Structural topology optimisation based on the Boundary Element and Level Set methods

ULLAH, BASEER

How to cite:

ULLAH, BASEER (2014) *Structural topology optimisation based on the Boundary Element and Level Set methods*, Durham theses, Durham University. Available at Durham E-Theses Online: http://etheses.dur.ac.uk/10659/

Structural topology optimisation based on the Boundary Element and Level Set methods

Baseer Ullah

Thesis submitted towards the
degree of Doctor of Philosophy

Computational Mechanics Group
School of Engineering and Computing Sciences
Durham University
United Kingdom

May 2014

Structural topology optimisation based on the Boundary Element and Level Set methods

Baseer Ullah

Abstract

The research work presented in this thesis is related to the development of structural optimisation algorithms based on the boundary element and level set methods for two and three-dimensional linear elastic problems. In the initial implementation, a stress based evolutionary structural optimisation (ESO) approach has been used to add and remove material simultaneously for the solution of two-dimensional optimisation problems. The level set method (LSM) is used to provide an implicit description of the structural geometry, which is also capable of automatically handling topological changes, i.e. holes merging with each other or with the boundary. The classical level set based optimisation methods are dependent on initial designs with pre-existing holes. However, the proposed method automatically introduces internal cavities utilising a stress based hole insertion criteria, and thereby eliminates the use of initial designs with pre-existing holes. A detailed study has also been carried out to investigate the relationship between a stress and topological derivative based hole insertion criteria within a boundary element method (BEM) and LSM framework. The evolving structural geometry (i.e. the zero level set contours) is represented by non-uniform rational b-splines (NURBS), providing a smooth geometry throughout the optimisation process and completely eliminating jagged edges.

The BEM and LSM are further combined with a shape sensitivity approach for the solution of minimum compliance problems in two-dimensions. The proposed sensitivity based method is capable of automatically inserting holes during the optimisation process using a topological derivative approach. In order to investigate the associated advantages and disadvantages of the evolutionary and sensitivity based optimisation methods a comparative study has also been carried out.

There are two advantages associated with the use of LSM in three-dimensional topology optimisation. Firstly, the LSM may readily be applied to three-dimensional space, and it is shown how this can be linked to a 3D BEM solver. Secondly, the holes appear automatically through the intersection of two surfaces moving towards each other. Therefore, the use of LSM eliminates the need for an additional hole insertion mechanism as both shape and topology optimisation can be performed at the same time. A complete algorithm is proposed and tested for BEM and LSM based topology optimisation in three-dimensions. Optimal geometries compare well against those in the literature for a range of benchmark examples.

Declaration

The work in this thesis is based on research carried out in the Computational Mechanics Group, School of Engineering and Computing Sciences, Durham University. No part of this report has been submitted elsewhere for any other degree or qualification and it is all my own work unless referenced to the contrary in the text.

Parts of this work have been published, submitted or presented in the following journal papers and conferences:

Journals

B. Ullah, J. Trevelyan and P.C. Matthews. Structural optimisation based on the boundary element and level set methods. *Computers and Structures*, 137 (2014), 14-30.

B. Ullah and J. Trevelyan. Correlation between hole insertion criteria in a boundary element and level set based topology optimisation method. *Engineering Analysis with Boundary Elements*, 37 (2013), 1457-1470.

Conferences

B. Ullah, J. Trevelyan and I. Ivrissimtzis. A three-dimensional implementation of the boundary element and level set based structural topology optimisation. In *Proceedings of the 22nd UK Conference of the Association for Computational Mechanics in Engineering*, University of Exeter, UK, April 2014.

B. Ullah and J. Trevelyan. A bi-directional evolutionary structural optimisation based on the level set and boundary element methods. In *Proceedings of the 9th UK Conference on Boundary Integral Methods*, University of Aberdeen, UK, July 2013.

B. Ullah, J. Trevelyan and P.C. Matthews. A comparison of stress criterion and topological derivative for hole insertion in level set based structural topology optimisation. *CM13: Proceedings of the International Conference on Computational Mechanics*, University of

Durham, UK, March 2013.

B. Ullah, J. Trevelyan and P.C. Matthews. Structural optimisation based on the boundary element and level set methods with NURBS geometry representation. In *Proceedings of the 10th World Congress on Computational Mechanics*, São Paulo, Brazil, July 2012.

B. Ullah, J. Trevelyan and P.C. Matthews. Structural optimisation using boundary element based level set methods. In *Proceedings of the 20th UK Conference of the Association for Computational Mechanics in Engineering*, The University of Manchester, UK, March 2012.

The following journal papers are also in preparation:

B. Ullah and J. Trevelyan. The use of sensitivities in a boundary element and level set based structural topology optimisation.

B. Ullah, J. Trevelyan and I. Ivrissimtzis. A three-dimensional implementation of the boundary element and level set based structural topology optimisation.

Acknowledgements

First of all, I am deeply indebted to my supervisor Professor Jon Trevelyan for his inspiring supervision, expertise, continual encouragements and sustained support. He provided not only the basic motivation for the thesis, but also lots of invaluable suggestions and insightful comments throughout the progress of the research, and it has been a privilege to work with him. This work would never have been completed without his contributions.

I am extremely grateful to Dr. Ioannis Ivrissimtzis (my current second supervisor) and Dr. Peter Matthews (my former second supervisor) for their precious help and advice.

I am also thankful to the many people for their help and many fruitful discussions. The individuals are too numerous to list. However, there are persons who have been especially helpful with their suggestions and guidance. Their contribution deserves a specific mention: Professor Charles Augarde, Professor Toshiro Matsumoto, Dr. H. Alicia Kim, Dr. Takayuki Yamada, Dr. M. Shadi Mohamed and Dr. Tim Foster.

I am grateful to the Durham University for the financial support given to me through the Durham Doctoral Studentship scheme.

Finally, I would like to express my sincere thanks and deep gratitude to my parents, brothers, sisters, wife and children (Umair and Uzair) for their patience, sacrifice, emotional support and continuous encouragement throughout the course of this long Journey.

Baseer Ullah
Durham, May 2014

Contents

List of Figures

List of Tables

Nomenclature & Abbreviations

Symbols

(ξ, η)	Parametric coordinates
(x, y, z)	Cartesian coordinates
$[A]$	System matrix
$[G]$	Displacement matrix
$[H]$	Traction matrix
$\delta_{i,j}$	Kronecker delta
ℓ	Lagrange multiplier
Γ	A boundary
Γ_0	Boundary with Dirichlet boundary conditions
Γ_1	Boundary with non-homogeneous Neumann boundary conditions
Γ_2	Boundary with homogeneous Neumann boundary conditions
μ	Shear modulus
ν	Poisson's ratio
Ω	A domain
Φ	Shape functions
$\phi(\vec{x})$	Level set function
ϕ_t	Temporary signed distance function
ρ	Density
$\sigma_1, \sigma_2, \sigma_3$	Principal stresses
σ_t	Stress range
σ_{Vmax}	Maximum von Mises stress
σ_{Vmin}	Minimum von Mises stress
σ_V	von Mises stress (MPa)
σ_Y	Yield stress
$\{b\}$	Right hand side vector
$\{x\}$	Solution vector
B	Control points
d	Level set grid size
D_{Tmin}	Minimum topological derivative
D_T	Topological derivative
F	Normal velocity
f_K	Specific stiffness
f_U	Specific strain energy (Nmm^4)
J	Bernstein basis function
$J(u)$	Compliance objective function (Nmm)
K	Stiffness
k	Order of the curve
k_T, f_T	Topological derivative threshold factor
k_V, f_V	von Mises stress threshold factor
M	Triangular surface mesh
N	B-spline basis function
n	Outward normal of a surface
P	position vector along the curve
p	A point
Q	Mesh quality
q	A point
Q_E	Element quality
Q_{min}	Minimum acceptable mesh quality
r	A radius
RR	Removal ratio
RR_i	Incremental removal ratio
t	A traction
T_{ij}	Traction kernel
$tr\sigma$	Trace of the stress tensor
$tr\varepsilon$	Trace of the strain tensor
U	Strain energy (Nmm)
u	A displacement

U_{ij}	Displacement kernel
V	Volume at the current iteration
V_f	Triangular mesh fixed vertex
V_f	Triangular mesh movable vertex
W	Strain energy density

Chapter 1

Introduction

1.1 Overview

The inherent aspiration of human nature for optimality (perfection) motivates engineers, scientists, and mathematicians to continue their search for extremes. Optimisation can be broadly defined as the act of obtaining the best outcome from a collection of available alternatives under given circumstances. In this modern era of global competition the tool of optimisation equipped the engineers to design and produce new, better and cost efficient products as well as to devise plans and procedures to optimise existing systems. The development of fast computers not only enhanced their utilisation in the field of optimisation research but also achieved speedup of the whole process.

1.2 Background and motivation

Structural optimisation is considered one of the most important and challenging fields in engineering optimisation. Structural optimisation arranges the assembly of structural elements for sustaining the applied load in the most efficient manner. The initial research work carried out in structural optimisation can be traced back to the seminal work of Michell [72] in 1904, who presented analytical methods for the optimisation of frame structures. Due to its simple nature and the availability of the analytical techniques research started earlier for frame or truss structures. Optimisation techniques for continuum structures emerged with the development of computational tools, which are capable of handling large-scale optimisation problems, and hence greatly accelerated the research work in this area. Numerous methods have been developed over the last decades describing various numerical techniques to generate structures that are optimal in terms of quantities such as weight, cost and stiffness.

Methods in the optimisation of continuum structure include shape and topology optimisation. In order to develop high performance structures, a topology optimisation technique can be used, which optimise a given structure by determining the best settings and geometries of cavities in the design domain. In an optimisation process, the structural geometry is continuously modified and as a result a new discretisation is always required at each optimisation step. In case of a standard finite element based discretisation, simple re-meshing usually leads to highly deformed finite elements, which can distort the accuracy of the finite element analysis.

Bendsøe and Kikuchi [14] first proposed a homogenisation method, which is based on a fixed discretisation of the design domain with finite elements. Each finite element is represented as a microstructure with microvoid, which are continuously modified for the optimisation of a given structure. However, this results optimal solutions having an infinitely fine porous structure with variable density. Therefore, the resulting optimal structures are very difficult to interpret from an engineering point of view. In order to avoid the occurrence of porous regions in an optimal design, instead of representing the finite elements as microstructure, densities are assigned to each finite element. This approach is commonly known as a solid isotropic material with penalization [101], which forces design to solid and void solutions. However, the optimal solutions obtained have still variable density elements around the structural boundary. Additionally, solutions have regions with alternate solid and void pattern (checkerboard pattern) and highly mesh dependent.

Xie and Steven [131] presented an evolutionary structural optimisation approach based on the progressive removal of inefficient elements of a fixed discretised finite element design domain. Although, this approach completely eliminates the occurrence of intermediate density regions within an optimal design. However, the removal of a complete element from the design resulted in the final structure with jagged edges along the structural boundary. Additionally, checkerboard patterns also exist within the optimal design obtained with the ESO approach and the optimal solutions are mesh dependent. Querin *et al.* [92, 93] presented a bi-directional evolutionary optimisation approach (BESO) by allowing the efficient material to be added while the inefficient material is removed. Huang and Xie [48] extended the initial BESO to a much improved algorithm for stiffness optimisation, which is also capable to eliminate the occurrence of the checkerboard patterns.

Instead of discretising a structural geometry using an element based discretisation approach, a boundary element and ESO based optimisation method has been proposed in [21]. The structural boundary is represented through splines, where the design variables are control points. In comparison with the domain based discretisation, a boundary based discretisation reduces the dimensionality of the problem and simplify the re-meshing task. Additionally, the structural response can be accurately predicated directly along the structural boundary. In the boundary based BESO approach [21] inefficient material can be removed through boundary movements and hole insertion within the design domain. However, due to an explicit boundary representation additional care is always required during hole merging with each other and with the boundary during the optimisation process.

However, there exists another class of topology optimisation, which are based on the level set method [78]. The level set based optimisation methods implicitly represents the structural geometry and provide a clear boundary description throughout the optimisation process. Additionally, the level set method is capable of handling topological changes automatically, i.e. hole merging with each other and with the boundary. A level set method is based on an Eulerian approach and works on an underlying Cartesian grid. An important part in the level set based approach is to link an implicit and a structural model of a continuously evolving geometry or to map the geometry and mechanical model. The mapping techniques can be broadly classified into a density and boundary based mapping techniques.

The density based mapping is a simple technique, which is based on the fixed FE discretisation and a density distribution. Finite element inside the geometry is represented as solid while those outside or within a hole as empty. However, the density varies within elements crossed by the structural boundary. This results in optimal design with ambiguous material along the structural boundary [136]. However, a boundary based mapping provides an accurate prediction of the local structural response along the boundary which is essential for e.g. accurate stress calculation [29]. In addition, the boundary based discretisation within a level set framework provides a clear description of the structural geometry without ambiguous material along the structural boundary.

The use of the boundary element method and LSM is in the very early stages. The methods presented to date (i.e. [3, 136]) are dependent on an initial gussed design with pre-existing holes, and have slow convergence rates. Further, these methods are limited to two-dimensional problems. Therefore, a detailed study is carried out in this thesis to propose efficient optimisation techniques within the BEM and LSM framework for both two and three-dimensional linear elastic problems.

1.3 Scope and outline

The aim of the this research work is to develop efficient and robust structural optimisation algorithms, starting from two-dimensions and further extending it to three-dimensions. In the proposed algorithms, the BEM is used as a structural analysis tool, which analyses the modified geometry at each optimisation iteration. The BEM is a well-established alternative to the finite element method (FEM) in structural analysis, and is attractive because it requires discretisation only at the structural boundary. This reduction of problem dimensionality considerably simplifies the re-meshing task, which can be performed efficiently and robustly. This is further combined with the LSM to implicitly represent the evolving structural geometry. In the proposed two-dimensional approach, ESO and sensitivity based optimisation techniques are used for the solution of minimum compliance problems. Further, during the optimisation iterations, the structural geometry is represented with a standard CAD format, i.e. NURBS, providing a smooth geometry throughout the optimisation process and completely eliminating jagged edges. In addition, the optimal geometry represented with NURBS can be used directly in other design processes. One of the most interesting features of the LSM is its natural extension to three-dimensional space. This feature has been effectively utilised during the extension of the two-dimensional approach to three-dimensions. Further, the use of LSM in three-dimensional topology optimisation allows automatic hole nucleation through the intersection of two surfaces moving towards each other.

Further, holes appear automatically through the intersection of two surfaces moving towards each other.

This thesis is structured into a total of 11 chapters as outlined below. Chapters 2 to 4 contain reviews and descriptions of background material. Chapter 5 to 10 contain novel ideas developed by the author.

Chapter 1: Introduction presents a brief overview of the structural optimisation and aim of this thesis.

Chapter 2: Structural Optimization Review demonstrates detailed literature review of the most commonly used structural optimisation methods.

Chapter 3: Boundary element method gives a comprehensive derivation of the boundary element method.

Chapter 4: Non-uniform rational b-splines presents the relevant theory and implementation details of the NURBS based geometry representation.

Chapter 5: LSM and BEM based structural optimisation discusses the implementation details of the LSM, BEM and ESO based optimisation technique for two-dimensional problems. An initial guessed design with pre-existing holes is used for the solution of optimisation problem.

Chapter 6: LSM and BEM based structural optimisation with a hole insertion mechanism extends the initial approach presented in Chapter 5 with the introduction of a hole insertion mechanism. During the numerical implementation, holes are automatically inserted within the structure using a von Mises stress based criteria.

Chapter 7: Correlation between hole insertion criteria investigates the relationship between stress and topological derivative based hole insertion criteria within BEM and LSM based topology optimisation framework.

Chapter 8: The use of sensitivities in a BEM and LSM based topology optimisation describes the implementation of shape sensitivities in an LSM and BEM based topology optimisation method. The proposed method is also equipped with a hole insertion mechanism based on the topological derivative approach.

Chapter 9: A comparison of ESO and sensitivity based optimisation method presents a comparative study of the use of ESO and sensitivity based optimisation techniques within an LSM and BEM framework. This study also discusses in details the associated advantages and disadvantages with each of the optimisation approaches.

Chapter 10: A 3D implementation of BEM and LSM based structural optimisation describes the extension of the two-dimensional approach presented in Chapter 5 to three-dimensions.

Chapter 11: Conclusions summarises the work carried out for this thesis and provides recommendations for future work.

Chapter 2

Structural Optimisation Review

2.1 Overview

This chapter presents an overview of structural optimisation methods. The chapter starts with a general definition and a mathematical description of the structural optimisation problem. The different types of structural optimisation problems are discussed afterwards. This is followed by a detailed discussion of the commonly used topology optimisation methods. Finally, the chapter closes with some concluding remarks on the presented literature.

2.2 Structural optimisation

Structural optimisation arranges the assembly of structural elements for sustaining the applied load in the most efficient manner. Such an arrangement of structural elements within a mechanical system largely depends on its application requirements. Structural engineers worldwide are driven by the search for a design that is in some sense optimal, making the most efficient use of materials. In order to support this search, an extensive body of literature has appeared over the last decades describing various numerical techniques to generate structures that are optimal in terms of quantities such as weight, cost and stiffness.

A general mathematical formulation of the optimisation problem is first presented in the following subsection. This is then followed by a discussion of the three different types of structural optimisation problems.

2.2.1 Mathematical form of structural optimisation problem

In mathematical terms an optimisation problem can be expressed as

$$\begin{aligned} \text{Min/Max:} \quad & f(x) \\ \text{Subject to:} \quad & h_i(x) \leq 0 \quad i = 1...n \\ & k_j(x) = 0 \quad j = 1...m \\ & x \in X \end{aligned} \tag{2.1}$$

Various terms used in Equation 2.1 are defined as below.

- **Objective function ($f(x)$):** The objective function identifies the best outcome of an optimisation process. The optimum can be achieved by minimising or maximising the objective function $f(x)$. In structural optimisation, weight, stress, stiffness or vibration frequencies are used as the objective functions. Optimisation of more than one objective function is known as multi-objective optimisation.

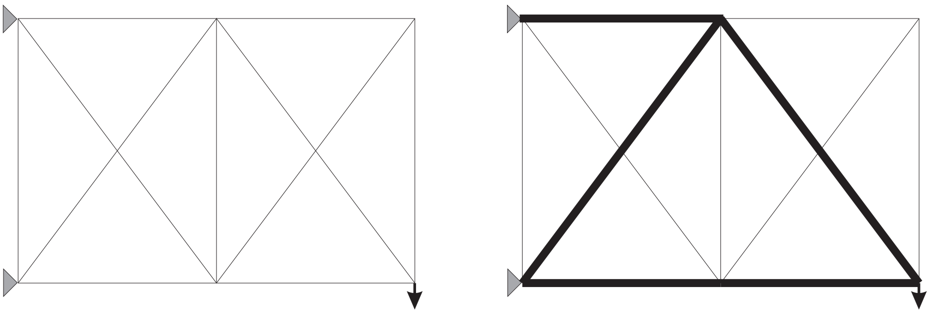
- **Design variable (x):** The function or vector which acts as an input to change the output of the design problem is known as the design variable. In structural optimisation, x is related to the parametrization of the geometry, and can be varied within the design space X. The design variable for the optimisation of a circular bar, for example, may be the diameter of the bar.

- **Constraints:** Constraints impose upper and/or lower limits on quantities (e.g. stress, displacement, volume etc) to be fulfilled in order to make the design feasible. Both equality $k(x) = 0$ and inequality $h(x) \leq 0$ constraints may be used in an optimisation process.

2.2.2 Types of structural optimisation problems

Structural optimisation can be broadly divided into three main categories, i.e.

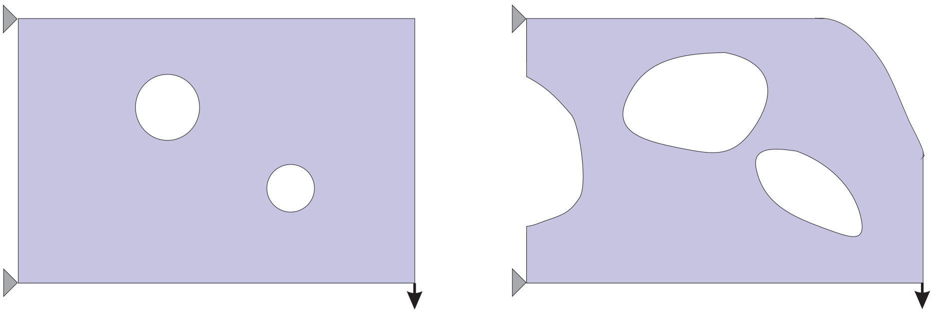
- **Size optimisation** A simple way of structural optimisation is the sizing optimisation. In this type of optimisation, the cross-sectional dimension (thickness, area) of each member of the structure is used as the design variable. Figure 2.1 shows the example of a truss structure optimised for maximum stiffness by varying the areas of the individual truss members. In size optimisation, the layout and the shape of the design remains fixed.



(a) Initial design (b) Size optimisation

Figure 2.1: Size optimisation example

- **Shape optimisation** In shape optimisation only the existing structural boundaries can be modified. Therefore, during the optimisation process, no holes can appear and no existing voids can disappear. The design variables in shape optimisation are those parameters which can vary the boundary of the structure. Figure 2.2 illustrates a shape optimisation example.



(a) Initial design

(b) Shape optimisation

Figure 2.2: Shape optimisation example

- **Topology optimisation** Topology optimisation is considered as the most powerful and versatile in structural optimisation. This type of optimisation allows to change the shape as well as the connectivity of material within the design domain. In the optimisation of truss structures, topology optimisation removes inefficient members thereby altering the connectivity of the structure and changes the topology of the structure. Similarly, topology optimisation can remove inefficient material and generate cavities in a design domain completely filled with material. The number, shape and locations of these cavities are all chosen as the design variables in topology Optimisation. Figure 2.3 illustrates a topology optimisation example.

Due to its simple nature and the availability of the analytical techniques research started earlier in size optimisation. This was followed by research studies carried out for the development of shape optimisation techniques. The development of numerical methods considerably accelerated research work related to shape optimisation. In comparison with size and shape optimisations, the nature of the topology optimisation problems is very complex because changes in size, shape and topology take place at the same

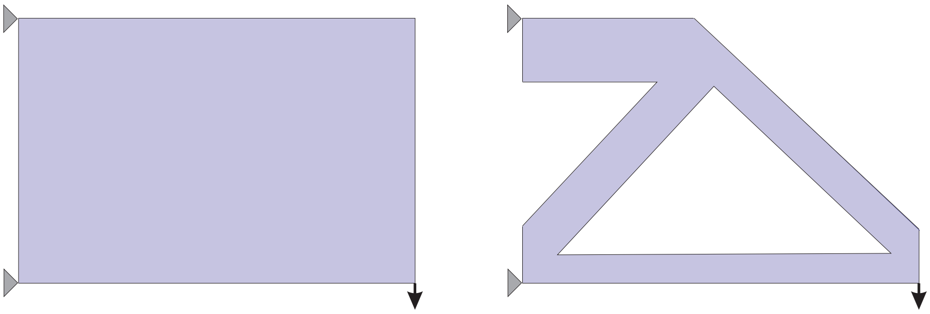

(a) Initial design (b) Topology optimisation

Figure 2.3: Topology optimisation example

time. This makes topology optimisation the most useful type when searching for high-performance structural configurations. The work presented in this thesis is also based on the development of topology optimisation techniques. In order to show how this work is embedded within the field of topology optimisation, the following sections present a brief overview of the most commonly used methods.

2.3 Topology Optimisation techniques

Topology optimisation techniques can be broadly classified into two groups based on the structural configuration, i.e low and high volume fractions structures. This classification is based on the ratio of the material volume to the volume of the design domain [100].

Trusses and grillages (beam systems) are analysed with the topology optimisation methods developed for low volume fraction structures. These methods have their origin in the seminal work of Michell [72], who presented the principles of optimum layouts for trusses with infinite bars of infinitesimal size with minimum weight. Later, Prager [89], Rozvany [99], and Prager and Rozvany [90] considerably improved and generalised the Michell's theory. Optimisation techniques for continuum (i.e. high volume fractions) structures emerged with the development of computational tools.

There are many general techniques available for optimisation problems, e.g. linear programming [138], nonlinear programming [113], hill climbing [50] random walks [88], response surfaces [88], simulated annealing [54], genetic algorithm [40] and particle swarm [106]. However, the literature review presented in this chapter is confined to some of the most frequently used and relevant techniques [28] in structural topology optimisation.

2.3.1 Homogenisation

Homogenisation method has been introduced by Bendsøe and Kikuchi [14], which describes the amount of material (i.e. density ρ) at each point of the design domain. Typically, this problem is represented by an initial fixed design domain that is discretised with a finite element mesh. The homogenisation method allows the possibility of a porous composite model ($0 < \rho < 1$) within an element, consisting of infinitesimally small but infinitely many unit cells. The design variables for each element are the length and height (e.g. a rectangular void) and their orientation in each element. For a given microstructure, the macroscopic properties such as the elastic modulus are computed from the microscopic properties of density and cell orientation by using the theory of homogenisation [80, 102]. However, the optimal microstructure produces topologies with checkerboard pattern, i.e. with large grey or porous areas with intermediate densities. The use of porous material in the optimal design makes this method impractical in most engineering applications [13, 108]. Moreover, the use of multiple design variables for each element increases the computational cost of this method.

2.3.2 Solid isotropic material with penalisation

In order to overcome the difficulties associated with the homogenisation method, Bendsøe proposed a density based approach [13], also called as the SIMP (solid isotropic material with penalisation) method [101]. SIMP method is based on the idea of using an isotropic material within each element of the FE model and is assumed to be a function of the penalized material density, described by an exponent power. This results in a solution where most of the elements are represented with only black and white designs. Elements with $\rho = 1$ are commonly represented as black and those with $\rho = 0$ as white regions in the structure. Intermediate densities are penalised in order to encourage 1-0 topologies. However, intermediate densities with ($0 < \rho < 1$) represented by different grey shades can still exist in the optimal design along the structural boundaries. The SIMP method is very popular and has been extensively used for the solution of a wide range of structural optimisation problems. Sigmund and Petersson [108] reported two main issues with the SIMP approach. Firstly, SIMP can generate results with checkerboard patterns. Secondly, it results in mesh dependent optimal solutions. Additional filtering techniques can be used to avoid the occurrence of checkerboard formations and the mesh dependency issues.

2.3.3 Evolutionary structural optimisation

The inspiration from nature, i.e. how structures such as bones, trees and shells achieve their optimum over a period of time in a specific environmental conditions, lead to the development of the ESO method presented by Xie and Steven [131]. Based on this approach, the design domain is first discretised into a uniform rectangular mesh. This is followed by an FE analysis to determine the stress distribution in the structure. Material is progressively removed from the low stressed regions of the structure based on some rejection criterion and this evolves the structure towards an optimal design. Later, Chu *et al.* [24, 25] presented an evolutionary approach for problems with stiffness constraint. The results obtained with the ESO approach provide a clear definition of the topology without any grey area [49].

Since its first appearance in 1993, ESO has been extend to topology optimisation of structures with buckling load [69], frequency [132] and temperature [65] constraints. The initial version of the ESO method is based on the gradual removal of inefficient material. However, in some cases it is possible that the material removed at early stages may be required in the later stages of the optimisation process. However, there is no such mechanism available to recover the material which has been removed from the structure and the result may not necessarily be the absolute optimum [49].

Querin *et al.* [92, 93] presented a bi-directional evolutionary optimisation approach (BESO) to overcome the deficiencies of the ESO approach. The BESO method extends the concept of ESO by allowing the efficient material to be added while the inefficient material is removed. Huang and Xie [48] extended the initial BESO to a much improved algorithm for stiffness optimisation.

However, like SIMP, ESO also suffers from checkerboard, mesh dependency and jagged edges issues. Kim *et al.* [57] proposed an efficient procedure to control the number of holes in the optimal design. Moreover, the proposed method is also capable of eliminating the formation of checkerboard patterns. In most cases the optimal designs obtained with the ESO approach have jagged boundaries, which result from the removal of an entire element during the optimisation process. To overcome these issues Kim *et al.* [56, 58] combined the ESO approach with the fixed grid finite element method. The proposed method effectively improved the computational efficiency of the ESO method, and optimal designs have been obtained with smoother boundaries.

While the FEM has been a popular method, it has some shortcomings when used as the analysis engine for optimisation methods. Haftka and Grandhi [44] highlighted the principal issue in shape optimisation, that it is difficult to ensure the accuracy of the analysis for a continuously changing finite element model; the change in the shape of a structure distorts the shape of the finite elements, with consequent deterioration in the accuracy of the stress solution. Moreover, the standard FEM requires re-meshing, usually of the complete design domain, incurring a high computational cost. The requirement of a smooth optimal geometry further increases the computational cost due to mesh refinement at the boundaries.

For these reasons it has been popular to use fixed grid FE approaches [39] to reduce the computational cost. This is attractive from the point of view of efficiency, but the accuracy of stresses in elements intersecting the problem boundaries may become compromised. Another possible alternative is the BEM [11], which requires discretisation only at the structural boundary and hence reduces the problem dimensionality. While the BEM has been exploited for shape optimisation with the sensitivity analysis in earlier works [23, 109, 110, 137], however, recently it has also been used within the ESO framework for topology optimisation problems.

Cervera and Trevelyan [21, 22] used BEM for topology optimisation of two and three-dimensional problems. In their ESO approach the moving boundary of the structure was represented by NURBS [97] explicitly, the spline control points being moved in response to local stress values. The developed algorithm creates internal cavities during the optimisation process based on the von Mises stress. Additional care was taken to handle hole merging during the optimisation process.

The boundary element based topological derivatives concept was used by Marczak [70] and Cisilino [26] for the topology optimisation of potential problems. Topological derivative correlates the change in the cost function with the insertion of a small hole in the design domain. This allows the nucleation of new holes anywhere in the design domain during the optimisation process. The derivation of the topological derivative formulation in these methods was based on the work of Novotny *et al.* [76], who presented a new computational approach based on the topological shape sensitivity analysis.

Carretero and Cisilino [19] presented a topology optimisation method for 2D elastic structures using the BEM and topological derivative approach. The proposed method evolves the initial design domain into an optimal design by progressively removing material from regions having the lowest values of topological derivative within and along the structural boundary. The optimal designs obtained are highly dependent on the number of points used during the calculation of topological derivatives within the design domain. A similar scheme has also been proposed by Marczak [71] for the optimisation of 2D elastic structures within a BEM framework.

In the above discussed BEM based topology optimisation methods an additional mechanism is always required to handle topological changes, i.e. holes merging with each other or with the boundary during the optimisation iterations. In addition, due to the use of a coarse discretisation the evolving geometry and the final optimal presented in [19] and [71] also exhibit jagged edges.

Bertsch *et al.* [15] proposed a BEM and ESO based optimisation method for the solution of 3D linear elastic problems. The optimisation problem was solved by incrementally removing material with the lowest values of the topological derivative [76]. The optimisation algorithm was applied to some basic problems in the literature of structural optimisation and the optimal designs obtained highly suffer from the jagged edges problem. The absence of a material addition mechanism further restricts the application of this method for a general class of optimisation problems.

2.3.4 Level set based structural topology optimisation

The level set (LS) method is an efficient numerical technique developed by Osher and Sethian [78] for the tracking of propagating interfaces. There is a wide variety of applications, including structural optimisation, in which LSM has been successfully implemented. The LS method uses the Eulerian approach to represent an evolving geometry implicitly. In an LS based structural optimisation, the structural geometry is first embedded as the zero level set of a higher dimensional function ϕ. In 2D this method works on an underlying fixed Cartesian grid. In most cases, the initial function ϕ is defined as the distance of a particular grid point from the boundary with a sign to indicate points either inside or outside of the boundary. Mathematically, it can be written as

$$\phi(\vec{x}) \begin{cases} < 0 & \vec{x} \in \text{(within the boundary)} \\ = 0 & \vec{x} \in \text{(interface)} \\ > 0 & \vec{x} \in \text{(outside the boundary)} \end{cases} \tag{2.2}$$

where $\vec{x}$ is a point in the level set domain. In the implicit representation, the connectivity of the discretisation does not need to be determined explicitly. This is one of the most interesting features of the implicit geometric representation, in that merging and breaking of curves in 2D and surfaces in 3D can be handled automatically.

The propagation of the structural boundary during the optimisation iterations can be linked with the evolution of ϕ as an initial value problem. This means that the position of the structural boundary at any time t is given by the zero level set of the function ϕ. A change in ϕ will modify the structural geometry accordingly. The level set function ϕ can be evolved through the solution of a Hamilton-Jacobi (HJ) equation [78]

$$\frac{\partial \phi}{\partial t} + F|\nabla \phi| = 0 \tag{2.3}$$

where F is the velocity in the normal direction and t is the virtual time. The normal velocity along the boundary can be computed from the structural response, e.g shape sensitivity analysis ([8, 122]). A positive velocity moves the boundary outward, whereas the negative velocity moves it in the inward direction.

Sethian and Wiegmann [104] first presented an LS based structural optimisation method. In their implementation, the boundary velocities are calculated through a von Mises stress based criterion. During the optimisation process holes were also inserted at the low stressed regions of the structure.

In most of the LS based optimisation methods shape sensitivity analysis is used for the normal velocity calculations, e.g. [8, 79, 122]. The normal velocity calculations in these methods are based on the value of the Lagrange multiplier. A fixed value of the Lagrange multiplier is used in [8], which may not guarantee constraint satisfaction. However, in the work of Osher and Santosa [79] and Wang *et al.* [122] computation of the Lagrange multiplier is carried out during the optimisation process.

Allaire *et al.* [8] independently proposed an LS based optimisation method for the solution of 2D and 3D optimisation problems with both linear and non-linear structural material. However, their approach is restrictive in that no new holes can be nucleated in 2D structural optimisation; moreover, the optimum solutions were highly dependent on the initially guessed topology. Further, Allaire and Jou [6] presented a shape sensitivity approach with LS method for the solutions of maximising the first eigenfrequency and compliance minimisation for multiple load problems.

The most challenging structural optimisation problems are those of topology optimisation, which remains an active research area. Eschenauer *et al.* [35] introduced the bubble method, which is based on the insertion of new holes in the structure and the subsequent use of a shape optimisation method to determine their optimal size and shape. This approach leads to the formulation of topological derivative [20, 111], which has been successfully implemented as a criterion for hole insertion [29] within the LS and shape sensitivity based optimisation approaches.

Amstuatz and Andreä [9] proposed a new algorithm for the level set based structural optimisation based on the topological derivative concept, which also allow holes nucleation. Instead of using the Hamilton-Jacobi equation to update the structural boundary, the proposed approach allows the update of the entire design in each optimisation iteration. However, the proposed method suffers from local minima and can only be used with a volume constraint.

A radial basis function (RBF) approach was introduced by Wang and Wang [125] to construct the implicit LS function and to discretise the initial value problem into an interpolation problem. Due to the use of multiquadric RBF, a relatively smooth level set evolution can be maintained without re-initialisation. In addition, nucleation of new holes has been allowed during the optimisation process. However, the proposed method is dependent on a fixed parameter to satisfy the volume constraint. Wang *et al.* [126] combined the RBF based level set optimisation method with a bisectioning algorithm, which exactly satisfies the volume constraint during the optimisation process.

Allaire and Jouve [7] combined the shape derivatives with topological derivatives approach to present a level set based optimisation method capable of automatic hole insertion. The proposed approach was shown to be independent of local minima but the implementation of topological derivatives is very difficult in numerical practice [68, 126] because the hole size is dependent on a single mesh cell which cannot be infinitesimally small as proposed in the method. In addition, the resulting optimal structures depend on the values of various parameters, which can affect the stability of the optimisation process [133].

In most of the LS based optimisation methods (e.g. [6, 122]) the structural boundary modifications take place through the solution of Equation (2.3). In which the time step size is limited by the Courant-Friedrichs-Lewy (CFL) condition. Luo *et al.* [68] proposed a semi-implicit additive operator splitting (AOS) scheme for the solution of Equation (2.3). This allows the selection of time step size independent of the CFL condition, thereby enhancing the stability and computational efficiency of the LS based optimisation method. However, the selection of a suitable time step size requires pre-knowledge from a few numerical tests because the use of a large time step size can result oscillations of the objective function.

Wei and Wang [127] presented the idea of topology optimisation using the piecewise constant LS (PCLS) method and is related to the phase field method. Piecewise smooth functions have been applied to generate a dynamic process for the geometric changes to form the exact constraints, eliminating the use of re-initialisation and the solution of Hamilton-Jacobi equation. In addition, the proposed method naturally nucleates holes during the optimisation process. However, the setting of certain parameters of the constraint functional for PCLS makes the optimal configurations dependent on the initial design.

Yamada *et al.* [133] combined the concept of fictitious interface energy with the LS based geometry representation for structural optimisation. The proposed method uses the fictitious interface energy as a regularisation factor during the numerical implementation. Instead of using the classical level set equation a reaction diffusion equation has been proposed for the update of LS function, which also eliminates re-initialisation. Holes are automatically nucleated during the optimisation process through the use of a topological derivative like function, and the update of the LS function through a reaction-diffusion equation.

Guo *et al.* [41] presented an LS based optimisation method with two different objective functions for the minimum stress design. The first objective function was based on the integral of von Mises stress over the whole structure whereas the second objective function uses the maximum von Mises stress in the design domain. An initial guess design with pre-existing holes was used for the solution of the optimisation problem.

Jia *et al.* [53] presented an evolutionary approach for the structural optimisation within an LSM framework. During the optimisation process, holes are automatically inserted around nodes with minimum strain energy values. However, the optimal solutions obtained are dependent on the initial guess designs.

Dunning and Kim [32] proposed a new optimisation method for two dimensional structures which automatically nucleates holes through the update of a secondary level set function. The proposed hole insertion mechanism is not dependent on the topological derivative approach. Further, the proposed method has been successfully applied to both single and multiple load cases.

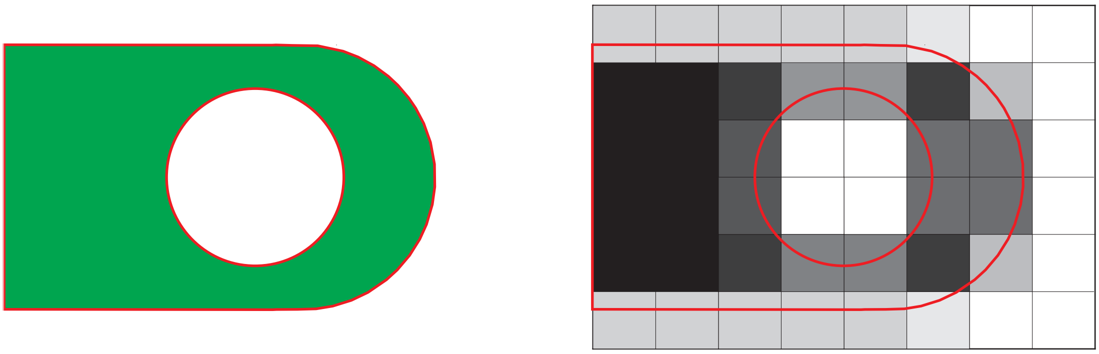
In addition to the most popular LS based optimisation approaches discussed above, there are numerous methods developed in the last decade for the solution of different types of problems both for two and three dimensions. The survey conducted by Van Dijk *et al.* [29] outlines some of these methods. LS based optimisation methods have been used for the solution of 3D problems by Allaire *et al.* [8]; Yamada *et al.* [133], problems with geometric nonlinearities by Allaire *et al.* [8]; Kwak and Cho [61], multiple materials by Wang and Wang [121]; Wang *et al.* [123], free vibration problems by Allaire and Jou [6]; Yamada *et al.* [133], loading uncertainty by Dunning *et al.* [33], stress minimisation by Allaire and Jouve [7]; Guo *et al.* [41]; Xia *et al.* [129], fluid problems by Amstuatz and Andreä [9], thermal problems by Ha and Cho [42]; Xia and Wang [130].

2.4 Overview of the LS based structural optimisation

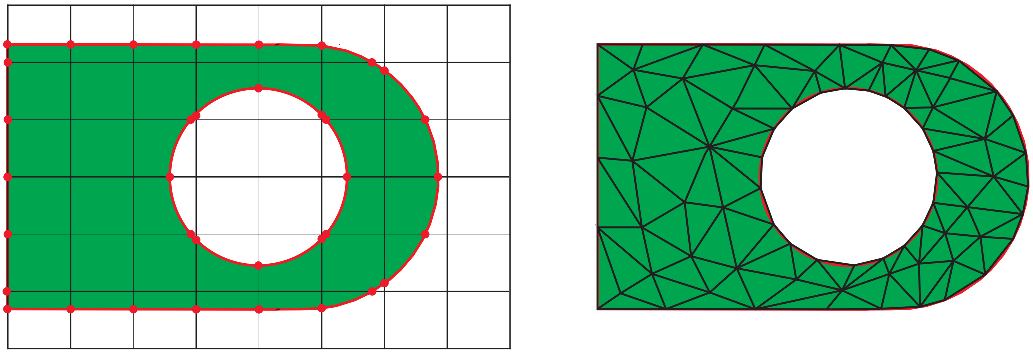
In the LS based optimisation approach, the selection of an effective structural performance measuring tool, and an efficient optimisation technique play an important role for the solution of the optimisation problems. The performance measuring tool predicts the structural response against the applied load and boundary conditions. These responses are then converted into a useful form by the optimisation technique. This is followed by an update of the LS function, which evolves the structural geometry. In the literature of LS based optimisation different approaches have been proposed for the structural performance measurement and optimisation techniques. The following sections discuss the associated advantages and disadvantages of both these techniques.

2.4.1 Structural performance measurement

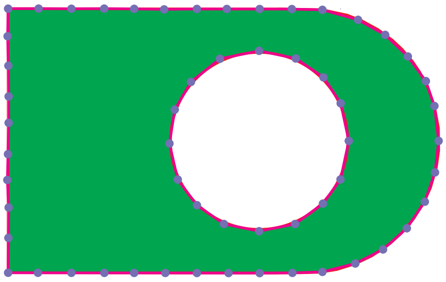
In an LS based optimisation approach, the performance of a candidate design can be measured through a geometry mapping technique [29], which projects the implicitly represented geometry onto the structural model. The most commonly used geometry mapping techniques are material distribution (density based), immersed boundary and conforming discretisation. Figure 2.4 displays a comparison of the different geometry mapping approaches; a detailed discussion is given below.



(a) Design domain (b) Density based mapping



(c) Immersed boundary mapping (d) Conforming mesh-1



(e) Conforming mesh-2

Figure 2.4: Geometry mapping approaches in LS based optimisation

Density based mapping

Most of the structural optimisation methods use the FEM as a structural analysis tool to predict the structural response. Each optimisation iteration modifies the structural geometry, and as a result the standard FEM cannot be used, which requires re-meshing, usually of the complete design domain, incurring a high computational cost. The requirement of a smooth optimal geometry further increases the computational cost due to mesh refinement at the boundaries.

To reduce the computational cost, most of the LS based optimisation methods utilise a fixed Eulerian type mesh with an "Ersatz material" approach [8] as an alternative FE analysis tool. During the optimisation process, the initial design domain (i.e. Figure 2.4a) is contained within a reference domain as shown in Figure 2.4b. The structural geometry is represented through a density distribution function, i.e. ($\eta < \rho < 1$) similar to the density based optimisation approach [108]. Solid material is represented by ($\rho = 1$) and holes in the structure are replaced by a specified minimum relative density ($\rho = \eta$). At each optimisation iteration FE analysis is carried out for the reference domain and the material densities in each element are adjusted accordingly, which exhibit changes in the structural geometry. Wang *et al.* [122] and Allaire *et al.* [8] initially implemented the density based approaches in their proposed LS based topology optimisation methods.

Although, the fixed grid is a simple approach, it is not effective to capture the exact geometry of the boundary [8] and in order to obtain an accurate solution near the boundary a highly dense grid distribution is required [51]. The density based mapping can also result into optimal designs with intermediate material densities regions along the structural boundary [136]. In addition, this can also result in non-smooth and indistinct boundary representation. A smoothed Heaviside function approach [68, 126] has been adopted to smooth the discontinuity at the boundary. However, the numerical integration of the stiffness matrix may be less accurate [124]. Dunning *et al.* [33] proposed a weighted least square approach to improve the accuracy of sensitivity calculations along the structural boundary within a fixed grid frame work.

Immersed boundary mapping

The immersed boundary approach uses a non body conforming fixed grid. Therefore, the structural geometry is not aligned with the grid and as depicted in Figure 2.4c can intersect some grid cells. This approach allows a clear boundary representation and avoids intermediate density material [29]. Sethian and Wiegmann [104] used the immersed interface method within a finite difference framework for the solution of LS based topology optimisation problems.

The extended FEM (X-FEM) can also be used to evaluate the required properties at the structural boundary through the local enrichment of elements intersected by the zero level set contour [38]. Different immersed boundary techniques can be used to impose boundary conditions along the evolving structural boundary [29]. Belytschko *et al.* [12] combined the implicit boundary representation with the X-FEM approach for the solution of topology optimisation problems. X-FEM has also been used by Van Miegroet and Duysix [119] for stress concentration minimisation using level set method. Recently, Wei *et al.* [128] used a fixed grid X-FEM with the level set method for the solution of minimum compliance problems.

Yamasaki *et al.* [136] developed a topology optimisation method for minimum compliance problems based on the immersed boundary mapping, boundary element and level set methods. In their work, the zero level set contours were mathematically represented by the FEM shape functions. Boundary elements were fitted for each finite element of the Eulerian mesh crossed by the zero level set contour.

The common problem reported in the implementation of the immersed boundary methods is the occurrence of small intersection of finite elements [119] or short boundary elements [136] while discretising the structural model. This can profoundly affect the accuracy of the structural response. The use of the immersed boundary techniques requires sophisticated codes and can make their implementation difficult and time consuming [29].

Conforming mesh

Some of the LS based optimisation methods use two types of meshes during the numerical implementation. The first one is a fixed Eulerian mesh which maintains the LS function throughout the optimisation process, whereas, the second mesh fits exactly the design domain. Two different approaches can be used to discretise the design domain, i.e. the domain discretisation (or FEM) and boundary only discretisation (BEM) as depicted in Figure 2.4d and 2.4e, respectively. This third type of mapping provides the most accurate analysis of the structural model and especially along the boundary.

The use of BEM with the level set method in structural optimisation was first used by Abe *et al.* [3]. In their implementation, at each optimisation iteration, the evolving structural boundary is re-constructed from the zero level set contours, which consists of line segments joining the zero level set intersection points. The resulting geometry is then meshed with linear boundary elements to perform the sensitivity analysis to calculate the boundary velocity for the next iteration. The boundary velocity is then extended to the underlying fixed regular grid. Structural geometry is evolved through the solution of LS equation. The proposed approach has also been extended for shape optimisation of sound scattering problems [2].

Ha and Cho [43] utilised an unstructured conforming discretisation approach for the optimisation of geometrically nonlinear structure within the the level set framework. In their work, the sensitivity information from the unstructured mesh are translated to the fixed Eulerian mesh using a distance weighting interpolation scheme.

Yamasaki *et al.* [135] presented a boundary tracking approach for level set based topology optimisation using the conforming discretisation approach. A mesh deformation mechanism has been used to move the nodes of the unstructured mesh towards the structural boundary using a geometry based re-initialisation scheme [134].

2.4.2 Optimisation techniques

In the LS based optimisation methods, an improvement in the design is mainly governed by changes in its shape. These changes can be carried out either with or without sensitivities based information, which can be used to compute the normal velocity along the boundary. Most of the LS based optimisation methods use shape sensitivity analysis to evolve an initial design towards an optimal solution. However, heuristic approach can also be used to carry out shape modifications.

Shape sensitivity analysis relates the response of a function towards a change in its shape. According to Allaire *et al.* [8], small variations of the boundary in the normal direction results into a change in the shape of the design domain. This suggests that shape sensitivities are only dependent on the normal boundary variations. Complete details of the shape sensitivity derivation and numerical implementation can be found in [8]. The methods presented by Wang *et.al* [122], Osher and Santosa [79], Allaire and Jou [6], Jia *et al.* [53], Dunning and Kim [32], etc. have used shape sensitivities information in their proposed methods. The sensitivity based techniques are popular because they are efficient although they require computation of suitably accurate gradients, which may not be available. Moreover, these methods can often have difficulties in dealing with local optima. They are complex algorithms that are difficult to implement efficiently.

The LS based optimisation method proposed by Sethian and Wiegmann [104] is based on the von Mises stress as a sensitivity for boundary modification. During the optimisation process, holes were inserted around the low stressed regions within the structure utilising the same criterion as used for boundary updates. Compared to their sensitivity-based counterparts, the non-sensitivity based method is simple to use, robust, and capable of dealing with almost any kind of structural optimisation problem, for example, ESO [49]. The ESO schemes have remained popular on account of their simplicity and extensive empirical evidence of the fact that their optimal solutions closely resemble those derived by more rigorous descent methods (e.g. Li *et al.* [64]).

2.5 Conclusions

This chapter presents a detailed discussion of the most popular methods used in the field of structural optimisation. Firstly, a Homogenisation method has been discussed, in which the optimal microstructure produces topologies with checkerboard pattern, i.e. large grey or porous areas with intermediate densities. The use of porous material in the optimal design makes this method impractical in most of the engineering applications. Moreover, the use of multiple design variables for each element increases the computational cost of this method. Similar to Homogenisation, SIMP can also generate results with checkerboard patterns. In addition, the optimal designs are highly mesh dependent and additional filtering techniques may be used to avoid the occurrence of checkerboard formations and the mesh dependency issues. However, like SIMP, ESO also suffers from checkerboard, mesh dependency and jagged edge issues, though, ESO eliminates the occurrence of grey regions in the structure. As a result of the various issues related with the Homogenisation, SIMP and ESO, it is very difficult to interpret the boundaries of the resulting optimal designs from engineering point of view.

The use of the LSM in structural optimisation has almost overcome the main issues related with the previously presented methods. Since its first use as a boundary tracking technique in structural optimisation, the LSM has been emerged as a powerful tool for the solution of numerous engineering problems. Though, the LS based optimisation methods are still in the development phase and there exist several issues, which need considerable attention. One of the most important issues is the use of an adequate mapping tool which can accurately measure the structural response. The density based mapping methods are efficient and easy to implement; however, the presence of intermediate density reduces the accuracy of the structural response and results in optimal designs with un-clear boundary. Hence, this limits the use of this approach for problems where higher accuracy is required at the structural boundary. These issues lead to the use of immersed boundary and body conforming mapping techniques in LS based optimisation methods.

In comparison with the immersed boundary mapping, the body conforming approach is attractive due to its simplicity and higher accuracy. However, the domain discretisation based body conforming mapping requires special care for a continuously changing structural geometry; that it is difficult to ensure the accuracy of the analysis for a continuously changing finite element model; the change in the shape of a structure distorts the shape of the finite elements, with consequent deterioration in the accuracy of the stress solution. Furthermore, the boundary based body mapping requires the boundary element on the level set boundary and avoids approximation at the boundary. The use of boundary based body conforming approach in the LS optimisation is in the very early stages, and relatively few methods are available. However, the convergence rates of these methods are very slow and the optimal designs are highly dependent on the initial guessed designs. The BEM is a well-established alternative to the FEM in structural analysis, and is attractive because it requires discretisation only at the design boundary. This reduction of problem dimensionality considerably simplifies the re-meshing task, which can be performed efficiently and robustly. Thus, its rapid and robust re-meshing and accurate boundary solutions make the boundary based body mapping method a natural choice for the solution of shape and topology optimisation problems. The combination of boundary based body mapping and level set method requires a comprehensive investigation to effectively utilise their attractive properties in the field of structural optimisation.

Chapter 3

Boundary Element Methods

3.1 Overview

In a level set based optimisation method, different approaches can be used to measure the structural response of a system. Most of these approaches have been discussed in detail in Section 2.4.1. Based on the conclusions in Section 2.5, i.e. the rapid and robust re-meshing, and accurate boundary solutions lead us to propose the BEM as a structural analysis tool in this work. This chapter presents the detailed mathematical derivation of BEM for linear elastic structural mechanics problems. Although, the BEM formulations are well-established and can be used for modelling of other physical phenomena, such as heat transfer, electrostatics, electromagnetics, acoustics, fracture mechanics, structural optimisation, fluid mechanics.

3.2 Evolution of boundary element method

The fundamentals of BEM can be traced bask to the 19th century with the classical formulations, for elasticity problems by Somigliana and Betti, and for potential problems by Fredholm. Later on, the classical formulations were followed by Mushkelishvili [74] and Mikhlin [73]. However, with the advent of computational resources BEM emerged and became a more generally applicable technique. Jaswon [52] and Symm [114] were among the first researchers who used BEM for the solution of potential problems. In 1967, Rizzo [95] used the BEM for the solution of elasticity problems in two-dimensions. This method has been further extended to the three-dimensions by Cruse [27]. In seventies, Lachat and Watson [62] introduced higher order elements, which further enhanced the BEM numerical capabilities. The development of efficient numerical techniques led towards the development of BEM based commercial softwares, e.g. CA [118] and BEASY [1]. The following sections present the mathematical derivation of BEM for linear elasticity. However, further details can be found in numerous books, e.g. Becker [11], Kane [55] and Trevelyan [117].

3.3 Boundary element formulation

3.3.1 Problem definition

Initially, a two-dimensional problem domain is considered for the derivation of a general form of elasticity formulation. This formulation is later extended into three-dimensions. Figure 3.1 displays the initial design domain Ω, with boundary Γ and outward normal n. The boundary Γ of the initial design domain Ω is decomposed into three parts as

$$\Gamma = \Gamma_0 \cup \Gamma_1 \cup \Gamma_2 \tag{3.1}$$

where Γ_0 corresponds to Dirichlet boundary conditions (where displacements are zeros), Γ_1 corresponds to non-homogeneous Neumann boundary conditions (where tractions are prescribed) and Γ_2 corresponds to homogeneous Neumann boundary conditions (traction

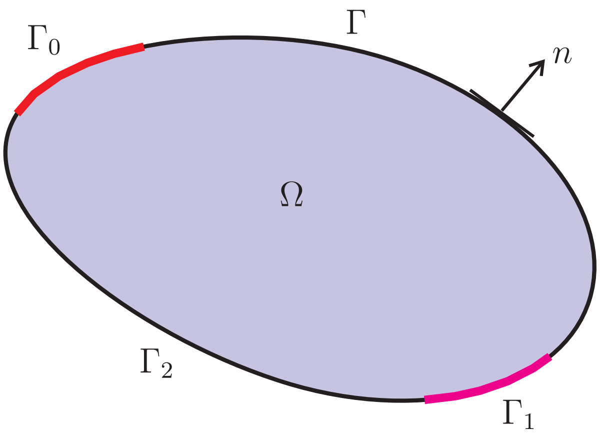


Figure 3.1: Definition of design domain

free and is allowed to vary during the optimisation process).

Let us assume the body is in equilibrium with the resultant of the applied loads being zero. The following equation holds for this state of equilibrium.

$$\int_\Gamma t_i d\Gamma + \int_\Omega b_i d\Omega = 0 \tag{3.2}$$

where t_i is the surface traction applied through surface Γ_1, and b_i is the body force applied through Ω. Equation (3.2) consists of both surface and volume integrals. For convenience, the divergence theorem, also known as Green's theorem, can be used to have both integrals in one form. This is a very useful identity, which relates the volume integral to a surface integral. According to this theorem

$$\int_\Omega \frac{\partial g_i}{\partial x_i} d\Omega = \int_\Omega g_{i,i} d\Omega = \int_\Gamma g_i n_i d\Gamma \tag{3.3}$$

where g is an arbitrary function with continuous first derivatives with respect to the Cartesian coordinate axes. Further, we may use the Cauchy stress transformation,

$$t_i = \sigma_{ij} n_j \tag{3.4}$$

where σ_{ij} is a stress component, and the Einstein summation conversion is assumed. The static equilibrium state can be written in the following form,

$$\int_{\Omega} \sigma_{ij,j} d\Omega + \int_{\Omega} b_i d\Omega = 0 \tag{3.5}$$

or

$$\int_{\Omega} (\sigma_{ij,j} + b_i) d\Omega = 0$$

3.3.2 The reciprocal or Betti's theorem

The reciprocal theorem is based on the principle of virtual work. Consider a body in a state of equilibrium with the following state of stresses and strains [11]:

- A set (1) of applied stresses $\sigma_{ij}^{(1)}$ that gives rise to a set of strains $\varepsilon_{ij}^{(1)}$.
- A different set (2) of applied stresses $\sigma_{ij}^{(2)}$ that gives rise to a set of strains $\varepsilon_{ij}^{(2)}$.

Betti's theorem states that the work done by the stresses of system (1) on the displacements of system (2) is equal to the work done by the stresses of system (2) on the displacements of system (1). Mathematically it can be written in the following form:

$$\int_{\Omega} \sigma_{ij}^{(1)} \varepsilon_{ij}^{(2)} d\Omega = \int_{\Omega} \sigma_{ij}^{(2)} \varepsilon_{ij}^{(1)} d\Omega \tag{3.6}$$

However, according to the strain-displacement relationship:

$$\varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \tag{3.7}$$

Therefore, it is more convenient to write the strains in (3.6) in the displacement form

$$\frac{1}{2} \int_{\Omega} \sigma_{ij}^{(1)} \left[\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right]^{(2)} d\Omega = \frac{1}{2} \int_{\Omega} \sigma_{ij}^{(2)} \left[\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right]^{(1)} d\Omega \tag{3.8}$$

Due to the repeated indices, Equation (3.8) can be re-written as

$$\int_{\Omega} \sigma_{ij}^{(1)} \frac{\partial u_i}{\partial x_j}^{(2)} d\Omega = \int_{\Omega} \sigma_{ij}^{(2)} \frac{\partial u_i}{\partial x_j}^{(1)} d\Omega \tag{3.9}$$

According to the product rule of differentiation

$$\frac{\partial(\sigma_{ij}u_i)}{\partial x_j} = \left(\frac{\partial \sigma_{ij}}{\partial x_j}\right)u_i + \sigma_{ij}\left(\frac{\partial u_i}{\partial x_j}\right) \tag{3.10}$$

$$\sigma_{ij}\left(\frac{\partial u_i}{\partial x_j}\right) = \frac{\partial(\sigma_{ij}u_i)}{\partial x_j} - \left(\frac{\partial \sigma_{ij}}{\partial x_j}\right)u_i \tag{3.11}$$

Consider the left hand side of (3.9) and substituting (3.11)

$$\int_\Omega \sigma_{ij}^{(1)}\frac{\partial u_i}{\partial x_j}^{(2)} d\Omega = \int_\Omega \left[\frac{\partial(\sigma_{ij}^{(1)}u_i^{(2)})}{\partial x_j} - \left(\frac{\partial \sigma_{ij}^{(1)}}{\partial x_j}\right)u_i^{(2)}\right] d\Omega \tag{3.12}$$

From Equation (3.5), the body force can be substituted for the second term on the right hand side of (3.12).

$$\int_\Omega \sigma_{ij}^{(1)}\frac{\partial u_i}{\partial x_j}^{(2)} d\Omega = \int_\Omega \left[\frac{\partial(\sigma_{ij}^{(1)}u_i^{(2)})}{\partial x_j}\right] d\Omega + \int_\Omega b_i^{(1)}u_i^{(2)} d\Omega \tag{3.13}$$

Further, the divergence theorem (3.3) can be used to transform the first volume integral on the right hand side of (3.13) into surface integral

$$\int_\Omega \sigma_{ij}^{(1)}\frac{\partial u_i}{\partial x_j}^{(2)} d\Omega = \int_\Gamma (\sigma_{ij}^{(1)}u_i^{(2)})n_j d\Gamma + \int_\Omega b_i^{(1)}u_i^{(2)} d\Omega \tag{3.14}$$

Re-writing the above equation for Cauchy stress transformation (3.4)

$$\int_\Omega \sigma_{ij}^{(1)}\frac{\partial u_i}{\partial x_j}^{(2)} d\Omega = \int_\Gamma (\sigma_{ij}^{(1)}n_j)u_i^{(2)} d\Gamma \ + \int_\Omega b_i^{(1)}u_i^{(2)} d\Omega \tag{3.15}$$

$$\int_\Omega \sigma_{ij}^{(1)}\frac{\partial u_i}{\partial x_j}^{(2)} d\Omega = \int_\Gamma t_i^{(1)}u_i^{(2)} d\Gamma \ + \int_\Omega b_i^{(1)}u_i^{(2)} d\Omega \tag{3.16}$$

Using the above procedure the final form of Betti's theorem results in the following expression given in [11]

$$\int_\Gamma t_i^{(1)}u_i^{(2)} d\Gamma \ + \int_\Omega b_i^{(1)}u_i^{(2)} d\Omega = \int_\Gamma t_i^{(2)}u_i^{(1)} d\Gamma \ + \int_\Omega b_i^{(2)}u_i^{(1)} d\Omega \tag{3.17}$$

3.3.3 Boundary integral equation

Equation(3.17) contains both surface and volume integrals. This equation can be transformed into a boundary integral equation (BIE) by some suitable assumptions. Stress system (1) is the problem to be solved and this will be facilitated through appropriate choice for the (arbitrary) stress system (2).

