 1/2$, a combination of both Lemmas 6.1.2 and 6.2.1 imply that for all $\lambda_h \in S_h$ such that $\lambda_h = \sum\limits_{i\in\mathcal{N}_\Gamma} \boldsymbol{\lambda}_i \boldsymbol{\psi}_i$, there exist constants d_1 and d_2 such that

$$d_1 \|\lambda_h\|_{1/2,\Gamma} \leq \|\boldsymbol{\lambda}\|_{H_{1/2}} \leq d_2 \|\lambda_h\|_{1/2,\Gamma}, \tag{6.2.12}$$

leading to the following Lemma, adapted from the presentation in [6]

Lemma 6.2.2. *Let $s(\cdot,\cdot)$ be defined as in (6.1.11b) and $\eta_h, \mu_h \in S_h$ such that $\eta_h = \sum\limits_{i\in\mathcal{N}_\Gamma} \boldsymbol{\eta}_i \boldsymbol{\psi}_i$, $\mu_h = \sum\limits_{i\in\mathcal{N}_\Gamma} \boldsymbol{\mu}_i \boldsymbol{\psi}_i$. Let S denote the matrix representation of $s(\cdot,\cdot)$ in the basis $\{\boldsymbol{\psi}_i\}_{i\in\mathcal{N}_\Gamma}$. Then, there exists constants $e_1, e_2, \hat{e}_1, \hat{e}_2$ such that*

$$e_1 \|\boldsymbol{\eta}\|^2_{H_{1/2}} \leq \boldsymbol{\eta}^T S \boldsymbol{\eta}, \quad \boldsymbol{\mu}^T S \boldsymbol{\eta} \leq e_2 \|\boldsymbol{\mu}\|_{H_{1/2}} \|\boldsymbol{\eta}\|_{H_{1/2}},$$

$$\hat{e}_1 \|\boldsymbol{\eta}\|^2_{\widehat{H}_{1/2}} \leq \boldsymbol{\eta}^T S \boldsymbol{\eta}, \quad \boldsymbol{\mu}^T S \boldsymbol{\eta} \leq \hat{e}_2 \|\boldsymbol{\mu}\|_{\widehat{H}_{1/2}} \|\boldsymbol{\eta}\|_{\widehat{H}_{1/2}},$$

for all $\eta_h, \mu_h \in S_h \subset \Lambda_{1/2}$.

Proof. Omitted, but (using [6]) the result can be seen to follow as a consequence of Lemma 6.1.2 and (6.2.12). □

6.2.5 Mesh Independence

Based on the presentation in Section 6.2.1 and the results outlined in the previous section, we consider the following candidate as a right preconditioner

$$P = \begin{pmatrix} K_{II} & K_{I\Gamma} \\ 0 & \widehat{H}_{1/2} \end{pmatrix}, \tag{6.2.13}$$

The product KP^{-1} can be written as

$$KP^{-1} = \begin{pmatrix} I & 0 \\ K_{\Gamma I}K_{II}^{-1} & S\widehat{H}_{1/2}^{-1} \end{pmatrix}. \tag{6.2.14}$$

As a consequence of the block structure, convergence of an iterative algorithm (such as GMRES) will depend on how well $\widehat{H}_{1/2}$ approximates the Schur complement. In particular, it is clear that the eigenvalues of (6.2.14) will either be equal to one, or agree with one of the eigenvalues of $S\widehat{H}_{1/2}^{-1}$. By describing the $\widehat{H}_{1/2}^{-1}$ field of values of the matrix $S\widehat{H}_{1/2}^{-1}$ as follows

$$\mathcal{F}_{\widehat{H}_{1/2}^{-1}}\left(S\widehat{H}_{1/2}^{-1}\right) := \left\{ z \in \mathbb{C} \;\middle|\; z = \frac{\langle \mathbf{x}, S\widehat{H}_{1/2}^{-1}\mathbf{x}\rangle_{\widehat{H}_{1/2}^{-1}}}{\langle \mathbf{x}, \mathbf{x}\rangle_{\widehat{H}_{1/2}^{-1}}} = \frac{\mathbf{x}^* S \mathbf{x}}{\mathbf{x}^* \widehat{H}_{1/2} \mathbf{x}},\ \mathbf{x} \in \mathbb{C}^{\hat{n}_\Gamma} \setminus \{0\} \right\},$$

we are in a position to postulate the following.

Proposition 6.2.1. *Under the assumption that the conditions in Lemma 6.2.2 are satisfied, the $\widehat{H}_{1/2}^{-1}$ field of values of the matrix $S\widehat{H}_{1/2}^{-1}$ are contained within the right half-plane, bounded independently of $\hat{n}_\Gamma$.*

Proof. Omitted, but essentially follows a similar result to [6]. □

Therefore, while Proposition 6.2.1 fails to encompass the number of considered subdomains (no mention of N), the expectation is that as the size of the problem increases, the number of GMRES iterations taken for comparable decompositions should remain constant. This behaviour will be illustrated with numerical examples in the next section.

Remark 6.2.1. *All of the results described in this section are just as applicable to $H_{1/2}$ as they are to $\widehat{H}_{1/2}$.*

6.2.6 Effective Evaluation of Fractional Powers of Matrices

From the definitions of both H_θ and $\widehat{H}_\theta$ given in (6.2.10) and (6.2.11) respectively, the need to evaluate fractional powers of matrices is evident. For relatively small problems, this can be achieved through the use of a generalised eigenvalue decomposition, as seen in (6.2.6). However, as the size of the interface problem increases, such an approach can be seen to provide computational issues due to carrying a complexity of $\mathcal{O}\left(n_\Gamma^3\right)$. Iterative approaches specifically designed for eigenvalue problems represent a viable alternative. For our work, both $L, M \in \mathbb{S}^{n_\Gamma}$, and so a generalised Lanczos procedure [137] may be used, where at each iterative step m a factorisation of the form

$$LV_m = MV_{m+1}\bar{T}_m, \tag{6.2.15}$$

is available, with the columns of the matrix $V_m = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m] \in \mathbb{R}^{n_\Gamma \times m}$ referred to as Lanczos vectors constructed such that $V_m^T M V_m = I_m$. The matrix $\bar{T}_m \in \mathbb{R}^{(m+1)\times m}$ can be viewed in terms of a symmetric tridiagonal matrix $T_m \in \mathbb{R}^{m\times m}$ augmented with an additional $(m+1)^{\text{th}}$ row containing a single non-zero entry, namely $t_{m+1,m}$.

In exact arithmetic, the algorithm for the pair of matrices (L, M) run to completion (namely $m = n_\Gamma$) provides the following factorisation

$$L = V^{-T}TV^{-1}, \qquad M = V^{-T}V^{-1}, \tag{6.2.16}$$

describing a similarity transformation between the matrix $M^{-1}L$ and T (where the subscript n_Γ has been dropped from both T and V for convenience). Based on the derivations provided in Appendix A.3, this transformation may then be used to write

$$H_\theta = \left(V\left(I_{n_\Gamma} + T^{1-\theta}\right)V^T\right)^{-1} = MV(I_{n_\Gamma} + T^{1-\theta})V^T M, \tag{6.2.17a}$$

$$\widehat{H}_\theta = \left(VT^{1-\theta}V^T\right)^{-1} = MVT^{1-\theta}V^T M. \tag{6.2.17b}$$

In terms of complexity, arriving at the full factorisation described in (6.2.16) will require $\mathcal{O}\left(n_\Gamma^3\right)$ operations. Whilst this may appear to provide no real advantage over the application of a generalised eigenvalue decomposition, the use of an iterative approach allows for early termination, suggesting that effective approximations may be obtained after $m << n_\Gamma$ steps whereby only m Lanczos basis vectors are constructed. Additionally, for our work the inverse of either H_θ or $\widehat{H}_\theta$ will only be required in the context of multiplication to a vector $\bar{\mathbf{z}} := \left(\mathbf{z}^{(c)}\right)_{c\in\mathbb{W}}$. On this basis, we look to use (6.2.17) in order to describe approximations to the matrix-vector products $H_\theta^{-1}\bar{\mathbf{z}}$ and $\widehat{H}_\theta^{-1}\bar{\mathbf{z}}$ as follows

$$\begin{aligned} H_\theta^{-1}\bar{\mathbf{z}} &\approx \bigoplus_1^d \left[V_m\left(I_m + T_m^{1-\theta}\right)^{-1} V_m^T\mathbf{z}\right], \\ \widehat{H}_\theta^{-1}\bar{\mathbf{z}} &\approx \bigoplus_1^d \left[V_m T_m^{\theta-1} V_m^T\mathbf{z}\right]. \end{aligned}$$

Initialising the Lanczos process for the matrix pair (L, M) with $\mathbf{v} = M^{-1}\mathbf{z}$ leads to

$$\begin{aligned} V_m^T\mathbf{z} &= V_m^T M M^{-1}\mathbf{z} \\ &= \mathbf{z}^T M^{-1} M M^{-1}\mathbf{z} \\ &= \mathbf{1}_m \|M^{-1}\mathbf{z}\|_M = \mathbf{1}_m\|\mathbf{z}\|_{M^{-1}}. \end{aligned}$$

Therefore, we have

$$\begin{aligned} H_\theta^{-1}\bar{\mathbf{z}} &\approx \bigoplus_1^d \left[V_m\left(I_m + T_m^{1-\theta}\right)^{-1} \mathbf{1}_m\|\mathbf{z}\|_{M^{-1}}\right], \\ \widehat{H}_\theta^{-1}\bar{\mathbf{z}} &\approx \bigoplus_1^d \left[V_m T_m^{\theta-1}\mathbf{1}_m\|\mathbf{z}\|_{M^{-1}}\right]. \end{aligned}$$

Alternatively, one can consider the Lanczos algorithm applied to the pair of matrices (M^{-1}, L^{-1}), with the main difference to the presentation up to this point being that the matrix V_m is now L^{-1} orthogonal as opposed to being M orthogonal (namely that $V_m^T L^{-1} V_m = I_m$). Whilst this may initially appear to be counter intuitive due to the need for additional matrix inversions, the smallest eigenvalues of the resulting tridiagonal matrix may provide a more effective representation to those of $M^{-1}L$ than the tridiagonal matrix resulting from use of the algorithm with the pair (L, M). As we wish to employ the resulting factorisation as a preconditioner, such a characteristic is desirable. When run to completion, use of the Lanczos algorithm with the pair (M^{-1}, L^{-1}) in exact arithmetic allows both H_θ and $\widehat{H}_\theta$ to be written as follows

$$H_\theta = MV\left(I_{n_\Gamma} + T^{1-\theta}\right)V^T L^{-1}, \tag{6.2.18a}$$

$$\widehat{H}_\theta = MVT^{1-\theta}V^T L^{-1}, \tag{6.2.18b}$$

with the details provided in Appendix A.3. As previously mentioned, application of the inverse of either H_θ or $\widehat{H}_\theta$ will only be required in the context of matrix-vector multiplication. Using (6.2.18), one can show[I] that for the pair (M^{-1}, L^{-1}), we have the following approximations

$$H_\theta^{-1}\bar{\mathbf{z}} \approx \bigoplus_1^d \left[V_m\left(T_m^{-1} + T_m^{-\theta}\right)^{-1} V_m^T L^{-1}\mathbf{z}\right],$$

$$\widehat{H}_\theta^{-1}\bar{\mathbf{z}} \approx \bigoplus_1^d \left[V_m T_m^\theta V_m^T L^{-1}\mathbf{z}\right].$$

Initialising the Lanczos process for the matrix pair (M^{-1}, L^{-1}) with $\mathbf{v} = \mathbf{z}$ gives

$$\begin{aligned} V_m^T L^{-1}\mathbf{z} &= \mathbf{1}_m \mathbf{z}^T L^{-1}\mathbf{z} \\ &= \mathbf{1}_m \|\mathbf{z}\|_{L^{-1}} = \mathbf{1}_m, \end{aligned}$$

[I] Further details in this case may be found in [5], however the presentation differs slightly from what is presented here.

due to the L^{-1} orthogonality of V_m. Using this result, we have that

$$\begin{aligned} H_\theta^{-1}\bar{\mathbf{z}} &\approx \bigoplus_1^d \left[V_m\left(T_m^{-1}+T_m^{-\theta}\right)^{-1}\mathbf{1}_m\right], \\ \widehat{H}_\theta^{-1}\bar{\mathbf{z}} &\approx \bigoplus_1^d \left[V_m T_m^\theta \mathbf{1}_m\right]. \end{aligned}$$

Regardless of the pairs of matrices under consideration, the complexity related to the approximations described will depend on the cost attributed to the inversion and application of both M and L. Whilst the structure of M allows this operation to be carried out in $\mathcal{O}\left(n_\Gamma\right)$ steps, the inversion of L is more demanding and alternative approaches will be discussed in the next section.

6.3 Numerical Results

We present various results in this section in order to illustrate our approach in practice. It should be noted that while the examples considered in this thesis involve a symmetric stiffness matrix, the choice of non-symmetric preconditioner described in (6.2.1) suggests GMRES as an appropriate iterative solver. With reference to [5], it is noted that this preconditioning strategy provides more favourable results than those obtained through standard symmetric Krylov solution methods. Results are shown for both the cantilever beam and the rotating plate problems described in Section 3.6, with zero body force and boundary traction $\mathbf{g} = \mathbf{1}$ being applied in all cases. Figures are provided for various different mesh and subdomain sizes, where a regular subdivision of the domain is considered for the cantilever beam problem. A similar division is considered for the rotating plate example, however relevant subdomains that either cover or contain part of the hole are removed or adapted accordingly. Therefore, certain subdomains will be larger than others for this example under a regular division. For a general non-regular subdivision (not considered within this work), the graph partitioning tool METIS may be used in order to partition the finite element mesh into non-regular subdomains.

The preconditioner is applied using the decomposition described in (6.2.2), where we consider the application of $\widehat{H}_\theta$ using both direct factorisations and also iterative approximations. The results for the different approaches are outlined in Table 6.1 with θ set equal to $1/2$ in all cases. As previously described, a direct factorisation may be realised through a generalised eigenvalue decomposition (denoted $\widehat{H}^{\mathrm{EV}}_{1/2}$), with iterative approximations obtained either through the use of the Lanczos (denoted $\widehat{H}^{\mathrm{L}}_{1/2}$) or the inverse Lanczos (denoted $\widehat{H}^{\mathrm{IL}}_{1/2}$) process, with the flexible variant of GMRES [152] used in order to account for the changing nature of the preconditioner at each Arnoldi iteration.

The row labelled $\widetilde{S} = I$ illustrates results for both test problems in the absence of interface preconditioning. By reading the relevant columns from top to bottom (for each problem), we observe a logarithmic dependence on the number of GMRES iterations for increasing mesh parameters. Reading these columns from left to right also suggests a logarithmic dependence on the number of subdomains. In the cases where an interface preconditioner is applied, we see that, in all cases, the number of GMRES iterations remains roughly constant as the mesh is refined, suggesting that solutions are obtained independently of the chosen mesh parameter.

Overall, preconditioning with $\widetilde{S} = \widehat{H}^{\mathrm{EV}}_{1/2}$ appears to consistently provide the lowest number of iterations for both problems. Nevertheless, use of either iterative approximation has the potential to provide significant computational savings for particularly high resolution problems. By directly comparing results for $\widetilde{S} = \widehat{H}^{\mathrm{L}}_{1/2}$ and $\widetilde{S} = \widehat{H}^{\mathrm{IL}}_{1/2}$, we see that preconditioning through the use of the inverse Lanczos process appears to provide more promising results. In particular, only 10 Lanczos basis vectors were required in both examples to provide an effective set of results. In comparison, the results obtained through the Lanczos process required a number of basis vectors dependent on the size of the interface problem, which for this work was determined through experimentation and set as $m = 2\sqrt{n_\Gamma}$.

Illustrated in Table 6.2 are results for selected values of θ based on experimentation, with the associated results for the cantilever beam problem recorded in Table 6.1 displayed

		Cantilever beam					Rotating plate			
$\widetilde{S}$	No. Subdomains:	4	16	64	256		4	16	60	240
I	$h = 1/16$	22	36	50	74	$h = 1/32$	25	52	83	123
	$1/32$	28	47	68	100	$1/64$	36	72	115	173
	$1/64$	41	66	96	138	$1/128$	48	100	161	243
	$1/128$	59	93	137	181	$1/256$	68	140	226	340
$\widetilde{S} = \widehat{H}^{\mathrm{EV}}_{1/2}$	$h = 1/16$	11	18	27	41	$h = 1/32$	9	16	25	36
	$1/32$	12	18	27	41	$1/64$	9	16	25	36
	$1/64$	12	18	28	41	$1/128$	9	16	24	36
	$1/128$	13	19	28	41	$1/256$	9	16	24	34
$\widetilde{S} = \widehat{H}^{\mathrm{L}}_{1/2}$	$h = 1/16$	12	18	28	41	$h = 1/32$	9	16	25	39
	$1/32$	13	19	28	42	$1/64$	9	16	25	38
	$1/64$	13	19	28	42	$1/128$	9	16	25	38
	$1/128$	14	19	29	41	$1/256$	9	16	24	37
$\widetilde{S} = \widehat{H}^{\mathrm{IL}}_{1/2}$	$h = 1/16$	12	18	28	41	$h = 1/32$	9	16	25	38
	$1/32$	12	17	25	41	$1/64$	10	17	26	36
	$1/64$	12	19	26	40	$1/128$	11	18	24	32
	$1/128$	13	19	28	39	$1/256$	12	19	27	35

Table 6.1: Number of GMRES iterations required to determine a suitably approximate solution to the linear elasticity problem under different interface preconditioning strategies. The case $\widetilde{S} = I$ describes the situation in which no interface preconditioning is applied. A direct approach through the use of a generalised eigenvalue decomposition is considered in the case where $\widetilde{S} = \widehat{H}^{\mathrm{EV}}_{1/2}$, with iterative approximations through the use of the Lanczos process and inverse Lanczos process described in the case where $\widetilde{S} = \widehat{H}^{\mathrm{L}}_{1/2}$ and $\widetilde{S} = \widehat{H}^{\mathrm{IL}}_{1/2}$ respectively.

in the first column of Table 6.2 to allow for a direct comparison. Whilst there is a slight compromise in the mesh independent performance seen previously, it is noted that different choices of θ can provide improved results when compared to those obtained by simply setting θ equal to $1/2$. This suggests that different values of θ are able to provide a closer approximation to the decay in the inverse of the associated Steklov-Poincaré operator.

In order to observe the computational benefits of our method, we look to provide rough estimates in order to gauge how our derived approach will perform in a parallel environment. Due to the non-overlapping nature of our approach, all subdomain solves can be carried out in parallel. As mentioned previously, the main issue surrounds the solution

	$\widetilde{S} = \widehat{H}^{\mathrm{IL}}_{1/2}$				$\widetilde{S} = \widehat{H}^{\mathrm{IL}}_{\mathrm{OPT}}$			
No. Subdomains:	4	16	64	256	4	16	64	256
θ:	0.5	0.5	0.5	0.5	0.5	0.6	0.7	0.75
$h = 1/16$	12	18	28	41	12	16	21	27
1/32	12	17	25	41	12	17	21	27
1/64	12	19	26	40	12	18	22	28
1/128	13	19	28	39	13	20	23	29

Table 6.2: Results to highlight the potential improvements in the preconditioner through use of different θ values. The values of θ recorded in the table were determined through experimentation. Results are shown only for the cantilever beam problem, however similar characteristics for the rotating plate problem hold analogously.

to the resulting interface problem. Within each application of our preconditioner to this problem, we are required to invert the discrete interface Laplacian. This issue is present in the Lanczos process, and also in the subsequent generalised eigenvalue decomposition that follows. Due to the structure of this matrix, these inversions can lead to a computational bottleneck for an increasing number of subdomains, and so we would like to consider an iterative approach to alleviate this issue.

The structure of the involved matrix suggests conjugate gradient as a suitable alternative, coupled with an appropriate preconditioning strategy (PCG). In this work, we propose to precondition by using the relevant contributions of L restricted to Γ_i, with the cross points removed to enable construction in parallel. The parallel CPU time taken for each GMRES iteration can then be realised by dividing the number of PCG iterations (#PCG) multiplied by the CPU time taken to apply the preconditioner (T_{Prec}) by the total number of faces involved in the construction of Γ (#Faces). By adding this contribution to the CPU time taken for one parallel subdomain solve (T_{SD}), we calculate the total CPU time by multiplying the result to the total number of GMRES iterations required to achieve convergence (#GMRES), so that the total CPU time may then be realised through the following formula

$$\text{CPU Time} = \#\text{GMRES} \times \left(T_{\mathrm{SD}} + \frac{\#\text{PCG} \times T_{\mathrm{Prec}}}{\#\text{Faces}} \right).$$

	$\widetilde{S} = \widehat{H}^{\mathrm{IL}}_{\mathrm{OPT}}$			
No. Subdomains:	4	16	64	256
θ:	0.5	0.6	0.7	0.75
$h = 1/16$	0.0169	0.0168	0.0176	0.0254
1/32	0.0635	0.0273	0.0199	0.0232
1/64	0.4455	0.1238	0.0384	0.0238
1/128	3.8716	1.0804	0.2529	0.0623
1/256	50.2476	13.2858	3.3204	0.7295

Table 6.3: Total CPU times (seconds) anticipated through the use of parallel computing for the cantilever beam problem.

