

Topology optimisation and simultaneous analysis and design: Material penalisation and local stress constraints

by

Dirk Pieter Munro

Thesis presented in partial fulfilment of the requirements for the degree of Master of Engineering (Mechanical) in the Faculty of Engineering at Stellenbosch University



Supervisor:
Prof. Albert A. Groenwold

April 2014

DECLARATION

By submitting this thesis electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Signature:

February 22, 2014

Abstract

We investigate the simultaneous analysis and design (SAND) formulation of the topology optimisation problem. The characteristics of the formulation are presented considering the simple compliance/weight constrained problem and the more complex local stress constrained case.

The problems are solved in an efficient sparse sequential approximate optimisation (SAO) framework with the SAND formulation showing a significant reduction in computational requirements compared to the traditional and inherently expensive nested analysis and design (NAND) approach. In SAND the state equations are included in the optimisation problem as a set of equality constraints and not solved exactly in each iteration, as would be the case in NAND. Decision and state variables are thus independent, resulting in an immensely sparse optimisation problem. The availability of simple exact analytic expressions for all the constraint functions (via the finite element method) allows for the construction of accurate approximate subproblems with little computational effort. Furthermore, material can be removed completely from the design domain with few complications, resulting in a decrease in subproblem size as the algorithm progresses, further reducing computation time.

The inclusion of void material in the design domain leads to the formulation of stress constraints as so-called ‘vanishing’ constraints. Furthermore, the SAND formulation provides a new perspective on the infamous singularity problem. Amongst other results, we present some test cases that seem to scale linearly in computational requirements for a specific range of problem sizes.

Opsomming

Die formulering van die topologie optimerings probleem as 'n gelyktydige analise en ontwerp (simultaneous analysis and design (SAND)) formulering word ondersoek. Die eienskappe van die formulering word bespreek in die konteks van die eenvoudige begrensde styfheid/gewig geval en die meer komplekse plaaslike spanning begrensde geval.

Die probleme word opgelos in 'n sekwentiele benaderde optimering (SBO; sequential approximate optimisation (SAO)) raamwerk met die SAND formulering, wat lei tot 'n wesenlike vermindering in berekenings vereistes benodig in vergelyking met die tradisionele en inherente duur geneste analise en ontwerp (nested analysis and design (NAND)) geval. In SAND word die vergelykings wat die respons van die struktuur beskryf met gelykheidsbegrensings in die optimerings probleem verteenwoordig. Die respons van die struktuur word dus nie presies opgelos in elke iterasie nie, soos in die geval van NAND wel gebeur. Alle optimerings veranderlikes is dus onafhanklik en lei tot 'n baie yl optimerings probleem. Deur middel van die eindige element metode is die analitiese vorm van alle begrensings beskikbaar en kan dit gebruik word om akkurate benaderde subprobleme op te stel sonder ekstra berekenings koste. Verder kan materiaal heeltemal verwyder uit van die ontwerpgebied met weinig komplikasies. Dit lei tot 'n verkleining van subprobleme soos die algoritme vordering maak wat berekenings tyd nog meer verminder.

Die feit dat materiaal heeltemal verwyder kan word van die ontwerp gebied lei tot die formulering van spannings begrensings as sogenaamde 'verdwynende' begrensings. Verder gee die SAND formulering 'n nuwe uitsig op die bekende singulariteitsprobleem. Met verskeie ander resultate word daar ook gewys dat dit voorkom of 'n spesifieke stel toetsprobleme lineêr skaal in berekenings tyd.

Acknowledgements

First and foremost, I would like to thank my supervisor Prof. A.A. Groenwold whose technical expertise has proved invaluable over the course of my study towards a degree in Master of Engineering. He has provided me with guidance in an academic, financial and, not least, personal sense with absolutely no compromise in ‘freedom of thought’, for which I am especially grateful. The mathematician, biologist, philosopher and writer Jacob Brownowski (1908 - 1974) conveys the latter idea in his book, *The Ascent of Man*: “It is important that students bring a certain ragamuffin, barefoot irreverence to their studies; they are not here to worship what is known, but to question it.” Prof. Groenwold not only facilitates, but tirelessly encourages such an approach.

Naturally my gratitude is extended to my family, although words and phrases seem inadequate. During my studies they have been resolute in financial and moral support and, above all, I thank them for encouraging and having trust in my academic pursuits.

Finally, financial support from the NRF is acknowledged.

Contents

Abstract	iii
Opsomming	iv
Acknowledgements	v
List of Figures	xi
List of Tables	xii
1 Introduction	1
2 Structural analysis and optimisation	4
2.1 Structural analysis	4
2.1.1 The finite element method	4
2.1.2 Design parametrisation	5
2.1.3 Terminology and notation	6
2.2 Structural optimisation	7
2.2.1 Topology optimisation	7
2.2.2 The material distribution method	8
2.2.3 Material penalisation	11
2.2.4 Ground structures considered for numerical experimentation	12
2.3 The unification of analysis and design	13
2.4 Mathematical optimisation	20
2.4.1 The method of Lagrange multipliers	20
2.4.2 Conditions for optimality	21
2.4.3 Sequential approximate optimisation	22
2.4.4 Solving the approximate diagonal QP subproblems	25

<i>CONTENTS</i>	vii
3 Material penalisation in SAND	28
3.1 Background	28
3.2 Introduction	29
3.3 Problem formulation	30
3.4 Sequential approximate optimisation and simultaneous analysis and design	31
3.5 Interpretation and penalisation of decision variables	36
3.5.1 Zero lower bounds	36
3.5.2 The Hashin-Shtrikman bound and realistic microstructures	39
3.5.3 Material penalisation	40
3.5.4 Evaluation of penalisation schemes	45
3.6 The selection of standard settings	50
3.6.1 Optimal designs for the two-bar truss	51
3.6.2 Optimal designs of the MBB beam	52
3.7 Discussion and conclusion	55
4 Stress constraints, the singularity problem and SAND	56
4.1 Introduction	56
4.2 Problem formulation	57
4.3 Sequential approximate optimisation	58
4.4 The singularity problem	59
4.5 Constraint formulation	66
4.5.1 Relaxed vanishing constraints	66
4.5.2 Alternative stress relaxation: Closing down and opening up	69
4.5.3 Inconsistent vanishing constraints	76
4.5.4 Stress constraints and progressive penalisation	77
4.6 Large scale problems and the MBB beam	79
4.7 Conclusion	82
5 Conclusion	84
References	86
Appendices	
A The direct stiffness method	89
B Plane stress analysis	96

CONTENTS

viii

C Q8 elemental stiffness

99

List of Figures

2.1	Boundary value problem of elasticity.	4
2.2	Generalised optimal material distribution problem.	9
2.3	Ground structures of test problems	13
2.4	Flow diagrams depicting NAND and SAND formulated structural optimisation problems.	18
2.5	Sparsity plot of the Jacobian matrix of a local stress constrained topology optimisation problem in a SAND setting with mesh multiplier $m = 2$ and the two-bar truss ground structure.	18
2.6	Scaling of storage requirements of a local stress constrained topology optimisation problem in a SAND setting with mesh multiplier m and the two-bar truss ground structure.	19
3.1	A comparison of the topologies generated for the weight minimization of the two-bar truss using various approximation strategies.	33
3.2	Norm of diagonal Hessian terms $(\bar{Q}_q^{\{k\}})$ related to state variables in each iteration for various second-order approximations.	34
3.3	Norm of diagonal Hessian terms $(\bar{Q}_x^{\{k\}})$ related to decision variables in each iteration for various second-order approximations.	35
3.4	A comparison of topologies generated with zero and non-zero lower bounds on decision variables.	37
3.5	CPU time per iteration	38
3.6	Number of decision variables active on lower bound per iteration	38
3.7	Simple isotropic material with penalisation (SIMP)	40
3.8	A comparison of simple isotropic material with penalisation with $p = 2$ and 3 and the Hashin-Shtrikman bound for a Poisson ratio of $\frac{1}{3}$	41
3.9	A comparison of SINH penalisation with $p = 2$ and 3 and the Hashin-Shtrikman bound for a Poisson ratio of $\frac{1}{3}$	42
3.10	A comparison of functions values as a function of x for various penalisation functions.	45

LIST OF FIGURES

x

3.11	A comparison of the topologies generated for the weight minimization of the two-bar truss using various material penalisation functions.	45
3.12	A comparison of gradient values as a function of x for various penalisation functions.	47
3.13	A comparison of curvature magnitudes as a function of x for various penalisation functions.	47
3.14	A comparison of the topologies generated for the weight minimization of the two-bar truss using various levels of progressive penalisation.	49
3.15	The generated topology for the weight minimization of the two-bar truss for $p = 3$ when ‘rounded’.	49
3.16	Convergence plot of $\phi_{B\&W}$ and p for the weight minimisation of the two-bar truss ground structure subject to a constraint on compliance.	49
3.17	Topologies generated by weight minimisation of the two-bar truss ground structure subject to a constraint on compliance.	51
3.18	Topologies generated by compliance minimisation of the two-bar truss ground structure subject to a constraint on volume.	52
3.19	Topologies generated by weight minimisation of the MBB beam ground structure subject to a constraint on compliance.	53
3.20	Topologies generated by compliance minimisation of the MBB beam structure subject to a constraint on volume.	54
4.1	A 1-D structure illustrating the singularity problem.	60
4.2	The feasible region defined by a stress constraint on element 1.	61
4.3	The feasible region defined by the relaxed constraint on element 1.	63
4.4	A representation of the feasible domain in a SAND setting.	64
4.5	The feasible region defined by a relaxed vanishing stress constraint in a SAND setting.	65
4.6	The feasible region defined by an inconsistent vanishing stress constraint in a SAND setting.	65
4.7	The generated topologies considering relaxed vanishing constraints (a) and the simple compliance constrained case (b).	68
4.8	A depiction of active stress constraints and elements on the upper bound for the generated topology with relaxed vanishing constraints.	68
4.9	A depiction of active stress constraints and solid/intermediate elements for the generated topology with relaxed vanishing constraints.	69
4.10	The generated topologies with stress constraints formulated as relaxed vanishing constraints with $\epsilon_0 = 0.2$ and 0.75	69
4.11	The effect of relaxation as a function of decision variable x_i and θ	71

LIST OF FIGURES

xi

4.12	Topologies generated for the weight minimization of the two-bar truss with alternative stress relaxation and a range of ϵ_0	72
4.13	A depiction of active stress constraints and elements at solid/intermediate states with the 'closing down' continuation strategy.	73
4.14	Convergence plot of $\phi_{B\&W}$ and ϵ with the 'closing down' continuation strategy.	74
4.15	Topologies generated for the weight minimization of the two-bar truss with alternative stress relaxation and a range of ϵ^*	74
4.16	Convergence plot of $\phi_{B\&W}$ and ϵ with the 'opening up' continuation strategy.	75
4.17	The generated topology with alternative stress relaxation and ϵ_0 fixed at 1×10^{-2}	75
4.18	A depiction of active stress constraints and elements at solid/intermediate states with the 'opening up' continuation strategy.	76
4.19	Topologies generated with inconsistent vanishing constraints	76
4.20	Topologies generated with inconsistent vanishing constraints and progressive penalisation	78
4.21	Topologies generated with alternative stress relaxation and progressive penalisation	78
4.22	Topologies generated by weight minimisation of the two-bar truss structure subject to local stress constraints with the 'closing-down' continuation strategy.	81
4.23	Topologies generated for weight minimisation of the two-bar truss ground structure with $m = 14$ subject to local stress constraints with the 'closing down' and 'opening up' continuation strategies.	81
4.24	Topologies generated by weight minimisation of the MBB structure with $m = 4$ subject to local stress constraints using the 'closing-down' and 'opening up' continuation strategies on relaxation.	82
A.1	1-D example of a structure to introduce the FE method	89
A.2	Bar element	90
A.3	Finite element discretisation	93
A.4	Q8 element	94
A.5	Plane stress	94
B.1	Differential isotropic element	96

List of Tables

3.1	A comparison of the results obtained for the weight minimization of the two-bar truss using various approximation strategies.	33
3.2	Comparison of solution quality and computation time for zero and non-zero lower bounds.	37
3.3	A comparison of the results obtained for the weight minimization of the two-bar truss using various penalisation functions.	46
3.4	A comparison of the results obtained for the weight minimization of the two-bar truss with various levels of progressive penalisation.	48
4.1	A comparison of the results obtained for the weight minimization of the two-bar truss with alternative stress relaxation and a range of ϵ_0	72
4.2	A comparison of the results obtained for the weight minimization of the two-bar truss with alternative stress relaxation and a range of ϵ^*	75
4.3	A comparison of the results obtained for the weight minimization of the two-bar truss with inconsistent vanishing constraints and two values for ζ	77
4.4	A comparison of the results obtained for the weight minimization of the two-bar truss with inconsistent vanishing constraints and progressive penalisation.	78
4.5	A comparison of the results obtained for the weight minimization of the two-bar truss with alternative stress relaxation and progressive penalisation.	79
4.6	Scaling of problem size and computation time considering the weight minimisation of the two-bar truss ground structure subject to local stress constraints and the ‘closing down’ continuation strategy.	80

Chapter 1

Introduction

The eminent Swiss mathematician Leonard Euler (1707 - 1783) once said:

...nothing at all takes place in the universe in which some rule of maximum or minimum does not appear¹.

With *maximisation* and *minimisation* both principles of *optimisation* we might paraphrase Euler's statement to mean "all processes in the universe can be described as an optimisation problem". This statement, although seemingly far-fetched, refers to the law of *conservation of energy*. To emphasise the universality of this concept we wish to quote Richard P. Feynman (1919 - 1988), winner of the Nobel Prize in Physics (1965):

There is a fact, or if you wish, *a law*, governing all natural phenomena that are known to date. There is no known exception to this law - it is exact as far as we know. The law is called the *conservation of energy*. It states that there is a certain quantity, which we call energy, that does not change in the manifold changes which nature undergoes. That is a most abstract idea, because it is a mathematical principle; it says that there is a numerical quantity which does not change when something happens. It is not a description of a mechanism, or anything concrete; it is just a strange fact that we can calculate some number and when we finish watching nature go through her tricks and calculate the number again, it is the same².

Therefore, these systems ("all natural phenomena") can be described by the *stability* of some quantity as a function of the properties of the system. Mathematically different kinds of stability are described by *maxima* or *minima* on the *energy* function of the system. That is, principles of optimisation. Optimisation is also found in other sciences such as economics and predominantly, as is the context of this thesis, engineering. Specifically the design of engineering systems, where the optimal set of choices to design a system is sought. Such a design is evaluated by some cost function related to all the possible choices the designer can make. The minima (and maxima) of

¹As quoted in *The Anthropic Cosmological Principle* (1986) by John D. Barrow and Frank J. Tipler, p. 150.

²Taken from *Six Easy Pieces: Essentials of physics explained by its most brilliant teacher* (2011) by Richard P. Feynmann, p. 69.

this cost function is defined by *stationary* points, or points of *stability*. In this thesis we consider problems of structural optimisation, and specifically topology optimisation. That is the optimal *shape* or *geometry* of things.

The statement by Euler quoted above might be generalised even further if we include “... *and a set of partial differential equations*”. A wide range of (if not all) physical phenomena can be described by a set of partial differential equations (PDE’s); examples include fluid flow, heat transfer, quantum mechanics, sound propagation, electric phenomena and elasticity. Problems of structural engineering (and design) are governed by the *boundary value problem of elasticity*, a set of PDE’s that describe the interaction of forces and material in a physical domain.

Topology optimisation is concerned with finding the optimal distribution or lay-out of material in the aforementioned domain. Problems of this nature are typically of high dimensionality and practical solutions can only be obtained by computational means. Typically the numerical optimisation and analysis phases are considered distinct in a computational sense. For a given design the set of PDE’s that describe the responses (or state) of the structure is solved exactly, hence the optimisation problem is one in the *design* space of the problem, distinct from the *state* space. This procedure is performed in an iterative fashion until some optima is converged to, hence referred to as *nested analysis and design* (NAND). In contrast, and the primary concern of this thesis, is the unification of *analysis* and *design* in the topology optimisation problem. That is, the set of PDE’s, simplified to a set of algebraic equations via the finite element method, is included in the optimisation problem as a set of constraints. Hence the optimisation of the structure occurs in *design* as well as *state* space or *simultaneous analysis and design* (SAND).

This thesis is by no means complete with respect to the unification of analysis and design in topology optimisation, for we only explore some important aspects of the problem; material penalisation and local stress constraints. Chapter 2 should be considered a study of the relevant literature. Structural analysis and optimisation is discussed before moving onto the unification of these procedures in SAND. Furthermore, the reasons leading to material penalisation, related to the severe computational requirements of a discrete problem, is discussed. Finally sequential approximate optimisation (SAO) and other techniques of mathematical programming, employed to solve the optimisation problem, are reviewed. In Chapter 3 various kinds of material penalisation are explored and evaluated in the context of SAND. Material penalisation enters the SAND problem directly via the set of constraints that make up the state equation. Therefore the construction of the approximate Hessian in each SAO subproblem and eventual convergence to the solution of the state equation is intimately linked to the penalisation scheme. Furthermore, because the state equation is not solved *per se* a value of exactly zero is allowed on the lower bound of material variables, typically not possible in NAND. Thereafter, in Chapter 4 the inclusion of local stress constraints is investigated. This is an infamous problem in the topology optimisation community, for not only is the problem inherently expensive in computational resources, but stress constraints suffer from the well-documented *singularity* problem. Stress constraints can be considered a specific type of *local* constraint and many characteristics of the said constraints in the SAND problem, especially concerning computational aspects, can be generalised to all *local* constraints. In other words, phenomena that occur on a local scale (each and every spatial position) in some physical domain. Chapter 5 concludes.

The reader should note that the nature of the work that compose this thesis is such that some core concepts of the formulation are linked to various characteristics. To promote ease of readability, chapters are written to be independent to a large extent, with concepts often repeated in the context of the characteristics investigated in the specific chapter.

Chapter 2

Structural analysis and optimisation

2.1 Structural analysis

2.1.1 The finite element method

The problem of determining an optimal structural design can be solved in any number of ways. Although a wide variety of procedures exist, each method requires, in some way or another, the relationship between design, applied loads, boundary conditions and structural responses to be established. Calculating responses such as deformations, displacements, stresses and strains for a particular design is referred to as the act of *structural analysis*. Furthermore, structural sensitivities, required by gradient based optimisation methods, capture changes in responses relative to a change in design. To analyse a continuum structure we seek the solution of a set of elliptic partial differential equations (PDE's) over an elastic domain with applied loads and boundary conditions, such as depicted in Figure 2.1, also referred to as the *boundary value problem of elasticity*. We consider the problem of structural optimisation via the discretised material distribution method [1], with the finite element (FE) method considered the predominant approach to analyse a continuum structure.

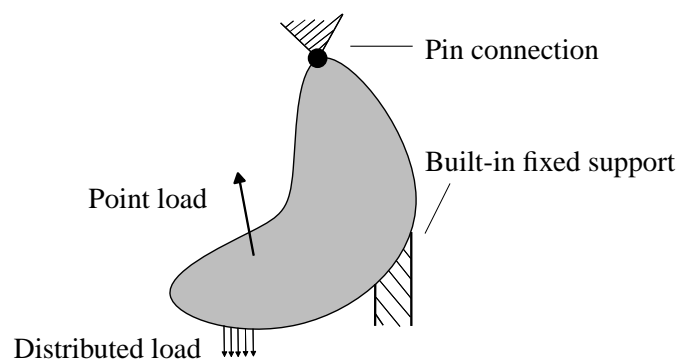


Figure 2.1: Boundary value problem of elasticity.

The FE method is a general numerical procedure to obtain an approximate solution to a boundary-value problem (BVP). A BVP is the problem of finding a specific solution to a partial differential equation which satisfies some conditions on the boundary of the domain. This is a problem that

arises in many branches of physics, such as the modelling of waves, electrostatics and potential theory.

The FE method relies on the principle of *discretization*; a domain in which *cause* and *effect* is governed by a set of PDE's is discretized into a number of finite elements. The solution of the original continuous problem, which is functional and yields the desired unknown quantities at infinitely many points in the domain, is usually not obtainable. Therefore *effects*, such as displacements, are approximated locally and the design domain discretized to reduce the system of PDE's to a system of algebraic equations. These are readily solved by a wide variety of methods.

Alexander Hrennikoff was the first to solve plane elasticity problems using a discretized lattice framework in 1941. However, the method of *discretization* dates to antiquity. The Greek mathematician Archimedes, who referred to it as the method of *exhaustion*, calculated the area under a curve by approximating the curve with many simple figures (figures he could calculate the area of). In fact, Archimedes is credited with being the first to calculate the value of π to reasonable accuracy using this method in around 250 BC. He did this by approximating the circle with regular polygons, if the number of sides of each polygon are increased the curve (and the value of π) is approximated ever more accurately. More than 2 centuries later, in 1956, Turner, Clough, Martin and Topp published a paper entitled "Stiffness and Deflection of Complex Structures" in which the term "finite-element method" is introduced. Although the FE method sets out to solve a substantially more complex problem than calculating the area bounded by a curve the concepts of *discretization* and *exhaustion* are consistent. That is, the FE method finds an discretized approximate solution to the original continuum problem and ever more accurate solutions can be obtained with finer discretizations, although this leads to an increase in problem size and computational effort.

The *direct stiffness method* is the most popular implementation of the FE method considering structural BVP's. In Appendix A the direct stiffness method is illustrated with a simple 1-D structure, followed by a description of the higher-order Q8 2-D finite elements considered throughout this thesis. Here we simply define the resulting system of equations (the *state equation*) employed to calculate structural responses. In other words *structural analysis* is performed by solving the system of equations

$$\mathbf{K}\mathbf{q} = \mathbf{f}, \quad (2.1)$$

where \mathbf{K} is the global stiffness matrix constructed from elemental stiffness matrices \mathbf{K}_e , \mathbf{f} is the vector of nodal forces and \mathbf{q} is the vector of nodal displacements.

2.1.2 Design parametrisation

Structural optimisation requires that the design of a structure be parametrised such that a mathematical expression describes the lay-out of the structure. Assume that the set of variables \mathbf{x} define the structure (in some sense) and that this is facilitated by parametrising global stiffness through

$$\mathbf{K}(\mathbf{x})\mathbf{q} = \mathbf{f}. \quad (2.2)$$

Note that whatever the interpretation of \mathbf{x} physically, for example material thickness h or modulus of elasticity E , design parametrisation enters the problem via global stiffness \mathbf{K} as a function of \mathbf{x} .

Furthermore, assume this relation is facilitated through the elemental stiffness matrix by a variable related to each element $x_e \in \mathbf{x}$ and a monotonically increasing function $\Pi(x_e)$ ¹ on $x_e \in [\tilde{x}, \hat{x}]$ (where \hat{x} and \tilde{x} denote upper - and lower bounds respectively). Constant per element, $\Pi(x_e)$ is factored out of the integral in equation (A.23), resulting in

$$\mathbf{K}_e(x_e) = \Pi(x_e) \int_{-1}^1 \int_{-1}^1 \mathbf{B}^T \mathbf{D} \mathbf{B} |\mathbf{J}| h ds dt \quad (2.3)$$

Grouping common terms and performing the integration we arrive at the elemental stiffness expression considered throughout this thesis

$$\mathbf{K}_e(x_e) = \frac{Eh}{1-\nu^2} \Pi(x_e) \bar{\mathbf{K}}_e, \quad (2.4)$$

with coefficients $\bar{\mathbf{K}}_e$ given in Appendix C. Typically stiffness coefficients of \mathbf{K}_e are calculated with some numerical integration scheme, such as Gauss's method. However, quadratic displacement and linear shape functions allow for analytic calculation of stiffness coefficients. Therefore initially the storage of only a 16×16 symmetric matrix is required. This matrix is subsequently employed as building block to construct the global stiffness matrix of an arbitrary sized problem. The repetitive nature of this procedure involving a large number of coordinate mappings and matrix multiplications is very well suited to a computer aided implementation.

2.1.3 Terminology and notation

To be clear, throughout this thesis we refer to equation (2.2) as the *equation of state* (or *state equation*). Solution of the state equation yields structural responses, that is, the state of the system. The vector of structural responses or nodal displacements or *state* variables is denoted by \mathbf{q} throughout. Any procedure that sets out to obtain the solution of the state equation will be referred to as a procedure of *structural analysis*.

To avoid ambiguities the vector \mathbf{x} (which defines the design of the structure) is referred to as *decision* variables. Optimisation problems considered in this work are all a function of decision and state variables, for example,

$$\begin{aligned} & \underset{\mathbf{x}, \mathbf{q}}{\text{minimize}} && f_0(\mathbf{x}, \mathbf{q}) \\ & \text{subject to} && f_j(\mathbf{x}, \mathbf{q}) \leq 0, \quad j = 1, 2, \dots, m \\ & && f_j(\mathbf{x}, \mathbf{q}) = 0, \quad j = m + 1, 2, \dots, m + p, \end{aligned}$$

with f_0 some objective function and f_j constraints (m and p the number inequality and equality constraints respectively). In this context the term *design* variables refers to decision and state variables; hence the set of *design* variables are defined as $\mathbf{x} \cup \mathbf{q}$.

¹Not to be confused with the symbol typically used to denote potential energy.

2.2 Structural optimisation

2.2.1 Topology optimisation

In 1638, four years before his death, Galileo Galilei published his final work entitled *Discorsi (Discourses and Mathematical Demonstrations Relating to Two New Sciences)* in which he summarised much of his work in physics over the preceding 30 years. As a result Galileo is often referred to as the “father of modern physics”, for the new sciences were *kinematics* and *strength of materials* [2]. From this point on now famous mathematicians such as Johan Bernoulli, Leonard Euler and Joseph-Louis Lagrange studied and solved direct and indirect structural analysis problems. Direct problems are those in which the design of a structure is known and structural responses (displacements, strains and stresses) are calculated and indirect problems are those in which structural responses are known and some geometrical aspects of the design are determined. Structural optimisation is a special kind of indirect problem where an optimal structure is sought given a set of boundary conditions and applied loads.

After the beginning of the 19th century direct problems received much attention while the inherently more complex indirect problems were avoided. It was only after the 1950's, due to advances made in the field of aerospace engineering (and specifically control theory) that structural optimisation became one of the most intensively studied branches of solid mechanics. Initially 1-D variational problems where the optimal shape of a beam is calculated were readily solved. Continuum problems in 2-D and 3-D proved to be much more complicated; solutions often not obtainable due to the high dimensionality of the problem. However these problems could be discretized and solved with structural analysis performed as per the FE method. In practice, approximating the responses of a structure to reasonable accuracy leads to an inherently large scale procedure in terms of computational requirements. Such problems are indeed solved, but the optimisation procedure, in conjunction with structural analysis, requires above average levels of computational resources for even modestly sized problems. Thus restricting the utility of structural optimisation in industry. We consider such 2-D problems in continuum space, discretized with the FE method, referred to as *topology optimisation via the material distribution method*.

The first reference to the concept of an *optimal topology* can be found in the paper “Generating optimal topologies in structural design using a homogenization method” by Bendsøe and Kikuchi [3]. Topology optimisation is the term commonly used to refer to the optimisation of a wide range of material distribution problems². However, strictly speaking, topology optimisation specifically refers to the optimisation of both the *material properties* and the *connectivity* of the domain [4]. To elaborate, a structural optimisation problem can be grouped into one of three categories:

- The *sizing* problem in which the objective is to find the optimal material properties of a domain. For example the size distribution of trusses or elastic plates composing a structure. Thus design variables define quantities such as thickness/area/elasticity of a member/plate/bar with a predetermined position. Importantly, elements are not removed entirely from the structure hence the connectivity is not subject to optimisation. This is also referred to as *free material optimisation* [5][6].

²And sometimes specifically the *minimum compliance* problem. Although we do not adhere to this connotation.

- *Shape* optimisation where the optimal shape of the domain is sought. The design variables in this case could for example define spline functions which in turn define the geometry of the domain. Therefore, in a sense only the connectivity of the structure is optimised. See [7] for an example of a shape optimisation problem.
- And finally the *topology* optimisation problem, where the optimal *lay-out* of material is sought in a domain. Topology optimisation requires the least amount of *a priori* knowledge compared to the aforementioned structural optimisation problems; the material properties and the connectivity of the domain are optimised. Topology optimisation sets out to determine if material should be present or absent at every spatial position in the design domain with no other states allowed. In this context the term *lay-out* or *topology* refers to both size *and* shape. As such topology or lay-out optimisation is considered more general than size and shape optimisation and capable of addressing all aspects of structural optimisation simultaneously [8].

2.2.2 The material distribution method

Considered the predominant approach to determine the optimal lay-out or topology of a continuum structure [8], the only known quantities being the applied loads and possible support locations. In general, let $\boldsymbol{x}(\boldsymbol{r})$ denote the presence or absence of material at spatial position \boldsymbol{r} . Given some structural objective $f_0(\boldsymbol{x})$ and constraints $f_j(\boldsymbol{x})$ the discrete programming problem (2.5) is aimed at finding the optimal distribution of material in the predefined design space Ω ,

$$\begin{aligned} & \min_{\boldsymbol{x}} f_0(\boldsymbol{x}) \\ & \text{subject to } f_j(\boldsymbol{x}) \leq 0 \quad j = 1, 2, \dots, m, \\ & \text{with } \boldsymbol{x}(\boldsymbol{r}) \in \{0, 1\} \quad \forall \boldsymbol{r} \in \Omega, \end{aligned} \quad (2.5)$$

where m is the number of constraints. Furthermore boundary conditions and applied loads are defined in terms of Ω . Note that the physical interpretation of discrete decision variables in problem (2.5) is completely arbitrary. Any interpretation is justifiable (and therefore unnecessary) as long as the presence or absence of material is reflected in the appropriate equations. That is, as long as stiffness is zero if no material is present and that of the original material otherwise. What is more, these problems are inherently large scale from a computational point of view. That is, in principle, material at each and every spatial position is variable leading to a large number of design variables and constraints.

We consider the simple *minimum compliance* problem subject to a single resource constraint as natural starting point in defining the general topology optimisation problem. The continuum formulation in structural analysis is reviewed, as proposed in [3], before reverting to the discretized problem via the FE method, the sole concern of this thesis. Consider a mechanical body Ω^m as part of a larger reference frame $\Omega \in \mathbb{R}^2$ as depicted in Figure 2.2. Analogous to terminology used in truss topology and sizing optimisation Ω is often referred to as the *ground structure*³.

³The set of all possible designs from which the optimal design is selected, or the so-called *structural universe* [9].

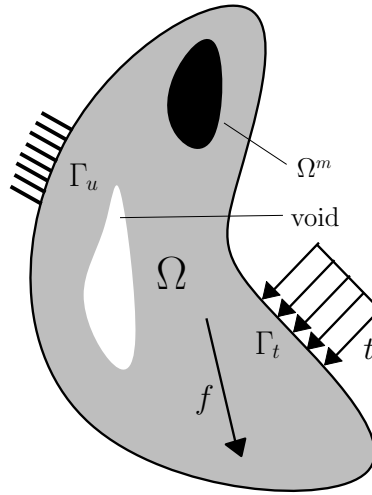


Figure 2.2: Generalised optimal material distribution problem.

Let the work done on the structure by external forces and boundary tractions be denoted by

$$l(\mathbf{u}) = \int_{\Omega} \mathbf{f} \mathbf{u} \, d\Omega + \int_{\Gamma_T} \mathbf{t} \mathbf{u} \, ds, \quad (2.6)$$

where \mathbf{u} is the field of displacements, \mathbf{f} is the body forces, \mathbf{t} is the surface tractions applied to the boundary (Γ_t) and ds a differential distance along the boundary.

Furthermore let the internal virtual work of the elastic body at equilibrium, for displacement field \mathbf{u} and virtual displacement field \mathbf{v} , be denoted by

$$a(\mathbf{u}, \mathbf{v}) = \int_{\Omega} C_{ijkl}(\mathbf{r}) \epsilon_{ij}(\mathbf{u}) \epsilon_{kl}(\mathbf{v}) \, d\Omega, \quad (2.7)$$

with linearised strains

$$\epsilon_{ij}(\mathbf{u}) = \frac{1}{2} \left(\frac{\partial u_i}{\partial r_j} + \frac{\partial v_j}{\partial r_i} \right). \quad (2.8)$$

This problem can be viewed as finding the optimal choice of stiffness tensor $C_{ijkl}(\mathbf{r})$ to minimise the compliance⁴ of the structure. Structural analysis is performed by requiring that internal virtual work be equal to work imparted to the structure by external forces. Therefore

$$a(\mathbf{u}, \mathbf{v}) = l(\mathbf{v}) \quad \text{with } \mathbf{v} \in \mathbf{U}, \quad (2.9)$$

where \mathbf{U} is the space of all kinematically admissible displacement fields. To minimise the compliance of the structure we wish to find \mathbf{u} and $C(\mathbf{r})$ for which

$$\begin{aligned} & \min_{C, \mathbf{u} \in \mathbf{U}} l(\mathbf{u}) \\ & \text{subject to } a(\mathbf{u}, \mathbf{v}) = l(\mathbf{v}) \quad \forall \mathbf{v} \in \mathbf{U}, \\ & C(\mathbf{r}) \in C_{\text{ad}} \quad \forall \mathbf{r} \in \Omega, \end{aligned} \quad (2.10)$$

⁴Equivalent to maximisation of stiffness.

with C_{ad} the set of all admissible stiffness tensors. The desired optimal topologies are solid-void designs; material is either present or absent at every spatial position with no other states allowed. The admissible set of stiffness tensors is therefore composed of the stiffness tensor of void material (zero) and that of the original solid isotropic material. The presence or absence of material at spatial position \mathbf{r} is expressed through a binary material distribution function $x(\mathbf{r}) \in \{0, 1\} \forall \mathbf{r} \in \Omega$. In turn the material compliance tensor is viewed as a function of \mathbf{r} via x ,

$$C(x(\mathbf{r})) = C(1) = C_0 \quad \forall \mathbf{r} \in \Omega^{\text{mat}} \quad (2.11)$$

$$C(x(\mathbf{r})) = C(0) = 0 \quad \forall \mathbf{r} \notin \Omega^{\text{mat}}. \quad (2.12)$$

Typically the amount of material is constrained by

$$\int_{\Omega^{\text{mat}}} 1 d\Omega = \int_{\Omega} x(\mathbf{r}) d\Omega \leq \bar{v}, \quad (2.13)$$

where \bar{v} is a limit on the total volume of the structure.

Apparently the continuum problem lacks solutions. As explained in [10] the compliance of the structure can be improved *ad infinitum* while keeping volume constant. This is possible with finer and finer distributions of the same amount of material. However, the non-existence problem can be addressed with the homogenization approach [3]. Briefly, material with a microstructure is introduced in the continuum formulation. The simplest formulation involves a microstructure that consists of solid isotropic material and void. The amount of solid to void material, hence the stiffness or density of the said microstructure is parametrised by decision variables. The macro material is anisotropic and allows for the incorporation of composite material properties in the topology optimisation problem. Importantly this approach allows for physical interpretation of decision variables that obtain non-binary values.

The homogenization approach resolves the non-existence problem by expanding the design space to include anisotropic materials. Other methods involve restricting the design space with some minimum length scale to address the non-existence issue, as explained in [10]. Discretization of the design domain and the introduction of a FE mesh introduces such a minimum length scale. However, decreasing the size of elements allows finer structural elements to become feasible, referred to as the problem of mesh dependency [11].

The primary reason for discretizing the structural domain in problem (2.10) is to allow for solution by computational/numerical means. As mentioned in Section 2.1 the typical approach is to consider a domain discretized as per the FE method. With regards to problem (2.10), according to [8] we seek the optimal composition of two fields, C and \mathbf{u} . Assuming a common finite element mesh for both fields with C_e constant with respect to a single element e , we arrive at the discretized minimum compliance problem

$$\begin{aligned} \min_{C_e, \mathbf{q}} \quad & \mathbf{f}^T \mathbf{q} \\ \text{subject to} \quad & \mathbf{K}(C_e) \mathbf{q} = \mathbf{f} \quad \forall C_e \in C_{\text{ad}}, \end{aligned} \quad (2.14)$$

with $e = 1, \dots, n$ and n the number of elements in the mesh. The continuous displacement field \mathbf{u} is replaced with the vector of nodal displacements \mathbf{q} and \mathbf{f} denotes nodal forces. It is here where

the state equation enters the optimisation problem. Including binary decision variables, problem (2.14) is rewritten as

$$\begin{aligned} \min_{x_e, \mathbf{q}} \quad & \mathbf{f}^T \mathbf{q} \\ \text{subject to} \quad & \mathbf{K}(x_e) \mathbf{q} = \mathbf{f} \quad \forall x_e \in \{0, 1\}, \end{aligned} \quad (2.15)$$

where stiffness coefficients are parametrised according to equation (2.3) with $\Pi(x_e)$ a step function from 0 to 1. In discretized form the resource constraint reduces to

$$\sum_{i=1}^{n_e} x_e = \bar{v}. \quad (2.16)$$

Here we introduce the concept of computational complexity and the Landau symbol \mathcal{O} . In computer science the efficiency of an algorithm is measured in *asymptotic complexity*. This notion captures an algorithm's scalability as problem size grows. For example, an algorithm with $\mathcal{O}(n)$ performance would require about twice as much time to run if problem size (n) is doubled. An algorithm with $\mathcal{O}(n^2)$ performance would take four times as long.

Let function $T(n)$ denote computational complexity; the time (or number of iterations) required by an algorithm to solve a problem as a function of problem size. Then writing, with function $t(n)$ some 'simple' expression in n (typically a polynomial),

$$T(n) = \mathcal{O}(t(n)) \quad (2.17)$$

implies there exists some constant c such that

$$T(n) \leq ct(n) \quad \text{for } n \rightarrow \infty. \quad (2.18)$$

The discrete topology optimisation problem solved via the material distribution method is known to be non-deterministic polynomial-time complete (NP-complete or NPC). Therefore, although a solution to the problem can be verified quickly, there exists no reliable way to find the solution in the first place. Simply put there exists no algorithm which can reliably find a solution bounded by (with constant a denoting the polynomial order)

$$T(n) = \mathcal{O}(n^a), \quad (2.19)$$

hence the time required to solve the problem increases severely with an increase in problem size. Relaxing discrete requirements on decision variables to $0 \leq x_e \leq 1$ the now continuous problem in decision variables can be solved by continuous gradient based non-linear programming methods. These methods allow solution in polynomial time, therefore relation (2.19) holds, albeit typically for values of $a \gg 1$. That is, the time required to solve the problem typically grows exponentially with problem size.