Point load or Kelvin solution

The Kelvin problem correlates the response of an infinite, homogenous, isotropic, elastic medium when a point (source point) load p is applied at an interior point and its effect is required at some other point(field point) Q anywhere in the domain. Two important conditions must be satisfied by this formulation.

1. When the distance between point p and Q tends to infinity the stresses must tend to zero.
2. When the distance between point p and Q tends to zero the stresses tend to infinity.

The point load can be represented by a *Dirac delta function* that is considered as a body load $b_i^{(2)}$. The advantage of using this function is that it is zero at all points x in the domain except at point x = X where it becomes infinity. Mathematically it can be written in the following form.

$$\Delta(\mathrm{X,x}) = \begin{cases} \infty & \mathrm{x} = \mathrm{X} \\ 0 & \text{otherwise} \end{cases} \tag{3.18}$$

The body force $b_i^{(2)}$ can now be expressed in the following form

$$b_i^{(2)} = \Delta(\mathrm{X,x}) e_i \tag{3.19}$$

where e_i is the unit vector in the coordinate direction i and gives the direction of the unit load at point X in the domain. The Dirac delta function has the following properties

$$\int_{-\infty}^{\infty} \Delta(\mathrm{X,x}) dx = 1 \tag{3.20}$$

$$\int_a^b \Delta(\mathrm{X},\mathrm{x}) f(x) dx = f(\mathrm{X}) \ , \ \ a < \mathrm{X} < b \tag{3.21}$$

Using property (3.21) the last term in (3.17) can be written for a source point p as

$$\int_\Omega b_i^{(2)} u_i^{(1)} d\Omega = \int_\Omega \Delta(p, \mathrm{x}) e_i u_i^{(1)} = u_i^{(1)}(p) e_i \tag{3.22}$$

According to [11] $u_i^{(1)}$ and $t_i^{(1)}$ are the unknown displacement and traction vectors. Therefore, we may substitute the following in (3.17) and assume no body forces

$$u_i^{(1)} = u_i \ , \ \ t_i^{(1)} = t_i(Q) \ , \ b_i^{(1)} = 0 \tag{3.23}$$

Similarly the traction and displacement fields corresponding to the point force can be written in the following form

$$u_i^{(2)} = U_{ij}(p, Q) e_j \ , \ t_i^{(2)} = T_{ij}(p, Q) e_j \ , \tag{3.24}$$

where Q is a point on Γ, p is a point inside the solution domain Ω, U_{ij} and T_{ij} are the fundamental solutions which will be discussed in detail in the following section. After substitution the final form of Equation (3.17) is

$$u_i(p) + \int_\Gamma T_{ij}(p, Q) u_i(Q) d\Gamma = \int_\Gamma U_{ij}(p, Q) t_i(Q) d\Gamma \tag{3.25}$$

Equation(3.25) is known as *Somigliana's identity for displacement.* It can be seen that (3.25) relates the displacement at a source point p to boundary values of displacement and traction. Capital letters (e.g. Q) in the above equation represent points on the boundary while lower case letters (e.g. p) represent points inside the domain. In the above equation the term $u_i(p)$ is not on the boundary. In the next section mathematical techniques will be used to transform these terms such that they contain quantities which can be evaluated at the boundary.

3.3.4 Fundamental solution

The solution of a governing differential equation due to a point load is commonly known as the fundamental solution. The *Navier* equation of elasticity and *Galerkin* vector will be used to derive the required fundamental solution.

Navier's equation of elasticity

The derivation of Navier's equation for displacement is based on the following well-known elasticity relationship. The purpose of this derivation is to find the decoupled set of relationships for the elasticity equations.

Equilibrium equation

$$\sigma_{ij,j} + b_i = 0 \tag{3.26}$$

Strain displacement relationship

$$\varepsilon_{ij} = \frac{1}{2}\left(u_{i,j} + u_{j,i}\right) \tag{3.27}$$

Stress-strain relationship (Hooke's Law)

$$\sigma_{ij} = \frac{2\mu\nu}{1-2\nu}\delta_{ij}\varepsilon_{mm} + 2\mu\varepsilon_{ij} \tag{3.28}$$

where μ is the shear modulus and ν is Poisson's ratio. δ_{ij} is the Kronecker delta:

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \tag{3.29}$$

It should be noted that in Equation (3.27) we used the mathematical (or tensorial) definition of shear strain, and this should be assumed throughout. The above relationships can be used to obtain the equilibrium equation in terms of displacement with a two-step approach. The first step substitutes Equation (3.27) into (3.28),

$$\sigma_{ij} = \frac{2\mu\nu}{1-2\nu}\delta_{ij}\left(u_{m,n}\right) + \mu\left(u_{i,j} + u_{j,i}\right) \tag{3.30}$$

The second step substitutes Equation (3.28) into (3.26) and gives the following simplified form in terms of displacement.

$$u_{i,jj} + \left(\frac{1}{1-2\nu}\right)u_{j,ij} = \frac{-b_i}{\mu} \tag{3.31}$$

Equation (3.31) is known as Navier's equation of elasticity for displacement. This equation yields three second order partial differential equations with just three displacements u_i unknowns but they are coupled [55]. Initially the basic elasticity equations (i.e. 3.26 to 3.28) contains fifteen unknowns with fifteen coupled differential equations. However, the fundamental solution requires uncoupled partial differential equations. Navier's equation of elasticity for displacement can be written in the following vector form.

$$\nabla^2\mathbf{u} + \frac{1}{1-2\nu}\nabla(\nabla.\mathbf{u}) = \frac{\mathbf{b}}{\mu} \tag{3.32}$$

where $\mathbf{u}$ and $\mathbf{b}$ are the vectors containing the displacement and body force components. For a point load the Navier's equation can be written as

$$u_{i,jj} + \left(\frac{1}{1-2\nu}\right)u_{j,ij} + \Delta(p,Q)e_i = 0; \tag{3.33}$$

Galerkin vector

Due to the coupling between the different displacement components, the set of equations (3.31) is difficult to solve analytically. There are various approaches for decoupling Navier's equations. The Galerkin vector approach described here will be used, which expresses the displacement vector in terms of another vector G which satisfies the Navier's equation.

$$u_i = G_{i,jj} - \frac{1}{2(1-\nu)} G_{j,ij} \tag{3.34}$$

Equation(3.34) can be written in the following vector form

$$\mathbf{u} = \nabla^2 \mathbf{G} - \frac{1}{2(1-\nu)} \nabla(\nabla.\mathbf{G}) \tag{3.35}$$

Vector **G** is known as the *Galerkin vector.* Now substituting the above equation into (3.31)

$$\nabla^4 G_i = \nabla^2(\nabla^2 G_i) = \frac{-b_i}{\mu} \tag{3.36}$$

After the substitution of displacement components expressed in Galerkin vector form, then, a completely decoupled set of equations has emerged. This is evident from the free index i which is not repeated. The operator (the Laplacian of the Laplacian) is called the *biharmonic operator.* In the case of zero body force, Equation (3.36) is known as the *biharmonic equation.* Further, for a point load it can be expressed in the following form

$$\nabla^2(\nabla^2 G_i)\mu + \Delta(p,Q)e_i = 0 \tag{3.37}$$

Three-dimensional problem

According to the *Kelvin solution*, the following Galerkin vector is a solution of Equation (3.36) [55],

$$G_i = \frac{1}{8\pi(1-\nu)} r(p,Q) \tag{3.38}$$

where $r(p,Q)$ is the distance between the source point p and field point Q i.e. $r(p,Q) = |Q-p|$. Substituting equation (3.38) into (3.36) gives

$$u_i = \frac{1}{16\mu\pi(1-\nu)} \left(\frac{1}{r(p,Q)}\right) \left[(3-4\nu)\delta_{ij} + \frac{\partial r(p,Q)}{\partial x_i}\frac{\partial r(p,Q)}{\partial x_j}\right] e_j \tag{3.39}$$

Equation (3.39) gives the displacement fundamental solution

$$U_{ij}(p,Q) = \frac{1}{16\mu\pi(1-\nu)}\left(\frac{1}{r(p,Q)}\right)\left[(3-4\nu)\delta_{ij} + \frac{\partial r(p,Q)}{\partial x_i}\frac{\partial r(p,Q)}{\partial x_j}\right] \tag{3.40}$$

$U_{ij}(p,Q)$ represents the displacement in the jth direction at point Q due to a point load acting in the ith direction at point p.

With the help of constitutive equation (3.28) a similar expression for the *traction fundamental solution* can be obtained as follows

$$\begin{aligned} t_i &= \frac{-1}{8\pi(1-\nu)r^2(p,Q)}\left(\frac{\partial r(p,Q)}{\partial n}\right)\left[(1-2\nu)\delta_{ij} + 3\frac{\partial r(p,Q)}{\partial x_i}\frac{\partial r(p,Q)}{\partial x_j}\right] \\ & -(1-2\nu)\left[\frac{\partial r(p,Q)}{\partial x_j}n_i - \frac{\partial r(p,Q)}{\partial x_i}n_j\right] e_j \end{aligned} \tag{3.41}$$

The term n_j denotes the components of the outward normal at the field point Q. From Equation (3.24) the displacement fundamental solution is

$$\begin{aligned} T_{ij}(p,Q) &= \frac{-1}{8\pi(1-\nu)r^2(p,Q)}\left(\frac{\partial r(p,Q)}{\partial n}\right)\left[(1-2\nu)\delta_{ij} + 3\frac{\partial r(p,Q)}{\partial x_i}\frac{\partial r(p,Q)}{\partial x_j}\right] \\ & -(1-2\nu)\left[\frac{\partial r(p,Q)}{\partial x_j}n_i - \frac{\partial r(p,Q)}{\partial x_i}n_j\right] \end{aligned} \tag{3.42}$$

$T_{ij}(p,Q)$ represents the traction in the jth direction at point Q due to a point load acting in the ith direction at point p.

Two-dimensional problem

Similarly the *fundamental solution* can be derived for two dimensional problem using the above procedure for three dimensions. The *Galerkin vector* for two dimensions is

$$G_i = \frac{1}{8\pi\mu(1-\nu)}r^2 \ln\left(\frac{1}{r}\right) e_i \tag{3.43}$$

where r is $r(p,Q)$ is the distance between source point p and field point Q. The plane strain fundamental solution for displacements and traction is given by

$$U_{ij}(p,Q) = \frac{1}{8\pi\mu(1-\nu)}\left[(3-4\nu)\ln\left(\frac{1}{r}\right)\delta_{ij} + \frac{\partial r}{\partial x_i}\frac{\partial r}{\partial x_j}\right] \tag{3.44}$$

$$\begin{aligned} T_{ij}(p,Q) \quad = \quad & \frac{-1}{4\pi(1-\nu)r}\left(\frac{\partial r}{\partial n}\right)\left[(1-2\nu)\delta_{ij} + 2\frac{\partial r}{\partial x_i}\frac{\partial r}{\partial x_j}\right] + \\ & \frac{1-2\nu}{4\pi(1-\nu)r}\left[\frac{\partial r}{\partial x_j}n_i - \frac{\partial r}{\partial x_i}n_j\right] \end{aligned} \tag{3.45}$$

The fundamental solutions can be easily computed because these expressions contain the known material properties and the distance between the points. Re-writing Equation (3.25)

$$u_i(p) + \int_\Gamma T_{ij}(p,Q)u_i(Q)d\Gamma = \int_\Gamma U_{ij}(p,Q)t_i(Q)d\Gamma$$

In the above equation, except for the first term on the left hand side all other terms are on the boundary. Placement of the source point p on the boundary (so now we denote this point as P) gives an expression entirely in the boundary terms. But on the other hand it produces mathematical complications in evaluating these integrals which become singular, because these terms are function of $\ln\frac{1}{r}$ and $\frac{1}{r}$ respectively (in case of a two dimensional problem). Similar complications also exist in the case of a three-dimensional problem.

There are logarithmic Gauss-Legendre schemes which can be used to integrate the U_{ij} term in case of two dimensional problems [117]. For the strongly singular integral containing $T_{ij}(p,Q)$ it is convenient to divide the surface into two portions. The altered surface consists of a semicircular or hemispherical surface around point of singularity at point P. The radius ρ of the modified surface will be set equal to zero in the limit which will convert it back to the original surface. Therefore $\Gamma - \Gamma_\epsilon$ and Γ_ϵ are the two surfaces for which the integral will be evaluated. The integral will be split into the following three parts

$$\begin{aligned}\int_\Gamma T_{ij}(P,Q)u_i(P)d\Gamma &= \lim_{\rho\to 0}\left[\int_{\Gamma-\Gamma_\epsilon} T_{ij}(P,Q)u_i(P)d\Gamma\right] + \lim_{\rho\to 0}\left[\int_{\Gamma_\epsilon} T_{ij}(P,Q)(u_i(x)-u_i(P))d\Gamma(x)\right]\\ &+ u_i(P)\lim_{\rho\to 0}\left[\int_{\Gamma_\epsilon} T_{ij}(P,Q)d\Gamma\right] \qquad (3.46)\end{aligned}$$

The breaking of integrals and taking the limit as the surface around singularity shrinks to zero has been called interpreting the integral in the Cauchy principal value sense. The first term on the right hand side is the Cauchy principal value integral, and the other terms may be considered as jump terms that arise as a result of the singularity. The second term on the right hand side vanishes since it contains $(u_i(x) - u_i(P))$. For convenience the last term can be written as

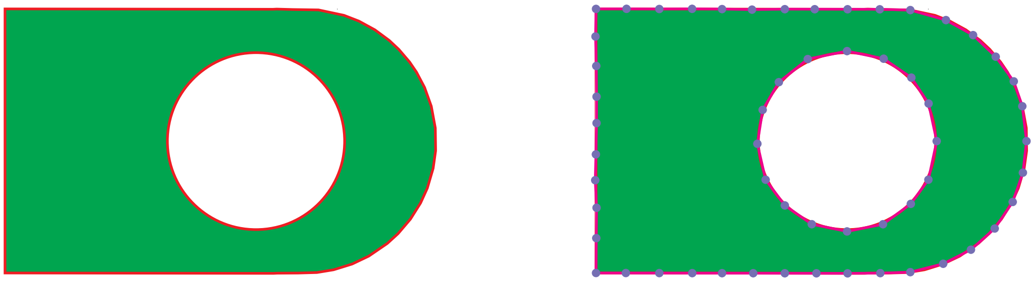
$$u_i(P)\lim_{r\to 0}\left[\int_{\Gamma_\epsilon} T_{ij}(P,Q)d\Gamma\right] = C_{ij}(P)u_j(P) \qquad (3.47)$$

Therefore the free term $C_{ij}(P)$ describes the local geometry around the point P. This term takes the value of 1 when p is completely inside the volume, 0 when when completely outside the volume , $\frac{1}{2}$ for a smooth boundary and for non-smooth boundary it may be found from the angle subtended by the domain at point P. In practice the explicit evaluation of both C_{ij} and the first term on the right hand side of Equation (3.46) can be avoided by simply considering rigid body motion. After shifting the source term to the boundary Equation (3.25) contains all the terms on the boundary, i.e.

$$C_{ij}(P)u_j(P) + \int_\Gamma T_{ij}(P,Q)u_j(P)d\Gamma = \int_\Gamma U_{ij}(P,Q)t_j(P)d\Gamma \qquad (3.48)$$

3.4 Numerical implementation

The solution of the BIE by using analytical methods is only possible for very simple problems. Complex problems need to be solved numerically. Numerical solution involves dividing the boundary into elements as shown in Figure 3.2, where dots represent nodes and line segments represent elements.



(a) Design domain (b) Boundary element mesh

Figure 3.2: Boundary element model

The element geometry and displacement may be characterized by values at the nodal points, and the number of nodal points increases the accuracy of the solution while making it more computationally expensive. There are N numbers of nodal points on the boundary; each node has four variables t_x,t_y, u_x and u_y, resulting in $4N$ variables for all the nodal points in case of a two dimensional problem. Additional components for three-dimensional problems increase the number of variables to six per node for traction and displacement. Each nodal point must have half of the variables prescribed as boundary conditions for the solution of any problem with a unique solution. Therefore, for a given nodal point either both displacements or both tractions or a traction and a displacement component should be prescribed. In a situation where there are no prescribed values of any kind it is often assumed that both tractions are equal to zero, i.e. the element lies on a free surface.

The solution of a particular problem in a two dimensional case requires $2N$ equations for $2N$ unknowns. The solution process for a particular problem is given below.

1. Divide the boundary into elements.

2. Using Equation (3.48) write an expression relating the traction and displacement components at every nodal point from node 1 to N by placing a force (traction) at node 1. This will result in two equations by considering one point force in both x and y directions.

3. Similarly, placing the load now on node 2 and repeating the above step results in another two equations in the same unknowns.

4. The above steps are repeated until the load is placed on the last node N.

5. Finally, the set of $2N$ linear equations (with $2N$ unknowns) from the above steps can be solved for displacement and traction at each node point.

6. Place the load at points internal to the material and solve Equation (3.25) to find the displacement at those points. Stress at a particular point can be obtained from the derivative form of Equation (3.25).

Further details of the above steps are completely described in the following steps.

3.4.1 Boundary discretisation

For numerical solution the boundary curve Γ must be divided into elements i.e $\Gamma_{eE}, E = 1, 2, 3, ..., N$.

$$\Gamma \; = \; \Gamma_{e1} \cup \Gamma_{e2} \cup \Gamma_{e3} \cup .... \cup \Gamma_{eN} \tag{3.49}$$

The mathematical form of this first level of discretisation can be written for Equation (3.48) as

$$C_{ij}(P)u_j(P) + \sum_{E=1}^{N} \int_{\Gamma_e E} T_{ij}(P,Q)u_j(P)d\Gamma = \sum_{E=1}^{N} \int_{\Gamma_e E} U_{ij}(P,Q)t_j(P)d\Gamma \tag{3.50}$$

3.4.2 Two-dimensional implementation

The geometry and variables (displacement and traction) value at any location between the nodes may be described using linear, quadratic, cubic or higher order polynomial interpolation functions. In the present case quadratic elements are used, which consist of three nodes in each element, one at each end and one at the mid point as shown in Figure 3.3. The local variable η used in this case has origin (i.e. zero value) at the mid node, and values -1 and $+1$ at the end nodes. This results into the following shape functions,

$$N_1(\eta) = \frac{\eta}{2}(\eta - 1) \tag{3.51}$$

$$N_2(\eta) = (1 - \eta)(1 + \eta) \tag{3.52}$$

$$N_3(\eta) = \frac{\eta}{2}(\eta + 1) \tag{3.53}$$

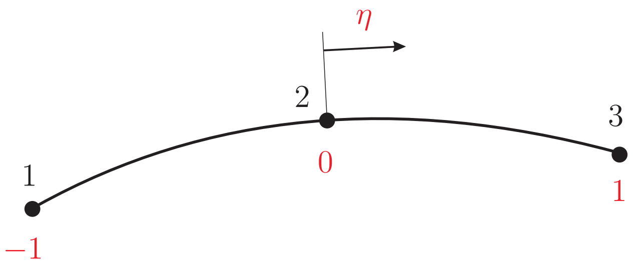


Figure 3.3: Quadratic element

The boundary geometry co-ordinates x and y , the unknown displacement field u_x and u_y and the traction field t_x and t_y are approximated for an element containing m nodes.

$$x(\eta) = \sum_{a}^{m} N_a(\eta) x_a = N_1(\eta)x_1 + N_2(\eta)x_2 + N_3(\eta)x_3 \tag{3.54}$$

$$y(\eta) = \sum_{a}^{m} N_a(\eta) y_a = N_1(\eta)y_1 + N_2(\eta)y_2 + N_3(\eta)y_3 \tag{3.55}$$

$$u_x(\eta) = \sum_{a}^{m} N_a(\eta) u_x^a = N_1(\eta)u_x^1 + N_2(\eta)u_x^2 + N_3(\eta)u_x^3 \tag{3.56}$$

$$u_y(\eta) = \sum_{a=1}^{m} N_a(\eta) u_y^a = N_1(\eta)u_y^1 + N_2(\eta)u_y^2 + N_3(\eta)u_y^3 \tag{3.57}$$

$$t_x(\eta) = \sum_{a=1}^{m} N_a(\eta) t_x^a = N_1(\eta)t_x^1 + N_2(\eta)t_x^2 + N_3(\eta)t_x^3 \tag{3.58}$$

$$t_y(\eta) = \sum_{a=1}^{m} N_a(\eta) t_y^a = N_1(\eta)t_y^1 + N_2(\eta)t_y^2 + N_3(\eta)t_y^3 \tag{3.59}$$

where $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ are the nodal coordinates, and the superscript on u and t

denotes the appropriate node number at which the displacement or traction is located.

The numerical integration performed on elements requires the transformation of variable from the boundary curve Γ to the local coordinate η. For example the variable x in the following one-dimensional integral can be transformed into another variable η as follows:

$$\int_{x_1}^{x_2} f(x)dx = \int_{\eta_1}^{\eta_2} f[x(\eta)]J(\eta)d\eta \tag{3.60}$$

The Jacobian $J(\eta)$ is defined as

$$J(\eta) = \frac{\partial x(\eta)}{\partial \eta} \tag{3.61}$$

For a two-dimensional case this transformation can be written in the following form

$$J(\eta) = \frac{d\Gamma}{d\eta} = \sqrt{\left(\frac{\partial x(\eta)}{\partial \eta}\right)^2 + \left(\frac{\partial y(\eta)}{\partial \eta}\right)^2} \tag{3.62}$$

After transformation the integrals in Equation (3.50) yield the following expressions

$$\int_{\Gamma_e E} T_{ij}(P,Q)u_j(P)d\Gamma = \left[\int_{-1}^{1} T_{ij}(P,Q)\left(\sum_{a=1}^{m} N_a(\eta)u_j^a\right)J(\eta)d\eta\right]^E \tag{3.63}$$

$$\int_{\Gamma_e E} U_{ij}(P,Q)t_j(P)d\Gamma = \left[\int_{-1}^{1} U_{ij}(P,Q)\left(\sum_{a=1}^{m} N_a(\eta)t_j^a\right)J(\eta)d\eta\right]^E \tag{3.64}$$

In these expressions it can be observed that the nodal point values of the jth component of traction and displacement are not a function of the variable of integration. Therefore these terms can be taken out from the integral signs. The expanded forms of the above equations are:

$$\begin{aligned}\int_{\Gamma_e E} T_{ij}(P,Q)u_j(P)d\Gamma \quad = \quad & \left[\int_{-1}^{1} T_{ij}(P,Q)N_1(\eta)J(\eta)d\eta\right]^E u_j^1 \\ & + \left[\int_{-1}^{1} T_{ij}(P,Q)N_2(\eta)J(\eta)d\eta\right]^E u_j^2 \\ & + \left[\int_{-1}^{1} T_{ij}(P,Q)N_3(\eta)J(\eta)d\eta\right]^E u_j^3\end{aligned} \tag{3.65}$$

$$\begin{aligned}\int_{\Gamma_e E} U_{ij}(P,Q)t_j(P)d\Gamma \;=\;& \left[\int_{-1}^{1} U_{ij}(P,Q)N_1(\eta)J(\eta)d\eta\right]^E t_j^1 \\ &+\left[\int_{-1}^{1} U_{ij}(P,Q)N_2(\eta)J(\eta)d\eta\right]^E t_j^2 \\ &+\left[\int_{-1}^{1} U_{ij}(P,Q)N_3(\eta)J(\eta)d\eta\right]^E t_j^3 \qquad (3.66)\end{aligned}$$

For simplicity we can write

$$\begin{aligned}T_{ij}^{Ea} &= \int_{-1}^{1} T_{ij}(P,Q)N_a(\eta)J(\eta)d\eta \\ U_{ij}^{Ea} &= \int_{-1}^{1} U_{ij}(P,Q)N_a(\eta)J(\eta)d\eta \qquad (3.67)\end{aligned}$$

Substituting the above expressions in Equation (3.65) and (3.66)

$$\int_{\Gamma_e E} U_{ij}(P,Q)t_j(P) = \sum_{a=1}^{m} T_{ij}^{Ea}u_j^{Ea} \qquad (3.68)$$

$$\int_{\Gamma_e E} U_{ij}(P,Q)t_j(P) = \sum_{a=1}^{m} U_{ij}^{Ea}t_j^{Ea} \qquad (3.69)$$

Finally substituting the above expressions into Equation (3.50) results in a discretised boundary element formulation

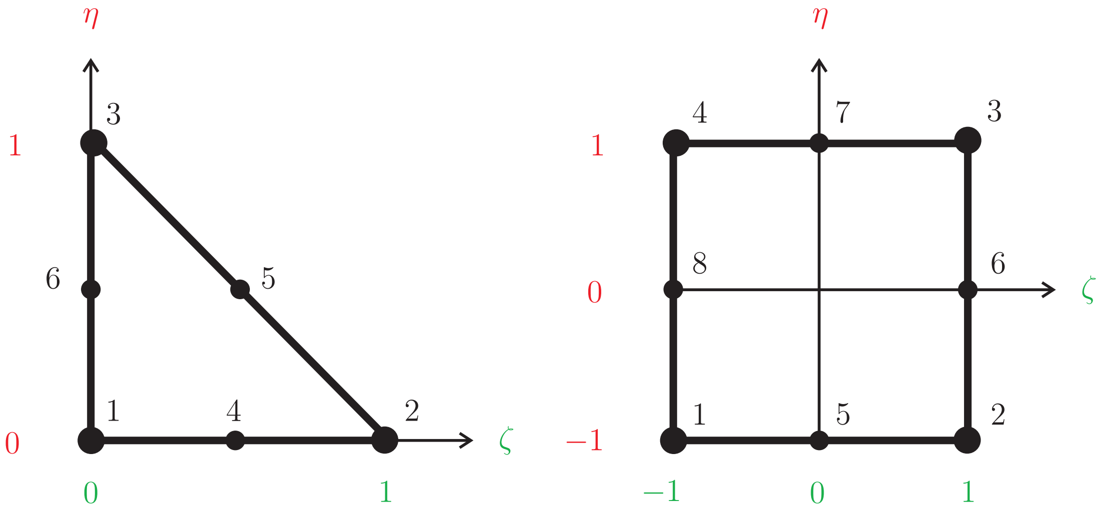
$$C_{ij}(P)u_j(P) + \sum_{E=1}^{N}\sum_{a=1}^{m} T_{ij}^{Ea}u_j^{Ea} = \sum_{E=1}^{N}\sum_{a=1}^{m} U_{ij}^{Ea}t_j^{Ea} \qquad i,j=1,2 \qquad (3.70)$$

Based on the selection of order of the shape functions, three types of systems of equation results

- **Isoparametric**: Same order of shape functions for variables and geometry
- **Superparametric**: Higher order shape functions for variables and low order for geometry
- **Subparametric**: Low order shape functions for variables and higher order for geometry

3.4.3 Three-dimensional implementation

In three-dimensional problems, the surface Γ may be discretised with either triangular or quadrilateral elements. Isoparametric boundary elements can be used to approximate both the geometry and solution variables. The local variables ζ and η are used to construct the shape functions. In case of three-dimensional triangular boundary element, $0 \le \zeta \le 1$ and $0 \le \eta \le 1$. Similarly, for quadrilateral elements, $-1 \le \zeta \le 1$ and $-1 \le \eta \le 1$. The two different types of three-dimensional elements are shown in Figure 3.4.



(a) Triangular element (b) Quadrilateral element

Figure 3.4: 3D quadratic boundary elements

$$x_j(\zeta,\eta) = \sum_{a=1}^{m} N_a(\zeta,\eta) x_j^a \tag{3.71}$$

$$u_j(\zeta,\eta) = \sum_{a=1}^{m} N_a(\zeta,\eta) u_j^a \tag{3.72}$$

$$t_j(\zeta,\eta) = \sum_{a=1}^{m} N_a(\zeta,\eta) t_j^a \tag{3.73}$$

The shape functions for a quadratic triangular element can be written in terms of local variable

$$
\begin{aligned}
N_1(\zeta,\eta) &= (1-\zeta-\eta)(1-2\zeta-2\eta) && (3.74)\\
N_2(\zeta,\eta) &= \zeta(1-\zeta) && (3.75)\\
N_3(\zeta,\eta) &= \eta(1-\eta) && (3.76)\\
N_4(\zeta,\eta) &= 4\zeta(1-\zeta-\eta) && (3.77)\\
N_5(\zeta,\eta) &= 4\zeta\eta && (3.78)\\
N_6(\zeta,\eta) &= 4\eta(1-\zeta-\eta) && (3.79)
\end{aligned}
$$

Using the steps followed in two-dimensions, the discretised equation for three-dimensions can be written as

$$C_{ij}(P)u_j(P)+\sum_{E=1}^{N}\sum_{a=1}^{m}T_{ij}^{Ea}u_j^{Ea}=\sum_{E=1}^{N}\sum_{a=1}^{m}U_{ij}^{Ea}t_j^{Ea} \qquad i,j=1,2,3 \tag{3.80}$$

The coefficients T_{ij}^{Ea} and U_{ij}^{Ea} consist of the double integral as given below

$$
\begin{aligned}
T_{ij}^{Ea} &= \int_{-1}^{1}\int_{-1}^{1}T_{ij}(P,Q)N_a(\zeta,\eta)J(\zeta,\eta)d\zeta d\eta\\
U_{ij}^{Ea} &= \int_{-1}^{1}\int_{-1}^{1}U_{ij}(P,Q)N_a(\zeta,\eta)J(\zeta,\eta)d\zeta d\eta && (3.81)
\end{aligned}
$$

3.4.4 Matrix form of BIE

The BIE for two-dimensions can be written in a matrix form by considering Equation (3.70) with $i=x$ and $y=j$

$$
\begin{bmatrix} C_{xx} & C_{xy}\\ C_{yx} & C_{yy}\end{bmatrix}\begin{Bmatrix} u_x^{Ea}\\ u_y^{Ea}\end{Bmatrix}+\sum_{E=1}^{NE}\sum_{a=1}^{m}\begin{bmatrix} T_{xx}^{Ea} & T_{xy}^{Ea}\\ T_{yx}^{Ea} & T_{yy}^{Ea}\end{bmatrix}\begin{Bmatrix} u_x^{Ea}\\ u_y^{Ea}\end{Bmatrix}=
$$

$$
\sum_{E=1}^{NE}\sum_{a=1}^{m}\begin{bmatrix} U_{xx}^{Ea} & U_{xy}^{Ea}\\ U_{yx}^{Ea} & U_{yy}^{Ea}\end{bmatrix}\begin{Bmatrix} t_x^{Ea}\\ t_y^{Ea}\end{Bmatrix} \tag{3.82}
$$

Placing the load point at each node and performing the integration results in a set of linear algebraic equations in the following form

$$[H]\{u\} = [G]\{t\} \tag{3.83}$$

The matrices $[H]$ and [G] contain known integrals of T_{ij}, U_{ij}, shape function and Jacobians. It can be seen the the term $C_{ij}(P)$ contributes to the diagonal terms of matrix $[H]$ when both the source and field points are at the same node. Therefore, for diagonal terms: $H_{ij} = C_{ij}\delta_{ij} + T_{ij}$.

A similar matrix formulation can be obtained for a three-dimensional case,

$$\begin{aligned}
&\begin{bmatrix} C_{xx} & C_{xy} & C_{xz} \\ C_{yx} & C_{yy} & C_{yz} \\ C_{zx} & C_{zy} & C_{zz} \end{bmatrix} \begin{Bmatrix} u_x^{Ea} \\ u_y^{Ea} \\ u_z^{Ea} \end{Bmatrix} + \sum_{E=1}^{NE}\sum_{a=1}^{m} \begin{bmatrix} T_{xx}^{Ea} & T_{xy}^{Ea} & T_{xz}^{Ea} \\ T_{yx}^{Ea} & T_{yy}^{Ea} & T_{yz}^{Ea} \\ T_{zx}^{ea} & T_{zy}^{Ea} & T_{zz}^{Ea} \end{bmatrix} \begin{Bmatrix} u_x^{Ea} \\ u_y^{Ea} \\ u_z^{Ea} \end{Bmatrix} = \\
&\qquad \sum_{E=1}^{NE}\sum_{a=1}^{m} \begin{bmatrix} U_{xx}^{Ea} & U_{xy}^{Ea} & U_{xz}^{Ea} \\ U_{yx}^{Ea} & U_{yy}^{Ea} & U_{yz}^{Ea} \\ U_{zx}^{Ea} & U_{zy}^{Ea} & U_{zz}^{Ea} \end{bmatrix} \begin{Bmatrix} t_x^{Ea} \\ t_y^{Ea} \\ t_z^{Ea} \end{Bmatrix}
\end{aligned} \tag{3.84}$$

The above equations can be written in a similar matrix form as obtained for the two-dimensional case.

3.4.5 Solution of BIE matrix system

The unknowns in the matrix formulation for both 2D and 3D cases appear on both sides of the equation. Thus, Equation (3.83) contains twice as many unknowns as the numbers of equations. This can be resolved by the application of a sufficient number of boundary conditions. This is normally achieved by prescribing either the the displacement component or the traction component in each coordinate direction, at each node. Therefore using column swapping between matrix H and G results in the following expression,

$$[A]\{x\} = [B]\{y\} \tag{3.85}$$

Matrices A and B contains the columns of H and G, the vector x consists of displacement and traction components that are still unknown while the vector y contains the known traction and displacement components that are prescribed as boundary conditions. Since both B and y are known, their multiplication yields

$$[A]\{x\} = \{b\} \tag{3.86}$$

The above equation can be solved using available techniques. Direct solvers such as partially pivoted Gauss elimination method and LU decomposition are successful. Because, the matrix A is fully populated and un-symmetric, Equation (3.86) can only be solved using GMRES and BiCG-Stab iterative solvers. The solution of tractions and displacements at the boundary can be obtained from Equation (3.86). The solution of stresses and displacement at any internal point can be obtained using Equation (3.25).

3.4.6 Scaling

The solution vector $\{x\}$ in Equation (3.86) consists of a mixture of both traction and displacement components with different units. In most practical engineering analysis, the values of traction terms exceed those of the displacement terms by several orders of magnitude. This suggests using a scalar scaling factor ψ to avoid ill-conditioning as a result of these differences in magnitude. Equation (3.83) can be written in the following scaled form

$$[H]\{u\} = \psi[G]\{t^{'}\} \tag{3.87}$$

where

$$\{t^{'}\} = \psi^{-1}\{t\} \tag{3.88}$$

An appropriate value of the scaling factor ψ can be determined by consideration of the geometric size and the material properties of the problem under analysis.

3.4.7 Internal solution

In some of the engineering applications stresses and displacements are also required to be calculated within the material domain. Once the displacements and tractions are available at the boundary, the displacement components at any internal point p can be easily obtained through Equation (3.25). Similarly, the strain at any point p can be obtained by the differentiation of Equation (3.25) with respect to the coordinate direction k,

$$u_{i,k}(p) = \int_\Gamma U_{ij,k}(p,Q)t_j d\Gamma - \int_\Gamma T_{ij,k}(p,Q)u_j d\Gamma \tag{3.89}$$

In the above equation $U_{ij,k}$ and $T_{ij,k}$ are the derivatives of the fundamental solutions, and the application of Hooke's law results in the following expression

$$\sigma_{ik}(p) = \int_\Gamma D_{jik}(p,Q)t_k d\Gamma - \int_\Gamma S_{jik}(p,Q)u_k d\Gamma \tag{3.90}$$

where D_{kij} and S_{kij} are obtained from $U_{ij,k}$ and $T_{ij,k}$ for a two-dimensional case, as given below

$$\begin{aligned} S_{jik} &= \frac{\mu}{2\pi(1-\nu)r^2} n_i [2\nu \frac{\partial r}{\partial x_j}\frac{\partial r}{\partial x_k} + (1-2\nu)\delta_{jk}] \\ &+ \frac{\mu}{2\pi(1-\nu)r^2} n_k [2\nu \frac{\partial r}{\partial x_i}\frac{\partial r}{\partial x_j} + (1-2\nu)\delta_{ij}] \\ &+ \frac{\mu}{2\pi(1-\nu)r^2} n_j [2(1-2\nu) \frac{\partial r}{\partial x_i}\frac{\partial r}{\partial x_k} + (1-4\nu)\delta_{ik}] \\ &+ \frac{\mu}{\pi(1-\nu)r^2} r_n [2(1-2\nu)\delta_{ik}\frac{\partial r}{\partial x_j} + \nu(\delta_{jk}\frac{\partial r}{\partial x_i} + \delta_{ij}\frac{\partial r}{\partial x_k}) \\ &-4\frac{\partial r}{\partial x_i}\frac{\partial r}{\partial x_j}\frac{\partial r}{\partial x_k}] \end{aligned} \tag{3.91}$$

$$D_{jik} = \frac{\mu}{4\pi(1-\nu)r}[2(1-2\nu)(\delta_{jk}\frac{\partial r}{\partial x_i} + \delta_{ij}\frac{\partial r}{\partial x_k} + \delta_{ik}\frac{\partial r}{\partial x_j}) + 2\frac{\partial r}{\partial x_i}\frac{\partial r}{\partial x_j}\frac{\partial r}{\partial x_k}] \tag{3.92}$$

There exists a higher order singularity because the fundamental solution is of order $\frac{1}{r}$ singularity and the derivative will be of order $\frac{1}{r^2}$. But in this case the source point does not lie on the boundary, so no singular integration is required in computing internal results but some care must have to be taken for internal points close to the boundary. Such care might involve using a higher order Gauss Legendre scheme or other useful schemes.

Chapter 4

Non-Uniform Rational B-Splines

4.1 Overview

Nonuniform rational B-splines (NURBS) are commonly used in computer aided design (CAD), computer aided manufacturing (CAM), computer aided engineering (CAE) and computer graphics for generating and representing curves and surfaces. NURBS are very versatile and flexible, and can be easily integrated with many programs to create high quality designs and models. The modelling capabilities of NURBS vary from aerospace exterior panels and automobiles bodies (where high mathematical precision is required) to mobile phones and toys. In this chapter a brief overview of the various parametric curves, i.e. Bézier, B-spline and NURBS is presented. In each case the mathematical expressions and properties of the curves are discussed with the help of graphical illustrations precisely computed using the developed algorithm for boundary representation. In the two-dimensional topology optimisation method NURBS are used to represent the evolving structural geometry in a standard CAD representation. A detailed description of the parametric curves and surfaces is available in [36]. In addition, a historical overview on NURBS can be found in [97].

4.2 Bézier curves

The shape description in design problems can be generally divided into two classes. The first class of the design problems depends on the functional requirements according to which the basic shape is achieved through experimental evaluation or mathematical calculations; examples are aircraft wings, engine manifolds, etc. In the second class the basic design is based on the functional as well as on the aesthetic requirements frequently termed as ab initio design. The most common examples are car bodies, aircraft fuselages, furniture and glassware. Pierre Bézier developed free form curves and surfaces known as Bézier curves for ab initio design [36] using the Bernstein basis or polynomial approximation functions. The NURBS curve is a generalized form of the Bézier curve. Mathematically, a parametric Bézier curve is is defined by

$$P(t) = \sum_{i=0}^{n} B_i J_{n,i}(t) \quad , \quad 0 \leq t \leq 1 \tag{4.1}$$

where the geometric coefficients B_i are called the control points, which form the control polygon, and the Bézier, or Bernstein basis function $J_{n,i}(t)$ is

$$J_{n,i}(t) = \left(\begin{array}{c} n \\ \\ i \end{array} \right) t_i (1-t)^{n-i} \quad , \quad (0)^0 \equiv 1 \tag{4.2}$$

with

$$\left(\begin{array}{c} n \\ \\ i \end{array} \right) = \frac{n!}{i!(n-i)!} \quad , \quad 0! \equiv 1 \tag{4.3}$$

A simple Bézier curve is shown in Figure 4.1. Some important properties of the Bézier curves are summarised as follows [97]

- The degree of the polynomial defining the curve segment is one less than the number of control points.
- The curve uses the first control point as the starting point and ends at the last

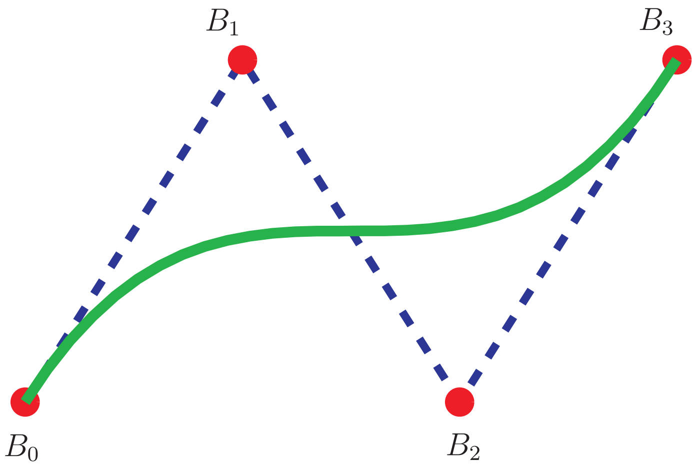


Figure 4.1: A Bézier curve and its control polygon

control point, but generally does not cross the inner control points.

- The tangent vectors at the ends of the curve have the same direction as the first and last polygon spans, respectively.
- The curve is always contained within the convex hull of the control polygon.
- The curve is unaffected during an affine transformation.

Some shortcomings of the Bézier curves are

- Equation (4.1) shows that each point on a Bézier curve is computed as a weighted sum of all control points. This means that each point is influenced by every control point. A change in the position of a control point modifies the complete curve (as shown in Figure 4.2) and the control is not sufficiently local.
- The degree of the curve is dependent on the number of control polygon vertices. A higher degree curve requires more vertices and thus limits its flexibility to define complex shapes.

Based on the above reasons, an alternative strategy can be to use curves which are piecewise polynomial and flexible enough to allow local control, e.g. B-splines (Basis splines).

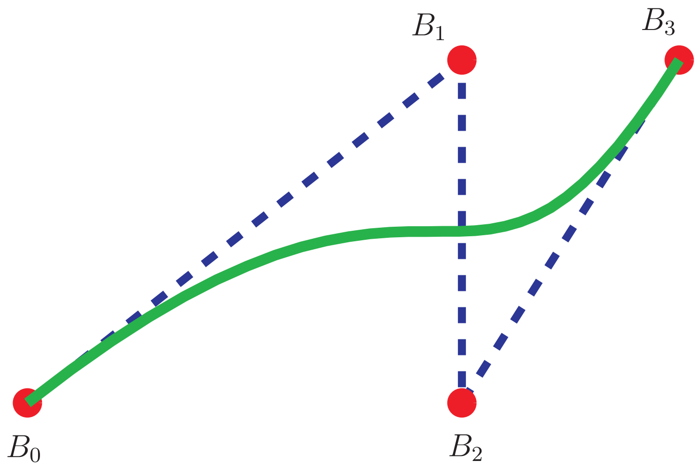


Figure 4.2: A modified Bézier curve and its control polygon

4.3 B-spline curve

B-spline curves are generalised form of the Bézier curves [97]. Similar to Bézier curves, the B-spline curves are controlled by a set of control points B_i lying on a control polygon. A B-spline curve consists of more than one Bézier curves with a built in mechanism of continuity between the curves segments. A few control points of the control polygon uniquely define and influence a segment in the B-spline curve. Moreover, the degree of the curve is independent of the number of control points. Mathematically a B-spline curve is given by [97]

$$P(t) = \sum_{i=1}^{n+1} B_i N_{i,k}(t) \quad , \quad t_{min} \leq t \leq t_{max}, \quad 2 \leq k \leq n+1 \tag{4.4}$$

where $P(t)$ is the position vector along the curve as a function of parameter t and B_i are the position vectors of the $n+1$ polygon vertices. The $N_{i,k}$ are the ith normalized B-spline basis functions of order k (degree of the curve = $k-1$) defined recursively as

$$N_{i,1}(t) = \begin{cases} 1 & \text{if } u_i \leq t \leq u_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{4.5a}$$

$$N_{i,k}(t) = \frac{(t-u_i)N_{i,k-1}(t)}{u_{i+k-1}-u_i} + \frac{(u_{i+k}-t)N_{i+1,k-1}(t)}{u_{i+k}-u_{i+1}} \tag{4.5b}$$

The u_i are the *knot values* of the *knot vector* $U = [u_1 \ \ u_2 \ \ . \ \ . \ \ . \ \ u_m]$. The knot vector satisfies the relation $u_i \leq u_{i+1}$; i.e. it is a monotonically non-decreasing series of real numbers. The number of control points, order of the curve and the size of knot vector m are related by

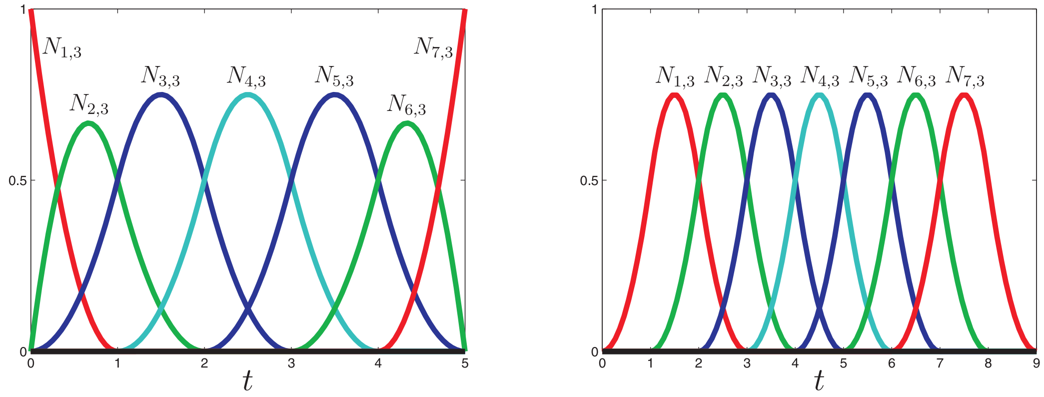
$$m = k + n + 1 \tag{4.6}$$

The most commonly used knot vectors are classified as periodic and open either in a uniform or nonuniform order. In a *periodic uniform* knot vector the knot values are evenly spaced whereas in an *open uniform* knot vector the first and last knot values are duplicated k times. *Periodic and open nonuniform* knot vectors may have either unequally spaced and/or multiple internal knots. Examples of the above knot vectors for $n + 1 = 7$ control polygon vertices and an order $k = 3$ (i.e. quadratic curve) are

$[0 \ \ 0 \ \ 0 \ \ 1 \ \ 2 \ \ 3 \ \ 4 \ \ 5 \ \ 5 \ \ 5]$	open uniform
$[0 \ \ 1 \ \ 2 \ \ 3 \ \ 4 \ \ 5 \ \ 6 \ \ 7 \ \ 8 \ \ 9]$	periodic uniform
$[0 \ \ 0 \ \ 0 \ \ 1 \ \ 2 \ \ 2 \ \ 4 \ \ 5 \ \ 5 \ \ 5]$	open nonuniform
$[0 \ \ 1 \ \ 2 \ \ 3 \ \ 3 \ \ 4 \ \ 5 \ \ 6 \ \ 7 \ \ 8]$	periodic nonuniform

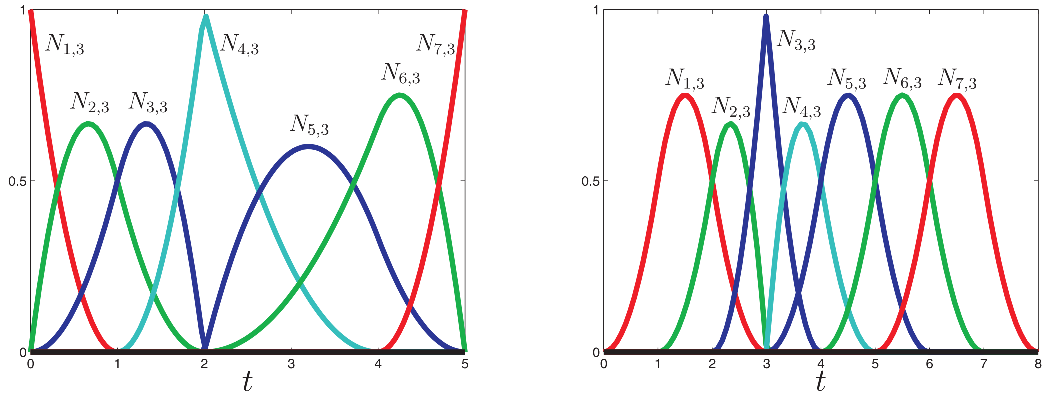
Figures 4.3a-d show the basis functions of the above discussed four different types of knot vectors using Equations (4.5a) and (4.5b) recursively. It is evident from the comparison of the basis functions that there exists symmetry in the basis functions of the uniform periodic and open knot vectors and that symmetry is lost for the nonuniform periodic and open knot vectors. In addition, cusps are developed in Figures 4.3c-d due to the multiple internal knot values in the nonuniform knot vectors.

Equation (4.4) shows that the selection of a knot vector has a strong influence on the basis functions $N_{i,k}$ of a B-spline curve. Applying the basis functions of the above discussed open uniform and nonuniform knot vectors and using Equation (4.4) the two B-spline curves obtained are shown in Figures 4.4 and 4.5, respectively. The number of intervals in the knot vector defines the number of segments in the B-spline curve. There



(a) Open uniform

(b) Periodic uniform



(c) Open nonuniform

(d) Periodic nonuniform

Figure 4.3: B-spline basis functions

are five intervals in the uniform knot vector: $0-1, 1-2, 2-3, 3-4, 4-5$ and as a result the B-spline curve has five segments. In the non-uniform B-spline curve one of the intervals starts and ends at the same point leading towards a cusp in the curve as shown in Figure 4.5. Furthermore, the curve passes through the first and last control points, i.e., B_1 and B_7, and its slope is equal to the slope of the first and last line segments of the control polygon. Figure 4.6 shows three B-spline curves obtained as a result of the movement of the control polygon vertex B_5 successively to two different positions, i.e., B_5' and B_5''. As a result of this local change three (or in general k) curve segments corresponding to the control polygon spans B_3B_4, B_4B_5 and B_5B_6, B_6B_7 are affected.

The most significant properties of the B-spline curves are

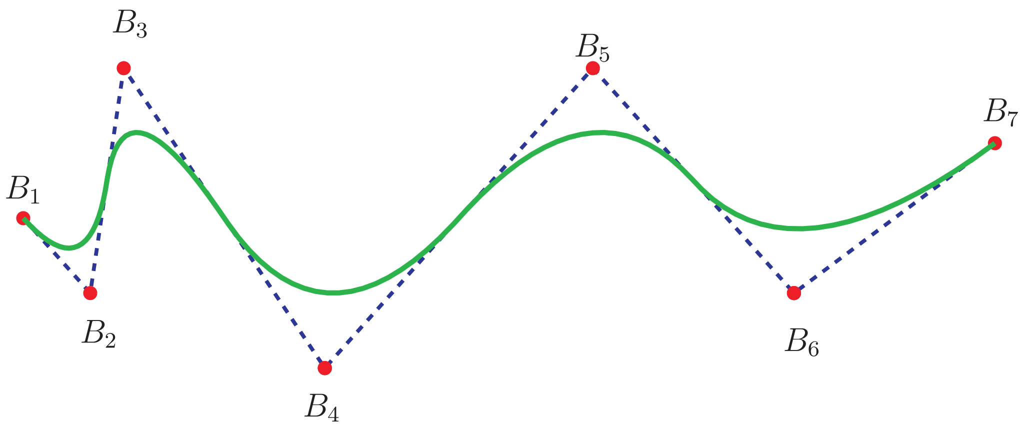


Figure 4.4: B-spline curve with open uniform knot vector

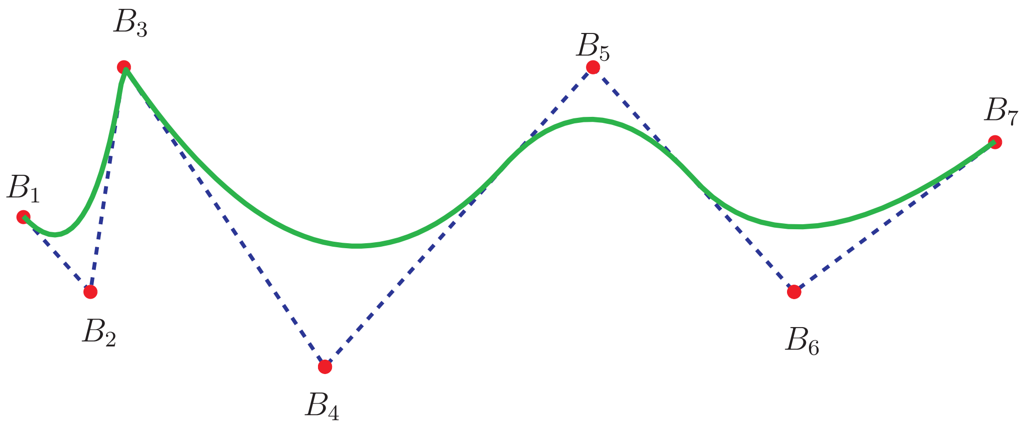


Figure 4.5: B-spline curve with open nonuniform knot vector

- The sum of the basis functions for any parameter value t is

$$\sum_{i=1}^{n+1} N_{i,k}(t) \equiv 1$$

- Each basis function satisfies the condition that

$$N_{i,k}(t) \geq 0$$

- The maximum order of the curve equals the number of control polygon vertices and the maximum degree is one less than the order of the curve.

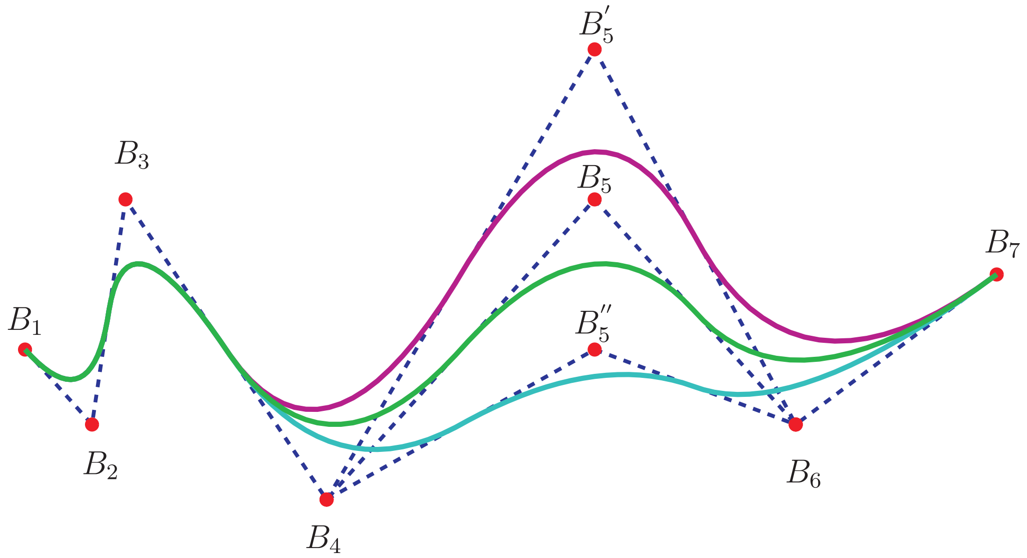


Figure 4.6: B-spline curves with different positions of the control point B_5

- If $n + 1 = k - 1$ and $U = [0\ \ 0\ \ 0\ \ .\ \ .\ \ .\ \ 1\ \ 1\ \ 1]$ then the resulting curve is a Bézier curve.
- Any point on the curve lies within the convex hull of k neighbouring points.
- In an affine transformation applied to the control polygon vertices transforms the curve.
- Strongly exhibits the variation diminishing property, i.e., the number of intersection points of any straight line with the curve are less than or equal to the number of intersection with the control polygon.

4.4 Rational B-spline curves

In a rational B-spline a weight is assigned to each control polygon vertex and that influences the shape of the basis functions. If a weight 1.0 is assigned to each vertex then it reduces to the form of a nonrational B-spline (or simply B-spline). The main advantage of a rational B-spline is that circles, conics and free form shapes commonly used in the computer graphics and CAD can be precisely represented. Rational curves are defined on the basis of homogeneous coordinates. Homogeneous coordinates are a used to represent N-dimensional coordinates with $N + 1$ numbers. In 2D, homogeneous coordinates represent a 2D point in terms of a 3D point with an additional non-zero coordinate axis h. For example a point $B = (x, y)$ is a point in Cartesian coordinates, the corresponding point in the 3D space is $B^h = (hx, hy, h)$. Conversely, a 3D point is converted back to a 2D one by dividing the first two coordinates by the third one.