The results for the investigations are displayed in both Tables 6.3 and 6.4, where CPU times (in seconds) are provided for differing mesh and subdomain sizes for each respective problem. From the tables, it can be seen that for relatively coarse meshes, we do not see a significant enough decrease in the CPU time to warrant the use of parallelism. This behaviour can be attributed to the computational complexity of sparse matrix inversion ($\mathcal{O}\left(k_B^2 \hat{n}\right)$, k_B is the bandwidth) for relatively small values of $\hat{n}$, and also the efficiency of the backslash command in MATLAB. However, notable savings in CPU time equating to roughly factor 4 for the cantilever beam problem can be seen for finer meshes. Slightly reduced speedup is noted (and anticipated) for the rotating plate problem due to the nature of the decomposition. Nevertheless, these figures are encouraging, as they suggest

	$\widetilde{S} = \widehat{H}^{\mathrm{IL}}_{\mathrm{OPT}}$			
No. Subdomains:	4	16	64	256
θ:	0.5	0.6	0.7	0.75
$h = 1/32$	0.1661	0.0436	0.0205	0.0211
1/64	1.4306	0.4149	0.1750	0.0950
1/128	10.2418	2.6437	0.7406	0.2082
1/256	123.2458	33.4478	9.2016	2.3681

Table 6.4: Total CPU times (seconds) anticipated through the use of parallel computing for the rotating plate problem.

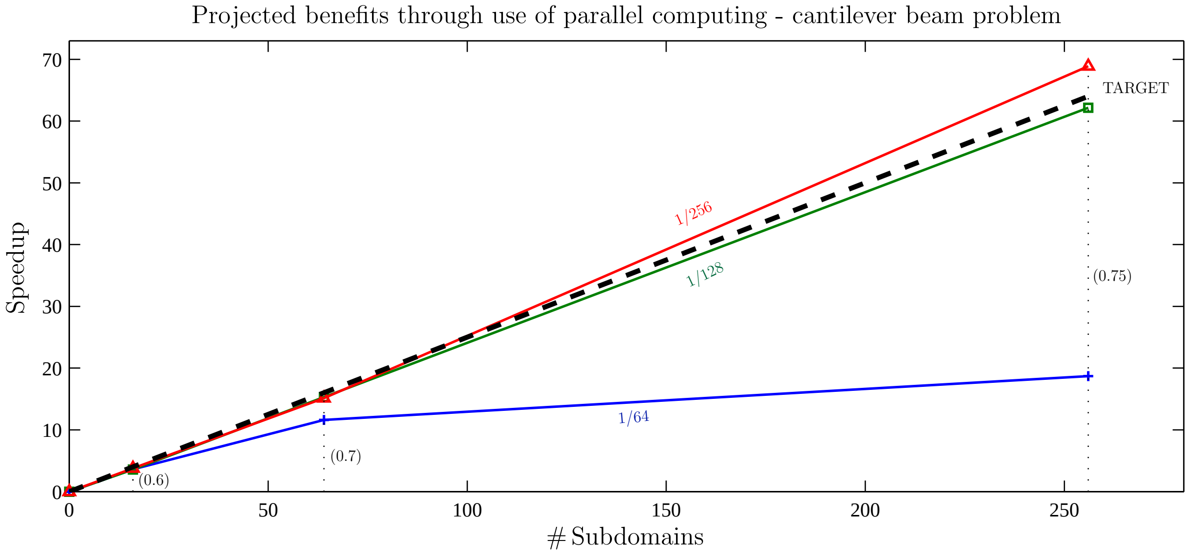


Figure 6.1: Graph to illustrate the anticipated speedup based on the figures presented in Table 6.3. The target line suggests the desired speedup for an increasing number of subdomains (processors).

that our approach is capable of significant speedup through the use of parallel architecture when compared directly to solving the problem globally on a single processor.

After collating the results in Tables 6.3 and 6.4, a general increase was noted in the number of GMRES iterations when compared directly to figures obtained through direct inversion of the interface Laplacian (as shown in Table 6.2 for the cantilever beam problem). The reason for this can be attributed to the use of inner PCG iterations. In particular, a logarithmic dependence on the mesh parameter was observed for cases involving smaller numbers of subdomains. However, the deterioration can be seen as an acceptable compromise, as the results for larger meshes suggest the use of an increasing number of subdomains for improved performance. It should be noted that the results obtained above were done so with a relatively coarse tolerance for PCG of 10^{-3}, as well as a reasonably modest number of PCG iterations (typically between 2 and 12) at each GMRES iteration for each of the test cases considered.

It should not be expected that continual speedup can be gained through the use of an increasing number of subdomains, as certain factors such as inter-processor communica-

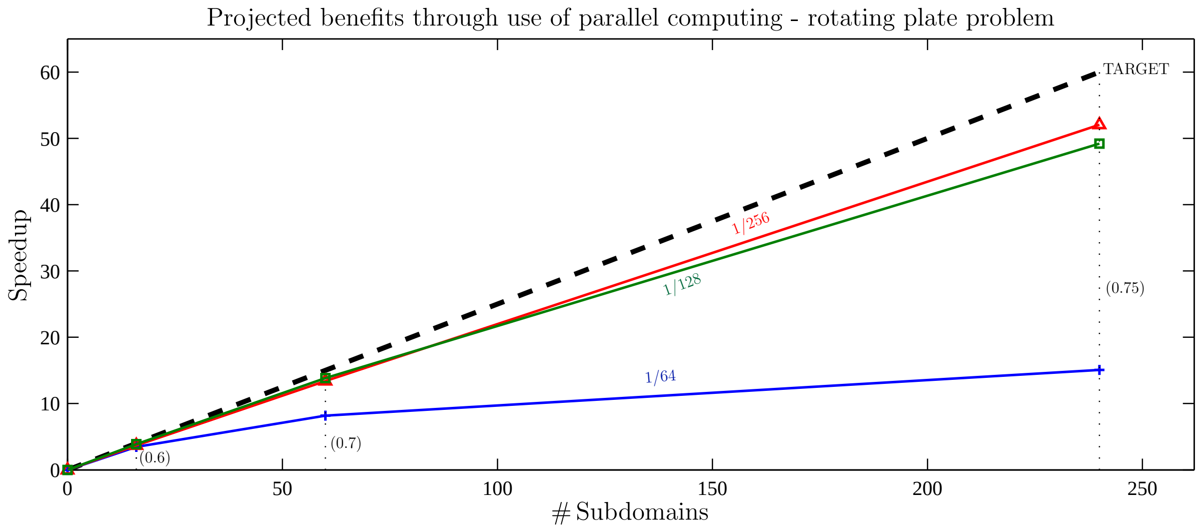


Figure 6.2: Graph to illustrate the anticipated speedup based on the figures presented in Table 6.4. The target line suggests the desired speedup for an increasing number of subdomains (processors).

tion between each of the three steps will begin to play an important role. Therefore, in terms of a regular subdivision, this would suggest an optimal decomposition of the domain based on the mesh parameter, and also possibly other contributing factors relating to computer hardware.

The algorithm has yet to be tested on parallel architecture, and thus the effects of inter-processor communication on our method are, as yet, unknown. However, both Figures 6.1 and 6.2 illustrates the projected speedup based on the data recorded in Tables 6.3 and 6.4 respectively. For both problems, the results determined for $h = 1/16$ and $h = 1/32$ are not shown, as the mesh in both of these cases was seen to be too coarse to warrant the use of parallelism. The number of subdomains considered in Table 6.3 is increased by a factor of 4, therefore the anticipation is that our figures will display speedup by the same amount. The target line is highlighted in Figure 6.1 and it can be seen that for finer meshes, this target is indeed met. Whilst speedup is still evident in the case $h = 1/64$, it is clear from the graph that the decrease in the CPU time with an increasing number of subdomains is below the desired value.

Figure 6.2 illustrates similar behaviour for the rotating plate problem. However, in this case it is noted that the presence of the hole leads to a reduced number of subdomains, and as such a reduced speedup factor of 3.75 is desired when increasing the number of subdomains from 16 to 60. The relevant target line is illustrated in the figure, where again finer meshes can be seen to display roughly the desired speedup. It should also be noted that the use of irregular subdomains for the rotating plate problem suggests that certain individual subdomain solves may be achieved faster than others - for our work, the above figures are based on data obtained from the largest subdomain considered in each case.

CHAPTER 7

NONOVERLAPPING DOMAIN DECOMPOSITION FOR TOPOLOGY OPTIMISATION

Based on the literature review presented in Section 5.5.4 and also the findings described in Chapter 6, we consider the application of domain decomposition to the minimum compliance formulation described in (3.5.8) in two different ways, both of which will be presented within this chapter.

7.1 Domain Decomposition for Fixed Point Approaches (OC/MMA)

7.1.1 Description of Approach

The first approach to be described will be based on the presentation of either the optimality criteria method or the method of moving asymptotes described in Sections 4.3 and 4.4 respectively. As mentioned in Section 4.4.2, both solution techniques can be viewed as fixed point type update schemes using an initial density value. In Section 5.5.4, it was reported in [23, 115] that the bulk of the computational effort will be concentrated on the finite element analysis, namely the repeated update of the displacement variables through the use of the equations of linear elasticity. Therefore, based on the descriptions provided in both Algorithm 4.2 and Algorithm 4.3, we propose to use GMRES as an iterative solution method in step 2(a) of either algorithm. The method will be preconditioned at each fixed point iteration with the associated matrix representation of the discrete fractional Sobolev norm $\widehat{H}_{\theta}$ described in (6.2.11). The resulting solution method is outlined in Algorithm 7.1, where the preconditioner P is defined as per (6.2.13), assembled at each fixed point iteration using the current density value.

Algorithm 7.1 *DOMAIN DECOMPOSITION FOR TOPOLOGY OPTIMISATION (FIXED POINT SOLN. METHODS)*

```
1. $C = 1,\ D = 1,\ Comp = 1.$ Set $\boldsymbol{\rho} := \boldsymbol{\rho}(\mathbf{x})$ equal to $(V_{vol}/m)\,\mathbf{1}_m$.
2. While $C > \mathcal{T}_C$ and $D > \mathcal{T}_D$, Do
    (a) $\mathbf{u} := GMRES\,(K(\boldsymbol{\rho}), \mathbf{f}, P(\boldsymbol{\rho})),\ \boldsymbol{\rho}_I := \boldsymbol{\rho}$.
    (b) Density update for $\boldsymbol{\rho}$ - as described for OC, MMA, etc in Chapter 4.
    (c) Filtering Procedure (dependent on $\mu$).
    (d) $C := |\,\mathbf{f}^T\mathbf{u} - Comp\,|,\ D := \|\boldsymbol{\rho} - \boldsymbol{\rho}_I\|_{\infty},\ Comp := \mathbf{f}^T\mathbf{u},\ k_{it} := k_{it} + 1$.
```

The tolerance used for GMRES will be based on an adaptive criterion (denoted $\mathcal{T}_{\text{GMRES}}$), and can be described as follows

$$\mathcal{T}_{\text{GMRES}} := G \times \|\boldsymbol{\rho} - \boldsymbol{\rho}_I\|_{\infty}^{q}, \tag{7.1.1}$$

encompassing the maximum change in density at each iterative step. The variables G and q represent tuning parameters, and may be chosen in order to suit the problem at hand. The expectation is that use of Algorithm 7.1 will yield results in a modest number of GMRES iterations, with a roughly constant number of GMRES iterations at each fixed point step. The results in Chapter 4 were obtained with an outer tolerance of $\mathcal{T} = 10^{-4}$. However, during numerical experimentation this tolerance was found to be quite harsh in that very few changes would be made to the design during the later iterations of either the OC method or the MMA. We therefore consider a more practical stopping tolerance as described in the algorithm that not only tracks the change in density but also the change in compliance.

7.1.2 Numerical Results

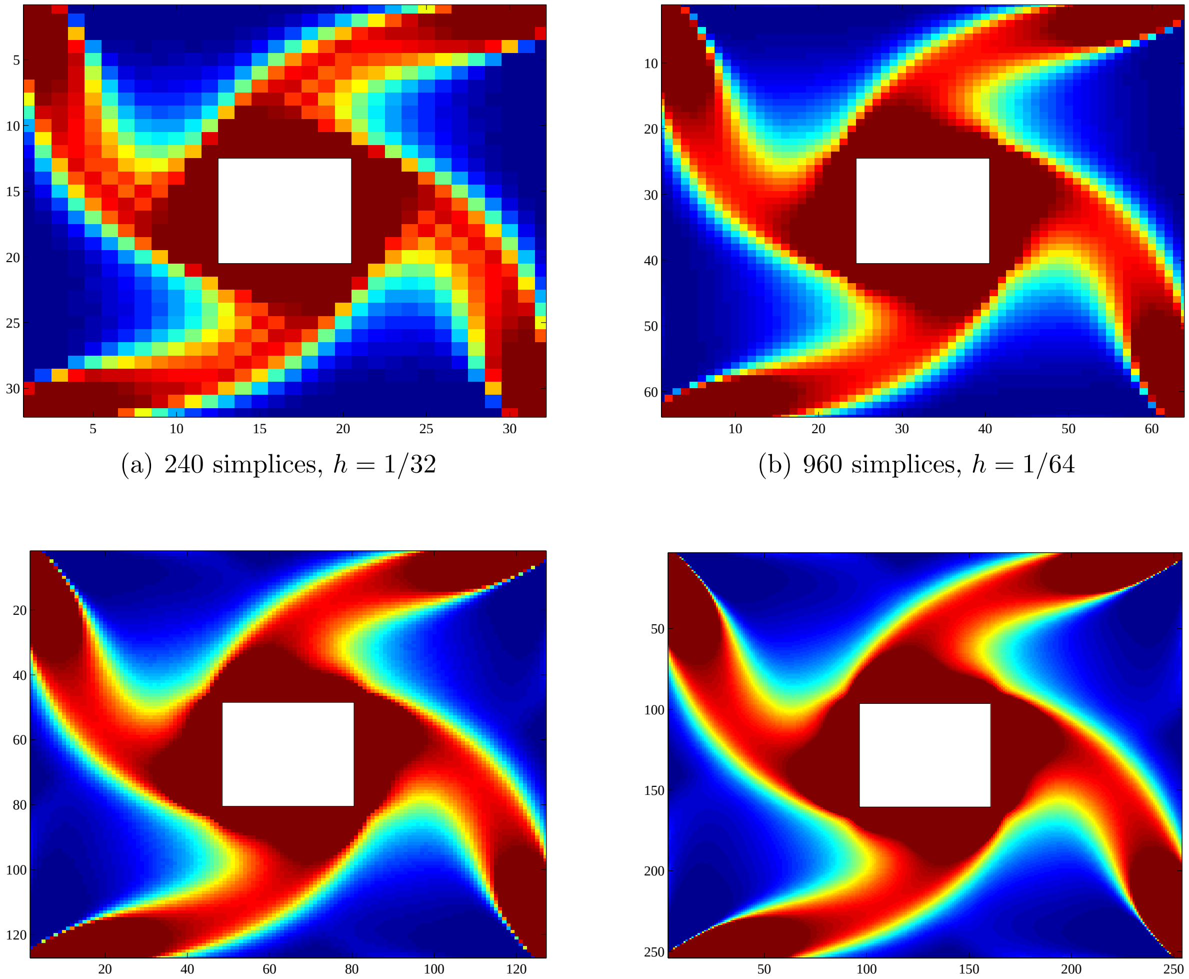
Based on the discussion in Section 4.4.2, results will be presented using the optimality criteria method. However, the method of moving asymptotes, or indeed any other acceptable update scheme for the density, may be considered instead. Importantly, a parallel implementation of the density update will not be considered within this work, however both Borrvall and Petersson [23] and Aage and Lazarov [1] describe an appropriate implementation for the MMA.

	Newton Its. (Avr. GMRES)				Total GMRES Its.			
No. Subdomains:	4	16	64	256	4	16	64	256
θ	0.5	0.6	0.7	0.75	0.5	0.6	0.7	0.75
$h = 1/16$	14 (9)	15 (15)	15 (21)	16 (29)	121	218	310	467
1/32	17 (11)	18 (18)	17 (26)	20 (33)	181	325	449	665
1/64	21 (12)	21 (19)	22 (30)	23 (39)	244	407	654	890
1/128	30 (12)	32 (21)	35 (36)	37 (44)	365	687	1274	1614

Table 7.1: Results for the cantilever beam problem solved using our preconditioning strategy for the solution to the equilibrium equations coupled with the OC method for the density update.

	Newton Its. (Avr. GMRES)				Total GMRES Its.			
No. Subdomains:	4	16	60	240	4	16	60	240
θ	0.5	0.6	0.7	0.75	0.5	0.6	0.7	0.75
$h = 1/32$	13 (6)	13 (14)	13 (23)	14(36)	83	176	297	499
1/64	20 (7)	18 (16)	17 (28)	17 (41)	147	285	475	691
1/128	22 (8)	21 (15)	20 (26)	22 (43)	172	321	511	945
1/256	26 (7)	27 (14)	27 (27)	29 (42)	191	388	719	1228

Table 7.2: Results for the rotating plate problem solved using our preconditioning strategy for the solution to the equilibrium equations coupled with the OC method for the density update.



(a) 240 simplices, $h = 1/32$

(b) 960 simplices, $h = 1/64$

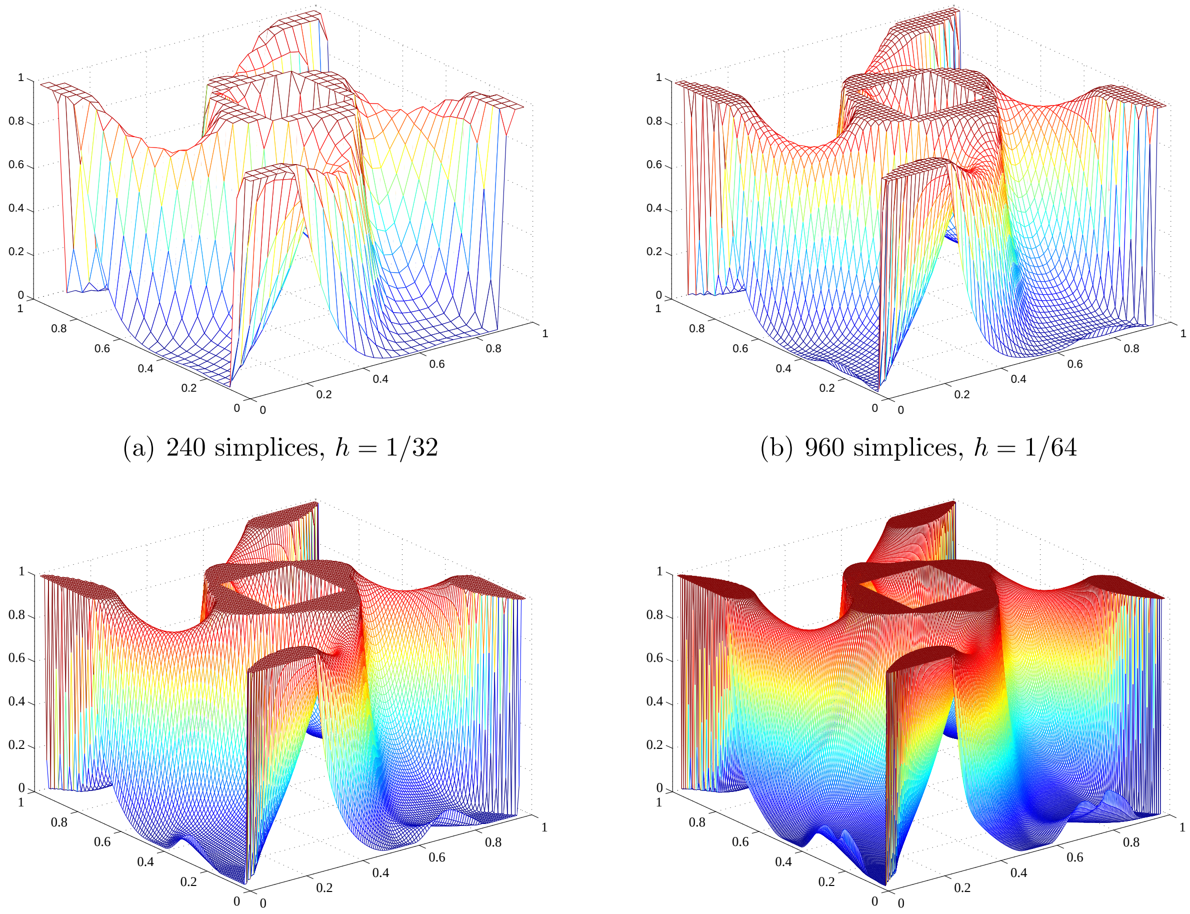
(c) 3840 simplices, $h = 1/128$

(d) 15360 simplices, $h = 1/256$

Figure 7.1: Illustration of optimal solution layouts for the rotating plate problem using differing numbers of simplices.