2.2.3 Material penalisation

The discrete requirement on decision variables is relaxed from a computational standpoint. Therefore, considering the topology optimisation problem, arriving at a discrete solution remains of

primary concern. Notwithstanding the homogenization approach [3], how should material at intermediate values ($0 < x_e < 1$) be interpreted? This is of no concern if a discrete solution to the continuous problem is obtained. For decision variables that acquire binary values, i.e. $x_e \in \{0, 1\}$, simply denotes the presence or absence of material (solid or void). Achieving primarily black-and-white solutions is often facilitated by penalising *grey* material (material at intermediate values), the solid isotropic material with penalisation (SIMP) approach [12] being most popular. Examples of alternative penalisation schemes are presented in [13] and [14] and investigated in Chapter 3. All these schemes rely on degrading the stiffness of an element at intermediate values, hopefully forcing the solution to solid-void designs. In this setting elements at intermediate values add relatively little to the stiffness of the structure and are thus encouraged to obtain values of zero (volume is typically minimised or constrained) or one if the corresponding material is required for the design to be feasible. Penalisation can also be incorporated in other constraints that are a function of decision variables with a number of interpretations, if any. It should be kept in mind that any penalisation increases the complexity of the optimisation problem. That is, penalisation is always a trade-off between sufficiently driving the solution to black-and-white designs and an increase in computational complexity.

We chose the popular SIMP method to introduce the notion of material penalisation. Relaxed (continuous) decision variables are simply replaced with a polynomial function of the said variable in the state equation. Referring to equation (2.3), the SIMP model is introduced by setting

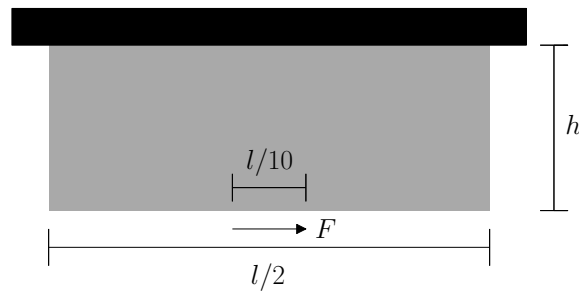
$$\Pi(x_e) = x_e^p \text{ for } p > 1, \quad (2.20)$$

with $e = 1, \dots, n$ (n denoting the number of elements in the mesh) and penalisation typically set at a fixed value of $p = 3$. Many authors comment on continuation strategies on the penalisation parameter to avoid convergence to a local minimum and drive solutions to solid-void designs. More on this in Chapter 3.

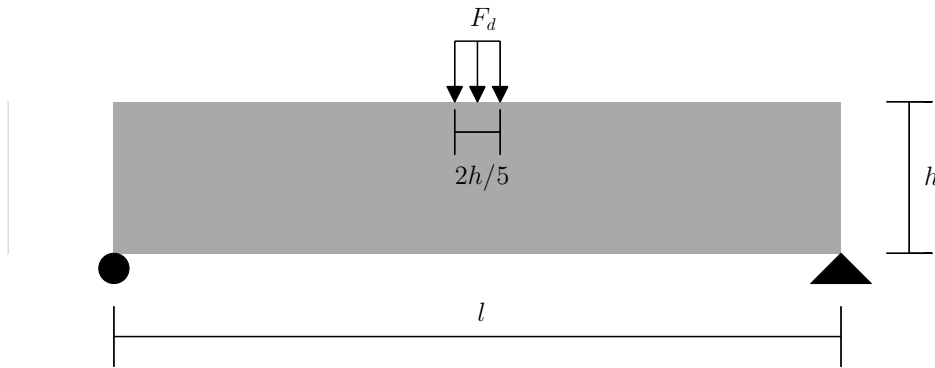
Finally, as mentioned before, the homogenization approach [3] is considered a successful attempt at interpreting grey material. A realistic composite material model can be constructed that mimics the SIMP interpolation model for specific values of p . For 2-D problems with Poisson ratio of $1/3$ it is required that $p \geq 3$ for the interpretation to hold. A more detailed investigation of material penalisation in a simultaneous analysis and design (SAND) setting is conducted in Chapter 3.

2.2.4 Ground structures considered for numerical experimentation

With the notion of a *ground structure* introduced in this section we chose this point to define the ground structures considered for numerical experiments. We consider two well known ground structures referred to as the 2-bar truss and MBB beam, depicted in Figure 2.3. In both instances applied force (F or F_d), global length scale (l), maximum thickness (h_0), yield stress (S_y), Young's modulus (E) and Poisson's ratio (ν) is indicated. The design domain is discretized with Q8 elements with width $15m$ and height $5m$ in the case of the two-bar truss and $30m$ and $5m$ in the case of the MBB beam, with m denoting a mesh multiplier. Due to symmetry, thus reducing problem size, only half the design domain is modelled when considering the MBB beam ground structure.



(a) The two-bar truss (unit thickness; plane stress).



(b) The MBB beam (unit thickness; plane stress).

Figure 2.3: Ground structures of test problems ($F = 6$ N, $F_d = 1$ N, $l = 6$ m, $h = 1$ m, $S_y = 20$ N/m², $E = 1$ N/m², $\nu = 0.3$).

2.3 The unification of analysis and design

Traditionally the optimisation of a structure is viewed as the unification of two distinct numerical or algorithmic procedures. We refer to these procedures as *structural analysis* and *numerical optimisation*⁵. One can think of this distinction as arising intuitively from the nature of the structural optimisation problem, and structural design in general. A designer (a *numerical optimisation* algorithm in our case) determines the geometric lay-out of material in some domain whilst ensuring the structure is fit for purpose. Therefore the material lay-out (along with other material properties) is described by *input* variables. These variables the designer can control, hence also the name *control* or *decision* variables. Furthermore, topology can refer to both the material properties and lay-out of material; the solution of the topology optimisation problem defines the optimal distribution of infinitesimal amounts of material, a description of both the macro and micro properties of the structure. Therefore, for some loading condition the response of the structure is exclusively a function of its topology. That is, from the viewpoint of the designer, who can only control the topology of the structure, a strict causal relationship exists between the design of the structure and its response. This relationship is defined by the concept or act of *structural analysis*. This view of structural optimisation, with a designer in explicit control of the topology and implicit control of structural responses, could be thought of as nested analysis and design (NAND). The NAND approach is an

⁵In keeping with this notion we have reviewed these topics as distinct concepts in the sections leading up to this discussion.

iterative procedure that requires alternate phases of structural analysis and numerical optimisation. That is, the equation of state is solved exactly in each iteration to obtain structural responses for a given design. In this thesis we explore an alternative to the NAND approach, namely simultaneous analysis and design (SAND), where the equation of state, in principle, need only be satisfied once the optimal solution is obtained. First we review the NAND formulation, after which SAND is introduced. The interested reader is referred to [15] for an extensive review and comparison of both formulations.

Consider the relaxed and penalised topology optimisation problem with structural analysis discretized as per the FE method in the form of minimum compliance subject to a resource constraint

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{q}} \quad & \mathbf{f}^T \mathbf{q} \\ \text{subject to} \quad & \mathbf{K}(\Pi(x_i)) \mathbf{q} = \mathbf{f} \\ & \sum_{e=1}^n x_e \leq \bar{v} \\ & 0 \leq x_i \leq 1 \quad i = 1, 2, \dots, n . \end{aligned} \tag{2.21}$$

Up until this point the state equation and state variables have been defined vaguely as ‘part of the optimisation problem’. We have not explicitly referred to the state equation as a constraint (or set of constraints) to ease the conceptual link to the traditional NAND formulation, even though it is clear that structural analysis enters the fundamental continuum minimum compliance problem (2.10) as a constraint on decision and state variables.

To be clear, Problem (2.21) is a optimisation problem in both decision and state variables. In contrast, NAND implies state variables \mathbf{q} are defined as an implicit function of decision variables \mathbf{x} , that is $\mathbf{q}(\mathbf{x})$. The state equation and state variables are eliminated from the optimisation problem with a procedure equivalent to direct substitution. NAND is a *nested* algorithm, with structural analysis performed for a given constant \mathbf{x} followed by an optimisation phase in \mathbf{x} only. This procedure is repeated in each iteration. Therefore, considering the optimisation problem, decision and state variables are no longer independent, but related implicitly by the state equation. We can depict this relationship as

$$\mathbf{q}(\mathbf{x}) = \mathbf{K}(\mathbf{x})^{-1} \mathbf{f} . \tag{2.22}$$

Importantly, the stiffness matrix $\mathbf{K}(\mathbf{x})$ becomes singular and responses can not be obtained if stiffness coefficients are allowed to attain values of exactly zero. Therefore it could be argued that structural optimisation problems solved in a NAND setting are not strictly speaking topology optimisation problems. Keep in mind that topology optimisation is concerned with optimising both the connectivity and material properties of the domain. For non-zero lower bounds on stiffness, thus material can not attain a state of void, one can say the connectivity of the domain is not subject to optimisation; the problem is actually one of sizing or free material optimisation [4].

The state equation is linear in state variables and typically non-linear in decision variables. However, in NAND decision variables are constant in the analysis phase, hence the equation of state reduces to a system of linear equations. Techniques from linear algebra could in principle be employed to solve the system, but an explicit functional form of \mathbf{q} in terms of \mathbf{x} is usually not obtainable. Therefore iterative numerical techniques are used to solve the typically large system of equations. Classical techniques such as Gaussian elimination, LU decomposition and Cholesky

factorisation all require about $\mathcal{O}(n^3)$ operations (with n the number of unknowns). Special classes of linear systems called symmetric diagonally dominant systems can be solved quite efficiently; an algorithm that solves these systems in $\mathcal{O}(n(\log(n))^2) \approx \mathcal{O}(n^{1.5})$ operations is proposed in [16] (the fastest algorithm of this kind we are aware of).

In addition to the structural responses gradient based optimisation methods require first-order sensitivities of all constraint functions. Consider a general constraint $f(\mathbf{x}, \mathbf{q})$. The first-derivative of $f(\mathbf{x}, \mathbf{q})$ to decision variables \mathbf{x} yields

$$\left. \frac{df(\mathbf{x}, \mathbf{q}(\mathbf{x}))}{d\mathbf{x}} \right|_{n \times 1} = \left. \frac{\partial f(\mathbf{x}, \mathbf{q}(\mathbf{x}))}{\partial \mathbf{x}} \right|_{n \times 1} + \left. \frac{d\mathbf{q}(\mathbf{x})}{d\mathbf{x}} \right|_{n \times u} \left. \frac{\partial f(\mathbf{x}, \mathbf{q}(\mathbf{x}))}{\partial \mathbf{q}} \right|_{u \times 1} \quad (2.23)$$

with n the number of decision variables and u the number of generalised displacements (or state variables). The partial derivatives in equation (2.23) present no particular difficulty. However, the calculation of the sensitivity derivatives $\frac{d\mathbf{q}}{d\mathbf{x}}$ require further analysis. Taking the total derivative of the state equation with respect to decision variables \mathbf{x} we obtain

$$\frac{\partial \mathbf{K}}{\partial \mathbf{x}} \mathbf{q} + \frac{d\mathbf{q}}{d\mathbf{x}} \mathbf{K} = 0 \quad (2.24)$$

therefore $\frac{d\mathbf{q}}{d\mathbf{x}}$ is the solution $\mathbf{Z}|_{u \times n}$ of the system

$$\mathbf{K} \mathbf{Z} = \mathbf{S} \quad (2.25)$$

where

$$\mathbf{S}|_{u \times n} = -\frac{\partial \mathbf{K}}{\partial \mathbf{x}} \mathbf{q}. \quad (2.26)$$

Equation (2.25) looks deceptively similar to the state equation. However, unknown \mathbf{Z} is not a vector but a $u \times n$ matrix. Therefore the solution of equation (2.25) has to be obtained by solving a number of linear systems.

The procedure to perform sensitivity analysis outlined above is called the *direct differentiation method*. An alternate approach called the *adjoint variable method* is more efficient under certain circumstances, especially considering local constraints (such as stress constraints). With the direct method c additional constraints would require the solution of c additional linear systems. Local stress constraints are typically applied to each element in the FE mesh, that is $c = n$, computationally a very expensive procedure. However, considering the adjoint variable method the number of constraints that enter the analysis phase can be reduced (and computation time decreased) with the use of an active set strategy.

Here we wish to quantify the computational complexity related to the structural analysis phase of a NAND formulated structural optimisation problem with local constraints. For the purposes of this discussion a *local* constraint applied to element i is defined as

$$g_i(x_i, \mathbf{q}_{elmi}) \leq 0, \quad (2.27)$$

a function of the *local* decision variable x_i and a subset of the state variable vector $\mathbf{q}_{elmi} \subset \mathbf{q}$. Where \mathbf{q}_{elmi} is the collection of nodal displacements related to element i .

Assume a linear system solver is available which requires an unknown order of operations with computational complexity (and effort) denoted by

$$SLS(n) = \mathcal{O}(n^s). \quad (2.28)$$

Therefore $SLS(n)$ operations are required, bounded by an unknown exponent s , to solve a linear system with n unknowns (where n is the number of state variables). Furthermore, assume sensitivity derivatives are obtained with the *adjoint variable method* and that an active set strategy is employed whereby the amount of constraints that enter the problem are $m = \alpha n$ with $0 < \alpha \leq 1$. Hence, solving equation (2.22) requires $SLS(n)$ operations, and in addition, solving equation (2.25) would require

$$T(n) = (\alpha)(n)SLS(n) \quad (2.29)$$

operations. Therefore, to calculate structural responses and sensitivity derivatives of a locally constrained problem in a NAND setting would require $(1 + \alpha n)SLS(n)$ operations. Considering asymptotic complexity (neglecting all but the highest order terms) and equation (2.28) computational requirements of the structural analysis phase in a NAND setting is

$$T_{NAND}(n) = (\alpha)(n)\mathcal{O}(n^s), \quad (2.30)$$

which reduces to

$$T_{NAND}(n) = \mathcal{O}(n^{s+1}). \quad (2.31)$$

From equation (2.31) it is clear that structural analysis in a NAND setting scales exponentially in computational requirements. Assuming the very efficient linear solver mentioned above could be employed ($s = 1.5$), the computational complexity of the structural analysis phase alone would scale to $\mathcal{O}(n^{2.5})$. This is certainly an idealised scenario and in practice solving a general linear system would require $s \approx 2$. That is, computational complexity closer to $\mathcal{O}(n^3)$ could be expected. This result is disastrous if very large problems are considered. For example, say a problem comprising of 100 unknowns has a run-time of 100 seconds, merely doubling the amount of unknowns would result in a run-time of about 800 seconds. This effect is amplified as the amount of unknowns are increased, resulting in a run-time of 100 000 seconds for a problem of 1000 unknowns (only 10 times larger than the original problem, but run time has increased by a factor of 1000).

Keep in mind that the computational complexity denoted in equation (2.31) only involves the structural analysis phase. Considering the optimisation phase computational complexity depends to a greater extent on the sparsity of the problem. Fleury considers problems solved with sequential quadratic programming or a dual approach and reports that solution times scale to $\mathcal{O}(n^3)$ [17] (where n is the number of decision variables). However, Fleury considers a dense algorithm and albeit much harder to generalise than the analysis phase, numerical experiments presented in [10] using a sparse dual algorithm confirm solution times scale roughly to $\mathcal{O}(n^3)$.

Note that the overall computational complexity in NAND will be the computational complexity of which ever procedure (analysis or optimisation) dominates⁶. Not only is it not possible to know

⁶Note that the procedures of structural analysis and numerical optimisation scale according to the number of state variables and decision variables respectively. However, the number of state variables and number of decision variables are related by a constant factor. Therefore, in terms of asymptotic computational complexity, it is arbitrary whether n is thought of as number of state or decision variables.

beforehand which will be the case for a general problem, but one procedure might dominate up until a specific problem size after which, for larger problems, the complexity of the other procedure takes over. Therefore, in line with the discussion above, we deduce that the overall computational complexity of solving a locally constrained structural optimisation problem in NAND is roughly $\mathcal{O}(n^3)$.

To summarise, the most popular approach to structural optimisation has been one where a clear distinction exists between structural analysis and numerical optimization phases. In this view only decision variables are treated as design variables. All other responses (or state variables) such as displacements, stresses and strains are implicit functions of decision variables. Hence state and decision variables are not independent but related by the state equation. This results in the need for expensive computational procedures to calculate sensitivity derivatives, especially considering local constraints. Alternatively, in SAND both decision (\boldsymbol{x}) and state variables (\boldsymbol{q}) are defined as design variables. Therefore the SAND formulation is more closely related to the fundamental problem (2.10) and an exact representation of the discretized problem (2.15). Note that per definition all design variables of an optimisation problem are independent. In our case the relation of \boldsymbol{x} to \boldsymbol{q} , the state equation, is imposed on the problem as a set of equality constraints, hence structural analysis and numerical optimisation is performed *simultaneously*. A flow diagram comparing NAND and SAND formulations is depicted in Figure 2.4.

Compared to NAND the SAND optimisation problem is much more complex. For both \boldsymbol{x} and \boldsymbol{q} defined as independent design variables the state equation forms a set of u non-linear equality constraints rendering the problem non-convex (with u the total degrees of freedom or nodal displacements). Therefore, in basic form, convergence to the global optimum can not be guaranteed. What's more, the problem typically suffers from numerical scaling issues, especially so considering structural optimisation problems where decision and state variables typically vary by orders of magnitude.

We have discussed the nature of computational scaling inherent to the locally constrained structural optimisation problem solved in a NAND setting. We showed calculation of sensitivity derivatives $\frac{dq}{dx}$, required because \boldsymbol{x} and \boldsymbol{q} are dependent causes scaling in computational complexity to roughly $\mathcal{O}(n^3)$. In contrast the SAND formulation requires decision and state variables to be independent. Therefore $\frac{dq}{dx} = 0$ and the expensive calculation of the constraint Jacobian reduces to the calculation of partial derivatives. What's more, simple analytical expressions of the constraint functions are available via the FE method (reducing storage requirements and simplifying the calculation of sensitivities). Further still, the constraint Jacobian is typically extremely sparse in a SAND setting since local constraints are per definition dependent on a small number of variables compared to overall problem size.

To elaborate, the storage and computational requirements of a gradient based optimisation algorithm is primarily that of the constraint Jacobian matrix. That is the number of non-zero first derivatives of all constraint functions to all design variables. In Figure 2.5 a sparsity plot of the Jacobian matrix of the local stress constrained topology optimisation problem in a SAND setting with mesh multiplier $m = 2$ and the two-bar truss ground structure is given. As is clear, the problem is very sparse, for this case a sparsity of 98.7% can be reported. In Figure 2.6 the number of non-zero elements in the constraint Jacobian is given as a function of total design variables (and mesh multiplier m given above each data point). Importantly the storage requirements of the lo-

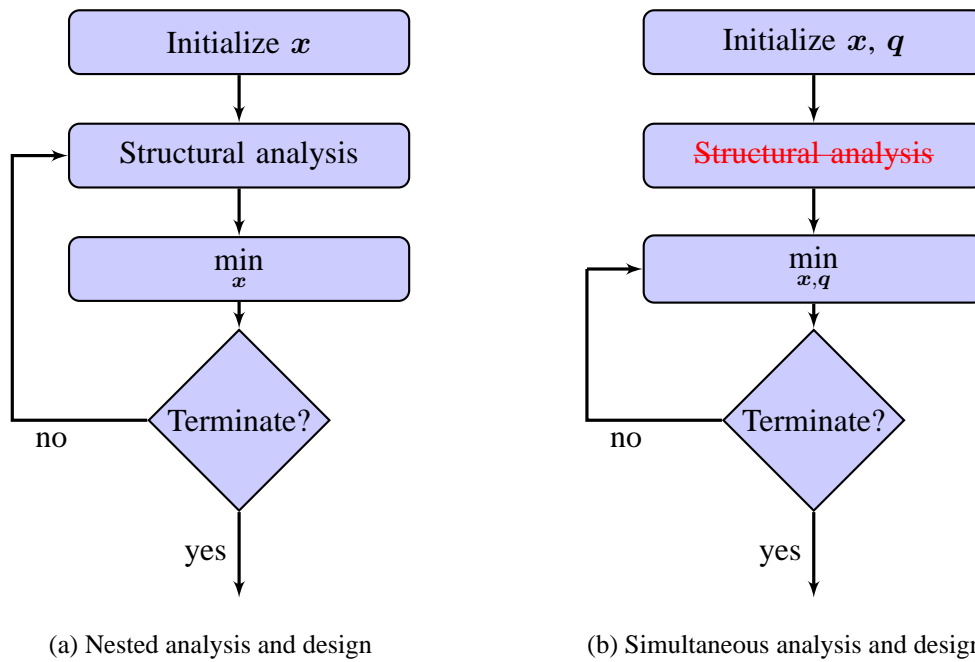


Figure 2.4: Flow diagrams depicting NAND and SAND formulated structural optimisation problems.

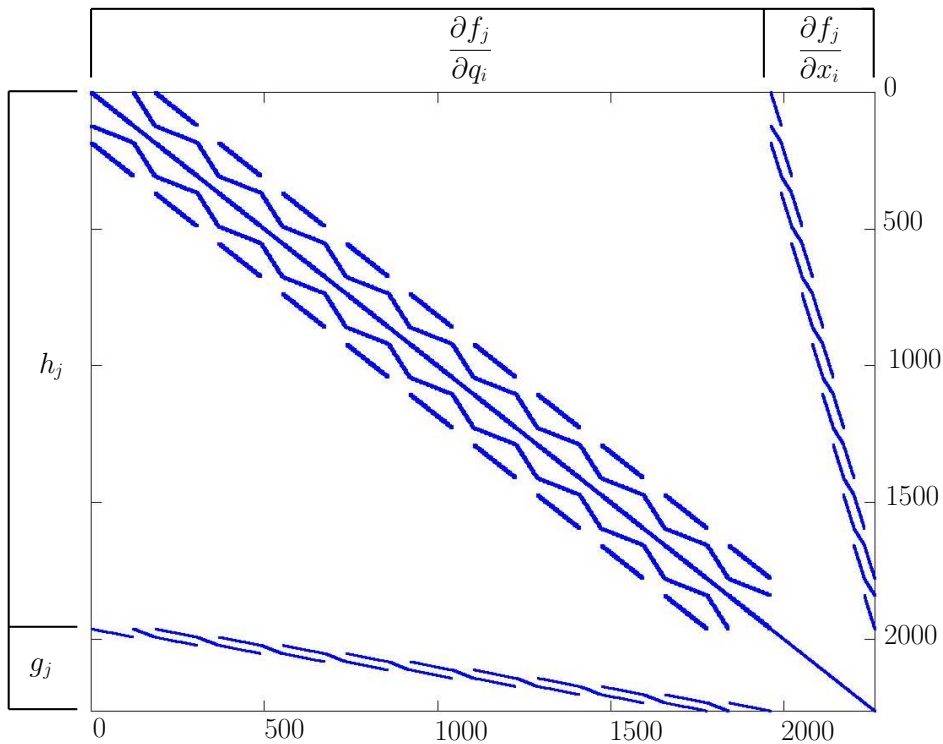


Figure 2.5: Sparsity plot of the Jacobian matrix of a local stress constrained topology optimisation problem in a SAND setting with mesh multiplier $m = 2$ and the two-bar truss ground structure.

cally constrained SAND problem scales to $\mathcal{O}(n)$. Although the numerical values displayed here is for a stress constrained problem with Q8 elements, the analysis would hold for any *local* constraint and discretization in a SAND setting.

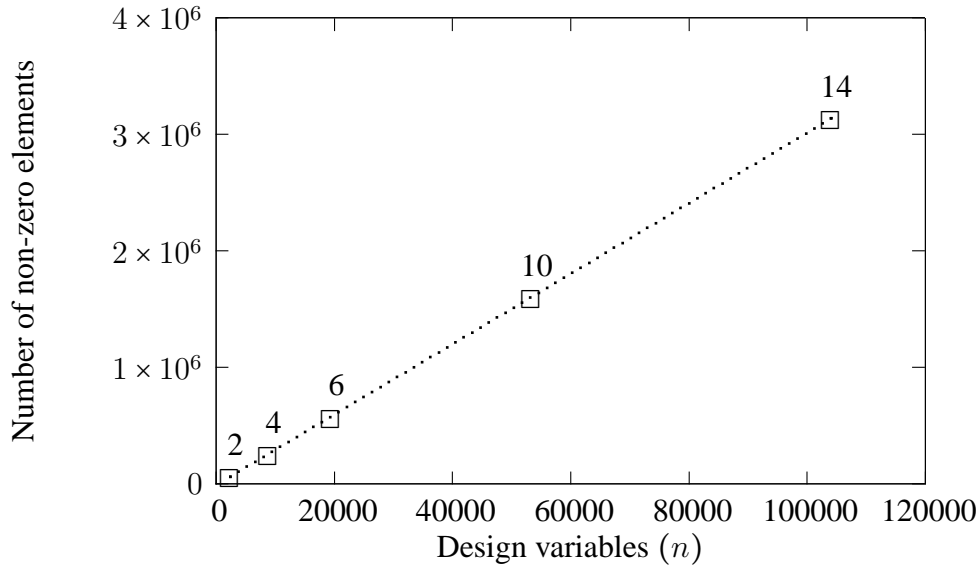


Figure 2.6: Scaling of storage requirements of a local stress constrained topology optimisation problem in a SAND setting with mesh multiplier m and the two-bar truss ground structure.

Naturally linear scaling of storage requirements is the best possible scenario. In comparison, although not quantified exactly, the equivalent NAND formulation of the problem would scale much worse. Considering only the procedure of structural analysis and the calculation of sensitivity derivatives for a locally constrained problem the NAND formulation will scale at least to $\mathcal{O}(n^3)$. That is assuming the storage requirements of the FE system, a $n \times n$ matrix, scales to $\mathcal{O}(n^2)$. The addition of n constraints would require the solution of n additional FE systems, resulting in an exponential increase of storage requirements as problem size grows. In contrast the SAND formulation is much better suited for solution by a sparse implementation of a gradient based optimisation algorithm.

As mentioned before, the NAND formulation does not allow the connectivity of the domain to be optimised (because stiffness coefficients may not obtain values of exactly zero). The latter is a result of the fact that the state equation is solved exactly, imposing on the optimisation problem an implicit reciprocal relation to decision variables. However, in SAND the state equation is not solved *per se* and global stiffness coefficients are allowed to obtain values of exactly zero. Therefore a zero lower bound on global stiffness coefficients pose no particular difficulty (except for possible numerical scaling issues). To be clear, a zero lower bound on stiffness coefficients is fundamental in optimising the topology of the structure; the connectivity and material properties of the domain. The complete removal of material from the design domain also results in successively smaller subproblems, reducing computational requirements.

Interestingly the SAND formulation sheds new light on the well documented stress singularity problem [18]. A zero lower bound on decision variables allow local constraints to be formulated

as vanishing constraints. Not only is this a more exact representation of the physical problem, but the removal of constraints that no longer contribute to the feasibility of the design reduces computational requirements. In addition structural responses (such as stress measures) are typically only a function of state variables. In a NAND setting an implicit reciprocal relation would be imposed on such responses. In SAND, with state variables independent design variables, these responses remain explicit functions of state variables. More on this in Chapter 4.

In conclusion we note that solving the state equation iteratively based on first-order sensitivity information is loosely equivalent to Newton's method for solving non-linear systems. Therefore, although the SAND optimisation problem is much more complex, solving the state equation iteratively based on gradient information (such as encountered in sequential approximate optimisation (SAO)) seems to be an appropriate solution procedure. Along with the characteristics mentioned above, such as computational requirements related to local constraints, optimising the connectivity of the domain and a suitable formulation of local stress constraints, we believe the SAND formulation to be very well suited to structural optimisation. Specifically large scale topology optimisation problems with local stress constraints.

2.4 Mathematical optimisation

Mathematical optimisation (or mathematical programming) is a procedure that aims to find the optimal or 'best' solution to some problem. In the design of engineering systems the problem is typically one of minimising effort or resources, or maximising benefit or efficiency, or both. A designer of such a system is tasked with the decisions that achieve either of these goals. However, the efficiency desired or resources required of/by an engineering system, that is a physical system, can always be translated to a mathematical function of these decisions. Once the system is described in this way, as a function of decision variables, a variety of mathematical techniques can be used to obtain the set of decision variables that define the optimal design, hence the optimal set of decisions.

Historically mathematical optimisation can be traced back to the early period of the Second World War. Military personnel were faced with the problem of allocating limited and often scarce resources to a wide variety of activities and operations with some desired outcome. For instance, how should a limited number of aircraft be deployed to attack some targets, defend others, transport goods and protect those transports from the enemy? The problem was presented to mathematicians and led to the development of linear programming, generally studied as part of a field known as operations research. These developments are believed to be instrumental in the British victory in the air battle over Europe [19].

2.4.1 The method of Lagrange multipliers

The design of an engineering system will more often than not be *constrained* in some way, e.g. the maximisation of the efficiency of a system with a limited amount of available resources. This limit on resources is represented by a constraint on the design. Therefore the optimal set of decision variables are confined to some region in the design space, referred to as the *feasible* region. To

find the set of decision variables that simply maximises or minimises some unconstrained function is relatively straight forward. First-order sensitivity information is used to find a stationary point with respect to these variables (the necessary conditions) and second-order curvature information describe the nature of the stationary point (the sufficient conditions). However, how can the optimal set of decision variables be determined if we now seek a stationary point of a function, whilst simultaneously satisfying a set of constraints?

The method of Lagrange multipliers, introduced by Joseph Louis Lagrange in 1788, was originally a reformulation of classical mechanics to determine the state or response of a constrained dynamic system. The method has found its way into mathematical optimisation due to the equivalence of a constrained optimisation problem and a constrained dynamic system. The problem is solved by seeking the minimum of the sum of potential and kinetic energy (referred to as the Lagrangian of the system). To introduce the concept consider the non-linear optimisation problem subject to equality and inequality constraints

$$\begin{aligned} \min_{\mathbf{y}} \quad & f_0(\mathbf{y}) \\ \text{subject to} \quad & f_j(\mathbf{y}) \leq 0, \quad j = 1, 2, \dots, m \\ & f_j(\mathbf{y}) = 0, \quad j = m + 1, m + 2, \dots, m + p \end{aligned} \quad (2.32)$$

with $f_0(\mathbf{y})$ the objective function and $f_j(\mathbf{y})$, $j = 1, 2, \dots, m$ and $f_j(\mathbf{y})$, $j = m + 1, m + 2, \dots, m + p$ the inequality and equality constraint functions respectively. Furthermore all functions are dependent on n real design variables with $\mathbf{y} \in \mathbb{R}^n$. The Lagrangian, L , is defined as a function of design variables \mathbf{y} and Lagrange multipliers $\boldsymbol{\lambda}$

$$L(\mathbf{y}, \boldsymbol{\lambda}, \boldsymbol{\mu}) = f_0(\mathbf{y}) + \sum_{j=1}^m f_j(\mathbf{y})\lambda_j + \sum_{j=m+1}^{m+p} f_j(\mathbf{y})\lambda_j \quad \text{with } \lambda_j \geq 0 \quad \text{for } j = 1, 2, \dots, m. \quad (2.33)$$

The solution of the optimisation problem is then

$$f_0(\mathbf{y}^*) = \min_{\mathbf{y}} (\max_{\boldsymbol{\lambda}} L(\mathbf{y}, \boldsymbol{\lambda})), \quad (2.34)$$

that is, the Lagrangian is the encoding of the constrained optimisation problem into a single function, and by minimising with respect to design variables, and maximising over Lagrange multipliers, the constrained stationary point is obtained. However now we are stuck with a minimisation and maximisation problem. In other words, how is the solution of equation (2.34) determined? The next topic introduces the necessary conditions for optimality, a set of equations that yield a solution to equation (2.34).

2.4.2 Conditions for optimality

Consider a general real valued function $f(x)$, a stationary point of which is obtained by calculating $\frac{df(x)}{dx} = 0$. However, such a point might be a maximum, minimum or even a saddle point. Therefore the requirement that first-order sensitivities be zero is referred to as the necessary condition for some point to be a maximum or minimum. Similarly the Karush-Kuhn-Tucker (KKT) conditions are the first-order necessary conditions for a point of a multidimensional constrained optimisation

problem to be a minimum or maximum. In other words a stationary point of function (2.34). First published in 1951, the KKT conditions for minimisation of a general optimisation problem is defined as

$$\begin{aligned} \frac{\partial f_0(\mathbf{y}^*)}{\partial y_i} + \sum_{j=1}^m \frac{\partial f_j(\mathbf{y}^*)}{\partial y_i} \lambda_j + \sum_{j=m+1}^{m+p} \frac{\partial f_j(\mathbf{y}^*)}{\partial y_i} \lambda_j &= 0, \quad i = 1, 2, \dots, n \\ f_j(\mathbf{y}^*) &\leq 0, \quad j = 1, 2, \dots, m \\ \lambda_j &\geq 0, \quad j = 1, 2, \dots, m \\ \lambda_j f_j(\mathbf{y}^*) &= 0, \quad j = 1, 2, \dots, m \\ f_j(\mathbf{y}^*) &= 0, \quad j = m + 1, m + 2, \dots, m + p \end{aligned} \quad (2.35)$$

A solution of the set of equations (2.35) yields a set of design variables that define a stationary point of the Lagrangian, hence the first-order necessary conditions for a solution to the nonlinear optimisation problem. However, a stationary point can only satisfy the KKT conditions if some regularity conditions are satisfied, also referred to as *constraint qualification*. A number of constraint qualifications can be employed depending on the nature of the functions that define the optimisation problem. In general it is required that the gradient vectors of all constraint functions be linearly independent. Ensuring, in other words, the system of equations (the first KKT condition)

$$\frac{\partial f_0(\mathbf{y}^*)}{\partial y_i} + \sum_{j=1}^m \frac{\partial f_j(\mathbf{y}^*)}{\partial y_i} \lambda_j + \sum_{j=m+1}^{m+p} \frac{\partial f_j(\mathbf{y}^*)}{\partial y_i} \lambda_j = 0, \quad i = 1, 2, \dots, n \quad (2.36)$$

has an unique solution. However, as noted in [19], verifying constraint qualification without knowing \mathbf{y}^* beforehand is a difficult procedure. In the light of this it can be shown that if the functions that define the optimisation problem satisfies some conditions constraint qualification will always hold. Referred to as a *convex* optimisation problem, constraint qualification always holds if all inequality and equality constraints are linear or, all equalities are linear, inequality constraints are convex, and one solution vector exists that lies completely in the feasible domain. What is more, if the optimisation problem is convex, the KKT conditions are necessary and sufficient for a stationary point to be optimal. This property of convex optimisation problems is exploited when the nonlinear optimisation problem is deconstructed to a sequence of convex problems, referred to as sequential convex programming (SCP), of which sequential approximate optimisation (SAO) is a derivative. Refer to [20] for a detailed discussion on convex optimisation.

2.4.3 Sequential approximate optimisation

Newton's method

To introduce the procedure of sequential approximate optimisation (SAO) consider Newtons method to solve a set of nonlinear equations. Also known as the Newton-Raphson method, first published in the 17th century and still the predominant numerical procedure to solve a set of nonlinear equations today. Consider the real valued one dimensional function $f(x)$, to find a stationary point x^* of $f(x)$ we require the solution of the equation

$$\frac{df(x^*)}{dx} = 0, \quad (2.37)$$

assuming x^* can not be obtained by analytical means we employ an iterative numerical procedure. Furthermore, assume only first- and second-order derivative information is available. Consider a quadratic approximation to the function $f(x)$, that is, a Taylor series expansion around some point $x^{\{k\}}$:

$$\bar{f}(x) = f(x^{\{k\}}) + \frac{df(x^{\{k\}})}{dx}(x - x^{\{k\}}) + \frac{1}{2} \frac{d^2f(x^{\{k\}})}{dx^2}(x - x^{\{k\}})^2. \quad (2.38)$$

The first derivative of equation (2.38), neglecting higher order terms, is

$$\frac{d\bar{f}(x)}{dx} = \frac{df(x^{\{k\}})}{dx} + \frac{d^2f(x^{\{k\}})}{dx^2}(x - x^{\{k\}}), \quad (2.39)$$

hence equation (2.39) is an approximation to the first derivative of function $f(x)$. Then, when set to zero, equation (2.39) is effectively an approximation to the stationary point of function $f(x)$. Therefore equation (2.39) can be rearranged to obtain an expression for successively improved (hopefully) approximations to the stationary point

$$x^{\{k+1\}} = x^{\{k\}} - \frac{\frac{df(x^{\{k\}})}{dx}}{\frac{d^2f(x^{\{k\}})}{dx^2}}. \quad (2.40)$$

The method can be extended quite easily to a set of multidimensional nonlinear equations. What is more, under some conditions Newton's method shows quadratic convergence, a very powerful property, for the error in approximating the stationary point reduces quadratically in each iteration [19]. Furthermore, studying Newton's method serves as a neat introduction to SAO; it can be shown that sequential quadratic programming (SQP), of which SAO is a derivative, is equivalent to applying Newton's method to the set of KKT conditions for an equality constrained problem [19].

SAO with function approximations based on incomplete series expansions: SAOi

SAO is considered the predominant approach to solve a wide range of nonlinear optimisation problems. In SAO the local behaviour of the problem in some point is approximated by a "simpler" surrogate optimisation problem. The solution of the surrogate problem (or subproblem) denotes the starting point of the next iteration. Our solution procedure is a form of sequential convex programming (SCP), therefore each approximate subproblem is convex, which leads to a couple of useful characteristics. Firstly, the minimum of each subproblem can be obtained with standard mathematical programming techniques and is guaranteed to be the minimum of the subproblem. Secondly, as discussed before, constraint qualification holds automatically for each subproblem.

In SQP the exact Hessian of the Lagrangian of the global problem is employed in each iteration. Naturally this requires the global problem to be convex. However, we consider nonlinear optimisation problems where this is not necessarily the case. Therefore we employ SAOi where function approximations are based on incomplete Taylor series expansions. This leads to the construction of an approximate diagonal Hessian of the Lagrangian in each iteration, with convexity enforced in a variety of ways. See [21] for more details.

Consider the inequality and equality constrained nonlinear optimisation problem \mathbb{P}_{NLP} , representative of the structural optimisation problems considered herein

$$\begin{aligned} \min_{\mathbf{y}} \quad & f_0(\mathbf{y}) \\ \text{subject to} \quad & f_j(\mathbf{y}) \leq 0, \quad j = 1, 2, \dots, m \\ & f_j(\mathbf{y}) = 0, \quad j = m + 1, m + 2, \dots, m + p, \end{aligned} \quad (2.41)$$

with $f_0(\mathbf{y})$ the objective function and $f_j(\mathbf{y})$, $j = 1, 2, \dots, m$ and $f_j(\mathbf{y})$, $j = m + 1, m + 2, \dots, m + p$ the inequality and equality constraint functions respectively. All functions are dependent on n real design variables with $\mathbf{y} \in \mathbb{R}^n$. All functions are assumed to be twice continuously differentiable but not convex. That is, for a given starting point $\mathbf{y}^{\{0\}}$ we seek a local minimiser $\mathbf{y}^{\{*\}}$ of problem \mathbb{P}_{NLP} .