Mathematically, a rational B-spline curve which is commonly known as NURBS can be obtained by modifying the B-spline equation (i.e., (4.4)) using homogenous coordinates as

$$P(t) = \sum_{i=1}^{n+1} B_i^h N_{i,k}(t) \quad , \quad t_{min} \leq t \leq t_{max}, \quad 2 \leq k \leq n+1 \tag{4.7}$$

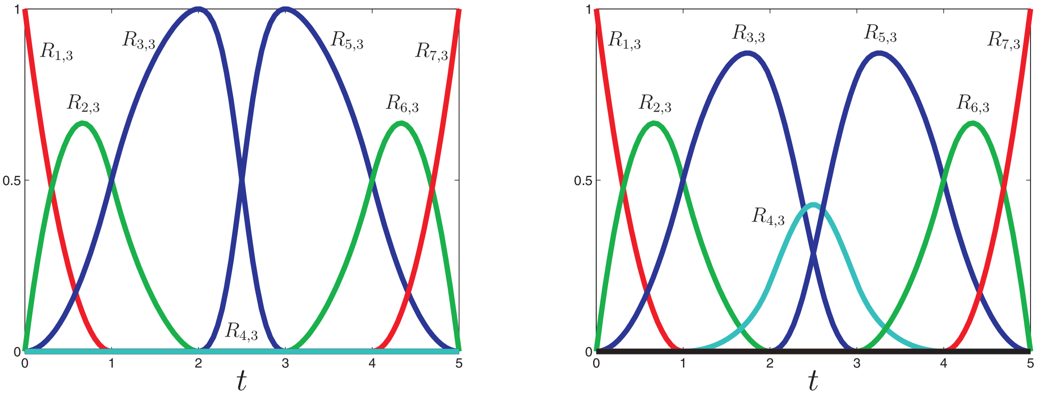
Dividing Equation (4.7) by the homogeneous coordinate project back to the Cartesian space results into a rational B-spline curve

$$P(t) = \frac{\sum_{i=1}^{n+1} B_i h_i N_{i,k}(t)}{\sum_{i=1}^{n+1} h_i N_{i,k}(t)} = \sum_{i=1}^{n+1} B_i R_{i,k}(t) \tag{4.8}$$

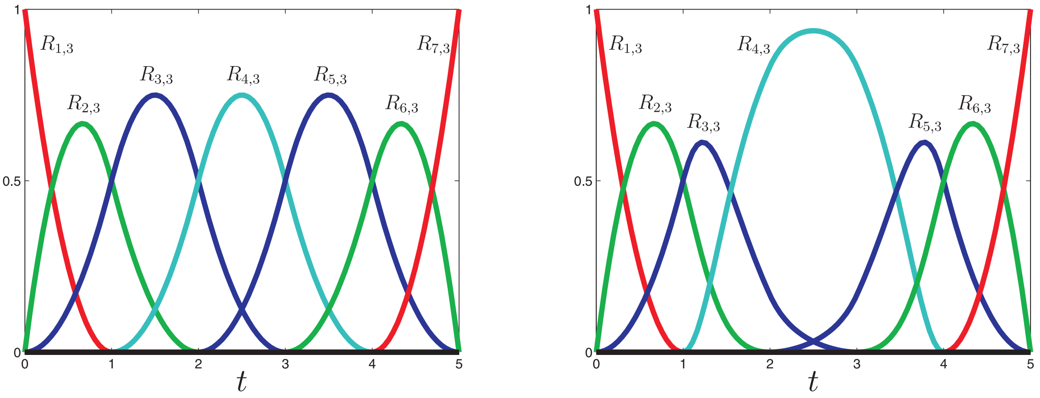
where $R_{i,k}(t)$ are the rational basis functions and $h_i \geq 0$ for all i.

$$R_{i,k}(t) = \frac{h_i N_{i,k}(t)}{\sum_{i=1}^{n+1} h_i N_{i,k}(t)} \tag{4.9}$$

Figure 4.7 shows the Rational B-spline basis functions obtained using Equations (4.5) and (4.9) for $n+1 = 7$, $k = 3$, an open uniform knot vector $U = [0\ \ 0\ \ 0\ \ 1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 5\ \ 5]$ and a homogenous coordinate vector $H = [1\ \ 1\ \ 1\ \ h_4\ \ 1\ \ 1\ \ 1]$, with $0 \leq h_4 \leq 5$. Figures 4.7a-d show the effect of homogenous coordinates h on the rational B-spline basis functions. In addition, a comparison has been shown in Figure 4.8 to highlight the effect of homogenous coordinates on the resulting rational B-spline curve. In Figure 4.7c with $h_4 = 1$ the rational B-spline basis functions are identical to the corresponding nonrational B-spline basis functions. Figure 4.7a shows that when $h_4 = 0$, $R_{4,3} = 0$ in each interval. Therefore, the control polygon vertex B_4 has no influence on the shape of the resulting rational B-spline curve and thus, vertices B_3 and B_5 are connected by a straight line segment. Further, Figures 4.7a-d show that an increase in h_4 increases $R_{4,3}$ and at the same time decreases $R_{3,3}$ and $R_{3,5}$. Thus, in Figure 4.8 the corresponding B-spline curve



(a) Periodic uniform (b) Open uniform



(c) Periodic nonuniform (d) Open nonuniform

Figure 4.7: Rational B-spline basis functions

is pulled closer to B_4 and hence, this verify the additional blending capabilities of the rational B-spline basis functions. In Figure 4.9 a similar behaviour can be observed for a higher order (i.e. $k = 4$) rational B-spline curve. In comparison with Figure 4.8, the high order curve for $h_4 = 0$ does not pass through B_3 and B_5 and as a result a straight line segment is replaced with a curve segment in that region.

Rational B-spline is a generalisation of the nonrational B-spline and therefore, their characteristics are almost identical to each other. In particular:

- The sum of the basis functions for any parameter value t is

$$\sum_{i=1}^{n+1} R_{i,k}(t) \equiv 1$$

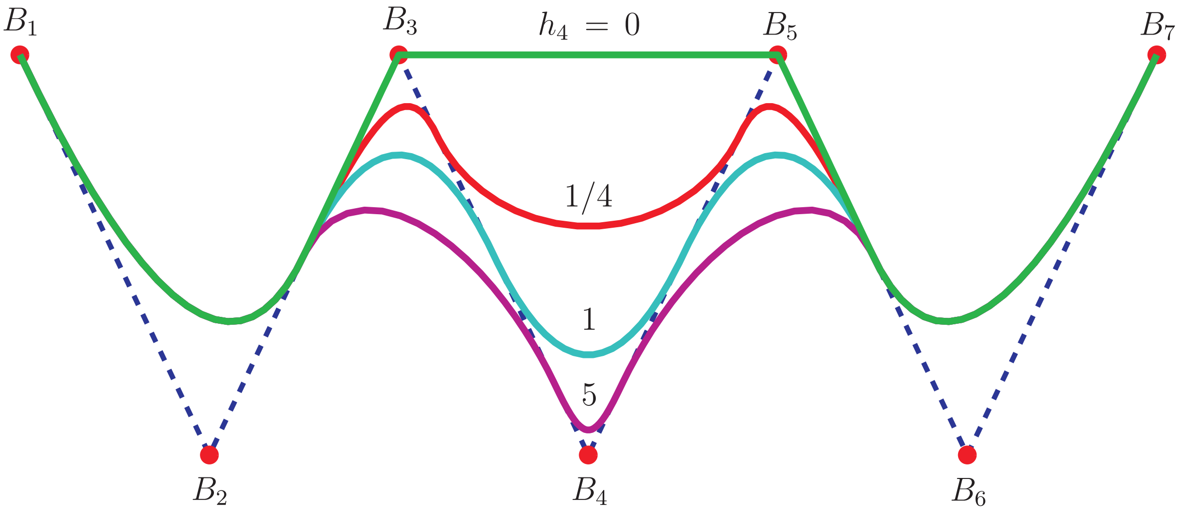


Figure 4.8: NURBS curves and its control polygon

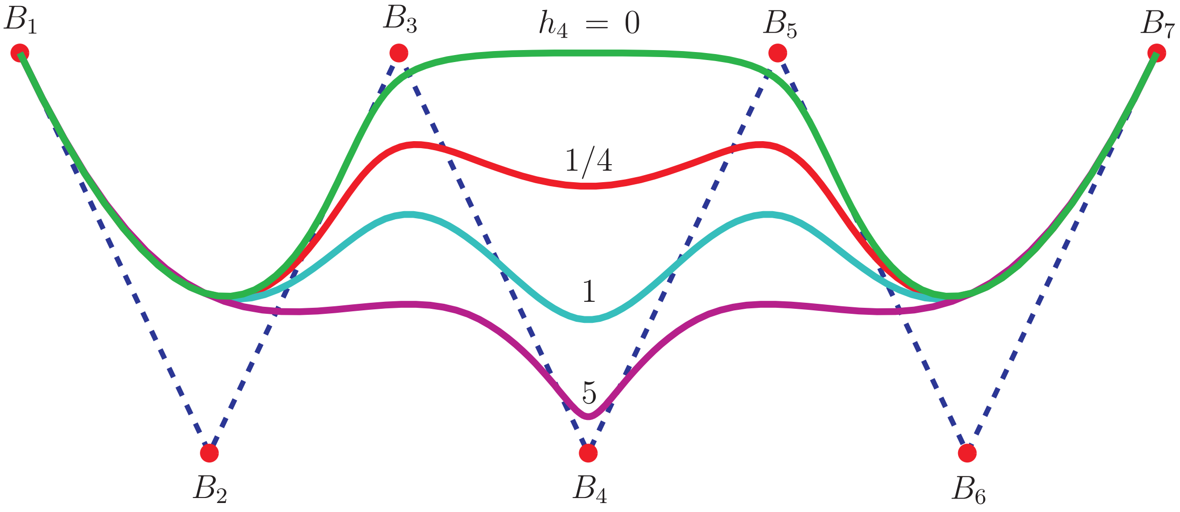


Figure 4.9: NURBS curves and its control polygon

- Each basis function satisfies the condition that

$$R_{i,k}(t) \geq 0$$

- The maximum order of the curve is equal to the number of control polygon vertices.
- If $h_i = 1$ for all i, then the rational basis functions are identical to the nonrational basis functions.
- A rational B-spline curve generally follows the shape of the control polygon.

- A rational B-spline curve lies within the convex hull of formed by k successive control polygon vertices when when $h_i > 0$.

4.5 NURBS curve fitting

In the previous sections B-spline curves (rational or nonrational) have been generated from a set of control points. In other words, data points are generated from the known numbers of control polygon vertices, knot vector and order of the curve which makes a B-spline curve. In this section the method of obtaining the control polygon vertices to generate a B-spline curve from a set of *known* data points is discussed in detail. Here we consider the case of a NURBS curve when all the control polygon vertices have $h_i = 1$ for all i, and the corresponding rational B-spline basis functions are equal to its nonrational counterparts, i.e., $R_{i,k}(t) = N_{i,k}(t)$.

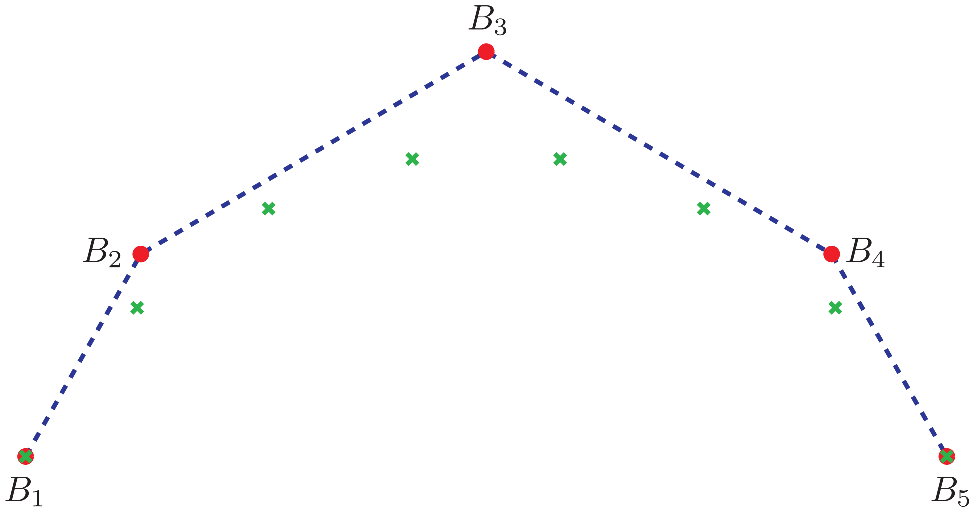


Figure 4.10: Determining a B-spline control polygon for a given set of data points

In order to proceed a schematic of the problem is depicted in Figure 4.10. Let us suppose that a set of j data points (represented by $\times$) lie on the B-spline curve, then it must satisfy Equation (4.4). Mathematically, for j data points Equation (4.4) can be written as

$$
\begin{aligned}
D_1(t_1) &= N_{1,k}(t_1)B_1 + N_{2,k}(t_1)B_2 + \cdots + N_{n+1,k}(t_1)B_{n+1} \\
D_2(t_2) &= N_{1,k}(t_2)B_2 + N_{2,k}(t_2)B_2 + \cdots + N_{n+1,k}(t_2)B_{n+1} \\
&\vdots \\
D_j(t_j) &= N_{1,k}(t_j)B_2 + N_{2,k}(t_j)B_2 + \cdots + N_{n+1,k}(t_j)B_{n+1}
\end{aligned}
$$

where $2 \leq k \leq n+1 \leq j$. The above system of equations can be written in a matrix form as

$$
[D] = [N]\,[B] \tag{4.10}
$$

where

$$
\begin{aligned}
[D]^T &= [D_1(t_1) \;\; D_2(t_2) \;\; \cdots \;\; D_j(t_j)] \\
[B]^T &= [B_1 \qquad B_2 \qquad \cdots \qquad B_{n+1}] \\
[N] &= \begin{pmatrix} N_{1,k}(t_1) & N_{2,k}(t_1) & \cdots & N_{n+1,k}(t_1) \\ N_{1,k}(t_2) & N_{2,k}(t_2) & \cdots & N_{n+1,k}(t_2) \\ \vdots & \vdots & \cdots & \vdots \\ N_{1,k}(t_j) & N_{2,k}(t_j) & \cdots & N_{n+1,k}(t_j) \end{pmatrix}
\end{aligned}
$$

The parameter t_j which measures the distance between the data points along the B-spline curve is approximated using the chord length between these points [97]. In a set of j data points the value at mth data point is

$$
\begin{aligned}
t_1 &= 0 \\
\frac{t_m}{t_{\max}} &= \frac{\sum_{r=2}^{m} |D_r - D_{r-1}|}{\sum_{r=2}^{j} |D_r - D_{r-1}|} \qquad m \geq 2
\end{aligned}
$$

The maximum value of the knot vector used is based on the $t_{\max}$ value. In cases where the number of data points are equal to the number of control polygon vertices, i.e. $2 \leq k \leq n+1 = j$, the corresponding $[N]$ matrix is square and the positions of the vertices are obtained as

$$[B] = [N]^{-1}[D] \qquad 2 \leq k \leq n+1 = j \tag{4.11}$$

Therefore, the resulting B-spline curve passes through each data point. The connectivity between the curve segments is everywhere C^{k-2} but may not be very smooth [97]. In order to obtain a smoother curve the control polygon vertices should be less than the number of data points, i.e., $2 \leq k \leq n+1 < j$. However, this results into a non-square $[N]$ matrix. A square $[N]$ matrix can be obtained by multiplying Equation (4.10) on both sides with $[N]^T$, using the fact that a matrix multiplied by its transpose is always square. As a result the control polygon vertices are given by

$$[N]^T\ [D] = [N]^T\ [N]\,[B]$$

$$[B] = \left[[N]^T\ [N]\right]^{-1}[D] \tag{4.12}$$

In writing (4.11) and (4.12) it is assumed that $[N]$ is known. The basis functions $N_{i,k}(t_j)$ can be obtained from known numbers of control polygon vertices $n+1$, order of the curve k, and the parameter value along the curve.

4.6 Conclusions

This chapter starts with a brief description of the mathematical definition of Bézier curves and their properties. Then, the B-spline curve mathematical definition is presented and some examples are discussed to highlight its flexibility and local control properties. Moreover, the rational B-splines mathematical formulation and their properties are discussed in detail. Finally, NURBS curve fitting procedure has been introduced and its implementation procedure is explained through mathematical formulation.

To conclude this chapter some important properties of NURBS are summarised as follows

- The available mathematical formulation equipped NURBS to exactly describe virtually any desired shape, including straight lines, conic sections, free form curves and surfaces.
- Bézier and B-spline curves are used as generalized from of NURBS.
- NURBS provide excellent flexibility to define and edit numerous shapes with fewer control points and weights and this makes NURBS attractive for design optimisation problems.
- NURBS are invariant in any transformation, i.e., rotation, translation and scaling.
- Evaluation of NURBS are based on reasonably fast and computationally stable and accurate algorithms [87].

In spite of several advantages, NURBS have also some drawbacks [87]

- In order to represent standard shapes with NURBS, e.g., a full circle, seven control points with their corresponding weights and ten knots are required and as a result extra storage is needed. However, in traditional representation only the centre, the radius and plane of the circle is required.

- NURBS are very sensitive to weight selection and an improper selection can result in a very bad parametrization, which can lead towards distorted shapes.

Chapter 5

LSM and BEM based structural optimisation

5.1 Overview

This chapter presents a new two-dimensional level set based structural optimisation method. The proposed method uses the BEM as a structural analysis tool, ESO as an optimisation technique and the evolving geometry is described with a standard CAD representation, i.e. NURBS. During the optimisation iterations, inefficient material is progressively removed from the low stressed regions and added to the high stressed regions of the structure. The LSM evolves the structural geometry towards an optimal design according to the optimisation criterion. The proposed optimisation method is tested against some benchmark examples in the field of structural optimisation to show its effectiveness.

5.2 Level set based evolutionary structural optimisation

This sections presents a new topology optimisation method based on the bi-directional ESO approach for two-dimensional linear elastic problems. Considering the drawbacks of the FE based BESO approach (Section 2.3.3), advantages of the BEM (Section 2.5), LSM (Section 2.3.4) and NURBS (Section 4.4), the proposed method is based on the BEM, LSM, and NURBS based geometry representation. In the present implementation, the von Mises stress σ_V criterion is used to remove material from the low stressed regions and add it to the high stressed regions of the structure. It should be noted that the material removal and addition only take place through the movement of the existing structural boundary and no hole insertion takes place during the optimisation iterations. The BE analysis provides σ_V at each node of the structural boundary. For clarification we present the definition of von Mises stress, σ_V, as

$$\sigma_V = \frac{1}{\sqrt{2}}\sqrt{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2} \tag{5.1}$$

where σ_1, σ_2 and σ_3 are the principal stresses. During the optimisation process, inefficient material, which needs to be progressively removed, is identified as the regions in the locality of nodes satisfying

$$\sigma_V < RR\,\sigma_{Vmax} \tag{5.2}$$

where RR is the removal ratio and σ_{Vmax} is the maximum von Mises stress in the initial design. Similarly, regions where material should be added are identified as those in the locality of the boundary nodes with high stresses satisfying

$$\sigma_V > \min(\sigma_{Vmax}, \sigma_Y) \tag{5.3}$$

where σ_Y is the material's yield stress. During the optimisation process the value of RR is periodically increased through an incremental removal ratio RR_i as,

$$RR = RR + RR_i \tag{5.4}$$

Each time RR_i is updated, when the combined volume of material experiencing $\sigma_V < RR\,\sigma_{Vmax}$ falls below a threshold of $0.4V$ (where V is the volume at the current iteration). Once the low and high stressed nodes are identified the LSM is then used to evolve the structural geometry.

5.3 Optimisation algorithm

Consider an elastic structure with design domain Ω and boundary Γ as shown in Figure 5.1. The idea to enhance the performance of a structure based on providing maximum possible stiffness against the applied loads is the basis of the maximum stiffness criterion. However, simply seeking to maximise stiffness will lead to an increase in the weight of the structure, because the design space will become completely filled with material. In order to enhance the structural performance from both the stiffness and efficient material utilization points of view the concept of specific stiffness was developed [112], being defined as,

$$f_K = \frac{K}{V} \tag{5.5}$$

where K is the stiffness and V is the volume of the structure. An equivalent concept in terms of the compliance is the specific strain energy, f_U, which is the product of strain energy U and the volume V of the structure [21], i.e.

$$f_U = UV \tag{5.6}$$

The expression used for strain energy calculation is,

$$U = \int_\Gamma \frac{1}{2} t_i u_i d\Gamma \tag{5.7}$$

where t_i is the traction and u_i is the displacement at a given node i. In practice, since the product $t_i u_i$ is non-zero only over elements on which a non zero traction boundary condition has been prescribed (assuming there are no non-zero displacement constraints applied) the integral involved in Equation (5.7) conveniently reduces to the integral taken only over these elements.

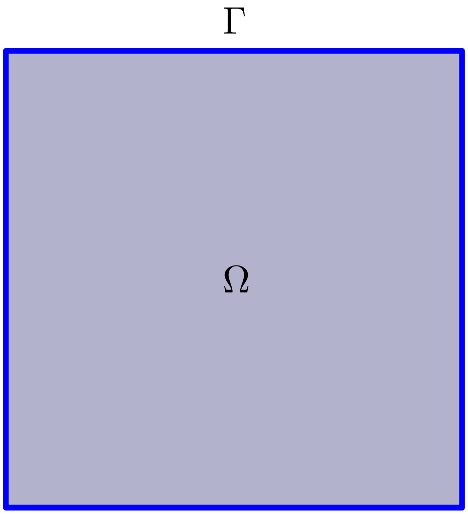


Figure 5.1: Definition of design domain

The optimisation progress can be monitored using the reduction in f_U, and the target volume fraction can be used as a stopping criterion. The volume fraction α at a given stage of the optimisation process can be defined as,

$$\alpha = V/V_0 \tag{5.8}$$

where V is the volume at the current iteration (this is interpreted as the area in a 2D representation) and V_0 the initial volume of the structure.

Finally, the optimisation problem can be stated as:

$$\text{Minimise: } f_U \tag{5.9}$$

$$\text{Subject to: } \alpha = \alpha_t$$

where α_t is the target volume fraction.

The proposed optimisation algorithm is illustrated in Figure 5.2 and summarised as follows:

1. Define structural geometry with applied loads and constraints.
2. Initialize level set grid with signed distance function to represent structural geometry implicitly.
3. Trace the zero level set contours and convert them into a standard CAD representation, i.e. NURBS.
4. Carry out boundary element analysis (BEA).
5. Compute velocity at each node point of the structural boundary using BE analysis results.
6. Extend boundary velocities to level set grid points in the narrow band.
7. Solve Equation (2.3) to update the level set function.
8. Repeat the above procedure from step 3, until the stopping criterion is satisfied.

The following sections discuss the implementation details of various steps used in the above optimisation algorithm. Many of these steps involve criteria of various types involving the comparison of stresses, volumes, etc., against various coefficients. These have been developed through extensive numerical testing on a range of optimisation problems. The proposed method uses the 2D version of the BEM analysis software Concept Analyst (CA) [118].

5.3.1 Structural geometry, loading and constraints

The optimisation process starts with the definition of structural geometry and the applied loading and constraints. The geometry of this initial structure is arbitrary, and is defined as a polygon in which each edge is a line segment which may be straight or curved. In

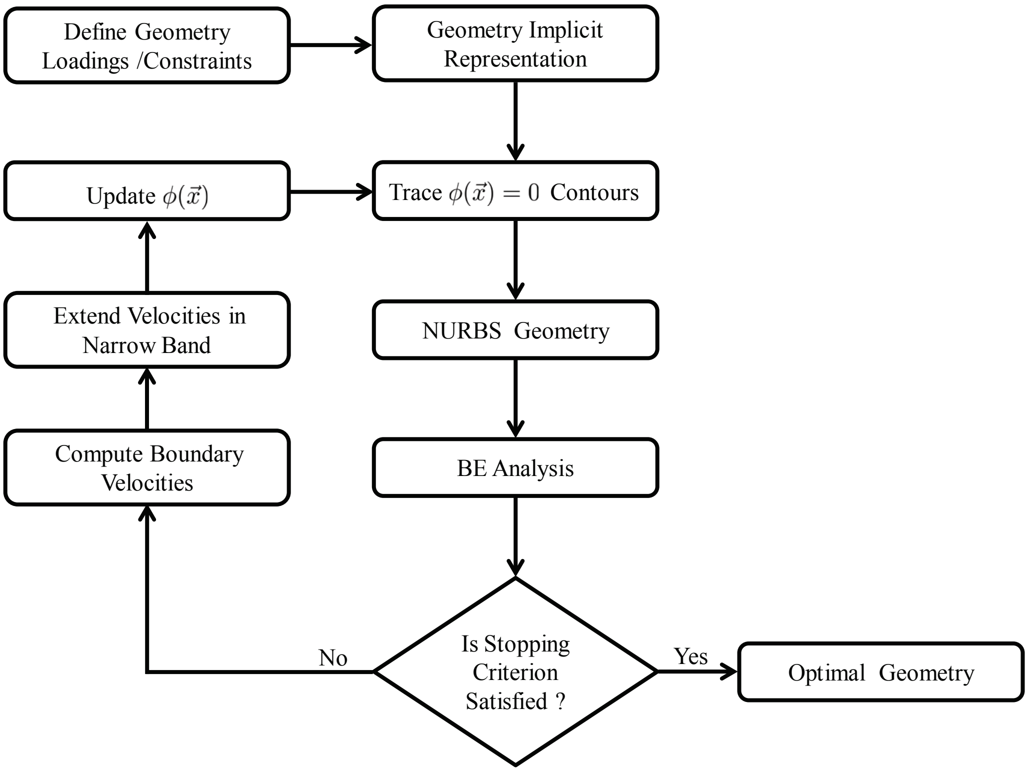


Figure 5.2: Optimisation flow chart

most research work of this type, the initial geometry is a simple rectangle. The boundary Γ of the initial design domain Ω is decomposed into three parts as shown in Figure 5.3, and given as

$$\Gamma = \Gamma_0 \cup \Gamma_1 \cup \Gamma_2 \tag{5.10}$$

where Γ_0 corresponds to Dirichlet boundary conditions (where displacements are zeros), Γ_1 corresponds to non-homogeneous Neumann boundary conditions (where tractions are prescribed) and Γ_2 corresponds to homogeneous Neumann boundary conditions (traction free and is allowed to vary during the optimisation process). The line segments (highlighted as red) with boundary conditions Γ_0 and Γ_1 remain fixed, and those with Γ_2 are allowed to be modified during the optimisation process.

5.3.2 Structural geometry implicit representation

The LSM is based on the Eulerian approach and works on an underlying Cartesian grid. A rectangular level set domain Ω_L is defined to capture all the possible geometry changes

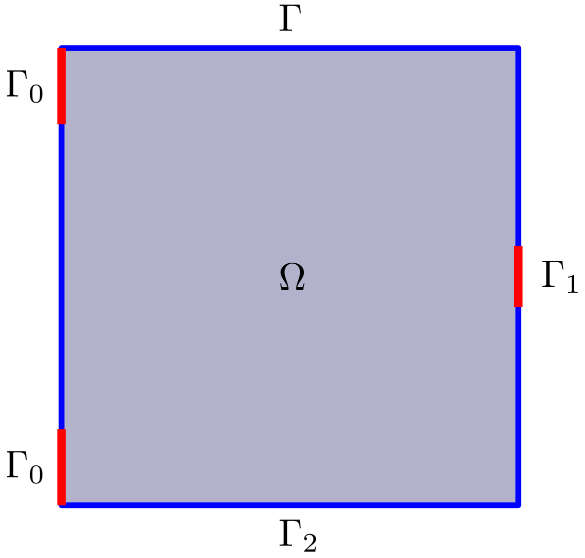


Figure 5.3: Definition of design domain

during the optimisation process. The proposed algorithm adds and removes material during the optimisation process. Therefore, the dimensions of Ω_L used in the current implementation are slightly bigger than the structural geometry. This allows some enlargement of the evolving geometry from the initial design.

In the numerical implementation, a suitable level set grid size, $d = 0.02D$, is used in the initial iterations, where D represents the largest dimension of the initial analysis model. During the optimisation process, once α reaches near the target volume fraction, a refined grid size $d = 0.01D$ is used afterwards. The use of a refined grid at the later stages of the optimisation process provides greater computational efficiency. The flexibility of using two different grid sizes is linked to the NURBS based representation of the evolving structural geometry (explained in Section 5.3.4). For complex design domain and boundary conditions the refined grid size should be used throughout the optimisation process.

After setting up the level set grid, the next is to represent the structural geometry implicitly. Therefore, the structural geometry is embedded as a higher dimensional function through the signed distance calculations, and this initializes the level set grid. The signed distance function $\phi(\vec{x})$ is defined as the minimum distance at a given level set grid point from the structural geometry. Further, we define Ω^- as the region contained within the boundary, Ω^+ as the union of the regions inside holes and the region of the design domain outside the boundary, and the contour $\partial\Omega$ as the interface between the non-overlapping regions Ω^- and Ω^+. These definitions are expressed as follows and shown in Figure 5.4.

$$\phi(\vec{x}) \begin{cases} < 0 & \vec{x} \in \Omega^- \\ = 0 & \vec{x} \in \partial\Omega \text{ or } \Gamma \\ > 0 & \vec{x} \in \Omega^+ \end{cases} \tag{5.11}$$

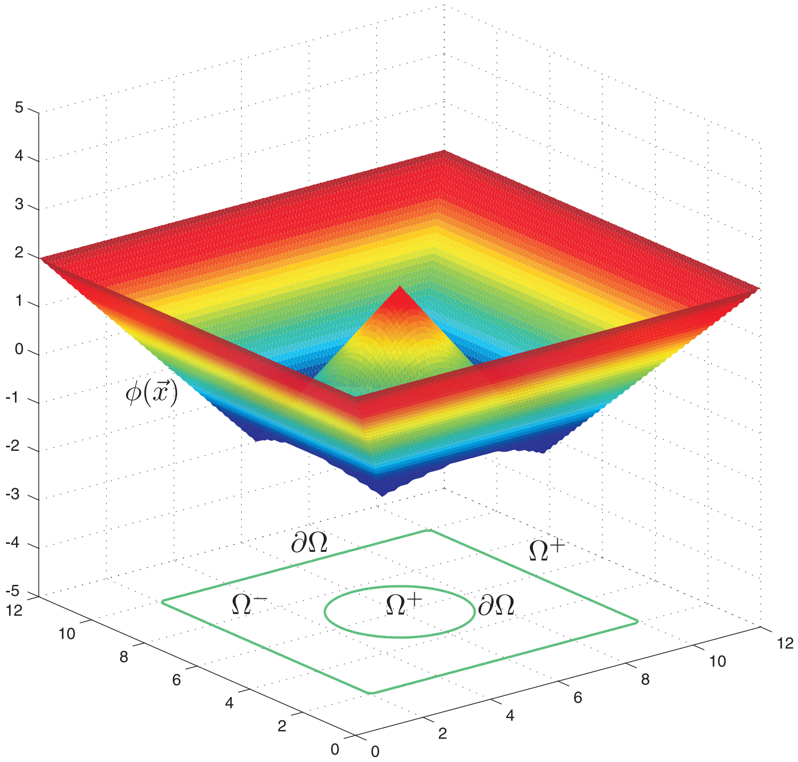


Figure 5.4: Geometry implicit representation

5.3.3 Algorithm for tracing the zero level set contours

During the optimisation process, at each iteration, the solution of the level set equation (i.e. Equation (2.3)) updates $\phi(\vec{x})$. As a result, the structural geometry is modified after each update of $\phi(\vec{x})$. In order to evaluate the structural performances, a BEM analysis is always required for the modified geometry. Therefore, the $\phi(\vec{x}) = 0$ contours (which represents the boundary of the modified geometry) need to be reconstructed from the level set grid.

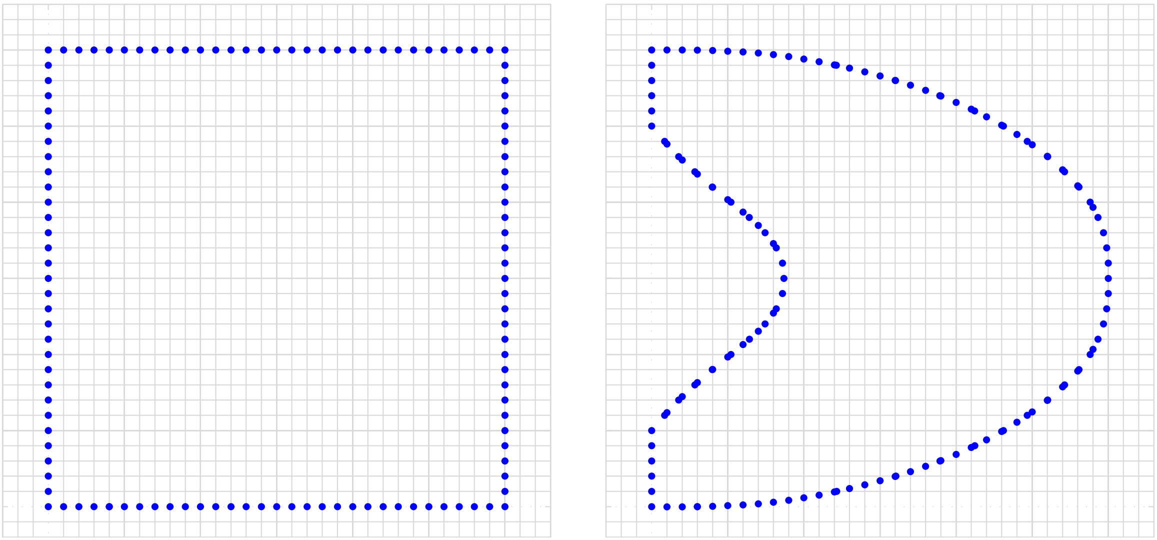
The $\phi(\vec{x}) = 0$ contours are traced with an efficient contour tracing algorithm developed within the CA software. This algorithm linearly interpolates the positions of the zero level set points at the intersections with the level set grid lines. The contour tracing algorithm starts from calculating the position of a zero level set intersection point, and proceeds to follow the contour $\phi(\vec{x}) = 0$ by locating adjacent intersection points, stopping when the starting point is reached and a closed contour has been defined.

In the current implementation, the algorithm traces the outer boundary in the first step, and searches internal cavities (holes) afterwards. As discussed in Section 5.3.1, the outer boundary is composed of multiple line segments. However, two NURBS curves describe the internal cavities, and are joined together at two points. The positions of these two points are known in advance. During the search for internal cavities, these two points are used as reference and the algorithm starts checking those grid cells near to these points. The algorithm terminates when there are no more zero level set contours to be traced in the computational domain. There are two advantages associated to this concept. Firstly, only those grid cells are checked where the zero level set exists, thereby reducing the computational cost of checking all the grid cells. Secondly, the intersection points obtained are in a regular order, through which a closed contour can easily be constructed.

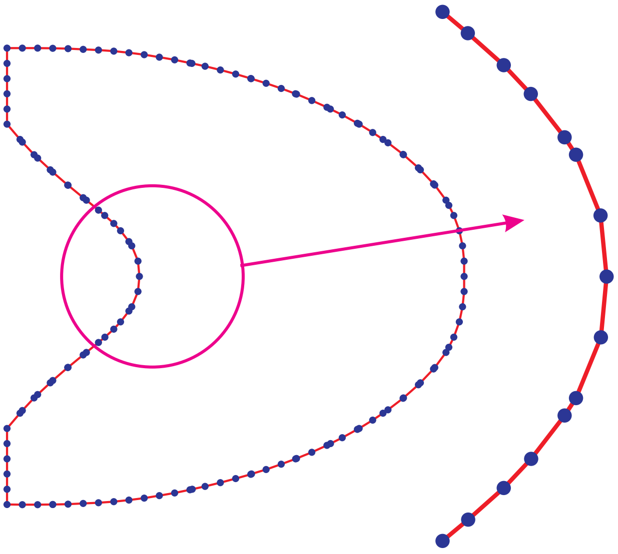
5.3.4 NURBS geometry

The positions of the intersection points calculated for the initial and the intermediate geometry at $\alpha = 0.70$ are displayed in Figure 5.5(a) and (b), respectively. It can be seen that, the adjacent intersection points are equidistant in the case of initial geometry. However, the distance varies considerably between the adjacent points for the intermediate geometry. In order to reconstruct the geometry, line segments are used to connect the zero level set intersection points as shown in Figure 5.5(c). This yields a non-smooth polygonal structural geometry with line segments of non-uniform length.

In the BE analysis if the zero level set intersection points are used directly as element



(a) Initial geometry (b) Intermediate geometry

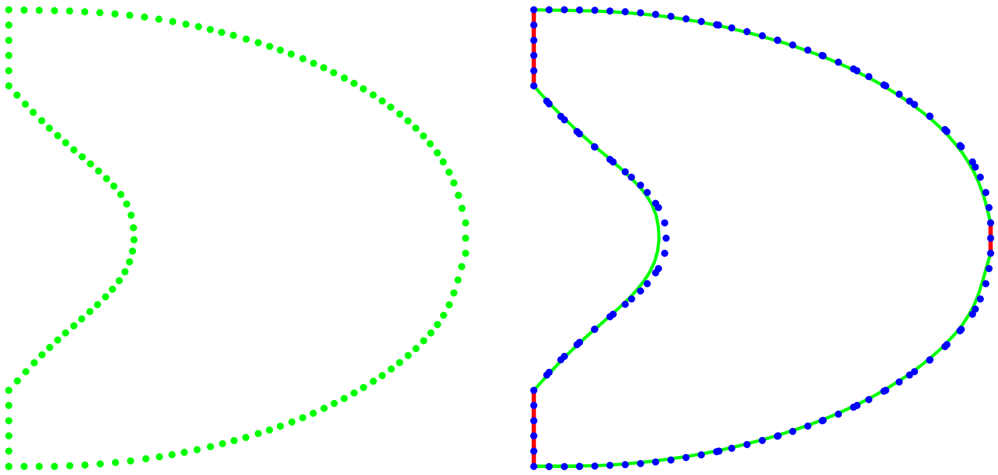


(c) Reconstructed geometry

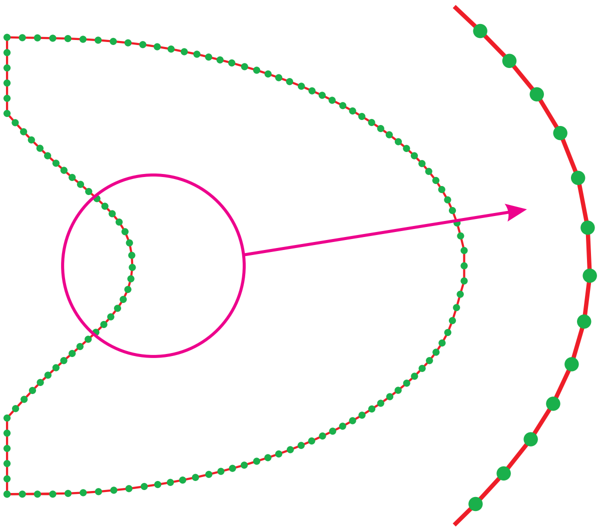
Figure 5.5: Zero level set intersection points at $\alpha = 0.70$

nodal points (as in [136]), two intersection points can lie very close to each other (for example see Figure 5.5(b)), and this can cause difficulties and instabilities during the BE analysis. In addition, the non-smoothness of the polygonal geometry can produce stress concentrations, which can mislead the optimisation process.

In order to overcome the above mentioned difficulties during the BE analysis, two options may be considered within this study. In the first option, a Laplacian smoothing procedure can be used, which iteratively modify the position of a point by simply averaging the location of the neighboring points. Figure 5.6(a) shows the new positions of the intersection points after three smoothing steps with the Laplacian smoothing. It should be noted that the positions of the intersection points at constraint locations remain fixed during the smoothing process. However, it can be seen in Figure 5.6(b) that the smoothing operation modifies the structural geometry, where the green lines represent the reconstructed geometry after smoothing. Moreover, with few smoothing steps, the reconstructed smoothed geometry depicted in Figure 5.6(c) shows little improvement over the non-smoothness of the polygonal geometry.



(a) With smoothing (b) With and without smoothing



(c) Reconstructed geometry With smoothing

Figure 5.6: Zero level set intersection points at $\alpha = 0.70$

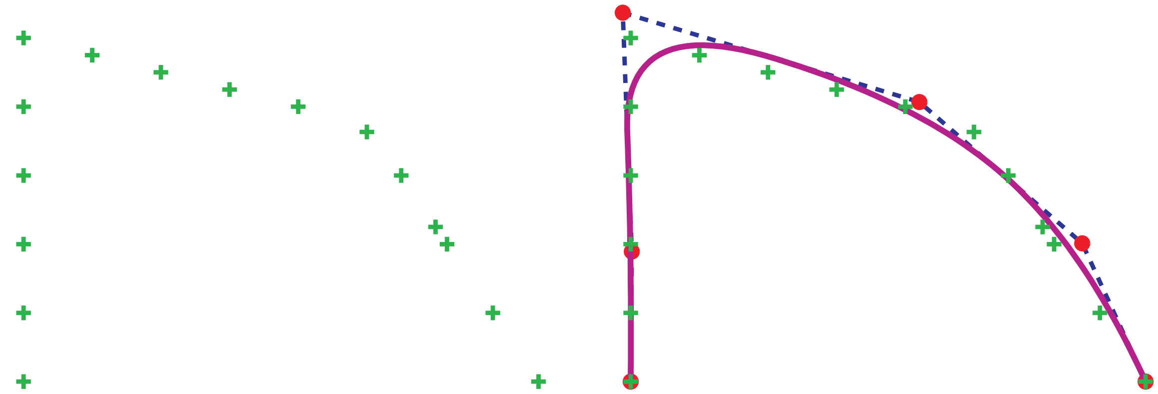
A better solution proposed in this study is to fit a NURBS curve through the zero level set intersection points. In order to proceed, we need to select an optimal number of control points for a given set of data points using the curve fitting procedure explained in Section 4.5. Therefore, three different sets of data points have been considered and tested with various combinations of control points as depicted in Figures 5.7, 5.8 and 5.9, respectively. Figure 5.7 show fittings of NURBS curves through sixteen data points with six, ten and sixteen control points. If the control points are equal to the number of data points as shown in Figure 5.7(d), the resulting curve passes through each data points with unwanted wiggles. As can be seen in Figure 5.7(b) and (c), with fewer control points than the data points result into a smooth curve fittings. However, in some cases fewer control points may result into a smooth curve but would not allow it to pass through maximum number of control points as depicted in Figure 5.8(b). In most cases fewer control points always guarantee a smooth curve fitting, i.e 5.8(d) and 5.9(d). Based on the results of numerous tests conducted the optimal control points used for different sets of data points are based on the data given in Table 5.1.

	External boundary		Internal boundary	
	Data points	Control points	Data points	Control points
1	0 - 24	5	0 - 14	5
2	25 - 59	8	15 - 24	7
3	60 - 89	13	25 - 44	13
4	> 89	20	45 - 79	19
5			> 80	25

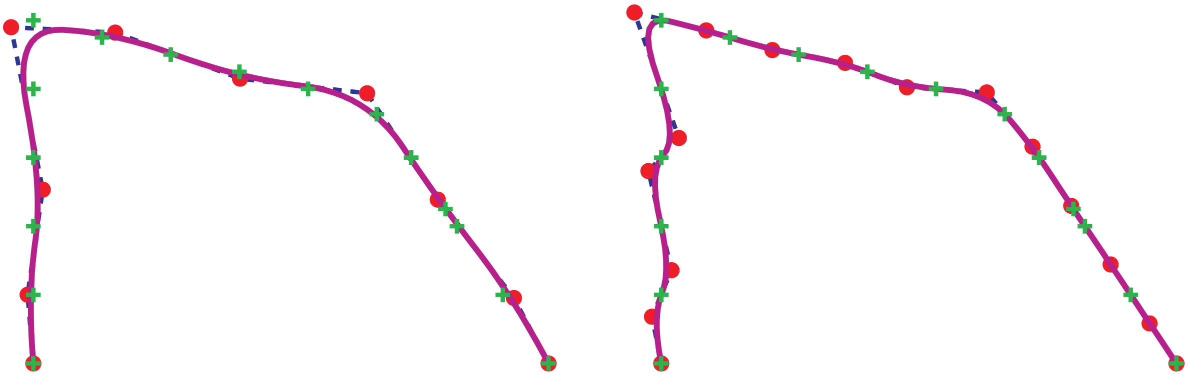
Table 5.1: Selection of control points for a given set of data points

Using the data provided in Table 5.1, the NURBS curves fitted through the zero level set intersection points are shown in Figure 5.10(a) for the modifiable portion of the structural geometry. It is evident from this fitting procedure, that each curve passes exactly through maximum intersection points. Similarly, Figure 5.10(b) and (c) shows the reconstructed geometry with NURBS representation with and without control points, respectively.

The automatic meshing facility within the CA software is used to define elements on each spline, using a setting which is designed to produce peak stresses to approximately

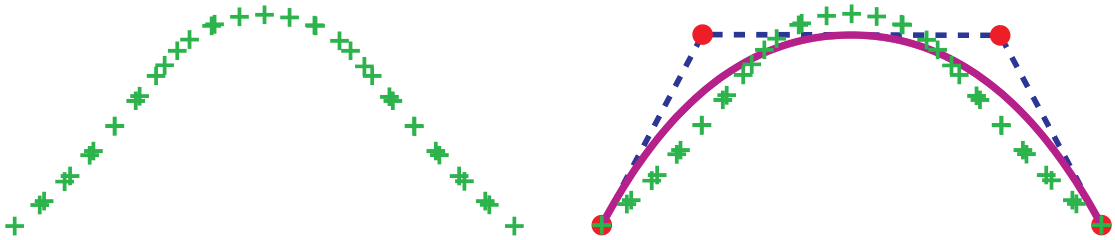


(a) Data points (b) Six control points

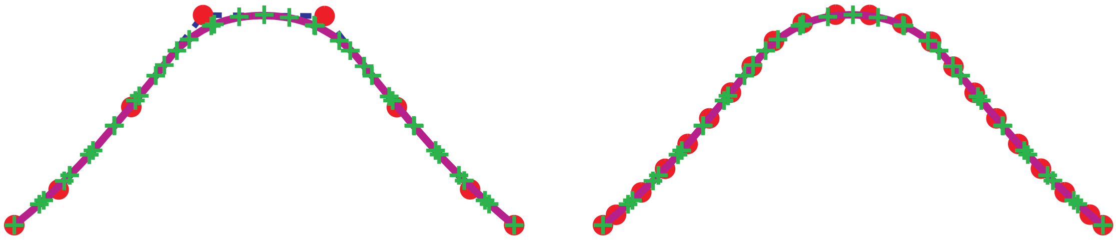


(c) Ten control points (d) Sixteen control points

Figure 5.7: B-spline curve fitting through 16 data points



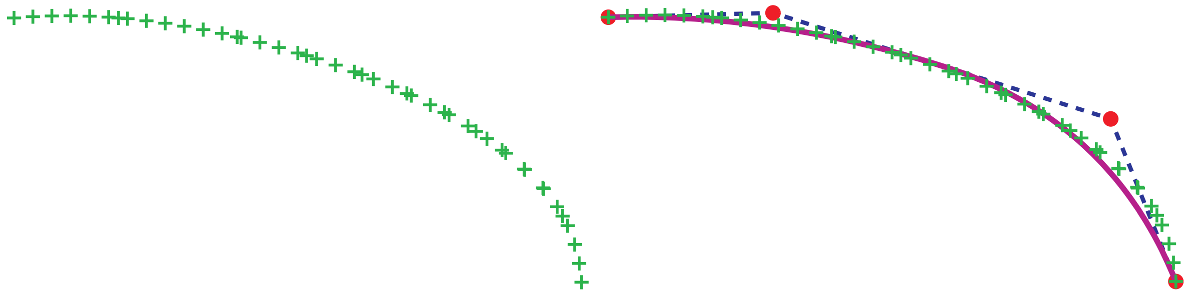
(a) Data points (b) Four control points



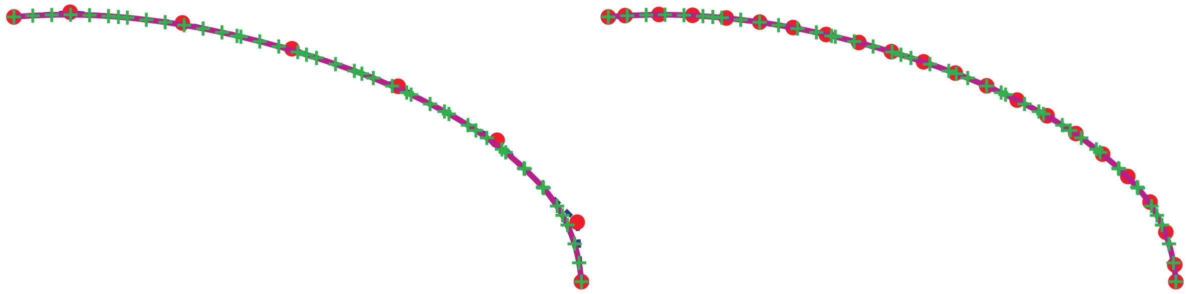
(c) Eight control points (d) Twenty two control points

Figure 5.8: B-spline curve fitting through 37 data points

1% accuracy, either with uniformly distributed boundary element with mid side nodes as shown in Figure 5.10(d) or with grading as required for good BEM meshing practice.



(a) Data points (b) Four control points

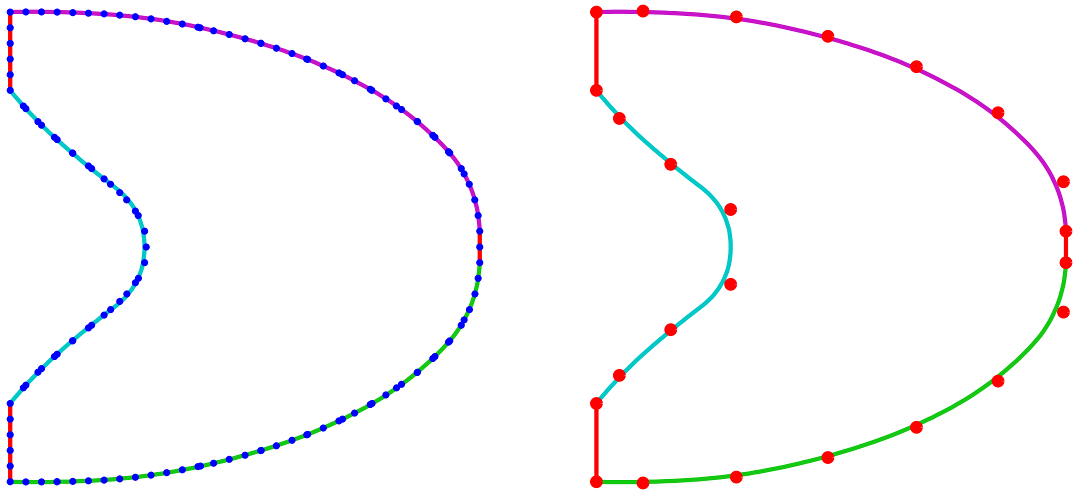


(c) Eight control points (d) Twenty two control points

Figure 5.9: B-spline curve fitting through 45 data points

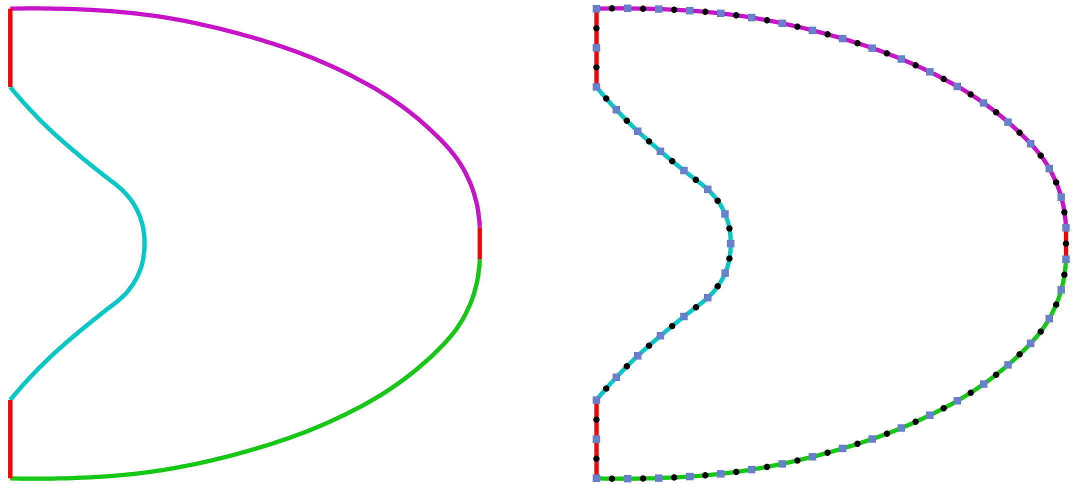
It can be seen that with a NURBS based geometry representation, BE mesh can be carried out independently of the level set grid size. This provides the freedom to use a suitable grid size based on the accuracy and the computational efficiency during the numerical implementation of the proposed method. A linear elastic stress analysis is then automatically initiated. It should be noted that the boundary based discretisation naturally avoids problems of checkerboarding that are well known to require care in FEM based optimisation schemes.

In the above discussion, a simple case of NURBS curve fitting for an external boundary representation without any internal cavities has been discussed in detail. However, in case of multiple internal cavities, the optimisation process can result in complex boundary movements. For explanation purposes, some portion of the cantilever beam of Example-3 (Section 5.4.3) with internal cavities is considered at iteration 85 and depicted in 5.11(a). The complete geometry representation at iteration 85 has been shown in Figure 5.28(d). In the following iterations, the structural geometry evolves and as a result hole merging takes place. It is evident from Figure 5.11(b) that a non smooth geometry can be easily reconstructed from the zero level set intersection points. However, this can result in stress



(a) NURBS through intersection points

(b) NURBS curves with control points



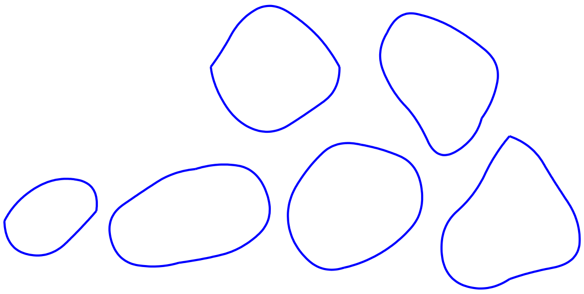
(c) NURBS curves

(d) NURBS with boundary elements and mid side nodes

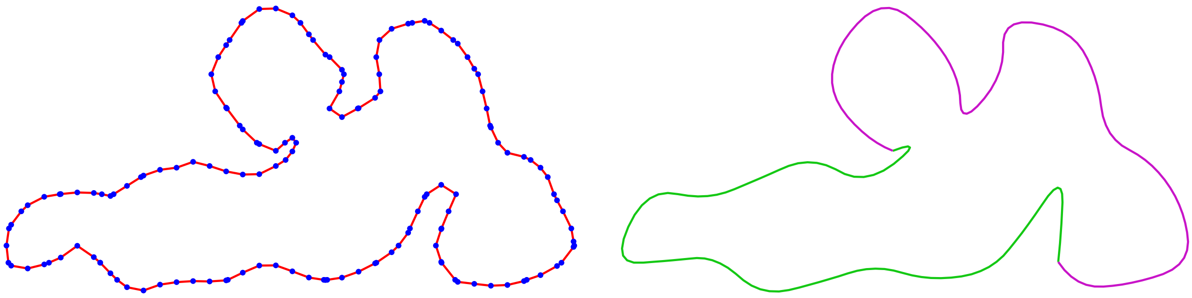
Figure 5.10: NURBS geometry representation at $\alpha = 0.70$

concentration around the sharp corners and can mislead the optimisation process. The NURBS fitting provides a smooth geometry (as depicted in Figure 5.11(c)) and avoids the occurrence of sharp corners. Further, the smooth geometry representation enhances the convergence of the optimisation process.

In the current implementation, each internal cavity has been represented by two NURBS curve. Once a given internal cavity has been traced out by the contour tracing algorithm, the zero level set intersection points are divided into half. A NURBS curve is then fitted to the first half and another one to the second half of the zero level set intersection points. The points at which two NURBS curves meet each other are defined as geometric reference points for each internal cavity. As explained in Section 5.3.3, these points are then used by the contour tracing algorithm to start search for internal cavities.



(a) Portion of the evolving geometry taken from Example-3 at iteration 85



(b) Reconstructed geometry at iteration 102

(c) NURBS curves

Figure 5.11: NURBS geometry representation for internal cavities

5.3.5 Computation of boundary velocities

Based on the proposed optimisation method, σ_V is calculated for each node point after carrying out the BE analysis for the modified geometry. In order to evolve the structural geometry using the LSM, the normal velocity (i.e. F) is assigned to each node point based on the optimisation criterion as discussed in Section 5.2. In the present LSM implementation, negative velocity moves the boundary inward to remove material and positive velocity moves it outward to adds material. Therefore, boundary nodes near the low stressed regions are assigned with negative velocity values, and positive values are assigned to nodes near the high stressed regions.

A relationship similar to that proposed by Sethian and Wiegmann [104] has been developed through numerous numerical experiments. The σ_V at each node point is converted to F, as depicted in Figure 5.12; the intervals shown can be characterised in terms of σ_V, RR, σ_Y, and σ_{Vmax}, as follows:

- $\sigma_V \in [0, \sigma_{t1}] \quad : \sigma_{t1} = 0.5\, RR\, \sigma_{Vmax}\,, F = -1$
- $\sigma_V \in [\sigma_{t1}, \sigma_{t2}] : \sigma_{t2} = 0.9\, RR\, \sigma_{Vmax}\,, F \in [-1, 0]$
- $\sigma_V \in [\sigma_{t2}, \sigma_{t3}] : \sigma_{t3} = 0.95\, \min(\sigma_{Vmax}, \sigma_Y)\,, F = 0$
- $\sigma_V \in [\sigma_{t3}, \sigma_{t4}] : \sigma_{t4} = \min(\sigma_{Vmax}, \sigma_Y)\,, F \in [0, 1]$
- $\sigma_V \in [\sigma_{t4}, \infty) : F = 1$

The LSM requires the velocity to be defined at each level set grid point. In this step only the boundary velocity is calculated; the velocity extension method explained in the following section is later used to extend the boundary velocities to the level set grid points.

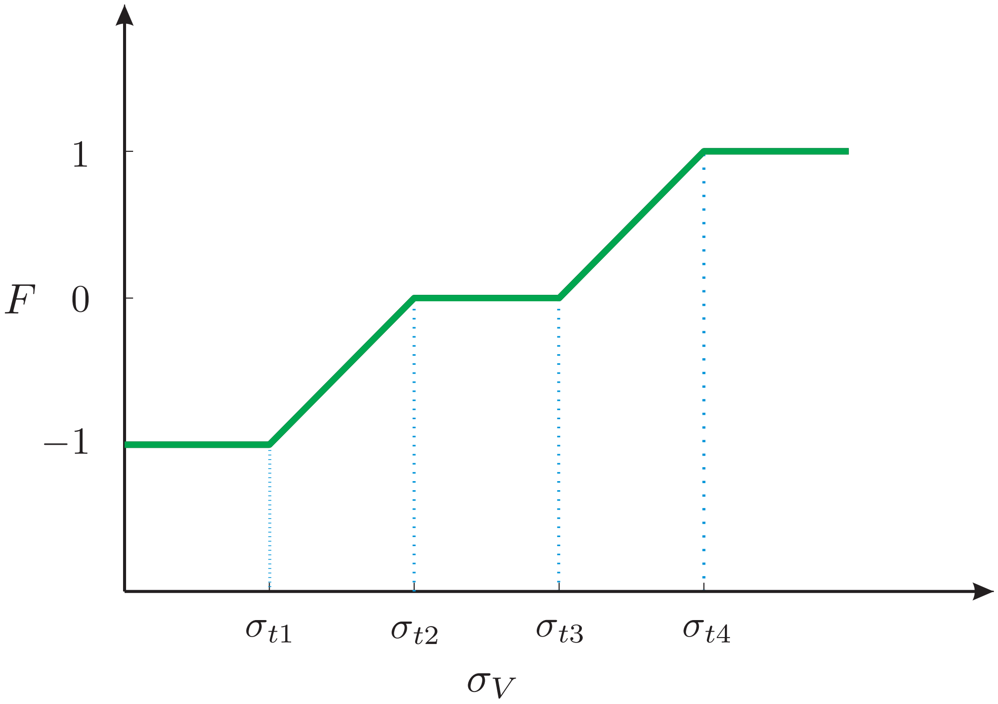


Figure 5.12: Conversion of σ_V to F

5.3.6 Velocity extension

In the present implementation, the boundary velocity is extended to the level set grid using the method of Adalsteinsson and Sethian [5]. This method works on the simultaneous construction of the temporary signed distance function ϕ_t and extension velocity F_{ext} as follow,

$$\nabla\phi_t \,.\, \nabla F_{ext} \,=\, 0 \tag{5.12}$$

The Fast Marching Method [103] is used for the construction of ϕ_t, which solves the following eikonal equation,

$$|\nabla\phi_t| \,=\, 1 \tag{5.13}$$

The level set function is re-initialized by the substitution of the temporary signed distance function (computed during the velocity extension method) for the current level set function. This provides a very fast and accurate way of re-initialization of the level set function [103]. Furthermore, the computational efficiency is achieved by extending the velocity to the grid points in the narrow band [4].