Tables 7.1 and 7.2 provide results illustrating the performance of our approach for the cantilever beam and rotating plate problems respectively, with zero body force and boundary traction set equal to 1 in both cases. For both test cases, the tolerances (denoted $\mathcal{T}_C$ and $\mathcal{T}_D$) in Algorithm 7.1 were set to 10^{-6} and 10^{-3} respectively. The tolerance for GMRES described in (7.1.1) involved both G and q being set to 0.5 and 0.25 respectively, with both values determined through experimentation.

All of the results involve the application of $\widehat{H}_\theta$ through use of the inverse Lanczos process. The total number of fixed point iterations is provided in the tables, along with the average number of GMRES iterations per fixed point step (bracketed). In both examples,



(a) 240 simplices, $h = 1/32$

(b) 960 simplices, $h = 1/64$

(c) 3840 simplices, $h = 1/128$

(d) 15360 simplices, $h = 1/256$

Figure 7.2: Density plot illustrations for each of the plots shown in Figure 7.1.

the number of fixed point iterations appears to increase for finer meshes. The average number of GMRES iterations also appears to increase in certain cases, however this is a consequence of the choice of θ. When $\theta = 0.5$ in all cases, the average number of GMRES iterations is seen to remain roughly constant (see Appendix A.4). Whilst we still see a logarithmic dependence on the average number of GMRES iterations for an increasing number of subdomains, the fixed point iterations remain roughly constant.

For both of the examples considered, the tables illustrate that an increasing number of fixed point iterations can be expected as the mesh is refined. Therefore, whilst the number of GMRES iterations per fixed point step can be assumed to remain roughly constant, no guarantees can be claimed regarding the total number of fixed point iterations, and thus the total number of GMRES iterations, required to achieve a suitable design. Despite the relatively low computational cost attributed to the density update, the essence of the problem lies in the separate treatment of both the design and state variables, meaning that no real control can be maintained over the required quantity of fixed point iterations. We therefore look to adapt the presentation of the interior point type approach described in Section 4.5 in order to use the work presented both here and also in Chapter 6 to develop an appropriate preconditioning strategy for the resulting Newton system.

7.2 Domain Decomposition for Interior Point Approach

7.2.1 Reduced Formulation

By using the presentation of the reduced problem (4.5.11), we consider a permutation of nodal indices based on their location within the domain such that the vector of unknowns may be written as $\mathbf{y}_2 = \left(\mathbf{y}_{2_I}^T, \mathbf{y}_{2_\Gamma}^T\right)^T := \left(\left(\mathbf{u}_I^T, \lambda\right), \mathbf{u}_\Gamma^T\right)^T$. Under this ordering and by defining $\mathbf{y}_1 := \left(\mathbf{u}_I^T, \lambda, \boldsymbol{\rho}^T, \boldsymbol{\kappa}^T, \boldsymbol{\delta}^T, \mathbf{u}_\Gamma^T\right)^T$, the system may be described in the following manner

$$J_1^{\mathrm{DD}} \Delta \mathbf{y}_2{}^{\{k_{\mathrm{it}}\}} := J_1^{\mathrm{DD}}\left(\mathbf{y}_1^{\{k_{\mathrm{it}}-1\}}\right) \begin{pmatrix} \Delta \mathbf{u}_I^{\{k_{\mathrm{it}}\}} \\ \Delta\lambda^{\{k_{\mathrm{it}}\}} \\ \hline \Delta \mathbf{u}_\Gamma^{\{k_{\mathrm{it}}\}} \end{pmatrix} = \mathcal{R}_1^{\mathrm{DD}}\left(\mathbf{y}_1^{\{k_{\mathrm{it}}-1\}}\right) =: \mathcal{R}_1^{\mathrm{DD}}. \tag{7.2.1}$$

where the block Jacobian matrix J_1^{DD} may be written down as follows

$$J_1^{\mathrm{DD}}(\mathbf{y}_1) = \left[\begin{array}{cc|c} K_{II}(\boldsymbol{\rho}) & 0 & K_{I\Gamma}(\boldsymbol{\rho}) \\ 0 & 0 & 0 \\ \hline K_{\Gamma I}(\boldsymbol{\rho}) & 0 & K_{\Gamma\Gamma}(\boldsymbol{\rho}) \end{array}\right] + \left[\begin{array}{c} B_I \\ \mathbf{1}_m^T \\ \hline B_\Gamma \end{array}\right] D \left[\begin{array}{cc|c} B_I^T & \mathbf{1}_m & B_\Gamma^T \end{array}\right] =: \begin{bmatrix} J_{1_{II}}^{\mathrm{DD}}(\mathbf{y}_1) & J_{1_{I\Gamma}}^{\mathrm{DD}}(\mathbf{y}_1) \\ J_{1_{\Gamma I}}^{\mathrm{DD}}(\mathbf{y}_1) & J_{1_{\Gamma\Gamma}}^{\mathrm{DD}}(\mathbf{y}_1) \end{bmatrix}, \tag{7.2.2}$$

such that $B_I := B\left(\mathbf{u}_I\right) \in \mathbb{R}^{\hat{n}_I \times m}$ and $B_\Gamma := B\left(\mathbf{u}_\Gamma\right) \in \mathbb{R}^{\hat{n}_\Gamma \times m}$, where the matrix $D := D(\boldsymbol{\rho})$ is as presented in (4.5.13). The precise definition of the residual vectors involved within $\mathcal{R}_1^{\mathrm{DD}}$ will not be given, but can be derived via the terms involved within $\mathcal{R}_3$ presented in

(4.5.12).

Using the presentation in Section 6.2, we look to precondition (7.2.1) as follows

$$\widetilde{P} = \begin{bmatrix} J^{\mathrm{DD}}_{1_{II}} & J^{\mathrm{DD}}_{1_{I\Gamma}} \\ 0 & \widetilde{S} \end{bmatrix},$$

with $\widetilde{S}$ representing an approximation to the Schur complement of J_1^{DD} based on the splitting in (7.2.2). As the Schur complement is centered around the interface displacement nodes, we consider preconditioning the matrix representation of the underlying discrete Steklov-Poincaré operator with $\widehat{H}_\theta$ described in (6.2.11). Provided that the terms from the product of matrices within (7.2.2) do not dominate the overall behaviour of the Schur complement, the expectation is for $\widehat{H}_\theta$ to provide an effective preconditioner for the resulting interface problem.

Using the same line search procedure described in Section 4.5.3, we are in a position to describe the resulting algorithm for the reduced primal-dual interior point algorithm, with the Newton step calculated through GMRES coupled with a preconditioner based on a decomposition of the domain. The full description is provided in Algorithm 7.2, following a similar (albeit streamlined) structure to the primal-dual Newton algorithm provided in Algorithm 4.4.

Algorithm 7.2 *PRIMAL-DUAL NEWTON KRYLOV METHOD (REDUCED)*

```
1. $k = 0, l = 0, r = 1, s = 1, IE = 1$.
2. $\mathbf{u} = K^{-1}\left(\left(V_{vol}/m\right)\mathbf{1}_m\right)\mathbf{f}$, $\boldsymbol{\rho} = \left(V_{vol}/m\right)\mathbf{1}_m$, $\lambda = 1$, $\boldsymbol{\kappa} = \mathbf{1}_m$, $\boldsymbol{\delta} = \mathbf{1}_m$.
3. $\mathbf{y}_1 := \left(\mathbf{u}_I^T, \lambda, \boldsymbol{\rho}^T, \boldsymbol{\kappa}^T, \boldsymbol{\delta}^T, \mathbf{u}_\Gamma^T\right)^T$, $\mathbf{y}_2 := \left(\mathbf{u}^T, \lambda\right)^T$.
4. While max$(r, s) > \mathcal{T}$, Do
    (a) While $IE > \mathcal{T}_N$, Do
        i. $\Delta\mathbf{y}_2 := GMRES\left(J_1^{DD}\left(\mathbf{y}_1\right), \mathcal{R}_1^{DD}\left(\mathbf{y}_1\right), \widetilde{P}\left(\mathbf{y}_1\right)\right)$.
        ii. $\Delta\boldsymbol{\rho} := D\mathcal{R}_{\boldsymbol{\rho}}$.
        iii. $\Delta\boldsymbol{\kappa} := X^{-1}\left(\mathcal{R}_1^{(4)} - M\Delta\boldsymbol{\rho}\right)_{\mathbf{y}_1}$.
        iv. $\Delta\boldsymbol{\delta} := \widetilde{X}^{-1}\left(\mathcal{R}_1^{(5)} + Q\Delta\boldsymbol{\rho}\right)_{\mathbf{y}_1}$.
        v. $\mathbf{y}_1 := \mathbf{y}_1 + \alpha_{LS}\Delta\mathbf{y}_1$.
        vi. $k_{it} := k_{it} + 1$.
        vii. $IE := \mathcal{R}_1^{DD}\left(\mathbf{y}_1\right)^T \Delta\mathbf{y}_2$.
    (b) $r := \beta_r r, s := \beta_s s$.
    (c) $l_{it} := l_{it} + 1$.
```

We now use Algorithm 7.2 to produce numerical results for the different test examples presented in Section 3.6.

7.2.2 Numerical Results

Illustrated in Tables 7.3 and 7.4 are respective results for the solution to the rotating plate and both the cantilever and MBB beam problems using the reduced primal-dual Newton Krylov algorithm described in Algorithm 7.2. The results were produced based on the matrix representation to the underlying discrete fractional Sobolev norm $\widehat{H}_\theta$ being applied through a generalised eigenvalue decomposition. The reasoning behind this is not only to allow for a direct comparison with work to be presented within forthcoming sections, but also as a consequence of the manner in which the domain is decomposed. Unlike the cantilever beam and rotating plate examples, the Dirichlet nodes present in the MBB beam example are located in the lower left and right corners of the domain. As we consider a regular subdivision of the domain, the domain is decomposed in such a fashion that the intersection between the interface Γ and the Dirichlet boundary $\partial\Omega_D$ is empty. As such, the interface Laplacian matrix described in (6.2.9) will be singular, with d eigenvalues equal to zero. Therefore, iterative alternatives used in the application of $\widehat{H}_\theta$ cannot be considered, since the generalised Lanczos process requires either M or L to be symmetric positive definite. Nevertheless, the nonsingularity of M allows for the realisation of a generalised eigenvalue decomposition for the matrix pair (L, M).

The outer and inner tolerances (denoted $\mathcal{T}$ and $\mathcal{T}_N$) in Algorithm 7.2 were set to 10^{-7} and 10^{-5} respectively. The tolerance for GMRES is defined through an adaptive criteria (again, denoted $\mathcal{T}_{\text{GMRES}}$) in the following manner

$$\mathcal{T}_{\text{GMRES}} := r^{p_1} \times \mathcal{T}^{p_2} \times \|\mathcal{R}_1^{DD}\|_2^{p_3}.$$

		Newton Its. (Avr. GMRES)				Total GMRES Its.			
No. Subdomains:		4	16	64	256	4	16	64	256
Problem	h \ θ	0.5	0.6	0.7	0.75	0.5	0.6	0.7	0.75
Cant. Beam	1/16	21 (10)	21 (19)	21 (34)	21 (75)	209	395	723	1803
	1/32	26 (8)	25 (13)	26 (29)	26 (64)	202	328	755	1662
	1/64	28 (6)	28 (12)	31 (22)	30 (47)	161	337	695	1407
	1/128	33 (4)	33 (10)	35 (18)	35 (39)	148	332	616	1369
MBB Beam	1/16	26 (11)	27 (28)	27 (53)	26 (87)	284	750	1434	2262
	1/32	30 (9)	30 (23)	31 (42)	30 (67)	268	694	1302	2091
	1/64	33 (8)	34 (19)	33 (37)	34 (59)	255	645	1220	1999
	1/128	38 (6)	38 (16)	37 (32)	37 (52)	245	589	1174	1940

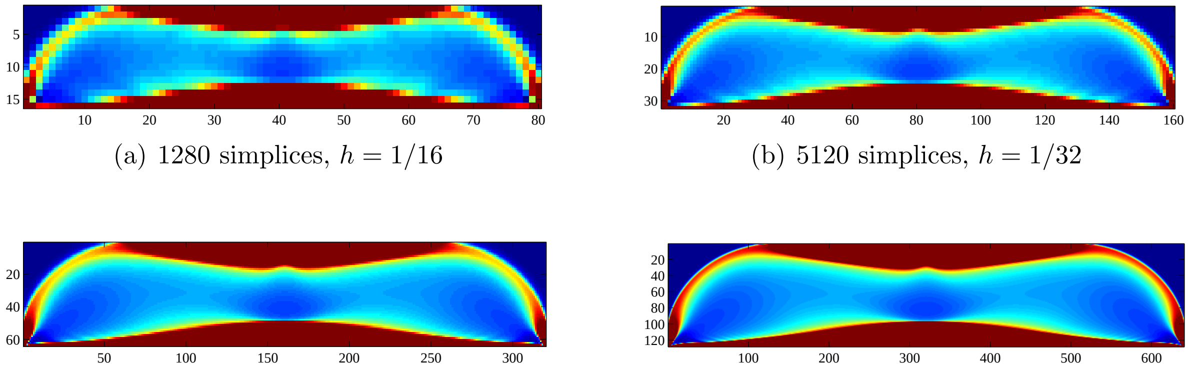
Table 7.3: Results for the cantilever and MBB beam problems solved using the reduced interior point solution method, with preconditioned GMRES used for the calculation of the Newton update.

	Newton Its. (Avr. GMRES)				Total GMRES Its.			
No. Subdomains:	4	16	60	240	4	16	60	240
θ / h	0.5	0.6	0.7	0.75	0.5	0.6	0.7	0.75
1/32	21 (9)	22 (19)	22 (34)	22 (59)	187	411	753	1304
1/64	24 (7)	24 (14)	24 (26)	24 (51)	160	344	632	1213
1/128	28 (5)	28 (11)	28 (22)	29 (40)	146	320	619	1173
1/256	32 (4)	32 (10)	32 (19)	34 (32)	121	305	592	1101

Table 7.4: Results for the rotating plate problem solved using the reduced interior point solution method, with preconditioned GMRES used for the calculation of the Newton update.

In general, the values p_1, p_2 and p_3 can be taken as $0.2, 0.75$ and 0.5 respectively, however when producing results these values were occasionally altered slightly to adapt to the problem at hand. This criteria and the relevant values were decided upon based on experimentation, and found to deliver effective results.

For both tables, the column on the left indicates the total number of Newton iterations, with the average number of GMRES iterations indicated in brackets. The column on the right lists the total number of GMRES iterations in each case. As in Section 6.3, reading the table from left to right provides an indication of how our algorithm performs for an increasing number of subdomains. From the results, a clear dependence on the number of subdomains considered can be seen, with an increasing number of subdomains leading to



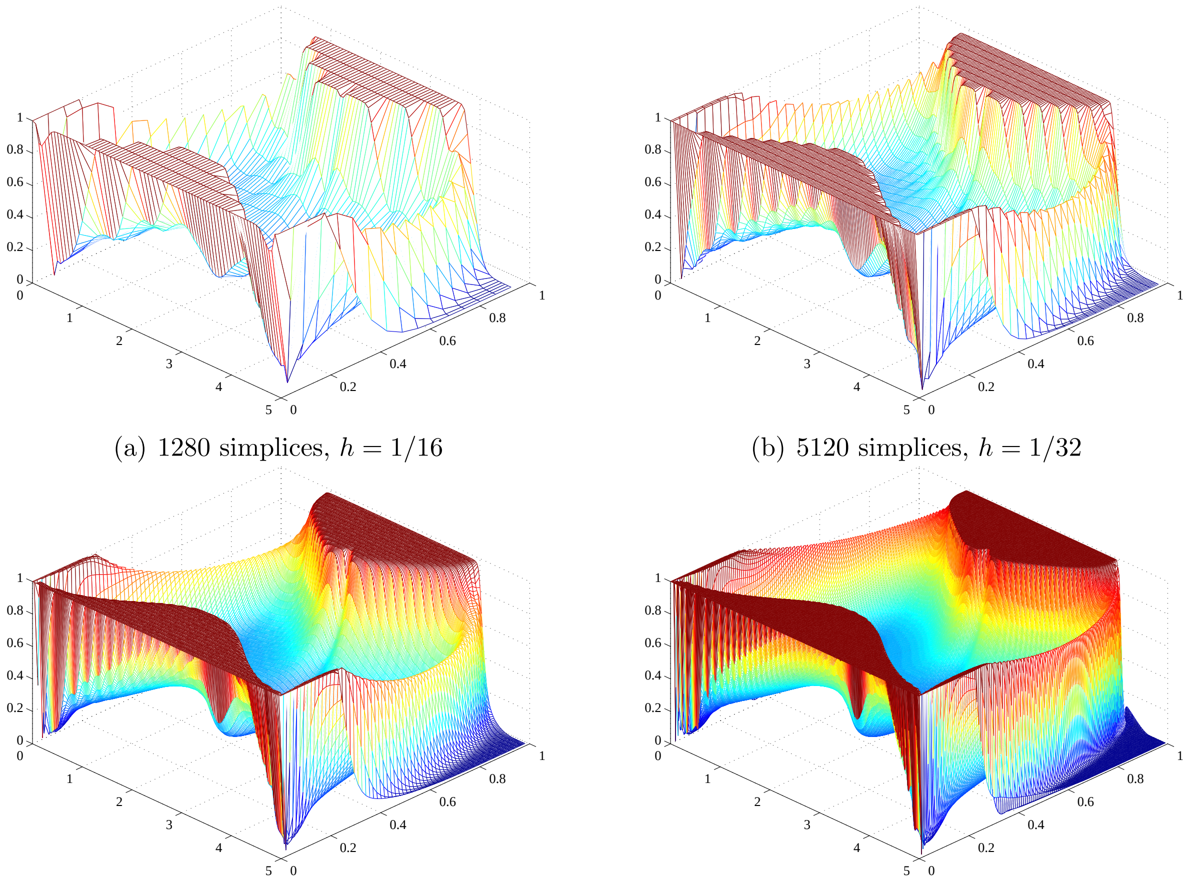
(a) 1280 simplices, $h = 1/16$

(b) 5120 simplices, $h = 1/32$

(c) 20480 simplices, $h = 1/64$

(d) 81920 simplices, $h = 1/128$

Figure 7.3: Illustration of optimal solution layouts for the rotating plate problem using differing numbers of simplices.



(a) 1280 simplices, $h = 1/16$

(b) 5120 simplices, $h = 1/32$

(c) 20480 simplices, $h = 1/64$

(d) 81920 simplices, $h = 1/128$

Figure 7.4: Density plot illustrations for each of the plots shown in Figure 7.3.

an increased number of GMRES iterations. However, this dependence is approximately logarithmic in nature, and not linear (or worse), suggesting that while an increase in the number of GMRES iterations is noted, the rise is of a magnitude less than the increase in the number of subdomains considered. In both cases, the total number of Newton iterations remains roughly constant for fixed mesh sizes.

Reading both tables from top to bottom indicates how our algorithm performs for differing mesh parameters. Both the average and the total number of GMRES iterations remain roughly constant (slightly decreasing) in all cases, suggesting that results are determined independently of the chosen mesh parameter.