In sequential approximate optimisation (SAO) successive subproblems $\mathbb{P}[k]$, $k = 1, 2, 3, \dots$ are constructed to approximate the local behaviour of problem \mathbb{P}_{NLP} at some point $\mathbf{y}^{\{k^*\}}$. The solution of the subproblem $\mathbb{P}[k]$, denoted by $\mathbf{y}^{\{k^*\}}$, forms the starting point $\mathbf{y}^{\{k+1\}}$ of the next iteration and subproblem $\mathbb{P}[k + 1]$.

In SAOi the approximate subproblems are based on quadratic incomplete Taylor series expansions. Approximations $\tilde{f}_j(\mathbf{y})$, $j = 1, 2, 3, \dots, m + p$ are constructed to the objective and all constraints functions such that

$$\tilde{f}_j(\mathbf{y}) = f_j^{\{k\}} + \sum_{i=1}^n \left(\frac{\partial f_j}{\partial y_i} \right)^{\{k\}} (y_i - y_i^{\{k\}}) + \frac{1}{2} \sum_{i=1}^n c_{2i_j}^{\{k\}} (y_i - y_i^{\{k\}})^2 \quad (2.42)$$

with $c_{2i_j}^{\{k\}}$ approximate second order diagonal Hessian terms (or curvatures). Note we employ the abbreviated notation

$$\begin{aligned} f_j^{\{k\}} &= f_j(\mathbf{y}^{\{k\}}), \\ \left(\frac{\partial f_j}{\partial x_i} \right)^{\{k\}} &= \frac{\partial f_j}{\partial x_i}(\mathbf{y}^{\{k\}}). \end{aligned} \quad (2.43)$$

Only diagonal Hessian terms are considered to reduce storage requirements and to simplify the enforcement of positive definiteness. Keep in mid that ensuring that a full (or merely non-diagonal) Hessian is positive definite is non-trivial. Using the diagonal quadratic approximations (2.42) an approximate primal subproblem $\mathbb{P}_P[k]$ at $\mathbf{y}^{\{k\}}$ is constructed

$$\begin{aligned} \min_{\mathbf{y}} \quad & \tilde{f}_0^{\{k\}}(\mathbf{y}) \\ \text{subject to} \quad & \tilde{f}_j^{\{k\}}(\mathbf{y}) \leq 0, \quad j = 1, 2, \dots, m \\ & \tilde{f}_j^{\{k\}}(\mathbf{y}) = 0, \quad j = m + 1, m + 2, \dots, m + p, \end{aligned} \quad (2.44)$$

consisting of a quadratic objective and constraint functions. Problem $\mathbb{P}_P[k]$ can be solved with any number of nonlinear programming techniques. One such technique is transforming the problem to a second order cone problem (SOCP), we hope to investigate this in the future. From the approximate primal problem $\mathbb{P}_P[k]$, where all constraint functions are quadratic and diagonal, we construct an approximate quadratic program with linear constraints, easily solved with a host

of nonlinear programming techniques. That is, linearising the equality constraints renders the subproblem convex. We define problem $\mathbb{P}_{PQ}[k]$ as

$$\begin{aligned} \min_{\mathbf{y}} \quad & \tilde{f}_0^{\{k\}}(\mathbf{y}) = f_0^{\{k\}} + \nabla^T f_0^{\{k\}} \mathbf{s} + \frac{1}{2} \mathbf{s}^T \mathbf{Q}^k \mathbf{s} \\ \text{subject to} \quad & \tilde{f}_j^{\{k\}}(\mathbf{y}) = f_j^{\{k\}} + \nabla^T f_j^{\{k\}} \mathbf{s} \leq 0, \quad j = 1, 2, \dots, m \\ & \tilde{f}_j^{\{k\}}(\mathbf{y}) = f_j^{\{k\}} + \nabla^T f_j^{\{k\}} \mathbf{s} = 0, \quad j = m+1, m+2, \dots, m+p \end{aligned} \quad (2.45)$$

with $\mathbf{s} = (\mathbf{y} - \mathbf{y}^k)$. In an attempt to include some curvature information of the constraints, $\mathbf{Q}^{\{k\}}$ is the approximate Hessian of the Lagrangian of problem $\mathbb{P}_P[k]$. However, the Lagrangian multipliers $\lambda^{\{k^*\}}$ at the solution is of course unknown. Therefore the multipliers $\lambda^{\{k\}}$ of the solution at the previous iteration are used to construct the Hessian of the Lagrangian. Hence,

$$Q_{ii}^k = c_{2i_0}^{\{k\}} + \sum_{j=1}^m \lambda_j^{\{k\}} c_{2i_j}^{\{k\}} + \sum_{j=m+1}^{m+p} \lambda_j^{\{k\}} c_{2i_j}^{\{k\}} > 0 \quad (2.46)$$

with $Q_{ij}^{\{k\}} = 0 \forall i \neq j$ with $i, j = 1, 2, 3, \dots, n$. To ensure the subproblem is convex it is required that the diagonal Hessian matrix is positive definite. That is, $Q_{ii}^{\{k\}} > 0 \forall i$.

At this point various approximation strategies can be employed to obtain constants $c_{2i_j}^{\{k\}}$. As an introduction to the concept we discuss the most rudimentary approximation strategy requiring the calculation and storage of only a single constant per constraint function, referred to as the *spherical* quadratic approximation [22]. The approximating functions, an incomplete series expansion, is written as

$$\tilde{f}_j(\mathbf{y}) = f_j^{\{k\}} + \sum_{i=1}^n \left(\frac{\partial f_j}{\partial y_i} \right)^{\{k\}} (y_i - y_i^{\{k\}}) + \frac{1}{2} c_{2j}^{\{k\}} \mathbf{I} (y_i - y_i^{\{k\}})^2 \quad (2.47)$$

with \mathbf{I} the identity matrix. The unknown constant $c_{2j}^{\{k\}}$ is obtained by enforcing

$$\tilde{f}^{\{k-1\}} = f^{\{k-1\}} \quad (2.48)$$

That is, the approximate function should equal the original function value at the previous iteration. After some algebra the expression for $c_{2j}^{\{k\}}$ is obtained as

$$c_{2j}^{\{k\}} = \frac{2[f_j^{\{k-1\}} - f_j^{\{k\}} - \nabla^T f_j^{\{k\}} (\mathbf{y}^{\{k-1\}} - \mathbf{y}^{\{k\}})]}{\|\mathbf{y}^{\{k-1\}} - \mathbf{y}^{\{k\}}\|_2^2}. \quad (2.49)$$

This is merely a single example of an approximation strategy in SAOi. The reader is referred to the thorough discussion on incomplete series expansions as basis for function approximations in [22]. In the context of SAND simple analytic expressions are available to calculate first- and second-order sensitivities, allowing the use of exact second-order information when constructing function approximations.

2.4.4 Solving the approximate diagonal QP subproblems

The SAOi algorithm constructs a surrogate subproblem that approximates the local behaviour of the global problem in the current iteration point. The subproblem, a convex approximate diagonal

quadratic program (QP), is defined with a quadratic objective function and linear equality and inequality constraints. To obtain the solution of the approximate diagonal QP subproblem a number of nonlinear programming techniques can be employed. Here we review two of the most popular.

Dual methods

In mathematical optimisation the concept of *duality* refers to the fact that optimisation problems can be viewed in two ways. Thus far we have referred to the *primal* problem, that is the problem in the space of *design* variables. However, associated with every *primal* problem is its *dual*, the problem cast in the space of *Lagrange* multipliers, with some interesting and useful properties linking the two perspectives. Simply put, where the primal problem is a minimisation problem over design variables the dual problem is one of maximisation over dual variables.

The author in [10] employs the SAOi algorithm and a dual solution procedure to solve the diagonal QP subproblems constructed to approximate the local behaviour of the topology optimisation problem in each iteration point. The author notes that the advantage of the dual formulation is in its simple structure. It is well known that if the primal problem is continuous and strictly convex the solution corresponds exactly to that of the dual problem. What is more, if the number of constraints in the primal problem is less than the number of the design variables, the dual problem is much smaller than the primal. Similarly, the authors in [23] employ the SAO algorithm CONLIN with a dual solution procedure to solve the approximate subproblems that arise from the local stress constrained topology optimisation problem.

Both the solution procedures mentioned above are applied to a NAND formulated topology optimisation problem. That is, the system of equality constraints that define the state equation are removed from the optimisation problem and solved exactly in the analysis phase of each iteration. Therefore the optimisation problems considered above frequently do have less constraints than design variables, exploiting the primary advantage of the dual formulation. Of course, we consider the SAND formulated topology optimisation problem where a number of equality constraints, equal to the number of state variables, is included in the problem, hence a large number of constraints. What is more, considering the local stress constrained problem, a number of inequality constraints equal to the number of decision variables is also included in the problem. Therefore, considering the SAND problem, the primary advantage of the dual method, that is its typically low dimensionality, is eradicated.

Interior point methods

Recently many authors have commented on the success of interior point methods in solving the SAND formulated Topology optimisation problem, see for example [24]. Throughout this thesis we employ the IBM® ILOG® CPLEX® BARRIER algorithm (based on an interior point method) to solve the approximate diagonal QP subproblems. In contrast to popular nonlinear programming algorithms such as SIMPLEX, interior point methods reach the optimal solution by traversing the interior of the feasible domain.

To briefly highlight some characteristics of an interior point algorithm, consider the general non-

linear optimisation problem defined before

$$\begin{aligned} \min_{\mathbf{y}} \quad & f_0(\mathbf{y}) \\ \text{subject to} \quad & f_j(\mathbf{y}) \leq 0, \quad j = 1, 2, \dots, m \\ & f_j(\mathbf{y}) = 0, \quad j = m + 1, m + 2, \dots, m + p, \end{aligned} \quad (2.50)$$

with $f_0(\mathbf{y})$ the objective function and $f_j(\mathbf{y})$, $j = 1, 2, \dots, m$ and $f_j(\mathbf{y})$, $j = m + 1, m + 2, \dots, m + p$ the inequality and equality constraint functions respectively. The idea behind the method is to reduce problem (2.50) to a sequence of linear equality constrained problems and apply Newton's method. This is achieved by incorporating a penalised version of the inequality constraints in the objective function. That is, problem (2.50) is replaced by

$$\begin{aligned} \min_{\mathbf{y}} \quad & f_0(\mathbf{y}) + \sum_{j=1}^m I_-(f_j(\mathbf{y})) \\ \text{subject to} \quad & \mathbf{A}\mathbf{y} = \mathbf{b}, \end{aligned} \quad (2.51)$$

where I_- is the indicator function

$$I_-(u) = \begin{cases} 0, & \text{if } u \leq 0 \\ \infty, & \text{if } u > 0 \end{cases} \quad (2.52)$$

and $\mathbf{A}\mathbf{y} = \mathbf{b}$ denotes linearised equality constraints with $\mathbf{A} \in \mathbb{R}^{p \times n}$ with n the number of design variables. However, although the problem is now one only in equality constraints, the objective function is not differentiable. To remedy this the indicator function is replaced with an approximate differentiable indicator function \check{I}_- such that $\check{I}_-(u) = -t \ln(-u)$. Here parameter t is a small positive scalar, often referred to as the *barrier* parameter. The logarithmic barrier function corresponding to (2.51) is then

$$f_0(\mathbf{y}) - t \sum_{j=1}^m \ln(-f_j(\mathbf{y})). \quad (2.53)$$

Newton's method can now be applied to the equality constrained problem, as discussed before, with definite rules in updating the barrier parameter. The reader is referred to the extensive literature on the subject for more details. Importantly, $\ln(-f_j(\mathbf{y}))$ is undefined if constraint j is infeasible. Therefore the method requires a completely feasible starting point in each iteration.

Chapter 3

Material penalisation in simultaneous analysis and design: Towards discrete solutions

3.1 Background

Topology optimisation formulated as a material distribution problem is fundamentally a discrete programming problem. The optimal topology or layout of an isotropic material is sought in a given design domain with boundary conditions and applied loads. The word *topology*, from Greek, refers to ‘place’ and ‘study’ or the ‘study of place’. That is, in topology optimisation of continuum structures we seek the optimal place or position of infinitesimal volumes of material to satisfy a set of constraints that describe physical restrictions on the design. In this form the problem actually lacks solutions. As discussed in section 2.2.2, the structure can be optimised *ad infinitum* by distributing material in ever finer microstructures. Therefore from the outset the design space is ‘artificially’ modified for solutions to even exist.

The homogenisation approach, proposed in [3], addresses the non-existence issue by expanding the design space to include anisotropic material. A method popular because the discrete programming problem is justifiably relaxed to a continuous one in decision variables, hence the said variables retain physical meaning at intermediate values. In other words, material is no longer restricted to either solid or void states at every spatial position but forms a continuous range of physical states between ‘solid’ and ‘void’.

The act of discretizing the structural analysis component of the problem is an example of restricting the design space, resulting in the existence of solutions. In effect discretization incorporates a minimum length scale on the topology of the structure. The same can be achieved by including a minimum length scale constraint in the continuum case. However, the problem is discretized primarily from a computational point of view. That is, discretization of structural analysis allows solution by numerical/algorithmic means, with solution in multidimensional continuum space impossible in most instances. Therefore problems to which the homogenisation approach is applied are typically (although not necessarily) discretized to allow solution by numerical means. For instance, in the paper which introduces the homogenization approach [3] a FE approximation scheme

is constructed and applied to the structural analysis component of the problem. From the outset we consider problems with structural analysis discretized via the FE method resulting in a large set of algebraic equations that describe structural responses.

Although the discretization of structural analysis results in the existence of solutions the topology optimisation problem is still discrete in decision variables. As discussed in section 2.2.2 the problem is NP-complete and cannot be solved in polynomial time. Therefore, the discrete requirement on decision variables is relaxed to allow solution by gradient based optimisation methods (in polynomial time). However, we are interested in solid-void designs, as should be the case considering the topology optimisation problem, hence material penalisation is required to drive material to purely solid-void states. A popular method is the simple isotropic material with penalisation (SIMP) approach, as mentioned in section 2.2.2 with more details following in this chapter. In other words, relaxation of discrete requirements on decision variables is primarily done from a computational point of view, similar to the discretization of structural analysis. That is, a problem with decision variables in continuous space can be solved in polynomial time with classical gradient based optimisation methods. However, decision variables at an intermediate state, not strictly solid or void, only represent physical material under some conditions and/or assumptions.

We close this discussion on the relaxed but penalised topology optimisation problem with discretized structural analysis by emphasising the desire to obtain primarily black-and-white (solid-and-void) designs. That is, in general, material at intermediate values can not be interpreted physically and, in principle, by solving the relaxed and penalised problem for computational reasons we hope to obtain solutions to the original discrete topology optimisation problem. Therefore solutions to the problem is primarily evaluated with a measure of the amount of material that attain strictly solid or void states, quantified by a so called *black-and-white* fraction, defined as

$$\phi_{B\&W} = \frac{n_{[0]} + n_{[1]}}{n}, \quad (3.1)$$

where n is the total number of decision variables and $n_{[1]}$ and $n_{[0]}$ denote the number of decision variables that attain a state of exactly solid or void respectively.

3.2 Introduction

In a simultaneous analysis and design (SAND) setting material penalisation enters the optimisation problem directly via the state equation. Decision variables in the state equation are replaced with monotonically increasing functions that artificially penalise the stiffness of material not strictly deemed solid. Conversely, in the nested analysis and design (NAND) formulation material penalisation enters the structural analysis component of the problem, distinct from the optimisation phase. Therefore material penalisation is facilitated via an implicit relation of decision variables to state variables due to the exact solution of the state equation in each iteration. In the light of this we investigate the characteristics of the basic SAND formulated relaxed and penalised topology optimisation problem with a single constraint on compliance or volume.

Once penalisation functions are applied to the SAND formulated problem analytic second-order information is not only available but an important factor in converging to the solution of the problem. In sequential approximate optimisation (SAO) diagonal second-order information is typically

used to construct conservative and convex approximate subproblems which guarantee convergence to a solution. However, in a SAND setting second-order information arises from the state equation, a set of equality constraints in the optimisation problem. Therefore we investigate some strategies to construct second-order approximations to the problem (or the state equation) while ensuring the subproblem is convex - with a positive definite Hessian - in each iteration.

In the SAND formulation of the problem the state equation that describes structural analysis is not solved *per se* and enters the optimisation problem as a set of constraints. Therefore unlike the nested case where explicit solution of the state equation inhibits a zero lower bound on decision variables the complete removal of material from the design domain is simply facilitated with a zero lower bound on decision variables. Not only is this an accurate representation of the problem physically, but the removal of variables and constraints from the problem which do not contribute to the feasibility of the design reduces computational requirements. That is, in effect, subproblems decrease in size as decision variables attain values of zero and material is removed from the design domain.

As mentioned before, material penalisation simply enters the SAND formulated problem by replacing decision variables with monotonically increasing functions in the state equation. Therefore various penalisation schemes can be tested by defining these functions in different ways. What is more, with the penalised state equation yielding second-order information the penalisation function is intimately linked to the nature of the approximate subproblem in each iteration. In addition, progressive penalisation can in principle increase the amount of material that assumes purely solid or void states. Here we test various approximation schemes (or functions) along with a preliminary investigation into progressive penalisation to establish some characteristics of the SAND formulated topology optimisation problem.

Finally we note that this chapter should be considered a chronicle of an investigation into some effective approximation and penalisation schemes in a SAND setting. Naturally it is impossible to test each and every strategy and permutation, but we hope to discover some underlying characteristics of the problem by testing approximation and penalisation schemes on a simple test problem. Once it is determined which strategies and permutations are most effective the solution procedure is applied to a range of problems in the final section.

3.3 Problem formulation

Departing from the fundamental topology optimisation problem discrete decision variables are relaxed and penalised via the state equation. We define two versions of this problem, minimum compliance subject to a constraint on volume, and minimum volume subject to a constraint on compliance.

Problem P_W^C :

$$\begin{aligned} \min_{x,q} \quad & \mathbf{f}^T \mathbf{q} \\ \text{subject to} \quad & \sum_{i=1}^n x_i \leq \bar{v}, \\ & \mathbf{K}(\Pi(x_i)) \mathbf{q} = \mathbf{f}, \\ & 0 \leq x_i \leq 1 \quad i = 1, 2, \dots, n. \end{aligned} \tag{3.2}$$

Problem P_C^W :

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{q}} \quad & \sum_{e=1}^n x_e \\ \text{subject to} \quad & \mathbf{f}^T \mathbf{q} \leq \bar{c}, \\ & \mathbf{K}(\Pi(x_i)) \mathbf{q} = \mathbf{f}, \\ & 0 \leq x_i \leq 1 \quad i = 1, 2, \dots, n. \end{aligned} \quad (3.3)$$

Function $\Pi(x_i)$ is continuous and monotonically increasing on $x_i \in [0, 1]$. The penalisation schemes presented in section 3.5.3 are merely different definitions of function $\Pi(x_i)$. We do not consider penalisation of decision variables in the objective function (minimum weight/volume) and/or inequality constraint (minimum compliance) in the hope that penalisation directly via the state equation, now part of the optimisation problem in a SAND setting, is sufficient in driving the design to a solid-void state.

The number of elements in the mesh and thus the number of decision variables is denoted by n and the number of state variables by u . In problem P_W^C and problem P_C^W the vector of decision variables have dimension $\mathbf{x} \in \mathbb{R}^n$ and state variables $\mathbf{q} \in \mathbb{R}^u$. Upper and lower bounds on decision variables are 0 and 1, denoting the presence or absence of material on the bounds. The vector \mathbf{f} denotes nodal forces (assumed to be design independent) along with the limits on compliance and volume denoted by \bar{c} and \bar{v} respectively.

3.4 Sequential approximate optimisation and simultaneous analysis and design

In this chapter we consider problems where the objective function and inequality constraints are linear and can be neglected in this analysis. That is, only the equality constraints (the penalised state equation) have non-zero diagonal Hessian information. Therefore the approximate Hessian of the Lagrangian reduces to the approximate Hessian of the Lagrangian of only the state equation, denoted by

$$\bar{Q}_{ii}^k = \sum_{j=1}^u \lambda_j^{\{k\}} c_{2i_j}^{\{k\}} > 0. \quad (3.4)$$

At this point various approximation strategies can be employed to obtain constants $c_{2i_j}^{\{k\}}$. We consider two approximation strategies with two ‘sub-strategies’ to ensure positive definiteness. The first approximation strategy, requiring the calculation and storage of only a single constant per constraint function, is referred to as the *spherical* quadratic approximation [22]. Refer to section 2.4.3 for more details.

Before moving on to the second approximation approach we discuss the sub-strategies to ensure the resulting approximate Hessian is positive definite, a continuation of the discussion in section 2.4.3. We are dealing with equality constraints where the concept of a conservative approximation breaks down and Lagrangian multipliers might assume positive or negative values. Positive definiteness of the approximate Hessian can be enforced by applying either of the following conditions, which we refer to as *convex* and *non-convex*. Note that the convexity does not apply in this sense to equality

constraints and the terms might be ambiguous. However, we use the terms to refer to the nature of curvature information included in constructing the approximate Hessian of the problem. That is, in the *convex* strategy only individual convex terms are included in the approximate Hessian, in the *non-convex* strategy non-convex terms are also included:

- **Convex:**

Only include terms in \bar{Q}_{ii}^k which satisfy $\lambda_j^{\{k\}} c_{2i_j}^{\{k\}} > 0$ for all i, j .

- **Non-convex:**

Include all terms in \bar{Q}_{ii}^k : Let $\lambda_j^{\{k\}} c_{2i_j}^{\{k\}}$ be of arbitrary sign for all i, j with positive definiteness enforced by

$$\bar{Q}_{ii}^k = \max\{\epsilon_h > 0, \sum_{j=1}^u \lambda_j^{\{k\}} c_{2i_j}^{\{k\}}\}$$

with $i = 1, 2, \dots, n$ and ϵ_h some small positive value (we employ 1×10^{-6}).

This is an extension from the work done in [25] wherein it is shown that the inclusion of some non-convex information increases the accuracy of the each subproblem leading to less iterations for convergence. However in [25] only inequality constraints are considered. Here we test the effects of the two strategies in approximating the equality constrained problem.

The second approximation strategy (the method to obtain values for $c_{2i_j}^{\{k\}}$) we refer to as the *analytic* quadratic approximation. Again the term is slightly ambiguous, chosen to emphasise the contrast to typical second-order approximation methods where values for $c_{2i_j}^{\{k\}}$ are calculated based on zero- and first-order information, as for example the *spherical* approximation discussed before. See also the work in [22] for a thorough overview of approximation strategies in the light of incomplete series expansions. Structural analysis based on the finite element (FE) method (as is typically the case) results in simple analytic expressions that form the set of nonlinear equations that make up the state equation. Obtaining analytic expressions for derivatives to the equations are fairly trivial. What is more, the nature of the equations, as the FE method itself, is very well suited to numerical computation. Therefore in SAND exact second-order information is available in analytical form and requires very little mathematical and computational effort to evaluate.

Considering the *analytic* quadratic approximation, again, only diagonal Hessian terms are considered to limit storage requirements and reduce positive definite requirements to the trivial case discussed above. Keep in mind that although all diagonal information is exact the resulting Hessian to the Lagrangian is still approximate. To be clear, only diagonal terms are considered and the Lagrangian multipliers from the previous iteration are used to construct the Lagrangian function.

The penalisation scheme $\Pi(x_i)$ and the resulting approximate Hessian is of course intimately linked. However, here we consider the simplest form of penalisation as per the SIMP material law to investigate the respective approximation strategies. Also, we wish to determine the effects of the two sub-strategies to enforce positive definiteness of the approximate Hessian to the Lagrangian. We consider problem P_C^W and the two bar truss ground structure with $m = 4$ for numerical experiments. In Table 3.1 results are presented for the various approximation strategies along with a graphical depiction of the respective topologies generated in Figure 3.1. Volume fraction (the

objective function) at solution is denoted by f_v , $\phi_{B\&W}$ is the black-and-white fraction, N_{iter} is the number of iterations to solution and T_{avg}^I is the average CPU time required per iteration in seconds (with the abbreviations that denote the approximation strategies and enforcement of positive definiteness intuitive).

Table 3.1: A comparison of the results obtained for the weight minimization of the two-bar truss using various approximation strategies.

Approximation	f_v	$\phi_{B\&W}$	N_{iter}	T_{avg}^I
SPH CON	0.334	0.218	27	2.859
SPH NONCON	0.251	0.914	23	1.749
ANA CON	0.250	0.987	220	0.735
ANA NONCON	0.250	0.992	29	1.337

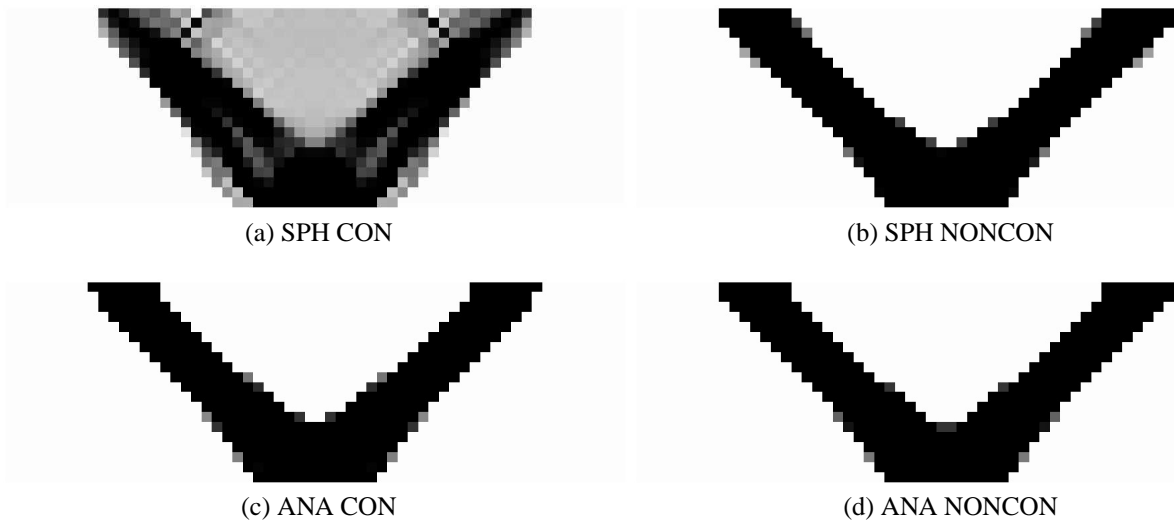


Figure 3.1: A comparison of the topologies generated for the weight minimization of the two-bar truss using various approximation strategies.

The *convex* and *spherical* approximation performs worst, clear from the results summarised in Table 3.1 and the graphical depiction of the topologies generated in Figure 3.1. Clearly the algorithm converges prematurely to a suboptimal solution. Graphically the three remaining cases have very little to separate them. However, the *convex* and *analytic* approximation requires an increased amount of iterations to convergence whilst the *nonconvex* and *spherical* approximation performs surprisingly well, although the *nonconvex* and *analytic* approximation ultimately converges to the highest black-and-white fraction in the least amount of time. To shed some light on the characteristics of the approximations we introduce the concepts of *state curvature* and *decision curvature*, i.e. measures to quantify the amount of curvature related to second derivatives to state variables and decision variables respectively in each subproblem (or iteration k). We employ the Euclidean

norm and define *state curvature* in iteration k as

$$\bar{Q}_q^{\{k\}} = \left[\sum_{i=1}^u \bar{Q}_{ii}^2 \right]^{\frac{1}{2}} \quad (3.5)$$

and *decision curvature* in iteration k as

$$\bar{Q}_x^{\{k\}} = \left[\sum_{i=u+1}^n \bar{Q}_{ii}^2 \right]^{\frac{1}{2}} \quad (3.6)$$

where u is the number of state variables and n the number of decision variables. Note that it is assumed the first u elements in the approximate diagonal Hessian are the second derivatives to state variables followed by n second derivatives to decision variables.

Figures 3.2 and 3.3 depict convergence plots for each of these curvature measures. Immediately of note is that although second-order derivatives to state variables is actually zero (the state equation is linear in state variables) the *spherical* approximations include curvature information to state variables, as seen from Figure 3.2. Interestingly it would seem this does not have a adverse effect on convergence since the *spherical* and *non-convex* approximation performs fairly well. Therefore it would seem convergence characteristics of the various approximations are solely related to the amount of curvature imposed in decision variables.

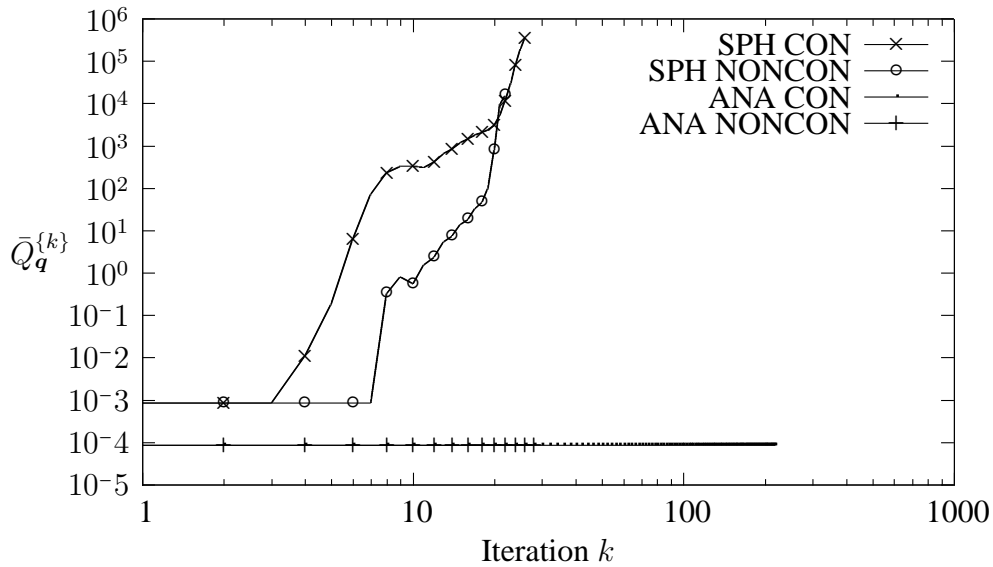


Figure 3.2: Norm of diagonal Hessian terms ($\bar{Q}_q^{\{k\}}$) related to state variables in each iteration for various second-order approximations.

Considering Figure 3.3 it would seem the *spherical* and *convex* approximation converges prematurely due to inflated or excessive curvatures in decision variables. Especially so in comparison to the curvatures for the *spherical* and *non-convex* approximation for $k < 20$. That is, although curvatures for the *spherical* and *non-convex* approximation is also inflated compared to the *analytic*

case, the inflated curvatures only occur ‘close’ to the optimal solution. From Figure 3.3 we might also deduce the reason behind the amount of iterations required for convergence concerning the *analytic* and *convex* approximation. As is clear, due to the inclusion of only convex terms curvatures are also inflated, similar to the *spherical* scenarios. However, for the *analytic* case curvatures are already inflated in early iterations. Curvature information can be viewed as a sort of move limit imposed on the step size of the algorithm, the *analytic* and *convex* approximation does not allow large enough steps to converge to the solution quickly, although the optimal solution is eventually reached. Hence we might say the approximation is too conservative, intuitive because only positive terms are added to the approximate Hessian. Finally, as one would expect, the *analytic* and *non-convex* approximation performs best due to the trade-off between accuracy of the approximation and ensuring the diagonal Hessian is positive definite.

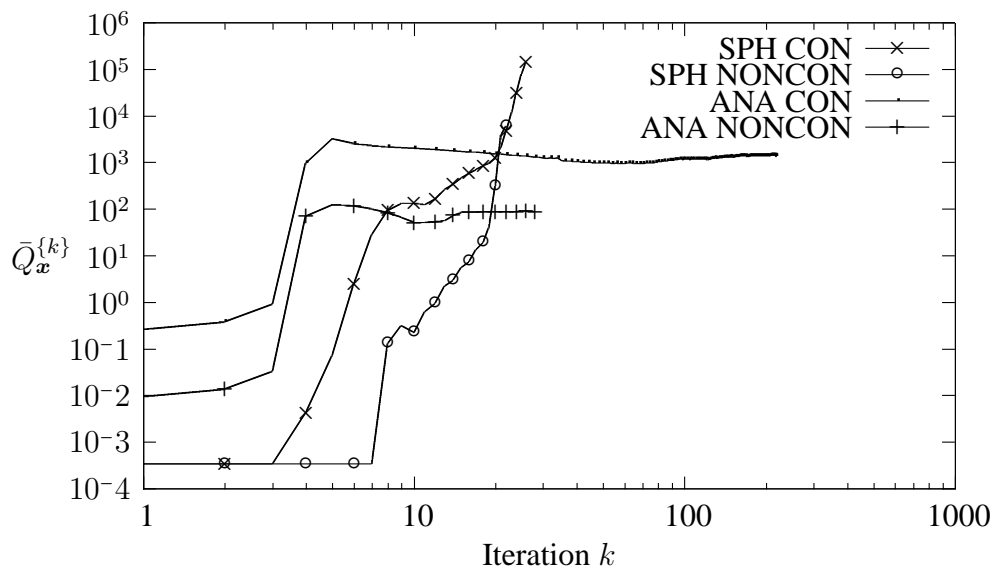


Figure 3.3: Norm of diagonal Hessian terms ($\bar{Q}_x^{\{k\}}$) related to decision variables in each iteration for various second-order approximations.

Also of interest is that, although the *convex* and *analytic* approximation requires an exceeding amount of iterations to converge, the average iteration times are fairly low, as indicated in Table 3.1. It seems that as more decision variables become active on the upper and especially lower bounds the subproblem becomes easier to solve. This observation forms a neat introduction to the following section where, along with other penalisation schemes, the computational benefits that arise from allowing decision variables to obtain a value of exactly zero on the lower bound is investigated.

3.5 Interpretation and penalisation of decision variables

3.5.1 Zero lower bounds

The topology optimisation problem solved via the traditional nested analysis and design (NAND) formulation involves the calculation of structural responses via the exact solution of the state equation in each iteration. Typically in this setting a non-zero lower bound is required on decision variables to avoid fatal ill-conditioning of the global stiffness matrix. In other words, with zero lower bounds on decision variables some elements attain zero stiffness which in turn causes the solution of the state equation to become undefined.

In contrast, in a SAND setting, the state equation is not solved *per se* allowing a value of exactly zero on the lower bound of decision variables. However, material derivatives vanish completely from the problem if material attains a state of absolute void; material can not return to the design space once removed. This might seem catastrophic, but the SAND formulated topology optimisation problem is non-convex anyway, therefore from the outset we can only hope to converge to a local stationary point of the problem. Note that the subproblems are constructed with a quadratic objective function and linear constraints, that is a convex subproblem, and constraint qualification always holds if a feasible solution exists according to Slater's condition. Briefly, Slater's condition states that constraint qualification always holds if the problem is convex and there exists a strictly feasible design vector. The interested reader is referred to the more detailed description of Slater's condition in [20].

The primary advantage of allowing absolutely void material on the lower bound is in computational benefits. That is, with a zero lower bound on decision variables the number of non-zero elements in the constraint Jacobian decreases as the SAO algorithm converges to the solution, hence reducing the time required to solve each subproblem. What is more, entire constraints can be removed from the problem if all the related decision variables attain a value of zero; if all material elements related to that constraint become void.

In [26] an algorithmic procedure is developed to alter the global stiffness matrix such that the state equation can be solved with zero lower bounds on decision variables in a NAND setting. Although the author claims the algorithm is "simple" a number of procedures are required to address the issue of the resulting discontinuity in structural responses, avoiding potential rigid body modes and the reintroduction of elements if the optimisation algorithm deems it necessary. The author concludes that the main advantage of zero lower bounds is in computational requirements. In a NAND setting the elemental stiffness matrix need not be assembled if the corresponding decision variable acquires a value of zero. In [27] and [28] truss problems are solved in a SAND setting with decision variables allowed to take on values of exactly zero. The SAND formulation being obviously beneficial, for trusses in the ground structure can be removed completely with no further complications. However, numerical issues are typically of concern, as discussed in the context of penalisation functions in the following section.

To illustrate the effect of a non-zero lower bound on decision variables we consider the two bar truss test problem. Specifically the problem of minimum weight subject to a constraint on compliance (Problem P_C^W) with SIMP penalisation at $p = 3$ and the *convex* and *analytic* approximation scheme. A fairly large lower bound is chosen (0.1) for illustrative purposes. However, we could

argue any non-zero lower bound, regardless of value, is problematic in interpretation of the final design. That is, material on the lower bound does contribute to the physical properties of the structure, however minute the effect might be. In Figure 3.4 the topologies generated with zero and non-zero lower bounds are depicted. Graphically the optimal designs are not completely identical, although with a relatively large non-zero lower bound the effect is obviously pronounced. We emphasise these results are presented for illustrative purposes with a primary focus on computational aspects to follow.

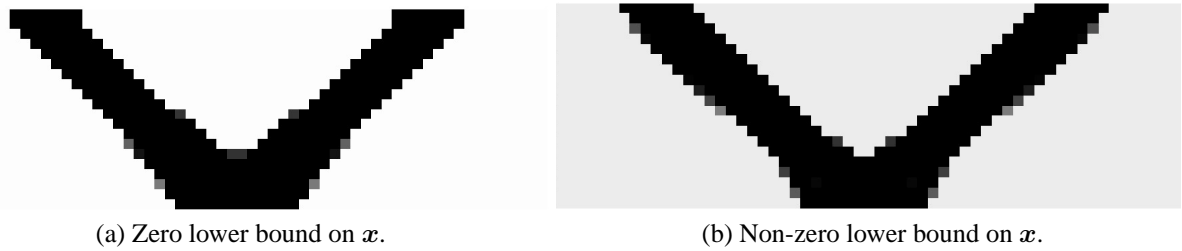


Figure 3.4: A comparison of topologies generated with zero and non-zero lower bounds on decision variables.

Table 3.2 is a comparison of the objective function values, expressed as volume fractions (f_v), and various aspects of the solution times. The total CPU time to solution, in seconds, is denoted by T_{total} and N_{iter} is the number of iterations. Furthermore, average time per iteration is denoted by T_{μ}^I in seconds. As would be expected, the volume fraction at optimality is much larger for the non-zero lower bound case. More importantly, the algorithm requires more than double the amount of time to converge to the optimal solution for the non-zero lower bound case, also requiring more iterations. The final column in Table 3.2 clearly illustrates the computational benefits which accompanies the zero lower bound case. Average CPU time per iteration is much less due to the progressive decrease in size of the subproblems as some decision variables and constraints no longer contribute to the feasibility of the design.

Table 3.2: Comparison of solution quality and computation time for zero and non-zero lower bounds.

Lower bound	f_v	T_{total}	N_{iter}	T_{μ}^I
0	0.2498	25.99	29	0.896
0.1	0.3252	61.69	42	1.469

Figure 3.5 is a plot of the CPU time per iteration as the algorithm progresses and Figure 3.6 a plot of the number of decision variables active on the lower bound at each iteration. Clearly the subproblems become smaller in size and easier to solve as more decision variables attain values of exactly zero.