5.3.7 Update of the level set function

After the velocity extension the level set Equation (2.3) is solved with an upwind finite difference approximation. The discretised form of (2.3) for a two dimensional case is

$$\phi_{ij}^{n+1} \,=\, \phi_{ij}^{n} \,-\Delta t[\max(F_{ij},0)\nabla^{+} \,+\min(F_{ij},0)\nabla^{-}] \tag{5.14}$$

where

$$\begin{aligned}\nabla^{+} \;&=\; \max(D_{ij}^{-x}\phi,0)^2 \,+\, \min(D_{ij}^{+x}\phi,0)^2 \,+\\ &\quad\; \max(D_{ij}^{-y}\phi,0)^2 \,+\, \min(D_{ij}^{+y}\phi,0)^2\end{aligned}$$

$$\begin{aligned}\nabla^{-} &= \max(D_{ij}^{+x}\phi, 0)^2 + \min(D_{ij}^{-x}\phi, 0)^2 + \\ &\quad \max(D_{ij}^{+y}\phi, 0)^2 + \min(D_{ij}^{-y}\phi, 0)^2\end{aligned}$$

$$D_{ij}^{+x}\phi = \frac{\phi_{i+1j}^{n} - \phi_{ij}^{n}}{d} \quad , \quad D_{ij}^{-x}\phi = \frac{\phi_{ij}^{n} - \phi_{i-1j}^{n}}{d} \tag{5.15}$$

$$D_{ij}^{+y}\phi = \frac{\phi_{ij+1}^{n} - \phi_{ij}^{n}}{d} \quad , \quad D_{ij}^{-y}\phi = \frac{\phi_{ij}^{n} - \phi_{ij-1}^{n}}{d} \tag{5.16}$$

In the above formulation, ϕ_{ij}^{n+1} represents the value of ϕ at a given point ij and at the current time step $n+1$, ϕ_{ij}^{n} represents the value of ϕ at the same point and at the previous time step n and Δt is the time step size. Similarly, D_{ij}^{+x}, D_{ij}^{-x}, D_{ij}^{+y} and D_{ij}^{-y} are the forward and backward finite difference operators in the x and y directions, respectively.

In the upwind finite difference scheme, the use of forward and backward differences is based on the sign of F at a given grid point ij. If ϕ moves from left to right with a positive F, then based on the method of characteristics [103], the value of ϕ^{n+1} at point ij has influence from the left grid point $i-1j$ and the backward difference operator will be selected for the update of ϕ. Similarly, for a negative F, if ϕ moves from right to left forward difference will be selected for the solution of the level set equation.

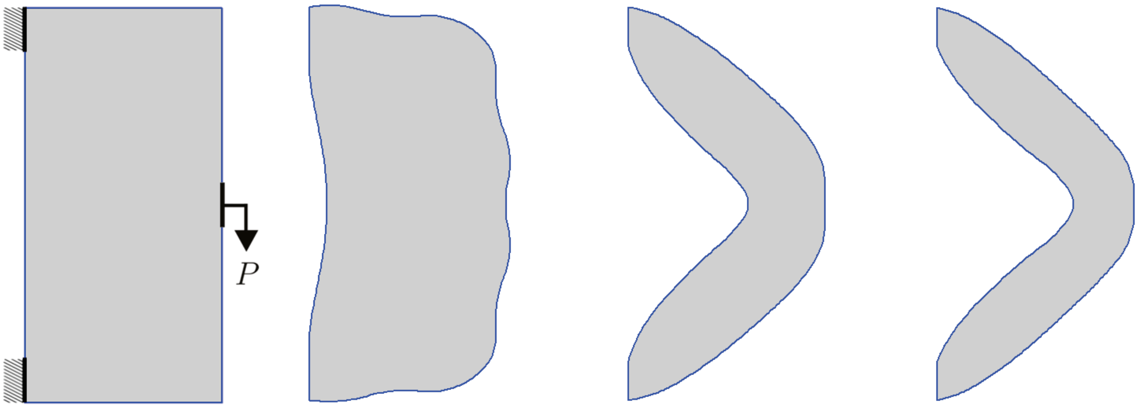
The use of the time step size for the solution of the level set equation is based on the CFL condition. Therefore, a time step size $\Delta t = 0.2d$ is used throughout the numerical implementation. Furthermore, in each optimisation iteration, a single update of the level set equation is carried out.

5.4 Examples

The validity and efficiency of the proposed optimisation method is tested against some benchmarking problems in the field of structural optimisation. The material properties used in these examples are: Poisson's ratio = 0.3, Young's modulus = 210 GPa, Yield stress = 280 MPa. Plane stress conditions are assumed with arbitrary thickness of 1 mm. In all examples the optimisation process starts with $RR = 0.01$. Each example is first solved with $RR_i = 0.01$ and then with $RR_i = 0.05$.

5.4.1 Example-1

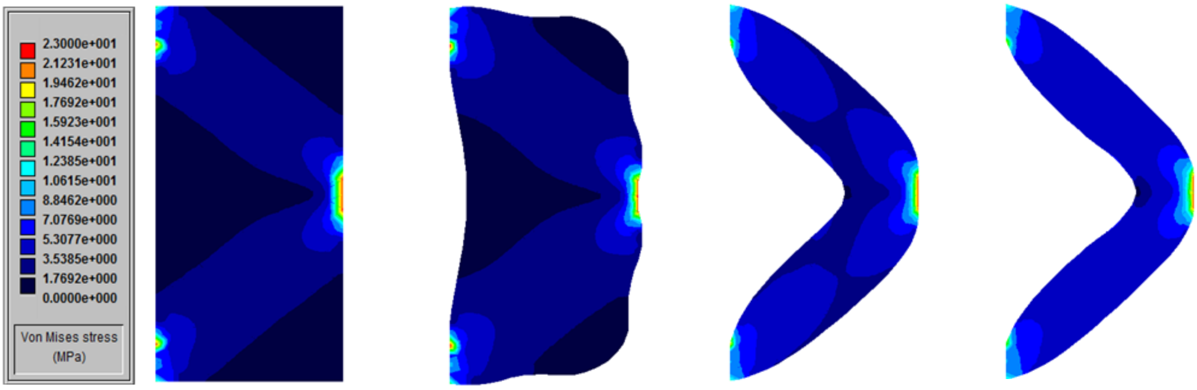
In the first example of this study, a short cantilever with an aspect ratio of 1 : 2 has been used for the solution of the optimisation problem. The initial design displayed in Figure 5.13(a) is constrained at the top and bottom portions of the left edge and loaded with $P = 100N$ at the middle of the right edge. The level set domain is discretised with 25×50 square cells and is further refined at $\alpha = 0.45$. The target volume fraction α used in this example is 0.40.



(a) Iteration 0 ($\alpha = 1$) (b) Iteration 10 ($\alpha = 0.89$) (c) Iteration 150 ($\alpha = 0.47$) (d) Iteration 275 ($\alpha = 0.40$)

Figure 5.13: Evolution of structural geometry for Example-1.

The results displayed in Figure 5.13(b)-(d) shows the evolution of the structural geometry at different stages of the optimisation process with $RR_i = 0.01$. The corresponding von Mises stress contours are depicted in Figure 5.14. It is evident from the stress contour plot of the initial design (i.e. Figure 5.14(a), that there exists considerable amount of low stressed material. During the optimisation process the structural boundary is gradually modified based on the stress values at each node point. This results in the removal of low stressed material through boundary movements only. The structural boundary evolves continuously during the optimisation iterations until the target volume fraction is reached, and the optimisation process terminates. The optimal design obtained at the target volume fraction, i.e. Figure 5.14(d) exhibits uniform stress contours. The optimal design obtained is similar to that available in the literature of the structural optimisation, e.g. [3, 122, 131].



(a) Iteration 0 (b) Iteration 10 (c) Iteration 150 (d) Iteration 275

Figure 5.14: Evolution of von Mises stress contours for Example-1.

During the optimisation process f_U is closely monitored and the results obtained at each iteration are displayed in Figure 5.15. In the initial iterations, low stressed material is successively removed and this resulted in a rapid decreases in f_U. In the following iterations, stresses along the structural boundary become uniform and a slow decrease can be seen up to the end of the optimisation process.

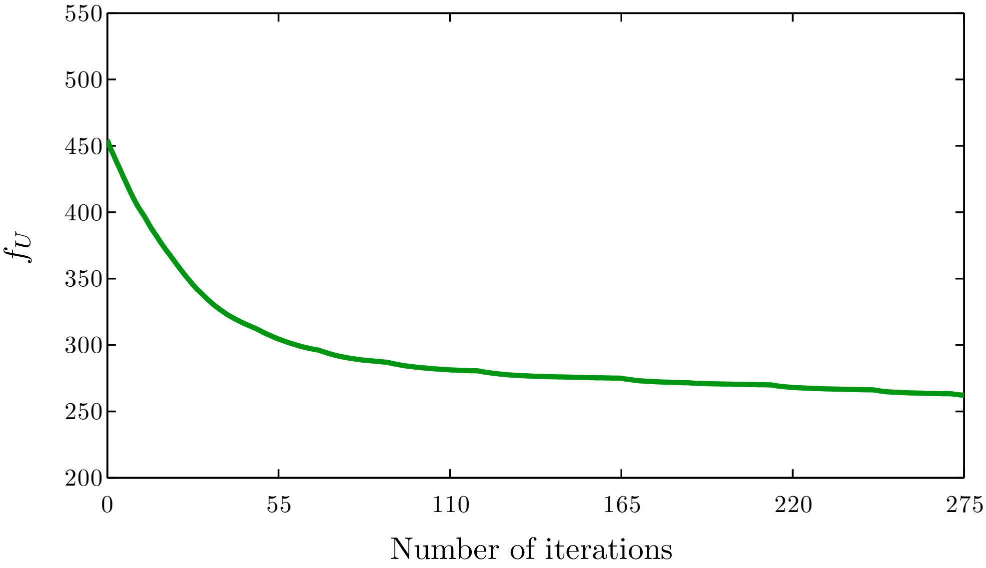
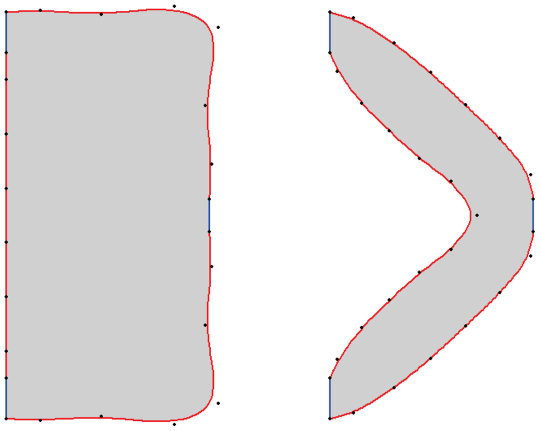


Figure 5.15: Evolution of f_U for Example-1.

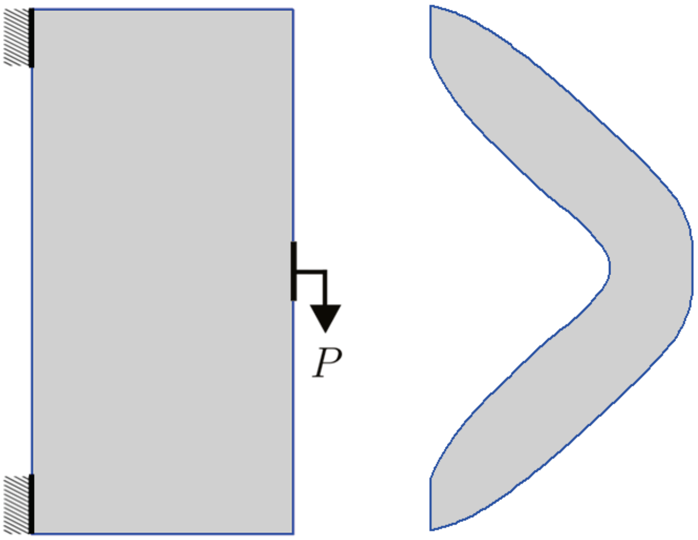
The distributions of NURBS control points in the initial and final designs are displayed in Figures 5.16(a) and (b), respectively. The number of control points used in the final geometry is considerably greater than the initial geometry, which shows excellent local control properties of the NURBS geometry, to maintain a smooth and well defined geometry.



(a) Initial design (b) Optimal design

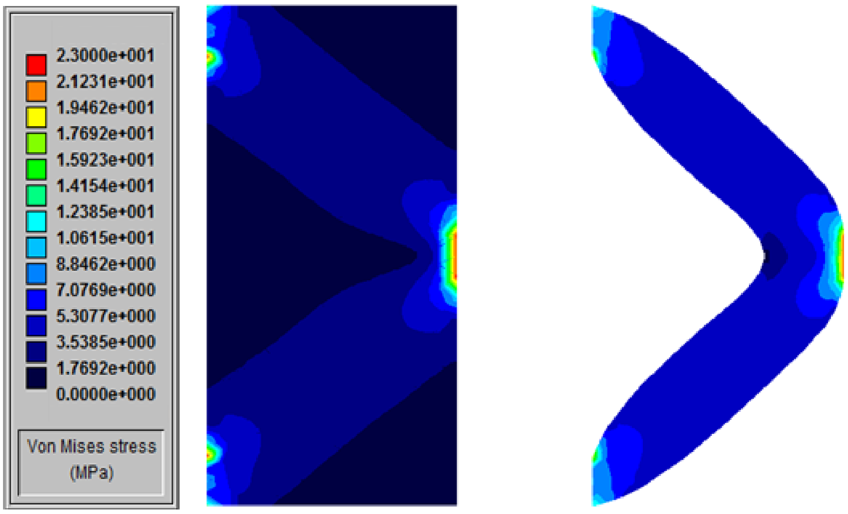
Figure 5.16: NURBS control points distribution for Example-1.

In the second case of this example, the optimisation problem is solved with a different material removal rate, i.e. $RR_i = 0.05$. The optimal design obtained is displayed in Figure 5.17(b). The corresponding von Mises contours and the evolution of f_U are depicted in Figures 5.18 and 5.19, respectively. It is evident from the results obtained, that an increase in the RR_i accelerates the optimisation process and the target volume fraction is achieved in fewer iterations than that of the first case of this example. Moreover, the results obtained in both cases are similar to each other.



(a) Iteration 0 ($\alpha = 1$) (b) Iteration 150 ($\alpha = 0.40$)

Figure 5.17: Initial and optimal designs for Example-1 with $RR_i = 0.05$.



(a) Iteration 0 (b) Iteration 150

Figure 5.18: von Mises stress contours of initial and optimal designs for Example-1 with $RR_i = 0.05$.

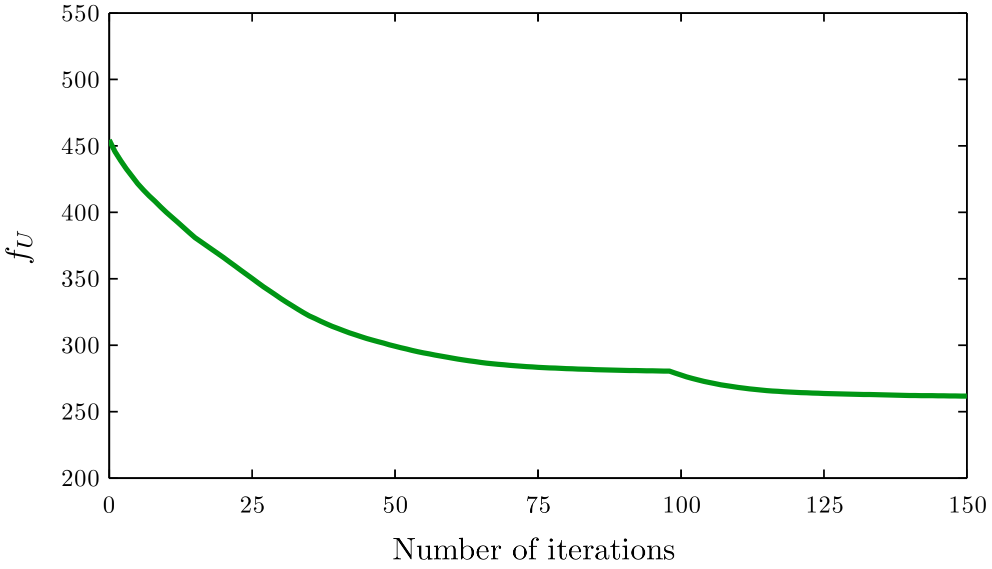
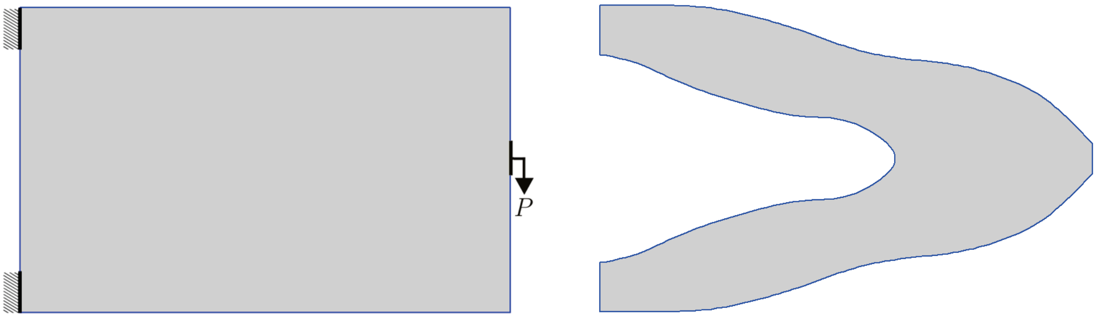


Figure 5.19: Evolution of f_U for Example-1 with $RR_i = 0.05$.

5.4.2 Example-2

The proposed optimisation method is further tested with another benchmark example of a short cantilever beam with an aspect ratio of 1.6:1 as depicted in Figure 5.20(a). The structural geometry is constraint at the top and bottom portions of the left edge and a load $P = 100N$ is applied at the middle of the right edge. The level set domain is discretised with 50×31 square cells.



(a) Iteration 0 ($\alpha = 1.0$)

(b) Iteration 948 ($\alpha = 0.50$)

Figure 5.20: Evolution of structural geometry for Example-2 without pre-existing holes.

Figure 5.20(b) shows the modified geometry at iteration 948 and $\alpha = 0.50$. It can be seen that the optimisation process starts from an initial design completely filled with the material and only shape optimisation is performed modifying the existing structural geometry. The intermediate geometry obtained is quite far from the optimal design of this type of problem in the literature of the structural optimisation. This indicates a deficiency of the direct level set based optimisation approach. In order to overcome this deficiency, the level set based optimisation methods either start from an initial guess design with pre-existing holes or an additional hole insertion mechanism is adopted. These two different strategies are studied in the 2D LSM based optimisation approach. The first strategy is implemented in this chapter and a hole insertion mechanism based optimisation method is presented in Chapter 6.

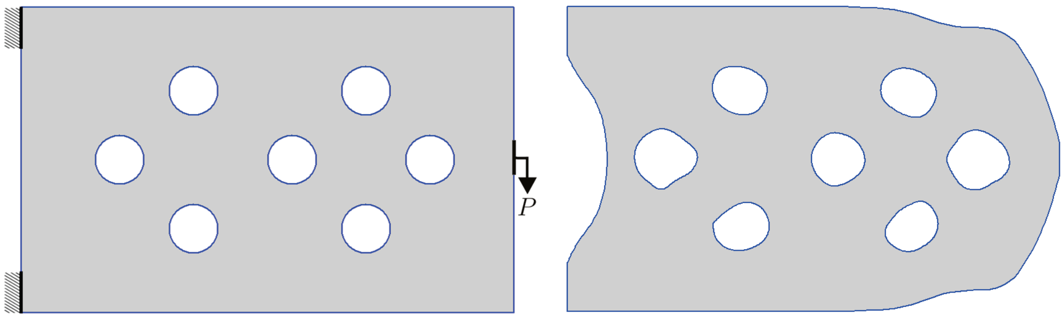
In order to demonstrate the efficiency of the proposed optimisation method, the optimisation problem is solved with pre-existing holes for an initial design domain as shown in Figure 5.20(a). The new initial design domain with applied loads/constraints is depicted in Figure 5.21. During the optimisation process, both external and internal boundaries continuously evolve at the same time. At iteration 35, hole merging takes place with the external boundary, and this is followed by further holes merging near iteration 70. In the subsequent iterations, only shape optimisation takes place and the optimisation process terminates at iteration 164 where the target volume fraction (i.e. $\alpha = 0.30$) is reached.

At some stages of the optimisation process, especially after hole merging, e.g. Figure 5.21(d), slight asymmetry can be observed. In Figure 5.21(c), the central right hand side hole and the adjacent upper and lower holes evolves and comes closer to each other during the optimisation process. This results an increase in the stress values of the neighboring nodes of the three holes. As the boundary evolution is based on the stress levels of the stress velocity relationship, i.e. Figure 5.12. A slight difference in the stress values along the boundary nodes of the central hole (near the upper and lower holes) may result in a slight asymmetric velocity distribution. However, in the subsequent iterations, the proposed method efficiently redistributes material within the design domain, which largely eliminates the asymmetric effects as demonstrated in Figure 5.21 (e). Hence, the asymmetry tends to be related hole merging at intermediate iterations, and the final solutions recover the symmetry.

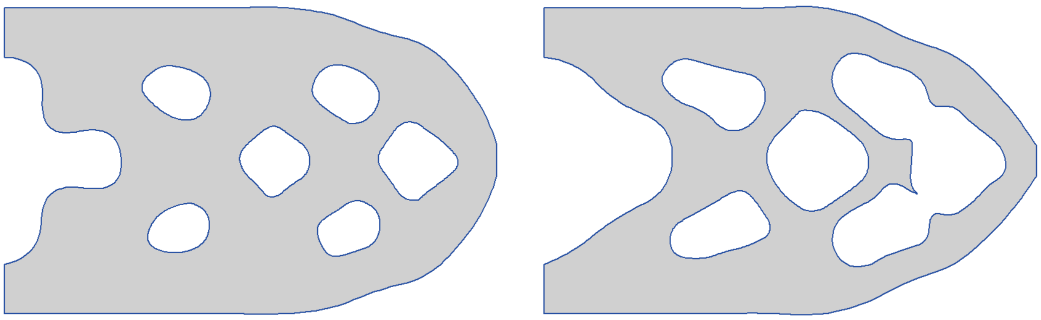
The results obtained demonstrate that both shape and topology optimisation can be efficiently performed with the proposed optimisation method. The optimum obtained in Figure 5.21(f) is in close agreement to that available in the literature, e.g. [3, 19, 21, 53, 71, 98, 136]. In comparison with 5.20(b), the optimisation problem solved in this case rapidly converges with the modified guessed topology and this validate the proposed optimisation method.

Figure 5.22 displays the von Mises stress contours of the initial and optimal designs. The stress contours of the optimal design indicate that the proposed optimisation method efficiently removed the low stressed material and effectively re-distributed it within the design domain. The optimal design obtained is approaching a fully stressed design.

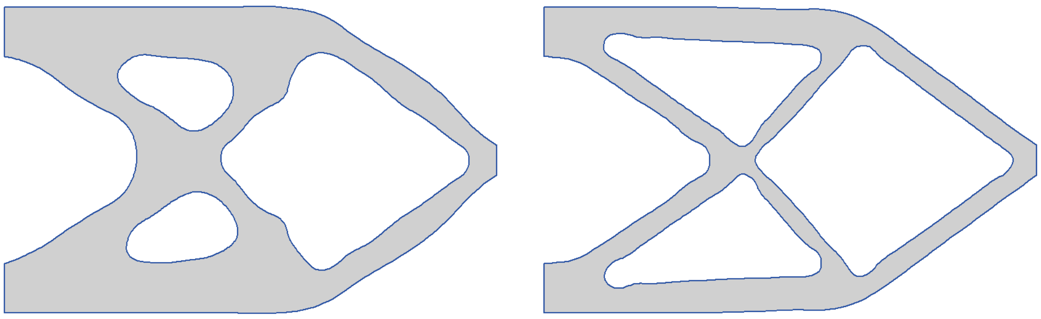
Figure 5.23 shows the evolution of fU during the optimisation process. The specific



(a) Iteration 0 ($\alpha = 0.92$) (b) Iteration 25 ($\alpha = 0.78$)



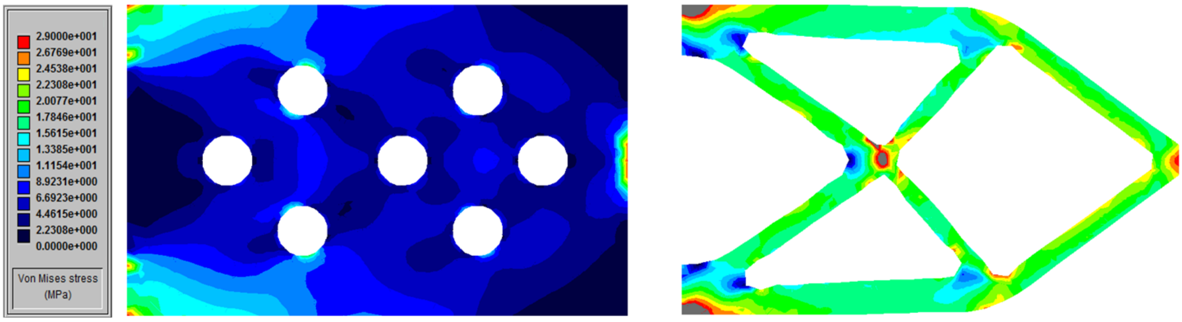
(c) Iteration 35 ($\alpha = 0.70$) (d) Iteration 67 ($\alpha = 0.52$)



(e) Iteration 75 ($\alpha = 0.44$) (f) Iteration 164 ($\alpha = 0.30$)

Figure 5.21: Evolution of structural geometry for Example-2.

strain energy decreases and the geometry evolve gradually as material is removed from the design domain. A high peak can be observed around iteration 67, caused by a significant change of topology resulting from the elimination of one or more bars in one iteration (related to hole merging), as can be seen from Figures 5.23(c)-(d). The occurrence of high peak in the solution of a similar problem has also been observed in a BESO based approach presented in [48], and is therefore not unique to our approach. The effect of this high peak dies out quickly in the proceeding iterations and slowly decreases fU until the target volume fraction is reached.



(a) Initial design (b) Optimal design

Figure 5.22: von Mises stress contours of initial and optimal designs for Example-2.

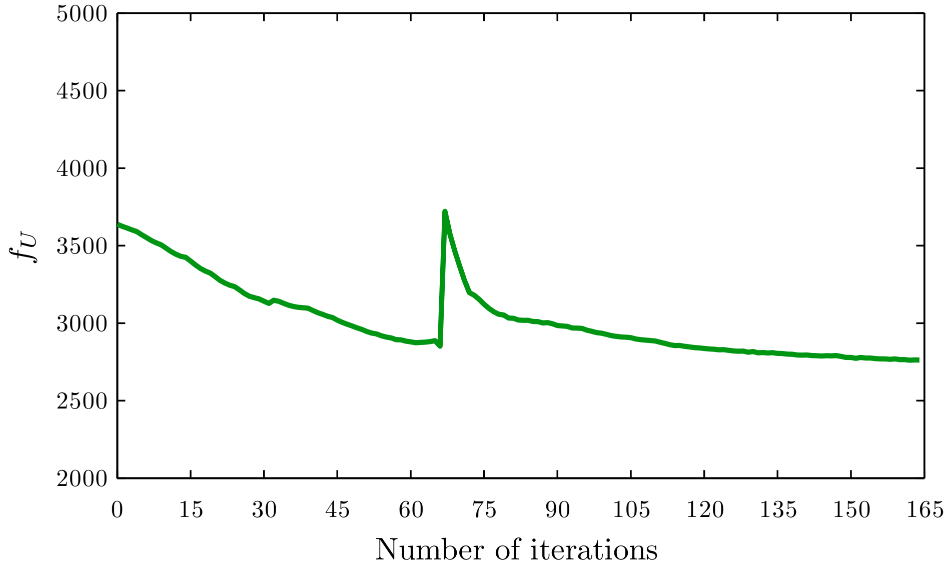


Figure 5.23: Evolution of f_U for Example-2.

The NURBS control points distribution for the initial and optimal configuration is depicted in 5.24. A well defined control points distribution in the optimal design validates the selection of parameters used for the NURBS curve fitting through the zero level set intersection points. However, the control points slightly oscillate around the corner of the holes (Figure 5.24(b)), this effect has also been explained in Section 5.3.4. In Figure 5.7(c), the curve passes through maximum control points and as a result slightly bulges out around the sharp corner. As in the current implementation, the NURBS curve fitting is based on the unit weight assigned to each control point (see Section 4.5), which may not allow a better control around the sharp corners. However, for large weight values the spline will almost go through the control points. This would require the selection of different weights for each control point of a single NURBS curve, which may lead to the solution of a nonlinear set of equations for the NURBS fitting.

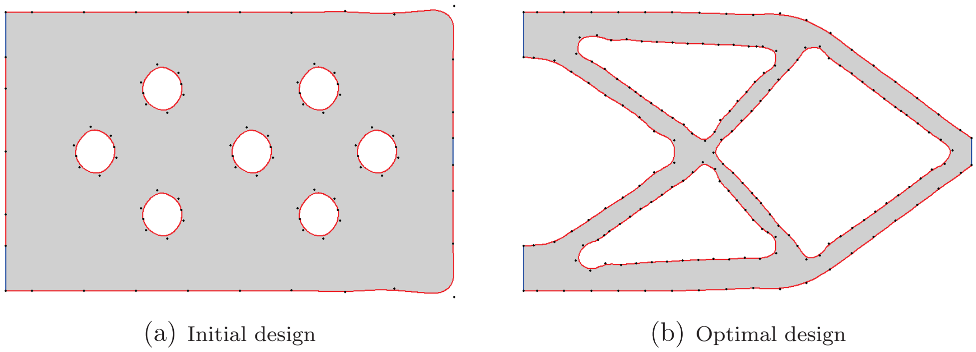
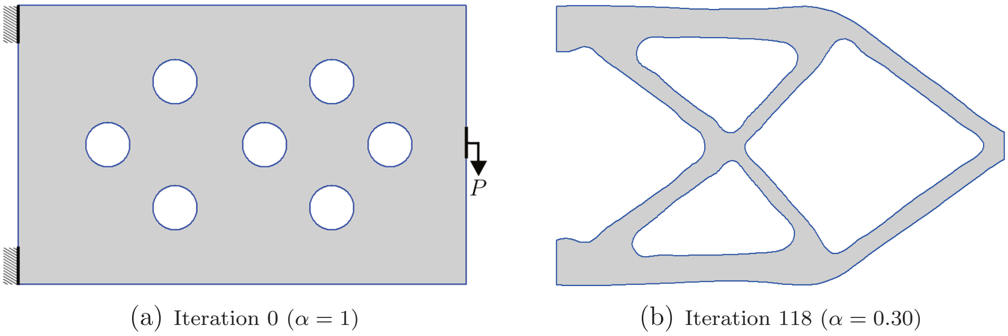

(a) Initial design (b) Optimal design

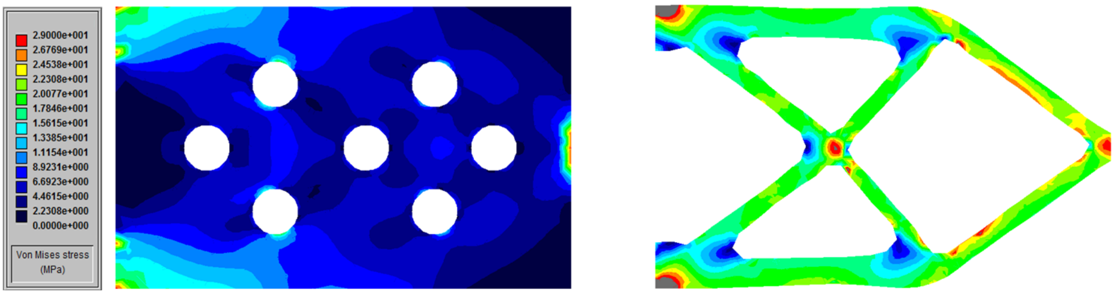
Figure 5.24: NURBS control points distribution for Example-2.

Similar to the previous example, the optimisation problem is solved with a slightly larger removal ratio, i.e $RR_i = 0.05$. It can be seen in Figure 5.25 that with an increased removal ratio, the target volume fraction is achieved in fewer iterations than that obtained in the previous case. The optimal topologies obtained in both cases are similar to each other and exhibits identical von Mises stress contours. Comparison of the evolution of f_U in both cases indicates that the optimisation process terminates with a slightly higher value in the later case as a result of the increased removal rate.



(a) Iteration 0 ($\alpha = 1$) (b) Iteration 118 ($\alpha = 0.30$)

Figure 5.25: Initial and optimal designs for Example-2 with $RR_i = 0.05$.



(a) Iteration 0

(b) Iteration 118

Figure 5.26: von Mises stress contours of initial and optimal designs for Example-2 with $RR_i = 0.05$.

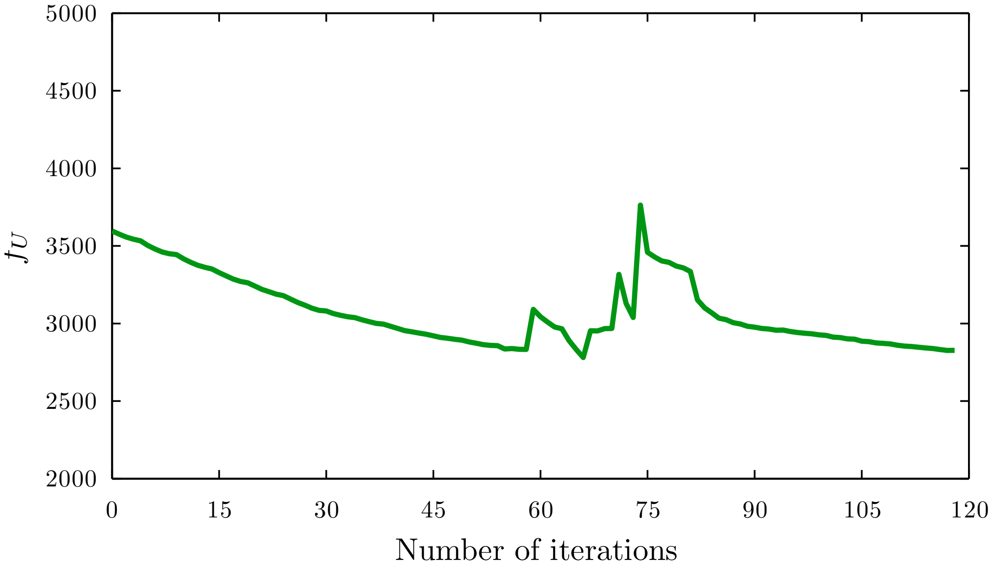


Figure 5.27: Evolution of f_U for Example-2 with $RR_i = 0.05$.

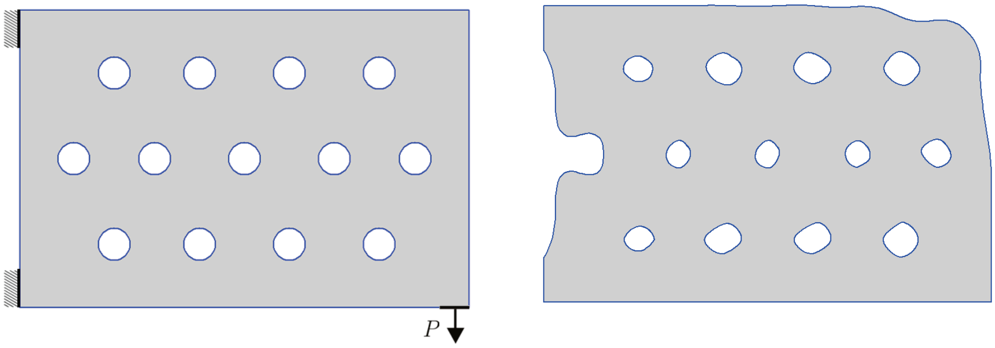
5.4.3 Example-3

The last example considered in this study is a cantilever beam with an aspect ratio of 1.5:1. The initial design domain, loading and boundary conditions are depicted in Figure 5.28(a). The optimisation process starts with an initial guessed design having pre-existing holes. The level set domain is discretised with 50 × 33 square cells and is further refined to 100 × 66 square cells at $\alpha = 0.35$. The optimisation problem is solved for a target volume fraction, i.e $\alpha = 0.30$.

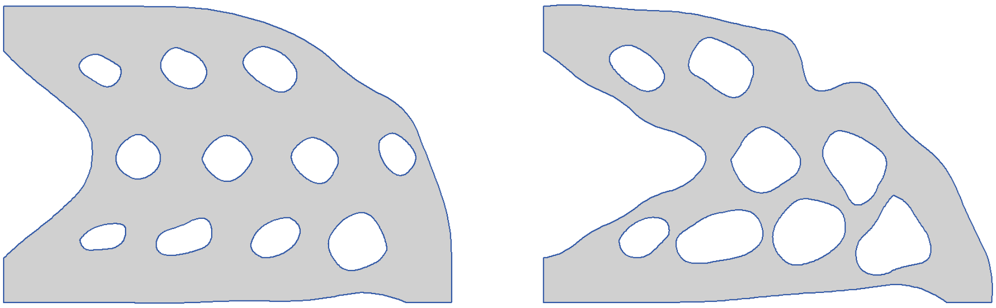
The evolution history of the structural geometry at different stages of the optimisation process is displayed in Figure 5.28(b)-(f). Structural geometry evolves during the optimisation process through boundary movements and results in holes merging with each other and with the boundary. In the current implementation, the optimisation process terminates at the target volume fraction. However, the proposed method can be improved with the addition of a constant volume constraint, which would provide optimal geometry with better smooth boundary description. The optimal design obtained closely matches to that available in the literature of this type of problem, e.g. [19, 66, 71, 136, 140]. The von Mises stress contours of the initial and optimal designs are depicted in Figure 5.22 and clearly demonstrate that the optimal design is approaching a fully stressed design.

The evolution of f_U recorded during the optimisation process is depicted in Figure 5.30. A similar trend can be seen as observed in the previous example. The high peaks recorded around iterations 80, 100 and 112 are mainly related to merging of the holes with the structural boundary. In the following iterations the optimisation method efficiently redistributes material and this results in a slow decrease in f_U. Figure 5.31 shows well-defined NURBS control points in the initial and final designs.

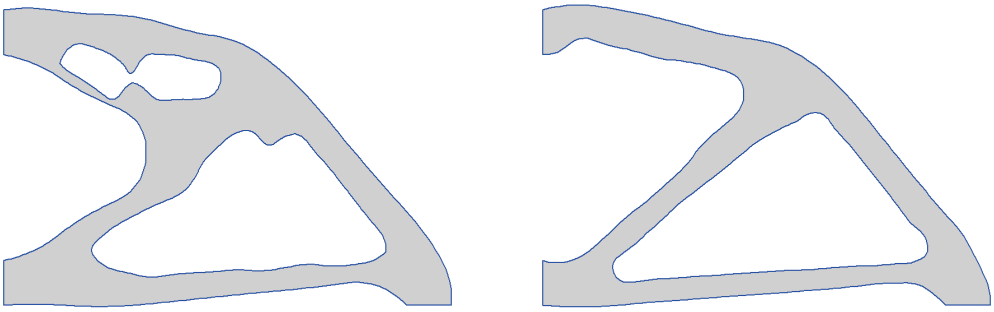
Figure 5.32 displays the initial and the optimal designs obtained with $RR_i = 0.05$. Comparison of the optimal design with different RR_i shows that the target volume fraction has been achieved in fewer iterations with an increase in the removal rate. However, the optimal topologies are slightly different. The von Mises stress contour plots (i.e. Figure 5.29(b) and 5.33)(b) demonstrate that the structural members are equally stressed in



(a) Iteration 0 ($\alpha = 0.93$) (b) Iteration 12 ($\alpha = 0.89$)



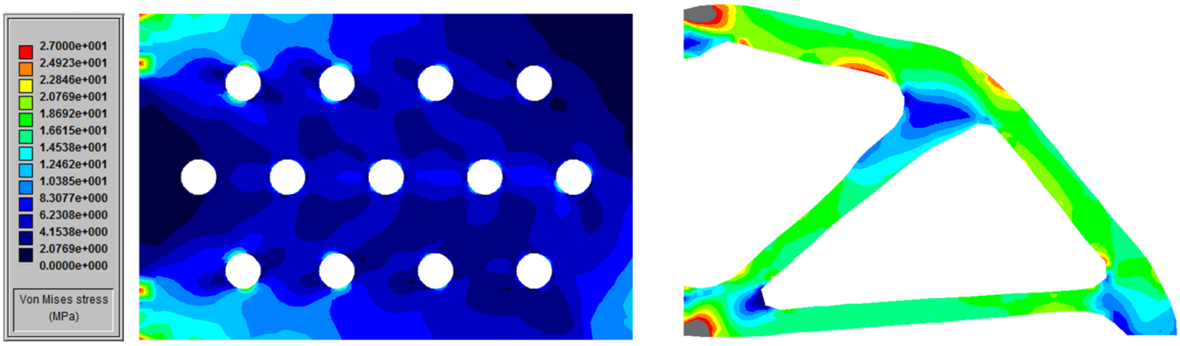
(c) Iteration 52 ($\alpha = 0.67$) (d) Iteration 85 ($\alpha = 0.47$)



(e) Iteration 111 ($\alpha = 0.35$) (f) Iteration 140 ($\alpha = 0.30$)

Figure 5.28: Evolution of structural geometry for Example-3.

both cases. In addition, the optimisation process terminates at approximately the same f_U in both cases. However, the high peaks observed in 5.34 from iteration 50 to 70 is mainly related to hole merging during the optimisation process. Further, the optimisation process terminates at the target volume fraction.



(a) Initial design (b) Optimal design

Figure 5.29: von Mises stress contours of initial and optimal designs for Example-3.

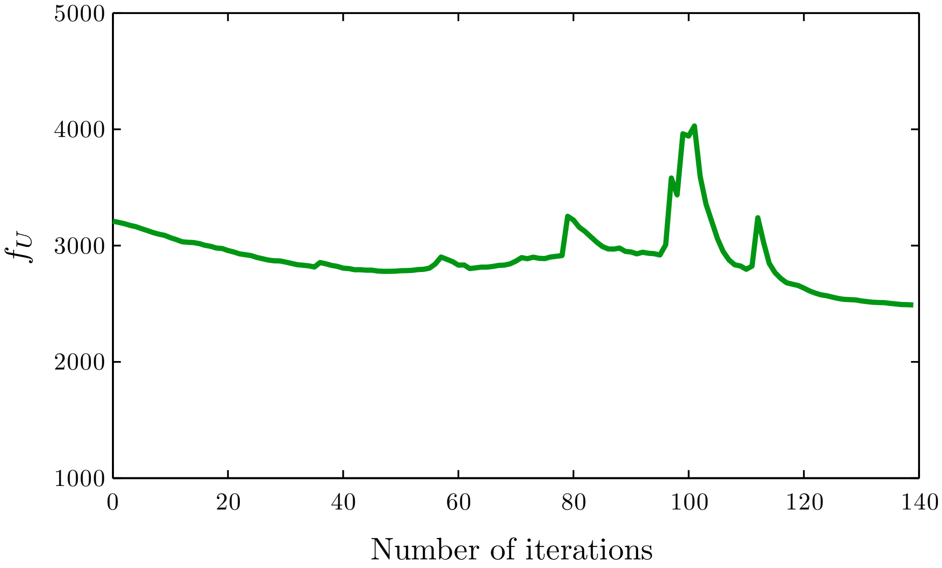
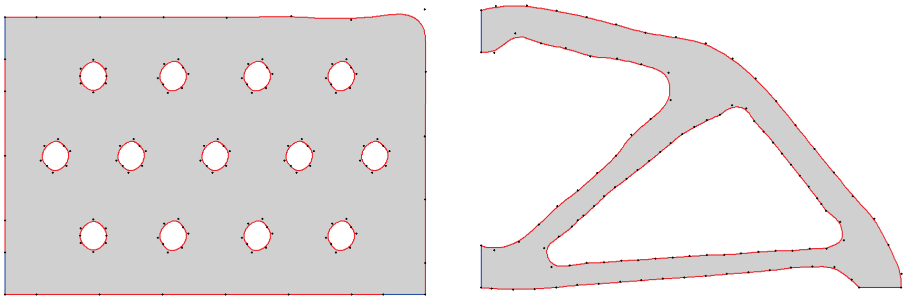
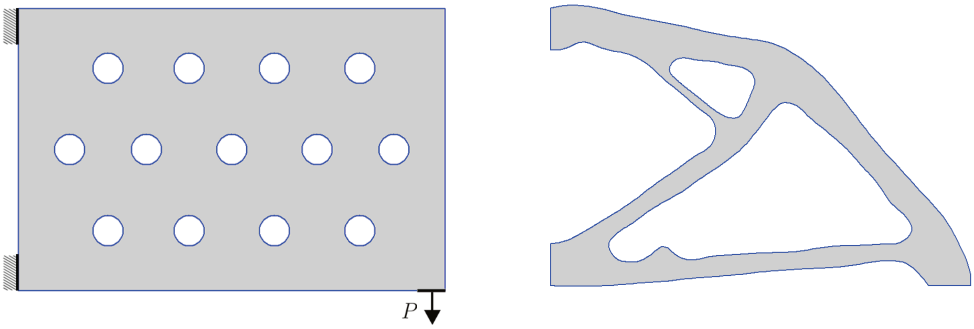


Figure 5.30: Evolution of the f_U for Example-3.



(a) Initial design (b) Optimal design

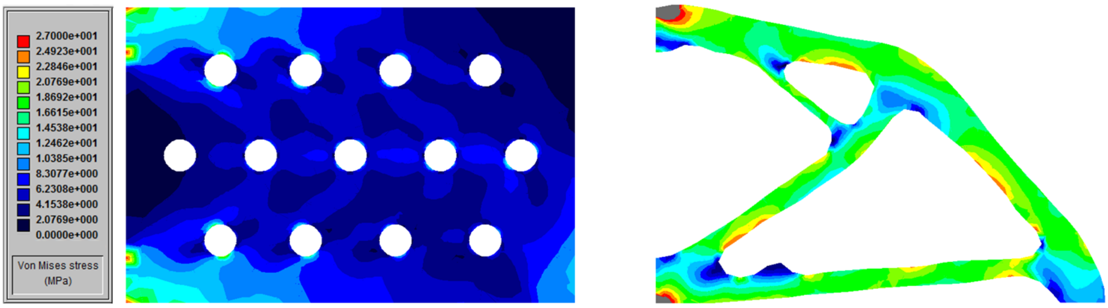
Figure 5.31: NURBS control points distribution for Example-3.



(a) Iteration 0 ($\alpha = 1$)

(b) Iteration 88 ($\alpha = 0.30$)

Figure 5.32: Evolution of f_U for Example-1.



(a) Iteration 0

(b) Iteration 88

Figure 5.33: von Mises stress contours of initial and optimal designs for Example-3 with $RR_i = 0.05$.

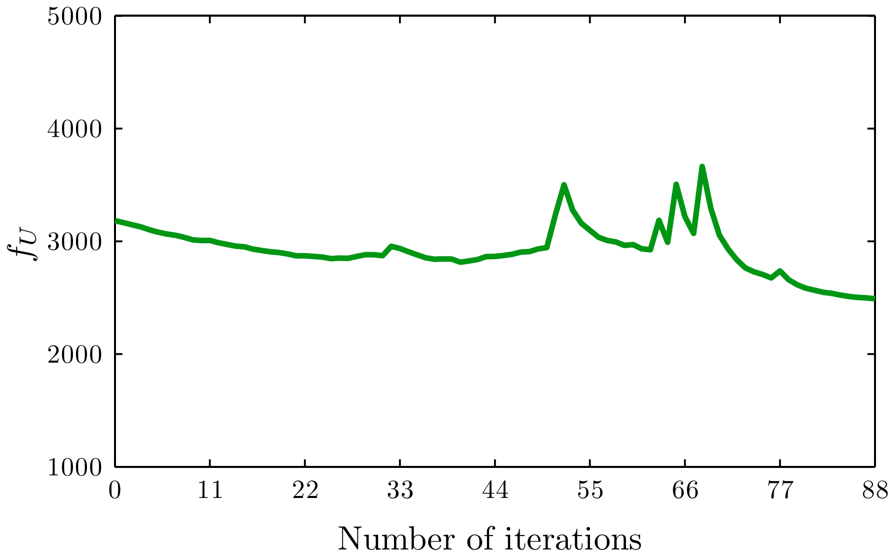


Figure 5.34: Evolution of f_U for Example-3 with $RR_i = 0.05$.

5.5 Conclusions

A bi-directional evolutionary structural optimisation scheme has been presented, which uses the LSM to control the evolving structural geometry. At each optimisation iteration, NURBS are fitted to a set of points lying on the zero level set contour, and these are automatically meshed with boundary elements. The von Mises stress results from the BEM linear elastic simulation are mapped to a distribution of the level set velocity function, which is then used to update the design geometry in preparation for the next iteration. Three different benchmark problems are solved with the proposed optimisation method and each one is further tested with a different material removal rate. The optimal design topologies and geometries obtained from the proposed method closely resemble the optima published in the literature of structural optimisation. The unique combination of BEM, evolutionary approach, LSM and NURBS provides an optimisation technique with fast and accurate structural analysis and with the added advantage of a smooth geometry both from the structural analysis as well as the manufacturing point of view.

Chapter 6

LSM and BEM based structural optimisation with a hole insertion mechanism

6.1 Overview

The two-dimensional level set based optimisation method presented in Section 5.3 is dependent on an initial guessed design with pre-existing holes. In two-dimensions, the solution of the HJ type level set equation does not allow hole nucleation during the optimisation process. However, as demonstrated through the numerical examples presented in Section 5.4, the use of LSM efficiently handles automatic hole merging and is capable of successfully optimising a given structure with pre-existing holes. The BEM and LSM based optimisation methods presented to date, i.e. [3, 136] are also dependent on an initial guessed design with pre-existing holes. This Chapter presents a new BEM and LSM based optimisation method which automatically inserts holes during the optimisation process using a von Mises stress based hole insertion mechanism. With the use of a hole insertion mechanism, both shape and topology optimisation can be performed at the same time, and this significantly enhances the capabilities of the initially proposed BEM and LSM based optimisation approach. Further, the optimisation problems can be solved with the proposed method for an initial design with or without pre-existing holes. The optimal designs obtained with the proposed method closely resemble those available in the literature for a range of benchmark examples in the field of topology optimisation.

6.2 Optimisation algorithm

The capabilities of the proposed level set based optimisation algorithm presented in Section 5.3 are further enhanced with the addition of a hole insertion mechanism. The modified algorithm is depicted in Figure 6.1 and summarised as follows:

1. Define structural geometry with applied loads and constraints.
2. Initialize the level set grid with signed distance function to embed the structural geometry into a implicit function.
3. Trace the zero level set contours and convert them into NURBS.
4. Carry out boundary element analysis.
5. Insert holes in the low stressed areas of the structure using the hole insertion criterion.
6. Compute velocity at each node point of the structural boundary using BE analysis results.
7. Extend boundary velocities to level set grid points in the narrow band.
8. Solve Equation (2.3) to update the level set function.
9. Repeat the above procedure from step 3, until the stopping criterion is satisfied.

Most of the steps followed in the above modified algorithm are similar to that presented in Figure 5.2. The detailed explanation of these steps can be found in Section 5.3. Therefore, only the newly introduced hole insertion mechanism and its implementation details are covered in the following sections.

6.2.1 Hole insertion

In a topology optimisation process, topological changes take place through the creation of cavities or holes within the design domain. As explained in Section 5.2, the von Mises

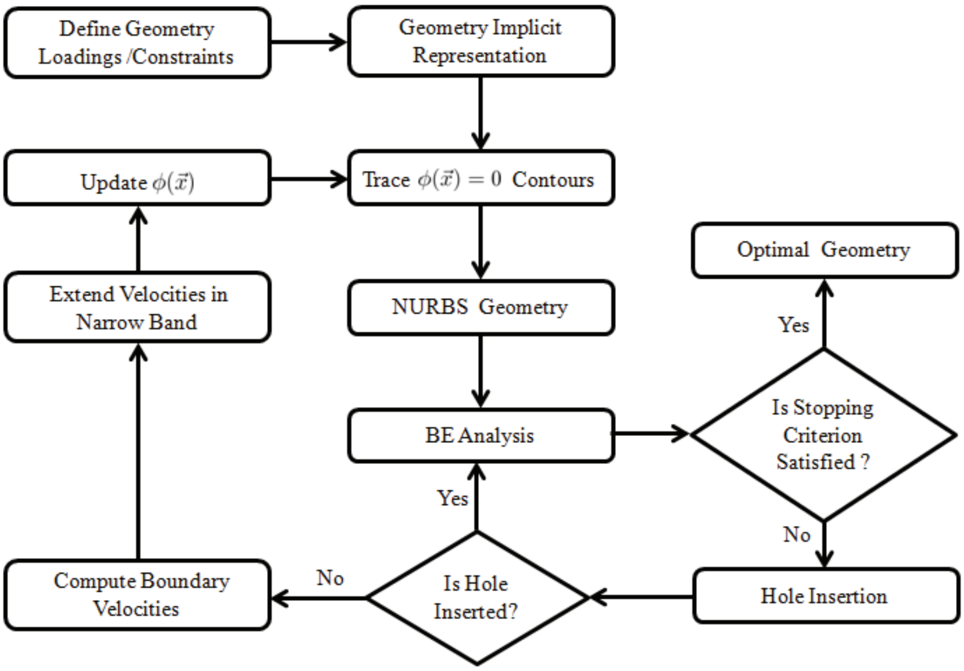


Figure 6.1: Optimisation flow chart

stress is used to evolve the structural boundary during the optimisation iterations. In the present implementation, the same criterion is used to remove inefficient material through hole insertion within the structure experiencing low stress [21].

In a BE analysis stresses within the structure are calculated at internal points. Although, these points are used to provide information for displaying stress contours, the von Mises stress at these points is also used to inform a criterion for hole insertion in the low stress regions in the structure. The CA software generates these points automatically using the following algorithm.

- Rings of internal points are defined around holes.
- Arcs of internal points are defined around fillets and re-entry corners.
- Lines of internal points are defined along possible neutral axis locations in bending.
- Remaining internal points, giving a total number equal to 1.5 × number of nodes, are placed randomly.
- A triangulation is generated from the 2D set of points (nodes and internal points).

- Laplacian smoothing is applied to the triangulation.
- Internal points too close to the boundary are repositioned.

The algorithm is designed to give smooth stress contours using a reasonable number of internal points (for computational efficiency) and has been refined over many years' usage in academia and industry. The ability to produce smooth contours indicates that a sufficiently detailed description of the stress field is available for the optimization process. It should be noted that the procedure for defining the internal point locations includes some randomness.

The implementation details of the hole insertion mechanism are displayed in Figure 6.2 and summarised as follows,

1. In a given set of internal points as depicted in Figure 6.2(a) identify internal points satisfying the following equation

   $$\sigma_i \leq RR\sigma_{t1} \tag{6.1}$$

   where σ_i is the von Mises stress at a given internal point and σ_{t1} is related to the stress range shown in Figure 5.12.

2. Sort the internal points identified in step 1 and depicted with ■ in Figure 6.2(b) in ascending von Mises Stress order.

3. The first internal point, i.e. the least stressed point from the above step depicted with ▲ in Figure 6.2(c), is used as a centre for the new hole.

4. Internal points satisfying a threshold stress level (related to Equation (6.1)) around the central point from step 2 are used to construct a convex polygon shown in Figure 6.2(d).

5. The vertices of the convex polygon are taken as control points to generate two NURBS curves as shown in Figure 6.2(e).

6. The end points of the two NURBS curves are defined as geometric reference points for the new hole (i.e. Figure 6.2(f)).

7. The above steps are repeated until there are no more internal points selected in step 1.

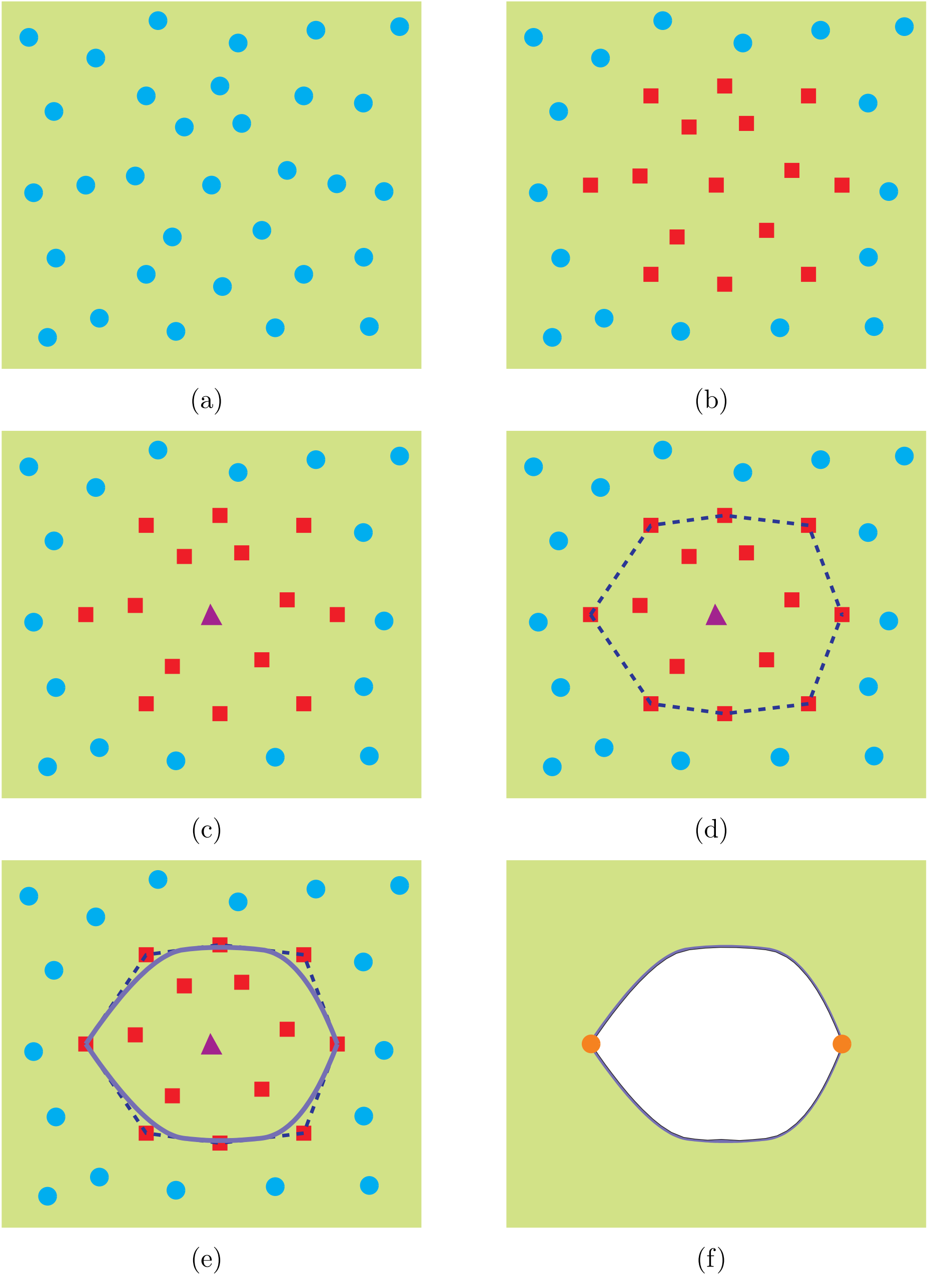


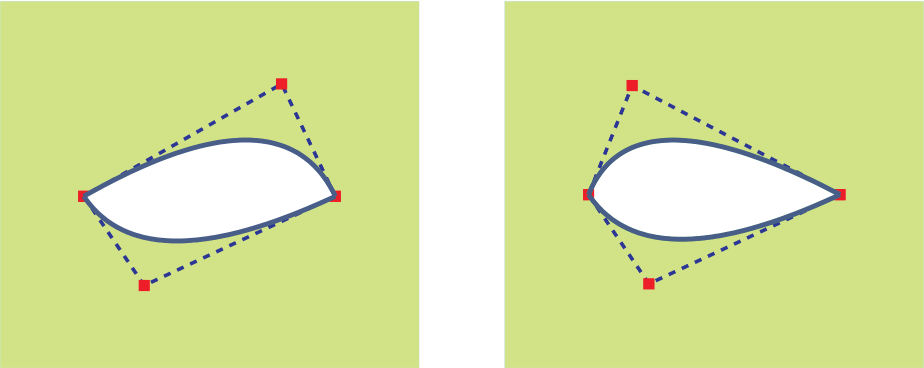
Figure 6.2: Creation of holes from internal points (● = internal points, ▲ = low stressed central internal point, ■ = low stressed internal points)

In situations when the number of internal points around the central point is less than five, then no hole insertion takes place and the next internal point is used to repeat the above steps for hole insertion around it. The hole insertion changes the structural geometry, which is re-analyzed with BEM for the new stress distribution. In addition, the level set grid is re-initialised after each hole insertion.