As per the investigation in Section 6.3, variants in the value of θ were found to provide improvements in the overall number of GMRES iterations, with the relevant values of θ highlighted in the table. A direct comparison with the figures recorded in Tables 7.1 and

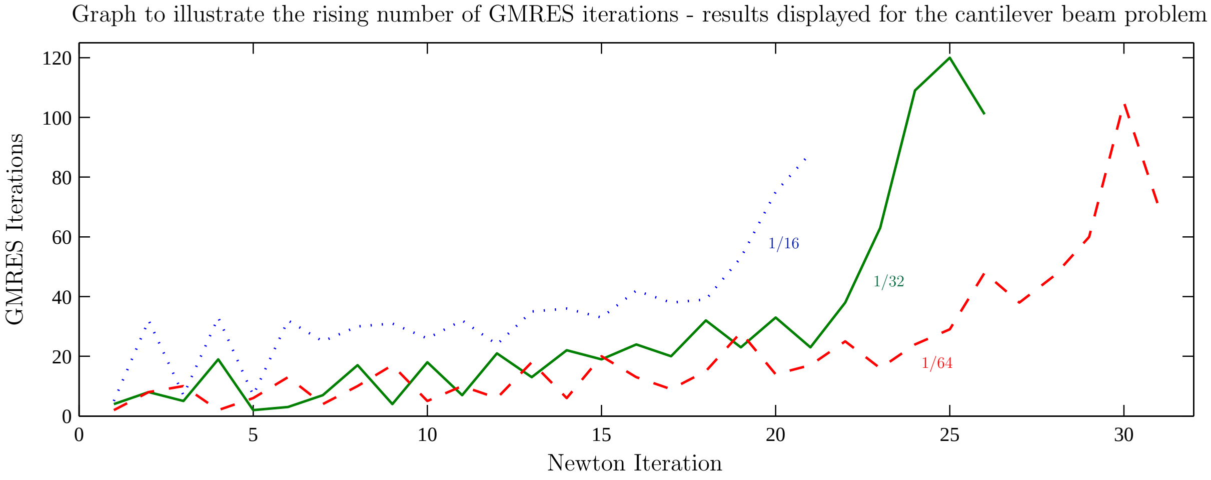


Figure 7.5: Graph to illustrate the rise in GMRES iterations during the course of the iterative method described in Algorithm 7.2. The data is shown based on a decomposition of the cantilever beam problem into 64 subdomains for different mesh parameters.

7.2 suggests that the reduced primal-dual algorithm has a better hold on the total number of GMRES iterations than the fixed point solution method described in Algorithm 7.1. Whilst the latter is able to deliver solutions to fairly coarse problems in a relatively small number of iterations, the former appears to be more robust for finer mesh parameters. It is also interesting to note that the total number of iterations displayed for the reduced primal-dual approach appears to decrease as the mesh is refined, whereas an increase is noted for those displayed for the fixed point approach. Nevertheless, from Figure 7.5, it is clear that a substantial number of GMRES iterations are required as the barrier parameters become notably small (and thus providing a reasonable approximation of zero). This behaviour is due in part to the choice of adaptive tolerance, however experimentation seems to suggest that the contribution arising from the product of matrices present in (7.2.2) begins to dominate the reduced system matrix due to inversion of terms tending to zero. Therefore, our preconditioner should be enhanced in order to compensate for this behaviour.

Whilst an analytical proof of spectral equivalence can be illustrated for the equations of linear elasticity, the precise role of the associated discrete Steklov-Poincaré operator within the Schur complement of J_1^{DD} is unclear due to its convoluted nature. Therefore, whilst the results suggest spectral equivalence, an analytical proof of concept is difficult to assert. As a result, we look to retain the original unreduced formulation and consider a modified (although equivalent) problem that aims to solve topology optimisation problems locally on subdomains.

7.2.3 Problem Reformulation

Using the notation presented in Section 5.5 for a nonoverlapping decomposition of the domain, we rewrite the VTS problem presented in (3.5.8) by introducing unknown variables $\mathcal{M}_k$ in the following way

$$\begin{aligned} \min_{\mathbf{u},\boldsymbol{\rho},\boldsymbol{\mathcal{M}}} \quad & \frac{1}{2}\mathbf{f}^T\mathbf{u} && \\ \text{subject to:} \quad & K(\boldsymbol{\rho})\mathbf{u} = \mathbf{f}, && \\ & \sum_{e\in D_k} \rho_e = \mathcal{M}_k && k = 1,\dots,N, \\ & \sum_{k=1}^{N} \mathcal{M}_k = V_{\text{vol}}, && \\ & 0 \le \underline{\rho} \le \rho_e \le \overline{\rho} && \forall e \in D. \end{aligned} \tag{7.2.3}$$

where $\boldsymbol{\mathcal{M}} := (\mathcal{M}_1, \dots, \mathcal{M}_N)^T$, such that $\mathcal{M}_k$ represents the volume on each subdomain Ω_k. The set D is decomposed as $D = \cup_{k=1}^{N} D_k$, with each D_k representing the index set of simplices contained within subdomain Ω_k such that $m_k := |D_k|$.

The associated barrier problem can be written down as follows

$$\begin{aligned} \min_{\mathbf{u},\boldsymbol{\rho},\boldsymbol{\mathcal{M}}} \quad & \frac{1}{2}\mathbf{f}^T\mathbf{u} - r\sum_{e\in D}\log(\rho_e - \underline{\rho}) - s\sum_{e\in D}\log(\overline{\rho} - \rho_e) \\ \text{subject to:} \quad & K(\boldsymbol{\rho})\mathbf{u} = \mathbf{f}, \\ & C^T\boldsymbol{\rho} = \boldsymbol{\mathcal{M}}, \\ & \sum_{k=1}^{N} \mathcal{M}_k = V_{\text{vol}}, \end{aligned} \tag{7.2.4}$$

where

$$C_{jk} := \begin{cases} 1 & \text{if} \quad j \in D_k, \\ 0 & \text{otherwise,} \end{cases}$$

with r and s used to avoid unboundedness at either extreme. The Lagrangian to (7.2.4) may be described as follows

$$\mathcal{L}_{IP_{\mathrm{E}}}^{(r,s)}\left(\mathbf{u}, \boldsymbol{\rho}, \boldsymbol{\mathcal{M}}, \mathbf{v}, \boldsymbol{\lambda}, \pi\right) := \frac{1}{2}\mathbf{f}^T\mathbf{u} - r\sum_{e\in D}\log(\rho_e - \underline{\rho}) - s\sum_{e\in D}\log(\overline{\rho} - \rho_e)$$
$$-\mathbf{v}^T\left(K(\boldsymbol{\rho})\mathbf{u} - \mathbf{f}\right) - \boldsymbol{\lambda}^T\left(C^T\boldsymbol{\rho} - \boldsymbol{\mathcal{M}}\right) - \pi\left(\sum_{k=1}^{N}\mathcal{M}_k - V_{\mathrm{vol}}\right).$$

In a similar manner to the presentation in Section 4.5, our constrained optimisation problem may be viewed in terms of the following unconstrained saddle point problem

$$\min_{\mathbf{u}, \boldsymbol{\rho}, \boldsymbol{\mathcal{M}}} \max_{\mathbf{v}, \boldsymbol{\lambda}, \pi} \mathcal{L}_{IP_{\mathrm{E}}}^{(r,s)}\left(\mathbf{u}, \boldsymbol{\rho}, \boldsymbol{\mathcal{M}}, \mathbf{v}, \boldsymbol{\lambda}, \pi\right),$$

with solution $\left(\mathbf{u}^{(r,s)}, \boldsymbol{\rho}^{(r,s)}, \boldsymbol{\mathcal{M}}^{(r,s)}\right)$, where $\mathbf{v} \in \mathbb{R}^{\hat{n}}$, $\boldsymbol{\lambda} \in \mathbb{R}^N$ and $\pi \in \mathbb{R}$ represent Lagrange multipliers. The convergence property outlined in (4.5.2) can be translated analogously to the expanded formulation (7.2.3), meaning that

$$\left(\mathbf{u}^{(r,s)}, \boldsymbol{\rho}^{(r,s)}, \boldsymbol{\mathcal{M}}^{(r,s)}\right) \to \left(\mathbf{u}^*, \boldsymbol{\rho}^*, \boldsymbol{\mathcal{M}}^*\right) \qquad (r, s \to 0). \tag{7.2.5}$$

Therefore, solving (7.2.4) in the limit as the barrier parameters tend to zero will yield a family of subproblems with corresponding solutions that converge to the solution to (7.2.3).

The first order necessary optimality conditions to (7.2.3) can be written down in the following manner

$$
\text{Stationarity:}\quad \begin{cases}
\nabla_{\mathbf{u}}\mathcal{L}_{IP_{\mathrm{E}}}^{(r,s)} = -K(\boldsymbol{\rho})\mathbf{v} + \dfrac{1}{2}\mathbf{f} = \mathbf{0}, \\
\nabla_{\boldsymbol{\rho}}\mathcal{L}_{IP_{\mathrm{E}}}^{(r,s)} = -\mathbf{z}_{\mathbf{v},1} - C\boldsymbol{\lambda} - rX^{-1}\mathbf{1}_m + s\widetilde{X}^{-1}\mathbf{1}_m = \mathbf{0}, \\
\nabla_{\boldsymbol{\mathcal{M}}}\mathcal{L}_{IP_{\mathrm{E}}}^{(r,s)} = \boldsymbol{\lambda} - \mathbf{1}_N\pi = \mathbf{0},
\end{cases}
$$

$$
\text{Primal Feas.:}\quad \begin{cases}
-K(\boldsymbol{\rho})\mathbf{u} + \mathbf{f} = \mathbf{0} \implies \mathbf{v} = \dfrac{1}{2}\mathbf{u}, \\
\boldsymbol{\mathcal{M}} - C^T\boldsymbol{\rho} = \mathbf{0}, \\
V_{\mathrm{vol}} - \displaystyle\sum_{k=1}^{N}\mathcal{M}_k = 0.
\end{cases}
$$

As in Section 4.5, in order to deal with the impending singularity of the Hessian to the Lagrangian $\mathcal{L}_{IP_{\mathrm{E}}}^{(r,s)}$ as both r and s tend to zero, non-negative variables $\hat{\boldsymbol{\kappa}}$ and $\hat{\boldsymbol{\delta}}$ defined as per (4.5.4) are introduced, which can be seen to converge to the respective associated Lagrange multipliers $\boldsymbol{\kappa}$ and $\boldsymbol{\delta}$ for the density constraints in (3.5.8) as a consequence of (7.2.5).

Through the elimination of $\mathbf{v}$ and the substitution of (4.5.6), the first order optimality conditions can be written under an appropriate ordering as

$$
\mathcal{R}_4 = \begin{pmatrix} \mathcal{R}_4^{(1)} \\ \mathcal{R}_4^{(2)} \\ \mathcal{R}_4^{(3)} \\ \mathcal{R}_4^{(4)} \\ \mathcal{R}_4^{(5)} \\ \mathcal{R}_4^{(6)} \\ \mathcal{R}_4^{(7)} \end{pmatrix} := \begin{pmatrix} \mathbf{f} - K(\boldsymbol{\rho})\mathbf{u} \\ \boldsymbol{\mathcal{M}} - C^T\boldsymbol{\rho} \\ -\frac{1}{2}\mathbf{z}_{\mathbf{u},1} - C\boldsymbol{\lambda}\,\mathbf{1}_m - \boldsymbol{\kappa} + \boldsymbol{\delta} \\ r\mathbf{1}_m - MX\mathbf{1}_m \\ s\mathbf{1}_m - Q\widetilde{X}\mathbf{1}_m \\ \boldsymbol{\lambda} - \mathbf{1}_N\pi \\ V_{\mathrm{vol}} - \mathbf{1}_N^T\boldsymbol{\mathcal{M}} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ 0 \end{pmatrix}. \tag{7.2.7}
$$

By defining $\bar{\mathbf{x}}_4 := \left(\mathbf{u}^T, \boldsymbol{\lambda}^T, \boldsymbol{\rho}^T, \boldsymbol{\kappa}^T, \boldsymbol{\delta}^T, \boldsymbol{\mathcal{M}}^T, \pi\right)^T$, we apply Newton's method to the nonlinear optimality conditions presented in (7.2.7) to arrive at a matrix-vector system of the form

$$J_4\left(\bar{\mathbf{x}}_4^{\{k_{\text{it}}-1\}}\right)\Delta\bar{\mathbf{x}}_4^{\{k_{\text{it}}\}} = \mathcal{R}_4\left(\bar{\mathbf{x}}_4^{\{k_{\text{it}}-1\}}\right),$$

where the Jacobian matrix J_4 corresponds to

$$J_4(\bar{\mathbf{x}}_4) = \begin{bmatrix} K(\boldsymbol{\rho}) & 0_{n\times N} & B(\mathbf{u}) & 0_{n\times m} & 0_{n\times m} & 0_{n\times N} & 0_{n\times 1} \\ 0_{N\times n} & 0_N & C^T & 0_{N\times m} & 0_{N\times m} & -I_N & 0_{N\times 1} \\ B(\mathbf{u})^T & C & 0_m & I_m & -I_m & 0_{m\times N} & 0_{m\times 1} \\ 0_{m\times n} & 0_{m\times N} & M & X_m & 0_m & 0_{m\times N} & 0_{m\times 1} \\ 0_{m\times n} & 0_{m\times N} & -Q & 0_m & \widetilde{X}_m & 0_{m\times N} & 0_{m\times 1} \\ 0_{N\times n} & -I_N & 0_{N\times m} & 0_{N\times m} & 0_{N\times m} & 0_N & \mathbf{1}_N \\ 0_{1\times n} & 0_{1\times N} & 0_{1\times m} & 0_{1\times m} & 0_{1\times m} & \mathbf{1}_N^T & 0 \end{bmatrix}.$$

The only nodal components of $\bar{\mathbf{x}}_4$ are the elements of the displacement vector $\mathbf{u}$. The multiplier π is a global variable that cannot be split. All other components, such as the densities and corresponding Lagrange multipliers, can be assigned strictly to subdomains and have no contribution on the interface. By introducing

$$\mathbf{y}_3 := \left(\mathbf{y}_{3_I}^T, \mathbf{y}_{3_\Gamma}^T\right)^T := \left(\left(\mathbf{u}_I^T, \boldsymbol{\lambda}^T, \boldsymbol{\rho}^T, \boldsymbol{\kappa}^T, \boldsymbol{\delta}^T\right), \left(\mathbf{u}_\Gamma^T, \boldsymbol{\mathcal{M}}^T, \pi\right)\right)^T,$$

so that nodes are viewed based on their location within the domain, we consider the following expanded Newton system

$$J_3^{\text{DD}}\Delta\mathbf{y}_3^{\{k_{\text{it}}\}} := J_3^{\text{DD}}\left(\mathbf{y}_1^{\{k_{\text{it}}-1\}}\right)\Delta\mathbf{y}_3^{\{k_{\text{it}}\}} = \mathcal{R}_3^{\text{DD}}\left(\mathbf{y}_1^{\{k_{\text{it}}-1\}}\right) =: \mathcal{R}_3^{\text{DD}},$$

where the Jacobian matrix J_3^{DD} has the following block 2×2 representation

$$J_3^{\mathrm{DD}} = \left[\begin{array}{c|c} J_{3_{II}}^{\mathrm{DD}} & J_{3_{I\Gamma}}^{\mathrm{DD}} \\ \hline J_{3_{\Gamma I}}^{\mathrm{DD}} & J_{3_{\Gamma\Gamma}}^{\mathrm{DD}} \end{array}\right] = \left[\begin{array}{ccccc|ccc} K_{II} & 0_{n_I\times N} & B_I & 0_{n_I\times m} & 0_{n_I\times m} & K_{I\Gamma} & 0_{n_I\times N} & 0_{n_I\times 1} \\ 0_{N\times n_I} & 0_N & C^T & 0_{N\times m} & 0_{N\times m} & 0_{N\times n_\Gamma} & -I_N & 0_{N\times 1} \\ B_I^T & C & 0_m & I_m & -I_m & B_\Gamma^T & 0_{m\times N} & 0_{m\times 1} \\ 0_{m\times n_I} & 0_{m\times N} & M & X_m & 0_m & 0_{m\times n_\Gamma} & 0_{m\times N} & 0_{m\times 1} \\ 0_{m\times n_I} & 0_{m\times N} & -Q & 0_m & \widetilde{X}_m & 0_{m\times n_\Gamma} & 0_{m\times N} & 0_{m\times 1} \\ \hline K_{\Gamma I} & 0_{n_\Gamma\times N} & B_\Gamma & 0_{n_\Gamma\times m} & 0_{n_\Gamma\times m} & K_{\Gamma\Gamma} & 0_{n_\Gamma\times N} & 0_{n_\Gamma\times 1} \\ 0_{N\times n_I} & -I_N & 0_{N\times m} & 0_{N\times m} & 0_{N\times m} & 0_{N\times n_\Gamma} & 0_N & \mathbf{1}_N \\ 0_{1\times n_I} & 0_{1\times N} & 0_{1\times m} & 0_{1\times m} & 0_{1\times m} & 0_{1\times n_\Gamma} & \mathbf{1}_N^T & 0 \end{array}\right].$$

Under a further permutation which lists all the unknowns corresponding to subdomains Ω_k, for each $k = 1, \dots, N$, it can be seen that $J_{3_{II}}^{\mathrm{DD}} := \bigoplus_{k=1}^{N} J_{3_{kk}}^{\mathrm{DD}}$, where each nonsingular $J_{3_{kk}}^{\mathrm{DD}}$ has the following form

$$J_{3_{kk}}^{\mathrm{DD}} := \begin{bmatrix} K_{kk} & & B_k & & \\ & & \mathbf{1}_{m_k}^T & & \\ B_k^T & \mathbf{1}_{m_k} & & I_{m_k} & -I_{m_k} \\ & & M_k & X_k & \\ & & -Q_k & & \widetilde{X}_k \end{bmatrix},$$

representing the Jacobian of the minimisation problem as posed over the subdomain Ω_k with Dirichlet boundary conditions enforced on $\partial\Omega_k$. Zero entries have been omitted in the above in order to highlight sparsity. The fact that $J_{3_{II}}^{\mathrm{DD}}$ can be written as a direct sum of local Jacobian matrices associated with similar local minimisation problems was achieved as a consequence of the reformulated problem presented in (7.2.3). It is not only an interesting feature to observe that a decomposition in this manner leads to a block representation consisting of local Jacobians associated with similar minimisation problems posed on subdomains, but also that each of the local problems inherits the well-posedness present within the global problem.

The sparse indefinite structure of J_3^{DD} suggests GMRES as an appropriate iterative solution method, which will be coupled with a right preconditioner of the form

$$\widetilde{P} := \begin{bmatrix} J^{\mathrm{DD}}_{3_{II}} & J^{\mathrm{DD}}_{3_{I\Gamma}} \\ 0 & \widetilde{S} \end{bmatrix}, \tag{7.2.8}$$

based on the presentation in Section 5.4.4. Additionally, the aforementioned observations regarding $J^{\mathrm{DD}}_{3_{II}}$ suggest that the structure allows for independent block matrix inversions, which is a particularly useful quality in the application of a right preconditioning strategy due to the representation seen in (5.4.6). The task now is to determine an appropriate approximation to the Schur complement that can be inverted, stored and applied in an efficient manner.

Unlike previous approaches seen until this point, the variables $\mathbf{u}_\Gamma, \boldsymbol{\mathcal{M}}$ and π are all considered as part of $\mathbf{y}_{3_\Gamma}$, meaning that the Schur complement S^{DD} will take on a block 3×3 structure. Using linear algebra, this may be written down in the following manner

$$S^{\mathrm{DD}} := \begin{bmatrix} S_{11} & S_{12} & \mathbf{0} \\ S_{12}^T & S_{22} & \mathbf{1}_N \\ \mathbf{0}^T & \mathbf{1}_N^T & 0 \end{bmatrix},$$

where the matrices $S_{11} \in \mathbb{R}^{\hat{n}_\Gamma \times \hat{n}_\Gamma}$, $S_{12} \in \mathbb{R}^{\hat{n}_\Gamma \times N}$ and $S_{22} \in \mathbb{R}^{N \times N}$. The exact forms of each of these respective matrices may be written down as follows

$$\begin{aligned} S_{11} = {} & S_{\Gamma\Gamma} - K_{\Gamma I}K_{II}^{-1}B_I ZCYC^T ZB_I^T K_{II}^{-1}K_{I\Gamma} + K_{\Gamma I}K_{II}^{-1}B_I ZB_I^T K_{II}^{-1}K_{I\Gamma} + \dots \\ & + K_{\Gamma I}K_{II}^{-1}B_I ZCYC^T ZB_\Gamma^T - K_{\Gamma I}K_{II}^{-1}B_I ZB_\Gamma^T + \dots \\ & + B_\Gamma ZCYC^T ZB_I^T K_{II}^{-1}K_{I\Gamma} - B_\Gamma ZB_I K_{II}^{-1}K_{I\Gamma} + \dots \\ & + B_\Gamma ZB_\Gamma^T - B_\Gamma ZCYC^T ZB_\Gamma^T, \\ S_{12} = {} & YC^T ZB_\Gamma^T - YC^T ZB_I^T K_{II}^{-1}K_{I\Gamma}, \\ S_{22} = {} & -Y, \end{aligned}$$

where $S_{\Gamma\Gamma}$ denotes the Schur complement for the stiffness matrix K defined as per (6.1.15), and

$$\begin{aligned} Z &:= \left(B_I^T K_{II}^{-1} B_I + X^{-1} M + \widetilde{X}^{-1} Q\right)^{-1}, \\ Y &:= \left(C^T Z C\right)^{-1}. \end{aligned}$$

From the above, it is clear that the matrix Z is a key factor in the assembly. This matrix is diagonally dominant, with an effective approximation provided by

$$\widetilde{Z} := \left(X^{-1} M + \widetilde{X}^{-1} Q\right)^{-1}.$$

Under this approximation, the matrix $\widetilde{S}_{22} := \widetilde{Y} := \left(C^T \widetilde{Z} C\right)^{-1}$ has a diagonal structure, whereas $\widetilde{S}_{12} := \widetilde{Y} C^T \widetilde{Z} \left(B_\Gamma^T - B_I^T K_{II}^{-1} K_{I\Gamma}\right)$ may be computed cheaply in parallel. We therefore propose to approximate both constraining blocks S_{12} and S_{22} by $\widetilde{S}_{12}$ and $\widetilde{S}_{22}$ respectively leading to the following approximation of S^{DD}

$$S_A^{\mathrm{DD}} := \begin{bmatrix} S_{11} & \widetilde{S}_{12} & \mathbf{0} \\ \widetilde{S}_{12}^T & \widetilde{S}_{22} & \mathbf{1}_N \\ \mathbf{0}^T & \mathbf{1}_N^T & 0 \end{bmatrix}.$$

The main concern now involves the matrix S_{11}, associated with the interface displacement nodes. This block dominates the Schur complement for the Jacobian, and so the aim is therefore to construct a preconditioner based on the structure of S_A^{DD} above with a suitable approximation $\widetilde{S}_{11}$ to S_{11}. In the context of saddle point problems, the notion of constraint preconditioning has been subject to extensive analysis, as described (for instance) in [86, 113]. The plan is to consider adaptations of existing results from the literature for our problem, so that an effective approximation to S_A^{DD} may be derived.