In conclusion we point out that zero lower bounds on decision variables (thus allowing material to attain a state of void) is a fundamental property of the topology optimisation problem. As mentioned before, topology optimisation is concerned with distributing solid isotropic material in

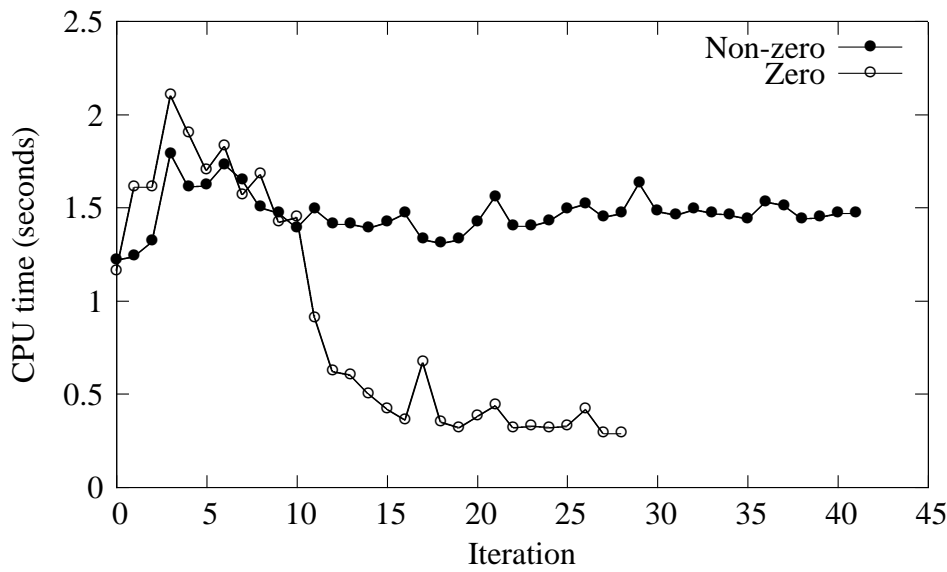


Figure 3.5: CPU time per iteration

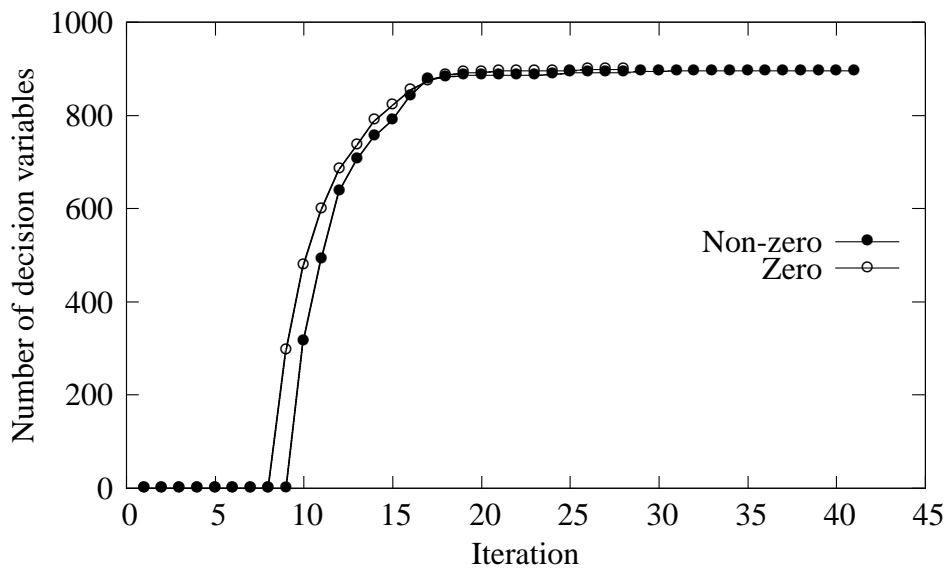


Figure 3.6: Number of decision variables active on lower bound per iteration

an optimal way, with the rest of the structure absolutely void, thus unifying the optimisation of the material properties and the connectivity of the domain. However, the main advantage of a zero lower bound on decision variables is in computational requirements. What is more, considering local constraints, a zero lower bound on decision variables provide a natural and mathematically consistent way of removing constraints from the problem. Although the reader should keep in mind the computational benefits illustrated above are due to a set of decision variables that attain a state of void, resulting in the ‘vanishing’ of the appropriate equality constraint in the state equation.

In conjunction with the definition of local constraints in section 2.3, consider a local constraint g_i on element i of the form

$$g_i(\mathbf{q}_{elmi}) \leq 0. \quad (3.7)$$

Typically such a constraint should not contribute to the feasibility of the design if the corresponding decision variable (x_i) attains a state of void, i.e. if the material to which the constraint is applied vanishes from the problem. A simple way of achieving this is by reformulating the constraint as a vanishing constraint, resulting in a mathematical program with vanishing constraints (MPVC) [29]. The local constraint is simply multiplied with decision variable $x_i \in [0, 1]$

$$x_i [g_i(\mathbf{q}_{elmi})] \leq 0. \quad (3.8)$$

Note that x_i is always positive, therefore the reformulation does not affect the feasibility of the constraint function for non-zero x_i . However, the constraint is always feasible if $x_i = 0$ - if the related material attains a state of void. This is much more complicated to achieve and, more importantly, to justify if decision variables have non-zero lower bounds. Furthermore, similar to the computational benefits highlighted above, local constraints related to material that attains a state of void no longer contribute to the design, reducing the size of the subproblem in each iteration. More on vanishing constraints in Chapter 4.

3.5.2 The Hashin-Shtrikman bound and realistic microstructures

Considering the relaxed topology optimisation problem material is penalised such that predominantly solid-void states are obtained at every spatial position in the design domain. Therefore severe penalisation might be required to approach purely discrete designs. Referring to equation (2.3) the penalisation function $\Pi(\mathbf{x})$ should in principle reduce the stiffness of any non-solid material to zero via the penalised state equation. However, in the context of SAO, increased severity of penalisation leads to less accurate approximate subproblems. This is especially problematic concerning the SAND formulation of the problem where structural analysis is performed via a linearised approximation to the state equation in each iteration (with curvature information only entering the problem via an approximate and separable Lagrangian function). Of course, ‘infinite’ penalisation is possible in principle, for example a very large value for SIMP parameter p , as depicted in Figure 3.7. However, clear for very high values of p , the penalisation function practically assumes the behaviour of a discrete function (the difficulty we are trying to avoid in the first place by relaxing discrete decision variables). Therefore, with a trade-off between computational complexity and accuracy of modelling discrete material properties, the optimal solution would typically contain some ‘grey’ material (not at values of exactly 1 or 0). Since material penalisation is considered to lead to a ‘fictitious material’ model, how should material at non-binary states be interpreted?

It turns out that, if the severity of material penalisation is sufficient, the stiffness of ‘grey’ material can be represented physically by a composite microstructure built up from void and the original solid isotropic material. Hence decision variables are typically considered to denote material ‘density’, since these variables could be viewed as the ratio of solid-void material in the microstructure. In [30] upper and lower bounds are derived for the effective elastic moduli of multiphase isotropic materials and in [31] material penalisation schemes are reviewed in the light of these bounds. For

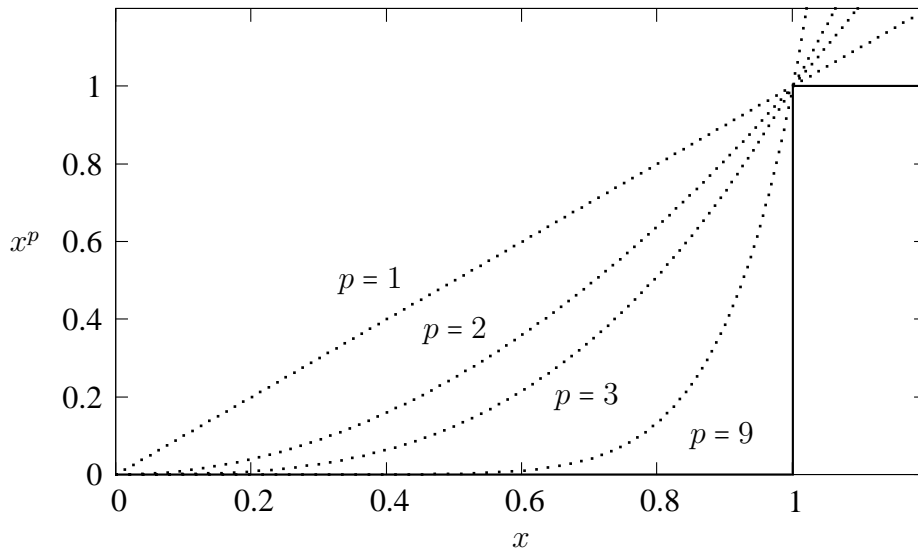


Figure 3.7: Simple isotropic material with penalisation (SIMP)

the sake of brevity, we do not derive the material stiffness bounds and the reader is referred to the aforementioned papers for more detail. Briefly, it can be shown that in 2-D with a Poisson ratio of $\frac{1}{3}$ the material penalisation function should satisfy

$$\Pi(x) \leq \frac{x}{3 - 2x}, \quad (3.9)$$

for decision variables at intermediate values to be related to a physical composite microstructure. Here x denotes the decision variable that describes the presence or absence of material at some spatial position. For the sake of brevity we simply denote the penalisation functions in terms of a single variable x .

For the relation defined in (3.9) it is assumed the Poisson ratio is independent of density, although in [31] the author shows the relation also holds for a constant Poisson ratio (we employ a value of $\frac{1}{3}$ throughout this thesis). The exact configuration of the composite structure can be obtained through an inverse homogenisation process [32]. Although this method provides a realistic means of interpreting non-binary decision variables at optimal solution, obtaining a purely solid-void design remains of primary concern. In the following section some popular material penalisation schemes are defined and tested in relation to the bound described in equation (3.9).

3.5.3 Material penalisation

SIMP

The most popular approach to material penalisation, as has been mentioned multiple times throughout this thesis, is referred to as the solid isotropic material with penalisation approach (SIMP), as proposed in [1]. By far the most simple and intuitive of all the penalisation schemes, SIMP is

defined as

$$\Pi_{SIMP}(x) = x^p \text{ for } p > 1. \quad (3.10)$$

In Figure 3.8 the penalisation scheme with $p = 2$ and 3 is compared to the Hashin-Shtrikman function. It is clear to see that the Hashin-Shtrikman bound is satisfied for $p \geq 3$.

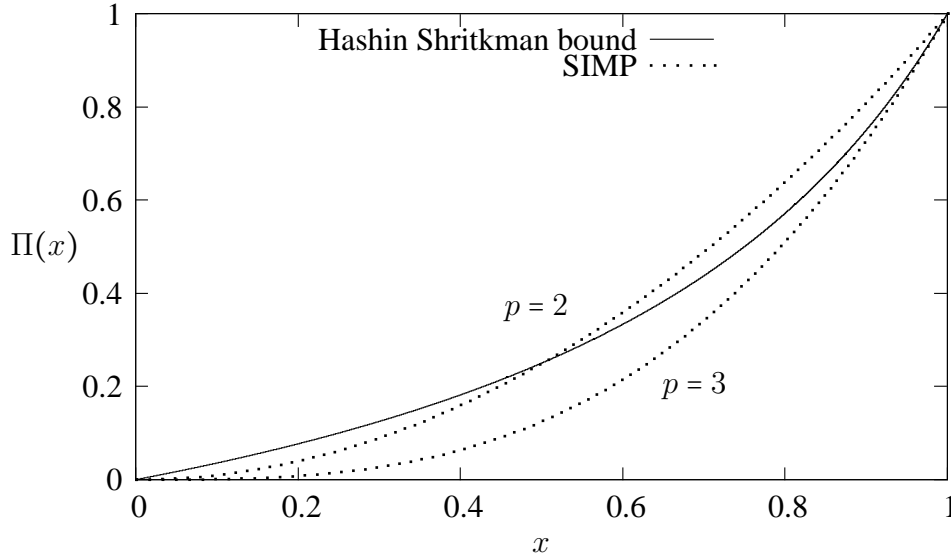


Figure 3.8: A comparison of simple isotropic material with penalisation with $p = 2$ and 3 and the Hashin-Shtrikman bound for a Poisson ratio of $\frac{1}{3}$.

SINH

In [13] the SINH method of material penalisation is introduced, which is defined as

$$\Pi_{SINH}(x) = \frac{\sinh(px)}{\sinh(p)} \text{ for } p > 1. \quad (3.11)$$

The author generalises the definition of penalty functions and applies independent penalisation schemes to the structural analysis component of the problem and other constraints (a volume constraint for example). Furthermore, the author conjectures that SINH penalisation leads to a well posed optimisation problem. An important distinction between SIMP and SINH penalisation is that while $\Pi_{SINH}(0) = 0$ the first derivative of the function does not vanish for $x = 0$. For this reason, with the emphasis on a zero lower bound on decision variables in a SAND setting, we investigate the effectiveness of the SINH penalisation scheme. Similar to SIMP, SINH penalisation satisfies the Hashin-Shtrikman bound for $p \geq 3$, as depicted in Figure 3.9.

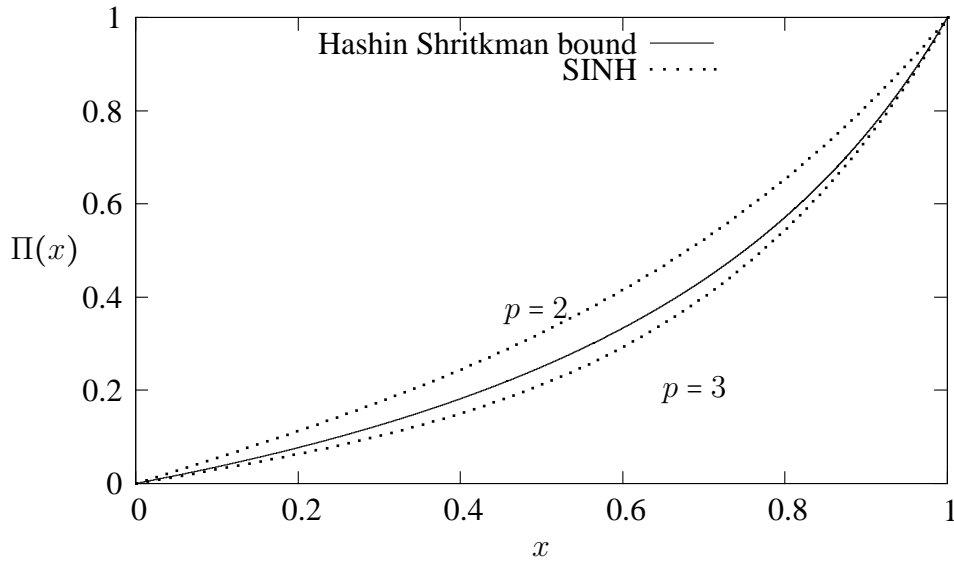


Figure 3.9: A comparison of SINH penalisation with $p = 2$ and 3 and the Hashin-Shtrikman bound for a Poisson ratio of $\frac{1}{3}$.

The Hashin-Shtrikman bound (H-S penalisation)

As noted by the author in [31] it is rather surprising that the Hashin-Shtrikman bound itself is not used more often as penalisation function. Consider a penalisation function defined as

$$\Pi_{H-S}(x) = \frac{x}{3 - 2x} . \quad (3.12)$$

Here the relation in equation (3.9) simply forms the penalisation function. We refer to this formulation as H-S penalisation. Interestingly, per definition this is the least ‘severe’ penalisation function one can employ while ensuring physical interpretation of grey material. However, similar to SINH penalisation the question remains how accurate an SAO algorithm based on incomplete Taylor series expansions can approximate the rational function compared to the polynomial natured SIMP function.

RAMP

The final penalisation scheme considered due to its popularity is often referred to as rational approximation of material properties (RAMP), as proposed in [33] (although not referred to in this manner in the original paper). In the extensive work on topology optimisation in [8] the term RAMP is coined. Defined in terms of inverse material stiffness, referred to as a linear concave interpolation [33],

$$\frac{1}{E(x)} = \frac{1}{E^{min}} + x \left[\frac{1}{E^0} - \frac{1}{E^{min}} \right] \quad (3.13)$$

which can be written as the rational function

$$E(x) = E^{min} + \frac{x}{1 + q(1 - x)} [E^0 - E^{min}] . \quad (3.14)$$

The RAMP penalisation scheme is proposed in the context of minimum compliance design. For large values of q the objective function becomes concave which is advantageous if we seek purely 1-0 designs located on the ‘border’ of the design space [33] [8]. However, values of q which satisfy the concave requirement is defined by

$$q \geq \frac{E^0 - E^{min}}{E^{min}} \quad (3.15)$$

which is undefined for $E^{min} = 0$. This is also clear from the initial definition in (3.13). Simply put, q has to be infinitely large for the compliance objective to become concave if a zero lower bound is employed on penalised stiffness. That is, the main advantage of this penalisation scheme can not be realised if we wish to employ zero lower bounds on decision variables (hence a zero lower bound on material stiffness); one of the main advantages associated with the SAND formulation of the problem. In the light of this and a desire for a general material penalisation scheme relevant to both minimum compliance and minimum weight problems we do not consider the RAMP scheme further.

On progressive penalisation and convergence of penalised problems

A number of authors comment on a scheme whereby progressive penalisation is applied to the relaxed topology optimisation problem. That is the severity of penalisation is increased as the algorithm converges to the optimal solution in the hope that purely solid-void designs are eventually obtained. Before considering such a scheme it is natural that we should first study the convergence properties of the relaxed but penalised topology optimisation problem in general. Until recently no proof could be given that the optima of the relaxed but penalised problem will converge to the optima of the discrete problem. However, in 2001, considering the simple discrete programming problem

$$\begin{aligned} \min_x \quad & f_0(x_e) \\ \text{subject to} \quad & \mathbf{x} \in \mathcal{D}, \\ & x_e \in \{0, 1\}, \\ & e = 1, 2, \dots, n \end{aligned} \quad (3.16)$$

where $\{0, 1\}$ indicates the discrete set, $\mathcal{D} \subset \mathcal{R}^n$ the feasible set and n the number of design variables, Reitz proved, under some severe assumptions, that problem (3.17)

$$\begin{aligned} \min_x \quad & f_0(x_e^p) \\ \text{subject to} \quad & \mathbf{x} \in \mathcal{D}, \\ & x_e \in [0, 1], \\ & e = 1, 2, \dots, n, \\ & p > 0 \end{aligned} \quad (3.17)$$

will converge to the solution of (3.16), assuming the global minimum of (3.17) can be obtained [34]. Note that here $[0, 1]$ denotes the continuous range $0 \leq x_e \leq 1$. For the proof to hold it is assumed that

(i) Problem (3.16) has only one constraint of the form

$$\sum_{e=1}^n x_e = \bar{v} \quad (3.18)$$

with \bar{v} the volume of the design;

(ii) Problem (3.16) has a unique solution;

(iii) function f is continuously differentiable and there exists constants $c_1, c_2 < 0$ such that

$$c_1 \leq \frac{\partial f(x_e)}{\partial x_i} \leq c_2 \quad (3.19)$$

for all $x_e \in [0, 1]$ and $e, i = 1, 2, \dots, n$.

Furthermore, Reitz showed convergence properties hold for penalisation functions which exhibit similar characteristics to the SIMP material law. In 2003 Martinez relaxed requirements (ii) and (iii) in Reitz's proof and showed it to hold for any monotonically increasing penalisation function [35]. That is, for severe enough penalisation any solution of (3.17) approaches a solution to (3.16). Martinez also showed that if a solution to (3.17) for a finite p is 'rounded' a solution to (3.16) is obtained. Although these results are promising, proof of convergence only holds for problems with a single constraint. What is more, global solutions are required, which might be very difficult to obtain since the relaxed and penalised problems are non-convex. To avoid convergence to a local minimum it is suggested a continuation approach is employed, where the penalised problems are solved while successively increasing penalisation. However, in [14] Stolpe and Svanberg show the global trajectories of the optimal solution may be discontinuous for continuously increasing penalisation. The global trajectory is defined as the path followed by the global optima of the successively penalised problems. The authors present examples where the global trajectory is discontinuous even though the original discrete problem has an unique solution. Furthermore, an example is presented where penalisation does not produce a purely black-and-white design, no matter how severe the penalisation. The authors conclude that a continuation approach on material penalisation might be a good heuristic, but convergence can not be proved unless some severe requirements/assumptions are satisfied.

To conclude, and in accordance with the results of the convergence studies outlined above, we consider progressive penalisation to be a heuristic procedure to increase black-and-white fractions. Therefore we conduct some numerical experiments in the following section to investigate the effectiveness of progressive penalisation in a SAND setting (along with the other penalisation approaches defined before). That is, how effective is a continuation strategy on material penalisation, facilitated directly via the state equation, to encourage or approach purely black-and-white designs?

3.5.4 Evaluation of penalisation schemes

Here we evaluate the various penalisation approaches presented in the previous section with numerical experiments. We consider problem P_C^W and the two bar truss ground structure with $m = 4$. To start of we present results for various penalisation functions with fixed severity of penalisation after which progressive penalisation is tested.

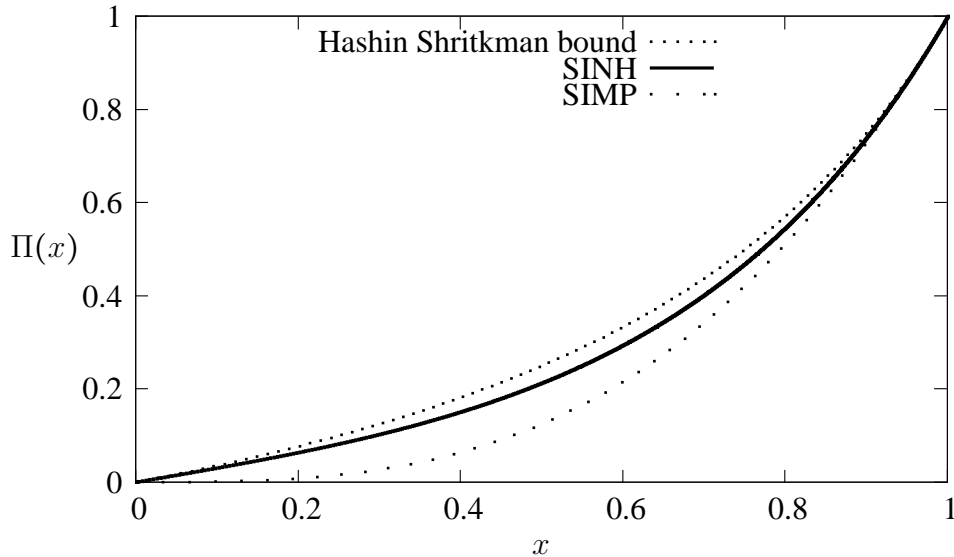


Figure 3.10: A comparison of functions values as a function of x for various penalisation functions.

We consider SIMP, SINH and H-S penalisation functions, as depicted in Figure 3.10, for numerical experiments. As discussed before RAMP penalisation is deemed inappropriate due to the non-zero lower bound requirement on material stiffness (or density). The resulting designs, associated with the three penalisation functions respectively, are depicted graphically in Figure 3.11. Both SIMP and SINH penalisation is formulated with $p = 3$ throughout the investigation of fixed penalisation, the minimum severity of penalisation for which grey material maintains physical meaning. The problems are solved with the SAOi algorithm and the *non-convex* and *analytic* approximations as discussed in section 3.4. Clearly the resulting designs are not identical. SINH penalisation leads to an optimal design where the upper regions of the two members that form the V-shape are

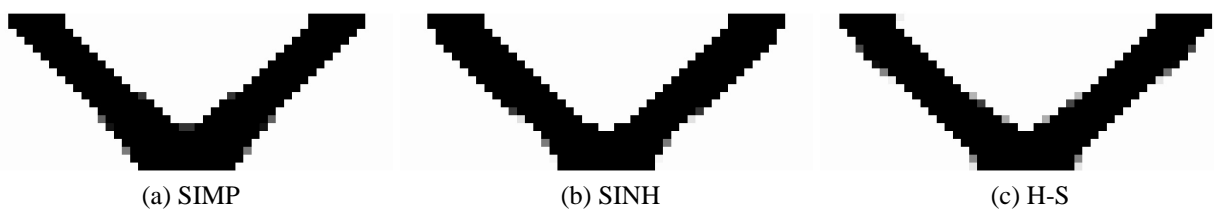


Figure 3.11: A comparison of the topologies generated for the weight minimization of the two-bar truss using various material penalisation functions.

wider, SIMP results in a design where the lower region of the V has more solid elements while H-S penalisation results in fairly uniform members. Based on these observations H-S penalisation performs best, since the optimal design should simply be an uniform and symmetric two bar truss. However, considering the comparison of black-and-white fractions at optimality, summarised in Table 3.3, H-S penalisation performs worst while SIMP performs best, although only slightly better than SINH penalisation. This should be expected, as can be seen from Figure 3.10 SIMP involves the most severe penalisation for all $x \in [0, 1]$, hence resulting in the greatest black-and-white fraction. However, by that rationale we would expect SINH and H-S penalisation to yield similar results in terms of black-and-white fractions. To shed some light on this discrepancy consider the first-order derivatives to x depicted in Figure 3.12. First of all, a clear distinction among the considered functions is that $\frac{d\Pi_{SIMP}}{dx}|_{x=0} = 0$ whereas $\frac{d\Pi_{SINH}}{dx}|_{x=0}, \frac{d\Pi_{H-S}}{dx}|_{x=0} \neq 0$. What is more, both SINH and H-S penalisation actually required a Jacobian filter for the algorithm to converge. We speculate this is due to numerical difficulties. The filter removes all terms from the Jacobian matrix less in magnitude than 1×10^{-4} . We cite numerical difficulties due to the nature of the state equation; a typical term in one of the non-linear equality constraints that compose the state equation would be of the form (neglecting various indices for the sake of brevity)

$$\dots + \Pi(x)q + \dots \quad (3.20)$$

where x denotes a decision variable and q some associated state variable. The first derivative to x results in

$$\dots + \frac{\partial \Pi(x)}{\partial x} q + \dots \quad (3.21)$$

Naturally, as x approaches 0, q might take on a relatively large value to satisfy the constraint. It would seem first-order information becomes unstable if $\frac{d\Pi}{dx}|_{x=0} \neq 0$ due to the multiplication with a non-zero q in $\frac{\partial \Pi(x)}{\partial x} q$. Stability in this context is off-course paramount, keeping in mind our function approximations are based on incomplete Taylor series expansions. What is more, it would seem natural that the polynomial natured SIMP function is more accurately approximated by a Taylor series expansion. However, although this sheds some light on the reasons behind the requirement of a Jacobian filter, it does not explain the different results yielded by SINH and H-S penalisation, especially considering H-S penalisation required substantially more iterations to converge. Going one step further, second-order derivatives to x of the various penalisation functions are displayed in Figure 3.13. Here a subtle but important distinction is noted, namely that the curvature of the SINH and SIMP functions tend to and acquire a value of zero as $x \rightarrow 0$ while the H-S function does not. Therefore, we deduce the reason why H-S penalisation required

Table 3.3: A comparison of the results obtained for the weight minimization of the two-bar truss using various penalisation functions.

Approximation	f_v	$\phi_{B\&W}$	N_{iter}
SIMP	0.250	0.992	29
SINH	0.251	0.993	31
H-S	0.251	0.982	46

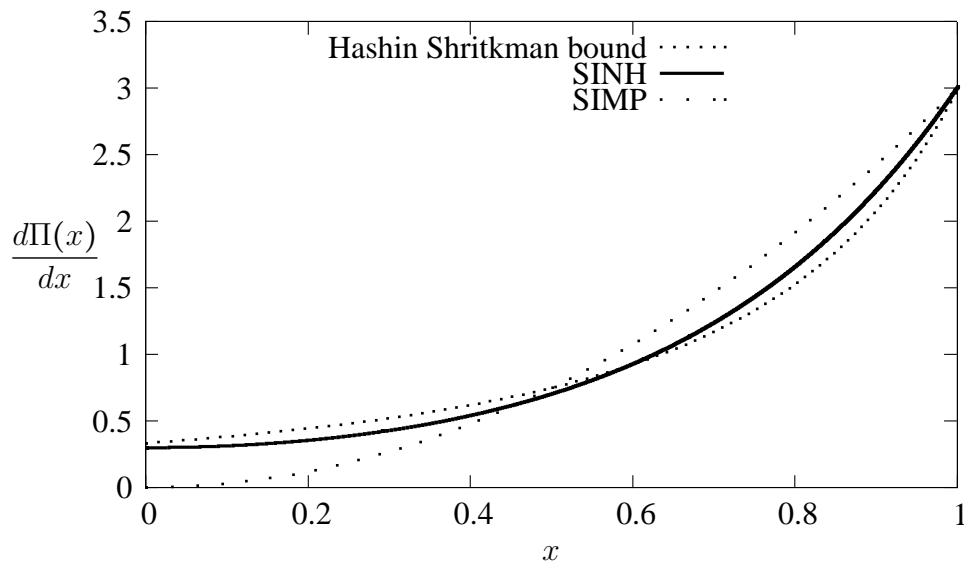


Figure 3.12: A comparison of gradient values as a function of x for various penalisation functions.

more iterations is due to larger curvature terms in the approximate Hessian in each iteration. This results in smaller steps taken by the algorithm, similar to the discussion in section 3.4 on the various approximation strategies.

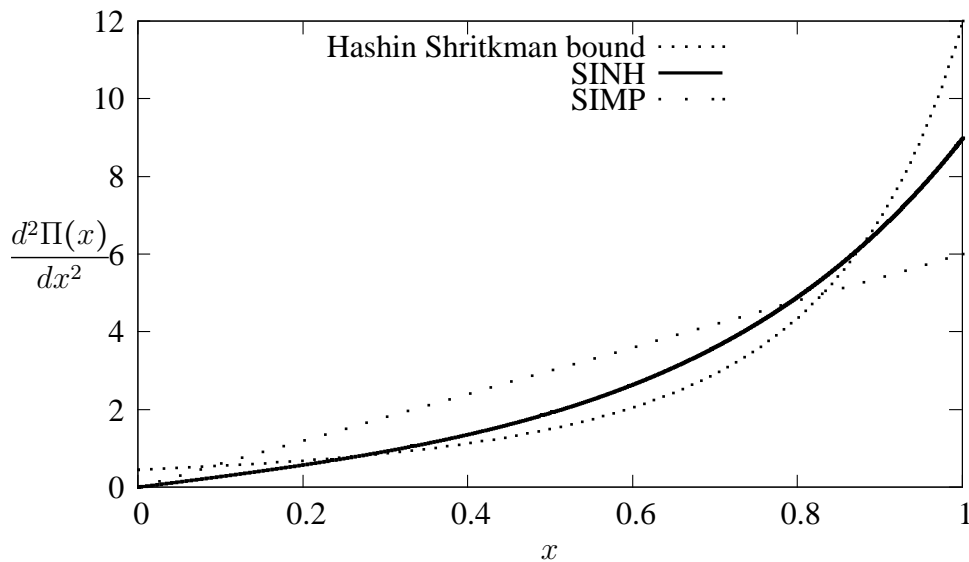


Figure 3.13: A comparison of curvature magnitudes as a function of x for various penalisation functions. .

From these results we maintain the SIMP approach is the most appropriate for the SAND for-

mulated problems solved in the SAO setting we consider. It would seem the SIMP function is approximated more accurately, especially so considering first- and second-order derivatives decrease towards and take on a value of 0 as $x \rightarrow 0$. Although SINH and H-S functions perform fairly well, it would seem numerical difficulties are more severe than with SIMP. Therefore we consider the SIMP approach in investigating the effectiveness of progressive material penalisation. As mentioned previously, we consider progressive penalisation a heuristic procedure to improve black-and-white fractions once an initial solution to the problem is obtained. As before, since at this point we know an appropriate solution is obtained for $p = 3$, the problem is initialised with this quantity. From here we test three cases, depicted in Table 3.4, along with p fixed at 3 to serve as benchmark. Parameter p is increased with $1.1p$ in each iteration whenever the maximum constraint violation is less than 1×10^{-5} . Results are presented for p limited to 6 and 30 respectively.

Clear from the results summarised in Table 3.4 and in conjunction with the results presented in [14], although progressive penalisation increases black-and-white fractions marginally, it does not lead to purely solid-void designs, even with very severe penalisation. A graphical depiction of the results are presented in Figure 3.14. Although a discernible difference is not apparent according to the numerical results, at least graphically the severely penalised design seems more binary of nature. That is, for $p = 30$ the optimal topology generated seems purely solid-void, even though numerically all decision variables do not assume these values exactly. Note that it is merely the so-called ‘grey’ elements at $p = 3$ which are pushed to a value of 1 as penalisation is increased. This seems intuitive, these elements are required to be non-zero at $p = 3$ for the design to be feasible, hence increasing penalisation (thus reducing the contribution of the element to the global stiffness of the structure for fixed x) merely requires these elements to approach values closer to 1 for the design to remain feasible. Furthermore, this is in line with the proof given by Martinez in [35] that a solution to the relaxed and penalised problem when ‘rounded’ would be a solution to the discrete problem. In Figure 3.15 the topology is presented when the solution for $p = 3$ is simply rounded to 0 and 1. Clearly the result is exactly the same as that obtained by applying more severe penalisation, confirming the statement in [35]. Interestingly, the ‘rounded’ topology is completely feasible, however, the compliance constraint is inactive with compliance of the structure at 198,552 (the compliance of the structure is limited to 200). That is, the ‘rounded’ design is slightly conservative and suboptimal compared to the design obtained for $p = 3$. It would seem this is the reason why purely solid-void designs are not obtained even with very severe penalisation.

Finally, to illustrate the asymptotic behaviour we observe from progressive penalisation in increasing black-and-white fractions, a convergence plot of both $\phi_{B\&W}$ and p are presented in Figure 3.16. Clearly, $\phi_{B\&W}$ approaches its upper value even for $p = 3$, after which the number of elements at

Table 3.4: A comparison of the results obtained for the weight minimization of the two-bar truss with various levels of progressive penalisation.

Approximation	f_v	$\phi_{B\&W}$	$n_{[0]}$	$n_{[1]}$	N_{iter}
$p^* = 3$	0.250	0.992	898	292	29
$p^* = 6$	0.251	0.995	900	294	81
$p^* = 30$	0.251	0.995	900	294	145

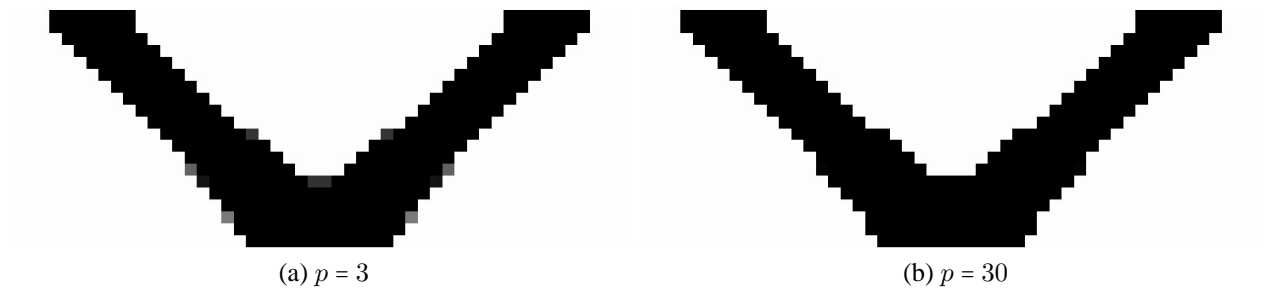


Figure 3.14: A comparison of the topologies generated for the weight minimization of the two-bar truss using various levels of progressive penalisation.

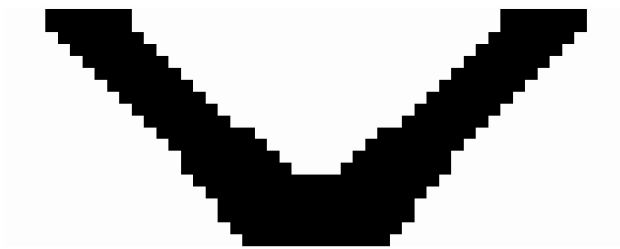


Figure 3.15: The generated topology for the weight minimization of the two-bar truss for $p = 3$ when 'rounded'.

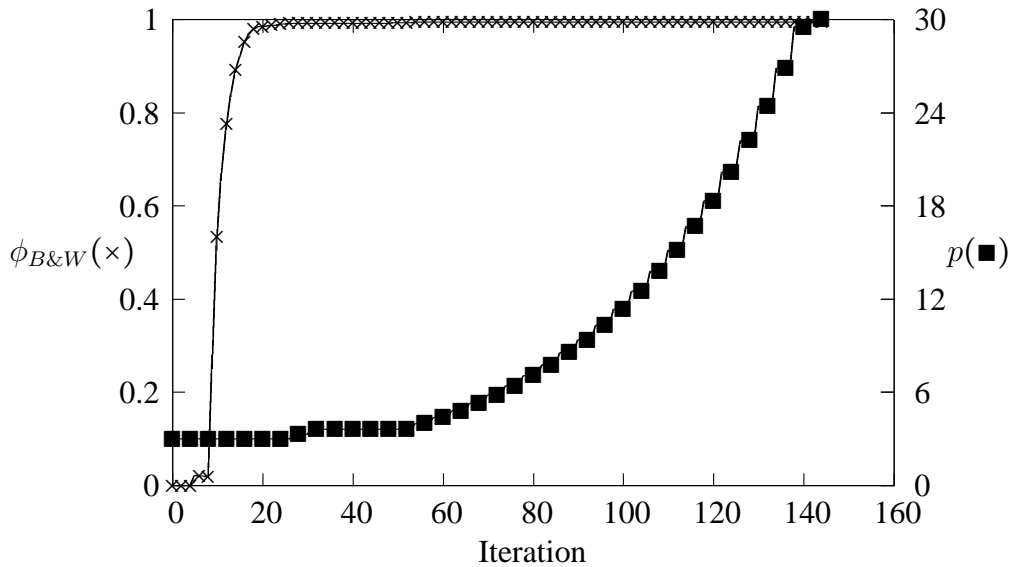


Figure 3.16: Convergence plot of $\phi_{B\&W}$ and p for the weight minimisation of the two-bar truss ground structure subject to a constraint on compliance.

exactly 0 or 1 increases marginally as p is increased to 30.