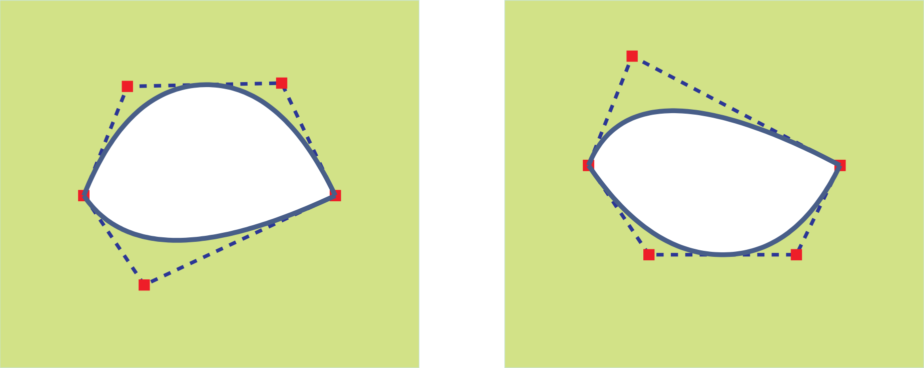
Figure 6.3 displays the variation of shape and size of the inserted hole with four, five, six, eight and twelve control points, respectively. An acceptable shape and size of hole can be inserted with four control points as depicted in Figure 6.3(a)-(b). In these cases, each NURBS curve have at least three control points, which is normally required for a standard description of the curve. However, the hole inserted in Figure 6.3(b) with four control points has sharp corner and may result in artificial stress concentration. On the other hand hole inserted with five control points, i.e. 6.3(c) and (d), respectively, have lower tendency towards the stress concentration effects. Further, it is evident from Figure 6.3(e)-(g), that an increase in the number of control points provides an optimum shape of the inserted hole.

Finally, the implementation of the proposed hole insertion mechanism addresses the three main requirements for hole insertion in a level set based optimisation method.

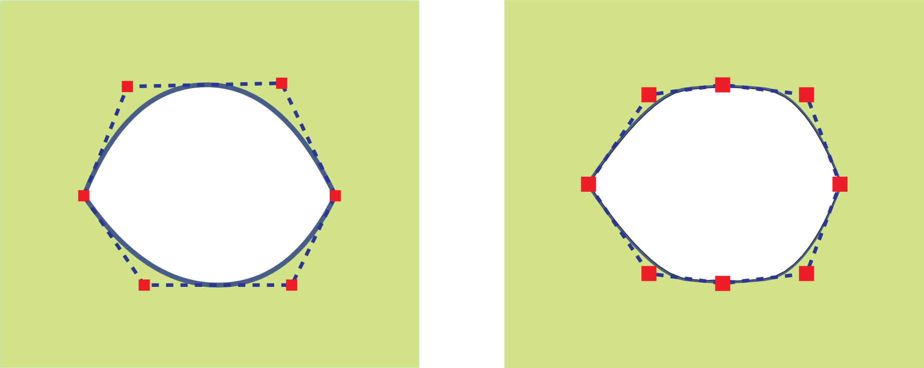
1. Where to insert a hole?
    - around the low stressed internal points satisfying Equation (6.1)
2. When to insert a hole?
    - when the number of internal points around the central point is greater than four
3. Shape of the inserted hole?
    - contour lines of the von Mises stress around the central point based on Equation (6.1).



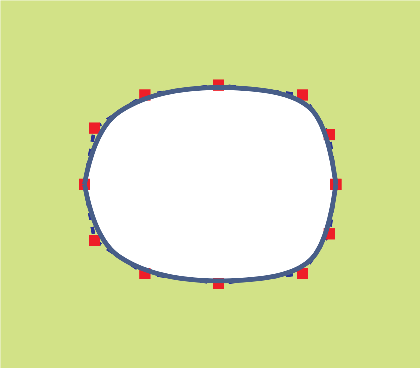
(a) Hole with four control points (b) Hole with four control points



(c) Hole with five control points (d) Hole with five control points



(e) Hole with six control points (f) Hole with eight control points



(g) Hole with twelve control points

Figure 6.3: Shape and size of the hole with four and five internal points

6.3 Examples

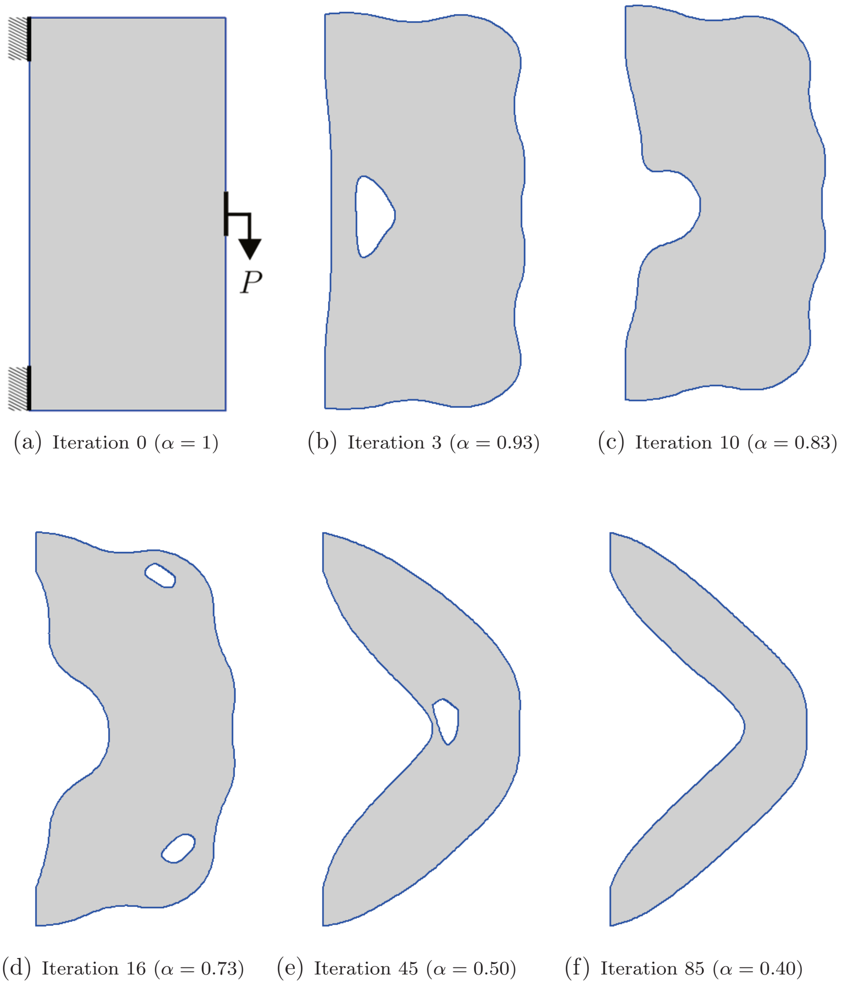
The validity and efficiency of the proposed optimisation method is tested against some benchmarking problems in the field of structural optimisation. The material properties used in these examples are: Poisson's ratio = 0.3, Young's modulus = 210 GPa, Yield stress = 280 MPa. Plane stress conditions are assumed with arbitrary thickness of 1 mm. In all examples the optimisation process starts with $RR = 0.01$ and $RR_i = 0.05$, unless otherwise stated.

6.3.1 Example-1

The proposed method is tested with the example of a short cantilever beam with an aspect ratio of 1:2. The structure is constrained at the top and bottom of the left edge and a load $P = 100N$ is applied at the centre of the right-hand edge of the beam. Figure 6.4(a) shows the initial geometry with loads and constraints. The optimisation problem is solved for $\alpha = 0.40$. The level set domain is discretised with 25×50 square cells and is further refined at $\alpha = 0.45$.

Figure 6.4(b)-(f) shows the evolution of the structural geometry at different stages of the optimisation process. The first automatic hole insertion takes at iteration 3 (Figure 6.4(b)), in the following iterations the structural geometry evolves and hole merges with the outer boundary at iteration 10. Two further holes can be seen in Figure 6.4(d) which then merge with the outer boundary in the subsequent iterations. A final hole insertion takes place at iteration 45 (Figure 6.4(e)) which merges in the next iteration with the outer boundary. At this stage when most of the inefficient material is removed from the design domain only shape optimisation is carried out to reach the target volume fraction at iteration 85. The optimal design obtained is similar to that presented in Figure 5.13(d) and 5.17(b). Further, in this example the optimisation problem is solved with fewer iterations than that without hole insertion as presented in Table 6.1.

The von Mises stress contour plots shown in Figure 6.5 indicate that the proposed hole insertion mechanism efficiently inserts holes in the low stressed regions of the design



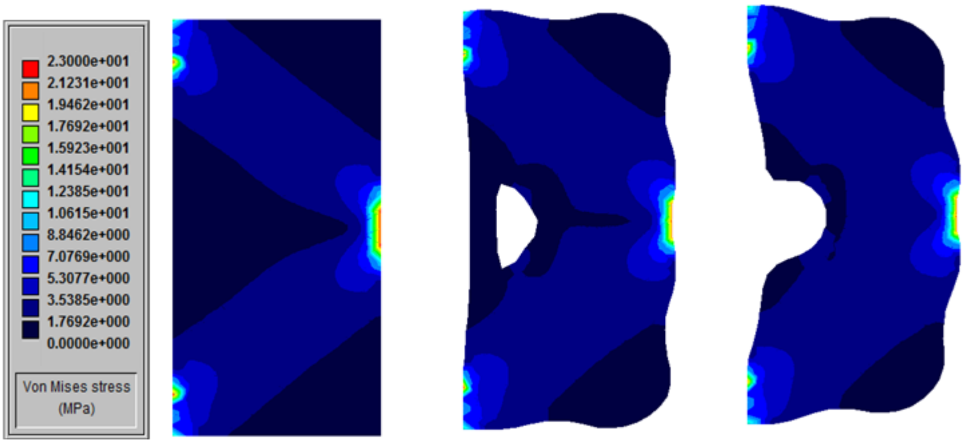
(a) Iteration 0 ($\alpha = 1$) (b) Iteration 3 ($\alpha = 0.93$) (c) Iteration 10 ($\alpha = 0.83$)

(d) Iteration 16 ($\alpha = 0.73$) (e) Iteration 45 ($\alpha = 0.50$) (f) Iteration 85 ($\alpha = 0.40$)

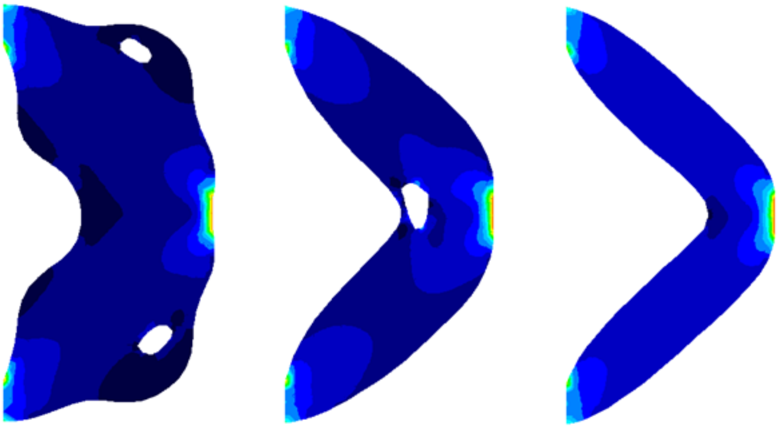
Figure 6.4: Evolution of structural geometry for Example-1.

domain. This clearly indicates that with use of the hole insertion mechanism both shape and topology optimisation can be performed at the same time. In addition, the optimal design obtained in Figure 6.5(f) illustrates quite uniform stress contours which further validate the efficient material distribution capability of the proposed algorithm.

Figure 6.6 displays the evolution history of f_U throughout the optimisation process. In the initial iterations material removal takes place through hole insertion and boundary movements and this results into a rapid decrease in f_U. The peak observed at iteration 30 is related to the hole merging with the outer boundary and its effect dies out in the following iterations. Once most of the inefficient material has been removed f_U slowly decreases up to the termination of the optimisation process. The final value of f_U is 0.15% higher than the one solved without hole insertion (Table 6.1). The control points



(a) Iteration 0 (b) Iteration 3 (c) Iteration 10



(d) Iteration 16 (e) Iteration 45 (f) Iteration 85

Figure 6.5: Evolution of von Mises stress contours for Example-1.

distribution for the initial and final designs are displayed in Figure 6.7.

	Case	Total iterations	Final f_U
Figure 5.19	Without hole insertion	150	261.70
Figure 6.6	With hole insertion	85	262.11

Table 6.1: Comparison of f_U

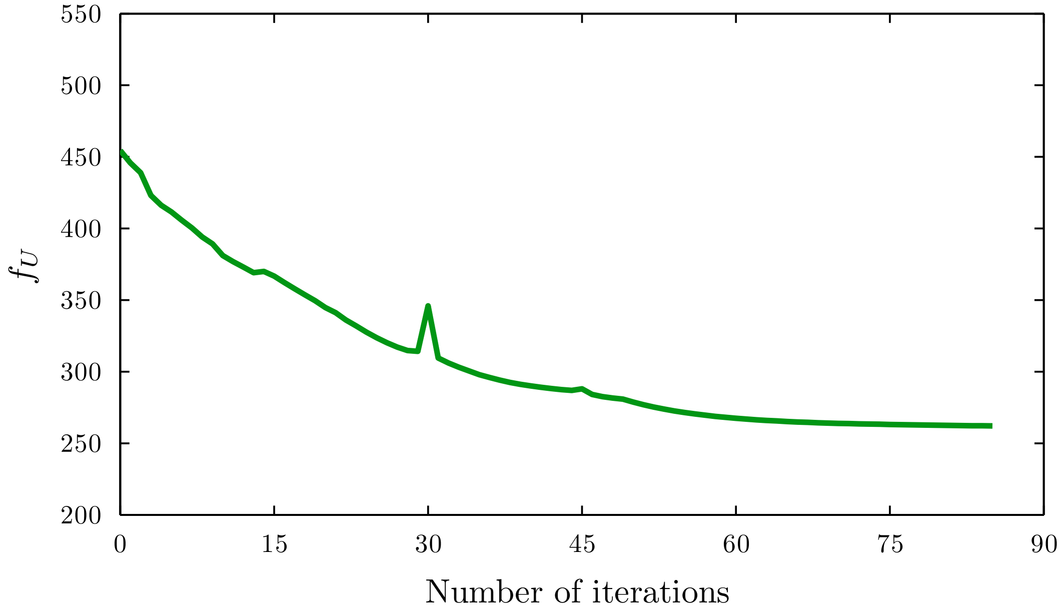
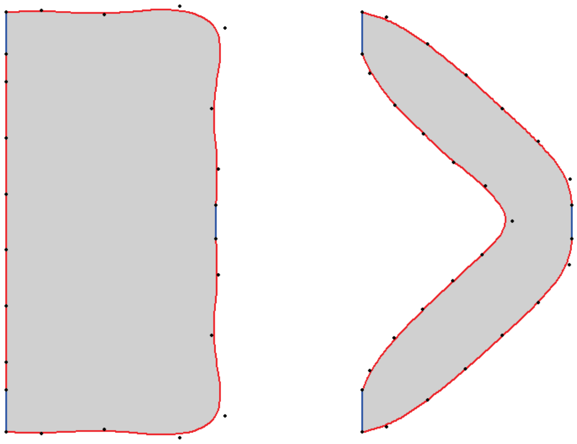


Figure 6.6: Evolution of f_U for Example-1.



(a) Initial design (b) Optimal design

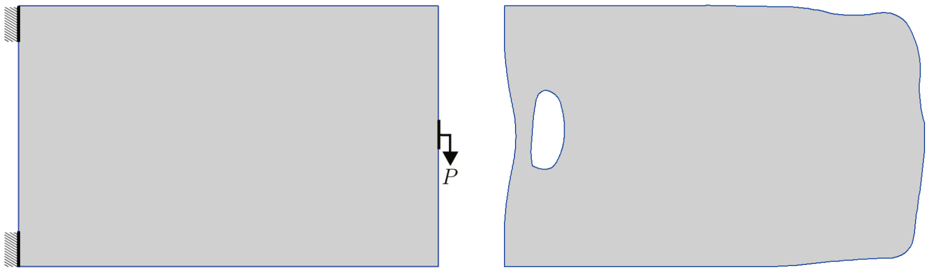
Figure 6.7: NURBS control points distribution for Example-1.

6.3.2 Example-2

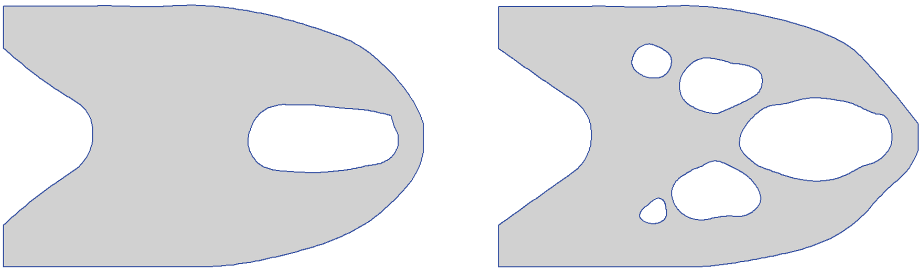
The second example considered in this study is a cantilever beam with an aspect ratio of 1.6:1. The structure is constrained at the top and bottom of the left edge and a load $P = 100N$ is applied at the middle of the right edge. Figure 6.8(a) shows the initial geometry with the applied load and constraints. The target volume fraction, i.e α used in this example is 0.30. The level set domain is discretised with 50×31 square cells and is further refined at $\alpha = 0.35$..

The first automatic hole insertion in a low stressed region occurs in iteration 5 (Figure 6.8(b)), and this hole then evolves and merges with the exterior boundary in the following iterations. The second hole appears in iteration 27, which then evolves over the next iterations until more holes are inserted as shown in Figure 6.8(d). The interior evolving boundaries merge to form larger holes as shown in Figure 6.8(e). The hole insertion, evolution and merging continue throughout the optimisation process which finally ends, when the target volume fraction is reached, with a topology shown in Figure 6.8(h). This figure closely resembles optimal geometries for this benchmark example in the previous works, e.g. [3, 19, 21, 53, 71, 98, 136].

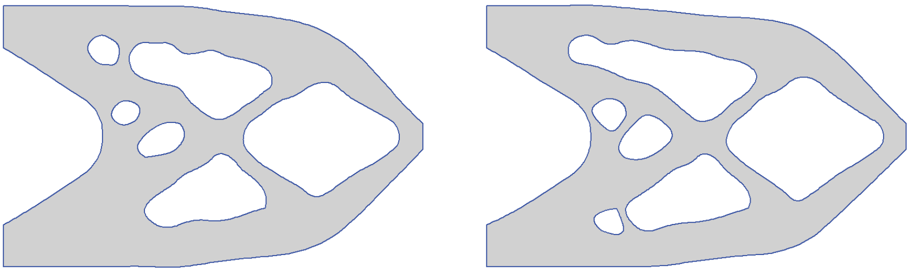
In the present implementation, hole merging takes place automatically and this eliminates the use of an additional mechanism as proposed in the BEM based ESO approach [21]. In the BEM and topological derivative based methods [19, 71], the structural geometry also suffers from jagged edges throughout the optimisation process. The use of these jagged edges within an optimisation process can generate artificial stress concentration regions within the structure, which can mislead the optimisation process. The occurrence of these artificial stress concentration regions can be avoided with the use of highly refined BEM meshes, but at the same time this will increase the computational cost of the optimisation process. In the proposed optimisation method, a NURBS based geometry representation completely eliminates these issues. In addition, the optimal geometry represented in a standard CAD format can be easily integrated within CAD/CAM based design processes. This demonstrates the effectiveness of the proposed method over



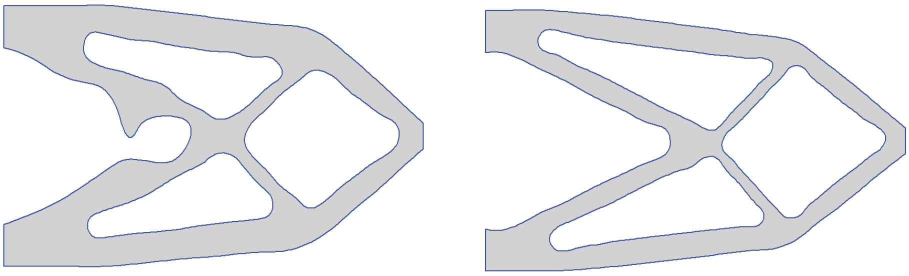
(a) Iteration 0 ($\alpha = 1$) (b) Iteration 5 ($\alpha = 0.95$)



(c) Iteration 27 ($\alpha = 0.72$) (d) Iteration 32 ($\alpha = 0.62$)



(e) Iteration 43 ($\alpha = 0.53$) (f) Iteration 56 ($\alpha = 0.48$)



(g) Iteration 76 ($\alpha = 0.41$) (h) Iteration 125 ($\alpha = 0.30$)

Figure 6.8: Evolution of structural geometry for Example-2.

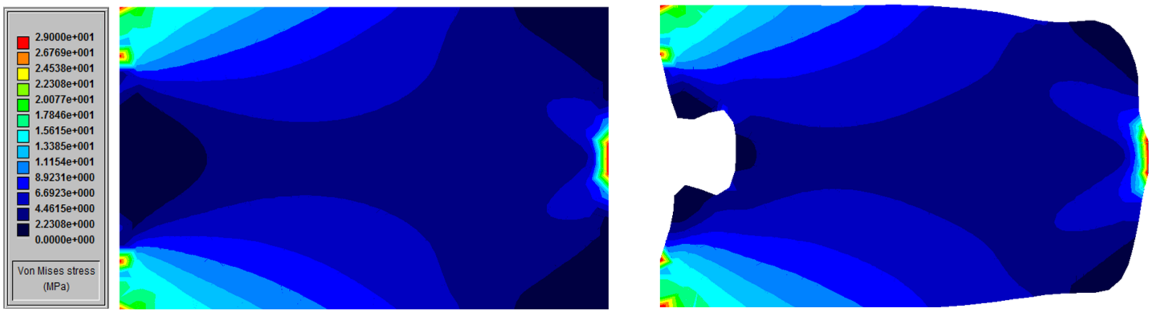
the other LSM based methods presented to date, which lacks this essential feature of the design process.

The available LSM and BEM based optimisation methods presented in [3, 136] always start from an initial guessed design with pre-existing holes. Therefore, the optimal design obtained with these methods are sensitive to the selection of the initial designs. Additionally, these methods have slow convergence for the example presented. However, the proposed optimisation method does not rely on the initial guess design with pre-existing holes and provide optimal solutions having fast convergence.

The von Mises stress distributions during the optimisations process are depicted in Figure 6.9. Comparison of these plots shows that the optimum structure is approaching a fully stressed design with a uniform stress distribution.

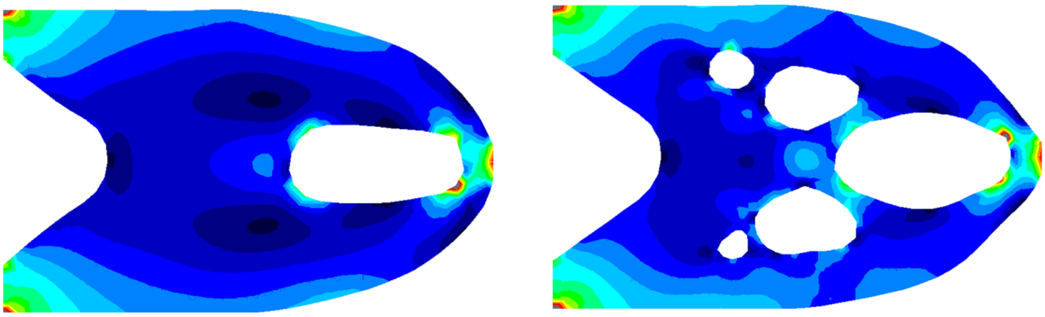
During the optimisation process, the specific strain energy performance indicator is recorded at each optimisation iteration and is depicted in Figure 6.10. During the initial iterations the material removal rate is high, and the specific strain energy decreases rapidly during the initial 26 iterations. The peaks at iterations 27 and 76 are related to the automatic hole insertion and hole merging with the exterior boundary; these peaks continue to be observed up to the last iteration. The amplitudes of these peaks are high for a new large dimension hole insertion, but these peaks die out through the optimisation process to reduce f_U. Finally, on termination of the optimisation process when the target volume fraction is achieved the specific strain energy decreases very slowly.

The distributions of NURBS control points in the initial and final designs are shown in Figures 6.11(a) and (b) respectively. Both plots show a well defined control point distribution. The number of control points in the final geometry is considerably greater than the initial geometry, which shows excellent local control properties of the NURBS geometry, to maintain a smooth and well defined geometry.



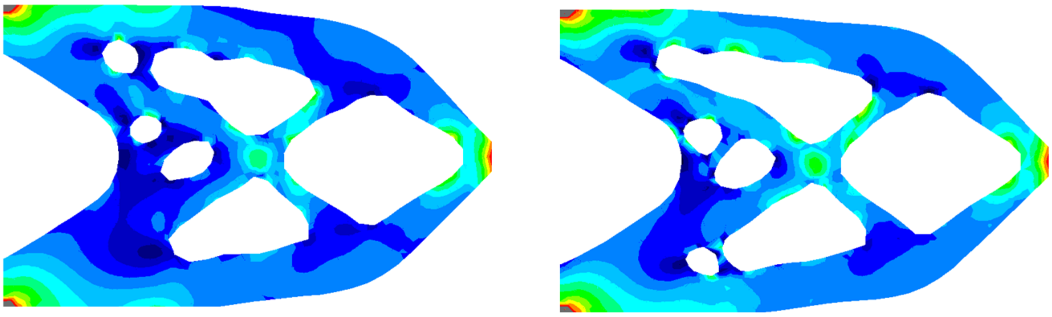
(a) Iteration 0

(b) Iteration 5



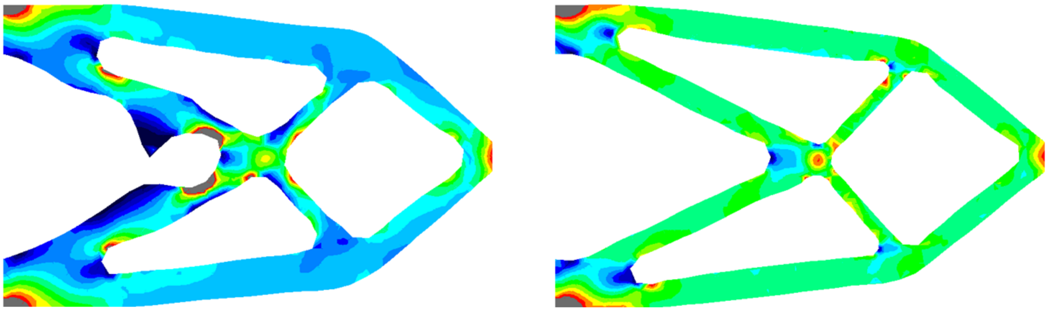
(c) Iteration 27

(d) Iteration 32



(e) Iteration 43

(f) Iteration 56



(g) Iteration 76

(h) Iteration 125

Figure 6.9: Evolution of von Mises stress contours for Example-2.

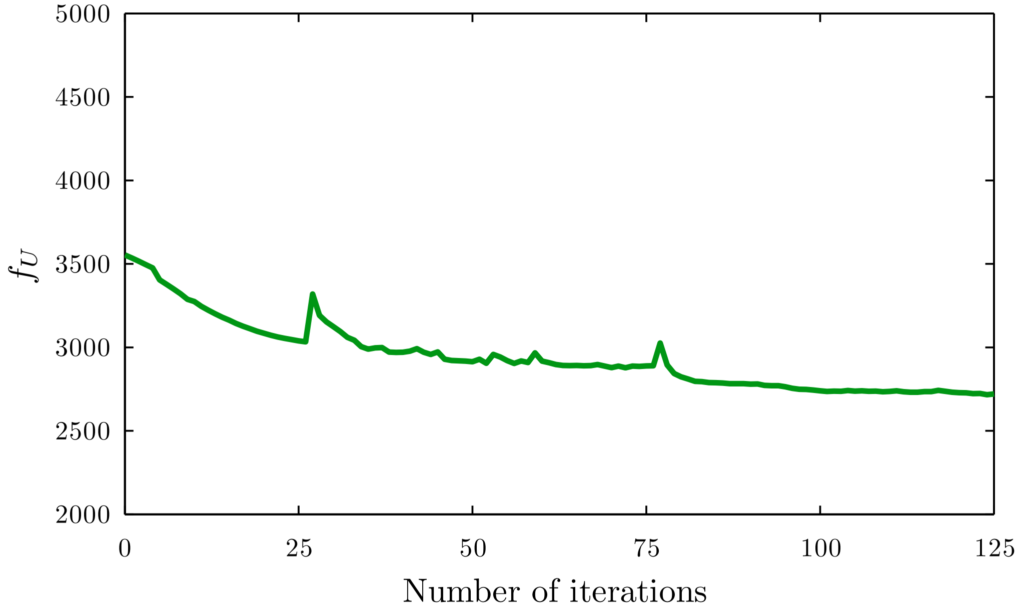


Figure 6.10: Evolution of f_U for Example-2.

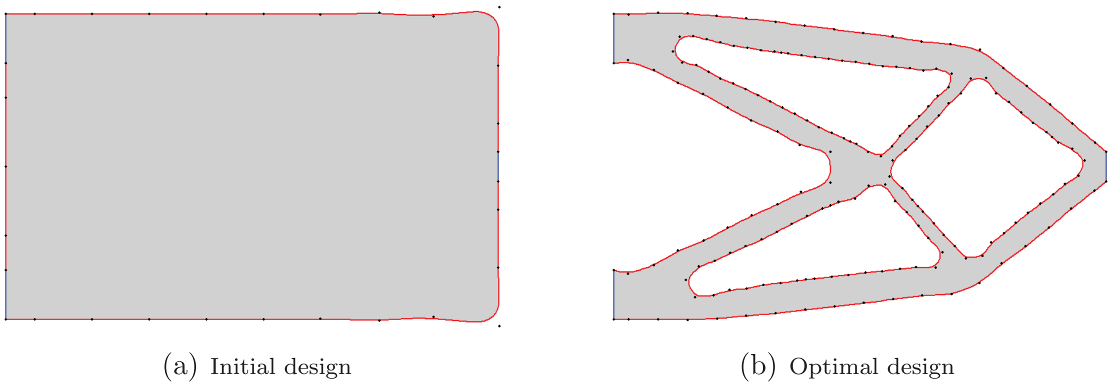

(a) Initial design
(b) Optimal design

Figure 6.11: NURBS control points distribution for Example-2.

6.3.3 Example-3

In the third example a cantilever beam has been used with an aspect ratio of 1.5:1. The structure is constrained at the top and bottom of the left edge and a load $P = 100N$ is applied in the downward direction at the right-hand end of the bottom edge. The initial geometry, with loading and constraints displayed, is shown in Figure 6.12(a). The level set domain is discretised with 50×33 square cells and is further refined at $\alpha = 0.35$.

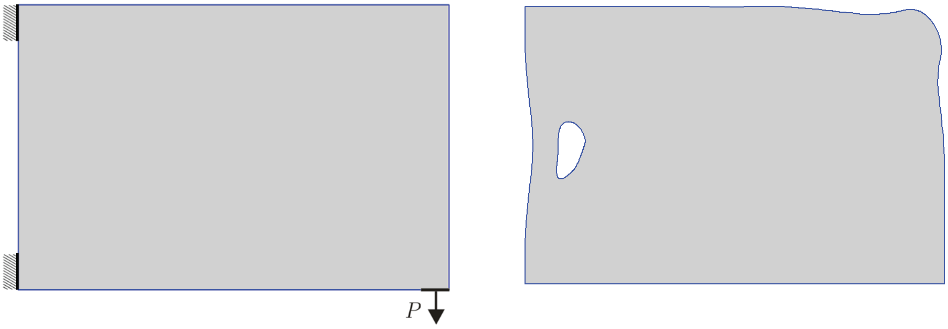
The hole insertion, evolution and merging with other holes at various iterations is shown, alongside the volume at each iteration, in the collected images in Figure 6.12. The final optimum design closely matches those commonly presented for this benchmark example in the topology optimisation literature, i.e. [19, 66, 71, 136, 140].

Similarly to the previous examples, the von Mises stress distribution plot in Figure 6.13 shows a nearly uniform von Mises stress field in the final optimum design.

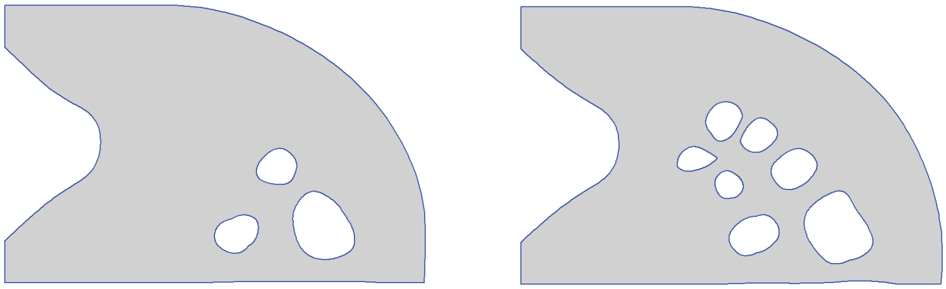
A similar trend of specific strain energy to the previous example is observed in this example shown in Figure 6.14. The peaks occur when a new hole is inserted in the design and then die out after a few iterations. High peaks are observed at iterations 96 and 120 when hole takes place merging with the boundary. This effect dies out in the ensuing iterations until the required volume fraction is reached. A comparison of the final values of f_U of the same problem solved in Section 5.4.3 is presented in Table 6.2, which shows that with the use of hole insertion mechanism provides a better optimum than that with pre-existing holes. Though, more number of iterations are required to reach the same target volume fraction in the current example. The control point distributions are also shown for both the initial and final designs in Figure 6.15.

	Total iterations	Final f_U
Figure 5.34	88	2490.44
Figure 6.14	160	1758.35

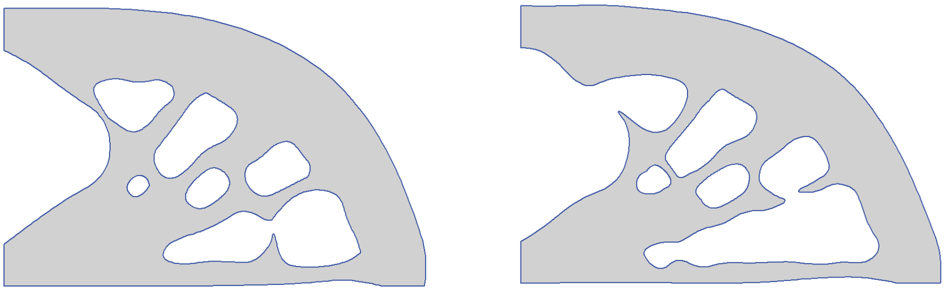
Table 6.2: Comparison of f_U



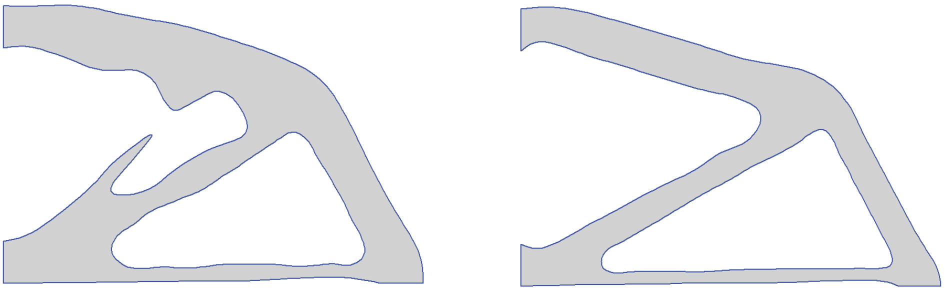
(a) Iteration 0 ($\alpha = 1$) (b) Iteration 3 ($\alpha = 0.98$)



(c) Iteration 53 ($\alpha = 0.75$) (d) Iteration 60 ($\alpha = 0.71$)

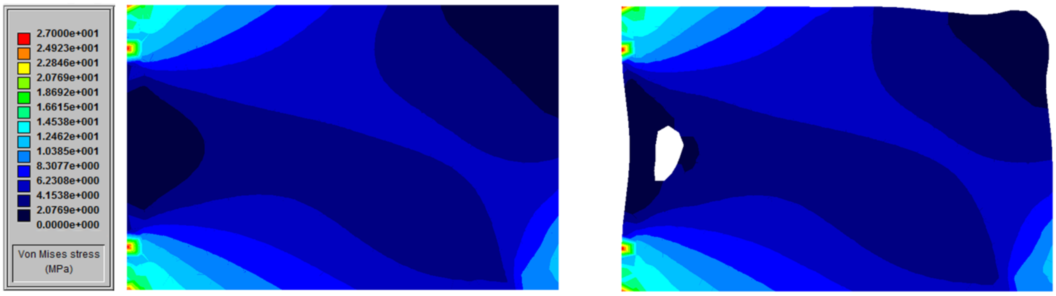


(e) Iteration 86 ($\alpha = 0.58$) (f) Iteration 96 ($\alpha = 0.52$)

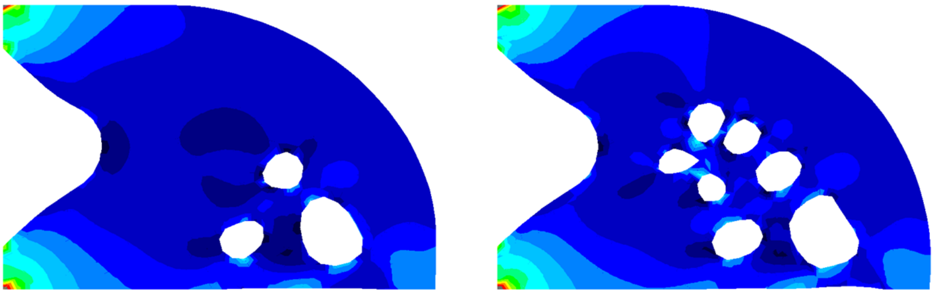


(g) Iteration 120 ($\alpha = 0.39$) (h) Iteration 160 ($\alpha = 0.30$)

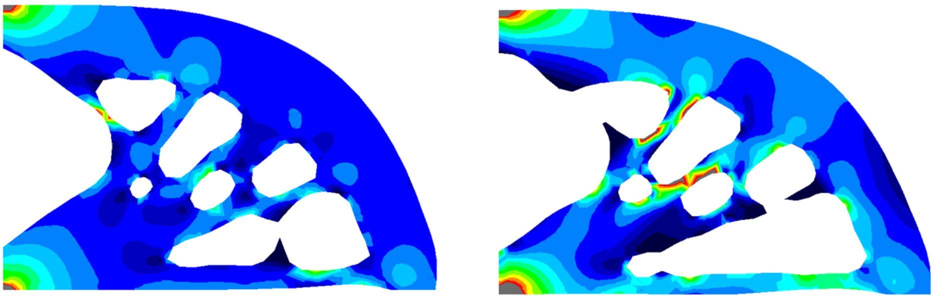
Figure 6.12: Evolution of structural geometry for Example-3.



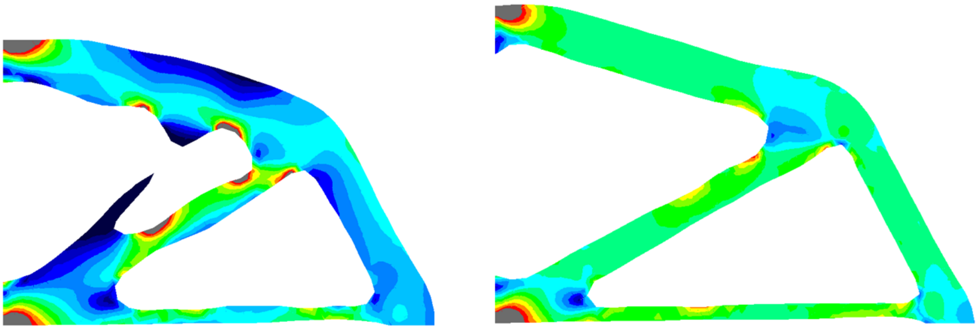
(a) Iteration 0 (b) Iteration 3



(c) Iteration 53 (d) Iteration 60



(e) Iteration 86 (f) Iteration 96



(g) Iteration 120 (h) Iteration 160

Figure 6.13: Evolution of von Mises stress contours for Example-3.

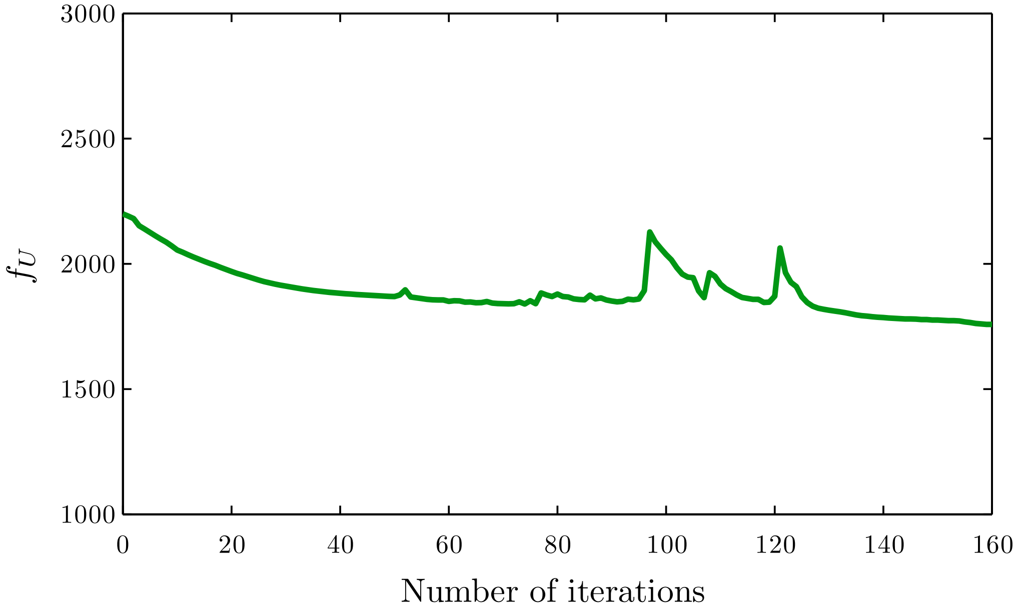
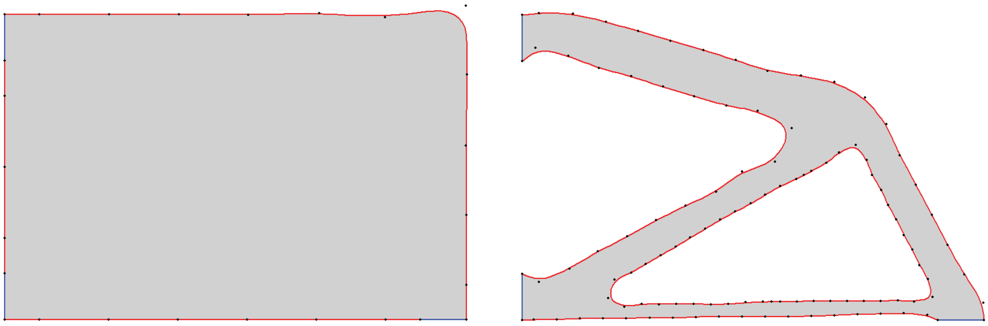


Figure 6.14: Evolution of f_U for Example-3.



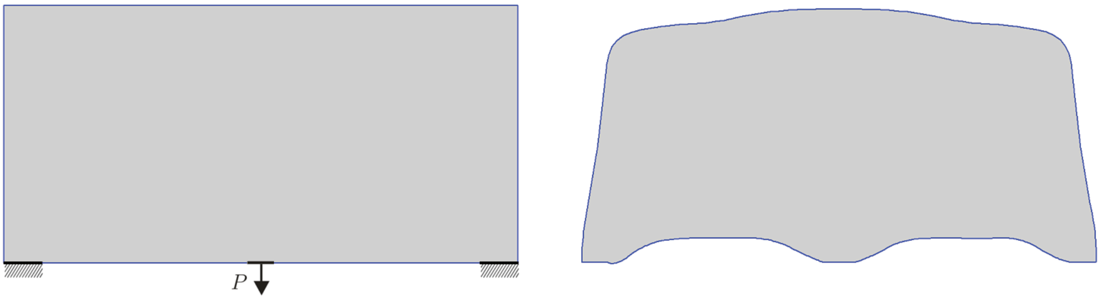
(a) Initial design (b) Optimal design

Figure 6.15: NURBS control points distribution for Example-3.

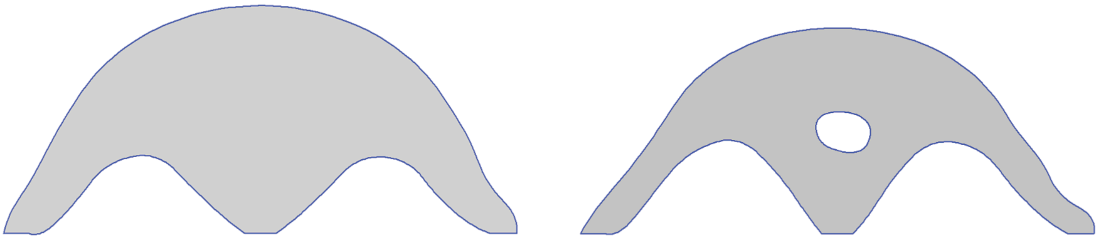
6.3.4 Example-4

The proposed method is finally tested with the geometric model of a Michell structure. The structure with an aspect ratio of 2:1 is constrained at the left and right hand sides of the bottom edge and a vertical downward load $P = 100N$ is applied at the middle portion of the same edge, as shown in Figure 6.16(a). The volume constraint for the optimal topology is set 0.20. The level set domain is discretised with 50×25 square cells and is further refined at $\alpha = 0.35$.

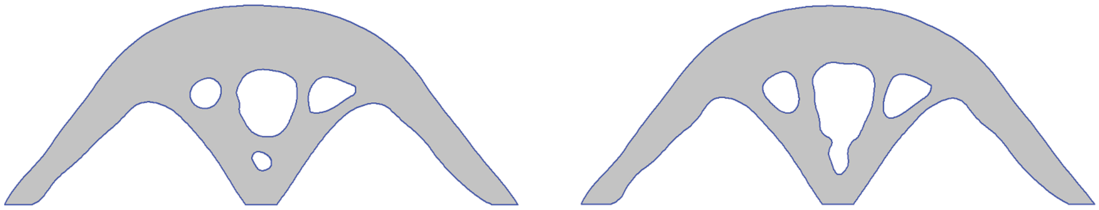
The topology evolution history is shown in Figure 6.16. The RR_i used in this example was 0.1. Due to some randomness of the internal points distribution, slight asymmetry can be observed in the final geometry. Following the previous examples the von Mises stress distribution plots, the specific strain energy evolution history and the control points distribution are shown in Figures 6.17, 6.18 and 6.19, respectively. Some oscillations of the specific strain energy can be observed in Figure 6.18, from iteration 88 to 102. This is mainly related to hole insertion and subsequent merging during the optimisation process as can be seen in Figure 6.18(d)-(f).



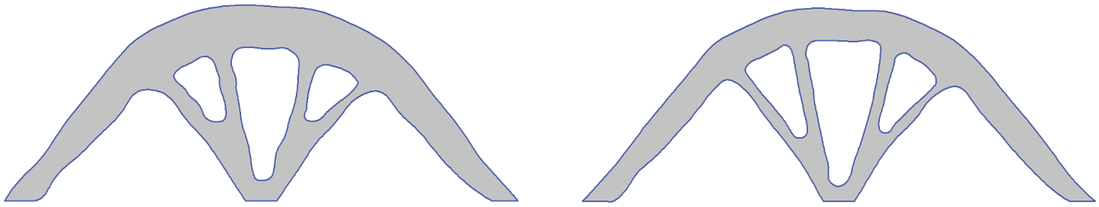
(a) Iteration 0 ($\alpha = 1$) (b) Iteration 11 ($\alpha = 0.86$)



(c) Iteration 65 ($\alpha = 0.50$) (d) Iteration 89 ($\alpha = 0.37$)

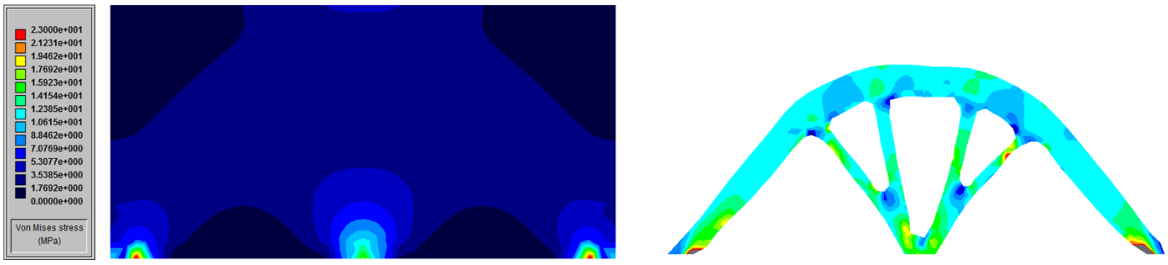


(e) Iteration 97 ($\alpha = 0.30$) (f) Iteration 102 ($\alpha = 0.28$)



(g) Iteration 116 ($\alpha = 0.25$) (h) Iteration 130 ($\alpha = 0.20$)

Figure 6.16: Evolution of structural geometry for Example-4.



(a) Iteration 0 (b) Iteration 130

Figure 6.17: von Mises stress contours of initial and optimal designs for Example-4.

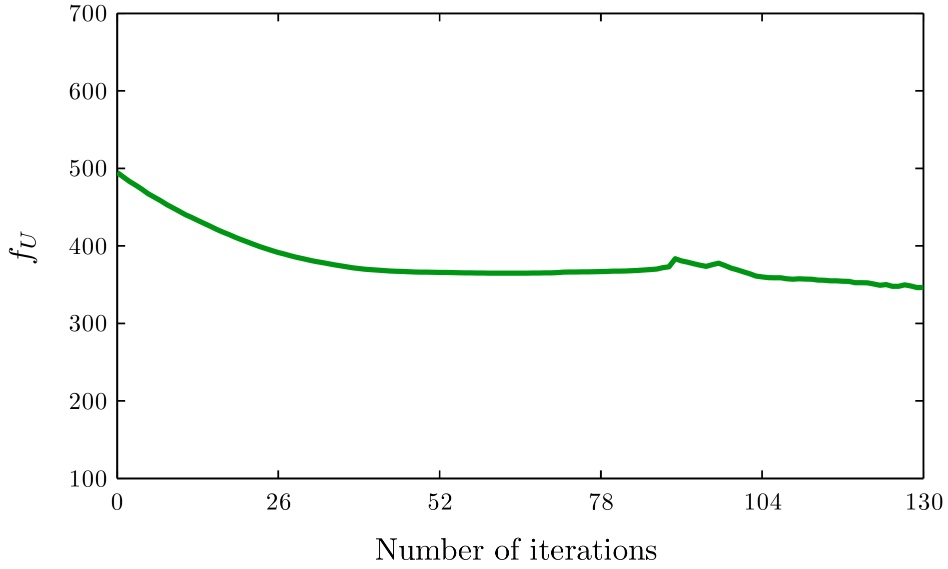
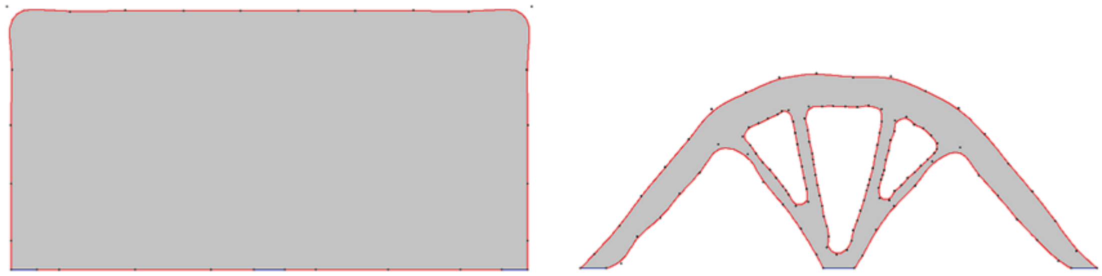


Figure 6.18: Evolution of f_U for Example-4.



(a) Initial design (b) Optimal design

Figure 6.19: NURBS control points distribution for Example-4.

6.4 Conclusions

The optimal design topologies and geometries obtained from the proposed method closely resemble the optima published for a range of benchmark examples in the field of structural optimisation. The method overcomes the deficiency of the direct level set based optimisation methods which are dependent on an initial guess topology with pre-existing holes. The unique combination of BEM, evolutionary approach, LSM and NURBS provides an optimisation technique with fast and accurate structural analysis, automatic insertion and merging of holes and with the added advantage of a smooth geometry both from the structural analysis as well as the manufacturing point of view. It was observed that during the optimisation iterations some of the results appeared to be asymmetric when the problem was symmetric. This is due to the fact that the hole insertion is based on the internal point distribution and in the present work there is some randomness in the algorithm that distributes these internal points in the design domain.

Chapter 7

Correlation between hole insertion criteria

7.1 Overview

In Chapter 6, a von stress based hole insertion criterion has been successfully implemented within a BEM and LSM based framework. However, most of the level set based optimisation methods use the topological derivative [19, 77] as a criterion for hole insertion. Therefore, a detailed study has been carried out in this Chapter to investigate the relationship between the von Mises stress and topological derivative based hole insertion criteria in a BEM and LSM based structural optimisation approach for two-dimensional linear elastic problems. Four different benchmark examples are considered in this study and each is tested against the two hole insertion criteria. The results obtained validate the proposed optimisation method and demonstrate a clear correlation between the two hole insertion criteria.

7.2 Comparison of hole insertion criteria

In Section 6.2.1, a stress based hole insertion criterion has been proposed and successfully implemented within an LSM and BEM based structural optimisation method. In the literature of the direct LSM based optimisation methods, various hole insertion mechanisms presented are based on the topological derivative approach, e.g. [7, 124, 126] etc. Therefore, a study is presented in the following sections which incorporate a topological derivative based hole insertion mechanism in a BEM and LSM based topology optimisation method. Further, a comparison has also been made between the stress and topological derivative based hole insertion mechanisms with respect to their mathematical formulation and the numerical implementation.

7.2.1 Criterion A: von Mises stress based hole insertion

Li *et al.* [64] showed that the criterion of von Mises stress in the classical ESO method is equivalent to the compliance minimisation criterion. Furthermore, it is suggested that the compliance minimisation problem can be solved by directly using the von Mises stress criterion, and vice versa. There is therefore no significant conflict in using a stress criterion alongside strain energy based performance indicator.

The first hole insertion criterion is based on the removal of material around the internal points with the lowest value of von Mises stress (σ_V). In order to make a direct comparison with the topological derivative based hole insertion mechanism, the criterion presented in Section 6.2.1 is slightly modified. Based on this modified criterion, holes are inserted around the internal points which satisfy the following conditions.

$$\sigma_V(i) \leq \ (1 + k_V)\ \sigma_{Vmin} \tag{7.1}$$

where $\sigma_V(i)$ is the von Mises stress at a given internal point i, σ_{Vmin} is the minimum value of von Mises stress over all internal points in the current iteration and k_V is the von Mises stress threshold factor. The value of σ_{Vmin} needs to be modified a little, since it is quite common for an internal point to be located in a region of very low σ_V (on the neutral axis in a bending problem, for example). Instead of using the minimum value, we use the average of the five smallest values of σ_V. The material removal during the optimisation process is also dependent on the value of k_V. If k_V is chosen to be very small the creation of holes is inhibited, whereas a large k_V will give rise to the insertion of very large holes which can destabilises the process. Based on the numerical tests conducted k_V should be used with values in range $0.3 \leq k_V \leq 0.6$. For simplicity we can write $(1 + k_V)$ as f_V. The complete details of the hole insertion procedure are discussed in Table 7.1.

7.2.2 Criterion B: Topological derivative based hole insertion

The second criterion is based on a sensitivity analysis, i.e. the topological derivative concept. The original concept of topological derivative is related to the sensitivity of a cost function when material is removed from the design domain through a small hole insertion. However, the difficulty of establishing a direct mapping between the two different domains (i.e. the domain with and without a hole) restricts its implementation in an optimisation problem. Novotny *et al.* [76] presented an alternative approach to overcome the difficulty associated with the original definition. Based on this new approach, a hole creation is equivalent to the idea of perturbing a pre-existing hole, whose radius tends to zero, thereby providing the possibility to establish a direct mapping between the initial and modified domains. This idea has been used for the derivation of the most useful and easy to implement formulation of the topological derivative (for details see [76]). In a BEM framework this concept has been used by Carretero and Cisilino [19], and Marczak [71], for the optimisation of 2D elasticity problems with the total strain energy as the cost function. In their work the topological derivative $D_T(\vec{x})$ used was a function of the stress invariants, i.e.

$$D_T(\vec{x}) = \frac{2}{1+\nu}\,\sigma \cdot \epsilon + \frac{3\nu - 1}{2(1-\nu^2)}\,tr\sigma tr\epsilon \tag{7.2}$$

where $tr\sigma$ and $tr\varepsilon$ represent the trace of the stress and strain tensors, respectively. According to this criterion, holes are inserted in the design domain around the internal points satisfying the following conditions.

$$D_T(i) \leq \, (1 + k_T)\, D_{Tmin} \tag{7.3}$$

where $D_T(i)$ is the topological derivative at a given internal point i, D_{Tmin} is the minimum value of topological derivative over all internal points in the current iteration and k_T is the topological derivative threshold factor. Similar to factor k_V, the size of the inserted hole is also dependent on the value of k_T. The selection of k_T is based on a correlation found between the two hole insertion criteria (discussed in detail in the following section) and is related to k_V. For simplicity we can write $(1 + k_T)$ as f_T. The hole insertion implementation details are discussed in Table 7.1.