It is clear from the expanded expression that S_{11} is convoluted in nature. However, due to the clear presence of $S_{\Gamma\Gamma}$, an intuitive approach would consider application of the results presented in Chapter 6 for the linear elasticity problem whilst retaining the constraining blocks due to the observations noted above. We therefore consider forming an approximation to S_A^{DD} in the following manner

$$S_0^{\mathrm{DD}} := \begin{bmatrix} S_{\Gamma\Gamma} & \widetilde{S}_{12} & \mathbf{0} \\ \widetilde{S}_{12}^T & \widetilde{S}_{22} & \mathbf{1}_N \\ \mathbf{0}^T & \mathbf{1}_N^T & 0 \end{bmatrix}. \tag{7.2.9}$$

In order to provide justification for this choice, an adaptation of a result presented in [86] by Keller, Gould and Wathen is described for the problem in question below.

Theorem 7.2.1. *Consider the generalised eigenvalue problem*

$$S_A^{DD}\bar{\mathbf{z}} = \lambda S_0^{DD}\bar{\mathbf{z}},$$

such that $\bar{\mathbf{z}} := \left(\bar{\mathbf{z}}_1^T, \bar{\mathbf{z}}_2^T\right)^T$ *with* $\mathbf{z} := \bar{\mathbf{z}}_1 \in \mathbb{R}^{\hat{n}_\Gamma+m-1}$. *Let the matrix* $\widetilde{G}$ *denote a basis for the nullspace of* $\mathbf{1}_N$. *Then,*

1. $\lambda = 1$ *with multiplicity* $N+1$.

2. *The remaining* $\hat{n}_\Gamma$ *eigenvalues satisfy the eigenvalue problem*

$$\left(S_{11} - W\right)\mathbf{z}_1 = \lambda\left(S_{\Gamma\Gamma} - W\right)\mathbf{z}_1, \tag{7.2.10}$$

where $W := \widetilde{S}_{12}\widetilde{G}\left(\widetilde{G}^T\widetilde{S}_{22}\widetilde{G}\right)^{-1}\left(\widetilde{S}_{12}\widetilde{G}\right)^T$.

Proof. Given in Appendix - see Theorem A.1. □

This results suggests that since $N+1$ eigenvalues are equal to 1, the increase in the size of the problem through the reformulation described in (7.2.3) is handled by retaining the constraining blocks within the preconditioner. Using (7.2.10), the main concern is clearly focussed on how effectively $S_{\Gamma\Gamma}$ approximates S_{11}. The expectation is that the dominant component of the spectrum of S_{11} will be well approximated by the associated eigenvalues of $S_{\Gamma\Gamma}$, which (if true) suggests that useful results may be obtained. Numerical figures displayed in the results section illustrate that this approximation does indeed lead to an effective, albeit non-practical, preconditioning strategy, suggesting further application of the results described in Chapter 6. More precisely, since Proposition 6.2.1 asserts spectral equivalence between $S_{\Gamma\Gamma}$ and a matrix representation of a discrete norm on S_h, the anticipation is for an effective preconditioning strategy to be realised by considering $\widetilde{S}_{11} = \widehat{H}_\theta$, with results obtained free from dependence of the underlying mesh parameter in the particular case of $\theta = 1/2$. We therefore propose to form an approximation S_1^{DD} to S^{DD} as follows

$$S_1^{\mathrm{DD}} := \begin{bmatrix} \widehat{H}_\theta & \widetilde{S}_{12} & \mathbf{0} \\ \widetilde{S}_{12}^T & \widetilde{S}_{22} & \mathbf{1}_N \\ \mathbf{0}^T & \mathbf{1}_N^T & 0 \end{bmatrix}.$$

The need to calculate fractional powers involved in the assembly of $\widehat{H}_\theta$ can be computationally prohibitive. Nevertheless, as discussed in Section 6.2.6, iterative alternatives to a generalised eigenvalue decomposition (or any other direct method) may be employed. Despite this, the main issue with the matrix S_1^{DD} used as an approximation to S^{DD} is that the application of its inverse to a vector is required at each GMRES iteration. It is possible (for instance) to consider iterative solution methods as opposed to direct computation of the inverse in order to alleviate this issue. The symmetric indefinite saddle point structure of S_1^{DD} would suggest use of the MINRES algorithm (or suitable alternatives), however this will not be considered further within this thesis. Instead, we would like to be able to exploit the block structure of the system (and also $\widehat{H}_\theta$) within our approximation to S_A^{DD} and ultimately S^{DD}. Based on Theorem 7.2.1, we consider a constrained counterpart to the matrix representation of the discrete norm presented in (6.2.11). By defining

$$L_C := \begin{bmatrix} \bigoplus_{i=1}^d L & \widetilde{S}_{12} & \mathbf{0} \\ \widetilde{S}_{12}^T & \widetilde{S}_{22} & \mathbf{1}_N \\ \mathbf{0} & \mathbf{1}_N^T & 0 \end{bmatrix}, \qquad M_C := \begin{bmatrix} \bigoplus_{i=1}^d M & \widetilde{S}_{12} & \mathbf{0} \\ \widetilde{S}_{12}^T & \widetilde{S}_{22} & \mathbf{1}_N \\ \mathbf{0} & \mathbf{1}_N^T & 0 \end{bmatrix},$$

we propose to form a practical approximation to S_A^{DD} in the following way

$$S_2^{\mathrm{DD}} := M_C \left(M_C^{-1} L_C\right)^{1-\theta}.$$

The indefiniteness suggests that the Lanczos procedure used as an alternative to the generalised eigenvalue decomposition for the evaluation of fractional matrix powers will not be applicable. An indefinite generalised Lanczos procedure can be considered with motivation from [138], which would allow for the generation of a three term recurrence with respect to the M_C^{-1} inner product. However, for this work results will be computed through the use of a generalised eigenvalue decomposition, since the indefinite Lanczos relies on the basis vectors being orthogonal with respect to an indefinite inner product, for which guarantees of linear independence cannot be assured.

Using the same line search procedure outlined in Section 4.5.3, we are in a position to describe the resulting primal-dual algorithm for the expanded topology optimisation problem described in (7.2.3), with the Newton step calculated through GMRES coupled with a preconditioner based on a decomposition of the domain. The full description is provided in Algorithm 7.3, bearing similarities to the primal-dual Newton algorithm provided in Algorithm 4.4.

Algorithm 7.3 *PRIMAL-DUAL NEWTON KRYLOV METHOD (EXPANDED)*

```
1. $k = 0, l = 0, r = 1, s = 1, IE = 1$.
2. $\mathbf{u} = K^{-1}\left(\left(V_{vol}/m\right)\, \mathbf{1}_m\right)\mathbf{f},\ \boldsymbol{\rho} = \left(V_{vol}/m\right)\, \mathbf{1}_m,\ \boldsymbol{\lambda} = \mathbf{1}_N,\ \boldsymbol{\kappa} = \mathbf{1}_m,\ \boldsymbol{\delta} = \mathbf{1}_m$.
3. $\mathbf{y}_3 := \left(\mathbf{u}_I^T, \boldsymbol{\lambda}^T, \boldsymbol{\rho}^T, \boldsymbol{\kappa}^T, \boldsymbol{\delta}^T, \mathbf{u}_\Gamma^T, \boldsymbol{\mathcal{M}}^T, \pi\right)^T$.
4. While $max(r, s) > \mathcal{T}$, Do
   (a) While $IE > \mathcal{T}_N$, Do
       i. $\Delta\mathbf{y}_3 := GMRES\left(J_3^{DD}\left(\mathbf{y}_3\right), \mathcal{R}_3^{DD}\left(\mathbf{y}_3\right), \widetilde{P}\left(\mathbf{y}_3\right)\right)$.
       ii. $\mathbf{y}_3 := \mathbf{y}_3 + \alpha_{LS}\Delta\mathbf{y}_3$.
       iii. $k_{it} := k_{it} + 1$.
       iv. $IE := \mathcal{R}_3^{DD}\left(\mathbf{y}_3\right)^T \Delta\mathbf{y}_3$.
   (b) $r := \beta_r r, s := \beta_s s$.
   (c) $l_{it} := l_{it} + 1$.
```

We now use Algorithm 7.3 to produce numerical results for the cantilever beam, MBB beam and rotating plate problems.

7.2.4 Numerical Results

Tables 7.5 and 7.6 provide results for the cantilever and MBB beam problems and also the rotating plate problem using the primal-dual Newton Krylov algorithm described in Algorithm 7.3. As in Section 7.2.1, the outer and inner tolerances (denoted $\mathcal{T}$ and $\mathcal{T}_N$) in Algorithm 7.3 were taken to be 10^{-7} and 10^{-5} respectively, with an adaptive stopping criteria used for GMRES as follows

$$\mathcal{T}_{\text{GMRES}} := r^{p_1} \times \mathcal{T}^{p_2} \times \|\mathcal{R}_3^{DD}\|_2^{p_3}.$$

The values q_1, q_2 and q_3 can be taken (in general) to be $0.2, 0.75$ and 0.5 respectively in a similar manner to the stopping tolerance described for the reduced primal-dual Newton Krylov algorithm presented in Section 7.2.1. As mentioned there, minor variations in the values of q_1, q_2 and q_3 were considered for certain test examples.

Each of the main rows in Table 7.5 provides a comparison between results obtained through the use of different approximations of S^{DD}. Included in the table are results for preconditioning with S_1^{DD} using $\widehat{H}_\theta$ for the interface displacement nodes, and the constrained preconditioner S_2^{DD}, as well as S_0^{DD} defined in (7.2.9). For both S_1^{DD} and S_2^{DD}, a generalised eigenvalue decomposition is used for the evaluation of $\widehat{H}_\theta$. This is to allow for a comparison between both approximations due to the indefiniteness of the matrices M_C and L_C present within S_2^{DD} in each of the test problems considered. Reading the table from left to right provides an indication as to how the solution method performs when the domain is decomposed into differing numbers of subdomains. By viewing the table in this manner, a logarithmic dependence on the total number of GMRES iterations can be seen. In comparison, reading the table from top to bottom (for each example) provides an indication of the performance of our solver under different mesh parameters. Here, we see that in all cases, the total number of iterations remains roughly constant (if anything, decreasing) suggesting solutions are obtained independently of the chosen mesh parameter. The values of θ listed in both Tables 7.5 and 7.6 were chosen by experimentation based on similar observations noted in Section 6.3, where it was found that different values of θ were able to provide a closer approximation to the decay of the inverse of the associated Steklov-Poincaré operator. Similar findings were found numerically in this work also, with the best values of θ used in order to produce the results recorded in the table.

By directly comparing the numerics from the main rows, it can be seen that preconditioning with S_1^{DD} appears to provide solutions with a smaller number of GMRES iterations than the constrained preconditioner S_2^{DD}. However, the fact that results are obtained in both cases free of dependence on the chosen mesh parameter suggests that S_2^{DD} represents a more practical preconditioner due to the associated inversion costs re-

			Newton Its. (Avr. GMRES)				Total GMRES Its.			
No. Subdomains:			4	16	64	256	4	16	64	256
Problem	Approx. to S^{DD}	h \ θ	0.5	0.6	0.7	0.75	0.5	0.6	0.7	0.75
Cant. Beam	S_0^{DD}	1/16	19 (5)	20 (11)	21 (15)	20 (36)	102	213	317	718
		1/32	19 (4)	21 (11)	23 (16)	21 (31)	84	238	361	643
		1/64	21 (4)	23 (9)	25 (13)	23 (27)	88	203	318	610
		1/128	22 (4)	23 (6)	26 (10)	24 (24)	82	145	247	564
	S_1^{DD}	1/16	22 (7)	22 (15)	21 (31)	21 (54)	158	321	655	1140
		1/32	23 (5)	23 (13)	25 (23)	25 (40)	123	294	573	989
		1/64	26 (4)	26 (11)	27 (20)	27 (34)	109	273	542	923
		1/128	28 (4)	28 (9)	29 (18)	29 (31)	104	260	527	890
	S_2^{DD}	1/16	22 (8)	22 (17)	22 (38)	22 (61)	182	377	829	1335
		1/32	24 (6)	24 (14)	25 (26)	25 (46)	138	333	654	1145
		1/64	27 (5)	28 (11)	28 (22)	28 (38)	125	318	615	1064
		1/128	29 (4)	29 (10)	29 (21)	30 (34)	119	303	597	1030
MBB Beam	S_0^{DD}	1/16	25 (6)	25 (12)	25 (23)	25 (46)	154	299	582	1147
		1/32	21 (5)	22 (7)	23 (18)	24 (36)	103	155	404	860
		1/64	21 (4)	23 (6)	24 (14)	23 (33)	78	143	328	751
		1/128	22 (3)	25 (4)	25 (11)	25 (28)	73	97	285	689
	S_1^{DD}	1/16	22 (12)	22 (24)	22 (46)	22 (73)	259	534	1013	1608
		1/32	24 (9)	25 (20)	25 (38)	25 (58)	216	495	939	1455
		1/64	27 (8)	27 (17)	28 (31)	28 (48)	205	449	873	1334
		1/128	30 (7)	29 (16)	31 (28)	31 (43)	199	451	864	1320
	S_2^{DD}	1/16	23 (13)	23 (27)	22 (54)	23 (86)	299	628	1189	1969
		1/32	25 (10)	26 (21)	26 (41)	26 (66)	254	540	1064	1705
		1/64	28 (9)	28 (19)	30 (34)	30 (53)	245	522	1033	1589
		1/128	32 (7)	32 (16)	31 (33)	32 (48)	239	515	1016	1546

Table 7.5: Number of iterations required to obtain the solution to (4.5.1) using the outlined solution method under preconditioning with S_0, S_1 and S_2 for a variety of different mesh and subdomain sizes. The results on the left of each respective column show the number of Newton iterations, with the bracketed numbers indicating the average number of GMRES iterations. The total number of GMRES iterations are displayed on the right.

quired in the application of S_1^{DD}. Importantly, both preconditioners when applied can be seen to exhibit similar characteristics to S_0^{DD}.

We now consider a direct comparison between each of the three solution methods described in Algorithms 7.1, 7.2 and 7.3 using Tables 7.1 to 7.6 for both the cantilever beam and also the rotating plate problems. In Section 7.1.2, Tables 7.1 and 7.2 displayed

	Newton Its. (Avr. GMRES)				Total GMRES Its.			
No. Subdomains:	4	16	60	240	4	16	60	240
h \ θ	0.5	0.6	0.7	0.75	0.5	0.6	0.7	0.75
1/32	19 (5)	19 (10)	20 (19)	21 (39)	95	188	389	813
1/64	22 (5)	22 (7)	23 (15)	22 (34)	99	159	346	737
1/128	24 (4)	25 (6)	25 (12)	25 (27)	86	141	312	683
1/256	26 (3)	25 (5)	26 (11)	26 (26)	83	118	295	665

Table 7.6: Results for the rotating plate problem solved using Algorithm 7.3 with $\widetilde{S} = S_2^{\mathrm{DD}}$ in (7.2.8) based on the observations in Table 7.5.

results for the fixed point solution method described in Algorithm 7.1. Whilst this approach was able to provide effective solutions for relatively coarse problems, a logarithmic dependence on the mesh parameter h was noted in the total number of GMRES iterations, largely due to the lack of control over the total number of fixed point iterations required for convergence. In comparison, the results for the reduced primal-dual Newton-Krylov algorithm provided in both Tables 7.3 and 7.4 showed significant improvements for finer mesh parameters, with roughly half the total number of GMRES iterations required for convergence in certain cases. However, a notable increase in the number of GMRES iterations was observed for particularly small barrier parameters, as illustrated in Figure 7.5.

The results for the expanded formulation displayed in Tables 7.5 and 7.6 (including those recorded for the MBB beam problem) bear similar characteristics to those recorded for the reduced formulation, however a general improvement is noted in both the total number of fixed point and GMRES iterations. Based on Theorem 7.2.1, this suggests that the Schur complement of $K(\boldsymbol{\rho})$ (namely $S_{\Gamma\Gamma}$) is able to provide an effective approximation to the component of the Schur complement for the expanded formulation related to the interface displacement nodes (namely S_{11}). Importantly, the results suggest that this approximation yields better results than the approximation of the Schur complement for the reduced system matrix described in (7.2.2). Nevertheless, questions should be asked regarding the size of the systems solved within each of the compared algorithms. The

Newton system associated with the expanded formulation involved a Jacobian matrix of size $\hat{n}+3m+2N+1$ square, whereas the reduced system was substantially smaller, being of size $\hat{n}+1$ square. Whilst one could argue that the size of the expanded system suggests that the associated computational costs would favour the reduced system, the full benefits of the reformulation outlined in (7.2.3) can only be realised through testing using parallel architecture.

CHAPTER 8

CONCLUDING REMARKS

8.1 Summary of the Thesis

In the introduction, it was mentioned that a key drawback in the widespread use of topology optimisation lies in the inherent large scale nature of the formulation of the problem. Even problems posed using relatively coarse mesh parameters can present computational issues, and so the task was to consider alternative approaches that allow for efficient wide scale use within a broad range of problems.

This thesis has explored different methodologies and approaches in the application of domain decomposition to problems arising in the field of topology optimisation. Based on current literature and also commonly used solution methods within the community, this has involved research into the application of domain decomposition to the governing equations of linear elasticity. Under a nonoverlapping decomposition of the domain, we have described a splitting of the original system into $2N$ decoupled systems posed purely on subdomains, along with an associated operator allowing for the determination of the interfacial component of the solution. The discretised system was solved using GMRES, coupled with a preconditioning strategy for the solution to the interface problem based on a matrix representation to a discrete fractional Sobolev norm.

Analytical and also numerical results were able to indicate mesh independent performance for our approach. However, direct application of the preconditioner involved the calculation of a square root of a matrix product. Whilst this may be achieved through use of a generalised eigenvalue decomposition, it was noted that the associated complexity for larger problems will present an issue for an increasing number of subdomains. However, iterative alternatives through use of the Lanczos or inverse Lanczos process have been explored in the literature, and were used and adapted to this work. The inverse Lanczos approach was found to deliver mesh independent results using only a relatively small number of basis vectors (independently of the chosen mesh parameter). However, direct inversion of the interface Laplacian matrix was required as part of this approach, with direct inversion potentially leading to computational issues. Instead, PCG iterations preconditioned using the interface Laplacian with the cross points removed was considered so that a parallel implementation may be realised. The resulting method was found to work well in practice, with notable speedup figures displayed for two different examples.

These observations motivated subsequent research and application to compliance minimisation problems in topology optimisation. Chapter 4 described commonly used solution methods for topology optimisation, including the optimality criteria method and the method of moving asymptotes. Both of these approaches can be viewed as fixed point type solution methods, involving separate treatment of the equilibrium equations from the rest of the first order necessary optimality conditions. It was noted in Section 5.5.4 that the majority of the computational effort was confined to the solution of these equations, which, at each fixed point step, amounts to solving the equations of linear elasticity. Therefore, application of the preconditioning strategy presented in the previous chapter was proposed, leading to the development of Algorithm 7.1. Although results were presented for the OC method, similar performance is obtained using the MMA. Whilst the computational results were promising, no assurances could be made over the total number of fixed point iterations required for convergence. A dependence on the mesh parameter was observed, largely due to variations in the values of θ in order to reduce the total number of GMRES iterations. Nevertheless, attention was turned to alternative approaches that were able to handle the nonlinearity in a more effective manner.