3.6 The selection of standard settings

Up until this point we have been concerned with determining the best approximation and penalisation strategies from a number of options (along with illustrating some properties related to a zero lower bound on decision variables). From the results presented in the previous sections we choose the *analytic* and *non-convex* approximation strategy along with the SIMP material law with p fixed at 3 as general solution strategy to solve problems P_C^W and P_W^C . The *analytic* and *non-convex* approximation strategy is chosen due to computational performance and superior solution quality. Although SINH and SIMP penalisation can not be separated on solution quality, our experiments show SINH penalisation to be numerically more unstable (requiring a Jacobian filter to converge). Therefore traditional SIMP penalisation is employed where both first- and second-order derivatives to decision variables disappear as material is removed from the domain. This property is not only beneficial in computational requirements but it would seem SIMP is numerically better behaved for this reason. Furthermore, we showed progressive material penalisation increases black-and-white fractions marginally and the optimum can simply be rounded to 0 and 1 to obtain a solution to the discrete problem. Therefore we employ fixed penalisation with SIMP parameter $p = 3$. Finally, we emphasise material is allowed to be removed completely from the design domain with a zero lower bound on decision variables.

The SAOi algorithm is employed with the IBM® ILOG® CPLEX® BARRIER algorithm as sub-problem solver. All computation runs were done on a desktop PC with the following specifications:

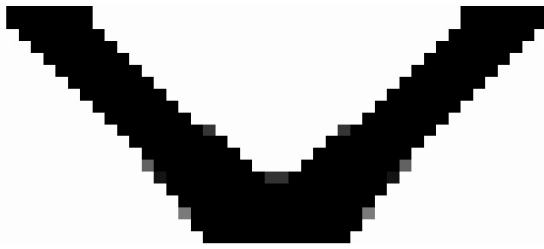
Processor:	Intel(R) Xeon(TM) 8 core CPU 3.73GHz
Memory:	Total memory (RAM), 31.5 GiB
Operating system:	Linux 2.6.34-12-desktop x86-64, openSUSE 11.3 (x86-64)

We consider two test problems in the form of the 2-bar truss and MBB beam as defined in section 2.2.4. All problems are initialised with state variables at 0. Problem P_C^W is initialised with all decision variables at 1 while Problem P_W^C is initialised with decision variables at 0.25 for the two bar truss and 0.35 for the MBB beam (such that the volume constraint is feasible in the first iteration). For the two bar truss and MBB beam \bar{v} is set at 0.25 and 0.35 respectively. Furthermore, the constraint on compliance (\bar{c}) is set at 200 and 250 for the two bar truss and MBB beam respectively. The algorithm is deemed to have converged once the Euclidean norm of the update in all design variables is less than 1×10^{-2} .

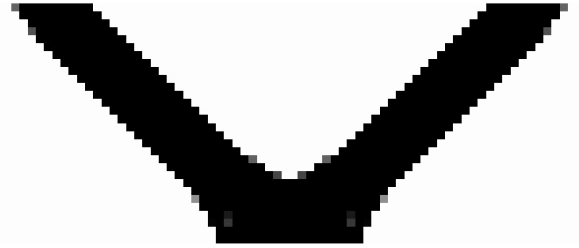
3.6.1 Optimal designs for the two-bar truss

Here results are presented for the minimum weight and compliance of the two-bar truss test problems (Figures 3.17a-d and Figures 3.18a-d). That is problems P_C^W and P_W^C with the two-bar truss ground structure as design domain. We consider four mesh discretisations, denoted by mesh multiplier m . Furthermore, problem size is summarised by the number of decision variables (n_d), state variables (n_s), equality constraints (n_e) and inequality constraints (n_i). The volume fraction at optimality is denoted by f_v , the number of decision variables that obtain a value of zero by $n_{[0]}$, number of decision variables at 1 by $n_{[1]}$, number of decision variables at intermediate values by $n_{[i]}$ and black-and-white fractions by $\phi_{B\&W}$. Finally, the number of iterations required is denoted by N_{iter} and the average CPU time required per iteration by T_{avg}^I .

Weight minimization of the two-bar truss



(a) Result for $m = 4$, $n_d = 1200$, $n_s = 7522$, $n_e = 7522$, $n_i = 1$: $f_v = 0.250$, $n_{[0]} = 898$, $n_{[1]} = 292$, $n_{[i]} = 10$, $\phi_{B\&W} = 0.992$, $T_{\mu}^I = 0.893$, $N_{iter} = 29$



(b) Result for $m = 6$, $n_d = 2700$, $n_s = 16682$, $n_e = 16682$, $n_i = 1$: $f_v = 0.243$, $n_{[0]} = 2042$, $n_{[1]} = 642$, $n_{[i]} = 16$, $\phi_{B\&W} = 0.994$, $T_{\mu}^I = 1.942$, $N_{iter} = 40$



(c) Result for $m = 10$, $n_d = 7500$, $n_s = 45802$, $n_e = 45802$, $n_i = 1$: $f_v = 0.239$, $n_{[0]} = 5702$, $n_{[1]} = 1756$, $n_{[i]} = 42$, $\phi_{B\&W} = 0.994$, $T_{\mu}^I = 6.573$, $N_{iter} = 65$



(d) Result for $m = 14$, $n_d = 14700$, $n_s = 89322$, $n_e = 89322$, $n_i = 1$: $f_v = 0.237$, $n_{[0]} = 11204$, $n_{[1]} = 3418$, $n_{[i]} = 78$, $\phi_{B\&W} = 0.995$, $T_{\mu}^I = 15.995$, $N_{iter} = 42$

Figure 3.17: Topologies generated by weight minimisation of the two-bar truss ground structure subject to a constraint on compliance.

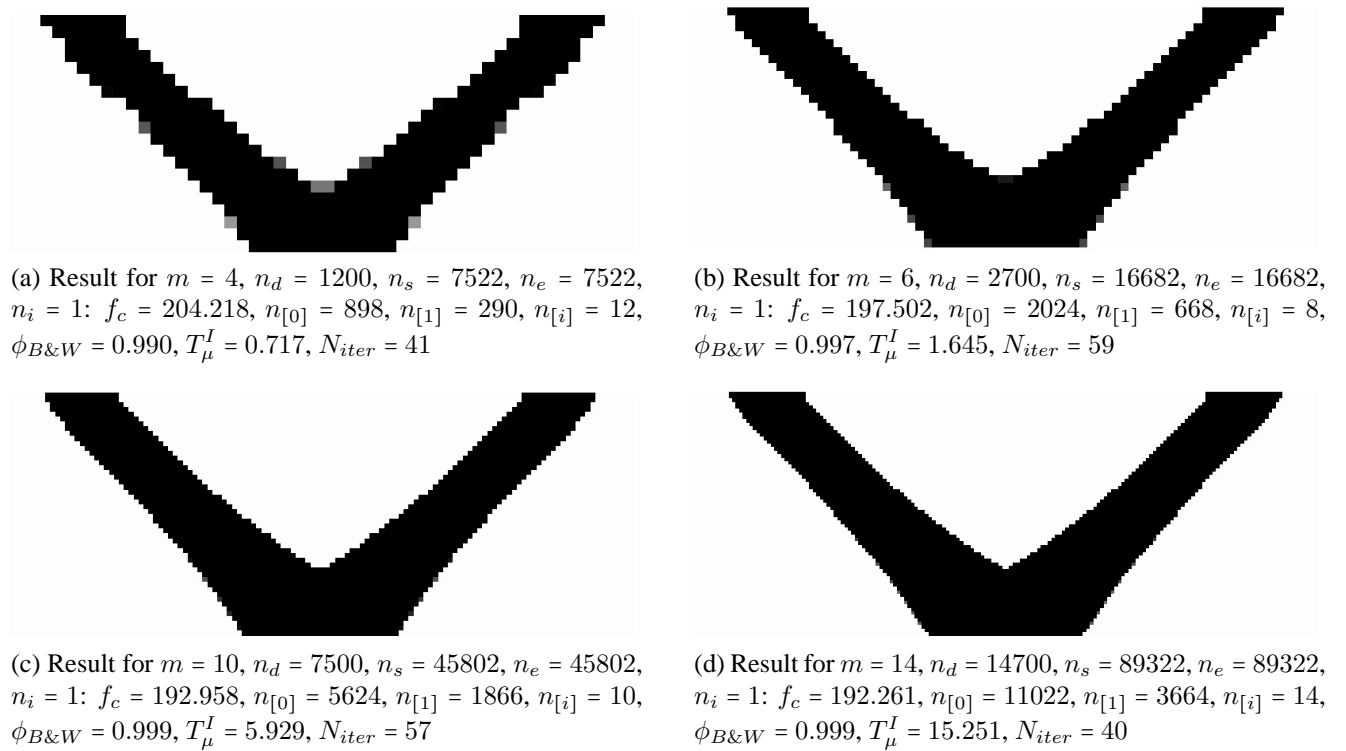
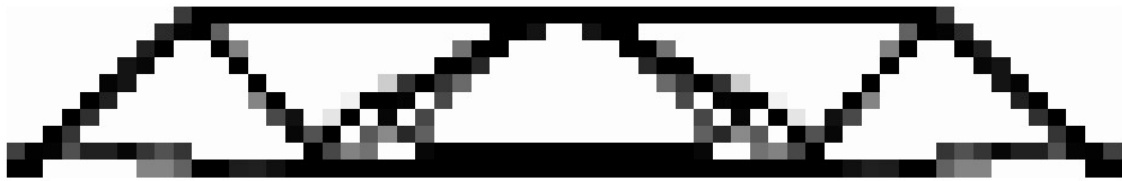
Minimum compliance of the two-bar truss

Figure 3.18: Topologies generated by compliance minimisation of the two-bar truss ground structure subject to a constraint on volume.

3.6.2 Optimal designs of the MBB beam

Here results are presented for the minimum weight and compliance of the MBB test problem (Figures 3.19a-d and Figures 3.20a-d). Hence problems P_C^W and P_W^C with the MBB ground structure as design domain.

Weight minimization of the MBB beam



(a) Result for $m = 2$, $n_d = 300$, $n_s = 1962$, $n_e = 1962$, $n_i = 1$: $f_v = 0.365$, $n_{[0]} = 181$, $n_{[1]} = 66$, $n_{[i]} = 53$, $\phi_{B\&W} = 0.823$, $T_\mu^I = 0.354$, $N_{iter} = 218$



(b) Result for $m = 4$, $n_d = 1200$, $n_s = 7522$, $n_e = 7522$, $n_i = 1$: $f_v = 0.327$, $n_{[0]} = 796$, $n_{[1]} = 334$, $n_{[i]} = 70$, $\phi_{B\&W} = 0.942$, $T_\mu^I = 1.949$, $N_{iter} = 255$

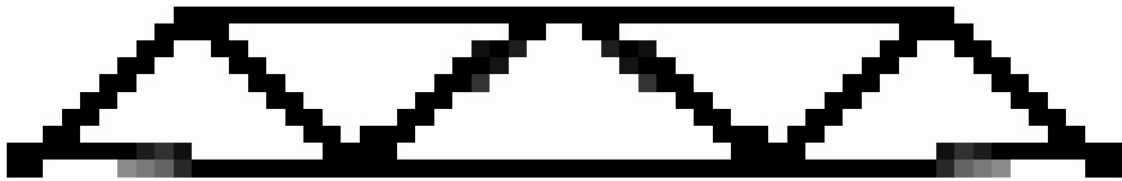


(c) Result for $m = 6$, $n_d = 2700$, $n_s = 16682$, $n_e = 16682$, $n_i = 1$: $f_v = 0.377$, $n_{[0]} = 1850$, $n_{[1]} = 753$, $n_{[i]} = 97$, $\phi_{B\&W} = 0.964$, $T_\mu^I = 5.490$, $N_{iter} = 209$

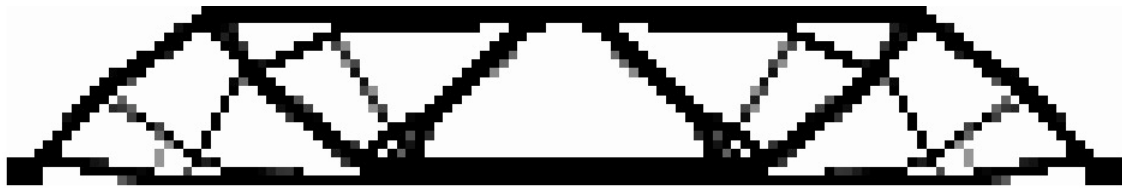


(d) Result for $m = 10$, $n_d = 7500$, $n_s = 45802$, $n_e = 45802$, $n_i = 1$: $f_v = 0.365$, $n_{[0]} = 5264$, $n_{[1]} = 2081$, $n_{[i]} = 155$, $\phi_{B\&W} = 0.979$, $T_\mu^I = 56.553$, $N_{iter} = 277$

Figure 3.19: Topologies generated by weight minimisation of the MBB beam ground structure subject to a constraint on compliance.

Minimum compliance of the MBB beam

(a) Result for $m = 2$, $n_d = 300$, $n_s = 1962$, $n_e = 1962$, $n_i = 1$: $f_c = 260.717$, $n_{[0]} = 193$, $n_{[1]} = 95$, $n_{[i]} = 12$, $\phi_{B\&W} = 0.960$, $T_\mu^I = 0.279$, $N_{iter} = 145$



(b) Result for $m = 4$, $n_d = 1200$, $n_s = 7522$, $n_e = 7522$, $n_i = 1$: $f_c = 235.551$, $n_{[0]} = 768$, $n_{[1]} = 367$, $n_{[i]} = 65$, $\phi_{B\&W} = 0.946$, $T_\mu^I = 1.187$, $N_{iter} = 177$



(c) Result for $m = 6$, $n_d = 2700$, $n_s = 16682$, $n_e = 16682$, $n_i = 1$: $f_c = 222.514$, $n_{[0]} = 1746$, $n_{[1]} = 889$, $n_{[i]} = 65$, $\phi_{B\&W} = 0.976$, $T_\mu^I = 2.767$, $N_{iter} = 196$



(d) Result for $m = 10$, $n_d = 7500$, $n_s = 45802$, $n_e = 45802$, $n_i = 1$: $f_c = 216.914$, $n_{[0]} = 4860$, $n_{[1]} = 2542$, $n_{[i]} = 98$, $\phi_{B\&W} = 0.987$, $T_\mu^I = 9.661$, $N_{iter} = 200$

Figure 3.20: Topologies generated by compliance minimisation of the MBB beam structure subject to a constraint on volume.

3.7 Discussion and conclusion

The simultaneous analysis and design (SAND) formulation of the topology optimisation problem allows the direct penalisation of material density/thickness in the state equation. We show that a number of fictitious material laws can be applied to drive the solution of the relaxed and continuous problem to that of the discrete problem. However, the traditional simple isotropic material with penalisation (SIMP) approach showed the most favourable characteristics.

The SAND formulation allows the complete removal of material with a zero lower bound on decision variables. Not only is the latter an exact representation of void material but also allows for a reduction in computational requirements. In the light of this advantage associated with the SAND formulation of the problem the SIMP approach would seem to be the most appropriate; first- and second-order sensitivities vanish from the problem along with material. However, the SAND problem is inherently non-convex, that is we can only hope to converge to the appropriate local optima, with material unable to re-enter the design domain once removed.

A brief investigation is conducted into progressive penalisation of the state equation. Although heuristic based, convergence can only be proved under severe assumptions, we show progressive penalisation can improve black-and-white fractions marginally. Although this procedure seems excessive considering the quality of solutions obtained with fixed penalisation.

The topologies generated for various sizes of the two-bar truss and MBB beam ground structures presented in sections 3.6.1 and 3.6.2 show very good solution quality. That is, all the designs approach a state of purely solid-and-void, with very few elements not exactly at 1 or 0 at the optimal solution. We emphasise that the elements deemed void can assume a value of exactly zero on the lower bound due to the removal of an explicit structural analysis phase in the SAND problem.

From the results presented in sections 3.6.1 and 3.6.2, and specifically that of the MBB beam ground structure, we see that the algorithm converges to different local optima for different mesh refinements. This indicates the need of a filter to ensure a consistent solution is obtained for different mesh discretisations. In a NAND setting, filters generally limit the sensitivities of the objective and constraint functions to decision variables, see for example [8]. Although heuristic, filters have been employed in this setting with great success. However, it is unclear and open for future work in how a filter can be applied to a SAND formulated problem. Should the sensitivities of the state equation be limited? Or should the filter be unified with the fictitious material interpolation scheme? We also speculate that a heuristic procedure can be incorporated in the construction of the approximate Hessian of the problem to act as some pseudo-filter.

Chapter 4

Local stress constraints, the singularity problem and simultaneous analysis and design

4.1 Introduction

Structural optimisation is concerned with finding the best possible structure for a given scenario or set of conditions. These conditions or constraints, define the set of feasible designs within which the optimal design is sought. Material failure is often of concern in structural design and the set of feasible structures should be defined accordingly. If not, the optimal design may not be fit for purpose, or worse, the structure might fail catastrophically. In light of this imposing material failure or *stress* constraints on a design is a crucial aspect of structural optimisation. However, material failure is a local phenomenon that can, in principle, occur at any spatial position in the structure. Therefore, unlike other characteristics associated with structural design such as compliance (a global property that can be described by a single constraint) limiting structural stress leads to very large scale problems with a large number of constraints. Despite the computational complexity related to problem size and local constraints, as discussed in Chapter 2, the stress constrained problem suffers from the well documented *singularity problem* [18]. Simply put, stress, a measure of force per unit area, is undefined if material assumes a state of void.

In Chapter 2 we showed that the simultaneous analysis and design (SAND) formulation of the structural optimisation problem reduces computational requirements related to local constraints. In addition, the SAND formulation of the stress constrained topology optimisation problem poses a number of subtle differences in formulation and interpretation of local stress constraints. Firstly, as one would expect, the stress state of some point in a structure is an explicit function of the state variables. Therefore, in a SAND formulated problem, with decision and state variables independent, stress constraints remain in this simple explicit functional form to state variables. In contrast a nested analysis and design (NAND) formulated problem involves exact structural analysis in each iteration and state variables are effectively removed from the optimisation problem by a procedure equivalent to direct substitution. In the latter setting dependence on state variables is transformed to a more complicated implicit reciprocal relation to decision variables. Secondly,

throughout this thesis great emphasis has been placed on the fact that the SAND formulated topology optimisation problem allows a value of exactly zero on the lower bound of decision variables, with little or no complications. In other words, material can be removed completely from the design domain and void material is represented exactly. Considering the stress constrained problem in SAND and the fact that material can assume a state of absolute void is even more interesting. As mentioned before, the stress constrained problem suffers from the singularity problem, a mathematical phenomena related to the fact that stress becomes undefined if material attains a state of void. In light of this we wish to investigate the singularity problem in a SAND setting and specifically the formulation of stress constraints as so-called ‘vanishing’ constraints [29].

The stress constrained singular optimum in structural topology optimisation was first demonstrated, in 1968, by Sved and Ginos [36]. Such is the complexity of the stress singularity problem that Sigmund and Bendsoe, arguably some of the most prevalent authors in topology optimisation literature, state that “*the best way to solve stress constrained problems has probably yet to be suggested*” [8]. The theoretical aspects of the singularity problem is typically studied in the context of simple truss structures [18][9][37]. Solution methods for SAND formulated problems are presented in [38] and [27] and solved by minimizing a penalty function with a conjugate gradient method. However, these problems are *sizing* or *free material* optimisation problems with very few examples of the stress constrained topology optimisation problem in literature. An exception is found in [23] where a penalised sizing problem is considered in the form of the homogenization approach in a NAND setting with stiffness penalised as per the SIMP strategy. Stress constraints are relaxed using an ϵ -relaxation approach such that the SAO algorithm, based on a dual solver, can approach the singular solution. In a NAND setting the singular optima is not actually part of the feasible domain and the authors in [23] state the lower bound on decision variables is an integral part of the relaxation scheme. NAND formulated large scale problems are solved in [10] with a similar algorithm. In reference to stress constraints and ϵ -relaxation the author states “To complicate matters, it is usually necessary to set a finite lower bound on an element cross-section or density to prevent numerical ill-conditioning in the analysis of the structure”. Therefore the role of a non-zero lower bound on decision variables in the stress constrained problem seems unclear and problematic. We wish to circumnavigate these difficulties by allowing a value of exactly zero on the lower bound of decision variables, i.e. an exact representation of void material.

This chapter is structured as follows. In section 4.2 the local stress constrained topology optimisation problem is formulated. A discussion of the relevant aspects of the SAO solution procedure is presented in section 4.3, followed by a discussion and comparison of the singularity problem in NAND and SAND. In section 4.5 stress constraints are formulated in an attempt to remedy the singularity problem. Some aspects of large scale problems, computational requirements and numerical difficulties are highlighted in section 4.6. Section 4.7 concludes the chapter.

4.2 Problem formulation

We consider the continuum topology optimisation problem with local stress constraints formulated as per the finite element (FE) method. For the sake of brevity only the case of weight minimization subject to local stress constraints is considered. Departing from Problem P_C^W the constraint on

compliance is removed¹ and a local stress constraint is applied to every element in the FE mesh, resulting in problem P_S^W :

$$\begin{aligned} \min_{\mathbf{x}, \mathbf{q}} \quad & \sum_{e=1}^n x_e \\ \text{subject to} \quad & g_i(\mathbf{q}_{elmi}) \leq 0, \\ & \mathbf{K}(\Pi(x_i))\mathbf{q} = \mathbf{f}, \\ & 0 \leq x_i \leq 1 \quad i = 1, 2, \dots, n \end{aligned} \quad (4.1)$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{q} \in \mathbb{R}^u$. The number of elements in the FE mesh and thus the number decision variables is denoted by n and the number of state variables by u . The vector \mathbf{f} denotes nodal forces (assumed to be design independent) and only Q8 elements are considered in the FE discretisation.

In keeping with Chapter 3 we employ the SIMP approach [31] with decision variables penalised with $\Pi(x_i) = x_i^p$ and $p = 3$ in the state equation. The value of p is chosen such that the artificial material law satisfies the Hashin-Shtrikman bound [8] which allows physical interpretation of decision variables at intermediate values. However, as mentioned before, to obtain solutions that approach purely solid and void designs is of primary concern.

In Appendix B the exact form of constraint $g_i(\mathbf{q}_{elmi}) \leq 0$ is derived via the FE method under plane stress.

4.3 Sequential approximate optimisation

To solve the stress constrained problem we employ the SAO algorithm, as throughout this thesis, referred to as SAOi (refer to Chapter 2) for more details. Here we wish to highlight some properties of stress constraints in this setting, with details on approximating the state equation in each subproblem discussed in Chapter 3.

Stress constraints limit the structural stress in each element in the form of an inequality constraint. Unlike equality constraints, concepts like convexity and conservatism are applicable, as discussed in Chapter 2. However, with the success of the so-called *non-convex* strategy to ensure the approximate Hessian of the Lagrangian is positive definite, discussed in Chapter 3, curvature information of stress constraints is simply included, along with the state equation, in the approximate Hessian of the Lagrangian with

$$\bar{Q}_{ii}^k = \max\{\epsilon_h > 0, \sum_{j=1}^{u+n} \lambda_j^{\{k\}} c_{2i_j}^{\{k\}}\}, \quad (4.2)$$

where u denotes the number of equality constraints and n the number of elements in the mesh, hence the number stress or inequality constraints. Therefore positive definiteness of the Hessian of the Lagrangian is enforced by requiring that each diagonal term, after all constraint curvatures are summed, be strictly positive. We employ a value of $\epsilon_h = 1 \times 10^{-6}$ throughout.

¹Numerical experiments showed that the final solution would typically be defined by either an active compliance constraint or active stress constraints, but not both. Therefore the singularity problem cannot reliably be isolated and investigated if the compliance of the structure is also constrained. Of-course, in a practical setting a constraint on compliance can and should be included if *a priori* information of that nature is available.

As mentioned before, in SAND simple analytic expressions are available to calculate exact second-order curvatures of constraint functions, referred to as the *analytic* quadratic approximation in Chapter 3. This is no different when considering stress constraints. What is more, in SAND, with stress constraints an explicit function of state variables, all diagonal curvatures are positive and constant with respect to state variables². In other words, considering SAND and SAOi, stress constraints are convex in state variables and diagonal terms are approximated exactly with a quadratic Taylor series expansion. However the Hessian of the Lagrangian is approximate because the Lagrange multipliers from the previous iteration are employed and only diagonal terms are considered.

Let $\mathbf{q}_{elmi} = [u_1, v_1, u_2, v_2, \dots, u_8, v_8]$, where u and v denote horizontal and vertical local displacement or local state variables such that, for example, $\mathbf{q}_{elmi}(1) = u_1$ and $\mathbf{q}_{elmi}(4) = v_2$. It can be shown, after some calculus, that the diagonal Hessian terms of a stress constraint (B.16) is given by two expressions:

$$\begin{aligned} \frac{\partial^2 g_i}{\partial \mathbf{q}_{elmi}(j)^2} &= \frac{3E^2}{2L^2\sigma_y^2(1+\nu^2)^2} \text{ for } j = 9, 12, 13, 16, \\ &= \frac{2E^2(1-\nu+\nu^2)}{L^2\sigma_y^2(1-\nu^2)^2} \text{ for } j = 10, 11, 14, 15, \\ &= 0 \text{ otherwise,} \end{aligned} \quad (4.3)$$

where L is the length of an element in the FE discretisation. Clearly, the non-zero diagonal Hessian terms are all positive and constant (material properties E, ν, σ_y are constant).

4.4 The singularity problem

In addition to computational resource requirements, stress constrained problems suffer from the so-called *singularity* phenomena. Sved and Ginos showed that stress constraints related to a void element should not contribute to the feasibility of the design [36]. From a physical point of view *stress*, a measure of force per unit area, is undefined for material that assumes a state of void. Sved en Ginos considered a simple truss structure (the same behaviour is exhibited for continuum problems solved via the material distribution method) and showed that redundant elements, with zero cross-sectional areas, experienced non-zero stresses resulting in the optimum solution being infeasible.

To illustrate the singularity problem in NAND consider the simple structure depicted in Figure 4.1. Bar areas are denoted by A_i , internal forces by P_i and applied force by F . Structural analysis involves the calculation of the free end displacement q , given by

$$q = \frac{FL}{E(A_1 + A_2)}. \quad (4.4)$$

Internal loads P_1, P_2 in each bar are calculated with

$$P_i = \frac{qA_iE}{L} \quad (4.5)$$

²Keeping in mind stress constraints are squared in equation (B.16) to ease the calculation of first- and second-order derivatives.

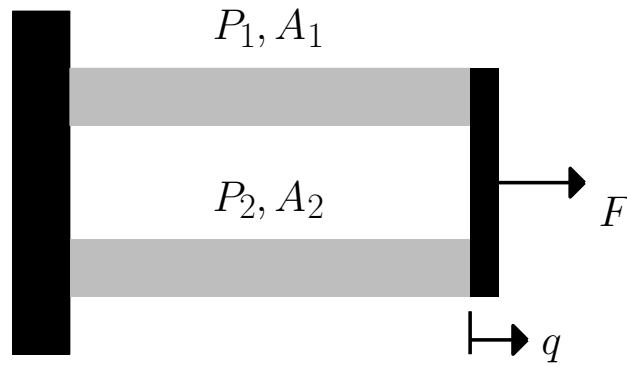


Figure 4.1: A 1-D structure illustrating the singularity problem.

and the resulting stresses in bar 1 and 2 are given by

$$\sigma_i = \frac{F}{A_1 + A_2} \quad (4.6)$$

Note that the calculation of stress as per equation (4.6) is effectively a NAND formulation; state variable q is removed from the expression by direct substitution. Furthermore, consider the case where area A_2 is kept constant and area A_1 is reduced to zero. Along with A_1 internal force P_1 would tend to zero with the stress in element 1 assuming a finite value. Naturally this is problematic, for how can a void element be stressed in any way? The same behaviour is observed for more complex systems and inhibits the removal of elements which approach a state of void. In the thorough discussion on the singularity problem presented in [4] this is referred to as the *limit value problem*. To show this, consider the stress constraint imposed on element 1 in the reciprocal NAND formulation considered above, with the following change in notation³ $x_1 = A_1$ and $x_2 = A_2$,

$$\frac{c}{x_1 + x_2} - 1 \leq 0. \quad (4.7)$$

In general constant c is a function of applied force, material and geometric properties. Consider the case of $c > 0$, the converse case of $c \leq 0$ being trivial. The constraint is rewritten as

$$(c - x_1 - x_2) \leq 0, \quad (4.8)$$

from which it is clear the feasible domain is defined by a line in the $x_2 - x_1$ plane, depicted in Figure 4.2a. Clearly, for c positive, the point $x_1 = 0$ and $x_2 = 0$ is not part of the feasible domain. Therefore the stress constraint on element 1 does not allow either element to attain a state of void, although a design with one of the bars removed, that is, $x_1 = 0$ or $x_2 = 0$, should be feasible in principle. To rectify this, the constraint is reformulated to represent a limit on force as opposed to stress. Considering equation (4.5) it is clear that forces tend to zero as material tends to void. The constraint is reformulated as

$$x_1(c - x_1 - x_2) \leq 0. \quad (4.9)$$

Note bar areas are positive, that is $x_i \geq 0$. Therefore multiplication by x_1 does not change the feasible domain inasmuch as it facilitates the ‘removal’ of the stress constraint when material

³The change in notation is made in the light of consistency, for material or decision variables are typically denoted, as throughout this thesis, with variable x .

approaches a state of void. The feasible region defined by constraint (4.9) on element 1 is depicted alongside the original formulation in Figure 4.2b with solid lines indicating the inclusive boundary of the feasible region (the interior of the feasible region is depicted in grey).

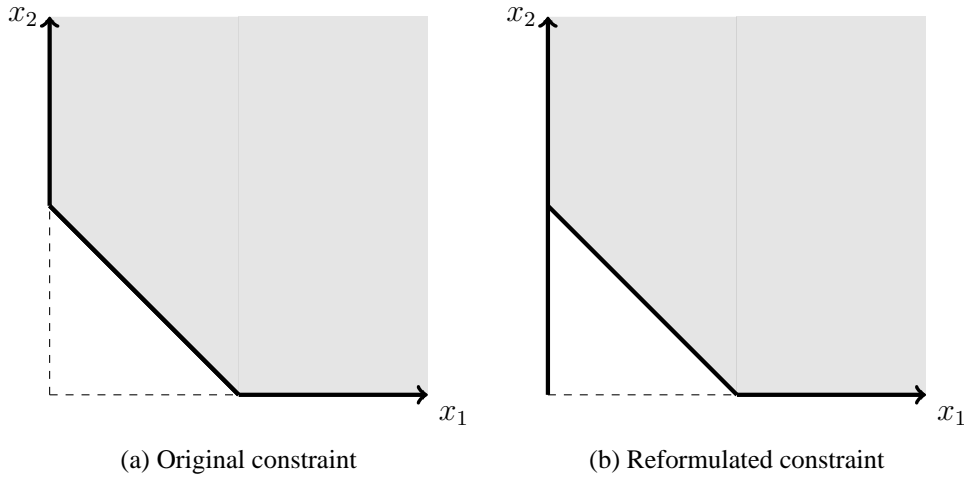


Figure 4.2: The feasible region defined by a stress constraint on element 1.

The reformulated constraint defines a domain where the line $x_1 = 0$ is feasible. What is more, for $x_1 = 0$ any value for x_2 is feasible with respect to the stress constraint on element 1, referred to as singular solution in a strict mathematical sense⁴. Note that the line is an one-dimensional object in two-dimensional space. That is, the feasible domain is degenerate⁵ for $x_1 = 0$ and standard constraint qualification does not hold. Simply put, standard constraint qualification (or the regularity conditions) gives the precise conditions for optimality, whereas the Karush-Kuhn-Tucker (KKT) conditions represent necessary conditions. In general constraint qualification holds if the Jacobian vector of active constraints is linearly independent [19]. To illustrate the problem with constraint qualification consider both stress constraints on elements 1 and 2

$$x_1(c - x_1 - x_2) \leq 0, \quad (4.10)$$

$$x_2(c - x_1 - x_2) \leq 0. \quad (4.11)$$

The constraint Jacobian $\nabla g(\mathbf{x})$ is written in matrix form with row 1 corresponding to the stress constraint on element 1, row 2 that of constraint 2, column 1 the derivative to x_1 and column 2 the derivative to x_2 , i.e.

$$\begin{bmatrix} c - 2x_1 - x_2 & -x_1 \\ -x_2 & c - x_1 - 2x_2 \end{bmatrix}. \quad (4.12)$$

Assume structural weight is minimised and that some lower bound is applied to element 2 such

⁴Note that this is in distinction to the so-called *singularity* problem, which refers to a range of phenomena specifically related to stress constraints. In a strict mathematical sense a singular solution is defined as a point or region where multiple solutions hold.

⁵In general the optimum solution might be located in a region of k -dimensional space where $k \ll n$, with n the overall dimensionality of the problem.

that element 1 is void at the optimal solution. Substitution of $x_1 = 0$ in equations (4.12) results in

$$\begin{bmatrix} c - x_2 & 0 \\ -x_2 & c - 2x_2 \end{bmatrix}. \quad (4.13)$$

Substitution of $x_1 = 0$ in equation (4.11) and considering that the optimal solution will be such that the remaining element is fully stressed, the constraint on element 2 reduces to

$$x_2(c - x_2) = 0. \quad (4.14)$$

Naturally, design variables are restricted to positive values, therefore $c - x_2 = 0$. Substitution in equation (4.13) results in

$$\begin{bmatrix} 0 & 0 \\ -x_2 & -x_2 \end{bmatrix}, \quad (4.15)$$

which is linearly dependent, since the set contains the zero vector. A set $(\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_k)$ of k vectors is *linearly dependent* if there exists scalars c_1, c_2, \dots, c_k , not all zero, such that

$$\mathbf{0} = c_1\mathbf{v}_1 + c_2\mathbf{v}_2 + \dots + c_k\mathbf{v}_k, \quad (4.16)$$

where $\mathbf{0}$ denotes the zero vector. Clearly the vectors are linearly dependent if the set contains the zero vector. For suppose $\mathbf{v}_1 = \mathbf{0}$, we just choose c_1 non-zero with all other scalars zero to satisfy equation (4.16). Stress constraints, as formulated above, are referred to as *vanishing* constraints in a specific species of optimisation problem called mathematical programs with vanishing constraints (MPVC's). The problem with constraint qualification is relieved by modifying standard constraint qualification such that constraints related to elements that attain a state of void is considered inactive. See [29] for a more detailed description. It should be noted, the SAO algorithm we employ, with all subproblems convex, ensures constraint qualification always holds (on a subproblem level) if a feasible solution exists according to Slater's condition [20].

Due to the degeneracy of the design space classical optimisation algorithms based on the KKT conditions are unable to reach the optima located in these regions [8]. That is, standard optimisation algorithms are unable to completely remove some material that approaches a state of void. Graphically the algorithm can not 'enter' the region defined by the line $x_1 = 0$. A common approach to circumvent this complication is to relax the stress constraints by 'opening up' the design space. Referring to Slater's condition, perturbing the design space in this way ensures that a feasible solution exists. Then as the optimal solution is approached relaxation is reduced until the original problem is solved (referred to as a continuation strategy). The most common approach of this nature is referred to as ϵ -relaxation [39]. The original problem is perturbed by applying a constant relaxation of value ϵ to stress constraints

$$x_1(c - x_1 - x_2) \leq \epsilon, \quad (4.17)$$

$$x_2(c - x_1 - x_2) \leq \epsilon. \quad (4.18)$$

The modified feasible domains for a single constraint and two values of ϵ are presented in Figure 4.5 (the shaded region depicts the interior of the feasible domain). Clearly for $\epsilon > 0$ the feasible domain is opened-up. That is, the feasible region no longer contains a degenerate domain and the optimal solution can be approached by a classical optimisation algorithm.

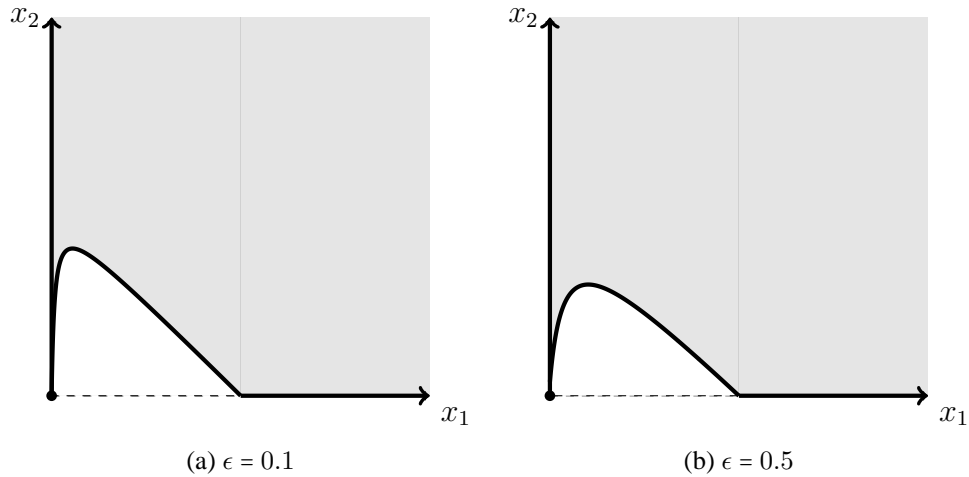


Figure 4.3: The feasible region defined by the relaxed constraint on element 1.

Typically an optimisation algorithm is initialised with ϵ set to some small non-zero value. Based on constraint violation or convergence measures ϵ is reduced to zero as the algorithm approaches the optimal solution. Although, keep in mind that the true singular solution can not be reached in a NAND setting due to a non-zero lower bound on decision variables. In other words, in NAND the line $x_1 = 0$ is not actually part of the feasible domain. That is, although standard constraint qualification does not hold, this does not prevent the algorithm from reaching the so-called degenerate region, it is the lower bound on decision variables that excludes the degenerate region from the feasible domain. Here we make a swift transition to the SAND setting where the said restriction is eliminated.

Reverting to the simple truss structure depicted in Figure 4.1, in a SAND setting structural analysis is performed by satisfying the constraint

$$E(A_1 + A_2)q = FL. \quad (4.19)$$

Written in terms of a single constant k , decision variables and the single state variable equation (4.19) reduces to

$$k(x_1 + x_2)q = 1. \quad (4.20)$$

In this setting a stress function takes the form

$$\sigma_i = \frac{q}{L} E, \quad (4.21)$$

with stress constraints limiting displacements according to

$$cq - 1 \leq 0, \quad (4.22)$$

where we restrict our analysis to a positive displacement (including the possibility of a negative constraint is trivial; it would simply require the addition of a stress constraint to limit compressive stress). Note that of all intents and purposes c in (4.22) is not equal to constant c in the NAND analysis, but we use this notation for the sake of consistency. As before consider the case where

x_2 takes on a non-zero value at the final solution. In parallel with the discussion on the problem in a NAND setting, the feasible domain is defined by the equality and inequality constraints in equations (4.20) and (4.22). A graphical representation of the feasible domain is presented in Figure 4.4a. Note that the feasible region defined by the equality constraint (4.20) is represented by a solid line (again the shaded region denotes the interior of the feasible domain). Importantly, loosely equivalent to the NAND setting, both constraints can not be satisfied for $x_1 = 0$, hence the stress constraint should be removed if material approaches a state of void. Again the stress constraint is reformulated to represent a constraint on force as opposed to stress. Resulting in the feasible region depicted in Figure 4.4b. Similar to the NAND setting the line $x_1 = 0$ now forms part of the feasible domain and allows bar 1 to be removed completely.