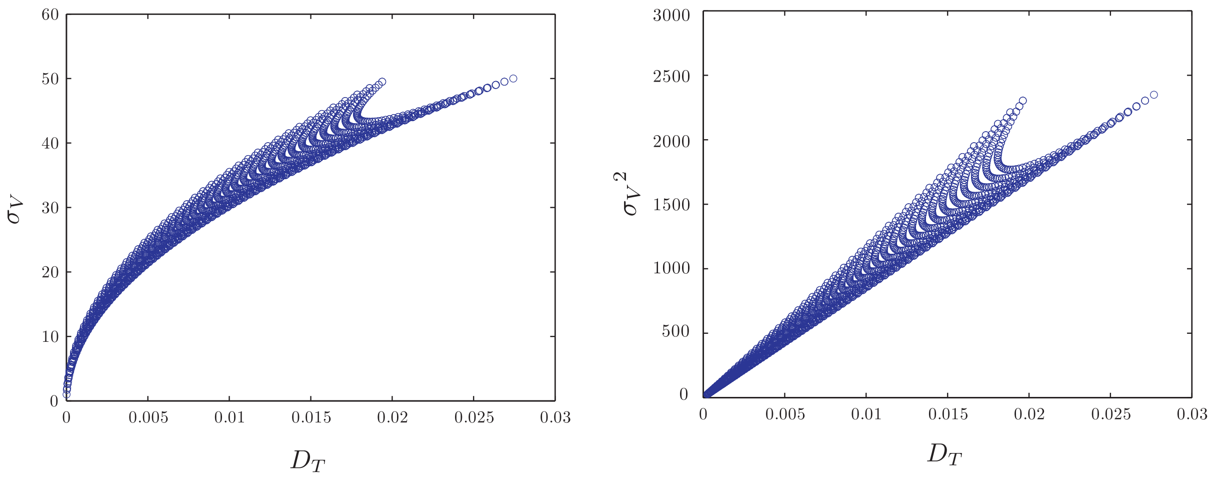
7.2.3 Correlation between criterion A and B

It can be seen that the expressions of σ_V (5.1) and topological derivative (7.2), are based on the stress invariants; this suggests a possible correlation between criteria A and B. In order to deduce this correlation we consider the results obtained for the calculation of σ_V and D_T for various stress states in a plane stress condition, i.e. $\sigma_1, \sigma_2 \in [0; 50], \sigma_3 = 0$. Figure 7.1 shows plots of σ_V and ${\sigma_V}^2$ against D_T, respectively. It is evident from this comparison, that an approximately linear relationship exists between ${\sigma_V}^2$ and D_T. Similarly, using the same stress ranges, some contour plots are generated for $(\sigma_V/\sigma_{Vmax})^2$ and (D_T/D_{Tmax}) (shown in Figure 7.2), which show this correlation between the two criteria in another form.

It is evident that the two approaches are strongly correlated when $\sigma_1 \approx \sigma_2$, suggesting a relationship

$$D_T(i) \;\approx\; C{\sigma_V}^2(i) \tag{7.4}$$

where C is a constant whose value is a function of the material properties. When σ_1 and σ_2 are very different, the behaviour still appears to correspond to (7.4) but with a different constant C. This behaviour is also evident from the straight lines bounding the point distribution in Figure 7.1(b). To proceed with an investigation into the correlation



(a) σ_V and D_T plot (b) ${\sigma_V}^2$ and D_T plot

Figure 7.1: Correlation between σ_V and D_T

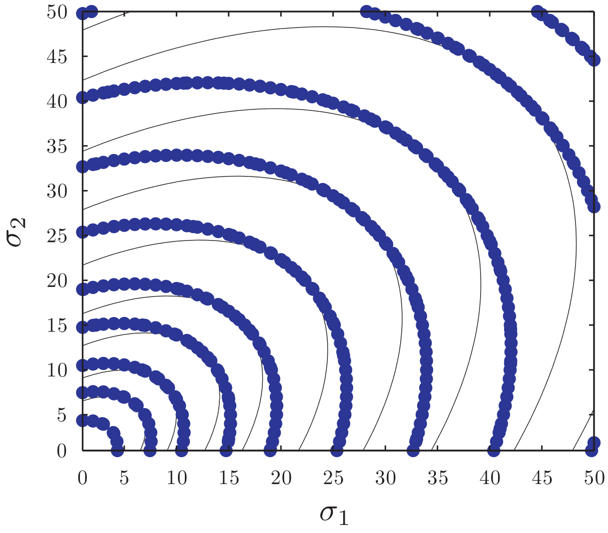


Figure 7.2: $(\sigma_V/\sigma_{Vmax})^2$ and (D_T/D_{Tmax}) contours (-,●)

between the two criteria, we will assume a quadratic relationship

$$D_T(i) \; = \; C{\sigma_V}^2(i) \tag{7.5}$$

which implies a relation between hole insertion factors

$$f_T \; = \; {f_V}^2 \tag{7.6}$$

All examples presented in this paper will have factors (f_V and f_T) chosen in accordance with this relation.

7.3 Optimisation algorithm

During the numerical implementation the same optimisation algorithm is used as presented in Section 6.2 and depicted in Figure 7.3 below. Most of the steps followed in the numerical implementation are already discussed in Section 6.2. However, the main steps followed in the modified hole insertion mechanism are presented in Table 7.1.

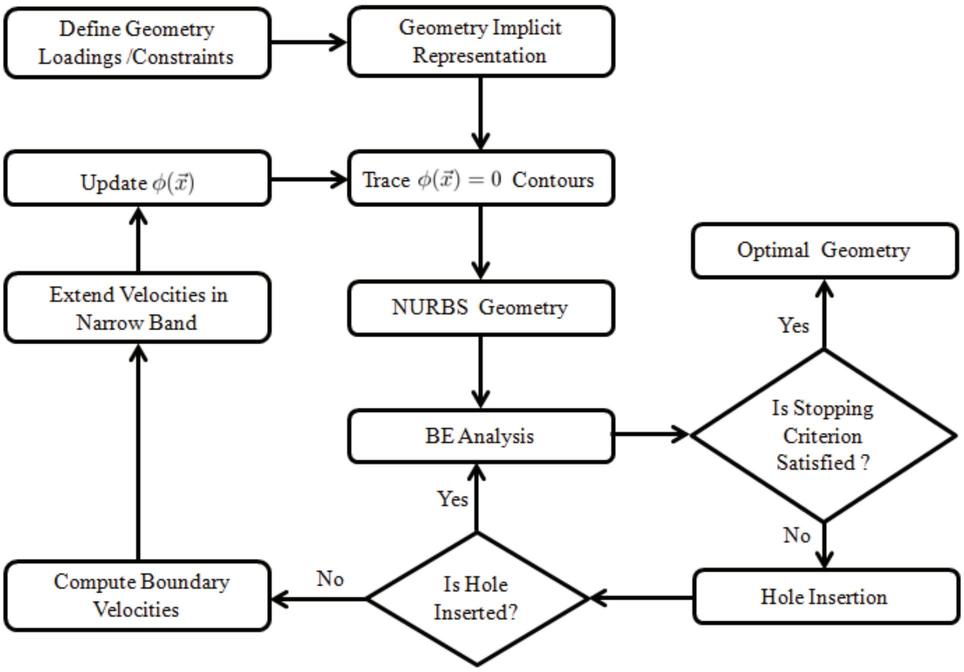


Figure 7.3: Optimisation flow chart

<table>
<tr><th></th><th>Criterion A</th><th>Criterion B</th></tr>
<tr><td>1</td><td>Sort all the internal points in ascending σ_V order.</td><td>Sort all the internal points in ascending $D_T(\vec{x})$ order.</td></tr>
<tr><td>2</td><td>Identify internal points satisfying (7.1)</td><td>Identify internal points satisfying (7.3)</td></tr>
<tr><td>3</td><td colspan="2">The first internal point from step 1 is used as a centre, depicted with ▲ in Figure 6.2(c), for the new hole. Similarly points identified in step 2 are depicted with ■ in Figure 6.2(b). If fewer than five such points are identified, abort the hole insertion</td></tr>
<tr><td>4</td><td>Internal points satisfying a threshold stress level around the central point from step 2, are used to construct a convex polygon shown in Figure 6.2(d).</td><td>Internal points satisfying a threshold topological derivative level around the central point from step 2, are used to construct a convex polygon shown in Figure 6.2(d).</td></tr>
<tr><td>5</td><td colspan="2">The vertices of the convex polygon are taken as control points to generate two NURBS curves (Figure 6.2(e)) to insert the new hole, as shown in Figure 6.2(f).</td></tr>
<tr><td>6</td><td colspan="2">The above steps are repeated until there are no more internal points selected in step 2.</td></tr>
</table>

Table 7.1: Hole insertion criteria

7.4 Examples

The validity and efficiency of the proposed optimisation method are tested against some benchmarking problems in the field of structural optimisation. The material properties used in these examples are: Poisson's ratio = 0.3, Young's modulus = 210 GPa, Yield stress = 280 MPa. Plane stress conditions are assumed with arbitrary thickness of 1 mm.

7.4.1 Example-1

The first example is a short cantilever beam with an aspect ratio of 1:1. The geometry of the structure shown in Figure 7.4, is constrained at the top and bottom of the left edge, and a load $P = 100N$ is applied in the downward direction at the right-hand end of the bottom edge of the beam. The level set domain is discretised with 30 × 30 square cells. The evolutionary parameters used during the optimisation process are $RR = 0.01$ and $RR_i = 0.01$. The optimisation process terminates at the specified volume fraction, i.e. when $\alpha = 0.35$.

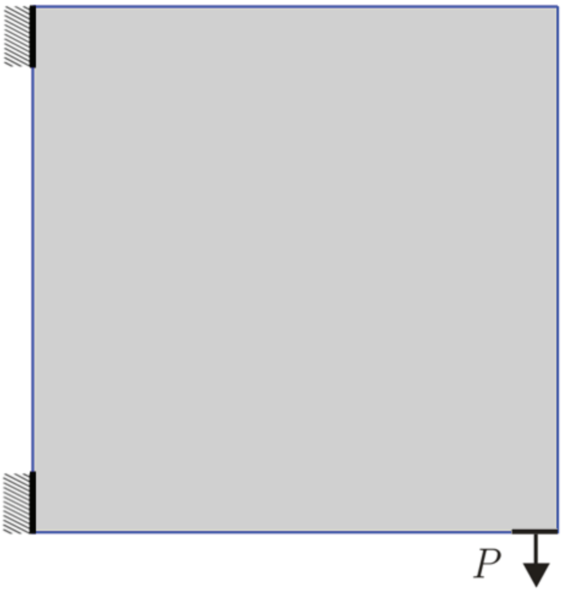


Figure 7.4: Design domain, loading and boundary conditions for Example-1.

In order to validate the selection of k_V and the correlation between the hole insertion criteria, five different cases are studied in this example. In each case the values f_V, f_T (used in accordance with (7.6) and the corresponding optimisation iterations used to reach the optimal design and the final f_U are shown in Table 7.2.

Test case	A			B		
	f_V	Total iterations	f_U	f_T	Total iterations	f_U
1	1.3	174	405	1.69	200	432
2	1.4	184	430	1.96	125	412
3	1.5	130	422	2.25	94	417
4	1.6	130	415	2.56	123	430
5	1.8	102	478	3.24	88	427

Table 7.2: Hole insertion factors and total number of optimisation iterations for Example 1.

In the first case of this example, the given structure is tested against each of the hole insertion criteria independently and the evolution of structural geometry at various volume fractions is shown in Figure 7.5. Comparison of results shows that although the size of holes is different, their insertion takes place in similar regions of the structure leading to a very similar final optimum design which closely resembles that commonly presented in the literature for this type of benchmark example, i.e. [19, 64]. It should be noted that the use of NURBS provides a very smooth geometry throughout the optimisation process without any jagged edges, providing a stable and accurate BE analysis.

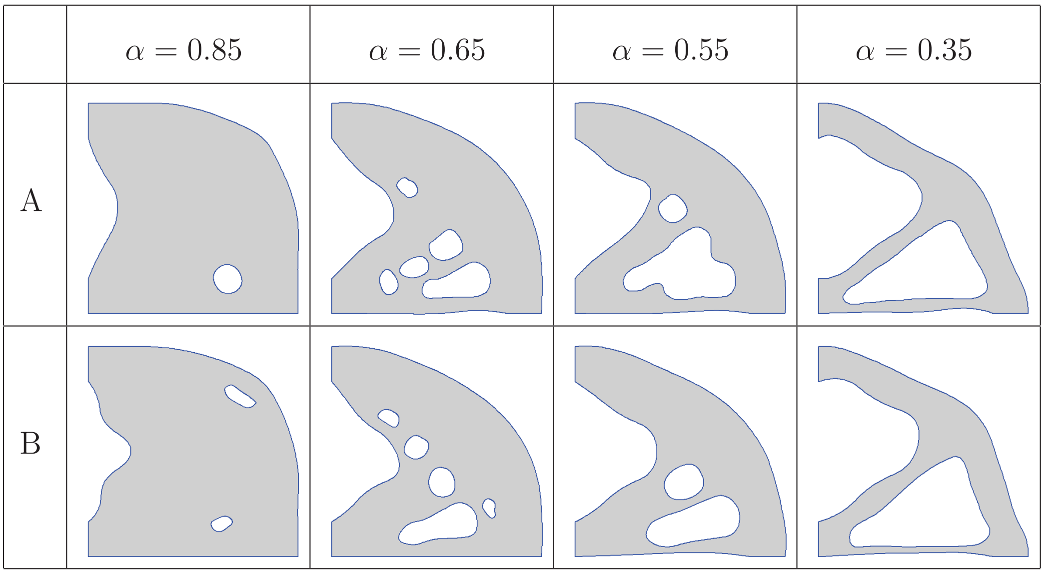


Figure 7.5: Evolution history for Example-1, Case 1.

In order to further validate the correlation between criteria A and B with different hole insertion factors, the results obtained in cases 2, 3 and 4 are presented in Figures 7.6, 7.7 and 7.8, respectively. The results shown in each of these cases indicate a similar behaviour of the evolving and the final optimal geometries, and strongly validate the proposed optimisation method. Furthermore, the results presented clearly demonstrate the correlation between the two criteria as well as the dependency of hole sizes and their insertion rates on the hole insertion factors. It can be seen by the comparison of results in all cases that there are more holes (and also large size holes) with large hole insertion factors at a given volume fraction (e.g. $\alpha = 0.65$), which causes the optimisation process to converge rapidly.

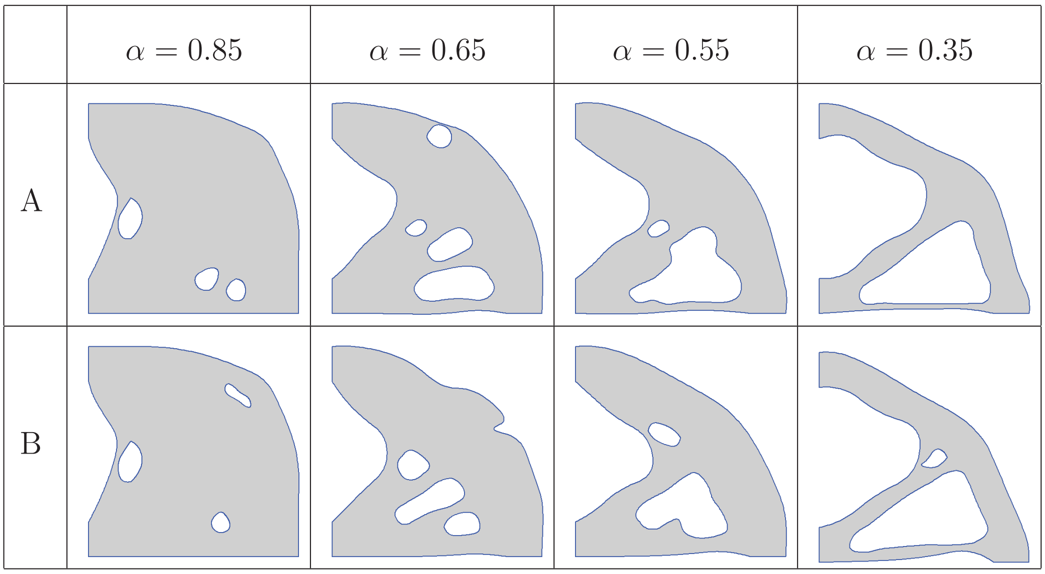


Figure 7.6: Evolution history for Example-1, Case 2.

In the last case of this example, the given structure is tested with higher values of the hole insertion factors. The results presented in Figure 7.9 further validate the dependency of holes sizes on the hole insertion factors, as discussed in the previous cases. In addition, it is evident from the comparison of case 5 with the previous four cases that although an increase in hole insertion factors accelerates the optimisation process, at the same time it destabilises the optimisation process leading towards an optimal design which is different from those obtained previously. This suggests that k_V should be used in the

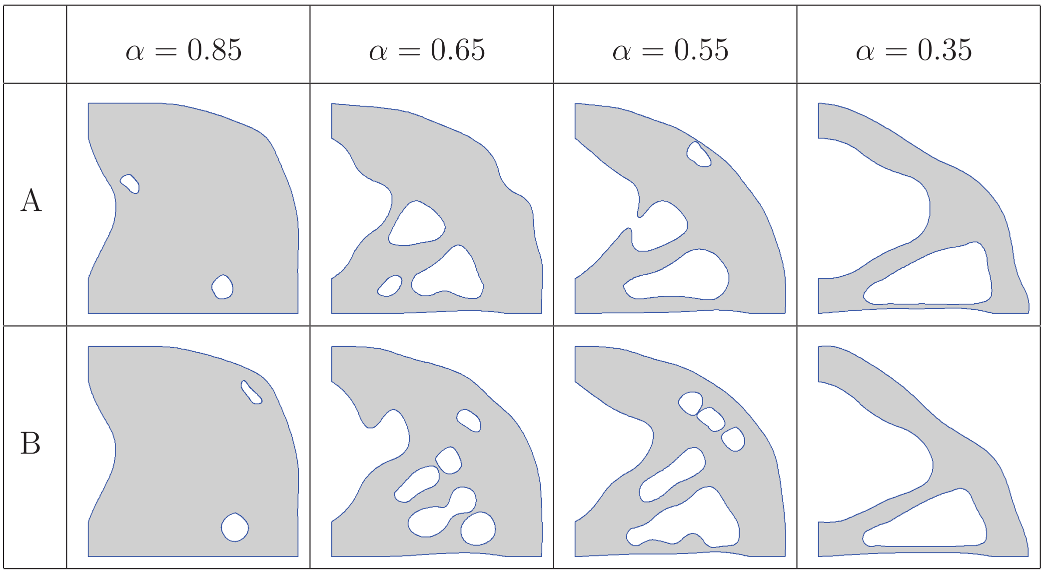


Figure 7.7: Evolution history for Example-1, Case 3.

range $0.3 \leq k_V \leq 0.6$, but based on the stability and optimum number of optimisation iterations a good choice would be either 0.5 or 0.6.

During the optimisation process the specific strain energy f_U is closely monitored for all cases and a comparison of the first two cases is shown in Figure 7.10. The evolution of f_U with respect to the volume fraction for both the cases shows a general reduction with both hole insertion criteria. During the initial iterations in both cases, the hole insertion and boundary movements cause f_U to decrease until α has reduced to 0.60, and then the behaviour starts diverging with some peaks. These peaks are related to the automatic hole insertion and hole merging with the exterior boundary and continue to be observed up to the final volume fraction. The magnitudes of these peaks are large when a hole is inserted near to the exterior boundary immediately merges with it, but then decay through the optimisation process to reduce f_U. Finally, on termination of the optimisation process when the target volume fraction is achieved, it appears that the specific strain energy is still decreasing, suggesting that extending the optimisation process by more iterations would enable further reduction in this performance indicator if desired. In both cases, the optimisation process terminates at different f_U levels with an approximate difference of 6% and 4% in the first and second case, respectively. The difference between f_U in

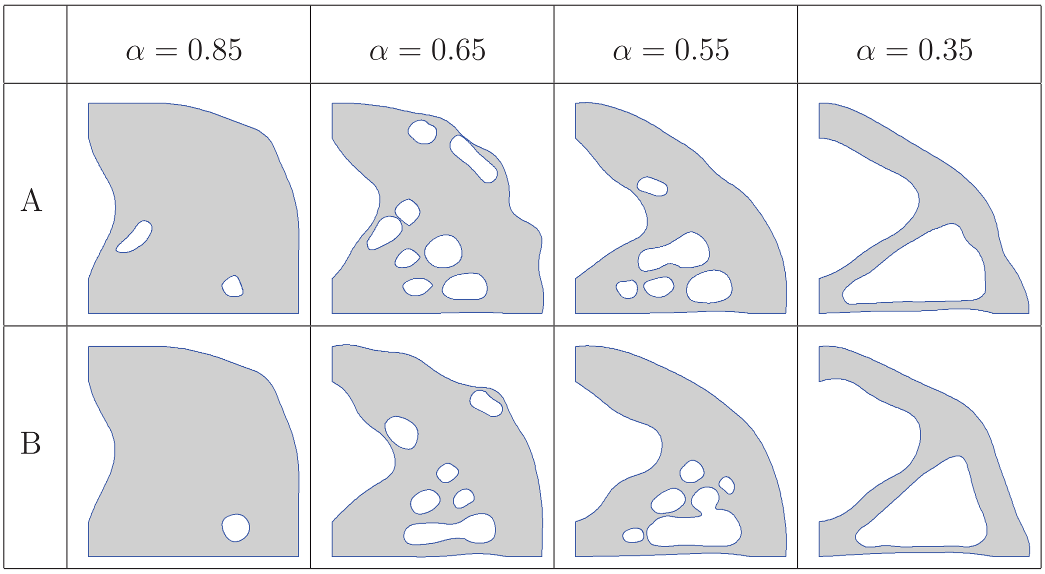


Figure 7.8: Evolution history for Example-1, Case 4.

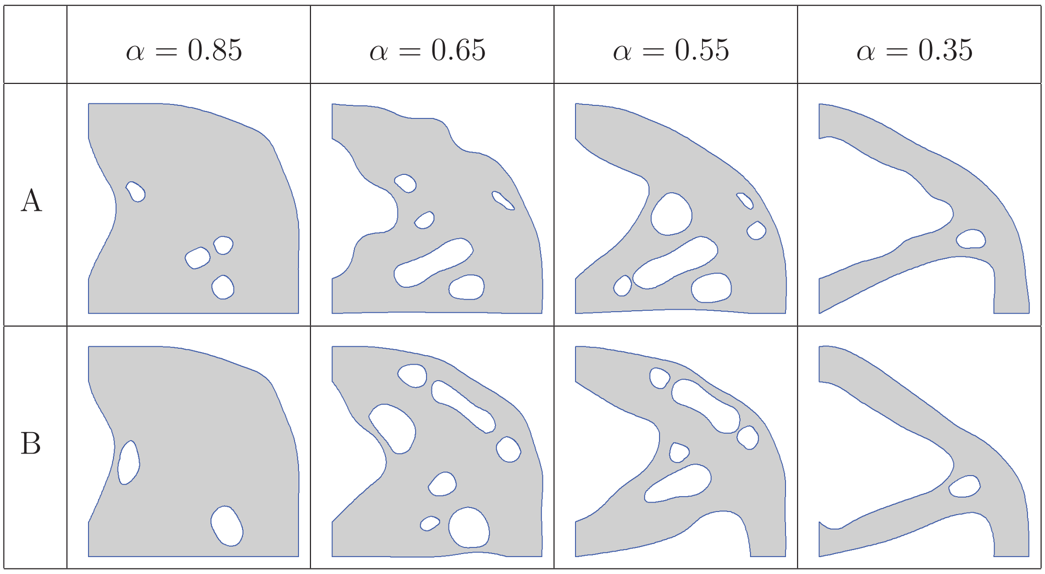
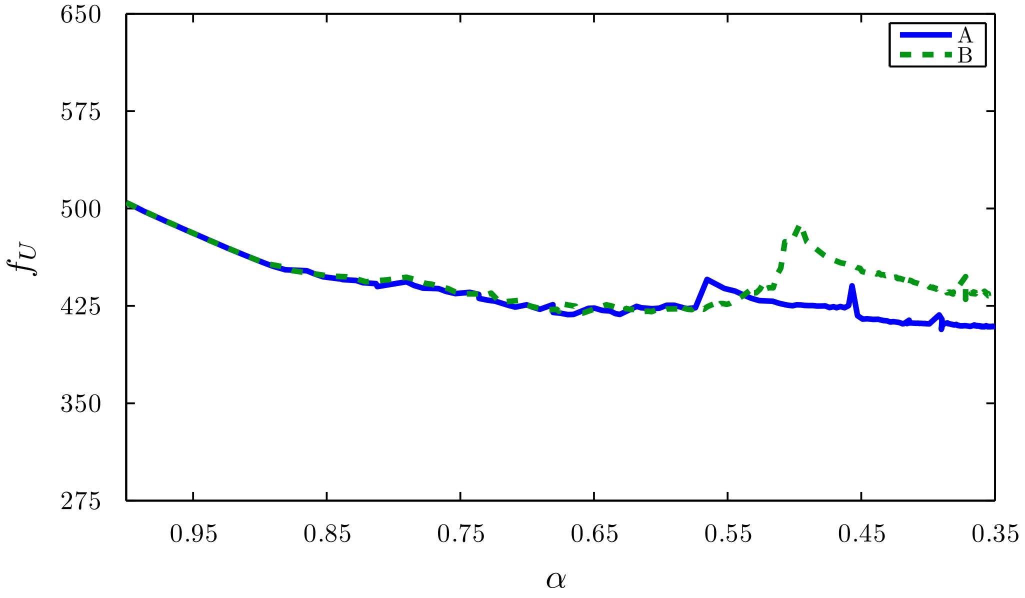


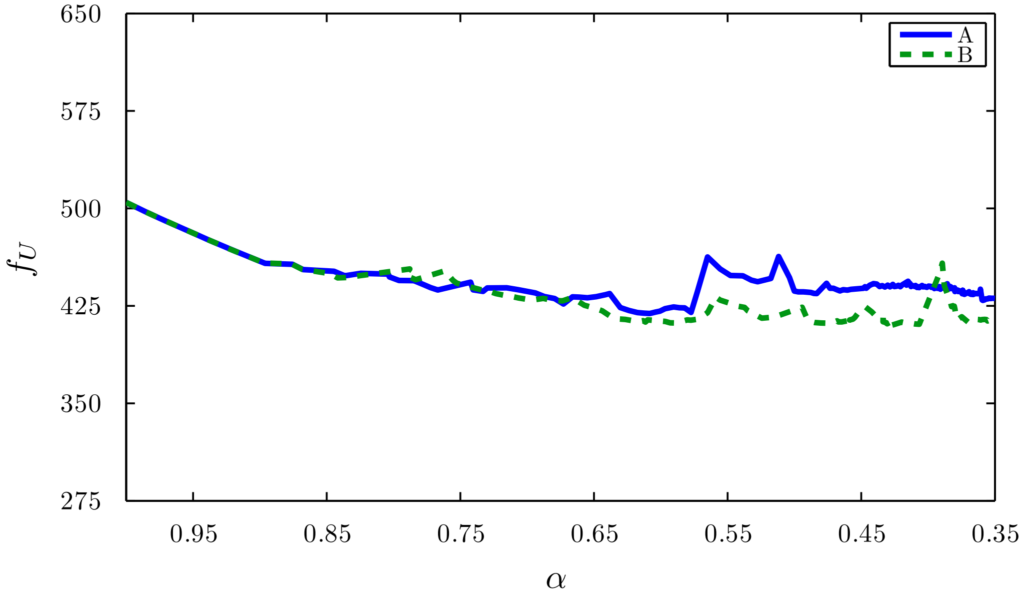
Figure 7.9: Evolution history for Example-1, Case 5.

each case with the two criteria is due to different peaks at different stages during their evolution. The randomness in the internal points causes the insertion of holes near the exterior boundary at different locations and with different sizes.

The robustness of the proposed optimisation method is further validated with different initial designs. For this purpose three different initial designs have been considered and the optimisation problem is solved using criterion A. The number of holes, f_V, total



(a) Evolution of f_U with $f_V = 1.3$ and $f_T = 1.69$



(b) Evolution of f_U with $f_V = 1.4$ and $f_T = 1.96$

Figure 7.10: Evolutions of f_U for Example-1, Cases 1 and 2.

number of iterations and f_U for each of the cases are shown in Table 7.3. The evolution of the structural design in each case is depicted in Figure 7.11. The proposed algorithm allows new hole insertion during the optimisation process. Comparison of results shows that for the three different initial designs the final optimal topology obtained is similar to that available in the literature of this type of benchmark example. In addition, the results demonstrate a reduction in the total number of optimisation iterations for the initial design with pre-existing holes with similar performance, i.e. f_U.

Initial design	No of holes	f_V	Total iterations	f_U
a	0	1.5	130	422
b	1	1.5	105	420
c	8	1.5	108	416

Table 7.3: Details of various parameters in the optimisation of different initial designs.

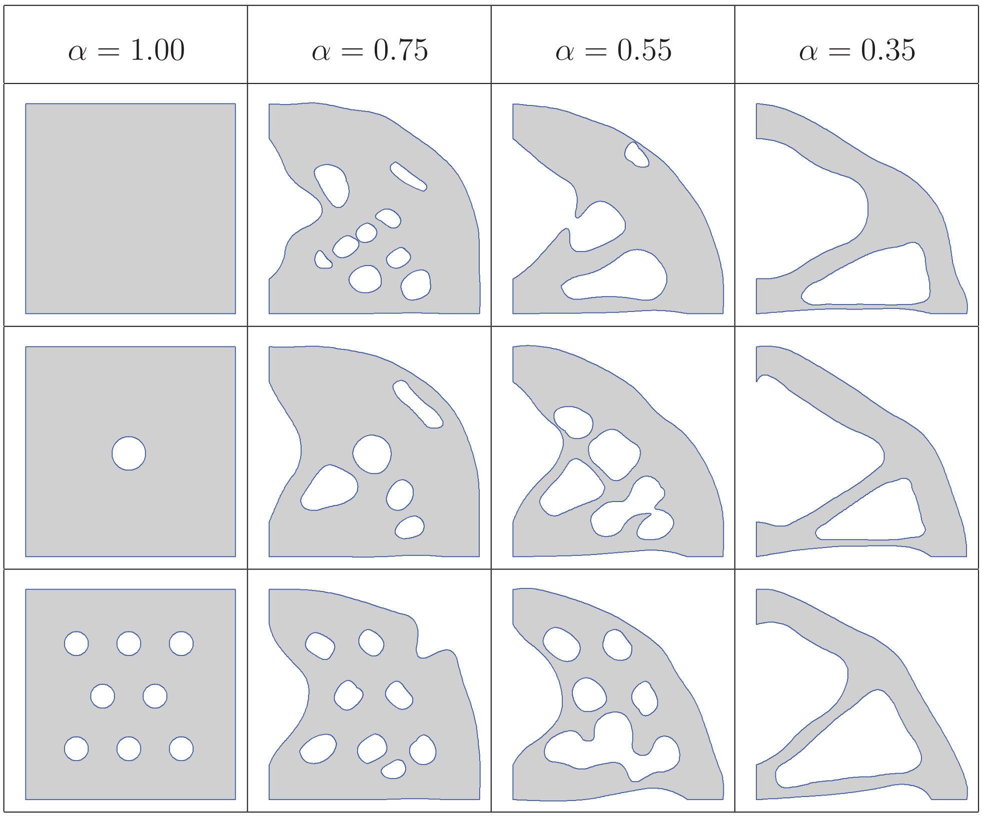


Figure 7.11: Evolution history for Example-1, Case 3, using different initial designs.

7.4.2 Example-2

In order to further validate the proposed optimisation method and the correlation between the two criteria, the second example is a cantilever beam with an aspect ratio of 1.6:1 as shown in Figure 7.12. The structure is constrained at the top and bottom of the left edge, and a load $P = 100N$ is applied in the downward direction at the middle of the right-hand edge. The level set domain is discretised with 40×25 square cells. In this particular example three different cases are studied in detail. The first two cases demonstrate the correlation between the two hole insertion criteria with a new geometry and constraints, using different hole insertion factors and in the third case a comparison has been made with an increase in RR_i. In all three cases $RR = 0.01$. The hole insertion factors, total number of optimisation iterations, f_U and RR_i used in each case are shown in Table 7.4. The specified minimum volume fraction, i.e α for this example is 0.35.

Figure 7.12: Design domain, loading and boundary conditions for cantilever beam.

Test case	A			B			RR_i
	f_V	Total iterations	f_U	f_T	Total iterations	f_U	
1	1.5	97	1767	2.25	90	1748	0.01
2	1.6	82	1770	2.56	81	1788	0.01
3	1.5	52	1744	2.25	50	1772	0.05

Table 7.4: Hole insertion factors and total number of optimisation iterations used for Example 2.

In the first case of this example, the two hole insertion criteria are compared in Figure 7.13. It should be noted that the values of hole insertion factors used are based on the results discussed in the previous example. Comparison of the results shows that holes are inserted in similar regions of the structure with the two different criteria at a given volume fraction (e.g. $\alpha = 0.75$). During the optimisation process the randomness of the internal points causes hole insertions at different locations with criteria A and B, respectively. However, the final optima obtained are very close to each other and also resemble the optimal design of this type of benchmark example in the literature, i.e. [3, 19, 21, 53, 71, 98, 136]. It is evident from the results that an increase in the hole insertion factors gives rise to slightly larger hole insertion in the design domain; this accelerates the optimisation process to converge rapidly towards the optimal design as seen in Table 7.4.

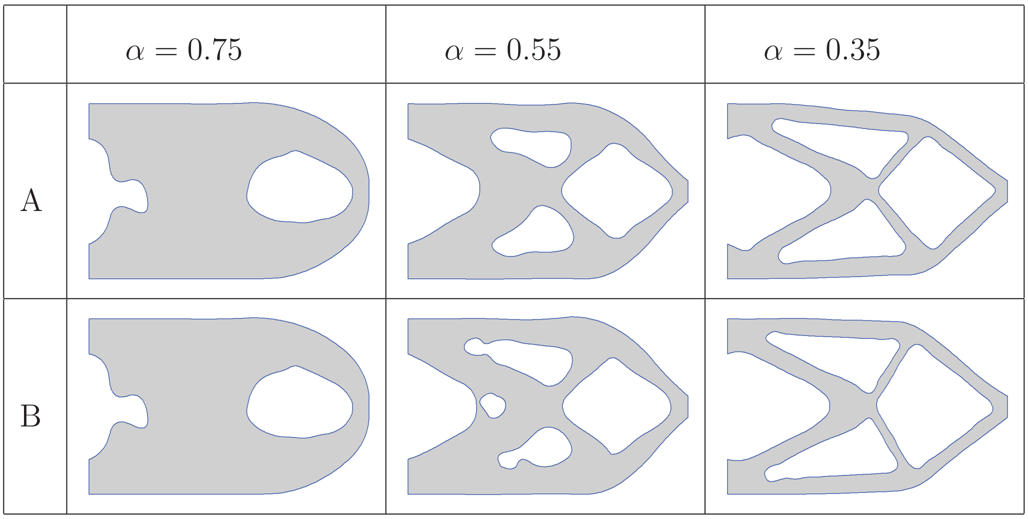


Figure 7.13: Evolution history for Example-2, Case 1.

In the third case a comparison has been made with a higher RR_i value. The results compared in Figure 7.15 show the same evolution of the optimal geometry as in the previous two cases (i.e. case 1 and 2), but the total number of iterations is considerably reduced (as shown in Table 7.4). Although Figure 7.15 shows that for Example-2 the use of $RR_i = 0.05$ can be successful in reducing the required number of iterations, it is recommended to use a lower value of $RR_i = 0.01$. This is because, in some cases a higher

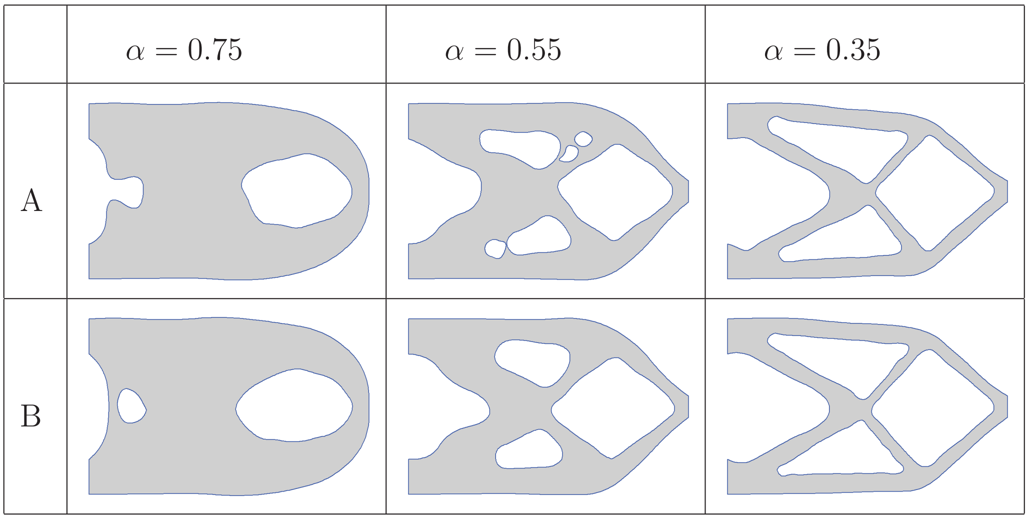


Figure 7.14: Evolution history for Example-2, Case 2.

value of RR_i causes the removal of an excessive amount of material, destabilizing the optimisation process and leading towards non-converged solutions.

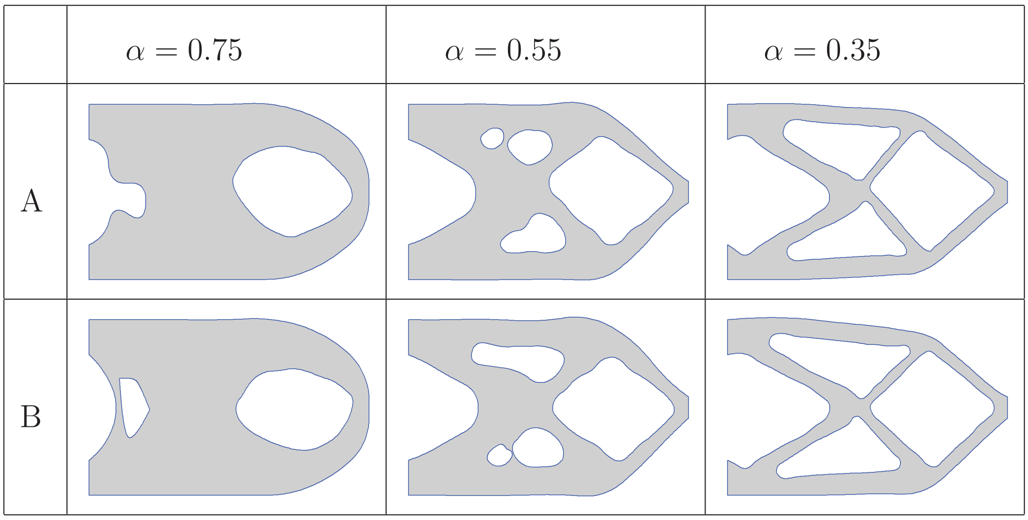
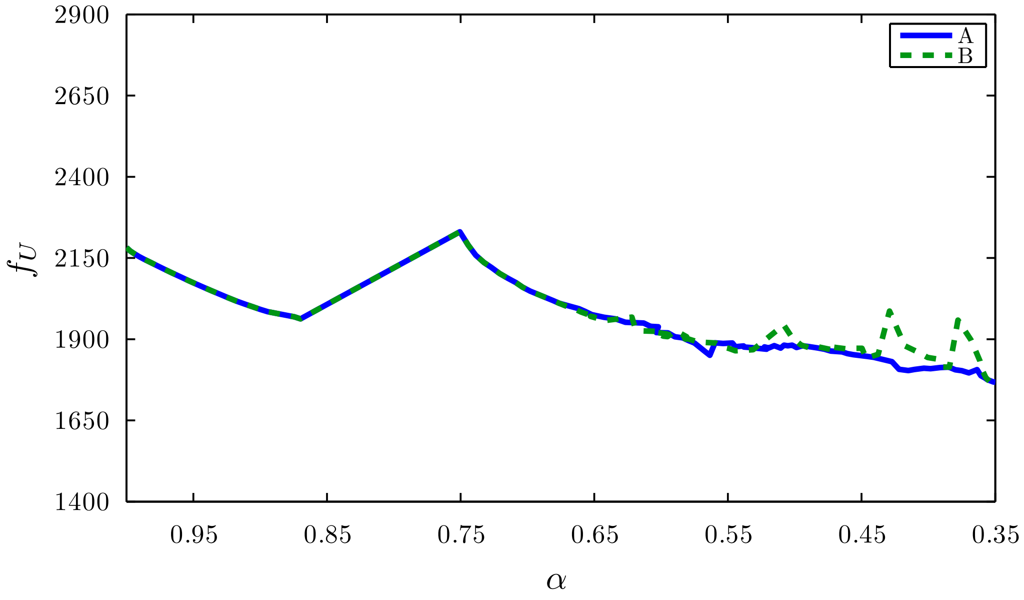
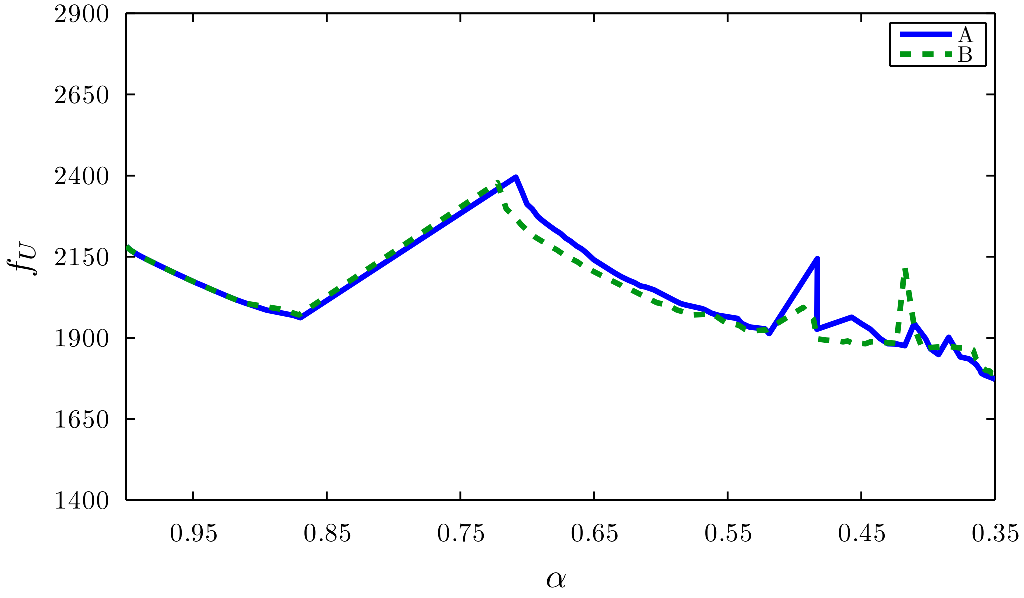


Figure 7.15: Evolution history for Example-2, Case 3.

A similar trend of specific strain energy to the previous example is also observed in this example for the first two cases shown in Figure 7.16. The peaks occur when a new hole of relatively large size is inserted in the design domain near to $\alpha = 0.75$ and then die out rapidly. It is also evident from this comparison that peaks in Figure 7.16(a) are lower than those in 7.16(b). This is due to the insertion of different size holes with different f_V and f_T in each case. The behaviour of f_U is almost identical up to $\alpha = 0.55$ in both cases with the two criteria; later on the additional hole insertions near the boundary and its immediate merging with it generate high peaks.



(a) Evolution of f_U with $f_V = 1.5$ and $f_T = 2.25$



(b) Evolution of f_U with $f_V = 1.6$ and $f_T = 2.56$

Figure 7.16: Evolutions of f_U for Example-2, Cases 1 and 2

7.4.3 Example-3

In this example we apply the proposed optimisation algorithm with different hole insertion criteria to another benchmark example in the field of topology optimisation known as the L-beam [7]. The model is constrained at the top edge and a load $P = 100N$ is applied at the middle of the right edge as shown in Figure 7.17. The level set domain is discretised with 50 × 50 square cells. The various factors used in this example are: $RR = 0.01$, $RR_i = 0.01$ and the optimisation process terminates when $\alpha = 0.45$.

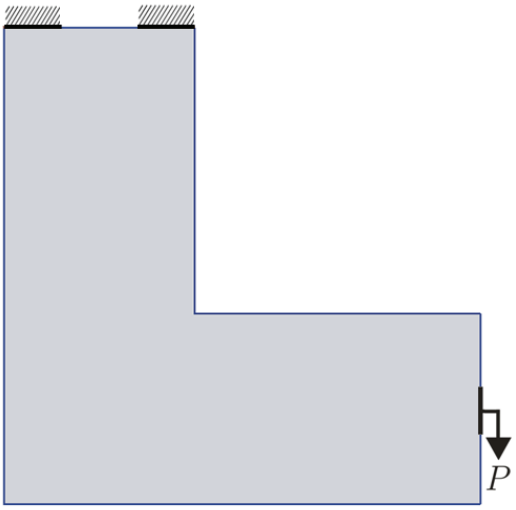


Figure 7.17: Design domain, loading and boundary conditions for L-beam.

The results obtained during the optimisation iterations at various volume fractions, α (with the two hole insertion criteria) are depicted in Figure 7.18. The hole insertion factors used in this example are $f_V = 1.6$ and $f_T = 2.56$. Comparison of the results shows the same behaviour of the evolving geometry as observed in the previous examples. The optimal design generated with the two criteria resembles those available in the literature [7, 129]. The evolution of f_U depicted in Figure 7.19 with criterion A and B is almost identical and follows the same trend as observed in the previous examples.

In order to validate the capability of the proposed optimisation method for handling the peak stresses, the von Mises stress distribution at different iterations is shown in Figure 7.20. Comparison of the stress distribution results shows that the optimisation method allows the peak stresses, observed at iteration 0, to reduce by distributing over

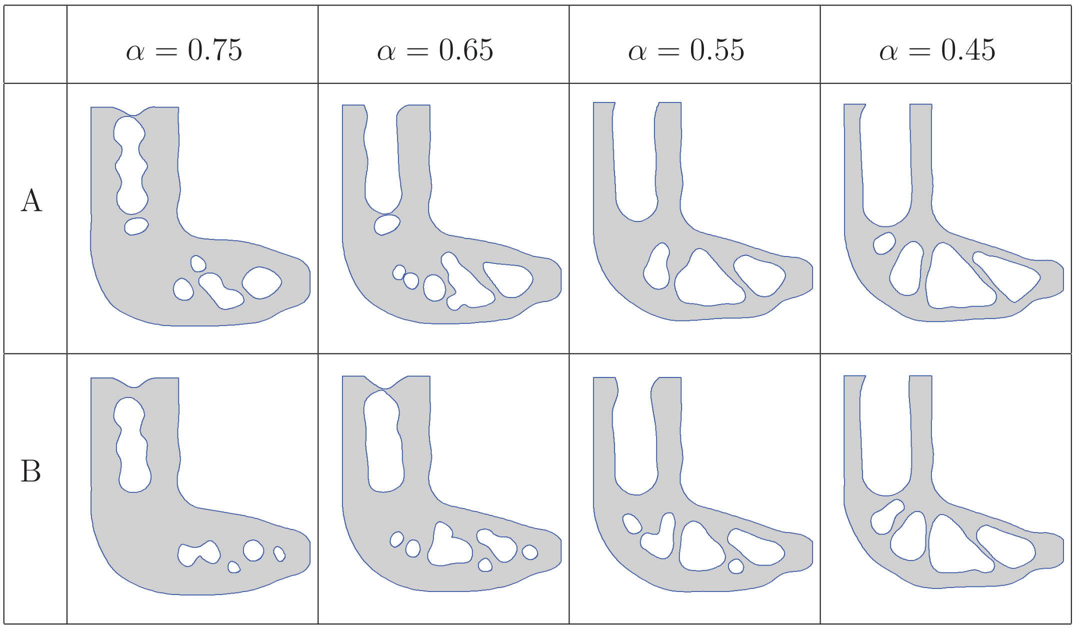


Figure 7.18: Evolution history for L-beam.

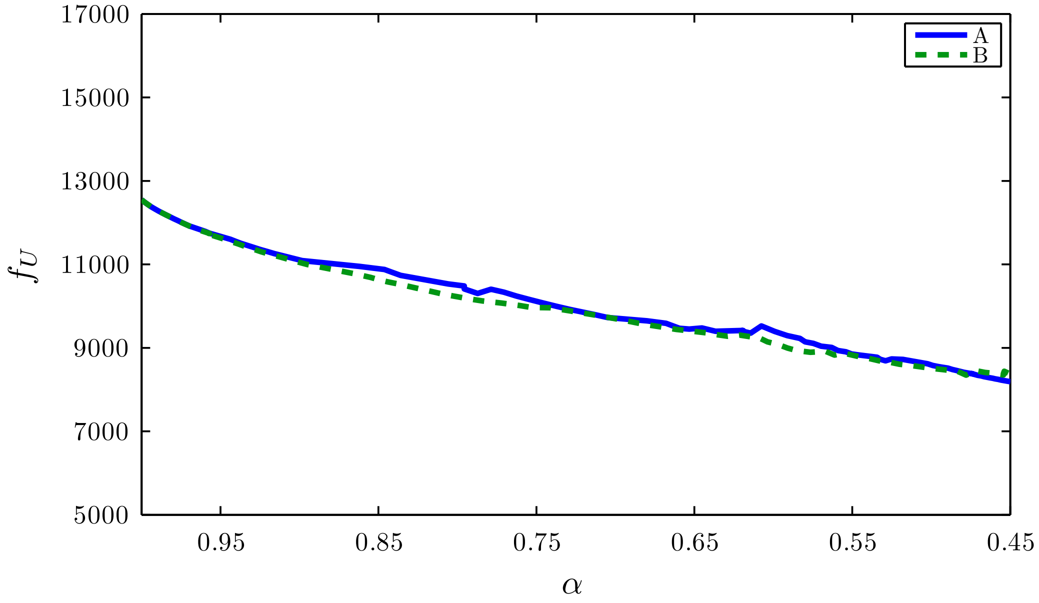
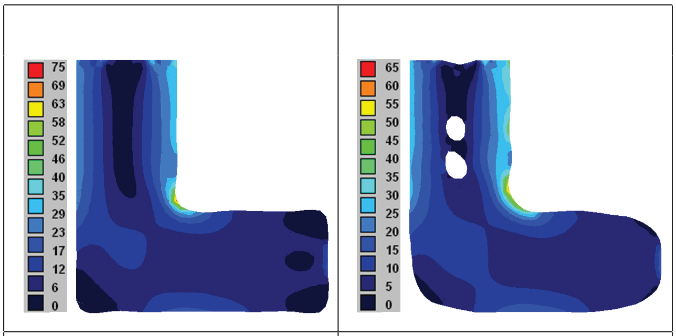
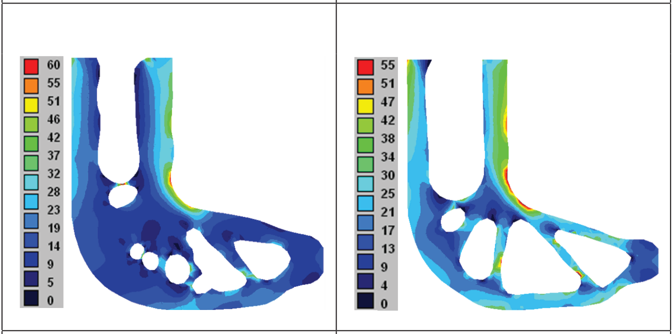


Figure 7.19: Evolution of f_U for L-beam.

a smoother surface in the subsequent iterations. This results in an optimal design with a maximum von Mises stress equal to 55. It is thus a feature of the approach that the use of NURBS automatically smooths the geometry enhancing the convergence towards a smooth optimum.



Iteration 0 Iteration 20



Iteration 47 Iteration 63

Figure 7.20: Evolution of von Mises stress contours for L-beam

7.4.4 Example-4

In the final example of this study we apply the proposed optimisation to Michell type structure. The geometry is depicted in Figure 7.21 with an aspect ratio of 1.5:1 [49, 72]. The structure is constrained around the circular hole in the structure, and a load $P = 100N$ is applied in the downward direction at the middle of the right edge of the beam. The level set domain is discretised with 38×26 square cells. The various factors used in this example are: $RR = 0.01$, $RR_i = 0.01$ and the optimisation process terminates when $\alpha = 0.5$.

During the optimisation process the evolution of the structural geometry is depicted in Figure 7.22 with the two hole insertion criterion, i.e. A and B, respectively. The hole insertion factors used in this example are those used in Example-3. Comparison of the results presented in Figure 7.22 shows the same behaviour during the optimisation process as observed in the previous examples. The optimal design generated with the two insertion criteria are similar and also very close to those available in the literature

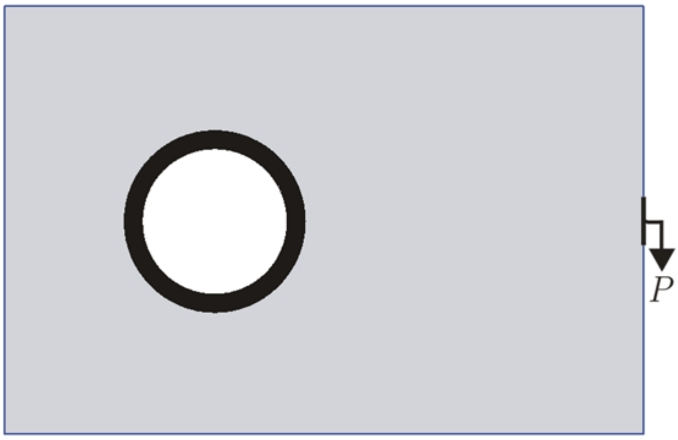


Figure 7.21: Design domain, loading and boundary conditions for Michell type structure.

[49, 72].

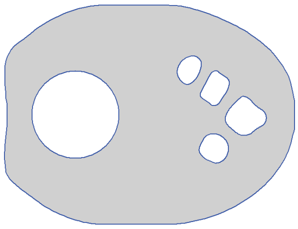
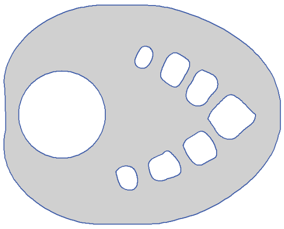
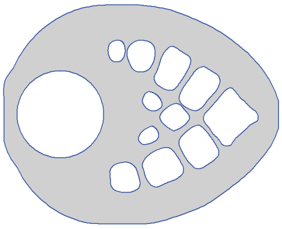
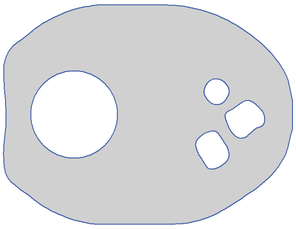
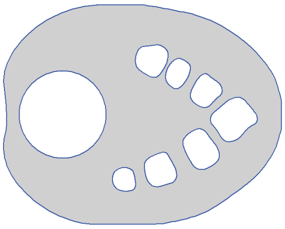
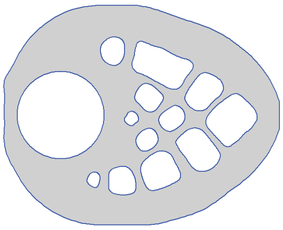
	$\alpha = 0.70$	$\alpha = 0.60$	$\alpha = 0.50$
A			
B			

Figure 7.22: Evolution history for Michell type structure.

The evolution history of f_U presented in Figure 7.23 shows similar behaviour to that observed in the previous examples. The evolution of f_U with the two hole insertion criteria are broadly coincident with each other throughout the optimisation process. Up to $\alpha = 0.70$, only boundary movements take place without any hole insertion. The value of f_U drops rapidly when the hole insertion starts in the design domain around $\alpha \approx 0.75$ which is then followed by a slow decrease in f_U until $\alpha = 0.50$ is reached.

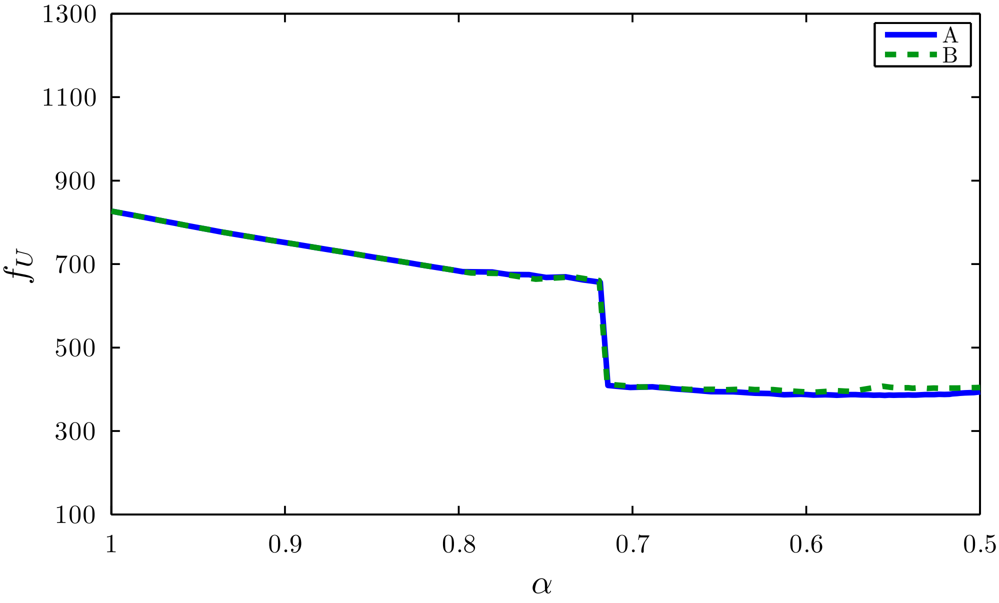


Figure 7.23: Evolution of f_U for Michell type structure.

7.5 Conclusions

In this study, the optimisation method presented in Section 5.2 has been used to study the effect of different hole insertion criteria in a BEM and LSM based structural optimisation approach. The research work presented to date using BEM and LSM based optimisation methods are dependent on initially guessed topologies. This optimisation method does not rely on an initially guessed topology. Instead two different criteria have been used to automatically insert holes during the optimisation process. The interesting correlation found between the two hole insertion criteria has been tested for four different benchmark examples. The results presented for these examples show (i) a close resemblance to optima published in the literature for those cases (ii) the robustness of the proposed optimisation method, and (iii) validation of the correlation between the two hole insertion criteria. This final result is important because it shows that an optimisation scheme driven by simple stress evaluations is able to produce an optimum, for stiffness-based optimisation problems, that is very strongly correlated, in both geometry and topology, with the optimum determined by schemes based on the calculation of design sensitivities.

Chapter 8

The use of sensitivities in a BEM and LSM based topology optimisation

8.1 Overview

Most of the level set based optimisation methods are based on the shape sensitivity formulations as discussed in Section 2.4.2. The shape sensitivities in most of these methods are computed through the FEM based " Ersatz material " approach [8] or X-FEM [12]. However, there are only two studies reported in the literature, i.e. [3, 136], in which the shape sensitivities are calculated with the BEM. In these methods, initial guess designs with pre-existing holes have only been considered for the solution of minimum compliance problems. Due to the absence of a hole nucleation mechanism in the proposed methods, the optimal designs are highly dependent on the initial guess designs. This Chapter presents an implementation of the use of sensitivities in a BEM, LSM and NURBS based optimisation method for minimum compliance problems. Further, in the present implementation, the topological derivative approach [19, 77] has been used as a criterion for hole insertion during the optimisation process. Therefore, the proposed method successfully overcomes the deficiencies associated with the previously presented BEM and LSM based optimisation methods [3, 136].