The other two algorithms presented in Chapter 7 were both based on a primal-dual interior point approach. These approaches deal with all of the constraints at the same time through the application of Newton's method to the associated equality constrained barrier problem. The resulting system was outlined in Chapter 4, along with commonly used solution stategies. One option is to exploit the diagonal structure of certain matrix blocks to reduce the size of the system matrix, with further reduction possible through use of an appropriate Schur complement.

In Section 7.2.1, the application of domain decomposition was considered for the reduced Newton system, with the resulting solution method described in Algorithm 7.2. The preconditioner used for the interface problem was again based on a matrix representation to a discrete fractional Sobolev norm, with results from use of this algorithm displayed for the MBB beam, cantilever beam and rotating plate examples. A direct comparison with results obtained using Algorithm 7.1 highlighted significant improvements for particularly fine mesh parameters, with savings in the total number of GMRES iterations in the region of 50% in some instances. Nevertheless, the number of GMRES iterations was found to grow substantially close to convergence in a number of cases. As the barrier parameters tended to zero, it was found that a component of the reduced Jacobian other than the stiffness matrix began to dominate behaviour. Consequently, our defined preconditioning strategy failed to provide an effective approximation to the inverse of the reduced Jacobian matrix close to convergence.

This observation led to further consideration of the original unreduced system, where the introduction of N additional variables was proposed, leading to an expanded nonlinear problem. Through consideration of the first order necessary optimality conditions for the associated equality constrained barrier problem, an alternative system was derived, involving an expanded Jacobian matrix. Splitting in a particular manner led to consideration of a 3×3 block Schur complement matrix, with straightforward approximations possible for certain blocks. Through Theorem 7.2.1, the main concern was seen to center on an efficient approximation of the Schur complement related to the interface displacement nodes. Further examination of this particular component indicated that a reasonable approximation amounted to the Schur complement of the stiffness matrix. Encouraging results suggested use of a matrix representation to a discrete fractional Sobolev norm, based on results displayed for the linear elasticity problem. Application of the resulting preconditioner was seen as an issue, leading to the consideration of a constrained fractional Sobolev norm based on constrained mass and Laplacian matrices. Results were produced for each of the MBB beam, cantilever beam and rotating plate problems, with figures suggesting spectral equivalence between the Schur complement and the constrained fractional Sobolev norm.

8.2 Future Investigations

Based on the findings mentioned above, we now describe a number of avenues for further exploration. In the short term, an iterative variant for the evaluation of fractional powers of indefinite matrices should be implemented (for instance, see [11]), since the direct approach used at present within both the reduced and expanded primal-dual interior point approaches will present computational issues for larger and more complex problems. Testing on parallel architecture should also be considered to enable a comparison of the computational complexity associated with each algorithm presented in Chapter 7.

Throughout this thesis, a decomposition of the domain (where considered) has been carried out in a regular fashion. Extensions to this work could potentially involve consideration of a non-regular decomposition based on expected areas of either high or low density. However, such areas are unknown for general problems and require an indication of what the final design will look like. During experimentation, it was noted that a rough outline of the final solution could be seen within a fairly small number of iterative steps for each of the algorithms presented within Chapter 7. Therefore, it would be interesting to determine whether an adaptive decomposition based on contrasting areas of density would be able to yield improved results.

The results recorded in Chapters 6 and 7 were shown to display a dependence on the number of subdomains considered. However, it was noted in Chapter 6 that variations in the value of θ within our interface preconditioner were able to provide more effective approximations to the decay in the inverse of the Steklov-Poincaré operator. Values were determined through numerical experimentation, however a thorough analysis is required to determine optimal values for a wide range of problems divided into (potentially) non-regular subdomains. A two-level method may be used to remove domain dependence, and can be applied within our framework. Such approaches look to determine solutions both globally and locally for two respective mesh parameters h_1 and h_2, with $h_1 > h_2$ so that h_1 represents a coarse mesh and h_2 a fine mesh. For the interested reader, further details are available in [191, pp. 55 – 86].

Another area for further exploration involves a decomposition through consideration of artificial forces on each subdomain. For the linear elasticity problem (for instance), a splitting into two subdomains could be viewed in the following manner

$$a(u, v) = F(v) \ \rightarrow \ \begin{cases} a(u_1, v) &= \bar{F}(v), \\ a(u_2, v) &= F(v) - \bar{F}(v), \end{cases}$$

with solution $u = u_1 + u_2$. The linear functional $\bar{F}$ remains to be determined, however a fixed point iterative process can be considered through an appropriate initial guess.

The primal-dual interior point solution methods presented in Chapter 7 can be viewed as part of a broader class of so-called Newton-Krylov type methods. As seen within the thesis, such approaches involve linearisation of nonlinear problems over the whole domain through use of a Newton-type algorithm. The resulting linear system is then solved using a Krylov method, coupled with an appropriate preconditioning strategy. This work has seen preconditioners based on a decomposition of the domain, targetting the resulting Schur complement problem. Through linearisation in this manner, the computational effort is distributed uniformly across the whole domain. For domains involving a relatively

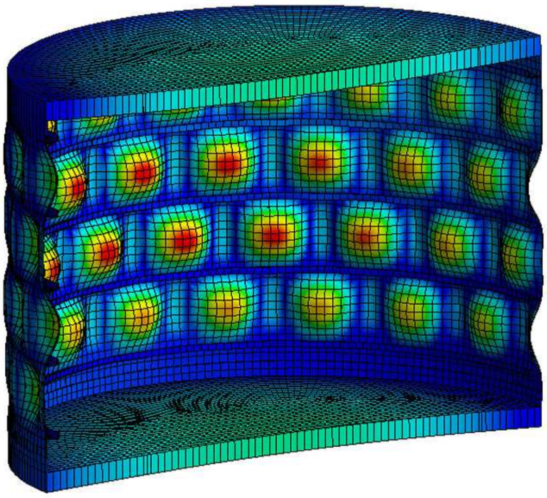


Figure 8.1: Illustration of the effect of nonlinearities on a three-dimensional domain. The output illustrates buckling of a stiffened vessel, as printed in [81] (reproduced within this thesis with kind permission).

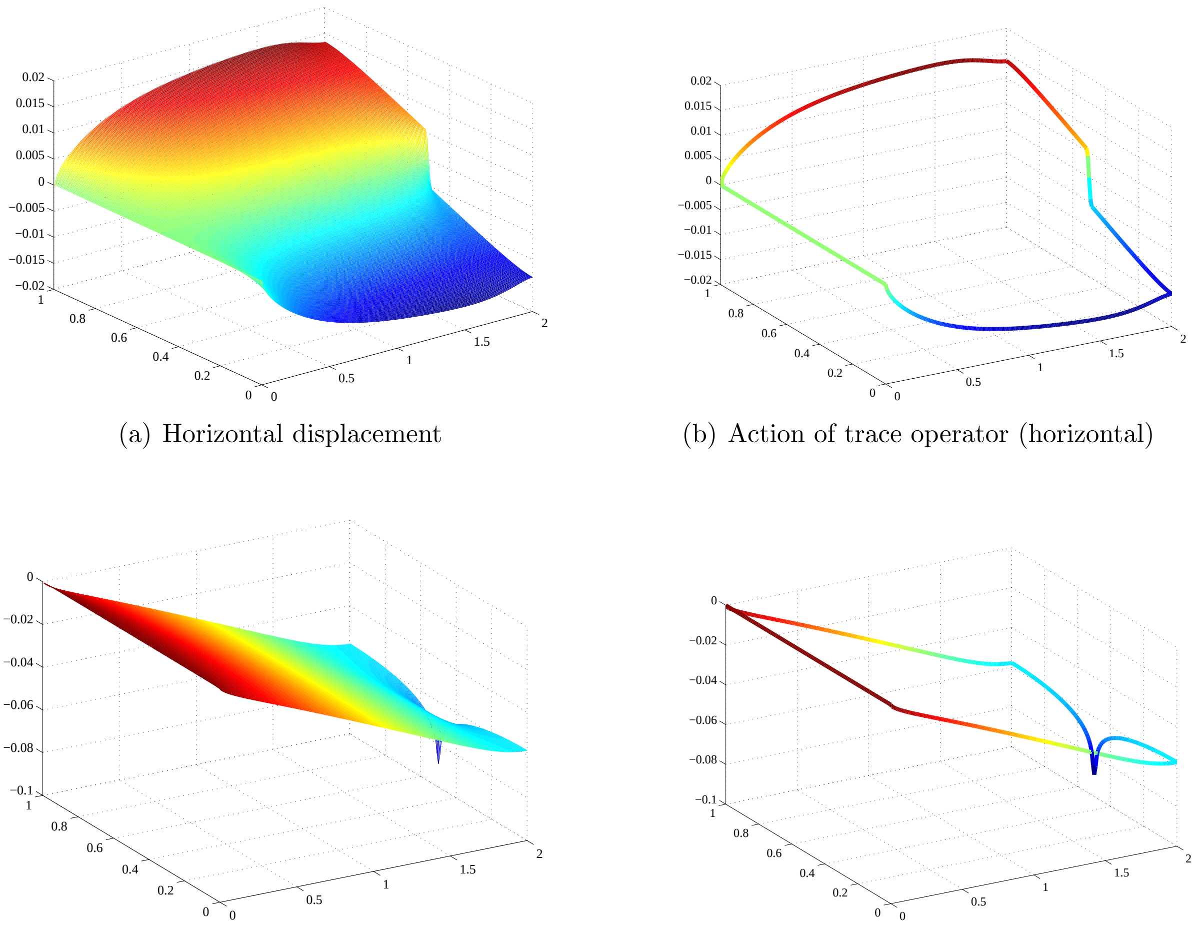
even distribution of nonlinearity, the application of Newton in the context of such methods can be seen to exhibit near quadratic convergence. However, when modelling problems in industry, the considered domains will typically involve localised areas of strong nonlinearity, as displayed (for instance) in obstacle, contact and fracture problems. Linearisation across the whole domain through (4.5.15) fails to exploit the local nature of nonlinearities, meaning that convergence across the whole domain becomes dictated by local phenomena. As a result, local nonlinearities may have a direct impact on the overall convergence of Newton's method, as illustrated in both [32] and [42]. Consequently, Newton-Krylov approaches can be expected to struggle when faced with domains containing local nonlinearities. A depiction of such a domain is provided in Figure 8.1 (as presented in [81]), illustrating nonlinear buckling of a stiffened vessel.

Therefore, it is natural to try to consider approaches in order to deal with the nonlinearities on a local scale, so that the overall effect on the rest of the problem is significantly reduced. Work to this effect has been considered based on an overlapping ([33] by Cai and Li) and a nonoverlapping ([139] by Pebrel et. al., and [162] by Sassi) decomposition of the domain. Potential areas of improvement were noted from each, and used in the development of [193], involving the splitting of a class of nonlinear problems into a three step procedure wrapped around a fixed point iteration. The first and third steps involved a total of $2N$ subproblems, enabling for the potential determination of solutions in parallel. The second step required the solution to a system posed purely on the interface, which was preconditioned at each fixed point step using a matrix representation to an appropriate discrete fractional Sobolev norm. Numerical results presented within the work were able to illustrate independence with relation to both the mesh size and the number of subdomains used. Comparisons to the corresponding Newton-Krylov method illustrated that our approach was shown to deliver competitive results, where the plan remains to extend the observations from this work to problems in topology optimisation.

APPENDIX

A.1 Illustrative Example of the Trace Operator

The following provides an illustrative example of the action of the trace operator described in Lemma 1.2.3 applied to the solution of a typical linear elasticity problem. The example considered here is the cantilever beam problem described in Section 3.6.1, with the figures in Figure 3.4 repeated here to enable a direct comparison.



(a) Horizontal displacement

(b) Action of trace operator (horizontal)

(c) Vertical displacement

(d) Action of trace operator (vertical)

Figure A.1: Illustration of the action of the trace operator described in Lemma 1.2.3. On the left, displacement profiles are provided in both horizontal and vertical directions, with the corresponding action of the trace operator provided on the right.

A.2 Derivation of Cauchy's First Law of Motion

This section is included in order to supplement the presentation provided in Section 2.1.

The strain caused as a result of the displacements u is characterised by the symmetric linearized strain tensor:

$$\boldsymbol{\epsilon}(u) = \{\epsilon_{ij}(u)\}_{i,j=1}^{d}, \qquad \epsilon_{ij}(u) = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right).$$

In a similar way, the stresses are characterized by the symmetric stress tensor,

$$\boldsymbol{\sigma}(u) = \{\sigma_{ij}(u)\}_{i,j=1}^{d},$$

which can be used to describe the contact forces acting on Ω. Each component of stress represents the force per unit area acting in the direction x_i on a surface with unit normal pointing in the direction of x_j. Therefore, the force on a small surface δs with outward pointing normal $\mathbf{n}$ can be expressed as $(\boldsymbol{\sigma}(u) \cdot \mathbf{n})\, \delta s$. Now, the total force F on the domain Ω can be written in terms of both internal and external components as

$$F := \int_{\Omega} f \,\mathrm{d}\mathbf{x} + \int_{\partial\Omega} \boldsymbol{\sigma}(u) \cdot \mathbf{n} \,\mathrm{d}s. \tag{A.1}$$

By application of the divergence theorem, namely

$$\int_{\partial\Omega} \boldsymbol{\sigma}(u) \cdot \mathbf{n} \,\mathrm{d}s = \int_{\Omega} \nabla \cdot \boldsymbol{\sigma}(u) \,\mathrm{d}\mathbf{x},$$

the equation (A.1) can be rewritten as

$$F = \int_{\Omega} \left(f + \nabla \cdot \boldsymbol{\sigma}(u)\right) \,\mathrm{d}\mathbf{x}, \tag{A.2}$$

where the operator $\nabla \cdot \boldsymbol{\sigma}(u)$ is defined in the following way

$$(\nabla \cdot \boldsymbol{\sigma}(u))_i = \sum_{j=1}^{d} \frac{\partial \sigma_{ij}(u)}{\partial x_j}.$$

In a state of elastostatic equilibrium, the resultant forces acting on the domain must sum to zero. In this situation, the loading rate will be sufficiently small for the acceleration to be treated as negligible. Using (A.2) we obtain

$$f + \nabla \cdot \boldsymbol{\sigma}(u) = 0,$$

corresponding to Cauchy's first law of motion, stating that the net force vanishes on every material particle over the whole domain.

A.3 Derivation of Matrix Representations using the Lanczos process

This material is designed to supplement the presentation in Section 6.2.6, and to illustrate to the interested reader the derivation of the respective representations of both H_θ and $\widehat{H}_\theta$ presented in (6.2.17) and (6.2.18).

In exact arithmetic, the Lanczos algorithm run to completion with the matrix pair (L, M) allows for H_θ and $\widehat{H}_\theta$ to be described as follows

$$\begin{aligned} H_\theta &= \bigoplus_1^d \left[M + M \left(M^{-1} L \right)^{1-\theta} \right] \\ &= \bigoplus_1^d \left[M + M V T^{1-\theta} V^{-1} \right] \\ &= \bigoplus_1^d \left[M V V^T M + M V T^{1-\theta} V^T M \right] \\ &= \bigoplus_1^d \left[M V \left(I_{n_\Gamma} + T^{1-\theta} \right) V^T M \right]. \end{aligned}$$

$$\begin{aligned}\widehat{H}_\theta &= \bigoplus_{1}^{d}\left[M\left(M^{-1}L\right)^{1-\theta}\right] \\ &= \bigoplus_{1}^{d}\left[MVT^{1-\theta}V^TM\right].\end{aligned}$$

Similarly (again in exact arithmetic), the Lanczos algorithm run to completion with the matrix pair (M^{-1}, L^{-1}) allows for H_θ and $\widehat{H}_\theta$ to be described as follows

$$\begin{aligned}H_\theta &= \bigoplus_{1}^{d}\left[M + MVT^{1-\theta}V^{-1}\right] \\ &= \bigoplus_{1}^{d}\left[M + MVT^{1-\theta}V^TL^{-1}\right] \\ &= \bigoplus_{1}^{d}\left[MVV^TL^{-1} + MVT^{1-\theta}V^TL^{-1}\right] \\ &= \bigoplus_{1}^{d}\left[MV\left(I_{n_\Gamma} + T^{1-\theta}\right)V^TL^{-1}\right]. \\ \widehat{H}_\theta &= \bigoplus_{1}^{d}\left[MVT^{1-\theta}V^TL^{-1}\right].\end{aligned}$$

A.4 Additional Results for Algorithm 7.1

	Newton Its. (Avr. GMRES)				Total GMRES Its.			
No. Subdomains:	4	16	64	256	4	16	64	256
θ	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
$h = 1/16$	14 (9)	14 (18)	16 (34)	16 (56)	121	257	536	894
1/32	17 (11)	17 (22)	18 (39)	19 (57)	181	374	696	1082
1/64	21 (12)	21 (23)	23 (38)	24 (56)	244	486	877	1349
1/128	30 (12)	32 (22)	34 (39)	35 (54)	365	694	1312	1881

Table A.1: Results for the cantilever beam problem solved using our preconditioning strategy for the solution to the equilibrium equations coupled with the OC method for the density update, with $\theta = 0.5$ in all cases.

A.5 Generalisation of Theorem 7.2.1

Theorem A.1. *Consider the generalised eigenvalue problem*

$$\mathcal{N}_1\bar{\mathbf{z}} = \lambda\mathcal{N}_2\bar{\mathbf{z}}, \tag{A.3}$$

such that $\bar{\mathbf{z}} := \left(\bar{\mathbf{z}}_1^T, \bar{\mathbf{z}}_2^T\right)^T$ *with* $\mathbf{z} := \bar{\mathbf{z}}_1 \in \mathbb{R}^{n+m-1}$*, where*

$$\mathcal{N}_1 := \begin{bmatrix} N_1 & B & \mathbf{0}_n \\ B^T & F & \mathbf{1}_m \\ \mathbf{0}_n^T & \mathbf{1}_m^T & 0 \end{bmatrix}, \qquad \mathcal{N}_2 := \begin{bmatrix} N_2 & B & \mathbf{0}_n \\ B^T & F & \mathbf{1}_m \\ \mathbf{0}_n^T & \mathbf{1}_m^T & 0 \end{bmatrix},$$

with $N_1, N_2 \in \mathbb{R}^{n\times n}$ *and* $F \in \mathbb{R}^{m\times m}$ *nonsingular. Let* $\widetilde{G}$ *be a basis for the nullspace of* $\mathbf{1}_m$*. Then,*

1. $\lambda = 1$ *with multiplicity* $m + 1$*.*

2. *The remaining* n *eigenvalues satisfy the eigenvalue problem*

$$\left(N_1 - W\right)\mathbf{z}_1 = \lambda\left(N_2 - W\right)\mathbf{z}_1, \tag{A.4}$$

where $W := B\widetilde{G}\left(\widetilde{G}^T F\widetilde{G}\right)^{-1}\left(B\widetilde{G}\right)^T$*.*

Proof. Following the presentation in [86], an appropriate QR factorisation may be considered, with an augmented matrix based on the resulting Q multiplied on both sides of (A.3) to yield a system of size $n + m + 1$, with exactly two eigenvalues equal to 1. The remaining $n + m - 1$ eigenvalues can then be seen to satisfy the following generalised eigenvalue problem

$$G^T\begin{bmatrix} N_1 & B \\ B^T & F \end{bmatrix}G\mathbf{z} = \lambda G^T\begin{bmatrix} N_2 & B \\ B^T & F \end{bmatrix}G\mathbf{z},$$

where G represents a basis for the null space of $\left[\mathbf{0}_n^T, \mathbf{1}_m^T\right]$. Since $\left[\mathbf{0}_n^T, \mathbf{1}_m^T\right]$ has rank 1, G may be written in the following form

$$G = \left[\begin{array}{c:c} I_n & 0_{n\times(m-1)} \\ \hdashline 0_{m\times n} & \widetilde{G} \end{array}\right], \tag{A.5}$$

where

$$\widetilde{G} := \begin{bmatrix} I_{m-1} \\ -\mathbf{1}_{m-1}^T \end{bmatrix}.$$

By using G as described in (A.5), we arrive at the following

$$\begin{bmatrix} N_1 & B\widetilde{G} \\ \widetilde{G}^T B^T & \widetilde{G}^T F \widetilde{G} \end{bmatrix} \mathbf{z} = \lambda \begin{bmatrix} N_2 & B\widetilde{G} \\ \widetilde{G}^T B^T & \widetilde{G}^T F \widetilde{G} \end{bmatrix} \mathbf{z}. \tag{A.6}$$

Written out, we have the following pair of simultaneous equations

$$N_1 \mathbf{z}_1 + B\widetilde{G}\mathbf{z}_2 = \lambda \left(N_2 \mathbf{z}_1 + B\widetilde{G}\mathbf{z}_2 \right), \tag{A.7a}$$

$$\widetilde{G}^T B^T \mathbf{z}_1 + \widetilde{G}^T F \widetilde{G}\mathbf{z}_2 = \lambda \left(\widetilde{G}^T B^T \mathbf{z}_1 + \widetilde{G}^T F \widetilde{G}\mathbf{z}_2 \right). \tag{A.7b}$$

We now distinguish the following two cases

1. Case 1: $\lambda = 1$. In this case, the equations balance by requiring $\mathbf{z}_1 = \mathbf{0}_n$. Therefore, the remaining eigenvectors of (A.6) amount to $\mathbf{e}_i$, with $i = n+1, \ldots, n+m-1$. Consequently, there are an additional $m-1$ eigenvalues equal to 1, which when added to the 2 mentioned earlier amounts to a total of $m+1$ eigenvalues equal to 1.