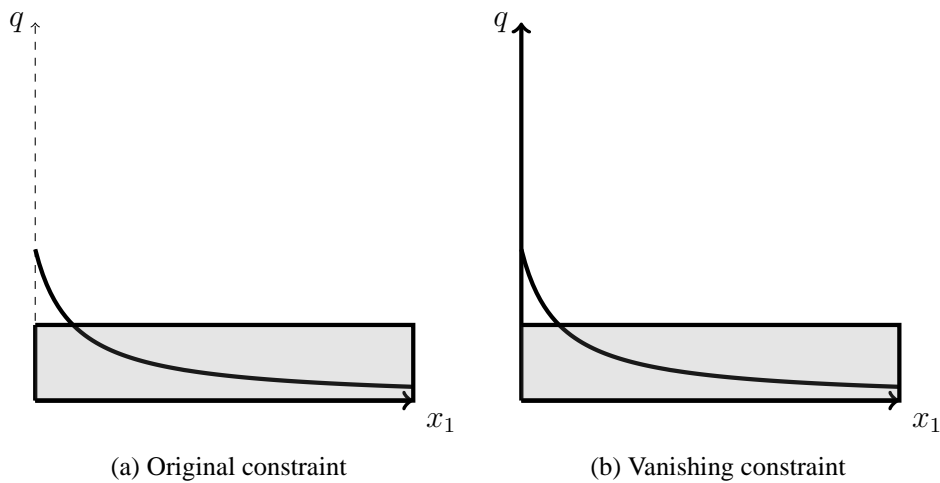


Figure 4.4: A representation of the feasible domain in a SAND setting.

From Figure 4.4 it can be seen, even though $x_1 = 0$ is now completely feasible, a region exists where both constraints can not be satisfied as $x_1 \rightarrow 0$. This is equivalent to the *limit value problem* mentioned before in discussing the NAND problem. That is, even though the design would be completely feasible if $x_1 = 0$, the element can only be reduced to some non-zero value where it experiences maximum stress. Whether the ‘jump’ across the infeasible zone can be made is highly dependent on the move limit of the algorithm, which is of course a parameter that can be tailored for specific problems. What is more, as discussed in Chapter 2, the interior point algorithm employed as subsolver requires completely feasible steps for the so-called *barrier* parameter to remain defined. To prevent this problem, in other words, to ensure the region approaching the singular solution is feasible (and partly for consistency with the traditional formulation of stress constraints in a NAND setting) we consider the ϵ -relaxed case, resulting in the feasible domains depicted in Figure 4.5. Clearly the domain is completely feasible for large enough ϵ as the optimiser approaches $x_1 = 0$. However, an appropriate value for ϵ is of course dependent on scaling of the stress constraint. This problem is investigated with numerical experiments in the following section.

To possibly reduce computational and storage requirements we investigate a stress constraint formulation with no dependency on decision variables. Although the resulting optimisation problem is not mathematically consistent, local stress constraints are manually removed from the subprob-

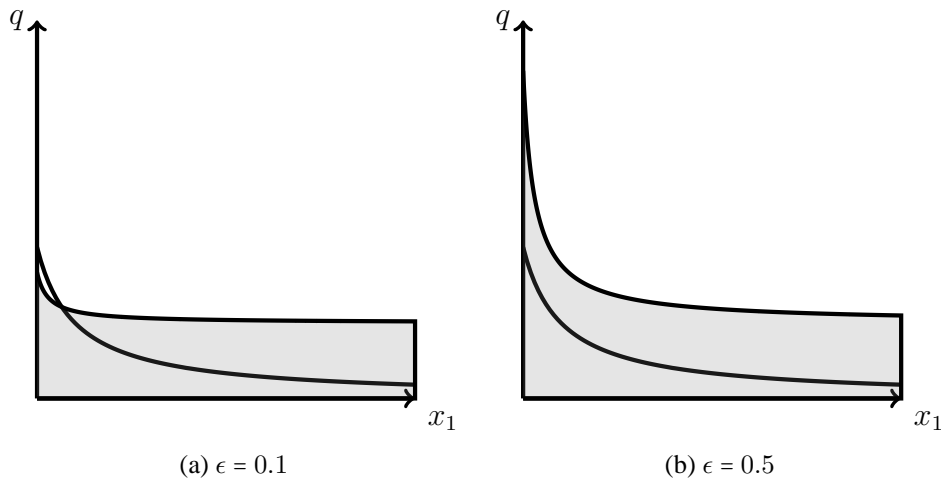


Figure 4.5: The feasible region defined by a relaxed vanishing stress constraint in a SAND setting.

lem if the corresponding decision variable reaches some non-zero value. Local stress constraints are defined as

$$cq - 1 \leq 0 \text{ if } x_e \geq \zeta \quad (4.23)$$

with ζ denoting the value below which the corresponding constraint is removed. This is clearly a heuristic procedure, referred to as an ‘inconsistent vanishing stress constraint’. The inconsistent feasible domains defined by this formulation is graphically represented in Figure 4.6 for two values of ζ . It is clear that such a formulation results in a ‘feasible’ region where the stress constraint would typically be infeasible, for the stress constraint is simply removed from the problem if $x_1 \leq \zeta$. The rationale is that the so-called *limit value problem*, intimately related to the singularity problem, can be circumvented in this manner.

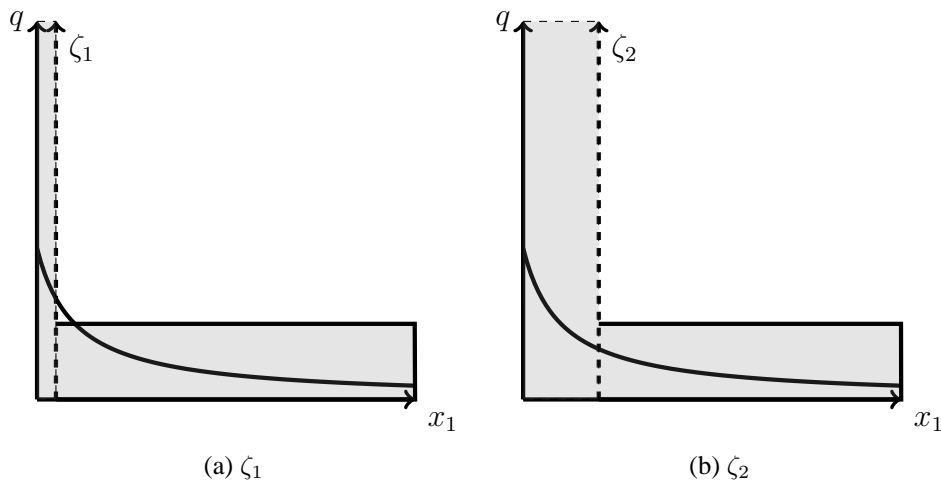


Figure 4.6: The feasible region defined by an inconsistent vanishing stress constraint in a SAND setting.

To conclude this discussion on the singularity problem in the context of the simple structure we note that both SAND and NAND formulations suffer from the said problem, although manifested in slightly different forms. In both settings stress constraints have to be removed in some way when the related material approaches a state of void as to not influence the feasibility of the design. Subsequently, in a NAND setting the optimal solution can not be reached because the resulting feasible domain is degenerate, therefore requiring relaxation which ‘opens-up’ the domain. However, it seems this problem is related to the requirement of a non-zero lower bound on decision variables to prevent numerical ill-conditioning in the structural analysis phase. In other words, a stress constraint would be feasible if the corresponding decision variable could attain a value of exactly zero. In contrast, in SAND a zero lower bound on decision variables poses no particular difficulty and such a region in the feasible domain can be reached by a classical optimisation algorithm. However, it would seem the so-called *limit value problem* inhibits this, for infeasible steps are required, which is highly dependent on the move limit of the algorithm. To prevent this stress constraints can be relaxed or removed prematurely so that the region approaching the singular solution is completely feasible. In the following section various permutations in formulating stress constraints are investigated and tested with numerical experiments.

4.5 Constraint formulation

4.5.1 Relaxed vanishing constraints

Here we chronicle the exploration and investigation of various stress constraint formulations. As the avid reader might have guessed, a wide variety of formulations and relaxation strategies can be suggested to combat the singularity problem, especially so in a SAND setting where the stress function remains explicitly dependent on state variables and a zero lower bound on decision variables allow the complete removal of stress constraints applied to void material in a mathematically consistent manner. Therefore, the work presented here is by no means complete with respect to stress constraints in a SAND setting and only serves as an initial investigation with great scope for future work.

Following the discussion in section 4.4 stress constraints are formulated with the ϵ -relaxation strategy as proposed in [39], traditionally the predominant approach in formulating the local stress constrained problem. Due to a zero lower bound on decision variables these constraints can be referred to as *vanishing* constraints and the optimisation problem a ‘mathematical program with vanishing constraints’ (MPVC) [29]. In other words if the element to which the stress constraint is applied assumes a state of void the constraint no longer influences the feasibility of the design. Therefore this formulation can be seen as a simple and mathematically consistent means to remove a constraint from the problem once $x_i = 0$. As shown in [36] stress constraints applied to elements which attain a state of void have to be removed as not to influence the feasibility of the design. As a function of the von Mises failure criterion a stress constraint applied to element i is formulated as

$$g_i(x_i, \mathbf{q}_{elmi}) = x_i \left[\frac{\sigma_{vm}(\mathbf{q}_{elmi})^2}{\sigma_y^2} - 1 \right] - \epsilon \leq 0, \quad (4.24)$$

with $i = 1, 2, \dots, n$ with n the number of elements in the mesh. Constraint $g_i(x_i, \mathbf{q}_{elmi})$ is a function

of the *local* decision variable x_i and a subset of the state variable vector $\mathbf{q}_{elmi} \subset \mathbf{q}$, where \mathbf{q}_{elmi} is the collection of nodal displacements related to element i . We study the nature of the stress constrained problem by applying the SAOi algorithm, as throughout this thesis, to problem \mathbf{P}_S^W and the two-bar truss ground structure with mesh multiplier $m = 4$.

The problem is initialised with decision variables at 1 and state variables at 0. Relaxation is initialised with $\epsilon = 0.1$ and subsequently reduced by $\epsilon/1.1$ if the maximum constraint violation is less than 1×10^{-3} in some iteration. Once $\epsilon < 1 \times 10^{-6}$ the relaxation parameter is zeroed, that is $\epsilon = 0$ at convergence. Zeroing the relaxation parameter is not possible in a NAND setting because a non-zero lower bound is typically imposed on decision variables. Therefore constraints applied to elements that approach a state of void continues to influence the feasibility of the design and some relaxation has to be kept up to final solution to ensure these constraints are indeed feasible. Furthermore, in a NAND setting stress constraints would typically be rearranged to a form that resembles

$$g_i(x_i, \mathbf{q}_{elmi}) = \left[\frac{\sigma_{vm}(\mathbf{q}_{elmi})^2}{\sigma_y^2} - 1 \right] - \frac{\epsilon}{x_i} \leq 0 \quad 0 < x_i \leq 1, \quad (4.25)$$

which is better behaved in a numerical sense and more suited to approximation strategies in common SAO algorithms, such as MMA and CONLIN [8] [23]. Clearly the constraint has to be feasible in its original form if $\epsilon = 0$ independent of x_i .

Considering relaxed vanishing constraints (4.24) it can be shown that the solution of the perturbed problem converges to a stationary point of the original MPVC as $\epsilon \rightarrow 0$ [40]. However, the authors comment this can only be achieved in practice for a ‘‘very small value’’ of ϵ , dependent on scaling of the constraints.

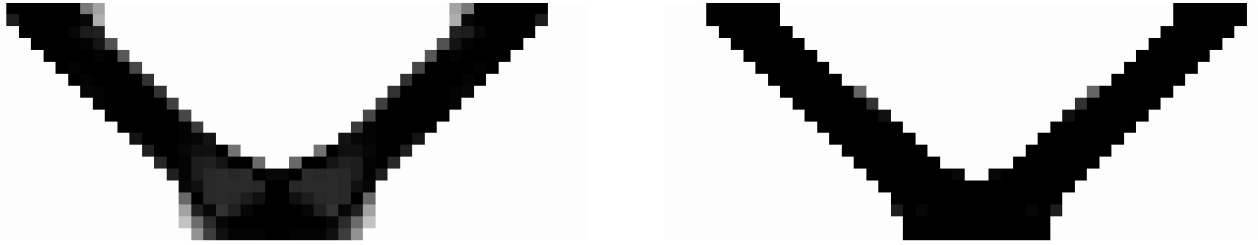
The solution of the stress constrained problem is depicted in Figure 4.7a. Relevant numerical values related to the solution is summarised in the caption of the figure. Problem size is summarised by the number of decision variables (n_d), state variables (n_s), equality constraints (n_e) and inequality constraints (n_i). The volume fraction at optimality is denoted by f_v , compliance at optimality by f_c , number of decision variables that obtain a value of zero by $n_{[0]}$, number of decision variables at 1 by $n_{[1]}$, number of decision variables at intermediate values by $n_{[i]}$ and black-and-white fraction by $\phi_{B\&W}$. Finally the number of iterations to solution is denoted by N_{iter} .

For comparative purposes we present the result of an optimisation run on the same problem, but with stress constraints removed and compliance limited to 232.19, that is the compliance of the stress constrained problem at optimality. The optimal topology is depicted in Figure 4.7b with numerical values again summarised in the caption. Immediately it is clear the stress constrained problem requires more material to be feasible. However, of concern is the fact that the stress constrained problem does not approach purely black-and-white designs, like the compliance constrained case. In Chapter 3 the problems in interpreting grey material is discussed at length, and here, considering the stress constrained problem, it is also of concern. On the bright side, because $\epsilon = 0$ at solution no constraints are unrealistically feasible due to relaxation. To be clear, consider constraint (4.24), for $x_i > 0$ and $\epsilon = 0$ the inequality

$$\left[\frac{\sigma_{vm}(\mathbf{q}_{elmi})^2}{\sigma_y^2} - 1 \right] \leq 0 \quad (4.26)$$

holds at solution. That is, stress constraints are feasible in their original ‘unrelaxed’ form, independent of x_i . However, clearly stress constraints effect the extent to which the solution is driven to

purely solid-void designs. Of course some penalisation can be applied to decision variables in the stress constraints, although here we wish to keep the constraint formulation as simple as possible and only penalise stiffness directly in the state equation (keeping in mind the numerical and scaling issues related to stress constraints already mentioned).

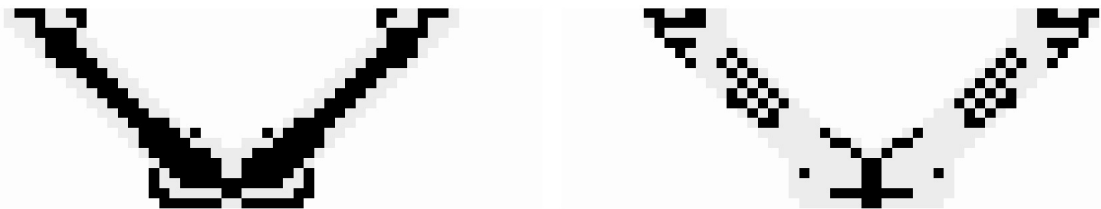


(a) Result for $m = 4$, $n_d = 1200$, $n_s = 7522$, $n_e = 7522$, $n_i = 1200$: $f_v = 0.233$, $f_c = 232.19$, $n_{[0]} = 898$, $n_{[1]} = 92$, $n_{[i]} = 210$, $\phi_{B\&W} = 0.825$, $N_{iter} = 197$

(b) Result for $m = 4$, $n_d = 1200$, $n_s = 7522$, $n_e = 7522$, $n_i = 1$: $f_v = 0.221$, $f_c = 232.19$, $n_{[0]} = 934$, $n_{[1]} = 254$, $n_{[i]} = 12$, $\phi_{B\&W} = 0.99$, $N_{iter} = 32$

Figure 4.7: The generated topologies considering relaxed vanishing constraints (a) and the simple compliance constrained case (b).

To shed some light on the unsatisfactory black-and-white fraction obtained for the stress constrained problem we investigate some properties related to active constraints and decision variables. Stress constraints deemed active (elements which satisfy the inequality $g_i(x_i, q_{elmi}) > -1 \times 10^{-6}$) are depicted in black in Figure 4.8a, with all non-zero elements depicted in grey. In Figure 4.8b all the elements on the upper bound are presented (in black), with all other non-zero elements once again depicted in grey.



(a) Active stress constraints.

(b) Elements on the upper bound.

Figure 4.8: A depiction of active stress constraints and elements on the upper bound for the generated topology with relaxed vanishing constraints.

From these results the interplay between fully stressed elements and decision variables is still unclear. It would seem fully stressed elements are predominantly those at intermediate values, as one would expect, since the amount of material in these elements are indirectly limited by the stress measure. However, in a pursuit of a fully stressed design close to predominantly solid-and-void this is undesirable. Figure 4.9a depicts the elements that are fully stressed and intermediate while Figure 4.9b depicts those that are fully stressed and on the upper bound (in both cases the mentioned elements are depicted black, with all other elements depicted in grey). From these figures it is quite clear that intermediate elements are predominantly those that are fully stressed. This is of course problematic, for how should these elements be interpreted physically?

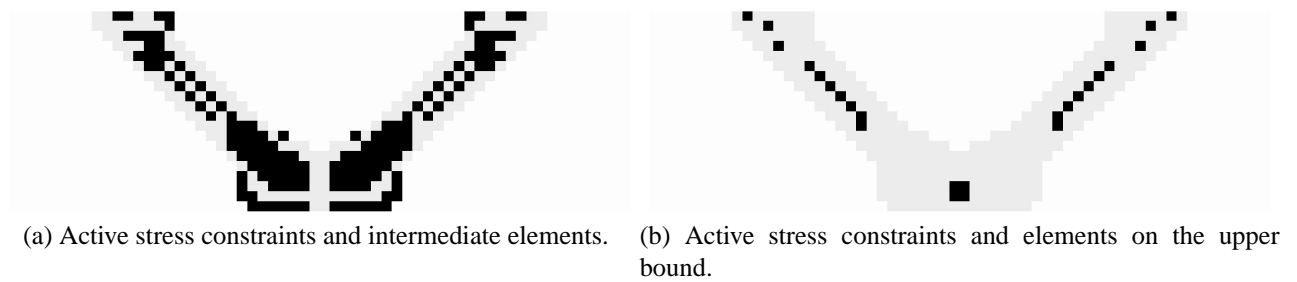


Figure 4.9: A depiction of active stress constraints and solid/intermediate elements for the generated topology with relaxed vanishing constraints.



Figure 4.10: The generated topologies with stress constraints formulated as relaxed vanishing constraints with $\epsilon_0 = 0.2$ and 0.75 .

In an attempt to increase the black-and-white fraction at convergence, and to shed more light on the characteristics of the SAND formulated problem, we do the same optimisation run as above with ϵ initialised at 0.2. The result is presented in Figure 4.10a. In comparison to the case where ϵ is initialised at 0.1 the black-and-white fraction is indeed improved. More elements are removed and more elements reach the upper bound value of 1. However, the algorithm does not converge and is terminated after 500 iterations. The relaxation parameter could be reduced to roughly 0.15 before the maximum constraint violation does not converge to a value less than 1×10^{-3} . Therefore too many elements are removed from the domain and the design would be infeasible if ϵ is reduced further. Here we see the problem with scaling of numerical parameters related to stress constraints mentioned before. Furthermore, since decision variables are allowed to take on values of exactly zero all material derivatives vanish as elements acquire a state of void. Therefore material can not return to the design space once removed. The result we see here is a subtle example of the fact that a different local optima is converged to for large (or too large) values of ϵ . In Figure 4.10b the extreme case is depicted for ϵ fixed at 0.75. In these cases, with relaxation initially too severe, the local optima changes too abruptly as ϵ is reduced and the algorithm does not converge.

4.5.2 Alternative stress relaxation: Closing down and opening up

The continuation strategy employed before is referred to as ‘closing down’ of the design space once the optimisation algorithm converges to the local perturbed optima. Here we investigate an alternative stress constraint formulation which allows, in principle, the ‘opening up’ of the design space as the algorithm converges to the local optima. These concepts are taken from the work done

in [10].

It is well known that the singularity problem, related to local stress constraints, can be negated by perturbing the original problem with standard ϵ -relaxation, as proposed in [39] and referred to as *vanishing* constraints in a SAND setting [29]. The magnitude of perturbation is reduced as the perturbed problems are successively solved, leading to the solution of the original problem. Convergence to the stationary point of the original problem is proved in [40]. The authors in [40] point out that in practice this only holds for sufficiently small values of ϵ , dependent on numerical scaling of the constraints. We have experienced similar difficulties related to the initial relaxation value; if too large a different local optima is found than that of the original problem and the algorithm can not converge to the desired solution.

In an attempt to circumvent or improve upon these difficulties we propose an adapted stress relaxation scheme. This scheme, an amalgamation of a so-called vanishing constraint and the relaxation scheme proposed in [10], seems to be more conservative in relaxing stress constraints. Note that the original formulation relaxes all elements, regardless of material state. Ideally one would wish to relax only elements that approach a state of void, for only these elements suffer from the singularity problem, or the limit value problem to be more specific. Furthermore, it might be the relaxation of solid elements (or nearly solid elements) that allows the algorithm to remove too much material, problematic if we wish to converge to the solution of the original problem, as illustrated in the previous section. Therefore we propose the following vanishing constraint formulation, we emphasise this is merely an adaptation of the formulation proposed in [10], given by

$$g_i(x_i, \mathbf{q}_{elmi}) = x_i \left[\frac{\sigma_{vm}(\mathbf{q}_{elmi})^2}{\sigma_y^2} (1 + \theta\epsilon) - 1 \right] - \epsilon \leq 0 \quad , \quad 0 \leq \theta \leq 1. \quad (4.27)$$

Although only a subtle difference, for $\theta = 1$ formulation (4.27) does not relax solid elements (elements at 1). That is, a stress constraint applied to an element with $x_i = 1$ is not relaxed and is required to be feasible with respect to the original constraint. For $\theta = 0$ constraint (4.27) reduces to the original vanishing constraint (4.24). In line with the discussion in [10] we define a *stress multiplier* denoted by S_θ . Rearranging constraint (4.27) gives

$$\sigma_{vm}(\mathbf{q}_{elmi})^2 \leq \left[\frac{\frac{\epsilon}{x_i} + 1}{1 + \theta\epsilon} \right] \sigma_y^2 \quad , \quad 0 \leq \theta \leq 1, \quad (4.28)$$

where the term in brackets $[\]$ is referred to as the *stress multiplier*. For the sake of clarity we define

$$S_\theta(\epsilon, \theta, x_i) = \frac{\frac{\epsilon}{x_i} + 1}{1 + \theta\epsilon} \quad , \quad 0 \leq \theta \leq 1, \quad (4.29)$$

from which it is clear $S_\theta \geq 1$ and denotes the amount by which the stress in the element is allowed to exceed the original stress limit due to relaxation. Note the stress terms are squared to simplify the calculation of first - and second-order sensitivities and does not influence the feasibility of the stress measure, clearly so once $S_\theta = 1$ for $\epsilon = 0$. In Figure 4.11 a comparison of S_θ for $\epsilon = 0.1$ and 0.2 as a function of x_i is depicted. In both figures S_θ is plotted for $\theta = 0$ and 1 . Considering the plots one can see the case where $\theta = 1$ elements at $x_i = 1$ are not relaxed, that is $S_\theta(\epsilon, 1, 1) = 1$ independent of ϵ . Furthermore, relaxation is consistently less for $\theta = 1$, hence a more conservative relaxation strategy.

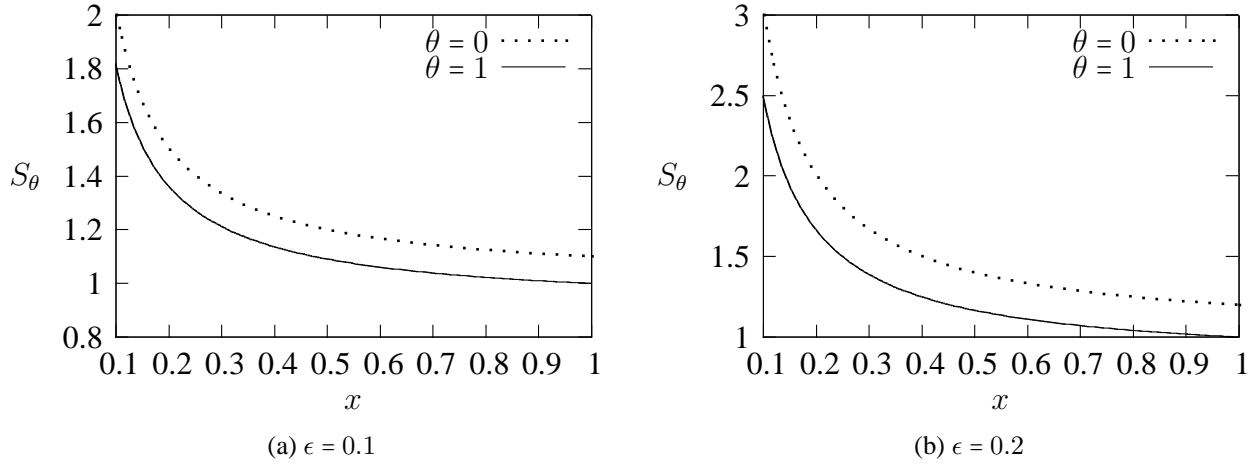


Figure 4.11: The effect of relaxation as a function of decision variable x_i and θ .

Convergence of such a relaxation scheme applied to vanishing constraints in a SAND setting is typically studied considering two functions, $G(\mathbf{y})$ and $H(\mathbf{y})$ (where \mathbf{y} denotes all the design variables, i.e. state and decision variables). The functions form a relaxed vanishing constraint according to

$$H(\mathbf{y})G(\mathbf{y}) \leq \epsilon, H(\mathbf{y}) \geq 0, \quad (4.30)$$

where $H(\mathbf{y}) = x_i$ in our case. In [40] convergence is proved where neither $H(\mathbf{y})$ or $G(\mathbf{y})$ is a function of ϵ . Therefore it is unclear whether the convergence proof holds if function $G(\mathbf{y})$ is also perturbed by the relaxation parameter, as is the case for $\theta = 1$ in constraint (4.27). Furthermore, the author in [10] claims that the constraint

$$g_i(x_i, \mathbf{q}_{elmi}) = \frac{\sigma_{vm}(\mathbf{q}_{elmi})^2}{\sigma_y^2}(1 + \theta\epsilon) - 1 - \frac{\epsilon}{x_i} \leq 0, \quad (4.31)$$

which is simply constraint (4.27) divided through by x_i , is consistent with the convergence proof provided in the original paper in which ϵ -relaxation is proposed [39]. If we can define the original and perturbed stress functions as $H(\mathbf{y})$ and $H_\epsilon(\mathbf{y})$ respectively

$$H(\mathbf{y}) = \left[\frac{\sigma_{vm}(\mathbf{q}_{elmi})^2}{\sigma_y^2} - 1 \right], \quad (4.32)$$

$$H_\epsilon(\mathbf{y}) = \left[\frac{\sigma_{vm}(\mathbf{q}_{elmi})^2}{\sigma_y^2}(1 + \theta\epsilon) - 1 \right], \quad (4.33)$$

it is easy to see that

$$H_\epsilon(\mathbf{y}) \geq H(\mathbf{y}), \quad (4.34)$$

that is, if the perturbed stress function is feasible for some set of design variables \mathbf{y} the original stress function would also be feasible (and $H_\epsilon(\mathbf{y}) = H(\mathbf{y})$ for $\epsilon = 0$).

The SAOi algorithm is applied to the problem with stress constraints formulated as per equation (4.27), with ϵ initially set at 0.1, 0.2, 0.5 and 0.75 respectively. Again the problem is initialised

with decision variables at 1 and state variables at 0. Relaxation is reduced by $\epsilon/1.1$ if the maximum constraint violation is less than 1×10^{-3} . Once $\epsilon < 1 \times 10^{-6}$ the relaxation parameter is zeroed. The topologies generated are depicted in Figure 4.12 with some numerical values related to the solutions summarised in Table 4.1.

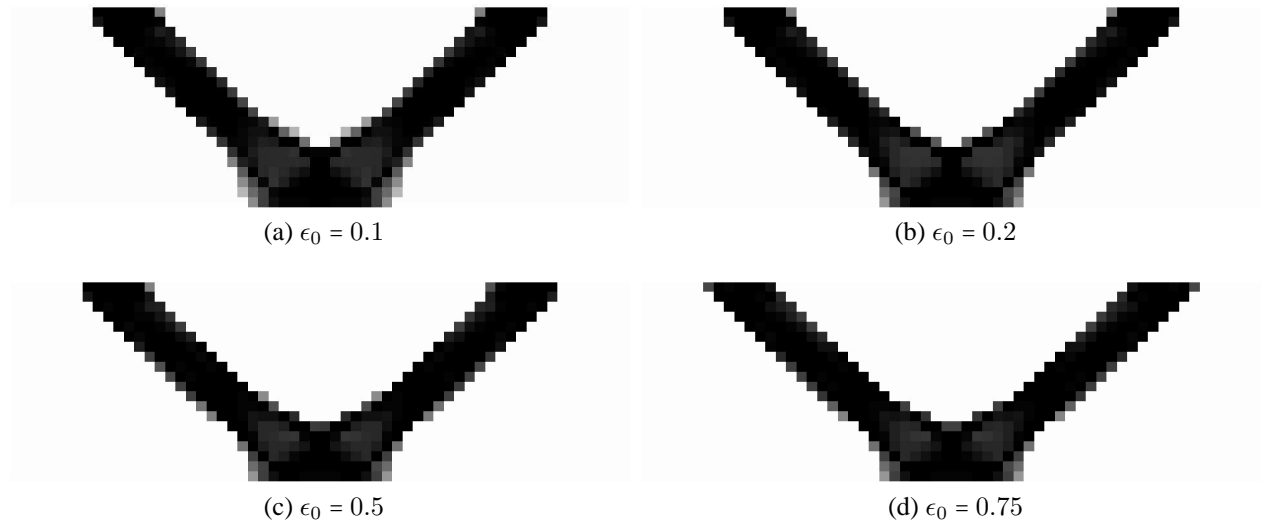


Figure 4.12: Topologies generated for the weight minimization of the two-bar truss with alternative stress relaxation and a range of ϵ_0 .

Table 4.1: A comparison of the results obtained for the weight minimization of the two-bar truss with alternative stress relaxation and a range of ϵ_0 .

ϵ_0	f_v	$\phi_{B\&W}$	$n_{[0]}$	$n_{[1]}$	N_{iter}
0.1	0.234	0.818	896	86	175
0.2	0.231	0.828	906	88	183
0.5	0.232	0.840	904	104	205
0.75	0.232	0.841	906	104	210

Clearly scaling of the relaxation parameter is much more robust. The adapted relaxation strategy with $\theta = 1$ allows ϵ to be initialised at values previously deemed to large, with improved black-and-white fractions obtained for higher values of initial relaxation. To investigate the distribution of fully stressed solid and void elements consider the case of $\epsilon_0 = 0.75$. Figure 4.13a depicts elements on the upper bound, Figure 4.13b depicts elements with active stress constraints, Figure 4.13c depicts elements that are active and intermediate and Figure 4.13d depicts elements that are deemed active and solid (the said elements are represented in black, with all other elements in grey). Clearly it is once again intermediate elements which are deemed active with respect to stress constraints, which seems to be pointing to the fact that either some penalisation should be

applied to the decision variables in the stress constraints or penalisation should be increased in the state equation.

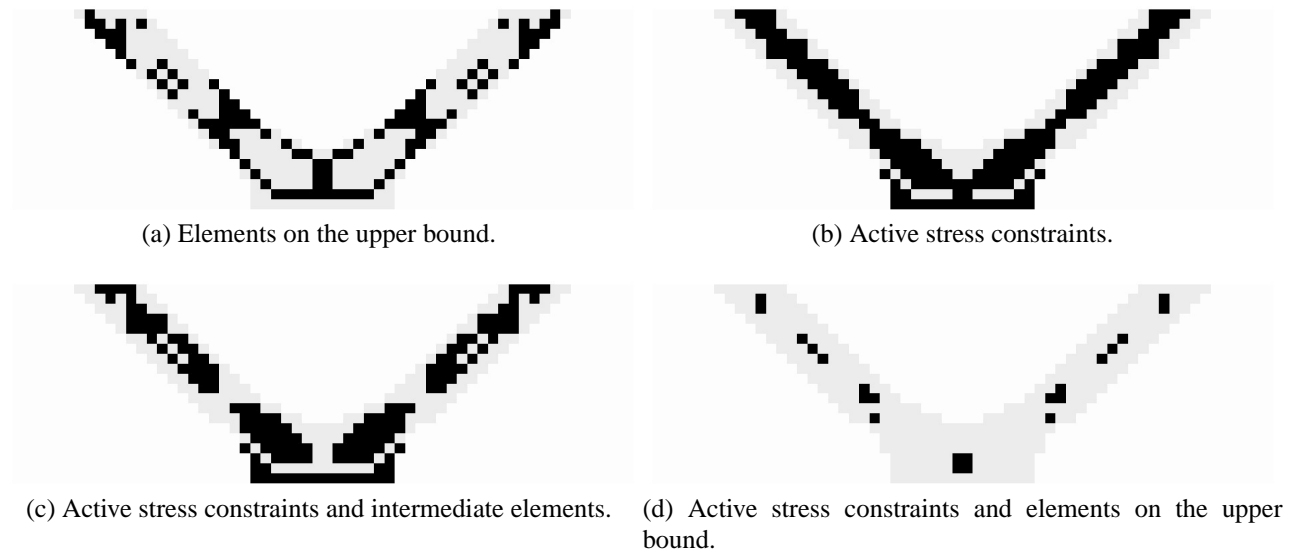


Figure 4.13: A depiction of active stress constraints and elements at solid/intermediate states with the ‘closing down’ continuation strategy.

A comparison of the results in Table 4.1 and the result for the standard vanishing constraint depicted in Figure 4.7a shows that black-and-white fractions are indeed improved for $\epsilon_0 > 0.2$, although less so for $\epsilon_0 = 0.1$. To shed some light on the relationship between relaxation and black-and-white fractions, and specifically the effect of the continuation strategy, a plot of ϵ and $\phi_{B\&W}$ is depicted in Figure 4.14 for $\epsilon_0 = 0.75$. We see that the black-and-white fraction at solution is obtained in the initial iterations, with closing down of the design space having only a marginal effect on $\phi_{B\&W}$.

Studying the convergence behaviour of $\phi_{B\&W}$ relative to ϵ serves as a neat introduction to the next relaxation strategy we investigate. The strategy, borrowed from the work done in [10], is based on the fact that stress constraint (4.27) with $\theta = 1$ is not relaxed if the related material is solid. That is, if predominantly black-and-white designs are obtained relaxation does not have to be reduced so that the original unperturbed problem is eventually solved. To be clear, if the final solution has decision variables exclusively at 0 and 1, the solution will be feasible with respect to the original unperturbed problem, even though ϵ is non-zero. We refer to this strategy as an ‘opening up’ strategy, in conjunction with the term used in [10]. The design space is ‘opened up’ successively after each perturbed problem is solved, therefore relaxation is initialised with $\epsilon_0 = 1 \times 10^{-2}$ and subsequently increased by $\epsilon \times 1.1$ if the maximum constraint violation is less than 1×10^{-3} in some iteration. The maximum value at which ϵ is limited, denoted by ϵ^* , is set at 0.1, 0.2, 0.5 and 1.0 respectively. The topologies generated is depicted in Figure 4.15 and the results summarised in Table 4.2.

Again black and white fractions are disappointing, even more so than the closing down strategy. However, this strategy does simplify and partly negate the problems with numerical scaling of parameter ϵ , since only a slightly perturbed problem is initially solved. In Figure 4.16 a convergence

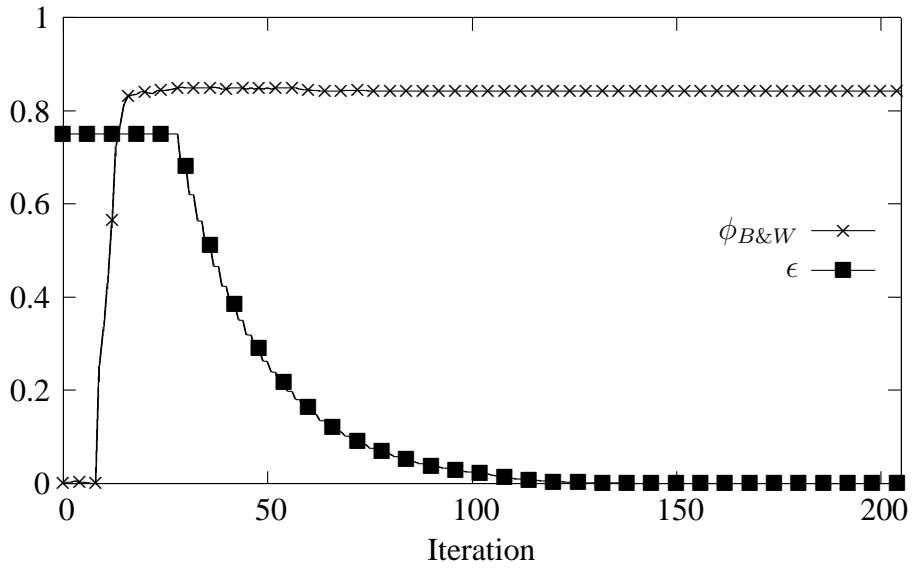


Figure 4.14: Convergence plot of $\phi_{B\&W}$ and ϵ with the ‘closing down’ continuation strategy.