8.2 Shape sensitivity analysis

In structural optimisation different objective functions can be used to evaluate the performance of a given structure subject to constraints in the design variables. In this study the design objective function is to find the optimal topology of a structure with minimum compliance subject to a volume constraint. Consider a design domain Ω with a boundary Γ as shown in Figure 8.1. The boundary Γ is decomposed such that

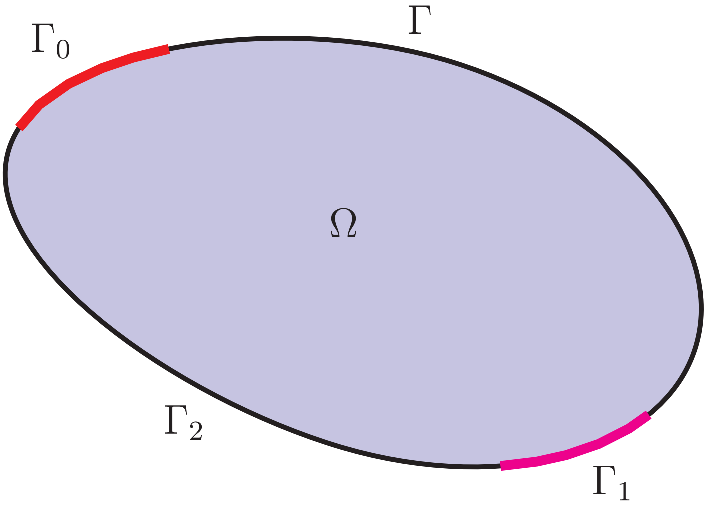


Figure 8.1: Design domain

$$\Gamma = \Gamma_0 \cup \Gamma_1 \cup \Gamma_2 \tag{8.1}$$

where Γ_0 corresponds to Dirichlet boundary conditions (where displacements are zeros), Γ_1 corresponds to non-homogeneous Neumann boundary conditions (where tractions are prescribed) and Γ_2 corresponds to homogeneous Neumann boundary conditions (traction free and is allowed to vary during the optimisation process). The objective function given in [109] can be written as

$$J(u) = \int_\Gamma \frac{1}{2} t_i u_i d\Gamma \tag{8.2}$$

where t_i is the traction and u_i is the displacement at a given node i. The optimisation problem can be expressed as finding Γ_2 to minimise $J(u)$, subject to the volume constraint

$$G = \int_{\Omega} d\Omega - V = 0 \tag{8.3}$$

where V is the target volume.

According to Soares and Choi [109], for linear material the first variation of objective function, (i.e. Equation (8.2)) becomes

$$\acute{J}(u) = -\int_{\Gamma_2} W v_n d\Gamma \tag{8.4}$$

where W is the strain energy density and v_n is the normal perturbation of the boundary. Similarly, the variation of the constraint functional (i.e. Equation (8.3)) given in [109] is

$$\acute{G} = \int_{\Gamma_2} v_n d\Gamma \tag{8.5}$$

Soares and Choi [109] used the Pshenichny linearisation method [91] of linear programming in combination with boundary element method to solve the optimisation problem.

Later on due to the development of the level set optimisation methods the above optimisation problem can be easily solved by embedding the structural geometry through an implicit function $\phi(\vec{x})$ such that the zero level set coincides with the structural boundary. This is mathematically represented through Equation 5.11.

Therefore, within the level set framework the compliance minimisation problem can be written as [8, 32, 126]

$$
\begin{aligned}
&\text{Minimise: } J(u,\phi) = \int_\Omega C\varepsilon(u)\varepsilon(u)H(\phi)d\Omega \\
&\text{Subject to: } \int_\Omega C\varepsilon(u)\varepsilon(v)H(\phi)d\Omega = \int_\Omega bvH(\phi)d\Omega + \int_\Gamma fvd\Gamma \\
&\qquad\qquad u|_{\Gamma_0} = 0 \quad \forall \in U \\
&\qquad\qquad G(\phi) = \int_\Omega H(\phi) - V = 0
\end{aligned}
\tag{8.6}
$$

where u is the displacement field, $\varepsilon(u)$ is strain field, C is the Hooke elasticity tensor, v denotes any permissible displacement field, U is the space of kinematically admissible displacement fields, $H(\phi)$ is the Heaviside function, b is the body force, and f represents the surface traction.

The optimisation problem can be solved with the Lagrange multiplier method [8, 32, 126] as:

$$\bar{J}(u,\phi) = J(u,\phi) + \ell G(\phi) \tag{8.7}$$

where ℓ is a positive Lagrange multiplier. The Karush-Kuhn-Tucker (KKT) optimality conditions require that for an optimal solution (or for a minimiser), the following conditions must be satisfied.

$$\acute{\bar{J}}(u,\phi) = 0 \tag{8.8}$$

In the work of Allaire *et al.* [8] the variation of the objective and volume constraint functions derived in terms of the level set are:

$$\acute{J}(u,\phi) = -\int_{\Gamma_2} Wv_n d\Gamma \tag{8.9}$$

$$\acute{G}(\phi) = \int_{\Gamma_2} v_n d\Gamma \tag{8.10}$$

where the negative sign in Equation 8.9 is used for the descent direction. Finally, the variation of the Lagrangian can be written as

$$\acute{\bar{J}}(u, \phi) = \int_{\Gamma_2} (\ell - W) v_n d\Gamma \tag{8.11}$$

The above formulation indicates that the shape derivatives can be easily obtained with surface integration. However, the level set method depends only on the normal velocity v_n and the calculation of surface integration is unnecessary [126]. Therefore, the velocity function v_n can be easily obtained as [8, 32, 126]

$$v_n = W - \ell \tag{8.12}$$

In the present study the boundary velocity can be accurately and efficiently calculated using the boundary element method. Furthermore, the BEM requires boundary elements on the level set boundary and avoids approximation at the boundary, which is the case for the fixed grid type approaches usually employed [8, 32, 126].

The propagation of the structural boundary during the optimisation process can be linked with the evolution of the function ϕ as an initial value problem. This means that the position of the structural boundary at any time t is given by the zero level set function ϕ. The structural boundary is evolved with the solution of the Hamilton-Jacobi equation [78]

$$\frac{\partial \phi}{\partial t} + v_n |\nabla \phi| = 0 \tag{8.13}$$

The normal velocity v_n is used as the advection velocity in the above formulation.

8.3 Optimisation algorithm

The sensitivity, BEM and LSM based topology optimisation method is implemented with two different approaches, as depicted in Figures 8.2 and 8.3, respectively. In the first approach, the optimisation method uses an initial guess topology with pre-existing holes, and there is no hole insertion during the optimisation process. An advantage associated with this approach is that it completely eliminates the use of internal points in the design domain and in this way the computational efficiency can be increased. In the second approach hole insertion takes place automatically through the topological derivative approach (as discussed in Section 7.2.2) during the optimisation process and an initial guess topology with or without pre-existing holes can be used.

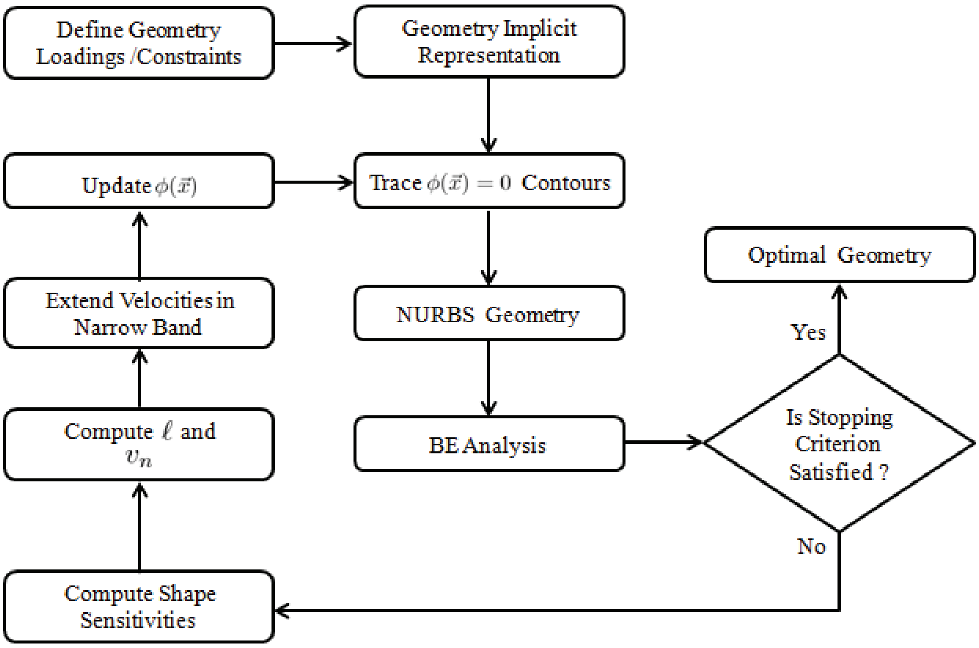


Figure 8.2: Optimisation flow chart 1

Once the initial geometry is defined the level set grid is initialised with a signed distance function as discussed in Section 5.3.2. Similarly, the zero level set contour tracing and NURBS fitting procedure is adopted in the same way as discussed in Section 5.3.3 and 5.3.4, respectively. At each iteration the BE analysis is carried out for the modified NURBS geometry and is followed by a stopping criterion check.

In the next step, holes are inserted in the design domain in accordance with Figure 8.3, and the structure is re-analysed after each hole insertion. In the previous hole insertion implementations discussed in Sections 6.2.1 and 7.3, there was some randomness in the algorithm that distributes internal points in the design domain. However, in the present case, level set grid points with $\phi(\vec{x}) < 0$ are used as internal points and hence this provides a regular grid of these points. Once there is no hole insertion the shape sensitivities are

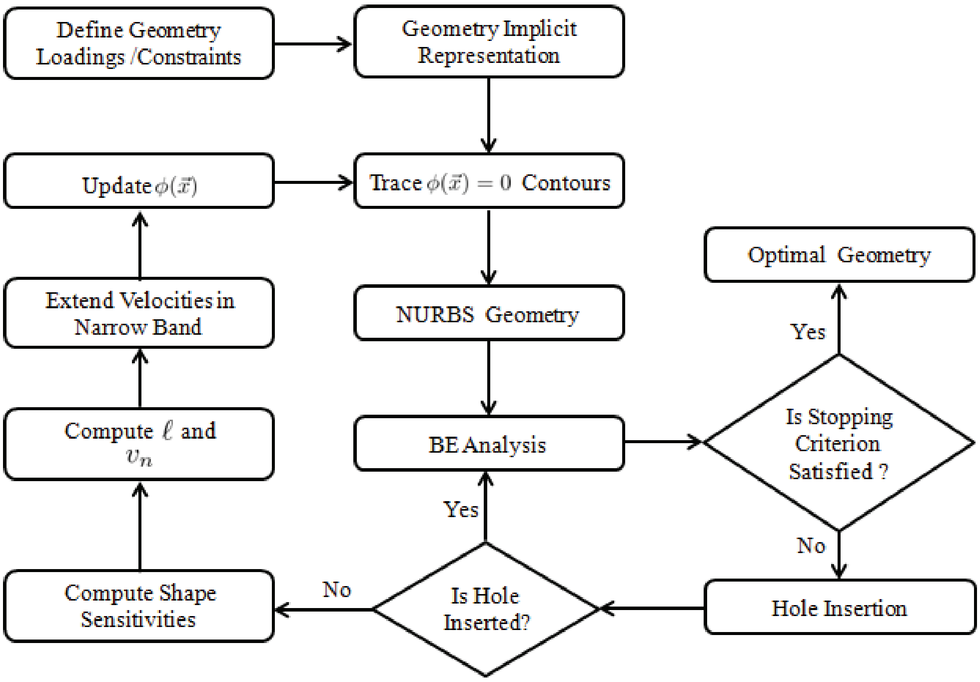


Figure 8.3: Optimisation flow chart 2

calculated at the structural boundary.

In order to evolve the structural geometry, shape sensitivities should be converted into boundary velocities using Equation (8.12). However, a necessary condition for the solution of (8.12) requires the Lagrange multiplier ℓ to be known in advance. In the present implementation, the bisectioning algorithm [126] discussed in Section 8.4.1 is used for the calculation of ℓ.

The velocity calculated in the previous step is extended to grid points in the narrow band around the zero level sets using the method developed in [4, 5]. At each iteration the level set function ϕ is updated through the solution of the Hamilton-Jacobi equation (i.e. Equation 8.13).

Details of the various components shown in Figures 8.2 and 8.3 have already been discussed in the previous Chapters, and the following section is devoted to the implementation details of the bisectioning algorithm.

8.3.1 Bisectioning algorithm

In order to solve Equation (8.12) for the shape derivatives the value of the Lagrange multiplier ℓ needs to be calculated. Allaire *et al.* [8] and Wang and Wang [125] used a fixed value for ℓ. In the literature different methods have been proposed (e.g. [32, 79, 122, 124]) for the calculation of ℓ which should satisfy the volume constraint during the optimisation process. Similar to the SIMP method [107] Wang *et al.* [126] implemented bisectioning algorithm for the calculation of ℓ which exactly satisfies the volume constraint during an LSM based optimisation process. An approach similar to the one used by Wang *et al.* [126] has also been used in the present study to calculate the ℓ for the solution of Equation (8.12) at each optimisation iteration.

During the optimisation process, the material volume is a monotonically decreasing function of ℓ. Using Equation (8.12), the shape derivative for the volume constraint can be re-written as [126]

$$\acute{G}(\phi) = \int_{\Gamma_2} (W - \ell) d\Gamma \tag{8.14}$$

Equation (8.14) shows that the value of $\acute{G}(\phi)$ increases with a low value of ℓ and decreases with a higher one. In other words, two different values, i.e. ℓ_1 and ℓ_2 can be used to set an upper and lower bound for ℓ. In the present implementation the bisectioning algorithm is initialised with $\ell_1 = 0$, and $\ell_2 = 500$. This suggests that with only ℓ_1, v_n will be positive, and the structural boundary will move in the outward direction, and this will increase the volume. Similarly, with ℓ_2, v_n will become negative and the structural boundary will move inward, and the volume will be decreased. The interval between ℓ_1 and ℓ_2 is repeatedly halved at a given iteration until it satisfies the convergence criterion [107]. The implementation details of the bisectioning algorithm is given below.

1. Initialise ℓ_1 and ℓ_2
2. Set $\bar{\phi} = \phi$

3. Halve the interval, i.e.

$$\ell = (\ell_1 + \ell_2)/2 \tag{8.15}$$

4. Calculate v_n using Equation (8.12)
5. Extend velocities to the grid points around the narrow band
6. Update the level set function, i.e.

$$\frac{\partial \bar{\phi}}{\partial t} + v_n |\nabla \bar{\phi}| = 0 \tag{8.16}$$

7. Trace the zero level set contours
8. Calculate the new volume
9. if $G(\bar{\phi}) > 0$ $\ell_1 = \ell$, otherwise $\ell_2 = \ell$
10. Terminate if $|\ell_2 - \ell_1| \leq 10^{-2}$, otherwise go to step 2.
11. Set $\phi = \bar{\phi}$

During the optimisation iterations, the above algorithm is used for the calculation of ℓ which exactly satisfies the volume constraint. Wang *et al.* [126] proposed that the normal velocities calculation at a constant volume act as mass conservative velocities and hence, this level set method can be generally considered as a mass conservative.

In the numerical implementation, it has been observed and also reported by [126], that if the volume of the initial design domain is far away from the target volume, then the optimisation method may produce some undesirable results. Therefore, in those situations, as proposed by [126] two different approaches have been used during the numerical implementation. In the first approach, the values of ℓ_1 and ℓ_2 (which bound ℓ) are selected in such a way, that in each case, a smooth progression towards the optimal design has been achieved. In the second approach, a fixed value of ℓ is used near to the target volume and the bisectioning algorithm is used afterwards, which calculates the correct ℓ which exactly satisfies the volume constraint.

8.4 Examples

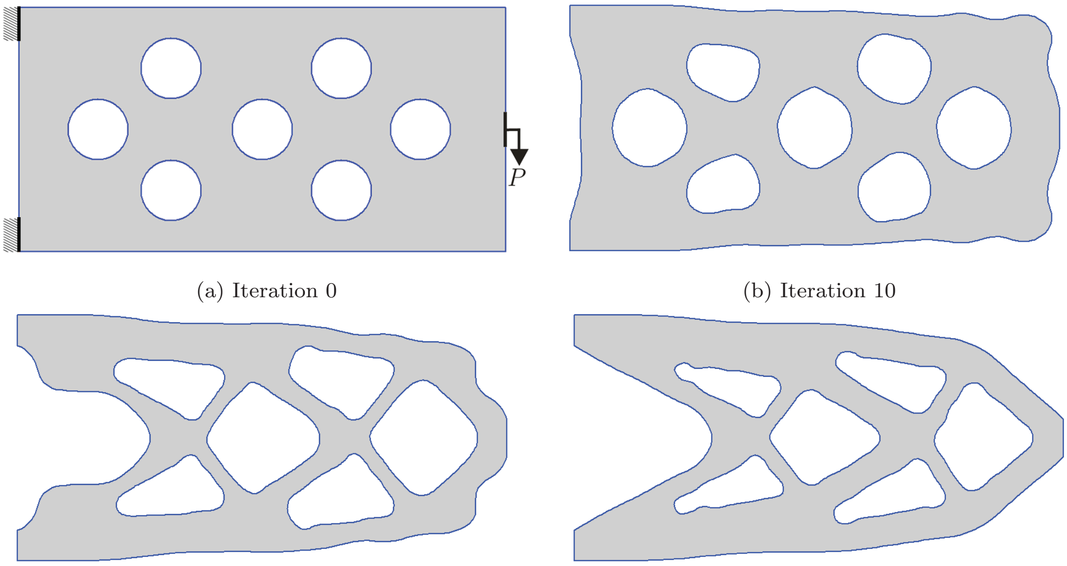
The validity and efficiency of the sensitivity based optimisation method is tested against some benchmarking problems in the field of structural optimisation. The material properties used in these examples are: Poisson's ratio = 0.3, Young's modulus = 210 GPa, Yield stress = 280 MPa. Plane stress conditions are assumed with arbitrary thickness of 1 mm. All examples are solved with a load $P = 100$ N. A time step size $dt = 0.0005$ is used in the present implementation. The optimisation terminates when the relative difference between the three successive iterations is less than 10^{-2} or when the given maximum number of iterations has been reached.

8.4.1 Example-1

In the first case of this example a cantilever beam with an aspect ratio of 2:1 is considered. The initial design with an applied load and boundary conditions is shown in Figure 8.4(a). The structure is constrained at the top and bottom of the right hand edge with zero displacement boundary conditions and the load P is applied at the middle of the right hand edge. The traction free boundary is represented by NURBS, and is allowed to vary during the optimisation process. The minimum compliance problem is solved for a target volume $V = 0.5V_0$.

In order to capture all possible boundary movements a fixed level set domain is used during the numerical implementation with size larger than the initial design. The level set domain is discretised into 80×40 square cells. The volume of the initial design domain is $0.87V_0$, which is far away from the target volume. Therefore, in this example, three different approaches are used for the solution of the minimum compliance problem. The implementation details of all these cases are discussed below and in addition, a comparison has also been made at the end.

In the first case, the bisectioning algorithm is used from the start of the optimisation process, and the results obtained are depicted in Figure 8.4(b)-(d). During the optimisation process, the structural boundary evolves into an optimal design through boundary movements and automatic merging of the hole with the outer boundary. It can be seen from the results obtained at iteration 32 and 125, that the present shape optimisation method efficiently redistributes material in the design domain such that the compliance is minimised at constant volume (as shown in Figure 8.5). The optimal geometry obtained is similar to those available in the literature [8, 125, 126].



(a) Iteration 0

(b) Iteration 10

(c) Iteration 32

(d) Iteration 125

Figure 8.4: Evolution of structural geometry for Example-1, first case

Figure 8.5 shows the convergence histories of the objective and the volume during the optimisation iterations. It can be seen that in the initial 22 iterations, the values used for ℓ_1 and ℓ_2 (which bounds ℓ) provide a smooth progression of the structural geometry. Afterwards, the structural volume reaches near to the target volume and at each iteration, the bisectioning algorithm calculates the value of ℓ which exactly satisfies the volume constraint. The evolution of objective function shows that due to a higher material volume, the compliance of the initial design is 1.22 and the material removal increases this value to 1.78 until the volume constraint is exactly satisfied. In the subsequent iterations, the compliance is gradually minimised at constant volume and the optimisation process terminates with a final compliance of 1.52 where the stopping criterion is satisfied.

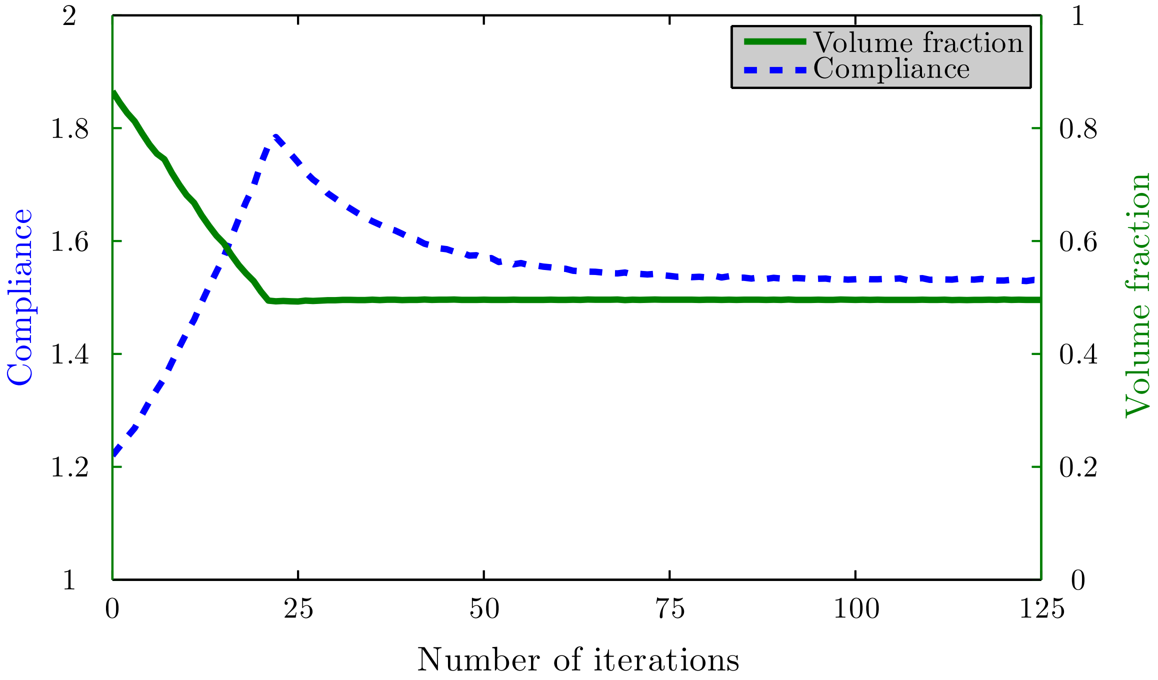
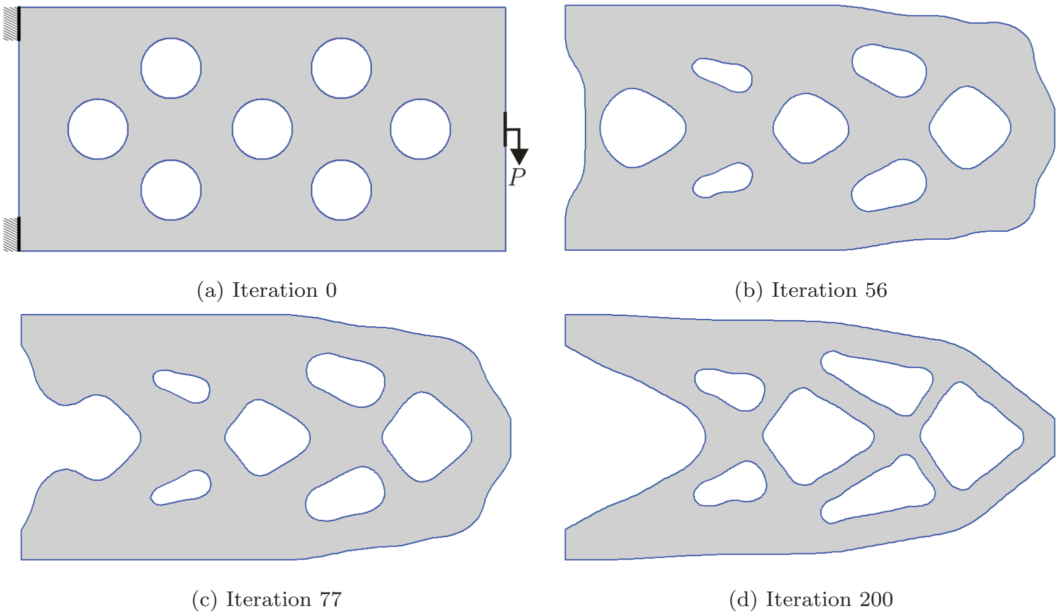


Figure 8.5: Convergence of objective function and volume for Example-1, first case

In the first case, the target volume fraction is far away from the current volume fraction, and the use of bisectioning algorithm removes more material in the initial iterations results in a rapid increase in the objective function. In such situations Wang *et al.* [126] proposed the use of a fixed ℓ near to the target volume fraction and bisectioning algorithm afterwards to calculate the exact Lagrange multiplier. Similar approach has also been used in References [8, 124]. Therefore, in the second case of this example, a fixed value of $\ell = 120$ is pre-specified up to $0.65V_0$ and the bisectioning algorithm is used afterwards which calculates ℓ that exactly satisfies the volume constraint. The evolutions of the structural geometry during the optimisation iterations are illustrated in Figure 8.6(b)-(d). The evolution of the structural geometry is analogous with the previous case, and the optimisation process comprised of the boundary movements and automatic hole merging. The results obtained at iteration 56 and 77 show that some of the holes expand while others shrink with a fixed value of ℓ. In the subsequent iterations, the bisectioning algorithm is used and the exact ℓ is calculated at each iteration. As a result the material within the design domain is redistributed at constant volume and this evolves the structural geometry into an optimal design.



(a) Iteration 0

(b) Iteration 56

(c) Iteration 77

(d) Iteration 200

Figure 8.6: Evolution of structural geometry for Example-1, second case

The variation of the objective function and the volume fraction at each optimisation iteration are displayed in Figure 8.7. In the initial iterations, the movements of the external boundary and the expansion and contraction of the internal holes give rise to a slow decrease in the volume, and this results into a decrease in the compliance from 1.22 to 1.14. In the following iterations, the compliance increases gradually until the structural volume is reduced to $0.65V_0$. Afterwards, the bisectioning algorithm is used to calculate ℓ which exactly satisfies the volume constraint, and this resulted into a rapid reduction in the volume and a corresponding increase in the compliance of the structure from 1.2 to 1.54. Once the volume constraint is satisfied, the topological changes take place in the subsequent iterations, and this gradually decreases the compliance to 1.48 and remains stable until the stopping criterion is satisfied.

In the last case of this example, the optimisation process starts from an initial guess topology as used in the previous two cases; in addition, holes are automatically inserted at the low strain energy regions using the topological derivative approach (discussed in Section 7.3) with $D_T = 3.5$. The optimisation problem for the minimum compliance is solved with the same approach as used in the second case of this example. A combination

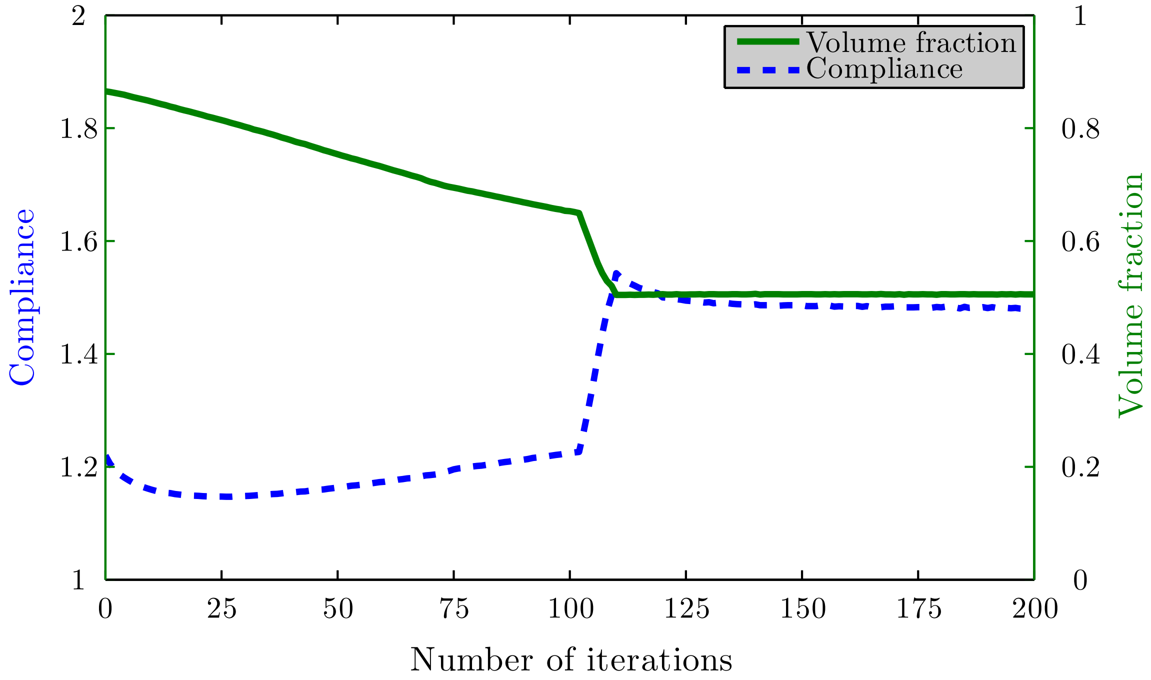
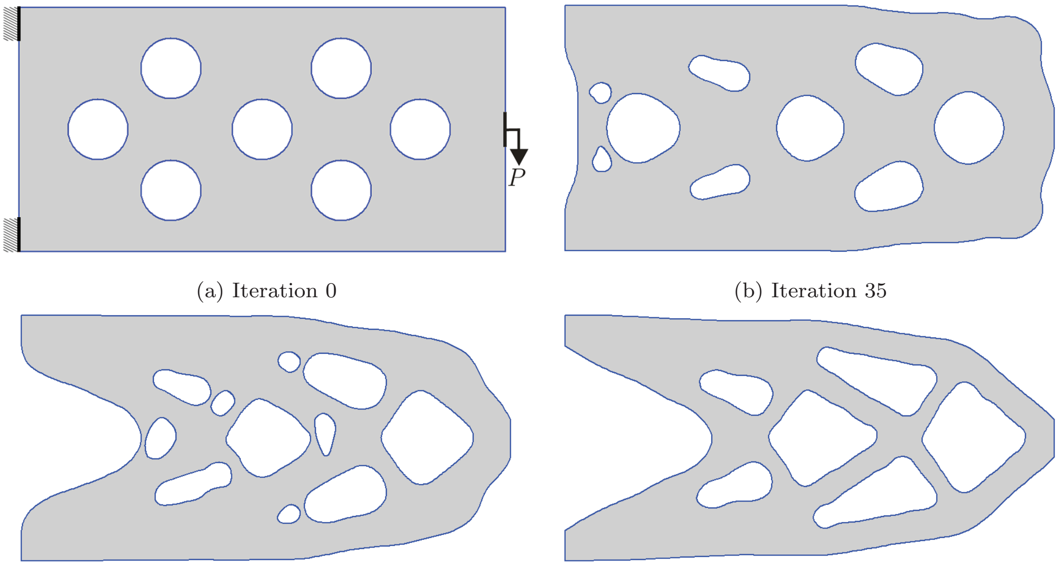


Figure 8.7: Convergence of objective function and volume for Example-1, second case

of fixed and exact of ℓ is used in the same way as discussed in the previous case. Figure 8.8 displays the structural topologies during the optimisation iterations. It can be seen that the initial geometry evolves into an optimal geometry through boundary movements, holes insertion and holes merging with each other and with the external boundary.

The evolution history of the objective function and volume fraction are depicted in Figure 8.9. This demonstrates that both functions follow similar trends in the initial iterations as observed in the previous case. Moreover, the hole insertion allows more material removal with a fixed ℓ and consequently, the specified volume for the bisectioning method reached in fewer iterations (i.e. at 90) than that in the previous case (i.e. at 101). The optimisation process terminates at a value of 1.48 when there is no further improvement in the objective function.

A comparison of the above three cases shows that in the first case the volume constraint is satisfied at iteration 22, and the objective function stabilises after iteration 88 with a final value of 1.52. In case two, the objective function stabilises at a value of 1.48 between iteration 110 and 140, and it takes 14 iterations (i.e. from 93 to 107) to stabilise with a



(a) Iteration 0

(b) Iteration 35

(c) Iteration 83

(d) Iteration 135

Figure 8.8: Evolution of structural geometry for Example-1, third case

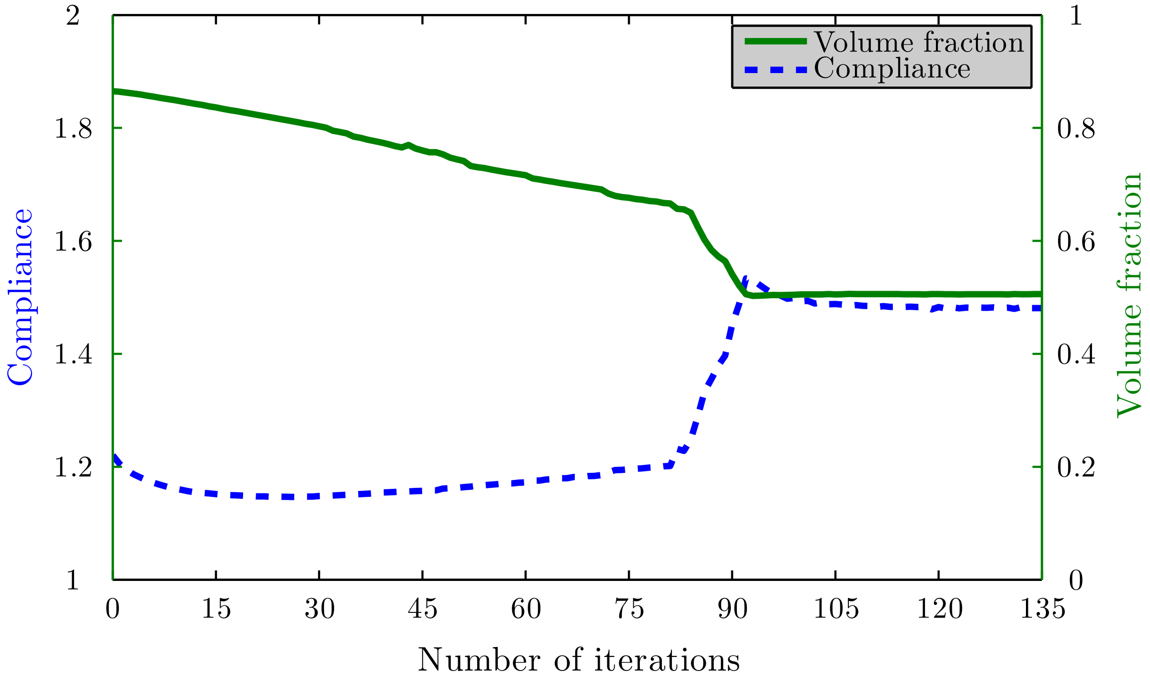
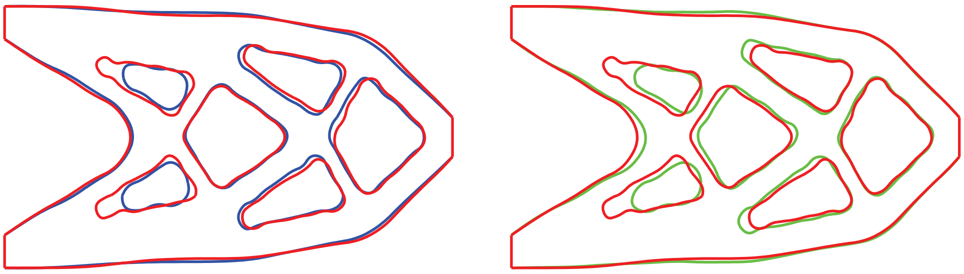


Figure 8.9: Convergence of objective function and volume for Example-1, third case

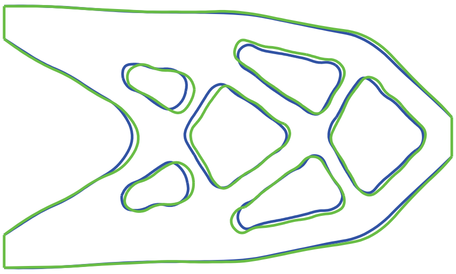
value of 1.48 in the last case. In the initial two cases only shape optimisation is carried out and there was no hole insertion during the optimisation process, and this completely eliminates the use of internal points. However, internal points are always required for hole insertion, and this makes the last approach computationally more expensive than the other two. The final objective function values are compared in Table 8.1. In addition, Figure 8.10 shows a comparison of the material distribution within all three cases. This comparison demonstrates that the positions of the internal shapes are approximately identical in all three cases, though their shapes vary slightly. In order to completely investigate the combination of BEM, LSM, NURBS and shape sensitivity analysis this method is further tested with different initial designs and boundary conditions in the following examples.

Case	Total iterations	Final compliance
1	125	1.52
2	200	1.48
3	135	1.48

Table 8.1: Comparison of the objective function



(a) Red (Case 1) and Blue (Case 2) (b) Red (Case 1) and Green (Case 3)



(c) Blue (Case 2) and Green (Case 3)

Figure 8.10: Comparison of final optima in all three cases for Example-1

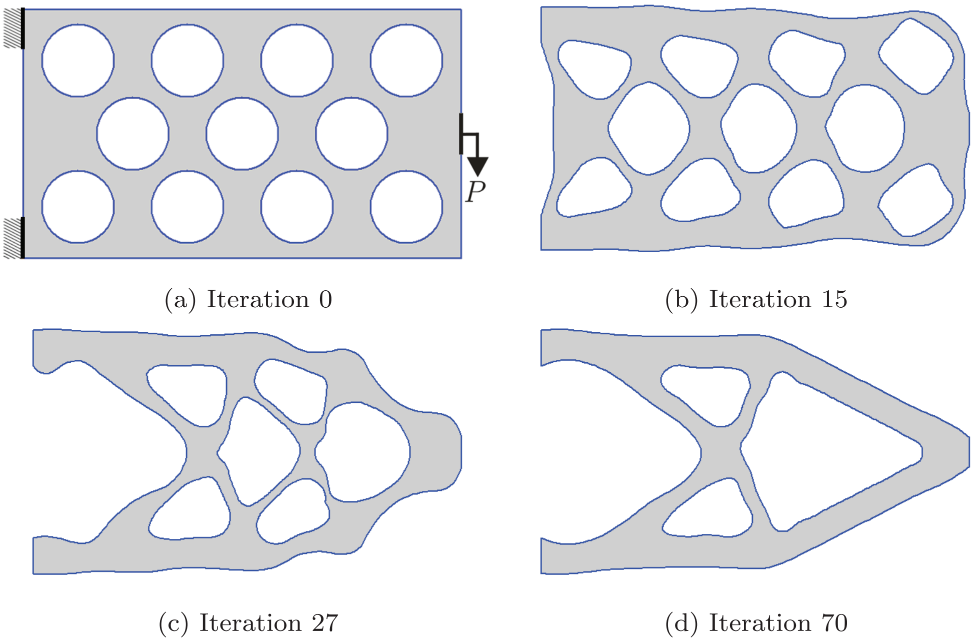
8.4.2 Example-2

In the second example, the minimum compliance problem is solved for a cantilever beam with an aspect ratio of 1.6:1 with different initial guessed designs. The zero displacement boundary conditions are prescribed at the top and bottom portions of the left hand edge and the structure is loaded at the middle of the right edge as shown in Figure 8.11(a). The specified target volume fraction for this example is $V = 0.35V_0$. The level set design domain is discretised into 60×34 square cells.

In this example, two different initial guesses are considered for the solution of the minimum compliance problem. These initial guessed designs are: cantilever beam with and without pre-existing holes. During the optimisation process, pre-existing holes are used with the first initial design, whereas hole insertion takes place automatically with the second one. Furthermore, the initial guess design with pre-existing holes is solved with an exact and a combination of fixed and exact ℓ.

In the first case, an exact ℓ is calculated throughout the optimisation process with the bisectioning algorithm for an initial guessed design as depicted in Figure 8.11(a). The intermediate results during the optimisation process are given in Figure 8.11(b)-(c) and the resulting optimal design with smooth boundary is shown in 8.11(d). The results obtained closely match to optima published in the literature [136] and demonstrates that both shape and topology optimisation take place simultaneously with the use of an implicit representation.

Figure 8.12 shows convergence histories of the objective function and volume fraction throughout the optimisation iterations. It can be seen that in the initial 5 iterations, the volume decreases slowly and as a result the objective function is lowered from 1.48 to 1.41. During these iterations, the velocity value obtained violated the CFL condition with the initial time step value used and therefore, the time step is lowered which resulted into a slow decrease in the volume . In the following iterations, the compliance rises to a maximum value of 2.01 at iteration 31 and then it decreases continuously until the volume constraint is exactly satisfied at iteration 35. Afterwards, the objective function remains



(a) Iteration 0 (b) Iteration 15

(c) Iteration 27 (d) Iteration 70

Figure 8.11: Evolution of structural geometry for Example-2, first case

stable with a value of 1.60 at iteration 54 until the stopping criterion is satisfied.

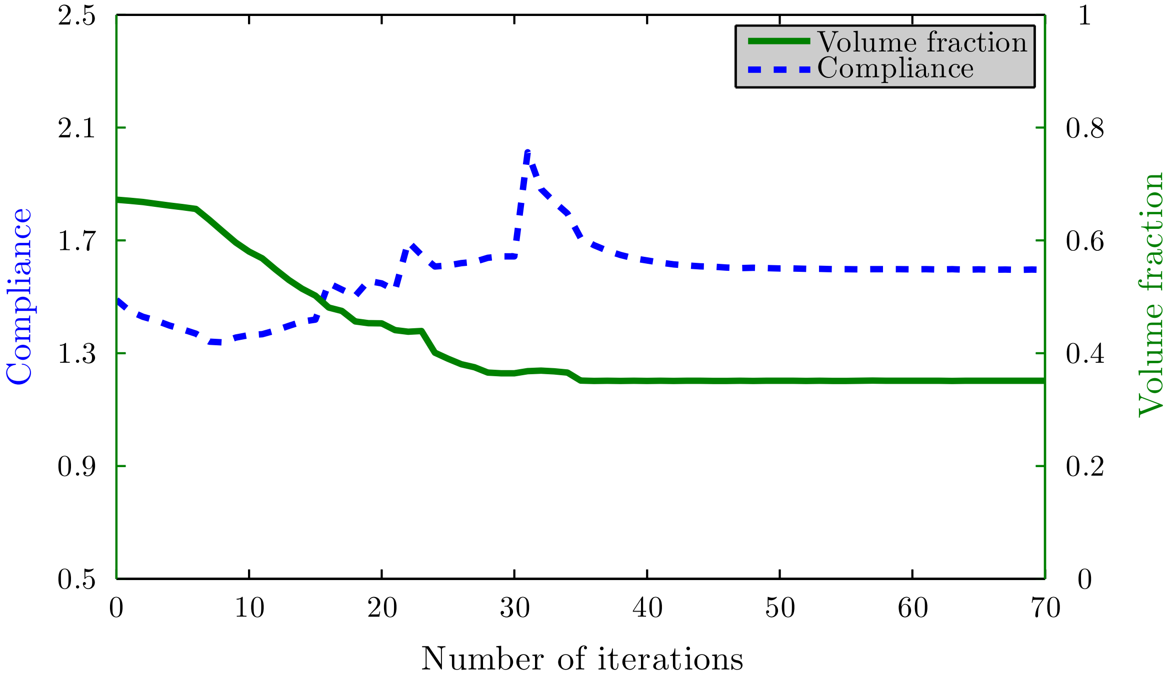
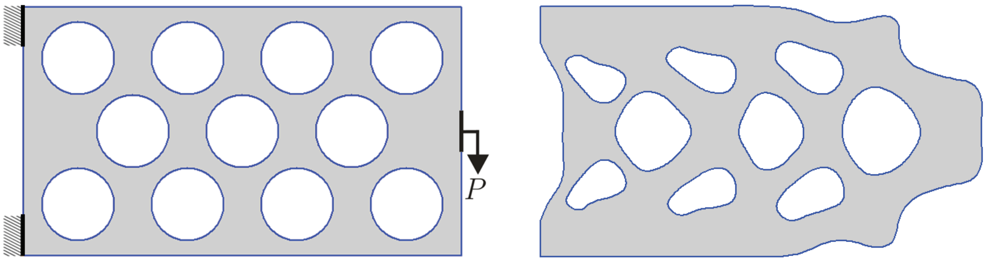
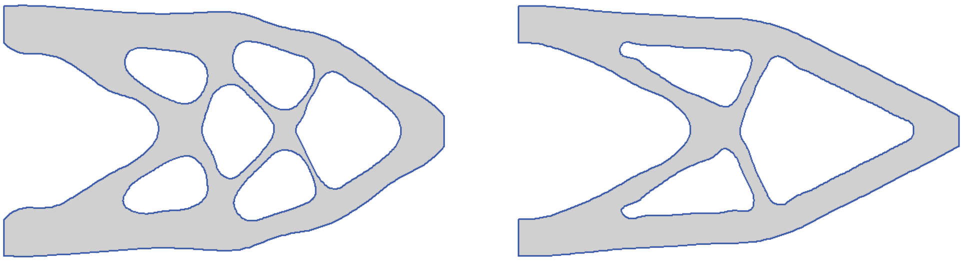


Figure 8.12: Convergence of objective function and volume for Example-2, first case

The initial design used in the second case of this example is the same as that used in the first case (i.e. Figure 8.13(a)) however, a combination of fixed and exact ℓ is used for this case. A fixed $\ell = 170$ is used up to $0.60V_0$ and afterwards, bisectioning algorithm is used for the calculation of exact ℓ. The use of fixed ℓ for a given volume is based on the results of various numerical tests conducted. The intermediate results during the optimisation process and the final optimal design are shown in Figure 8.13(b)-(d).



(a) Iteration 0 (b) Iteration 39



(c) Iteration 66 (d) Iteration 150

Figure 8.13: Evolution of structural geometry for Example-2, second case

Figure 8.14 displays the convergence histories of the objective function and volume fraction throughout the optimisation iterations. It can be seen that up to iteration 13, the fixed ℓ provides shrinkage and expansion of the pre-existing holes at the same time, and this causes a net increase in the structure volume from $0.67V_0$ to $0.70V_0$. This reduces the objective function from an initial value of 1.48 to 1.40. The subsequent shrinkage and expansion of the pre-existing holes continue, and this results into a slow decrease in the volume. On the other hand, the objective function is further reduced to 0.97 at iteration 58. At this stage, the volume reaches to the pre-specified value (i.e. $0.60V_0$) and the fixed ℓ is replaced by the exact one computed via the bisectioning algorithm. This causes a rapid volume decrease, and the objective function rises to 1.92 at iteration 68. In the following iterations, due to high velocity value the time step size is lowered and this resulted in a very slow increase in the structural volume and a corresponding reduction in the objective function. Afterwards, the volume constraint is exactly satisfied, and through shape optimisation, the objective function is reduced from 1.92 to 1.54 and remains stable until the stopping criterion is satisfied.

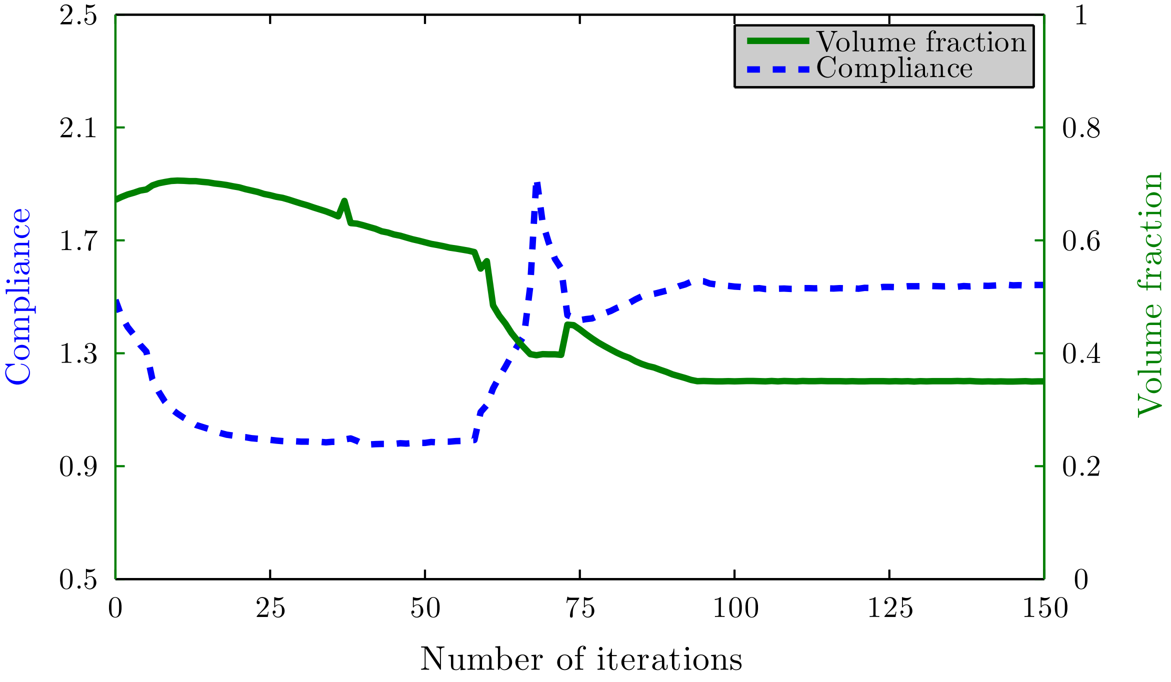
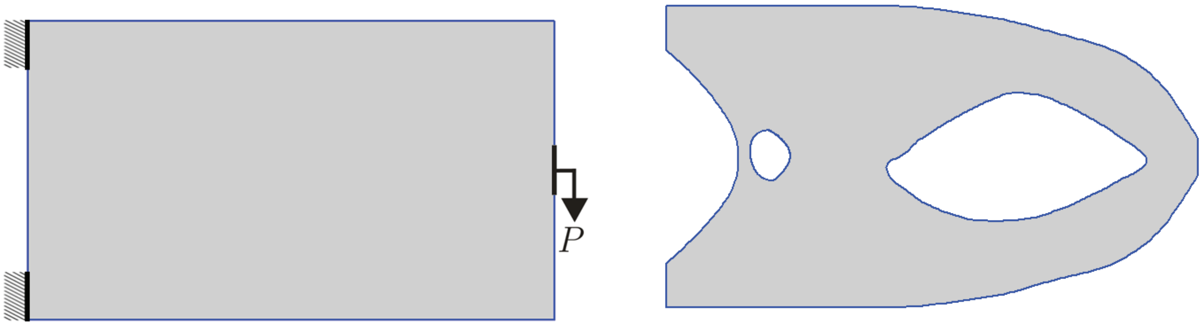
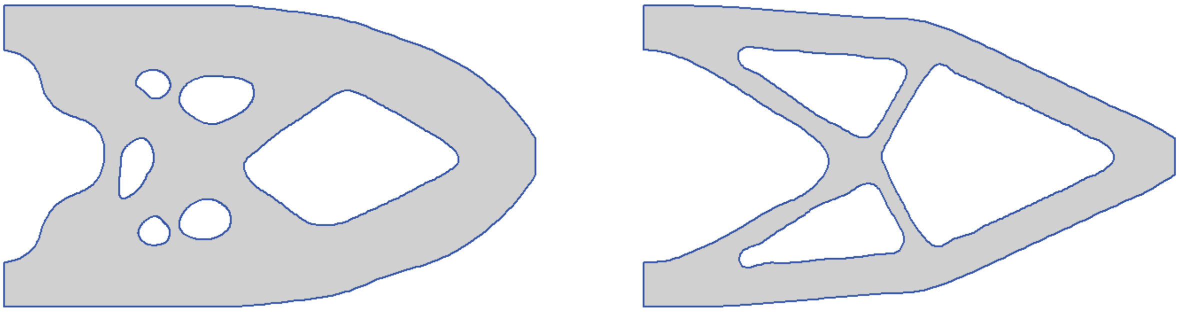


Figure 8.14: Convergence of objective function and volume for Example-2, second case

In the final case of this example, there are no pre-existing holes in the initial design as shown in Figure 8.15(a). Instead, automatic holes insertion takes place in the design domain during the optimisation iterations using the topological derivative approach with $D_T = 2.6$. Similar to the previous case, a fixed ℓ is used up to a pre-specified volume $0.60V_0$ and in the following iterations, the bisectioning algorithm is used for an exact ℓ. It should be noted, that hole insertion only takes place with a fixed ℓ in the current implementation. Otherwise, a rapid volume reduction through bisectioning algorithm may not allow a sufficient number of holes during the optimisation process. The evolution of the structural geometry comprised of boundary movements, hole insertion and hole merging with the boundary and each other, and is depicted in Figure 8.15(b)-(c). The optimal design shown in Figure 8.15(d) is quite similar to the previous cases and those available in the literature, e.g. [133].



(a) Iteration 0 (b) Iteration 97



(c) Iteration 112 (d) Iteration 200

Figure 8.15: Evolution of structural geometry for Example-2, third case

Figure 8.16 displays the convergence histories of the objective function and volume fraction during the optimisation iterations. During the initial iteration, the fixed ℓ provides an approximately linear increase in the objective function and a corresponding decrease in the volume constraint. This continues until the first hole insertion takes place at iteration 92. The insertion of a large hole size raises the objective function from 0.77 to 1.16. In the following iterations, hole insertions continued to take place, and this causes slight fluctuation in the objective function. Once the pre-specified volume is achieved, the bisectioning algorithm is used in the following iteration for the calculation of exact ℓ, and this rapidly increases the objective function from 0.98 to 1.54 through iteration 118 to 147. Afterwards, the volume converges and slightly decreases the objective function to a value of 1.51 and remains stable until the stopping criterion is satisfied.

Finally, a comparison of the above three cases is presented in Table 8.2, which shows that the objective function of the last case is better than the remaining two cases. Like-

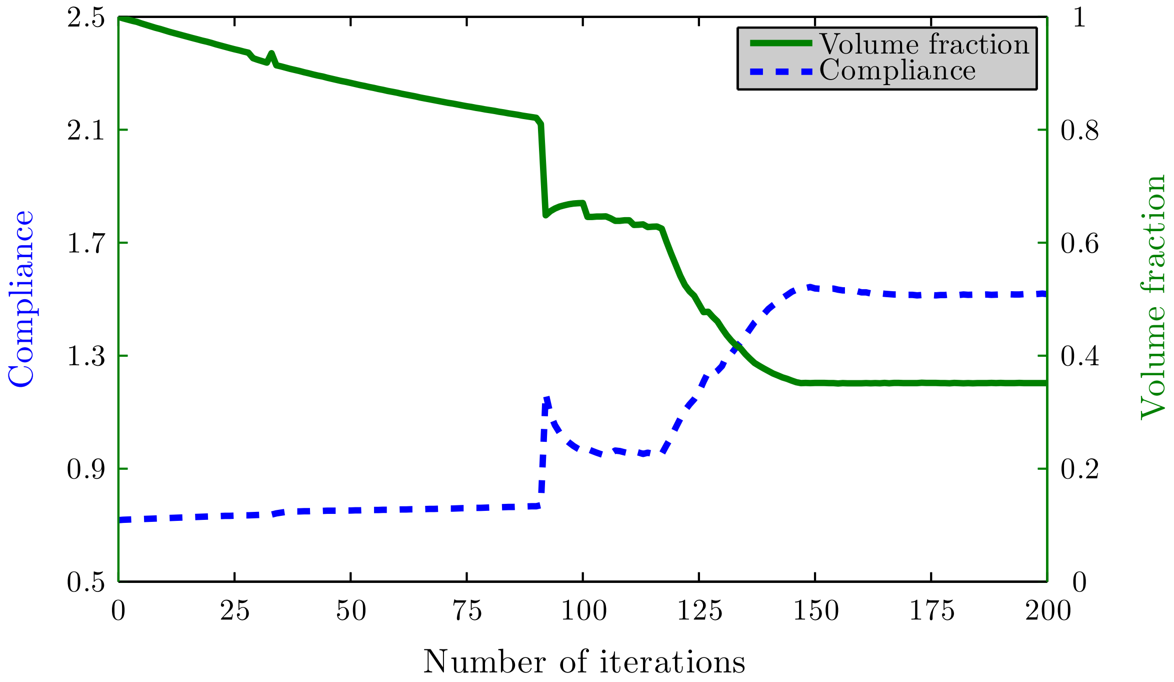


Figure 8.16: Convergence of objective function and volume for Example-2, third case

wise, the objective function of the second case is better than the first one. This comparison demonstrates that with the use of a fixed ℓ in the initial iteration provides a better final optimum. Furthermore, it can be seen that the final optimum of the third case is slightly better than the second one. However, the third case is computationally more expensive than the second one. Actually, the minimum compliance problem in the second case can be easily solved without any internal points, whereas those are necessary for hole insertion during the optimisation process.

Case	Total iterations	Final compliance
1	70	1.64
2	150	1.54
3	200	1.51

Table 8.2: Comparison of the objective function

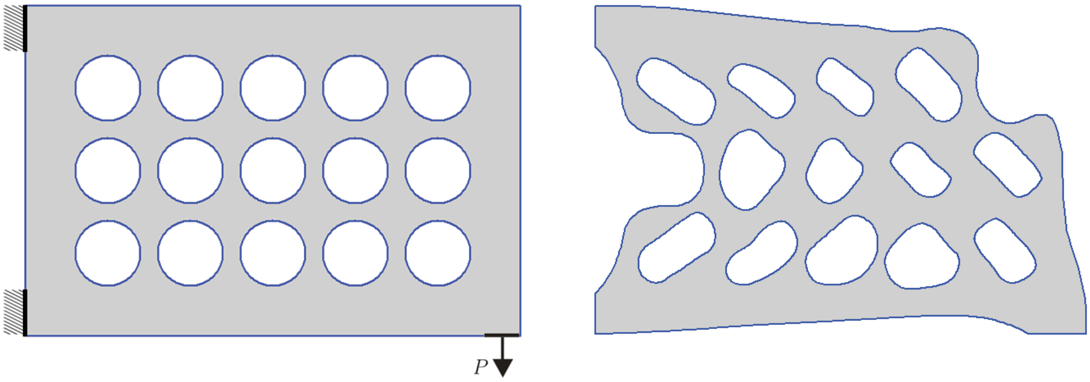
8.4.3 Example-3

The minimum compliance problem for the third example is solved for a cantilever beam with an aspect ratio of 1.5:1 with different initial guessed designs. The zero displacement boundary conditions are prescribed at the top and bottom portions of the left hand edge and the structure is loaded at the right of the bottom edge as shown in Figure 8.17(a). The specified target volume fraction for this example is $V = 0.35V_0$. The level set design domain is discretised into 60 × 40 square cells.

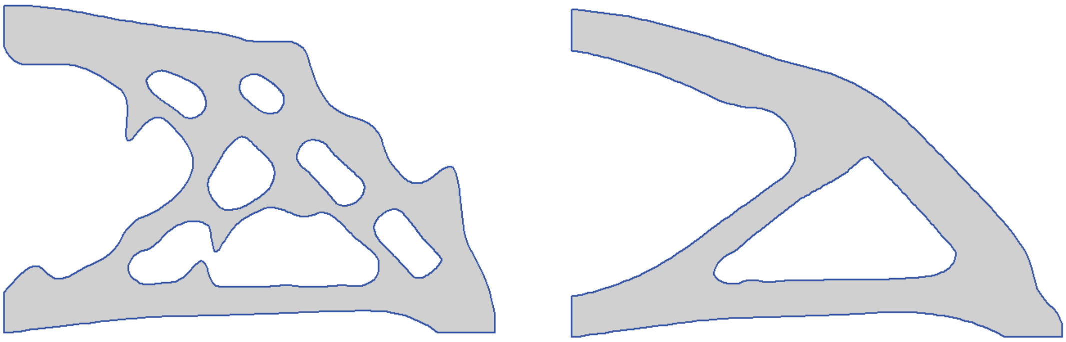
In this example, two different initial designs (i.e. with and without pre-existing holes) are used for the solution of the minimum compliance problem. Similar to the previous example (i.e. Example-2), the initial design with pre-existing holes is solved with an exact and a combination of fixed and exact ℓ. The second initial design is solved with hole insertion during the optimisation process and with the combination of fixed and exact ℓ approach. Therefore, three different cases are studied in this example and the results obtained in each case are discussed below.