2. Case 2: $\lambda \neq 1$. In order to balance terms in (A.7b), it must hold that

$$\widetilde{G}^T B^T \mathbf{z}_1 + \widetilde{G}^T F \widetilde{G}\mathbf{z}_2 = \mathbf{0}_{m-1} \implies \mathbf{z}_2 = -\left(\widetilde{G}^T F \widetilde{G} \right)^{-1} \widetilde{G}^T B^T \mathbf{z}_1.$$

   Substitution of $\mathbf{z}_2$ written in this manner into (A.7a) leads to the generalised eigenvalue problem of size $n \times n$ described in (A.4), as required.

□

LIST OF REFERENCES

[1] N. Aage and B. S. Lazarov. Parallel framework for topology optimization using the method of moving asymptotes. *Structural and Multidisciplinary Optimization*, 47(4):493 – 505, 2013.

[2] R. A. Adams and J. J. F. Fournier. *Sobolev Spaces*, volume 140. Academic Pr, 2003.

[3] S. Agmon, A. Douglis, and L. Nirenberg. Estimates Near the Boundary for Solutions of Elliptic Partial Differential Equations Satisfying General Boundary Conditions. I. *Comm. Pure Appl. Math.*, 12:623 – 727, 1959.

[4] E. Andreassen, A. Clausen, M. Schevenels, B. S. Lazarov, and O. Sigmund. Efficient topology optimization in matlab using 88 lines of code. *Structural and Multidisciplinary Optimization*, 43:1 – 16, 2011.

[5] M. Arioli, D. Kourounis, and D. Loghin. Discrete fractional Sobolev norms for domain decomposition preconditioning. *IMA J. Numer. Anal.*, 33(1):318 – 342, 2013.

[6] M. Arioli and D. Loghin. Discrete Interpolation Norms with Applications. *SIAM J. Numer. Anal.*, 47(4):2924 – 2951, 2009.

[7] W. E. Arnoldi. The principle of minimized iterations in the solution of the matrix eigenvalue problem. *Quarterly of Applied Mathematics*, 9(1):17 – 29, 1951.

[8] O. Axelsson. Conjugate gradient type methods for unsymmetric and inconsistent systems of linear equations. *Linear Algebra Appl.*, 29:1 – 16, 1980.

[9] O. Axelsson. A generalized conjugate gradient, least square method. *Numer. Math.*, 51(2):209 – 227, 1987.

[10] I. Babuška. On the Schwarz algorithm in the theory of differential equations of mathematical physics. *Math. J*, 8:328 – 342, 1958.

[11] Z. Bai, T. Ericsson, and T. Kowalski. Symmetric Indefinite Lanczos Method. *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide, Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst, eds., SIAM, Philadelphia*, pages 249 – 260, 2000.

[12] A. Ben-Tal, M. Kočvara, A. Nemirovski, and J. Zowe. Free Material Design via Semidefinite Programming: The Multiload Case with Contact Conditions. *SIAM review*, 42(4):695 – 715, 2000.

[13] M. P. Bendsøe. Optimal shape design as a material distribution problem. *Structural and Multidisciplinary Optimization*, 1(4):193 – 202, 1989.

[14] M. P. Bendsøe and N. Kikuchi. Generating optimal topologies in structural design using a homogenization method. *Computer Methods in Applied Mechanics and Engineering*, 71(2):197 – 224, 1988.

[15] M. P. Bendsøe and O. Sigmund. Material interpolation schemes in topology optimization. *Archive of Applied Mechanics*, 69(9-10):635 – 654, 1999.

[16] M. P. Bendsøe and O. Sigmund. *Topology Optimization: Theory, Methods, and Applications*. Springer Verlag, Providence, Rhode Island, 2003.

[17] M. Berggren and F. Kasolis. Weak Material Approximation of Holes with Traction-Free Boundaries. *SIAM J. Numer. Anal.*, 50(4):1827 – 1848, 2012.

[18] S. I. Birbil, J. B. G. Frenk, and G. J. Still. An elementary proof of the Fritz-John and Karush-Kuhn-Tucker conditions in nonlinear programming. *European Journal of Operational Research*, 180(1):479 – 484, 2007.

[19] G. Biros and O. Ghattas. Parallel Lagrange–Newton–Krylov–Schur Methods for PDE-Constrained Optimization. Part I: The Krylov–Schur Solver. *SIAM J. Sci. Comput.*, 27(2):687–713 (electronic), 2005.

[20] G. Biros and O. Ghattas. Parallel Lagrange–Newton–Krylov–Schur Methods for PDE-Constrained Optimization. Part II: The Lagrange–Newton Solver and Its Application to Optimal Control of Steady Viscous Flows. *SIAM J. Sci. Comput.*, 27(2):714–739 (electronic), 2005.

[21] P. E. Bjørstad and O. B. Widlund. Iterative Methods for the Solution of Elliptic Problems on Regions Partitioned into Substructures. *SIAM J. Numer. Anal.*, 23(6):1097–1120, 1986.

[22] M. Bogomolny and O. Amir. Conceptual design of reinforced concrete structures using topology optimization with elastoplastic material modeling. *International Journal for Numerical Methods in Engineering*, 90(13):1578 – 1597, 2012.

[23] T. Borrvall and J. Petersson. Large-scale topology optimization in 3D using parallel computing. *Computer Methods in Applied Mechanics and Engineering*, 190(46-47):6201 – 6229, 2001.

[24] T. Borrvall and J. Petersson. Topology optimization using regularized intermediate density control. *Computer Methods in Applied Mechanics and Engineering*, 190(37-38):4911 – 4928, 2001.

[25] M. Botchev. A.N. Krylov: A Short Biography
`http://www.staff.science.uu.nl/ vorst102/kryl.html`
Accessed 27/09/2014.

[26] J. H. Bramble, J. E. Pasciak, and A. H. Schatz. The Construction of Preconditioners for Elliptic Problems by Substructuring I-IV. *Mathematics of Computation.* 47(175):103 – 134; 49(184):1 – 16; 51(184):415 – 430; 53(187):1 – 24, 1986 – 1989.

[27] S. C. Brenner. The Condition Number of the Schur Complement in Domain Decomposition. *Numer. Math*, pages 187 – 203, 1999.

[28] F. Brezzi and L. D. Marini. A three-field domain decomposition method. *Contemporary Mathematics*, 157, 1994.

[29] F. E. Browder. On some approximation methods for solutions of the Dirichlet problem for linear elliptic equations of arbitrary order. *J. Math. Mech*, 7:69 – 80, 1958.

[30] T. E. Bruns and O. Sigmund. Toward the topology design of mechanisms that exhibit snap-through behavior. *Computer Methods in Applied Mechanics and Engineering*, 193(36):3973 – 4000, 2004.

[31] R. H. Byrd, M. E. Hribar, and J. Nocedal. An Interior Point Algorithm for Large-Scale Nonlinear Programming. *SIAM J. Optim.*, 9(4):877–900, 1999. Dedicated to John E. Dennis, Jr., on his 60th birthday.

[32] X. C. Cai and D. E. Keyes. Nonlinearly Preconditioned Inexact Newton Algorithms. *SIAM J. Sci. Comput.*, 24(1):183–200 (electronic), 2002.

[33] X. C. Cai and X. Li. Inexact Newton Methods with Restricted Additive Schwarz based Nonlinear Elimination for Problems with High Local Nonlinearity. *SIAM J. Sci. Comput.*, 33(2):746–762, 2011.

[34] C. W. Carroll. The Created Response Surface Technique for Optimizing Restrained Systems. *Operations Research*, 9(2):169 – 184, 1961.

[35] M. Cavazzuti, A. Baldini, E. Bertocchi, D. Costi, E. Torricelli, and P. Moruzzi. High performance automotive chassis design: A topology optimization based approach. *Structural and Multidisciplinary Optimization*, 44(1):45 – 56, 2011.

[36] M. Cavazzuti, D. Costi, A. Baldini, and P. Moruzzi. Automotive Chassis Topology Optimization: a Comparison Between Spider and Coupé Designs. In *Proceedings of the World Congress on Engineering*, volume 3, pages 6 – 8, 2011.

[37] T. F. Chan and T. P. Mathew. Domain decomposition algorithms. *Acta numerica*, 3(1):61 – 143, 1994.

[38] G. Cheng and Z. Jiang. Study on topology optimization with stress constraints. *Engineering Optimization*, 20(2):129 – 148, 1992.

[39] G. D. Cheng and X. Guo. ε-relaxed approach in structural topology optimization. *Structural and Multidisciplinary Optimization*, 13(4):258 – 266, 1997.

[40] G. Chiandussi, I. Gaviglio, and A. Ibba. Topology Optimisation of an Automotive Component Without Final Volume Constraint Specification. *Advances in Engineering Software*, 35(10):609 – 617, 2004.

[41] I. Colominas, J. Parıs, F. Navarrina, and M. Casteleiro. High Performance Parallel Computing in Structural Topology Optimization. In *Proceedings of the 12th International Conference on Civil, Structural and Environmental Engineering Computing, BHV Topping, LF Costa Neves, RC Barros,(Editors), Civil-Comp Press, Stirlingshire, United Kingdom, paper*, volume 234, 2009.

[42] P. Cresta, O. Allix, C. Rey, and S. Guinard. Nonlinear localization strategies for domain decomposition methods: application to post-buckling analyses. *Computer Methods in Applied Mechanics and Engineering*, 196(8):1436 – 1446, 2007.

[43] G. C. A. DeRose Jr and A. R. Diaz. Solving three-dimensional layout optimization problems using fixed scale wavelets. *Computational mechanics*, 25(2):274 – 285, 2000.

[44] Q. V. Dinh, R. Glowinski, B. Mantel, J. Periaux, and P. Perrier. Subdomain solutions of nonlinear problems in fluid dynamics on parallel processors. In *5th International Symposium on Computational Methods in Applied Sciences and Engineering, Versailles, Amsterdam*, 1981.

[45] J. Dongarra and F. Sullivan. Guest Editors' Introduction: The Top 10 Algorithms. *Computing in Science & Engineering*, pages 22 – 23, 2000.

[46] M. Dryja. A capacitance matrix method for Dirichlet problem on polygon region. *Numerische Mathematik*, 39(1):51 – 64, 1982.

[47] M. Dryja. A finite element – Capacitance method for elliptic problems on regions partitioned into subregions. *Numerische Mathematik*, 44(2):153 – 168, 1984.

[48] I. S. Duff, A. M. Erisman, and J. K. Reid. *Direct Methods for Sparse Matrices.* Monographs on Numerical Analysis. The Clarendon Press, Oxford University Press, New York, second edition, 1989. Oxford Science Publications.

[49] P. Duysinx and M. P. Bendsøe. Topology optimization of continuum structures with local stress constraints. *International Journal for Numerical Methods in Engineering*, 43(8):1453 – 1478, 1998.

[50] A. S. El-Bakry, R. A. Tapia, T. Tsuchiya, and Y. Zhang. On the formulation and theory of the Newton interior-point method for nonlinear programming. *Journal of Optimization Theory and Applications*, 89(3):507 – 541, 1996.

[51] H. C. Elman. *Iterative Methods for Large Sparse Nonsymmetric System of Linear Equations.* PhD thesis, Yale University, Computer Science Dept., New Haven, CT, 1982.

[52] L. Euler. Novi commentarii Academiae Scientiarum Imperialis Petropolitanae, volume 5. Petropolis, Typis Academiae Scientarum, 1760.

[53] L. Euler. Sur la force des colonnes. *Memoires de L'Academie des Sciences et Belles-Lettres*, 13:252 – 282, 1759.

[54] L. C. Evans. *Partial Differential Equations.* Graduate studies in mathematics. American Mathematical Society, 1998.

[55] A. Evgrafov, C. J. Rupp, K. Maute, and M. L. Dunn. Large-scale parallel topology optimization using a dual-primal substructuring solver. *Structural and Multidisciplinary Optimization*, 36(4):329 – 345, 2008.

[56] C. Farhat, M. Lesoinne, P. LeTallec, K. Pierson, and D. Rixen. FETI-DP: a dual–primal unified FETI method Part I: A faster alternative to the two-level FETI method. *International Journal for Numerical Methods in Engineering*, 50(7):1523 – 1544, 2001.

[57] A. V. Fiacco and G. P. McCormick. Nonlinear Programming: Sequential Unconstrained Minimization Techniques. *Wiley, New York*, 1968.

[58] A. V. Fiacco and G. P. McCormick. *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, volume 4 of *Classics in Applied Mathematics.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, second edition, 1990.

[59] A. Forsgren, P. E. Gill, and J. R. Shinnerl. Stability of Symmetric Ill-Conditioned Systems Arising in Interior Methods for Constrained Optimization. *SIAM J. Matrix Anal. Appl.*, 17(1):187 – 211, 1996.

[60] K. R. Frisch. The Logarithmic Potential Method of Convex Programming. *Memorandum of May*, 13, 1955.

[61] M. J. Gander et al. Schwarz methods over the course of time. *Electronic Transactions on Numerical Analysis*, 31:228 – 255, 2008.

[62] R. Glowinski and O. Pironneau. Numerical Methods for the First Biharmonic Equation and the Two-Dimensional Stokes Problem. *SIAM Rev.*, 21(2):167 – 212, 1979.

[63] S. Grihon, L. Krog, A. Tucker, and K. Hertel. A380 Weight Savings Using Numerical Structural Optimization. In *20th AAAF Colloquium on Material for Aerospace Applications, Paris, France*, pages 763 – 766, 2004.

[64] C. Grossmann and H. G. Roos. *Numerical Treatment of Partial Differential Equations.* Universitext. Springer, Berlin, 2007. Translated and revised from the 3rd (2005) German edition by M. Stynes.

[65] Martin Grötschel, editor. *Optimization Stories*, volume Extra Volume ISMP (2012). DOCUMENTA MATHEMATICA, Journal der Deutschen Mathematiker-Vereinigung, 2012.

[66] M. E. Gurtin. *An Introduction To Continuum Mechanics.* Mathematics in science and engineering. Elsevier India, 2003.

[67] J. Haslinger and R. A. E. Mäkinen. *Introduction to Shape Optimization: Theory, Approximation, and Computation*, volume 7. Society for Industrial Mathematics, 2003.

[68] J. Haslinger and P. Neittaanmäki. *Finite element approximation for optimal shape, material and topology design.* Wiley Chichester, 1996.

[69] B. Hassani and E. Hinton. A Review of Homogenization and Topology Optimization I – Homogenization Theory for Media with Periodic Structure. *Computers & Structures*, 69(6):707 – 717, 1998.

[70] B. Hassani and E. Hinton. A Review of Homogenization and Topology Optimization II – Analytical and Numerical Solution of Homogenization Equations. *Computers & Structures*, 69(6):719 – 738, 1998.

[71] B. Hassani and E. Hinton. A Review of Homogenization and Topology Optimization III – Topology Optimization using Optimality Criteria. *Computers & Structures*, 69(6):739 – 756, 1998.

[72] M. R. Hestenes and E. Stiefel. Methods of Conjugate Gradients for Solving Linear Systems. *J. Research Nat. Bur. Standards*, 49:409 – 436 (1953), 1952.

[73] R. Hooke and T. A Jeeves. Direct Search Solution of Numerical and Statistical Problems. *Journal of the Association for Computing Machinery*, 8(2):212 – 219, 1961.

[74] R. H. W. Hoppe, C. Linsenmann, and S. I. Petrova. Primal-dual Newton methods in structural optimization. *Comput. Vis. Sci.*, 9(2):71 – 87, 2006.

[75] R. H. W. Hoppe and S. I. Petrova. Applications of Primal-Dual Interior Methods in Structural Optimization. *Comput. Methods Appl. Math.*, 3(1):159 – 176 (electronic), 2003. Dedicated to Raytcho Lazarov.

[76] R. H. W. Hoppe and S. I. Petrova. Primal–Dual Newton Interior Point Methods in Shape and Topology Optimization. *Numerical Linear Algebra with Applications*, 11(5-6):413 – 429, 2004.

[77] R. H. W. Hoppe, S. I. Petrova, and V. Schultz. 3D Structural Optimization in Electromagnetics. In *Proc. of the 13th Int. Conf. "Domain Decomposition Methods and Applications", Lyon, October 9 – 12 2000*, pages 479 – 486. CIMNE, Barecelona, 2002.

[78] R. H. W. Hoppe, S. I. Petrova, and V. Schulz. Primal-Dual Newton-type Interior-Point Method for Topology Optimization. *J. Optim. Theory Appl.*, 114(3):545 – 571, 2002.

[79] R. A. Horn and C. R. Johnson. *Matrix Analysis.* Cambridge University Press, Cambridge, 1990. Corrected reprint of the 1985 original.

[80] A. S. Householder and F. L. Bauer. On certain methods for expanding the characteristic polynomial. *Numer. Math.*, 1:29 – 37, 1959.

[81] ANSYS Inc. Nonlinear Mechanics for Structural Engineering Simulation from ANSYS.
`http://www.ansys.com/Products/Simulation+Technology/Structural+Analysis/Nonlinear+Mechanics+for+Structural+Engineering+Simulation+from+ANSYS`
Accessed 27/09/2014.

[82] I. C. F. Ipsen. A Note on Preconditioning Nonsymmetric Matrices. *SIAM J. Sci. Comput.*, 23(3):1050 – 1051, 2002.

[83] F. Jarre, M. Kocvara, and J. Zowe. Interior Point Methods for Mechanical Design Problems. *SIAM Journal on Optimization*, 8(4):1084 – 1107, 1998.

[84] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395, 1984.

[85] G. Karypis and V. Kumar. A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs. *SIAM J. Sci. Comput.*, 20(1):359 – 392 (electronic), 1998.

[86] C. Keller, N. I. M. Gould, and A. J. Wathen. Constraint Preconditioning for Indefinite Linear Systems. *SIAM J. Matrix Anal. Appl.*, 21(4):1300 – 1317, 2000.

[87] T. S. Kim, J. E. Kim, and Y. Y. Kim. Parallelized structural topology optimization for eigenvalue problems. *International Journal of Solids and Structures*, 41(9-10):2623 – 2641, 2004.

[88] A. Klawonn, O. B. Widlund, and M. Dryja. Dual-Primal FETI Methods for Three-Dimensional Elliptic Problems with Heterogeneous Coefficients. *SIAM J. Numer. Anal.*, 40(1):159 – 179 (electronic), 2002.

[89] R. V. Kohn and G. Strang. Optimal design and relaxation of variational problems. I. *Comm. Pure Appl. Math.*, 39(1):113 – 137, 1986.

[90] R. V. Kohn and G. Strang. Optimal design and relaxation of variational problems. II. *Comm. Pure Appl. Math.*, 39(2):139 – 182, 1986.

[91] R. V. Kohn and G. Strang. Optimal design and relaxation of variational problems. III. *Comm. Pure Appl. Math.*, 39(3):353 – 377, 1986.

[92] A. N. Kolmogorov and S. V. Fomīn. *Introductory Real Analysis.* Dover Publications, Inc., New York, 1975. Translated from the second Russian edition and edited by Richard A. Silverman, Corrected reprinting.

[93] L. Y. Kolotilina and A. Y. Yeremin. Factorized Sparse Approximate Inverse Preconditionings. I. Theory. *SIAM J. Matrix Anal. Appl.*, 14(1):45 – 58, 1993.

[94] V. G. Korneev and U. Langer. Domain Decomposition Methods and Preconditioning. *Encyclopedia of computational mechanics*, 2004.

[95] M. Kočvara, D. Loghin, and J. Turner. Constraint Interface Preconditioning for Topology Optimization Problems. *To appear in SIAM J. Sci. Statist. Comput. (SISC): Copper Mountain Special Section 2014*, Colorado, USA; 6-11 April 2014.

[96] L. Krog, A. Tucker, M. Kemp, and R. Boyd. Application of Topology, Sizing and Shape Optimization Methods to Optimal Design of Aircraft Components. In *Proc. of the 4th Altair Technology Conference, Versailles*, 2004.