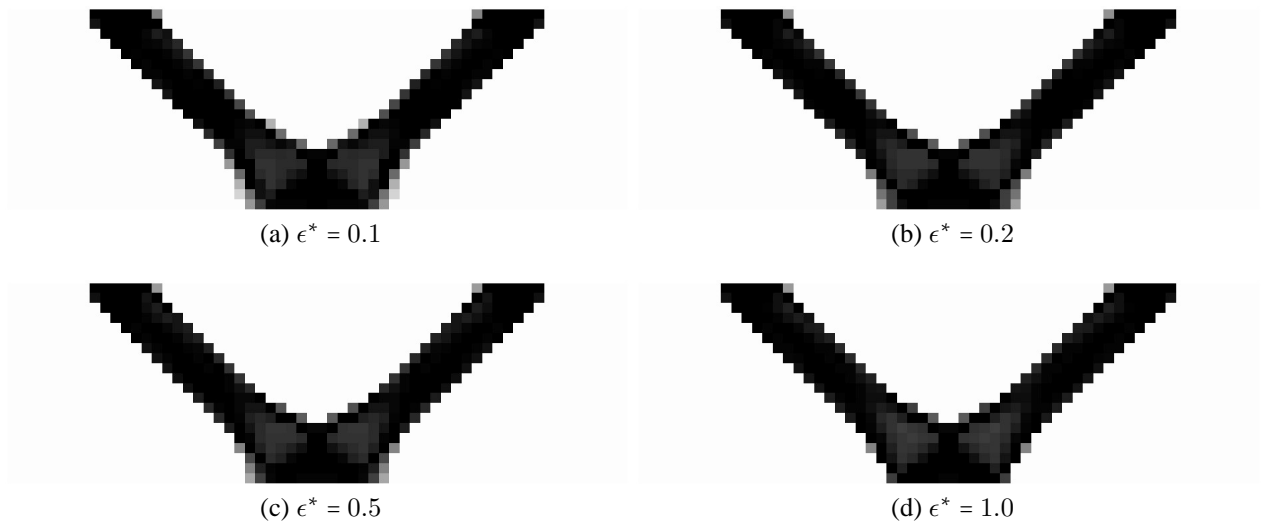


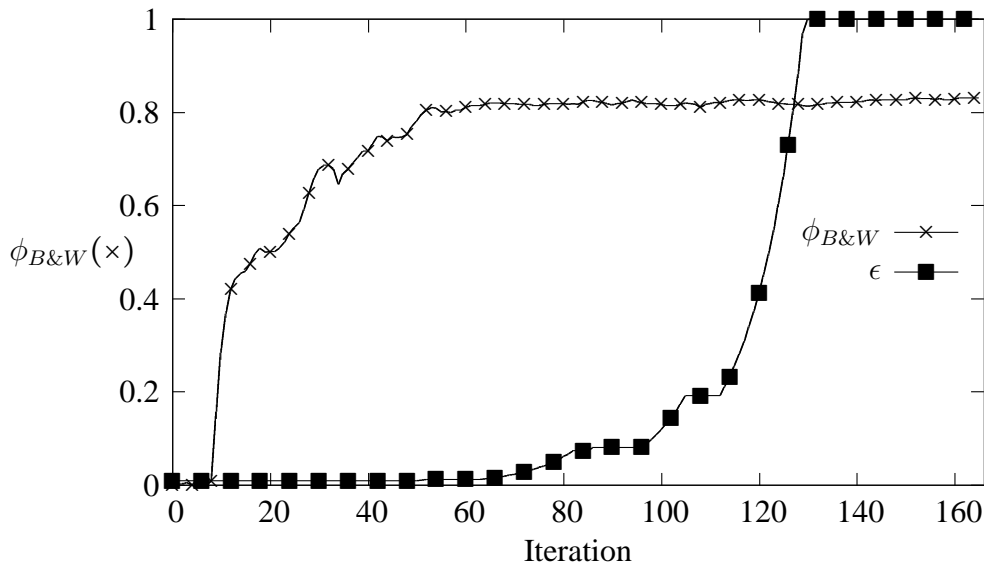
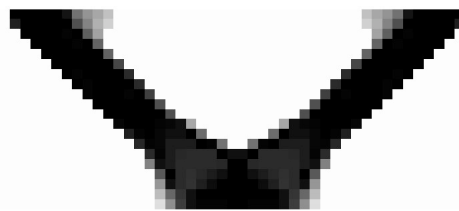
Figure 4.15: Topologies generated for the weight minimization of the two-bar truss with alternative stress relaxation and a range of ϵ^* .

plot of $\phi_{B\&W}$ and ϵ is presented for the ‘opening up’ strategy and $\epsilon^* = 1.0$. We see opening up the design space has only a marginal effect in increasing black and white fractions, similar to the closing down strategy. However, considering the design obtained when ϵ is fixed at $\epsilon_0 = 1 \times 10^{-2}$ in Figure 4.17, we see some elements are indeed removed as ϵ is increased. However, with this strategy grey elements can not be tolerated as these are relaxed and exceed the allowable stress limit.

For the sake of consistency we repeat the plots of solid/intermediate elements with active stress constraints in Figure 4.18, as presented for the ‘closing-down’ scenario. Clearly it is once again

Table 4.2: A comparison of the results obtained for the weight minimization of the two-bar truss with alternative stress relaxation and a range of ϵ^* .

ϵ^*	f_v	$\phi_{B\&W}$	$n_{[0]}$	$n_{[1]}$	N_{iter}
0.1	0.233	0.820	898	86	123
0.2	0.231	0.827	906	86	126
0.5	0.230	0.820	906	78	152
1.0	0.229	0.832	910	88	166


 Figure 4.16: Convergence plot of $\phi_{B\&W}$ and ϵ with the ‘opening up’ continuation strategy.

 Figure 4.17: The generated topology with alternative stress relaxation and ϵ_0 fixed at 1×10^{-2} .

elements at intermediate values that are fully stressed. As mentioned before, these results seem to point to the fact that the black-and-white fractions should be improved with more severe penalisation in the state equation or by penalising decision variables in the stress constraints. It would seem the manner in which the design space is perturbed via relaxation of stress constraints as the algorithm converges has only a marginal effect on the extent to which purely black-and-white designs are obtained.

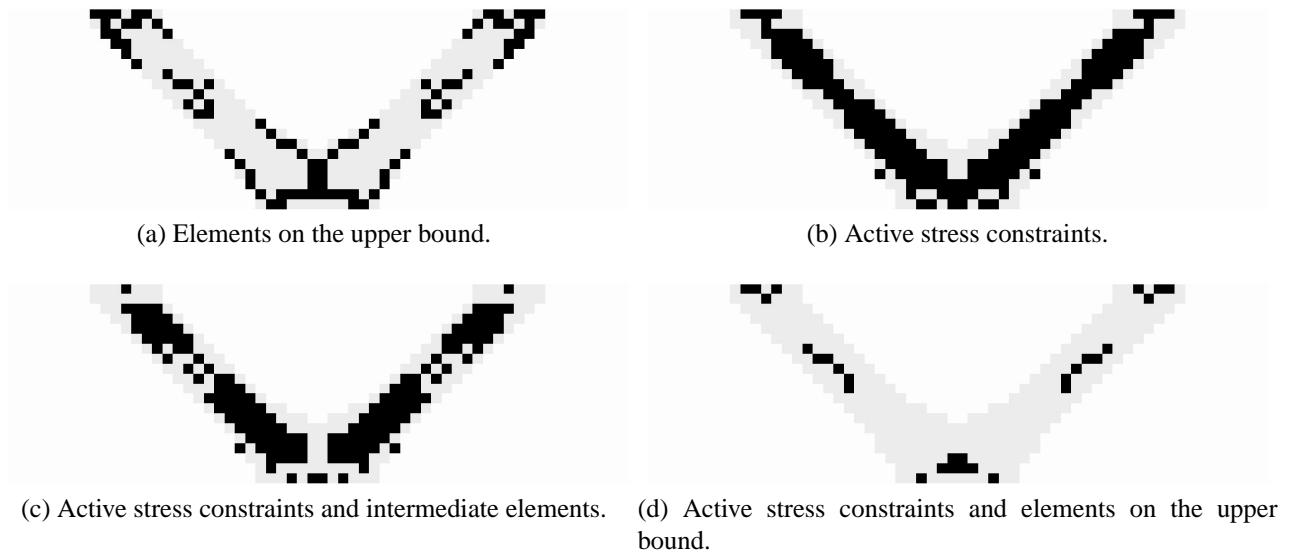


Figure 4.18: A depiction of active stress constraints and elements at solid/intermediate states with the ‘opening up’ continuation strategy.

4.5.3 Inconsistent vanishing constraints

As mentioned before, in a SAND setting stress constraints can be formulated as a function of only state variables. However, stress constraints have to be removed for material that attains (or approaches) a state of void, referred to as the limit value problem. Here this is achieved by manually removing stress constraints from the subproblem if the corresponding decision variable obtains a value lower than some constant ζ . Formally, we define ‘inconsistent vanishing constraints’ as

$$g_i(x_i, \mathbf{q}_{elmi}) = \left[\frac{\sigma_{vm}(\mathbf{q}_{elmi})^2}{\sigma_y^2} - 1 \right] \leq 0 \text{ only if } x_i \geq \zeta. \quad (4.35)$$



Figure 4.19: Topologies generated for the weight minimization of the two-bar truss with inconsistent vanishing constraints and two values for ζ .

Two values of ζ are tested with numerical experiments, 0.1 and 0.5. The topologies generated are depicted in Figure 4.19 and the results summarised in Table 4.3. Of course this procedure is heuristic and the solution obtained is highly dependent on the value chosen for ζ . However, we have shown the solution obtained considering traditional relaxation of stress constraints is also

Table 4.3: A comparison of the results obtained for the weight minimization of the two-bar truss with inconsistent vanishing constraints and two values for ζ .

ζ	f_v	$\phi_{B\&W}$	$n_{[0]}$	$n_{[1]}$	N_{iter}
0.1	0.240	0.815	876	102	71
0.5	0.231	0.827	908	86	35

highly dependent on numerical scaling of the problem. Furthermore, this strategy to combat the limit value problem is quite simple and stress constraints remain in the original form, as derived in Appendix B, with no dependency on decision variables. Further still, clear from the results in Table 4.3, the algorithm requires substantially less iterations to converge to the local stationary point. Therefore we can deduce that, in conjunction with penalisation via the state equation, elements with decision variables at $x_i \leq 0.5$ contribute very little to the design for this specific problem. The solutions obtained here are practically the same as those obtained previously considering consistent relaxation strategies. Although these results are presented as a brief investigation into the possibility of such a formulation, and by no means complete, it would seem this formulation is an avenue which can be explored in the future.

In the following section we conduct a similar preliminary investigation into simultaneously applying progressive penalisation to the state equation to drive decision variables to a purely 1-0. In principle then, if all decision variables assume a value of either 0 or 1 at final solution, with stress constraints only applied to each element deemed solid, the formulation proposed above would seem justifiable to some extent.

4.5.4 Stress constraints and progressive penalisation

Here we briefly investigate progressive penalisation and the SAND formulated stress constrained problem. Once again it should be emphasised that the work presented here is by no means complete, and only serves as an initial investigation into the subject. First of, consider the so-called ‘inconsistent vanishing constraints’ defined before. The constant ζ is set to 0.5 (as previously) however, here SIMP parameter p is increased according to $1.1p$ in any iteration if the maximum constraint violation is less than 1×10^{-3} (with p initialised at 3). Three cases are tested with different limits on p , denoted by $p^* = 10, 30, 100$. The topologies generated are depicted in Figure 4.20 with results summarised in Table 4.4.

The results show progressive penalisation does increase black-and-white fractions, albeit exclusively by increasing the amount of elements that approach a state of solid or 1. The amount of elements deemed void does not increase with more severe penalisation. In other words, in increasing penalisation it is only intermediate elements that are driven to 1, for these elements are necessary for the design to be feasible. An increase in penalisation causes an element not at exactly 1 to contribute less to the global stiffness of the structure, hence the element is driven to a state of 1 for the structure to remain feasible. Although graphically the designs seem to approach that of purely solid-and-void, strictly speaking the number of elements at exactly 0 and 1 ($\phi_{B\&W}$) is rather

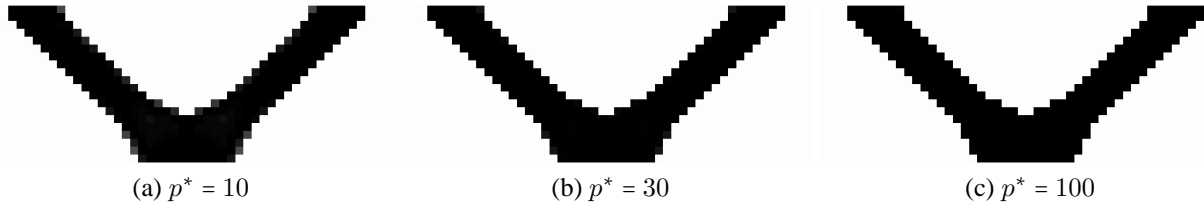


Figure 4.20: Topologies generated for the weight minimization of the two-bar truss with inconsistent vanishing constraints and progressive penalisation.

Table 4.4: A comparison of the results obtained for the weight minimization of the two-bar truss with inconsistent vanishing constraints and progressive penalisation.

Approximation	f_v	$\phi_{B\&W}$	$n_{[0]}$	$n_{[1]}$	N_{iter}
10	0.239	0.850	908	112	104
30	0.242	0.858	908	122	129
100	0.243	0.858	908	122	162

low. Especially considering the severity of penalisation, as much as 100 at convergence in the last case considered. Furthermore, the black-and-white fractions of both cases $p^* = 30$ and $p^* = 100$ are equal. Hence it would seem some limit is approach in driving the solution to that of purely solid-and-void.

Here follows the final formulation we consider in this chronicle of the investigation into stress constraints in a SAND setting. Stress constraints formulated per equation (4.27) with $\theta = 1$ are considered. Keep in mind that the relaxation scheme is formulated such that stress constraints applied to elements at exactly 1 is not relaxed. Therefore ϵ can be fixed at some value and progressive penalisation applied to the problem in an attempt to drive the solution to a state of purely solid-and-void. If this is achieved, all stress constraints would have either vanished from the problem or not be relaxed, independent of the value of ϵ .

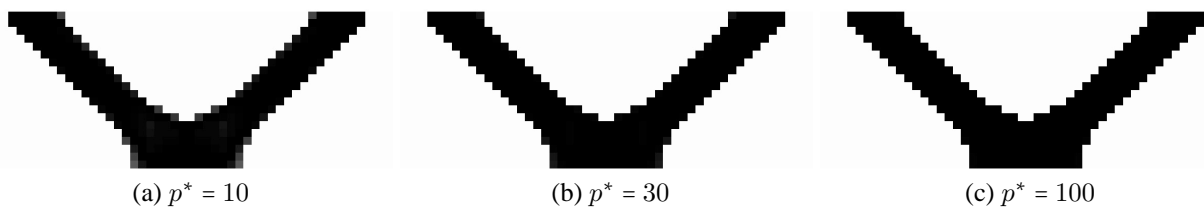


Figure 4.21: Topologies generated for the weight minimization of the two-bar truss with alternative stress relaxation and progressive penalisation.

Here some results are presented for $\epsilon = 0.2$ and exactly the same continuation strategy on p as described above. The topologies generated are depicted in Figure 4.21 with the results summarised in Table 4.5. Once again black-and-white fractions are not as high as expected, considering the severity of penalisation. Similar to the results above the topologies seem to be approaching a

Table 4.5: A comparison of the results obtained for the weight minimization of the two-bar truss with alternative stress relaxation and progressive penalisation.

Approximation	f_v	$\phi_{B\&W}$	$n_{[0]}$	$n_{[1]}$	N_{iter}
10	0.240	0.847	906	110	78
30	0.243	0.852	906	116	133
100	0.245	0.857	906	122	159

state of purely solid-and-void, although the elements are not exactly at 0 or 1. Furthermore, once again the number of elements that vanish from the problem does not increase with more severe penalisation. It is only intermediate elements that are driven to a state of 1 as penalisation is increased.

These results seem to indicate that simply setting

$$x_i^* = 1 \text{ if } x_i^* > 0 \text{ for } i = 1, 2, \dots, n \quad (4.36)$$

at convergence for $p = 3$, given this can be justified with regards to the specific relaxation strategy, is a viable alternative to progressive penalisation in this setting. Rounding of decision variables to obtain purely black-and-white solutions is discussed in the section on progressive penalisation in Chapter 3.

4.6 Large scale problems and the MBB beam

Thus far we have explored a variety of permutations of the local stress constrained problem in a SAND setting. Throughout this thesis, and specifically discussed in Chapter 2, great emphasis has been placed on the benefits in computational requirements of the SAND formulated problem, requiring no explicit structural analysis and leading to immense sparsity of the locally constrained problem. The topology optimisation problem subject to local stress constraints, with structural analysis discretised via the FE method, is inherently a very large scale problem. Especially so if we desire the discretisation of structural analysis to be very accurate, hence requiring a large number of small elements in the FE mesh. In the light of this we explore the computational scaling of the problem for finer and finer mesh discretisations.

Consider the weight minimisation of the two-bar truss ground structure with mesh multiplier $m = 4, 6, 8$ and 10. As throughout this chapter we consider the problem of weight minimisation subject to local stress constraints solved with the SAOi algorithm. Stress constraints are formulated as per equation (4.27) and the so-called ‘closing-down’ continuation strategy on relaxation is considered. Although we have previously reduced the relaxation parameter ϵ to exactly zero, here due to numerical difficulties related to the large scale nature of the problems ϵ is reduced to 1×10^{-2} . The relaxation parameter is initialised at $\epsilon_0 = 0.2$ and further details on the continuation strategy is exactly the same as considered previously. Furthermore, the same hardware platform as detailed in Chapter 3 is employed.

In Figure 4.22 the series of topologies generated with the ‘closing-down’ strategy is depicted with some numerical values at solution summarised in the caption of each figure. We see the black-and-white fractions at solution improves with finer and finer mesh discretisations. However, even for the largest problem, the number of ‘grey’ elements at solution is still of concern. At a closer inspection it seems a different local optima is converged to for $m = 10$, considering the width of the V structure. This of course highlights the difficulties in scaling of the relaxation parameter.

In Table 4.6 some results related to problem size and computation time is summarised. Considering the number of decision (n_d) and state (n_s) variables we see that problem size roughly doubles with successive mesh discretisations. Therefore we would expect computation time (denoted by T_{CPU}) to, for instance, quadruple with successive mesh discretisations if computation time scaled to $\mathcal{O}(n^2)$, the expected values denoted by $T_{CPU}^*(n^2)$. However, it would seem computation time scales more to $\mathcal{O}(n)$, the expected values denoted by $T_{CPU}^*(n)$, although only roughly so. What is more, we can only present this analysis for the two-bar truss test problem and specific range of problem sizes. In spite of this, the observation that the local stress constrained SAND formulated problem could scale linearly in computation time is promising, considering the immense computational resources required to solve an equivalent NAND formulated problem.

In Figure 4.23 the topologies generated for $m = 14$ considering the so-called ‘closing down’ and ‘opening up’ continuation strategies on relaxation are depicted. The ‘opening up’ case is initialised with $\epsilon_0 = 1 \times 10^{-2}$ with ϵ increased unto a value of 1 as the algorithm converges to the stationary point. On close inspection we see the ‘closing down’ result is actually asymmetric. We speculate this is due to the severe scaling difficulties related to the relaxation approach and such a large problem size. We can also report the algorithm had to take some infeasible steps approaching the local optima, with difficulties related to infeasible subproblems highlighted before. Therefore the ‘opening up’ result is also presented to show the solution, although different, does not suffer from the same difficulties in scaling of relaxation.

Table 4.6: Scaling of problem size and computation time considering the weight minimisation of the two-bar truss ground structure subject to local stress constraints and the ‘closing down’ continuation strategy.

m	n_d	n_s	T_{CPU}	$T_{CPU}^*(n)$	$T_{CPU}^*(n^2)$
4	1200	7522	78		
6	2700	16682	262	156	312
8	4800	29442	527	524	1048
10	7500	45802	931	1054	2108

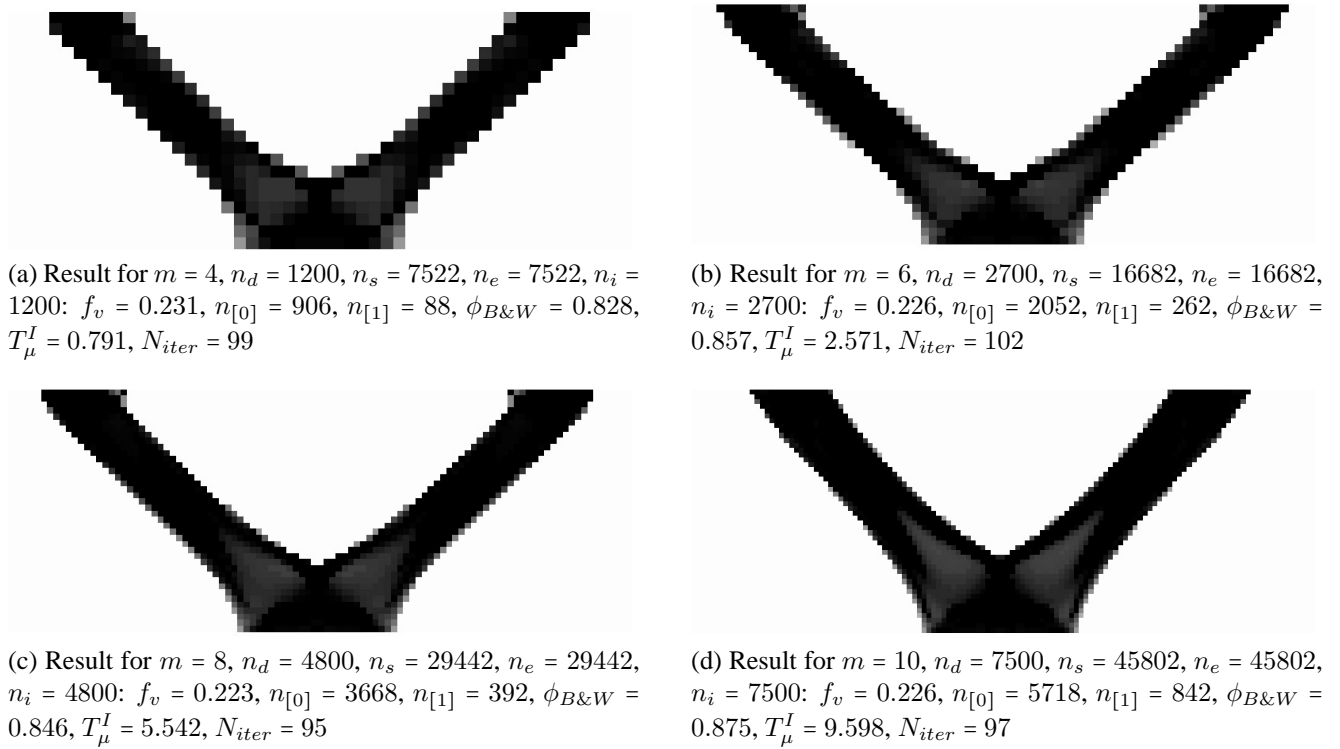


Figure 4.22: Topologies generated by weight minimisation of the two-bar truss structure subject to local stress constraints with the ‘closing-down’ continuation strategy.

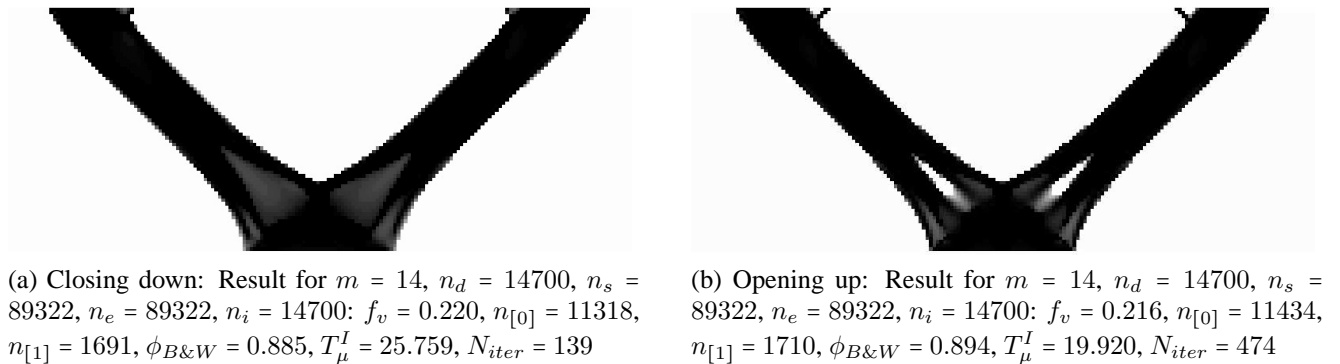
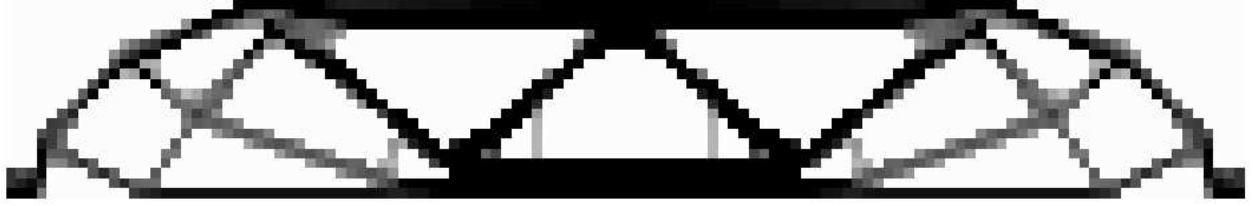


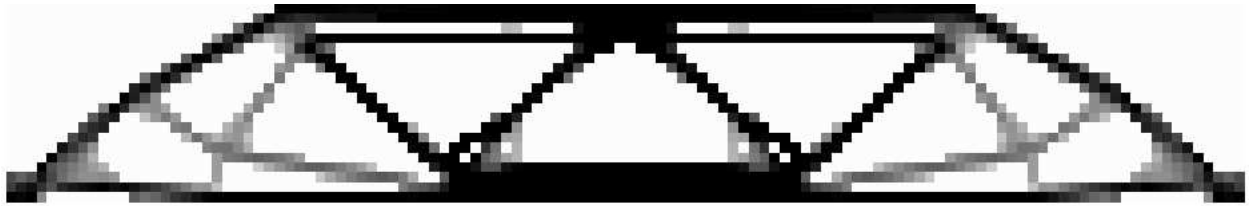
Figure 4.23: Topologies generated for weight minimisation of the two-bar truss ground structure with $m = 14$ subject to local stress constraints with the ‘closing down’ and ‘opening up’ continuation strategies.

Finally the solution procedures described above are applied to the MBB ground structure with $m = 4$. The topologies generated are depicted in Figure 4.24 with some numerical values summarised in the caption of the figures. Clearly the designs are not of good quality. Obtaining good quality solutions to the weight minimisation of the MBB ground structure subject to local stress constraints is well known to be a very difficult problem in numerical optimisation, see for example [10]. The black-and-white fractions of the designs are unsatisfactory and the algorithm required an exceeding amount of iterations to converge. These results highlight once again the problems in numerical scaling of the optimisation problem, especially so considering local stress constraints (and the

relaxation of these constraints).



(a) Closing down: Result for $m = 4$, $n_d = 1200$, $n_s = 7522$, $n_e = 7522$, $n_i = 1200$: $f_v = 0.346$, $n_{[0]} = 720$, $n_{[1]} = 211$, $\phi_{B\&W} = 0.776$, $T_\mu^I = 1.471$, $N_{iter} = 181$



(b) Opening up: Result for $m = 4$, $n_d = 1200$, $n_s = 7522$, $n_e = 7522$, $n_i = 1200$: $f_v = 0.326$, $n_{[0]} = 732$, $n_{[1]} = 172$, $\phi_{B\&W} = 0.753$, $T_\mu^I = 1.581$, $N_{iter} = 554$

Figure 4.24: Topologies generated by weight minimisation of the MBB structure with $m = 4$ subject to local stress constraints using the 'closing-down' and 'opening up' continuation strategies on relaxation.

4.7 Conclusion

The stress constrained problem is well-known to be one of the most complex problems in numerical optimisation. Not only is the problem inherently large scale, with a large number of local constraints, but these constraints suffer from the well documented singularity problem. In general stress constraints define a complex and degenerate feasible domain with the optimal solution inaccessible to classical optimisation algorithms. Therefore it is required that stress constraints be perturbed or relaxed, in some sense, so that the optimal solution can be approached. However, to ensure the solution is physically feasible it is desired that the algorithm converges to the local optima of the original unperturbed problem.

The main advantage of the SAND formulated stress constrained problem is in the reduction of computational requirements. In Chapter 2 we showed that the reduction in computational complexity related to the SAND formulated problem with structural analysis not performed explicitly in each iteration. Furthermore, with the state equation not solved *per se* in each iteration, a value of exactly zero on the lower bound of decision variables poses no particular difficulty. Not only is this an exact representation of void material, but stress constraints can be formulated as *vanishing* constraints, that is the feasibility of the design is not influenced by stress constraints applied to material that attain a state of void. In NAND material and stress constraints can not be removed

completely in this way and the solution to the original unperturbed problem, strictly speaking, cannot be obtained.

In this chapter we show the singularity phenomena is inherent to the stress constrained problem, hence not only a result of the NAND formulation with explicit and exact structural analysis. However, the SAND formulated problem does offer some new avenues in preventing the singularity phenomena. In SAND stress constraints can be expressed as an explicit function of state variables, much simpler and better posed than the implicit reciprocal relation inherent to the NAND formulated case. Furthermore, as mentioned above, the formulation of stress constraints as *vanishing* constraints seems to be a more accurate representation of the fundamental problem, with plenty of scope for future work in how the said constraints are relaxed and scaled numerically.

Finally, it would seem numerical issues are the predominant difficulty considering the SAND formulated stress constrained problem. This could be expected since decision variables are allowed to take on values of exactly zero on the lower bound. However, it would seem numerical scaling of relaxation parameters and material penalisation is the main reason for the said difficulties. Furthermore, fundamentally the topology optimisation problem is a discrete programming problem. That is, strictly speaking, material is only allowed to be either completely solid or void, 1 or 0. The problem is relaxed from a computational point of view and penalised so that, in principle, discrete solutions to the continuous problem can be obtained. However, this is much more difficult to achieve considering the stress constrained case. None of the formulations proposed in this chapter achieved satisfactory black-and-white fractions. Once again, a great deal of future work can be dedicated to this area.

The work in this chapter was introduced with a quote by some of the most widely published authors in the field of topology optimisation; in 2003 Sigmund and Bendsoe stated “*the best way to solve stress constrained problems has probably yet to be suggested*” [8]. This statement, a decade later, still holds true. However, the SAND formulation of the stress constrained problem does relax the severe computational requirements and, it would seem, is a promising avenue to explore the singularity problem further.

Chapter 5

Conclusion

Structural optimisation generalised as a topology optimisation problem typically leads to very large scale problems with severe computational resource requirements. What is more, these problems are computationally complex, involving a variety of assumptions, relaxations and restrictions for a solution to be feasible. In the light of this, although structural optimisation could be an invaluable tool in structural design (and design in general), very few applications have found its way into industry. However, with the rapid advance of computer science and the power of computational devices this becomes more feasible every day. In this thesis we explored the formulation of the topology optimisation problem as a *simultaneous analysis and design* problem. In Chapter 2 we showed that the SAND problem could be considered a more accurate representation of the fundamental problem compared to the traditional nested analysis and design (NAND) case. Furthermore, it was shown, with theoretical arguments, that the SAND problem reduces computational and storage requirements, especially when considering local constraints.

In Chapter 3 the argument for the SAND interpretation is extended to the fact that material can be removed completely from the design space, facilitated with a lower bound of exactly zero on decision variables with little-or-no complications. This is a feature typically not possible in a NAND setting due to the numerical ill-conditioning in the analysis phase. What is more, the complete removal of material causes subproblems to become smaller and easier to solve as the algorithm progresses, reducing computation time. Various material penalisation functions were tested, with the popular simple isotropic material with penalisation (SIMP) approach most effective in a SAND setting. Partly due to the fact that along with the SIMP function value, first- and second-order sensitivities take on values of zero as material takes on a state of void. Interestingly, these properties seem to reduce numerical instabilities. This aspect is also problematic, for once removed, material can not re-enter the design domain. However, including the set of non-linear equality constraints that describe structural analysis in the problem inherently leads to a non-convex problem. That is we can only hope to find a local minima in the first place, highly dependent on the initialisation of variables and other parameters. Furthermore, the quality of solutions considering these simply constrained problems are promising. Especially considering the degree to which purely solid-void designs are approached by direct penalisation of material in the state equation. A preliminary investigation into a continuation strategy on material penalisation (progressive penalisation) is conducted with results encouraging, albeit only in a heuristic sense. Finally, a variety of problem sizes are considered, once again, solution quality is promising. However, plenty of future work can be

dedicated to the design of filters in a SAND setting to alleviate the problem of mesh dependency. In Chapter 4 local stress constraints were investigated in the SAND problem. A well known complication, referred to as the *singularity* problem, was discussed and compared in NAND and SAND settings. The singularity problem is inherently linked to the removal of material from the design domain, hence the lower bound on decision variables form an integral part of this investigation in a SAND setting. We showed that the singularity problem is inherent to the stress constrained problem, for it is a physical phenomena, related to the fact that stress in void material is undefined. In other words, stress constraints applied to material that attains a state of void should no longer contribute to the feasibility of the design. In a NAND setting, with a non-zero lower bound on decision variables the problem is ambiguous and the singular solution does not form part of the design domain. A zero lower bound on decision variables in a SAND setting allows stress constraints to be formulated as ‘vanishing’ constraints. This formulation allows the complete removal of constraints in a mathematically consistent manner. However, due to the interior point method subsolver employed in the SAOi algorithm ϵ -relaxation is required to ensure the design domain is completely feasible approaching the singular solution. The work presented on this topic is by no means complete and we speculate the SAND formulation of the problem might in the future be employed to completely negate the singularity phenomena.

Finally, throughout this thesis we argued that the SAND formulated problem, although in some respects more complex than the equivalent NAND problem, does reduce computational requirements in general, even if only considering that structural analysis is not performed explicitly in each iteration. Furthermore, in Chapter 4 we showed that the local stress constrained problem, traditionally requiring severe amounts of computational resources, could scale linearly in computation time in a SAND setting. This is by no means a general statement, for this could only be achieved for a specific problem, although scaling of this nature is very promising.

The unification of *analysis* and *design*, or state space and design space in the optimisation problem, although only considered in the context of structural optimisation in this thesis, could be considered an universal technique to solve a wide range of problems. As discussed in Chapter 1, the unification of partial differential equations and numerical optimisation can be applied to any physical system. Throughout this thesis we show that such an unification could lead to a more accurate representation of the physical problem and a massive reduction in computational requirements.

References

- [1] M.P. Bendsøe. Optimal shape design as a material distribution problem. *Struct. Optim.*, 1:193–202, 1989.
- [2] P. Villaggio. Sixty years of solid mechanics. *Meccanica*, 46(6):1171–1189, 2011.
- [3] 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.
- [4] G. Cheng and X. Guo. A note on a jellyfish-like feasible domain in structural topology optimization. *Engineering optimization*, 31(1):1–24, 1998.
- [5] J. Zowe, M. Kočvara, and M.P. Bendsøe. Free material optimization via mathematical programming. *Mathematical programming*, 79(1-3):445–466, 1997.
- [6] M. Kocvara and M. Stingl. Free material optimization for stress constraints. *Structural and Multidisciplinary Optimization*, 33(4):323–336, 2007.
- [7] 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.
- [8] M.P. Bendsøe and O. Sigmund. *Topology optimization: Theory, methods and applications*. Springer, Berlin, 2003.
- [9] G. Cheng and Z. Jiang. Study on topology optimization with stress constraints. *Engineering Optimization*, 20(2):129–148, 1992.
- [10] D.W. Wood. *Dual sequential approximation methods in structural optimization*. PhD thesis, University of Stellenbosch, Stellenbosch, South-Africa, 2012.
- [11] O. Sigmund and J. Peterson. Numerical instabilities in topology optimization: A survey on procedures dealing with checkerboards, mesh-dependencies and local minima. *Struct. Optim.*, 16:68–75, 1998.
- [12] M.P. Bendsøe and O. Sigmund. Material interpolation schemes in topology optimization. *Archive of Applied Mechanics*, 69(9-10):635–654, 1999.
- [13] T.E. Bruns. A reevaluation of the SIMP method with filtering and an alternative formulation for solidvoid topology optimization. *Structural and Multidisciplinary Optimization*, 30(6):428–436, 2005.

- [14] M. Stolpe and K. Svanberg. On the trajectories of penalization methods for topology optimization. *Structural and Multidisciplinary Optimization*, 21(2):128–139, 2001.
- [15] J.S. Arora and Q. Wang. Review of formulations for structural and mechanical system optimization. *Structural and Multidisciplinary Optimization*, 30, 4:251 – 272, 2005.
- [16] I. Koutis, G.L. Miller, and R. Peng. Approaching optimality for solving sdd linear systems. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*, pages 235–244. IEEE, 2010.
- [17] C. Fleury. Structural optimisation methods for large scale problems: Computational time issues. In *8th World Congress on Structural and Multidisciplinary Optimisation*, Lisbon, Portugal, June 2009. WCSMO.
- [18] U. Kirsch. On singular topologies in optimum structural design. *Structural Optimization*, 2(3):133–142, 1990.
- [19] S.S. Rao. *Engineering optimization: Theory and practice*. John Wiley & Sons, 2009.
- [20] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, Cambridge, UK, 7th edition, 2004.
- [21] A.A. Groenwold, L.F.P. Etman, and S. Tosserams. Globally convergent optimization algorithm using conservative convex separable diagonal quadratic approximations. *AIAA J.*, 47:2649–2657, 2009.
- [22] A.A. Groenwold, L.F.P. Etman, J.A. Snyman, and J.E. Rooda. Incomplete series expansion for function approximation. *Structural and Multidisciplinary Optimization*, 34(1):21–40, 2007.
- [23] P. Duysinx and M.P. Bendsøe. Topology optimization of continuum structures with local stress constraints. *Int. J. Numer. Meth. Enging.*, 43:1453–1478, 1998.
- [24] A.G. Weldeyesus M. Stolpe, K. Marmaras. The variable thickness sheet problem revisited. In *10th World Congress on Structural and Multidisciplinary Optimization*, Orlando, Florida, USA, May 2013. Oral presentation.
- [25] A.A. Groenwold. Positive definite separable quadratic programs for non-convex problems. *Structural and Multidisciplinary Optimization*, 46(6):795–802, 2012.
- [26] T.E. Bruns. Zero density lower bounds in topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 196(13):566 – 578, 2006.
- [27] S. Sankaranarayanan, R.T. Haftka, and R.K. Kapania. Truss topology optimization with simultaneous analysis and design. *AIAA journal*, 32(2):420–424, 1994.
- [28] M.P. Bendse, A. Ben-Tal, and J. Zowe. Optimization methods for truss geometry and topology design. *Structural optimization*, 7(3):141–159, 1994.