[97] L. Krog, A. Tucker, M. Kemp, and R. Boyd. Topology Optimization of Aircraft Wing Box Ribs. In *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, pages 1 – 11, 2004.

[98] L. Krog, A. Tucker, and G. Rollema. Application of Topology, Sizing and Shape Optimization Methods to Optimal Design of Aircraft Components. In *3rd Altair UK HyperWorks Users Conference*, 2002.

[99] G. Kron. A Set of Principles to Interconnect the Solutions of Physical Systems. *Journal of Applied Physics*, 24(8):965 – 980, 1953.

[100] A. N. Krylov. On the numerical solution of the equation by which in technical questions frequencies of small oscillations of material systems are determined (Russian Translation). *Izvestija AN SSSR (News of Academy of Sciences of the USSR)*, 7(4):491 – 539, 1931.

[101] J. L. Lagrange. Sur la figure des colonnes. *Miscellanea Taurinensia*, 5:123 – 166, 1770.

[102] C. Lanczos. Solution of Systems of Linear Equations by Minimized-Iterations. *J. Research Nat. Bur. Standards*, 49:33 – 53, 1952.

[103] U.D. Larsen, O. Sigmund, and S. Bouwstra. Design and Fabrication of Compliant Mechanisms and Material Structures with Negative Poissons Ratio. *J. of Micro-ElectroMechanical Systems*, 6(2):99 – 106, 1997.

[104] P. D. Lax. *Functional Analysis*. Pure and Applied Mathematics (New York). Wiley-Interscience [John Wiley & Sons], New York, 2002.

[105] P. Le Tallec. Domain Decomposition Methods in Computational Mechanics. *Computational mechanics advances*, 1(2):121 – 220, 1994.

[106] V. I. Lebedev and V. I. Agoshkov. Poincaré-Steklov operators and their applications in analysis. *Academy of Sciences USSR: Moskow*, 1983.

[107] Y. Li, X. Xin, N. Kikuchi, and K. Saitou. Optimal shape and location of piezoelectric materials for topology optimization of flextensional actuators. In *Proc. of the 2001 Genetic and Evolutionary Computation Conference, GECCO-2001*, pages 1085 – 1089, 2001.

[108] Q. Q. Liang, Y. M. Xie, and G. P. Steven. Optimal selection of topologies for the minimum-weight design of continuum structures with stress constraints. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 213(8):755 – 762, 1999.

[109] J. L. Lions and E. Magenes. *Problèmes aux limites non homogènes et applications. I*, volume 3. Dunod, Paris, 1968.

[110] P. L. Lions. On the Schwarz Alternating Method I. In *First International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pages 1 – 42, 1988.

[111] P. L. Lions. On the Schwarz Alternating Method II: Stochastic Interpretation and Order Properties. In *Domain Decomposition Methods-Proceedings of the Second International Symposium on Domain Decomposition Methods for Partial Differential Equations (Los Angeles, January 14-16, 1988), SIAM, Philadelphia*, pages 47 – 71, 1988.

[112] P. L. Lions. On the Schwarz Alternating Method III: A Variant for Nonoverlapping Subdomains. In *Third international Symposium on Domain Decomposition Methods for Partial Differential Equations*, volume 6, pages 202 – 223. SIAM: Philadelphia, PA, 1990.

[113] L. Lukšan and J. Vlček. Indefinitely preconditioned inexact Newton method for large sparse equality constrained non-linear programming problems. *Numer. Linear Algebra Appl.*, 5(3):219 – 247, 1998.

[114] B. Maar and V. Schulz. Interior point multigrid methods for topology optimization. *Structural and Multidisciplinary Optimization*, 19(3):214 – 224, 2000.

[115] A. Mahdavi, R. Balaji, M. Frecker, and E. M. Mockensturm. Topology optimization of 2D continua for minimum compliance using parallel computing. *Structural and Multidisciplinary Optimization*, 32(2):121 – 132, 2006.

[116] A. Makrizi, B. Radi, and A. El Hami. Solution of the Topology Optimization Problem Based Subdomains Method. *Applied Mathematical Sciences*, 2(41):2029 – 2045, 2008.

[117] J. Mandel and R. Tezaur. On the convergence of a dual-primal substructuring method. *Numer. Math.*, 88(3):543 – 558, 2001.

[118] J. M. Martinez. A note on the theoretical convergence properties of the SIMP method. *Structural and Multidisciplinary Optimization*, 29(4):319 – 323, 2005.

[119] K. Maute and D. Frangopol. Reliability-based design of MEMS mechanisms by topology optimization. *Computers & Structures*, 81(8):813 – 824, 2003.

[120] N. G. Meyers and J. Serrin. $H = W$. *Proc. Nat. Acad. Sci. U.S.A.*, 51:1055 – 1056, 1964.

[121] A. G. M. Michell. LVIII. The limits of economy of material in frame-structures. *Philosophical Magazine Series 6*, 8(47):589 – 597, 1904.

[122] S. G. Mikhlin. Uber den Schwarzschen Algorithmus. *Dokl. Akad. Nauk SSSR, n. Ser.*, 77:569 – 571, 1951.

[123] H. P. Mlejnek. Some aspects of the genesis of structures. *Structural and Multidisciplinary Optimization*, 5(1):64 – 69, 1992.

[124] D. Morgenstern. Begründung des alternierenden Verfahrens durch Orthogonalprojektion. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, 36(7-8):255 – 256, 1956.

[125] F. Natterer. A Sobolev Space Analysis of Picture Reconstruction. *SIAM Journal on Applied Mathematics*, pages 402 – 411, 1980.

[126] F. Navarrina Martínez, I. Muíños Pantín, et al. Topology optimization of structures: a minimum weight approach with stress constraints. *Advances in Engineering Software*, 36:599 – 606, 2004.

[127] J. Nečas and I. Hlaváček. *Mathematical Theory of Elastic and Elasto-Plastic Bodies: An Introduction*, volume 3 of *Studies in Applied Mechanics.* Elsevier Scientific Publishing Co., Amsterdam-New York, 1980.

[128] J. A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7(4):308 – 313, 1965.

[129] Y. Nesterov and A. Nemirovskii. *Interior-Point Polynomial Algorithms in Convex Programming*, volume 13 of *SIAM Studies in Applied Mathematics.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1994.

[130] A. Neubauer. An a Posteriori Parameter Choice for Tikhonov Regularization in Hilbert Scales leading to Optimal Convergence Rates. *SIAM J. Numer. Anal.*, 25(6):1313–1326, 1988.

[131] R. Nevanlinna. Über das alternierende Verfahren von Schwarz. *Journal für die reine und angewandte Mathematik (Crelles Journal)*, 1939(180):121 – 128, 1939.

[132] J. Nocedal and S. J. Wright. *Numerical Optimization.* Springer Verlag, 1999.

[133] C. C. Paige and M. A. Saunders. Solution of Sparse Indefinite Systems of Linear Equations. *SIAM J. Numer. Anal.*, 12(4):617 – 629, 1975.

[134] J. Parıs, F. Navarrina, I. Colominas, and M. Casteleiro. Advances in the statement of stress constraints in structural topology optimization. In *Proceedings of the 4th International Conference on Advanced Computational Methods in Engineering ACOMEN*, 2008.

[135] J. París, F. Navarrina, I. Colominas, and M. Casteleiro. Block Aggregation of Stress Constraints in Topology Optimization of Structures. *Advances in Engineering Software*, 41(3):433 – 441, 2010.

[136] B. N. Parlett. *The Symmetric Eigenvalue Problem.* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1980. Prentice-Hall Series in Computational Mathematics.

[137] B. N. Parlett. *The Symmetric Eigenvalue Problem*, volume 20 of *Classics in Applied Mathematics.* Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1998. Corrected reprint of the 1980 original.

[138] B. N. Parlett and H. C. Chen. Use of Indefinite Pencils for Computing Damped Natural Modes. *Linear Algebra Appl.*, 140:53 – 88, 1990.

[139] J. Pebrel, C. Rey, P. Gosselet, et al. A Nonlinear Dual Domain Decomposition Method: Application to Structural Problems with Damage. *International Journal of Multiscale Computational Engineering*, 6(3):251 – 262, 2008.

[140] N. L. Pedersen. Topology optimization of laminated plates with prestress. *Computers & Structures*, 80(78):559 – 570, 2002.

[141] J. T. Pereira, E. A. Fancello, and C. S. Barcellos. Topology optimization of continuum structures with material failure constraints. *Structural and Multidisciplinary Optimization*, 26(1):50 – 66, 2004.

[142] J. S. Przemieniecki. Matrix Structural Analysis of Substructures. *Am. Inst. Aero. Astro. J*, 1:138 – 147, 1963.

[143] A. Quarteroni. *Numerical Models for Differential Problems*, volume 2. Springer Verlag Italia, Milan, 2009.

[144] A. Quarteroni and A. Valli. *Domain Decomposition Methods for Partial Differential Equations.* Oxford University Press, USA, 1999.

[145] C. Reed. Applications of Optistruct Optimization to Body in White Design. *Proceedings of Altair Engineering Event, Coventry, UK*, 2002.

[146] F. Riesz and B. Sz-Nagy. Functional Analysis. *Blackie and Son Limited*, 1956.

[147] A. Rietz. Sufficiency of a finite exponent in SIMP (power law) methods. *Structural and Multidisciplinary Optimization*, 21(2):159 – 163, 2001.

[148] M. P. Rossow and J. E. Taylor. A Finite Element Method for the Optimal Design of Variable Thickness Sheets. *AIAA Journal*, 11(11):1566 – 1569, 1973.

[149] G. I. N. Rozvany. Aims, scope, methods, history and unified terminology of computer-aided topology optimization in structural mechanics. *Structural and Multidisciplinary Optimization*, 21(2):90 – 108, 2001.

[150] Y. Saad. Krylov subspace methods for solving large unsymmetric linear systems. *Math. Comp.*, 37(155):105 – 126, 1981.

[151] Y. Saad. *Numerical Methods for Large Eigenvalue Problems.* Algorithms and Architectures for Advanced Scientific Computing. Manchester University Press, Manchester; Halsted Press [John Wiley & Sons, Inc.], New York, 1992.

[152] Y. Saad. A Flexible Inner-Outer Preconditioned GMRES Algorithm. *SIAM J. Sci. Comput.*, 14(2):461 – 469, 1993.

[153] Y. Saad. *Iterative Methods for Sparse Linear Systems.* Society for Industrial and Applied Mathematics, Philadelphia, PA, second edition, 2003.

[154] Y. Saad and M. H. Schultz. Conjugate gradient-like algorithms for solving nonsymmetric linear systems. *Math. Comp.*, 44(170):417 – 424, 1985.

[155] Y. Saad and M. H. Schultz. GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM J. Sci. Statist. Comput.*, 7(3):856 – 869, 1986.

[156] B. Sadlik. Bending a Cantilevered Beam
`http://www.capphysics.ca/PhysLab/Phys114115/exp_2%20%20loaded%20beam/content/BEAM.pdf`
Accessed 27/09/2014.

[157] W. Saleem, F. Yuqing, and W. Yunqiao. Application of Topology Optimization and Manufacturing Simulations - A new trend in design of Aircraft components. In *Proceedings of the International MultiConference of Engineers and Computer Scientists*, volume 2. Citeseer, 2008.

[158] G. Sangalli. Quasi Optimality of the SUPG Method for the One-Dimensional Advection-Diffusion Problem. *SIAM J. Numer. Anal.*, 41(4):1528–1542 (electronic), 2003.

[159] G. Sangalli. A Uniform Analysis of Nonsymmetric and Coercive Linear Operators. *SIAM J. Math. Anal.*, 36(6):2033–2048 (electronic), 2005.

[160] G. Sangalli. Robust a-posteriori estimator for advection-diffusion-reaction problems. *Mathematics of Computation*, 77(261):41, 2008.

[161] Ö. Sardan, V. Eichhorn, D. H. Petersen, S. Fatikow, O. Sigmund, and P. Boggild. Rapid prototyping of nanotube-based devices using topology-optimized microgrippers. *Nanotechnology*, 19(49):495 – 503, 2008.

[162] T. Sassi. A domain decomposition algorithm for nonlinear interface problem. In *Domain Decomposition Methods in Science and Engineering*, pages 467 – 474 (electronic). Natl. Auton. Univ. Mex., México, 2003.

[163] M. Save, W. Prager, G. Sacchi, and W. H. Warner. *Structural Optimization: Optimality Criteria*, volume 1 of *Mathematical Concepts and Methods in Science and Engineering*. Plenum Press, New York, London, 1985.

[164] S. Schmidt and V. Schulz. A 2589 Line Topology Optimization Code Written for the Graphics Card. *Computing and Visualization in Science*, 14(6):249 – 256, 2011.

[165] G. Schuhmacher, M. Stettner, R. Zotemantel, O. J. O'Leary, and M. Wagner. Optimization Assisted Structural Design of a New Military Transport Aircraft. *AIAA*, 4641(2004):E2, 2004.

[166] H. A. Schwarz. Über einige Abbildungseigenschaften. *J. Reine Angew. Numer. Math*, 70:105 – 120, 1869.

[167] O. Sigmund. Materials with prescribed constitutive parameters: An inverse homogenization problem. *International Journal of Solids and Structures*, 31(17):2313 – 2329, 1994.

[168] O. Sigmund. On the Design of Compliant Mechanisms Using Topology Optimization. *Journal of Structural Mechanics*, 25(4):493 – 524, 1997.

[169] O. Sigmund. Systematic design of microactuators using topology optimization. In *5th Annual International Symposium on Smart Structures and Materials*, pages 23 – 31. International Society for Optics and Photonics, 1998.

[170] O. Sigmund. A 99 line topology optimization code written in Matlab. *Structural and Multidisciplinary Optimization*, 21(2):120 – 127, 2001.

[171] O. Sigmund. On the usefulness of non-gradient approaches in topology optimization. *Structural and Multidisciplinary Optimization*, 43(5):589 – 596, 2011.

[172] O. Sigmund and J. Petersson. Numerical instabilities in topology optimization: a survey on procedures dealing with checkerboards, mesh-dependencies and local minima. *Structural and Multidisciplinary Optimization*, 16(1):68 – 75, 1998.

[173] O. Sigmund and S. Torquato. Design of smart composite materials using topology optimization. *Smart Materials and Structures*, 8(3):365, 1999.

[174] B. F. Smith, P. E. Bjørstad, and W. D. Gropp. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations.* Cambridge University Press, New York, 2004.

[175] S. L. Sobolev. Schwarz algorithm in the theory of elasticity. In *Dokl. Acad. Nauk., USSR*, volume 2, pages 235 – 238, 1936.

[176] M. Sofonea and A. Matei. *Variational Inequalities with Applications - A study of Antiplane Frictional Contact Problems*, volume 18 of *Advances in Mechanics and Mathematics*. Springer, New York, 2009.

[177] W. Spendley, G. R. Hext, and F. R. Himsworth. Sequential Application of Simplex Designs in Optimisation and Evolutionary Operation. *Technometrics*, 4:441 – 461, 1962.

[178] I. Stakgold and M. Holst. *Green's Functions and Boundary Value Problems.* Pure and Applied Mathematics (Hoboken). John Wiley & Sons Inc., Hoboken, NJ, Third edition, 2011.

[179] M. Stolpe and K. Svanberg. An alternative interpolation scheme for minimum compliance topology optimization. *Structural and Multidisciplinary Optimization*, 22(2):116 – 124, 2001.

[180] M. Stolpe and K. Svanberg. On the trajectories of penalization methods for topology optimization. *Structural and Multidisciplinary Optimization*, 21(2):128 – 139, 2001.

[181] K. Suzuki and N. Kikuchi. A homogenization method for shape and topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 93(3):291 – 318, 1991.

[182] K. Svanberg. The method of moving asymptotes - a new method for structural optimization. *International Journal for Numerical Methods in Engineering*, 24(2):359 – 373, 1987.

[183] K. Svanberg. A globally convergent version of MMA without linesearch. In *Proceedings of the First World Congress of Structural and Multidisciplinary Optimization*, volume 28, pages 9 – 16. Goslar, Germany, 1995.

[184] K. Svanberg. A Class of Globally Convergent Optimization Methods based on Conservative Convex Separable Approximations. *SIAM J. Optim.*, 12(2):555–573 (electronic), 2001/02.

[185] C. C. Swan and I. Kosaka. Voigt-Reuss topology optimization for structures with linear elastic material behaviours. *International Journal for Numerical Methods in Engineering*, 40(16):3033 – 3058, 1997.

[186] C. C. Swan and I. Kosaka. Voigt-Reuss topology optimization for structures with nonlinear material behaviors. *International Journal for Numerical Methods in Engineering*, 40(20):3785 – 3814, 1997.

[187] U. Tautenhahn. Error Estimates for Regularization Methods in Hilbert Scales. *SIAM J. Numer. Anal.*, 33(6):2120–2130, 1996.

[188] H. L. Tenek and I. Hagiwara. Eigenfrequency Maximization of Plates by Optimization of Topology using Homogenization and Mathematical Programming. *JSME International Journal Series*, 37(4):667 – 677, 1994.

[189] V. Torczon. On the Convergence of the Multidirectional Search Algorithm. *SIAM J. Optim.*, 1(1):123 – 145, 1991.

[190] V. Torczon. On the Convergence of Pattern Search Algorithms. *SIAM J. Optim.*, 7(1):1 – 25, 1997.

[191] A. Toselli and O. B. Widlund. *Domain Decomposition Methods – Algorithms and Theory.* Springer Verlag, 2005.

[192] L. N. Trefethen and D. Bau. *Numerical Linear Algebra.* SIAM: Society for Industrial and Applied Mathematics, 1997.

[193] J. Turner, M. Kočvara, and D. Loghin. A nonlinear domain decomposition technique for scalar elliptic PDEs. In *Proceedings of the 21st International Conference on Domain Decomposition Methods, INRIA Rennes-Bretagne-Atlantique, France*, 2013.

[194] J. Turner, M. Kočvara, and D. Loghin. Parallel Solution of the Linear Elasticity Problem with Applications in Topology Optimization. In *Proceedings of the 4th Annual BEAR PGR Conference, University of Birmingham, United Kingdom*, 2013.

[195] K. Vemaganti and W. E. Lawrence. Parallel methods for optimality criteria-based topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 194(34-35):3637 – 3667, 2005.

[196] E. Wadbro and M. Berggren. Megapixel Topology Optimization on a Graphics Processing Unit. *SIAM Rev.*, 51(4):707 – 721, 2009.

[197] S. Wang, E. de Sturler, and G. H. Paulino. Large-Scale Topology Optimization using Preconditioned Krylov Subspace Methods with Recycling. *Internat. J. Numer. Methods Engrg.*, 69(12):2441 – 2468, 2007.

[198] O. B. Widlund. The Development of Coarse Spaces for Domain Decomposition Algorithms. In *Domain Decomposition Methods in Science and Engineering XVIII*, volume 70 of *Lecture Notes in Computational Science and Engineering*, pages 241 – 248. Springer Berlin Heidelberg, 2009.

[199] J. Wloka. *Partial Differential Equations.* Cambridge University Press, Cambridge, 1987. Translated from the German by C. B. Thomas and M. J. Thomas.

[200] M. H. Wright. Ill-Conditioning and Computational Error in Interior Methods for Nonlinear Programming. *SIAM J. Optim.*, 9(1):84–111 (electronic), 1999.

[201] S. J. Wright. *Primal-Dual Interior-Point Methods*, volume 54. SIAM, 1997.

[202] J. Xu. Iterative Methods by Space Decomposition and Subspace Correction: A Unifying Approach. *SIAM review*, pages 581 – 613, 1992.

[203] J. Xu and J. Zou. Some nonoverlapping domain decomposition methods. *SIAM review*, pages 857 – 914, 1998.

[204] R. J. Yang and C. J. Chen. Stress-based topology optimization. *Structural and Multidisciplinary Optimization*, 12(2):98 – 105, 1996.

[205] L. Yin and G. K. Ananthasuresh. Topology optimization of compliant mechanisms with multiple materials using a peak function material interpolation scheme. *Structural and Multidisciplinary Optimization*, 23(1):49 – 62, 2001.

[206] D. M. Young. *Iterative Solution of Large Linear Systems.* Dover Publications, Inc., Mineola, NY, 2003. Unabridged republication of the 1971 edition [Academic Press, New York-London, MR 305568].

[207] D. M. Young and K. C. Jea. Generalized conjugate-gradient acceleration of nonsymmetrizable iterative methods. *Linear Algebra Appl.*, 34:159 – 194, 1980.

[208] M. Zhou and G. I. N. Rozvany. The COC algorithm, Part II: Topological, geometrical and generalized shape optimization. *Computer Methods in Applied Mechanics and Engineering*, 89(1-3):309 – 336, 1991.