- [29] W. Achtziger and C. Kanzow. Mathematical programs with vanishing constraints: optimality conditions and constraint qualifications. *Mathematical Programming*, 114(1):69–99, 2008.
- [30] Z. Hashin and S. Shtrikman. A variational approach to the theory of the elastic behaviour of multiphase materials. *Journal of the Mechanics and Physics of Solids*, 11(2):127 – 140, 1963.
- [31] M.P. Bendsøe and O. Sigmund. Material interpolation schemes in topology optimization. *Archive of applied mathematics*, 69:635–654, 1999.
- [32] O. Sigmund. Tailoring materials with prescribed elastic properties. *Mechanics of Materials*, 20(4):351–368, 1995.
- [33] M. Stolpe and K. Svanberg. An alternative interpolation scheme for minimum compliance topology optimization. *Structural and Multidisciplinary Optimization*, 22(2):116–124, 2001.
- [34] A. Rietz. Sufficiency of a finite exponent in SIMP (power law) methods. *Structural and Multidisciplinary Optimization*, 21(2):159–163, 2001.
- [35] J.M. Martinez. A note on the theoretical convergence properties of the SIMP method. *Structural and Multidisciplinary Optimization*, 29(4):319–323, 2005.
- [36] G. Sved and Z. Ginos. Structural optimization under multiple loading. *International Journal of Mechanical Sciences*, 10(10):803 – 805, 1968.
- [37] G. Xu and C. Gengdong. Epsilon-continuation approach for truss topology optimization. *Acta Mechanica Sinica*, 20(5):526–533, 2004.
- [38] R.T. Haftka. Simultaneous analysis and design. *AIAA journal*, 23(7):1099–1103, 1985.
- [39] G.D. Cheng and X. Guo. ϵ -relaxed approach in structural topology optimization. *Struct. Optim.*, 13:258–266, 1997.
- [40] W. Achtziger, C. Kanzow, and T. Hoheisel. On a relaxation method for mathematical programs with vanishing constraints. *GAMM-Mitteilungen*, 35(2):110–130, 2012.
- [41] D.L. Logan. *A first course in the finite element method*. Global Engineering, Stamford, USA, 2011.
- [42] A. Diaz and O. Sigmund. Checkerboard patterns in layout optimization. *Structural optimization*, 10(1):40–45, 1995.
- [43] P.P. Benham, R.J. Crawford, and C.G. Armstrong. *Mechanics of Engineering Materials*. Longman Group, 1996.

Appendix A

The direct stiffness method

Considered the most common implementation of the finite element method, the direct stiffness method, as the name suggests, is rooted in problems of structural analysis. However, non-structural boundary value problems (BVP's) of fluid flow, heat transfer and electrical networks are also solved by applying the appropriate physical law (Fourier's law, Darcy's law and Ohm's law respectively) and a conservation principle (usually the conservation of energy). This is also referred to as the *matrix stiffness method*, involving a large number of repetitive matrix operations, making it especially well suited to computer aided analysis¹.

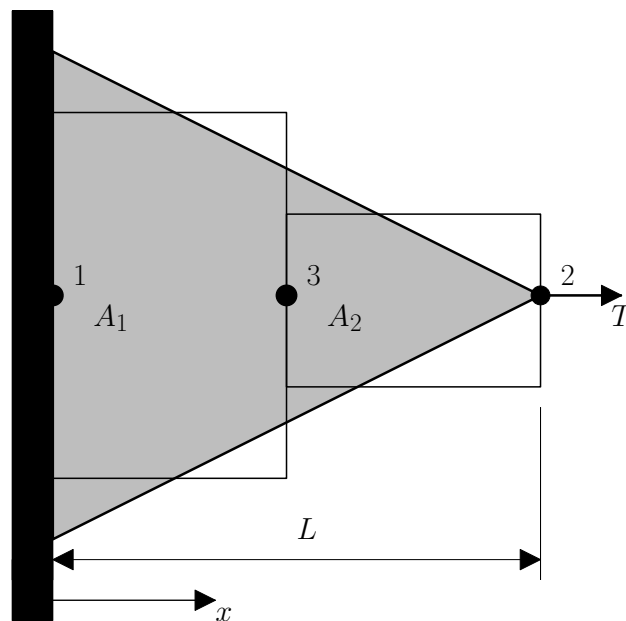


Figure A.1: 1-D example of a structure to introduce the FE method

To illustrate the FE method consider the structure depicted in Figure A.1. Although fairly simple, this example illustrates most principles underlying the FE method with little loss of generality when considering more complex systems. We present only a summary of the basic concepts, the interested reader is referred to [41] for a complete description of the method.

¹Most of the work or *exhaustion*, as Archimedes put it, is dealt with by the computer.

Assume the structure only resists forces in the direction parallel to the applied force, rendering the responses of the structure one dimensional. As an initial approximation to the problem the structure is discretised into two bar elements, as illustrated in Figure A.1. Assume all material obeys *Hooke's law*, which states force and displacement are proportional by a constant k , the material stiffness.

To develop the FE formulation consider the generic bar element with local coordinate system x as depicted in Figure A.2. As building block of the FE method the *elemental stiffness matrix*, denoted by K_e , relates forces to displacements in the local coordinate system by

$$\mathbf{f}_e = K_e \mathbf{d}_e \quad (\text{A.1})$$

where the elemental *degrees of freedom* determine the dimension of matrix K_e and vectors $\mathbf{f}_e, \mathbf{d}_e$.

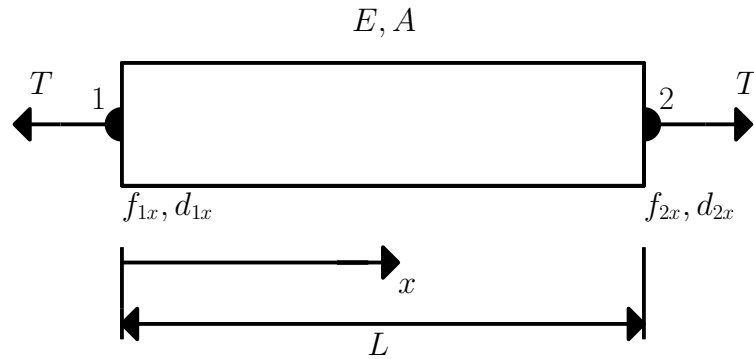


Figure A.2: Bar element

In this case the element has two degrees of freedom (two nodes with displacement in a single direction) resulting in the following local stiffness relation

$$\begin{Bmatrix} f_{1x} \\ f_{2x} \end{Bmatrix} = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} \begin{Bmatrix} d_{1x} \\ d_{2x} \end{Bmatrix} \quad (\text{A.2})$$

where k_{ij} represents the force f_i in the i^{th} degree of freedom resulting in a unit displacement d_j in the j^{th} degree of freedom while all other displacements are zero. To obtain the unknown displacements (solving the system of equations) constants k_{ij} are precalculated in terms of quantities derived from the elemental geometry and material stiffness.

Now a function $u(x)$ is chosen to represent the deformation of the element under loading (referred to as the *shape function*). We wish to express this relation as an interpolation between nodal displacements such that boundary conditions (and forces) can be applied directly. In this case linear shape functions are employed, an exact local representation of responses, since we assume a linear material law. However, the geometry of the elements do not coincide with the geometry of the structure, hence the solution is approximate. In terms of shape functions N_i we define

$$u = [N_1 \ N_2] \begin{Bmatrix} d_{1x} \\ d_{2x} \end{Bmatrix}, \quad (\text{A.3})$$

where

$$N_1 = 1 - \frac{x}{L} \quad \text{and} \quad N_2 = \frac{x}{L}, \quad (\text{A.4})$$

with N_i such that a value of unity is obtained at node i and zero at all other nodes.

Next we define the strain-displacement and stress-strain relations. Consider the tensile force T , resulting in a deformation of δ . The strain-displacement relation is defined as

$$\epsilon = \frac{\delta}{L} = \frac{d_{2x} - d_{1x}}{L}, \quad (\text{A.5})$$

and the stress-strain relation, according to Hooke's law is given by

$$\sigma = \frac{T}{A} = E\epsilon, \quad (\text{A.6})$$

with E denoting the Young's modulus of the material. Now, using equations (A.5) and (A.6), we arrive at

$$T = \frac{AE(d_{2x} - d_{1x})}{L}, \quad (\text{A.7})$$

which relates elemental force and nodal displacements. In other words we are approaching the relation initially defined by the *elemental stiffness matrix*.

Finally the elemental stiffness matrix is derived by considering nodal forces and force equilibrium such that

$$f_{1x} = -T \quad \text{and} \quad f_{2x} = T. \quad (\text{A.8})$$

Employing equation (A.7) we obtain

$$T = -f_{1x} = \frac{AE}{L}(d_{2x} - d_{1x}), \quad (\text{A.9})$$

$$T = f_{2x} = \frac{AE}{L}(d_{2x} - d_{1x}).$$

Simplifying equation (A.9) yields

$$\begin{aligned} f_{1x} &= -k(d_{2x} - d_{1x}), \\ f_{2x} &= k(d_{2x} - d_{1x}), \end{aligned} \quad (\text{A.10})$$

with $k = \frac{AE}{L}$. Equation (A.10) can also be expressed as a matrix equation

$$\begin{Bmatrix} f_{1x} \\ f_{2x} \end{Bmatrix} = \begin{bmatrix} k & -k \\ -k & k \end{bmatrix} \begin{Bmatrix} d_{1x} \\ d_{2x} \end{Bmatrix}, \quad (\text{A.11})$$

For element 1 we have

$$\begin{Bmatrix} f_{1x}^{(1)} \\ f_{3x}^{(1)} \end{Bmatrix} = \frac{A^{(1)}E}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} d_{1x}^{(1)} \\ d_{3x}^{(1)} \end{Bmatrix}, \quad (\text{A.12})$$

and for element 2 we have

$$\begin{Bmatrix} f_{3x}^{(2)} \\ f_{2x}^{(2)} \end{Bmatrix} = \frac{A^{(2)}E}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{Bmatrix} d_{3x}^{(2)} \\ d_{2x}^{(2)} \end{Bmatrix}, \quad (\text{A.13})$$

where the superscripts denote element number and subscripts node number. Note in our example the only difference between elements is their areas A_1 and A_2 . However, it would be of little extra effort to consider elements with different material properties and/or geometries (the length of the element in this case).

To perform structural analysis we are required to solve equations (A.12) and (A.13) simultaneously. This requires the local stiffness relations to be assembled in a global stiffness relation. Elements should remain connected at node 3, referred to as the *continuity* or *compatibility requirement*

$$d_{3x}^{(1)} = d_{3x}^{(2)} = d_{3x}, \quad (\text{A.14})$$

which along with the principle of force equilibrium yields the global stiffness matrix. However, we consider a more convenient method to assemble the local stiffness equations, called the *direct stiffness method*². In short, local degrees of freedom are mapped to global degrees of freedom and elemental stiffness equations superimposed to form the global stiffness equation. Expanding the stiffness equation of element 1 to global coordinates we obtain

$$\begin{matrix} & d_{1x} & d_{2x} & d_{3x} \\ \begin{Bmatrix} f_{1x}^{(1)} \\ f_{2x}^{(1)} \\ f_{3x}^{(1)} \end{Bmatrix} & = & \frac{A^{(1)}E}{L} \begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix} & \begin{Bmatrix} d_{1x}^{(1)} \\ d_{2x}^{(1)} \\ d_{3x}^{(1)} \end{Bmatrix} \end{matrix} \cdot \quad (\text{A.15})$$

Similarly for element 2 we have

$$\begin{matrix} & d_{1x} & d_{2x} & d_{3x} \\ \begin{Bmatrix} f_{1x}^{(2)} \\ f_{2x}^{(2)} \\ f_{3x}^{(2)} \end{Bmatrix} & = & \frac{A^{(2)}E}{L} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{bmatrix} & \begin{Bmatrix} d_{1x}^{(2)} \\ d_{2x}^{(2)} \\ d_{3x}^{(2)} \end{Bmatrix} \end{matrix} \cdot \quad (\text{A.16})$$

Considering force equilibrium at each node yields

$$\begin{Bmatrix} f_{1x}^{(1)} \\ 0 \\ f_{3x}^{(1)} \end{Bmatrix} + \begin{Bmatrix} 0 \\ f_{2x}^{(2)} \\ f_{3x}^{(2)} \end{Bmatrix} = \begin{Bmatrix} F_{1x} \\ F_{3x} \\ F_{3x} \end{Bmatrix} \cdot \quad (\text{A.17})$$

Using equations (A.15), (A.16) and (A.17) we arrive at the global stiffness equation in matrix form

$$\begin{Bmatrix} F_{1x} \\ F_{3x} \\ F_{3x} \end{Bmatrix} = \frac{A^{(2)}E}{L} \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 2 \end{bmatrix} \begin{Bmatrix} d_{1x} \\ d_{2x} \\ d_{3x} \end{Bmatrix}, \quad (\text{A.18})$$

also referred to as the *set of equilibrium equations* or the *equation of state*. The condition that node 1 remains fixed, the single boundary condition on the problem, is applied by fixing $d_{1x} = 0$.

The solution of the state equation yields only responses at nodes 2 and 3 with a linear interpolation assumed intermittently. To approximate the problem ever more accurately finer and finer

²The term is also used to refer to the method as a whole.

mesh discretisations can be employed, as illustrated in Figure A.3. However, an increase in accuracy is proportional to an increase in problem size which in turn is proportional to an increase in computational resource requirements.

The state equation of a structure discretised with an arbitrary sized mesh is denoted by

$$K\mathbf{q} = \mathbf{f}, \quad (\text{A.19})$$

where $K \in \mathbb{R}^{n \times n}$ is the global stiffness matrix, the global nodal displacements $\mathbf{q} \in \mathbb{R}^n$ and global nodal forces $\mathbf{f} \in \mathbb{R}^n$ with n the total degrees of freedom (number of nodes times degrees of freedom per node, neglecting constrained nodes due to the application of boundary conditions).

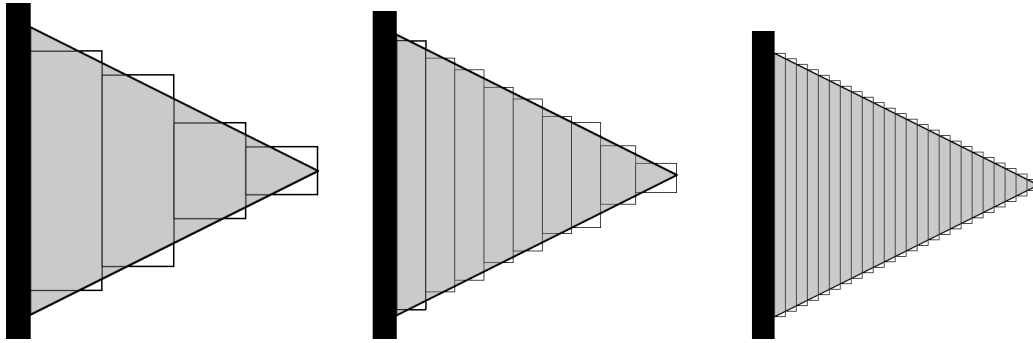


Figure A.3: Finite element discretisation

In this thesis we consider 2-D structures subject to plane-stress. The domain is discretised into Q8 displacement based square elements throughout. Local ($s - t$) and global ($x - y$) coordinate systems are defined as per Figure A.4. In other words, each element is defined with 8 nodes and each node possesses two degrees of freedom, one vertical and one horizontal. This results in a 2-D problem with a total of 16 degrees of freedom per element. Square Q4 elements, with 8 degrees of freedom, are not employed due to the tendency of higher-order elements to alleviate the problem of checkerboarding [42], a problem in topology optimisation when a structure acquires artificially high stiffness due to a crude approximation of local displacements (and strains).

Here we discuss some basic concepts of higher-order elements in 2-D relevant to this work, the reader is once again referred to the comprehensive description in [41]. The stress/strain relation for plane-stress, depicted in Figure A.5, is defined as

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = D \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix}, \quad (\text{A.20})$$

with normal stresses denoted by σ_i and shear stress τ_{xy} . The stress-strain matrix is then defined as

$$D = \frac{E}{1 - \nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}, \quad (\text{A.21})$$

with Poisson's ratio and Young's modulus denoted by ν and E respectively.

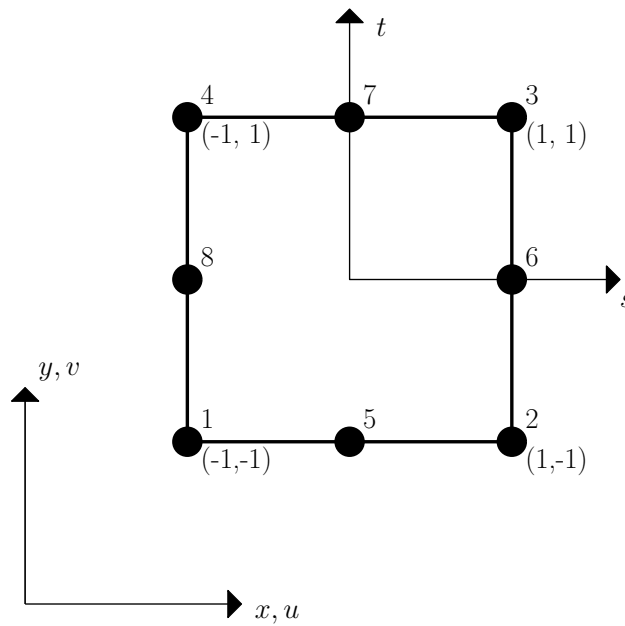


Figure A.4: Q8 element

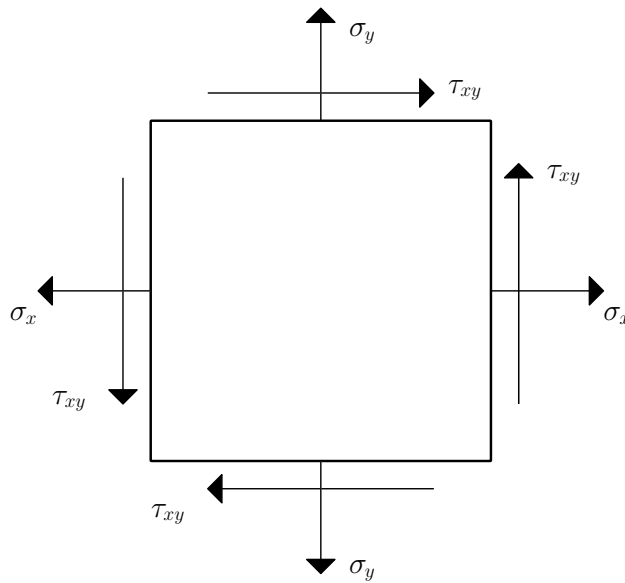


Figure A.5: Plane stress

Applying the principle of minimum potential energy an expression for the elemental stiffness matrix is obtained

$$K_e = \int \int_A [B]^T D [B] h dx dy, \tag{A.22}$$

where A denotes the area of the element in the $x - y$ plane and h the material thickness normal to the plane. Transforming the integral to the local coordinate system, in which integration is much simpler, we obtain

$$K_e = \int_{-1}^1 \int_{-1}^1 [B]^T D [B] |J| h ds dt, \tag{A.23}$$

where the determinant $|J|$ relates an infinitesimal area in the $x - y$ coordinate system to the $s - t$ coordinate system. Therefore

$$J = \begin{bmatrix} \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \\ \frac{\partial x}{\partial t} & \frac{\partial y}{\partial t} \end{bmatrix}. \quad (\text{A.24})$$

The operator matrix $[B]$ relates elemental displacement and strain through

$$\begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix} = \frac{1}{|J|} \begin{bmatrix} \frac{\partial y}{\partial t} \frac{\partial(\cdot)}{\partial s} - \frac{\partial y}{\partial s} \frac{\partial(\cdot)}{\partial t} & 0 \\ 0 & \frac{\partial x}{\partial s} \frac{\partial(\cdot)}{\partial t} - \frac{\partial y}{\partial s} \frac{\partial(\cdot)}{\partial s} \\ \frac{\partial x}{\partial s} \frac{\partial(\cdot)}{\partial t} - \frac{\partial y}{\partial s} \frac{\partial(\cdot)}{\partial s} & \frac{\partial y}{\partial t} \frac{\partial(\cdot)}{\partial s} - \frac{\partial y}{\partial s} \frac{\partial(\cdot)}{\partial t} \end{bmatrix} \begin{Bmatrix} u \\ v \end{Bmatrix}, \quad (\text{A.25})$$

where u and v are quadratic functions in s and t interpolated between nodal displacement u_1, u_2, \dots, u_8 and v_1, v_2, \dots, v_8 .

Appendix B

Plane stress analysis

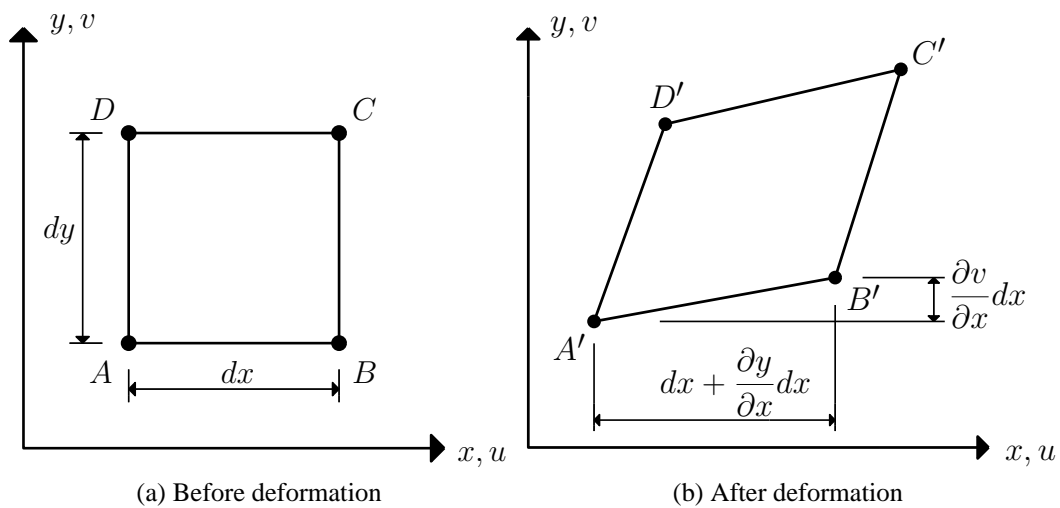


Figure B.1: Differential isotropic element

Consider the 2-D isotropic differential element depicted before and after deformation in Figure B.1a and B.1b. Applying an external load will result in a deformation or change in the shape of the body, referred to as *strain*. Normal strain (or engineering normal strain) is defined as the change in length divided by the original length of an imaginary line in the structure [41]. Referring to Figure B.1, strain in the x -direction is defined as

$$\epsilon_x = \frac{A'B' - AB}{AB} . \quad (\text{B.1})$$

Considering a differential distance $AB = dx$ and the Pythagorean triangle we get

$$(A'B')^2 = \left(dx + \frac{\partial u}{\partial x} dx \right)^2 + \left(\frac{\partial v}{\partial x} dx \right)^2 , \quad (\text{B.2})$$

which, assuming small strains and neglecting lower order terms, yields

$$A'B' = dx + \frac{\partial u}{\partial x} dx . \quad (\text{B.3})$$

Using equation (B.2) and (B.3) in (B.1) we obtain the differential expression for strain in the x -direction

$$\epsilon_x = \frac{\partial u}{\partial x}. \quad (\text{B.4})$$

Similarly, considering line AD we have

$$\epsilon_y = \frac{\partial v}{\partial y} \quad (\text{B.5})$$

in the y -direction. Shear strain is defined as the change in angle between two imaginary lines in the structure. Consistent with the assumption of small strains the change in angle DAB is defined as

$$\gamma_{xy} = \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}. \quad (\text{B.6})$$

Hooke's law states that a stress in the positive x direction produces a positive strain. That is,

$$\epsilon_x = \frac{\sigma_x}{E}, \quad (\text{B.7})$$

with E the modulus of elasticity. A positive stress in the y direction produces a negative strain in the x direction according to

$$\epsilon_x = -\nu \frac{\sigma_y}{E}, \quad (\text{B.8})$$

with ν Poisson's ratio. Under the assumption of plane stress and employing the principle of superposition we arrive at the expression for total stress in the x direction

$$\epsilon_x = \frac{\sigma_x}{E} - \nu \frac{\sigma_y}{E} \quad (\text{B.9})$$

and similarly in the y direction

$$\epsilon_y = \frac{\sigma_y}{E} - \nu \frac{\sigma_x}{E} \quad (\text{B.10})$$

and finally solving for stresses we obtain

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = [D] \begin{Bmatrix} \epsilon_x \\ \epsilon_y \\ \gamma_{xy} \end{Bmatrix}, \quad (\text{B.11})$$

with

$$D = \frac{E}{1-\nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}. \quad (\text{B.12})$$

Throughout this thesis structural analysis is performed via the finite element (FE) method as discussed in Chapter 2. Elemental deformation is defined by shape functions $u(x)$ and $v(y)$ with strain calculated per equations (B.4), (B.5) and (B.6). We consider linear shape functions interpolated between nodal displacements resulting in the following expression for stress in the center of the element

$$\boldsymbol{\sigma} = \begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = [D(E, \nu)] \begin{Bmatrix} \frac{u_6 - u_8}{L} \\ \frac{v_7 - v_5}{L} \\ \frac{u_7 - u_5}{L} + \frac{v_6 - v_8}{L} \end{Bmatrix} \quad (\text{B.13})$$

with the plane axes and nodal coordinates depicted in Figure A.4. The stress-strain matrix is denoted by $[D(E, \nu)]$ to emphasize it as a function of material properties E and ν . In SAND nodal displacements or *state* variables are defined as independent design variables. Therefore, in general, the stress state of an element is an explicit function of material properties and state variables

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}(E, \nu, \mathbf{q}) . \quad (\text{B.14})$$

The load at which material fails is determined in accordance with the von Mises failure criterion (or maximum distortion energy criterion [43]) and material yield stress as per

$$\sigma_{vm}(E, \nu, \mathbf{q}) = \sqrt{\sigma_x^2 - \sigma_x \sigma_y + \sigma_y^2 + 3\tau_{xy}^2} \leq \sigma_y . \quad (\text{B.15})$$

Squaring equation (B.15) to simplify the analytic form of first - and second order derivatives and rewriting as an inequality constraint we arrive at the basic form of stress constraint considered in this thesis, i.e.

$$g_i(E, \nu, \sigma_y, \mathbf{q}_{elmi}) = \left[\frac{\sigma_{vm}(E, \nu, \mathbf{q}_{elmi})}{\sigma_y} \right]^2 - 1 \leq 0 , \quad (\text{B.16})$$

with $i = 1, 2, \dots, n$ and n the number of elements in the FE mesh. Constraint $g_i(E, \nu, \sigma_y, \mathbf{q}_{elmi})$ is a function of material properties and a subset of the state variable vector $\mathbf{q}_{elmi} \subset \mathbf{q}$, where \mathbf{q}_{elmi} is the collection of nodal displacements related to element i . With regards to squaring of the constraint, clearly this does not effect the feasible set, and the same is done in some recent work on the stress constrained problem, see for example [40].

Note that if material thickness is viewed as decision variables, stress and stress constraints remain only a function of state variables. As discussed in Chapter 3 a popular interpretation of decision variables is referred to as the *homogenization* approach [3]. In contrast to the material thickness view, decision variables describe the density of a porous material which mimic special composite structures under certain conditions. In this view, decision variables parametrise material density which is deemed to decrease material stiffness and increase local stresses as the material becomes more porous [23]; as decision variables approach zero. Material stiffness is modified according to $E x_e^p$ and local stresses via a reciprocal term $\frac{1}{x_e^q}$. Local stresses are then written in a form that resembles

$$\sigma_i = \frac{x_e^p}{x_e^q} E \epsilon_i . \quad (\text{B.17})$$

In [23] the authors construct a rank 2 composite material to mimic the behaviour of the porous solid-void material. From a limit analysis of stresses in the composite material they deduce local stresses should remain finite and non-zero at zero density. The only choice of q that satisfies this requirement, the ‘‘coherency requirement’’, is $p = q$. That is, the term $\frac{x_e^p}{x_e^q} = 1$ and stress constraints are again independent of decision variables, resulting in exactly the same expression as in the material thickness view of decision variables.

Appendix C

Q8 elemental stiffness

Here we present the FORTRAN code for the symmetric matrix \bar{K}_e where nu denotes ν .

```

ke(1,1) = -13.d0/45.d0*nu+13.d0/15.d0
ke(2,1) = 17.d0/72.d0*nu + 17.d0/72.d0
ke(3,1) = -17.d0/180.d0*nu+73.d0/180.d0
ke(4,1) = -1.d0/8.d0*nu+1.d0/24.d0
ke(5,1) = -23.d0/180.d0*nu+23.d0/60.d0
ke(6,1) = 7.d0/72.d0*nu+7.d0/72.d0
ke(7,1) = -7.d0/45.d0*nu+31.d0/90.d0
ke(8,1) = 1.d0/8.d0*nu-1.d0/24.d0
ke(9,1) = -1.d0/30.d0*nu-77.d0/90.d0
ke(10,1) = 7.d0/18.d0*nu-5.d0/18.d0
ke(11,1) = 2.d0/9.d0*nu-13.d0/45.d0
ke(12,1) = -1.d0/18.d0*nu-1.d0/18.d0
ke(13,1) = 1.d0/30.d0*nu-43.d0/90.d0
ke(14,1) = -1.d0/18.d0*nu-1.d0/18.d0
ke(15,1) = 4.d0/9.d0*nu-17.d0/45.d0
ke(16,1) = -11.d0/18.d0*nu+1.d0/18.d0

ke(2,2) = -13.d0/45.d0*nu+13.d0/15.d0
ke(3,2) = 1.d0/8.d0*nu-1.d0/24.d0
ke(4,2) = -7.d0/45.d0*nu+31.d0/90.d0
ke(5,2) = 7.d0/72.d0*nu+7.d0/72.d0
ke(6,2) = -23.d0/180.d0*nu+23.d0/60.d0
ke(7,2) = -1.d0/8.d0*nu+1.d0/24.d0
ke(8,2) = -17.d0/180.d0*nu+73.d0/180.d0
ke(9,2) = -11.d0/18.d0*nu+1.d0/18.d0
ke(10,2) = 4.d0/9.d0*nu-17.d0/45.d0
ke(11,2) = -1.d0/18.d0*nu-1.d0/18.d0
ke(12,2) = 1.d0/30.d0*nu-43.d0/90.d0
ke(13,2) = -1.d0/18.d0*nu-1.d0/18.d0

```

APPENDIX C. Q8 ELEMENTAL STIFFNESS

100

$$\begin{aligned}
ke(14,2) &= 2.d0/9.d0*nu-13.d0/45.d0 \\
ke(15,2) &= 7.d0/18.d0*nu-5.d0/18.d0 \\
ke(16,2) &= -1.d0/30.d0*nu-77.d0/90.d0 \\
\\
ke(3,3) &= -13.d0/45.d0*nu+13.d0/15.d0 \\
ke(4,3) &= -17.d0/72.d0*nu-17.d0/72.d0 \\
ke(5,3) &= -7.d0/45.d0*nu+31.d0/90.d0 \\
ke(6,3) &= -1.d0/8.d0*nu+1.d0/24.d0 \\
ke(7,3) &= -23.d0/180.d0*nu+23.d0/60.d0 \\
ke(8,3) &= -7.d0/72.d0*nu-7.d0/72.d0 \\
ke(9,3) &= -1.d0/30.d0*nu-77.d0/90.d0 \\
ke(10,3) &= -7.d0/18.d0*nu+5.d0/18.d0 \\
ke(11,3) &= 4.d0/9.d0*nu-17.d0/45.d0 \\
ke(12,3) &= 11.d0/18.d0*nu-1.d0/18.d0 \\
ke(13,3) &= 1.d0/30.d0*nu-43.d0/90.d0 \\
ke(14,3) &= 1.d0/18.d0*nu+1.d0/18.d0 \\
ke(15,3) &= 2.d0/9.d0*nu-13.d0/45.d0 \\
ke(16,3) &= 1.d0/18.d0*nu+1.d0/18.d0 \\
\\
ke(4,4) &= -13.d0/45.d0*nu+13.d0/15.d0 \\
ke(5,4) &= 1.d0/8.d0*nu-1.d0/24.d0 \\
ke(6,4) &= -17.d0/180.d0*nu+73.d0/180.d0 \\
ke(7,4) &= -7.d0/72.d0*nu-7.d0/72.d0 \\
ke(8,4) &= -23.d0/180.d0*nu+23.d0/60.d0 \\
ke(9,4) &= 11.d0/18.d0*nu-1.d0/18.d0 \\
ke(10,4) &= 4.d0/9.d0*nu-17.d0/45.d0 \\
ke(11,4) &= -7.d0/18.d0*nu+5.d0/18.d0 \\
ke(12,4) &= -1.d0/30.d0*nu-77.d0/90.d0 \\
ke(13,4) &= 1.d0/18.d0*nu+1.d0/18.d0 \\
ke(14,4) &= 2.d0/9.d0*nu-13.d0/45.d0 \\
ke(15,4) &= 1.d0/18.d0*nu+1.d0/18.d0 \\
ke(16,4) &= 1.d0/30.d0*nu-43.d0/90.d0 \\
\\
ke(5,5) &= -13.d0/45.d0*nu+13.d0/15.d0 \\
ke(6,5) &= 17.d0/72.d0*nu+17.d0/72.d0 \\
ke(7,5) &= -17.d0/180.d0*nu+73.d0/180.d0 \\
ke(8,5) &= -1.d0/8.d0*nu+1.d0/24.d0 \\
ke(9,5) &= 1.d0/30.d0*nu-43.d0/90.d0 \\
ke(10,5) &= -1.d0/18.d0*nu -1.d0/18.d0 \\
ke(11,5) &= 4.d0/9.d0*nu-17.d0/45.d0 \\
ke(12,5) &= -11.d0/18.d0*nu+1.d0/18.d0 \\
ke(13,5) &= -1.d0/30.d0*nu-77.d0/90.d0 \\
ke(14,5) &= 7.d0/18.d0*nu -5.d0/18.d0 \\
ke(15,5) &= 2.d0/9.d0*nu - 13.d0/45.d0 \\
ke(16,5) &= -1.d0/18.d0*nu-1.d0/18.d0
\end{aligned}$$

APPENDIX C. Q8 ELEMENTAL STIFFNESS

101

$$\begin{aligned}
ke(6,6) &= -13.d0/45.d0*nu+13.d0/15.d0 \\
ke(7,6) &= 1.d0/8.d0*nu-1.d0/24.d0 \\
ke(8,6) &= -7.d0/45.d0*nu+31.d0/90.d0 \\
ke(9,6) &= -1.d0/18.d0*nu-1.d0/18.d0 \\
ke(10,6) &= 2.d0/9.d0*nu - 13.d0/45.d0 \\
ke(11,6) &= 7.d0/18.d0*nu - 5.d0/18.d0 \\
ke(12,6) &= -1.d0/30.d0*nu - 77.d0/90.d0 \\
ke(13,6) &= -11.d0/18.d0*nu+1.d0/18.d0 \\
ke(14,6) &= 4.d0/9.d0*nu -17.d0/45.d0 \\
ke(15,6) &= -1.d0/18.d0*nu - 1.d0/18.d0 \\
ke(16,6) &= 1.d0/30.d0*nu-43.d0/90.d0 \\
\\
ke(7,7) &= -13.d0/45.d0*nu+13.d0/15.d0 \\
ke(8,7) &= -17.d0/72.d0*nu-17.d0/72.d0 \\
ke(9,7) &= 1.d0/30.d0*nu-43.d0/90.d0 \\
ke(10,7) &= 1.d0/18.d0*nu+1.d0/18.d0 \\
ke(11,7) &= 2.d0/9.d0*nu-13.d0/45.d0 \\
ke(12,7) &= 1.d0/18.d0*nu+1.d0/18.d0 \\
ke(13,7) &= -1.d0/30.d0*nu-77.d0/90.d0 \\
ke(14,7) &= -7.d0/18.d0*nu+5.d0/18.d0 \\
ke(15,7) &= 4.d0/9.d0*nu - 17.d0/45.d0 \\
ke(16,7) &= 11.d0/18.d0*nu-1.d0/18.d0 \\
\\
ke(8,8) &= -13.d0/45.d0*nu+13.d0/15.d0 \\
ke(9,8) &= 1.d0/18.d0*nu+1.d0/18.d0 \\
ke(10,8) &= 2.d0/9.d0*nu-13.d0/45.d0 \\
ke(11,8) &= 1.d0/18.d0*nu+1.d0/18.d0 \\
ke(12,8) &= 1.d0/30.d0*nu-43.d0/90.d0 \\
ke(13,8) &= 11.d0/18.d0*nu-1.d0/18.d0 \\
ke(14,8) &= 4.d0/9.d0*nu-17.d0/45.d0 \\
ke(15,8) &= -7.d0/18.d0*nu+5.d0/18.d0 \\
ke(16,8) &= -1.d0/30.d0*nu-77.d0/90.d0 \\
\\
ke(9,9) &= -4.d0/15.d0*nu+92.d0/45.d0 \\
ke(10,9) &= 0.d0 \\
ke(11,9) &= 0.d0 \\
ke(12,9) &= -2.d0/9.d0*nu - 2.d0/9.d0 \\
ke(13,9) &= 4.d0/15.d0*nu + 28.d0/45.d0 \\
ke(14,9) &= 0.d0 \\
ke(15,9) &= 0.d0 \\
ke(16,9) &= 2.d0/9.d0*nu + 2.d0/9.d0 \\
\\
ke(10,10) &= -8.d0/9.d0*nu+64.d0/45.d0
\end{aligned}$$

APPENDIX C. Q8 ELEMENTAL STIFFNESS

102

$$ke(11,10) = -2.d0/9.d0*nu - 2.d0/9.d0$$

$$ke(12,10) = 0.d0$$

$$ke(13,10) = 0.d0$$

$$ke(14,10) = -4.d0/9.d0*nu - 4.d0/45.d0$$

$$ke(15,10) = 2.d0/9.d0*nu + 2.d0/9.d0$$

$$ke(16,10) = 0.d0$$

$$ke(11,11) = -8.d0/9.d0*nu + 64.d0/45.d0$$

$$ke(12,11) = 0.d0$$

$$ke(13,11) = 0.d0$$

$$ke(14,11) = 2.d0/9.d0*nu + 2.d0/9.d0$$

$$ke(15,11) = -4.d0/9.d0*nu - 4.d0/45.d0$$

$$ke(16,11) = 0.d0$$

$$ke(12,12) = -4.d0/15.d0*nu + 92.d0/45.d0$$

$$ke(13,12) = 2.d0/9.d0*nu + 2.d0/9.d0$$

$$ke(14,12) = 0.d0$$

$$ke(15,12) = 0.d0$$

$$ke(16,12) = 4.d0/15.d0*nu + 28.d0/45.d0$$

$$ke(13,13) = -4.d0/15.d0*nu + 92.d0/45.d0$$

$$ke(14,13) = 0.d0$$

$$ke(15,13) = 0.d0$$

$$ke(16,13) = -2.d0/9.d0*nu - 2.d0/9.d0$$

$$ke(14,14) = -8.d0/9.d0*nu + 64.d0/45.d0$$

$$ke(15,14) = -2.d0/9.d0*nu - 2.d0/9.d0$$

$$ke(16,14) = 0.d0$$

$$ke(15,15) = -8.d0/9.d0*nu + 64.d0/45.d0$$

$$ke(16,15) = 0.d0$$

$$ke(16,16) = -4.d0/15.d0*nu + 92.d0/45.d0$$