In the first case, the optimisation process starts from an initial guess design as shown in Figure 8.17(a). The evolution of the structural geometry at some stages of the optimisation process and final optima are depicted in Figure 8.17(c)-(d). It is evident from the results displayed, that in addition to hole merging, some holes shrink and disappear automatically during the optimisation iterations.

The convergence histories of the objective function and the volume fraction are depicted in Figure 8.18. In the present case the bisectioning algorithm is used from the start of the optimisation process. This results into a rapid increase in the objective function (i.e. from 0.81 to 1.41) and a corresponding decrease in the volume. Once the volume constraint converges at iteration 25, the subsequent iterations, relatively large number, are used to carry out shape optimisation only and this results into a material re-distribution within the design domain. It can be seen that once the volume constraint is satisfied with pre-existing holes the objective function initially decreases and then stabilises at a value of 1.08 in the following iterations.



(a) Iteration 0 (b) Iteration 33



(c) Iteration 55 (d) Iteration 250

Figure 8.17: Evolution of structural geometry for Example-3, first case

The results obtained with a fixed and exact ℓ for an initial design are depicted in Figure 8.19. The fixed Lagrange multiplier used in this case is $\ell = 250$ up to a pre-specified volume of $0.50V_0$. Figure 8.20 presents the convergence histories of the objective function and the volume constraint throughout the optimisation process. The results demonstrate that in the initial iterations, the volume decreases slowly and results into a corresponding increase in the objective function. The maximum value of the objective function recorded in this case was 1.87, which is reduced to 1.03 and remains stable in the subsequent iterations.

In the third case, the optimisation process starts from a completely filled design domain as shown in Figure 8.21(a). Both shape and topology optimisation take place simultaneously during the solution of the minimum compliance problem. The hole insertion is based on the topological derivative approach with $D_T = 3.0$. The fixed and exact combination of ℓ is used during the optimisation process. A fixed value of the Lagrange multiplier, i.e.

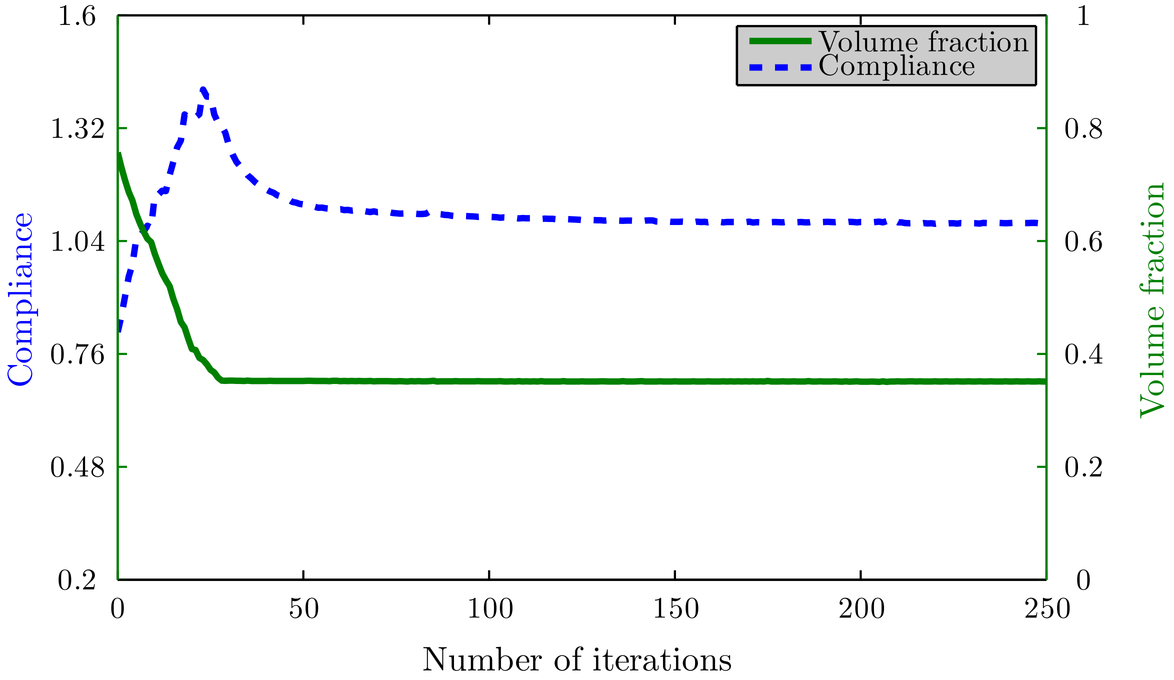
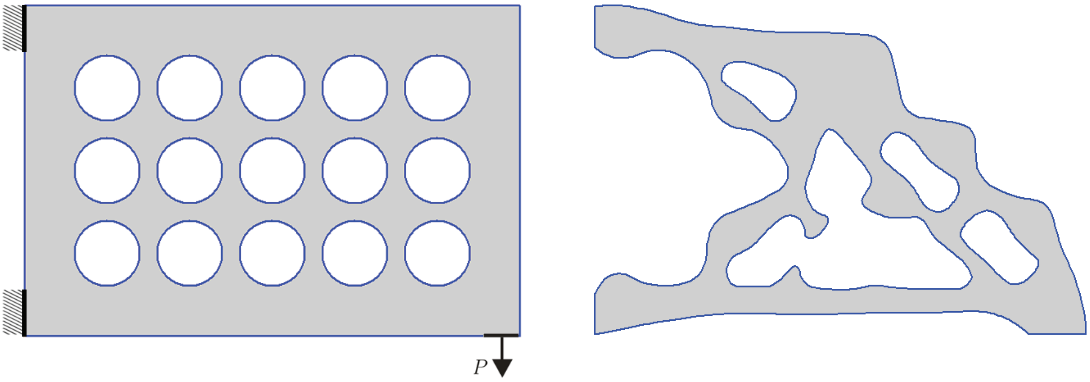


Figure 8.18: Convergence of objective function and volume for Example-3, first case

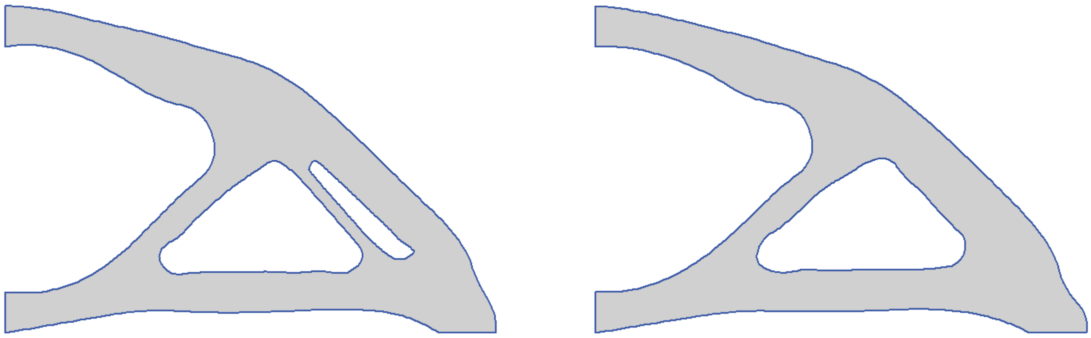
$\ell = 100$ is used from V_0 to $0.6V_0$ and the hole insertion is restricted within this interval. Afterwards, an exact ℓ is calculated using the bisectioning algorithm. The evolution of the structural geometry at various stages of the optimisation process is depicted in Figure 8.21.

The evolution histories of the objective function and volume fraction at each optimisation iteration are depicted in Figure 8.22. The optimisation process starts from a minimum value of the objective function, and then it slowly increases due to the boundary movements and holes insertion. The use of the bisectioning algorithm raises this value to 1.09 at iteration 216. Once the volume constraint is satisfied, the objective function is reduced to 1.05 and remains stable afterwards.

A comparison of the above three cases studied in this example further verifies the usefulness of the second case as observed in the previous examples.



(a) Iteration 0 (b) Iteration 26



(c) Iteration 150 (d) Iteration 250

Figure 8.19: Evolution of structural geometry for Example-3, second case

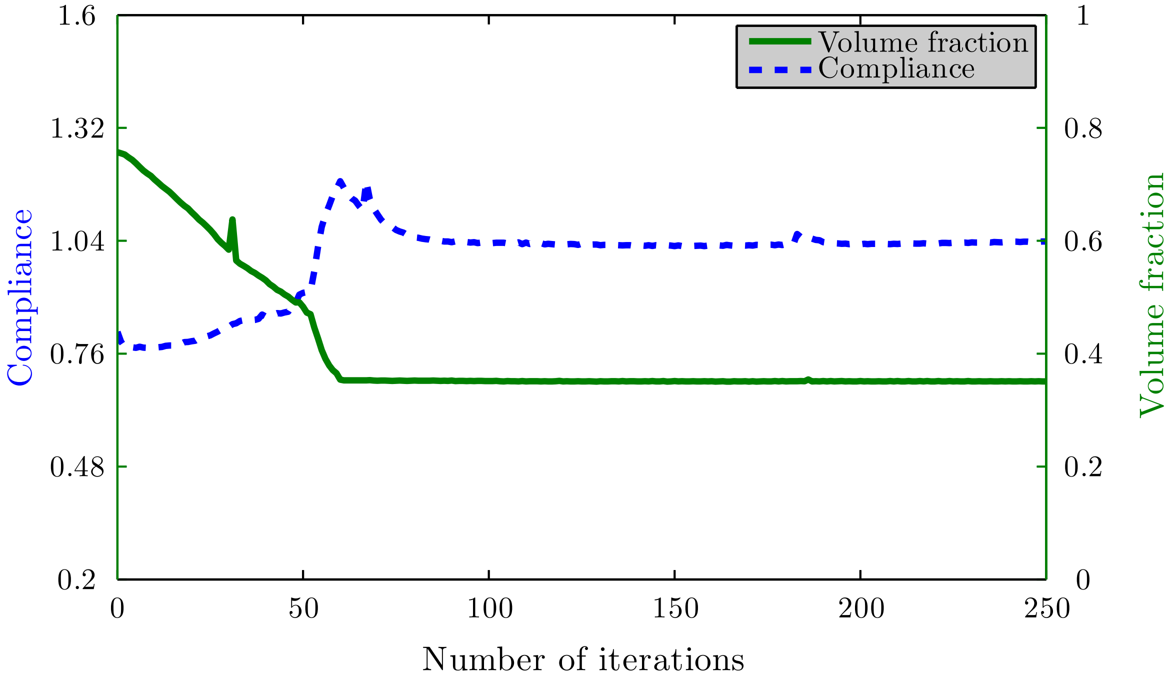
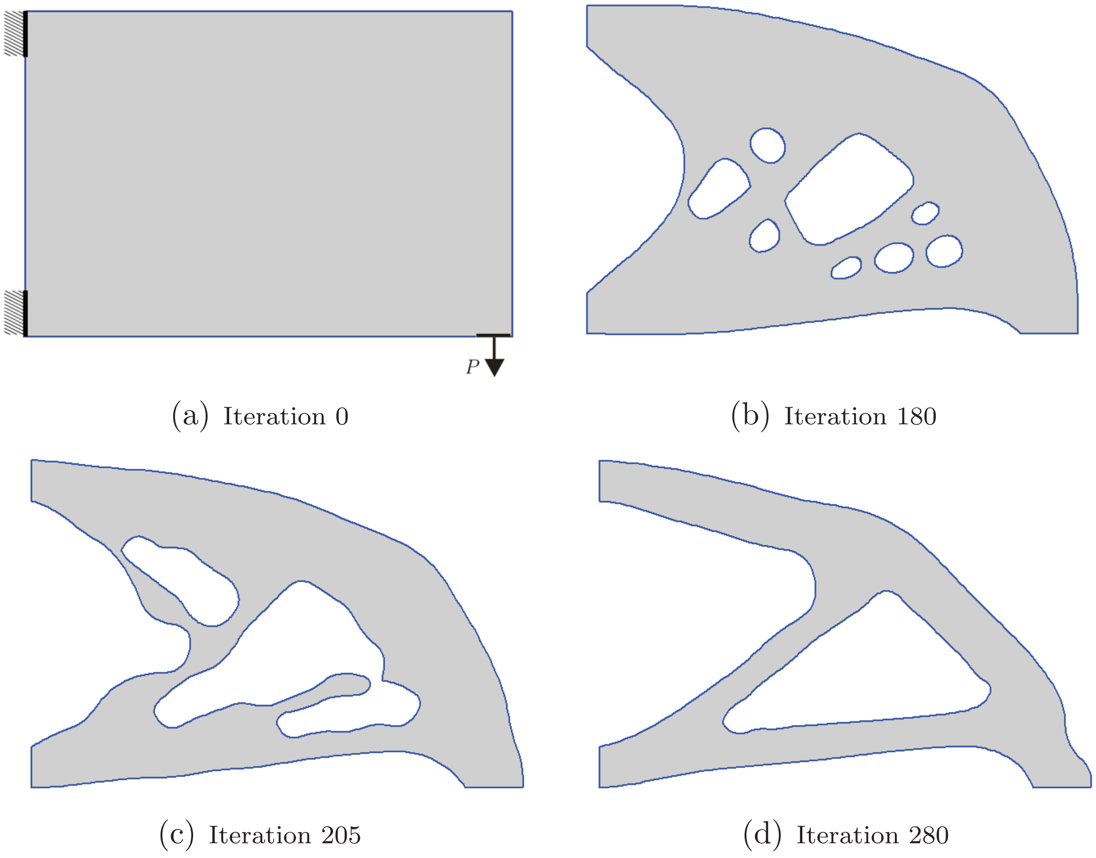


Figure 8.20: Convergence of objective function and volume for Example-3, second case



(a) Iteration 0

(b) Iteration 180

(c) Iteration 205

(d) Iteration 280

Figure 8.21: Evolution of structural geometry for Example-3, third case

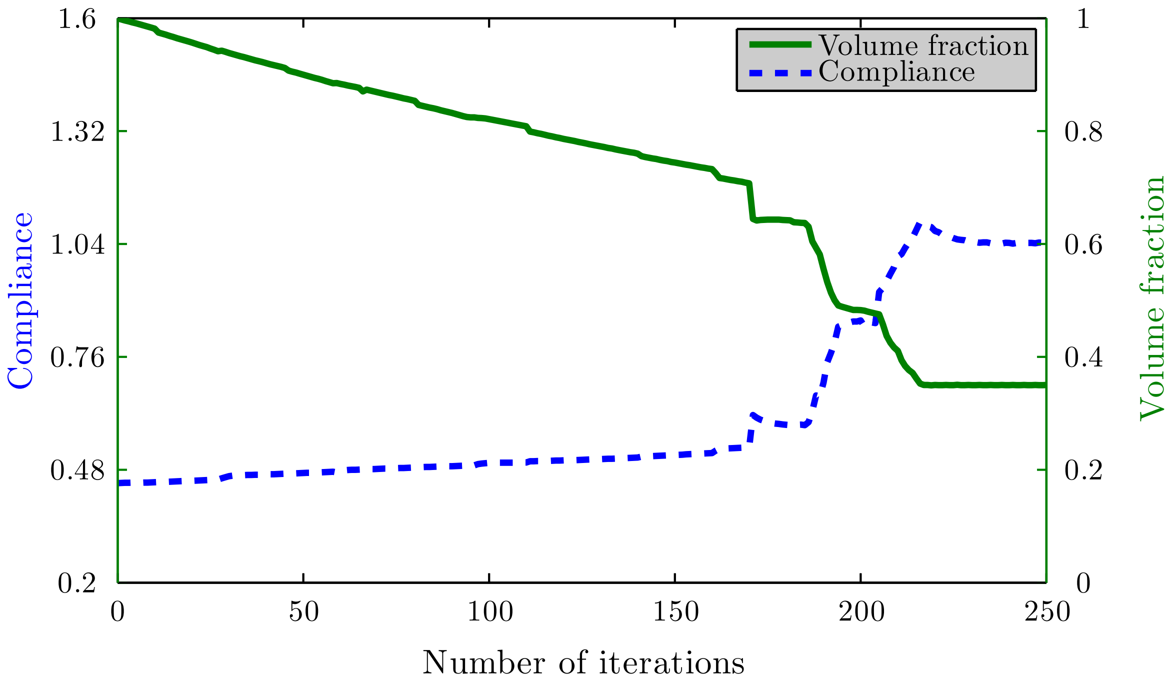


Figure 8.22: Convergence of objective function and volume for Example-3, third case

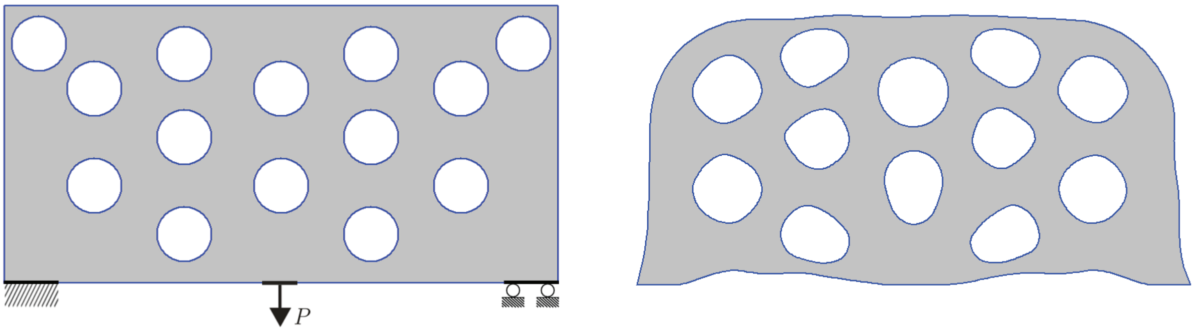
8.4.4 Example-4

The final example considered in this study is the Michell's type structure as shown in Figure 8.23(a) with an aspect ratio of 2:1. The zero displacement boundary conditions are applied in all directions at the right portions, and the load is applied at the middle of the bottom edge. Furthermore, the right hand portion of the bottom edge is constrained in the vertical direction and is allowed to move in the horizontal direction. The level set design domain is discretised with 80×40 square cells. The minimum compliance problem is solved for a target volume of $V = 0.35V_0$.

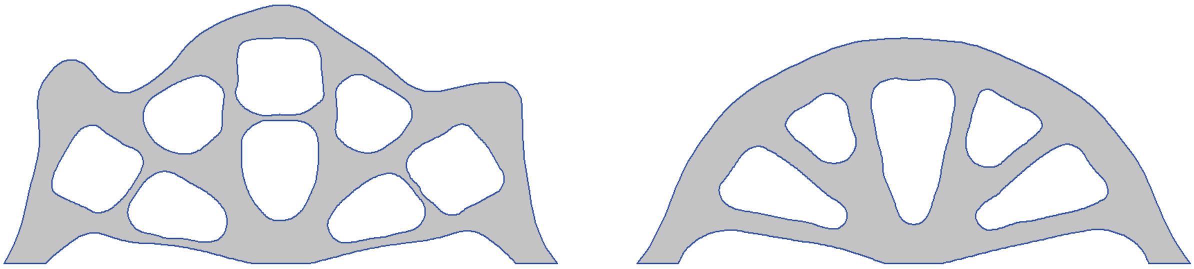
Two different initial guess designs are considered for the solution of the minimum compliance problem of the Michell's type structure. The fixed and exact combination ℓ approach has been used during the solution of the minimum compliance problem and the results obtained are discussed below.

An initial design with pre-existing holes is considered in the first case of this example. A fixed value of the Lagrange multiplier $\ell = 95$ is pre-specified up to $0.45V_0$. The bisectioning algorithm is initialised with $\ell_1 = 0$ and $\ell_2 = 300$. The evolution of the structural geometry at different stages of the optimisation process and the final optimal design is depicted in Figure 8.23(b)-(d). It can be seen that the final optimum obtained is identical to that presented in [8]. Similar trends of the objective and volume fraction can be seen from the convergence histories depicted in Figure 8.24 as observed in the previous cases of this example.

Finally, the sensitivity based optimisation method is tested against a Michell's type structure with completely filled initial guess design as shown in Figure 8.25(a). Similar to the previous case of this example, the minimum compliance problem is solved with a fixed and exact combination of ℓ. During the optimisation process, boundary movements are accompanied with hole insertion which evolves the initial design into an optimal design as shown in Figure 8.25(b)-(d). The evolution histories of the objective function and volume are displayed in Figure 8.26.



(a) Iteration 0 (b) Iteration 45



(c) Iteration 110 (d) Iteration 350

Figure 8.23: Evolution of structural geometry for Example-4, first case

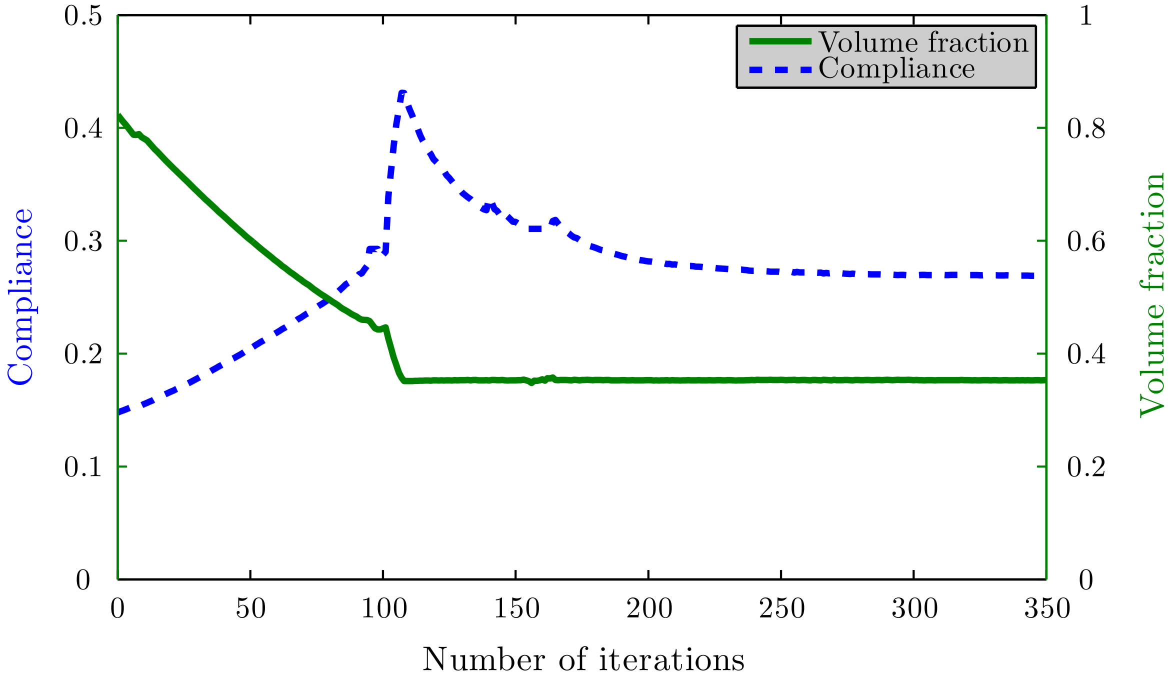
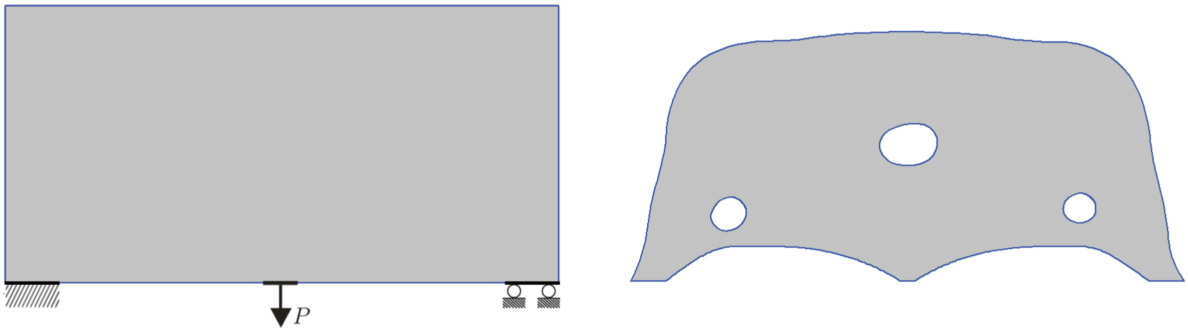
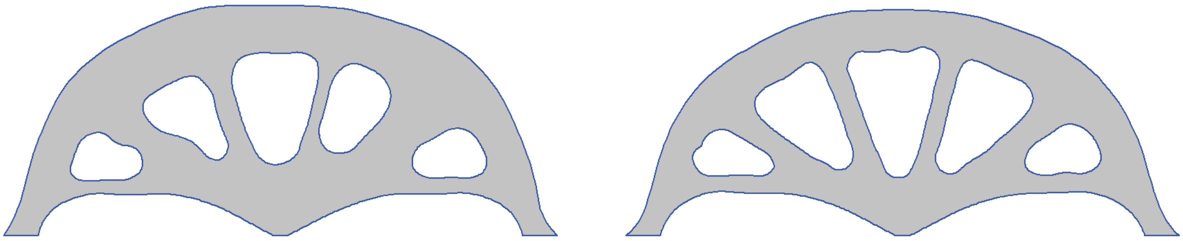


Figure 8.24: Convergence of objective function and volume for Example-4, first case



(a) Iteration 0 (b) Iteration 110



(c) Iteration 140 (d) Iteration 350

Figure 8.25: Evolution of structural geometry for Example-4, second case

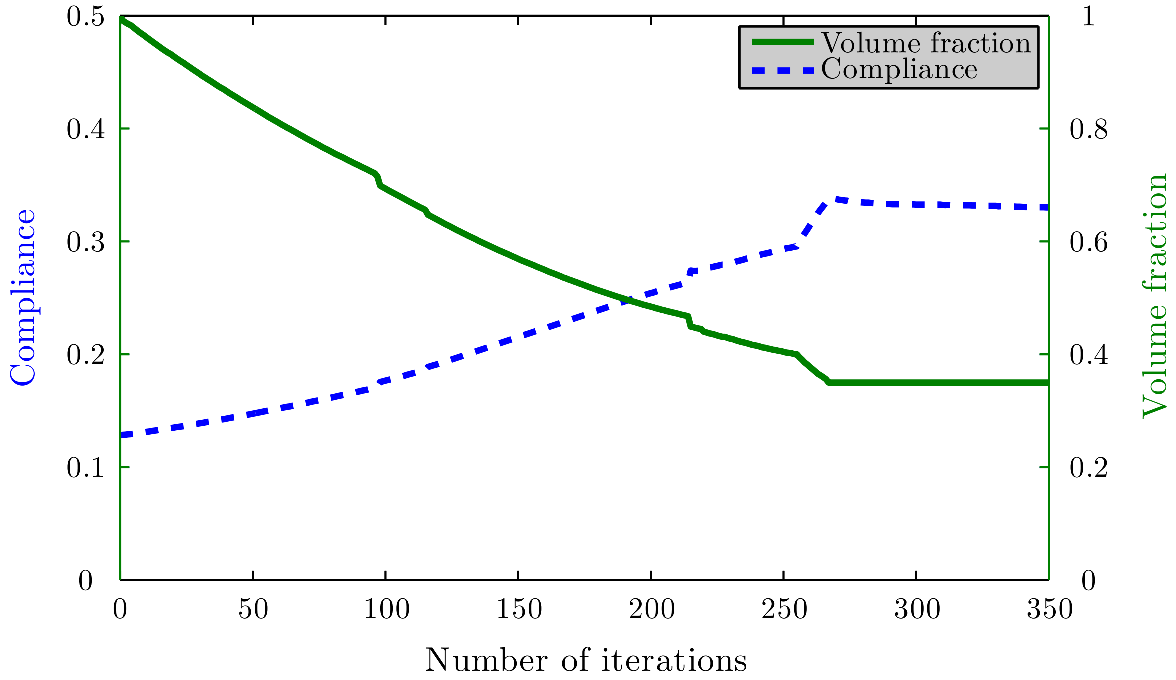


Figure 8.26: Convergence of objective function and volume for Example-4, second case

8.5 Conclusions

In this chapter the implementation of a sensitivity analysis, BEM and LSM based topology optimisation method is discussed in detail. The present method is applied to four different types of benchmark examples for shape and topology optimisation and the results obtained are in close agreement with those available in the literature. It has been observed that for a given problem different initial designs can be used and the optimisation method provides quite similar optimal solutions. In the current implementation internal points can be eliminated, and the computational efficiency can be enhanced if an initial guessed design with pre-existing holes is used. The results obtained show that the topological derivative approach for hole insertion (previously studied with an evolutionary structural optimisation approach) also fits well within the current implementation. Therefore, during the optimisation process, both shape and topology optimisation can be performed at the same time, and this overcomes the deficiency of a hole insertion mechanism in the previously presented BEM and LSM based optimisation methods, i.e. [3, 136]. In the present implementation, the bisectioning algorithm accurately calculates the Lagrange multiplier and thus provides smooth convergence of the objective function and the volume constraint. Furthermore, the use of NURBS provides a smooth optimal geometry and enhances the convergence of the optimisation method. It has been observed, that the final compliance obtained with a fixed and exact combination of Lagrange multiplier is relatively smaller than that obtained with an exact Lagrange multiplier. However, the fixed Lagrange multiplier used during the numerical implementation is based on the numerical experience and additional efforts will always be required to select an optimum value.

Chapter 9

A comparison of the evolutionary and sensitivities based optimisation methods

9.1 Overview

In this thesis two different optimisation algorithms have been presented in Sections 5.3 and 8.3, respectively. Both these approaches have been developed within the LSM and BEM framework with NURBS based geometry representation. The method presented in Section 5.3 is based on the ESO approach, and the shape sensitivity approach has been implemented in Section 8.3. It has been found that both methods resulted in similar optimal designs. In order to further investigate the similarities between the results obtained with the two different approaches, this chapter presents a comparative study of the two methods. A direct comparison of the two methods is not straight forward in a sense that the sensitivity based approach is driven by the objective function while in the classical ESO approach, there is no such mechanism available, which can directly minimise the objective function. Though, the ESO approach is supposed to minimise the compliance volume product [115]. In the LSM based evolutionary approach presented in Section 5.3, the optimisation process terminates at the target volume fraction. Nevertheless, the results obtained are in close agreement to those available in the literature. However, sensitivity based optimisation approach (presented in Section 8.3) is capable to carry out the optimisation process at constant volume once the target volume fraction has been achieved. In order to make a direct comparison of the results obtained with the two different methods the LSM based evolutionary optimisation method needs to be equipped with a mechanism which can accomplish the optimisation process at constant volume once the target volume fraction has been achieved. Therefore, in this Chapter, the LSM based evolutionary optimisation method presented in Section 5.3 is further improved with the addition of a constant volume mechanism through a bisectioning algorithm. Three different benchmark examples are considered in this Chapter, and each one is first solved with the modified evolutionary approach, i.e. LSM-ESO, and then with the sensitivity optimisation method (LSM-Sensitivity). The results obtained with the two different approaches are then compared to each other.

9.2 Optimisation algorithm

In this comparative study, each optimisation problem is solved using the algorithms presented in Figures 5.2 and 8.2, respectively. In addition, the numerical implementation of the constant volume mechanism within an LSM based optimisation approach is presented in Section 9.2.1. In order to make this comparison simple, the optimisation process in both cases starts from an initial guessed design with pre-existing holes. During each optimisation process the strain energy (compliance) is calculated at each iteration using Equation (5.7).

9.2.1 Constant volume constraint within an LSM based evolutionary optimisation

During the sensitivity based optimisation process front velocities have been calculated through strain energy densities coupled with the Lagrange multiplier approach, as explained in Section 8.2. Following that, Section 8.3.1 explains the corresponding constant volume constraint implementation in detail. However, a different approach has been presented in Section 5.3, where the front velocities are calculated through the von Mises stress and velocity relationship depicted in Figure 5.12. Due to this difference in the velocity calculations, the LSM based evolutionary optimisation method requires a new implementation of the constant volume constraint. The approach presented here is based on the bisectioning algorithm but implemented in a different way than that presented in Section 8.3.1.

In the proposed LSM-ESO method once the volume of the design domain reaches near the target volume a two step approach is used to add and remove material at constant volume. In the first step, the algorithm calculates the amount of material which can be added and removed based on the stress distribution within the design domain. The algorithm uses a bisectioning algorithm in the second step, which precisely adjusts the removal and addition rate, and hence the material removal and addition take place at constant volume. The first step of this algorithm is implemented as follow

1. Set $\bar{\phi} = \phi$.

2. After analysing the structure with BEM, select all those nodes along the structural boundary with

   $$\sigma_V \leq \sigma_{t2} \tag{9.1}$$

   where σ_{t2} is the stress level corresponding to Figure 5.12. Assign velocities $-1 \times l_{Rm}$ to all those nodes, where $l_{Rm} = 1$, is used as the material removal factor.

3. Extend velocities calculated in step 2 to grid points in the narrow band and update $\bar{\phi}$, i.e.

   $$\frac{\partial \bar{\phi}}{\partial t} + F|\nabla \bar{\phi}| = 0 \tag{9.2}$$

4. Trace the zero level set contours and calculate the new volume V_N. The material removed around the low stressed nodes, i.e. V_R is given as

   $$V_R = V - V_N \tag{9.3}$$

   where V is the volume calculated before the level set update.

5. Set $\bar{\phi} = \phi$.

6. In a similar way select all those nodes with

   $$\sigma_V \geq \sigma_{t3} \tag{9.4}$$

   where σ_{t3} is the stress level corresponding to Figure 5.12. Assign velocities $+1 \times l_{Ad}$ to all those points, where $l_{Ad} = 1$ represents a material addition factor.

7. Use step 3 and solve Equation (9.2) in accordance with the new velocities calculated in step 6.

8. Trace the zero level set contours and calculate the new volume V_N. The material added around the high stressed nodes, i.e. V_A is given as

$$V_A = V_N - V \tag{9.5}$$

In some cases if the material addition is near to zero, then the expression of σ_{t3} is relaxed through a removal factor R_A as follows

$$\sigma_{t3} = (0.95 - R_A)\min(\sigma_{Vmax}, \sigma_Y) \tag{9.6}$$

where $R_A = 0.1$, and is incremented by 0.1 until sufficient material addition takes place.

During the numerical implementation, it has been observed that V_R is always greater than V_A. There are two options available to make addition and removal equal to each other. It can be seen that the material addition takes place with velocity $+1 \times l_{Ad}$ with $l_{Ad} = 1$. If material addition rate is increased in accordance with the removal rate, then l_{Ad} requires to be more than 1 and in some cases this may violate the CFL condition. However, if the removal rate is decreased in accordance with the addition rate, then l_{Rm} will be less than 1 and in this case the CFL condition will always be satisfied. A bisectioning scheme is proposed below, which add and remove material at the same rate thus, maintaining a constant volume during the optimisation process.

In the implementation of the bisectioning algorithm two additional factors are introduced which bounds l_{Rm}. The complete algorithm is explained in the following steps.

1. Initialise $l_{Rm1} = 0$ and $l_{Rm2} = 1$
2. Set $\bar{\phi} = \phi$
3. Half the interval, i.e.

$$l_{Rm} = (l_{Rm1} + l_{Rm2})/2 \tag{9.7}$$

4. Assign velocities $-1 \times l_{Rm}$ to all those points according to Equation (9.1).
5. Extend velocities to grid points in the narrow band.
6. Solve Equation 9.2.
7. Trace the zero level set contours and calculate V_R using Equation (9.3).
8. if $V_A \leq V_R$, $l_{Rm2} = l_{Rm}$ else $l_{Rm1} = l_{Rm}$.
9. Terminate if $|V_A - V_R| \leq 10^{-2}$, otherwise go to step 2.
10. Set $\bar{\phi} = \phi$
11. Use $l_{Rm} = 0.5 \times l_{Rm}$ (where l_{Rm} is calculated from step 8) and $l_{Ad} = 1$ and assign velocities to node points according to Equation (9.1) and (9.4).
12. Extend velocities calculated in the above step to grid points in the narrow band.
13. Update ϕ using Equation (2.3).

In step 11, the l_{Rm} is re-adjusted to half of the actual value calculated in step 8. This is because, the volume calculated with the actual value was lower than the target volume. Therefore, the modified l_{Rm} used is based on the results of the numerical tests carried out, which exactly satisfies the volume constraint.

9.3 Examples

In order to make a comparison of the LSM based evolutionary and sensitivity optimisation methods the material properties and assumptions made for the solution of the optimisation problems are exactly the same as used in Sections 5.4 and 8.4, respectively. Both methods compute normal velocities with different approaches. However, the time step sizes used for the solution of level set equation in both cases is in accordance with the CFL condition. In both optimisation methods, the function evaluation is the same until the structure reaches the target volume fraction. Afterwards, the LSM-ESO uses a two step approach to carry out the optimisation process at constant volume, which requires a total of 21 steps. However, the LSM-Sensitivity approach requires 11 steps for each iteration at constant volume.

9.3.1 Example-1

In the first example a cantilever beam with an aspect ratio of 2:1 is considered. The initial design with an applied load and boundary conditions is depicted in Figure 9.1. The structure is constrained at the top and bottom of the left hand edge with zero displacement boundary conditions, and a load $P = 100$N is applied at the middle of the right hand edge. The traction free boundary is represented by NURBS, and is allowed to vary during the optimisation process. The optimisation problem for minimum compliance is solved for a volume fraction $\alpha = 0.5$. The level set domain is discretised into 80×40 square cells.

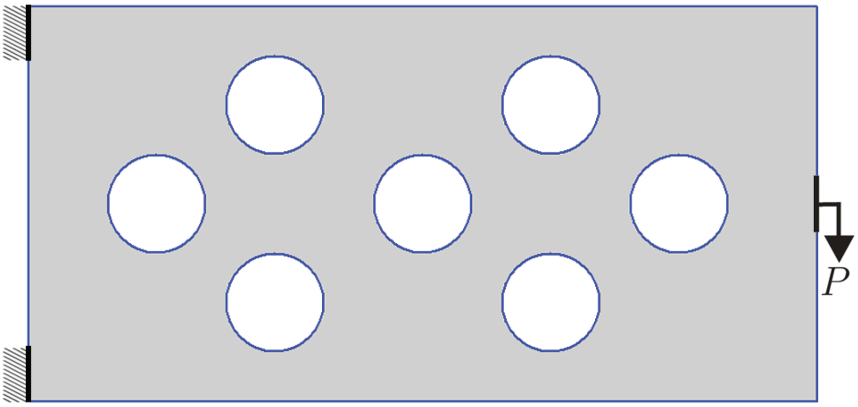
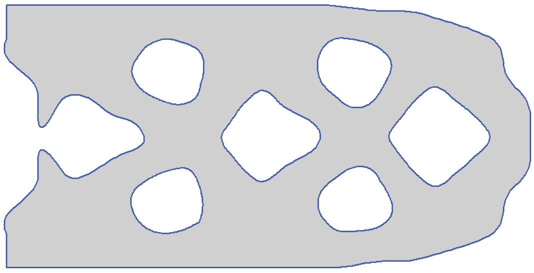


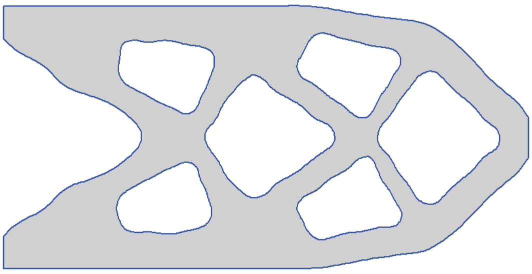
Figure 9.1: Initial geometry with applied loads and constraints for Example-1

The optimisation problem is first solved with the LSM-ESO approach and the constant volume algorithm is used once $\alpha = 0.50$ is reached at iteration 60. The same initial design is used for the solution of the optimisation problem using LSM-Sensitivity approach. A fixed Lagrange multiplier, i.e. $\ell = 120$ is used up to $\alpha = 0.65$ and the bisectioning algorithm is used afterwards to calculate ℓ which exactly satisfies the volume constraint.

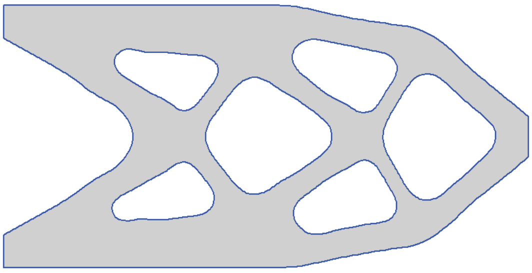
Figure 9.2 shows the evolution of the structural geometry using the LSM-ESO approach. Similarly, Figure 9.3 displays results obtained with the LSM-Sensitivity method. Comparison of the results shows that both methods result in optimal designs with similar topologies; however, the geometries are slightly different from each other. Figure 9.2(b) shows the results obtained at iteration 59 with the LSM-ESO where the structural geometry reaches the target volume fraction. From iteration 60 to 200, material addition and removal takes place continuously at constant volume, and this results in an optimal design with smoother geometry and better material distribution than that at iteration 59. Figure 9.4 shows the evolution of the compliance and volume at each iteration with



(a) Iteration 33

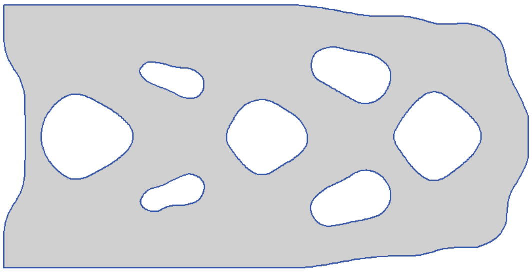


(b) Iteration 59

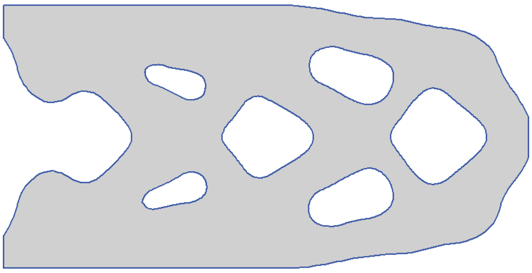


(c) Iteration 200

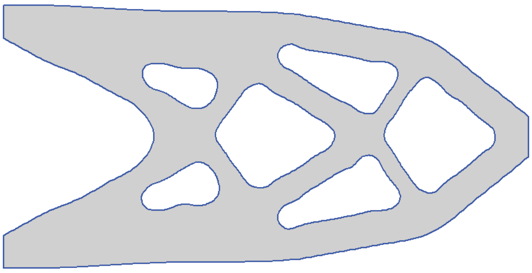
Figure 9.2: Evolution of structural geometry for Example-1, using LSM-ESO method



(a) Iteration 56



(b) Iteration 77



(c) Iteration 200

Figure 9.3: Evolution of structural geometry for Example-1, using LSM-Sensitivity method

the LSM-ESO optimisation method. It can be seen, that the volume of the structure decreases slowly in the initial iterations and results into a corresponding increase in the compliance. Once the target volume fraction has been achieved, material addition and removal takes place at constant volume and the compliance remains stable at 1.51 until iteration 100 is reached. In the subsequent iterations, the compliance first decreases and then slightly increases up to iteration 148. A further decrease in the compliance can be observed afterwards and remains stable at 1.49 until the end of the optimisation process. It is evident from the evolution history of the structural volume, that the use of bisectioning algorithm exactly satisfies the volume constraint during the optimisation process.

The evolution of the compliance and volume at each optimisation iteration are recorded and displayed in Figure 9.5 with the LSM-Sensitivity optimisation method. In the initial iterations, a slow decrease in the volume resulted in a decrease in the compliance. In the following iterations, the compliance increases gradually up to $\alpha = 0.65$. Afterwards, the bisectioning algorithm is used to calculate ℓ which exactly satisfied the volume constraint, and resulted in a rapid reduction in the volume and a corresponding increase in the compliance of the structure. Once the volume constraint is satisfied, the topological changes take place in the subsequent iterations, and this gradually decreases the compliance to 1.48 and remains stable afterwards.

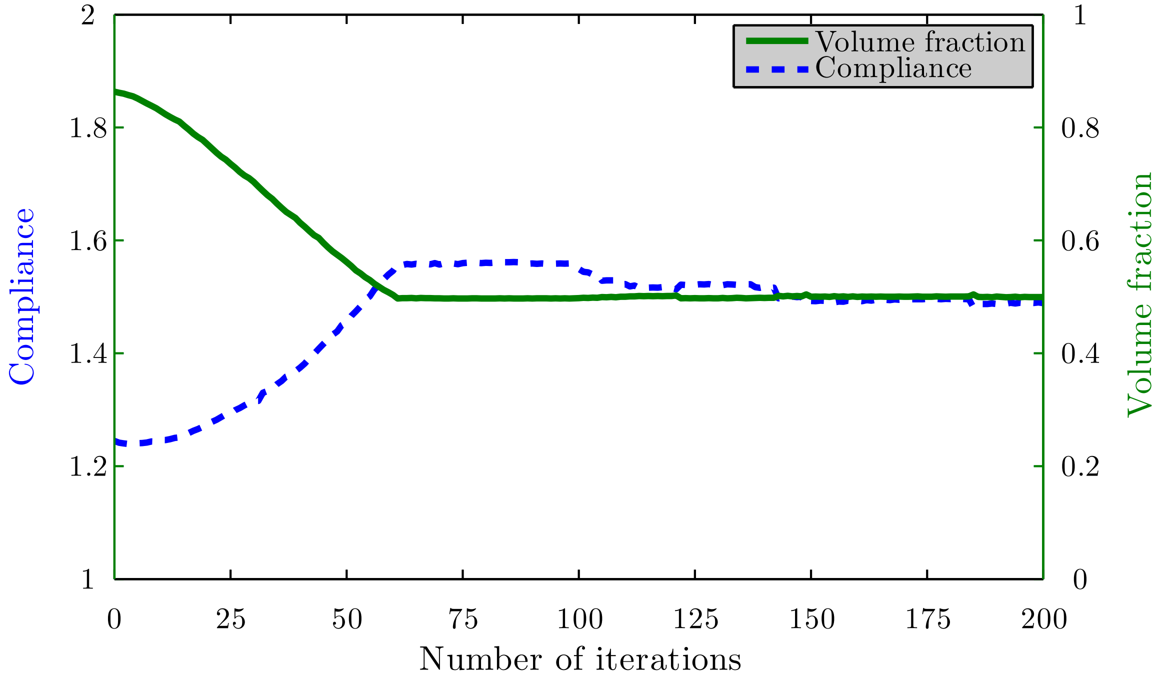


Figure 9.4: Evolution of compliance and volume for Example-1, using LSM-ESO

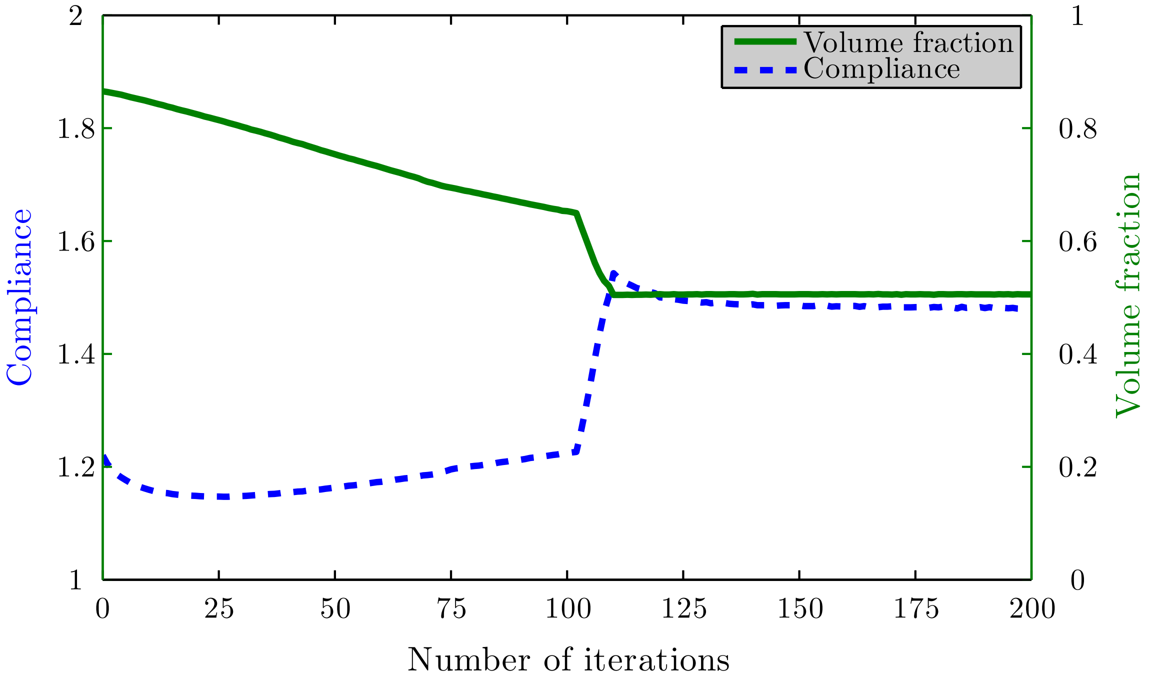


Figure 9.5: Evolution of compliance and volume for Example-1, using LSM-Sensitivity

Finally, the evolution of the compliances with the two optimisation methods are directly compared in Figure 9.6 and also presented in Table 9.1. This comparison suggests that evolutions of the compliances are slightly different from each other; however, the final compliances are approximately equal to each other. Moreover, the volume converged more rapidly with the LSM-ESO approach than that with the LSM-Sensitivity method.

	LSM-ESO	LSM-Sensitivity
Target volume fraction	0.50	0.50
No of iterations used to reach the target volume fraction	60	110
Compliance at target volume fraction	1.54	1.54
Total number of iterations	200	200
Final compliance	1.49	1.48

Table 9.1: Comparison of LSM-ESO and LSM-Sensitivity for Example-1

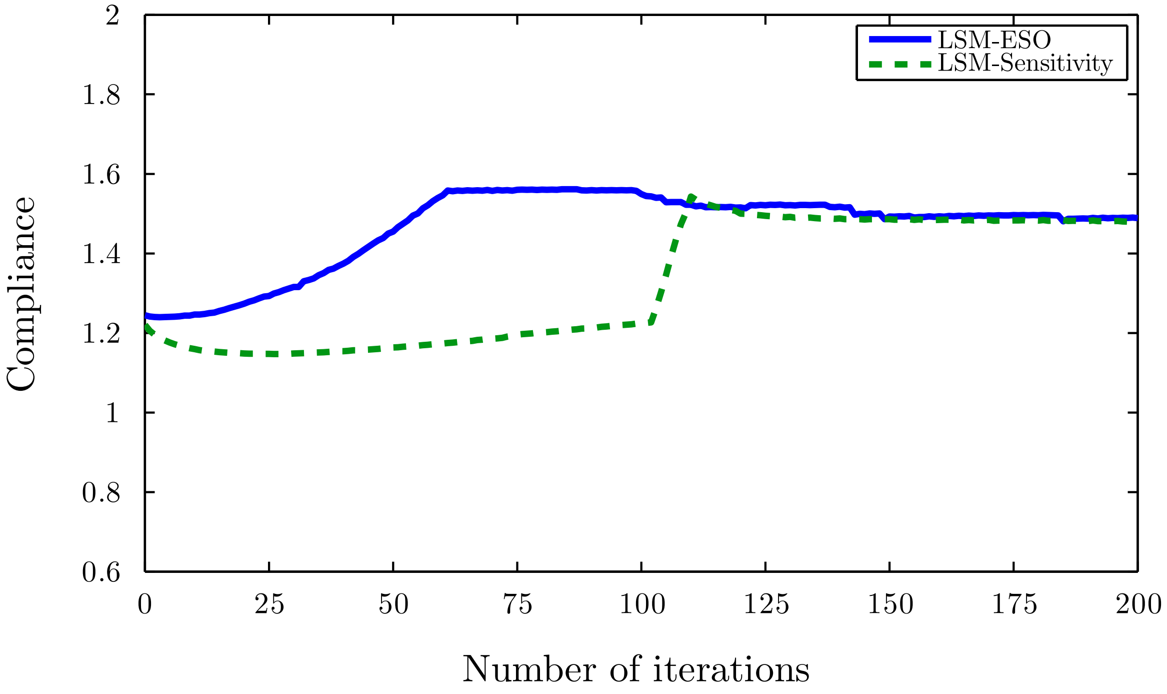


Figure 9.6: Comparison of evolutions of compliances using LSM-ESO and LSM-Sensitivity, for Example-1

9.3.2 Example-2

In the second example, the optimisation problem is solved for a cantilever beam with an aspect ratio of 1.5:1. The zero displacement boundary conditions are described at the top and bottom portions of the left hand edge, and the structure is loaded with $P = 100$N at the middle of the right edge as shown in Figure 9.7. The specified target volume fraction used for this example is $\alpha = 0.35$. The level set design domain is discretised into 60×40 square cells.

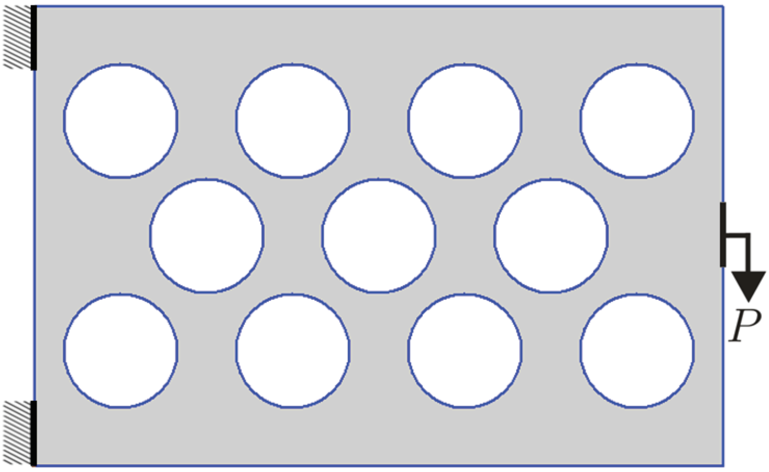
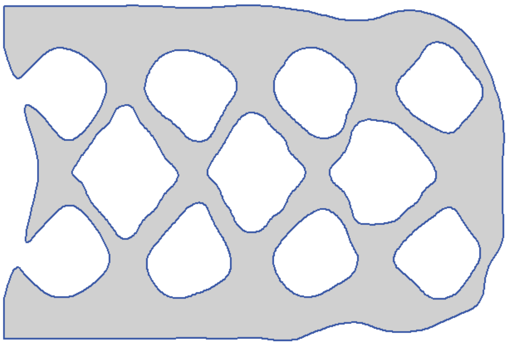


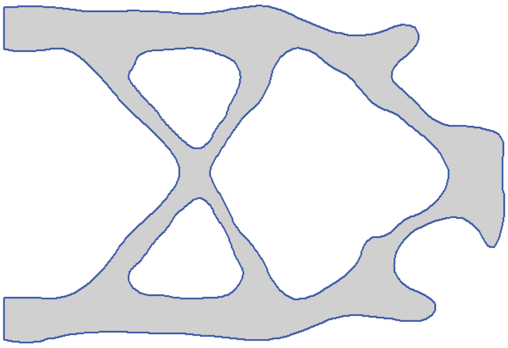
Figure 9.7: Initial geometry with applied loads and constraints for Example-2

In the second example, the LSM-ESO and LSM-Sensitivity methods are used for the solution of the optimisation problem. The evolutionary approach uses the maximum σ_V as a reference stress for material removal and addition. In the sensitivity based approach, a fixed value of $\ell = 200$ is initially selected up to $\alpha = 0.45$ and the bisectioning algorithm is then used in the subsequent iterations. The results obtained with the two different methods are depicted in Figures 9.8 and 9.9, respectively. It is evident from the comparison of results with the two different approaches, that the structural geometries evolve in a similar way. During this evolution process boundary movements take place, and this is accompanied with hole merging with each other and with the boundary.

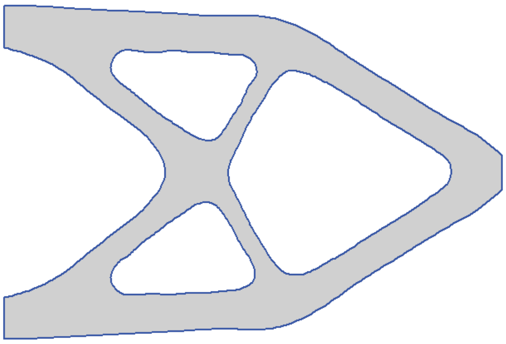
Figure 9.8(b) displays the intermediate geometry obtained near to the target volume fraction. From iteration 38 to 200, only shape optimisation is performed through material



(a) Iteration 24

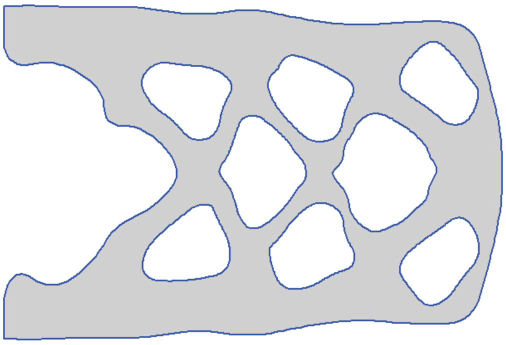


(b) Iteration 37

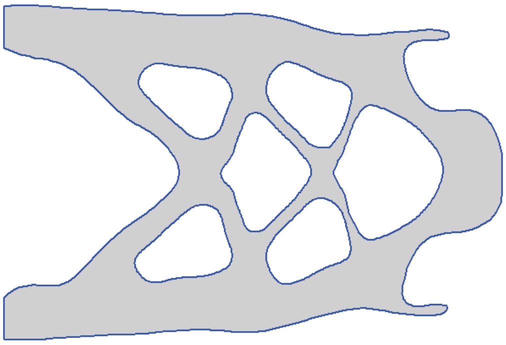


(c) Iteration 150

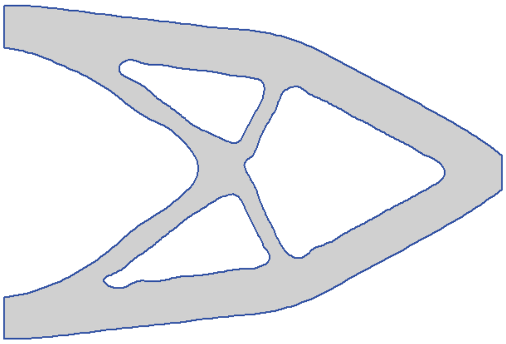
Figure 9.8: Evolution of structural geometry for Example-2, using LSM-ESO method



(a) Iteration 20



(b) Iteration 31



(c) Iteration 150

Figure 9.9: Evolution of structural geometry for Example-2, using LSM-Sensitivity method

addition and removal at constant volume. The constant volume implementation provides an efficient way of redistributing material within the design domain and results in an optimal design with more realistic geometrical description than that at iteration 37. In addition, the optimal design closely resembles the results available in the literature, e.g. [133]. A similar behaviour can also be observed during the evolution of the structural geometry with the LSM-Sensitivity implementation. However, the optimal design shown in 9.8(c) has better material distribution and smoother geometrical description than that obtained in Figure 9.9(c).

Figure 9.10 shows the evolution of compliance and volume throughout the optimisation process. In the initial iterations the structural volume slowly decreases and the compliance increases accordingly. At iteration 36, high peak can be observed, which is mainly related to the holes merging in the design domain. In the subsequent iterations, the evolving geometry reached the target volume fraction near iteration 37. Afterwards, material addition and removal take place at approximately constant volume and as a result the compliance of the structure is minimised. This shows that with the use of constant volume implementation, the LSM-ESO method is capable to minimise the compliance of the structure through material re-distribution within the design domain. The optimisation process terminates with a final compliance value of 1.07.

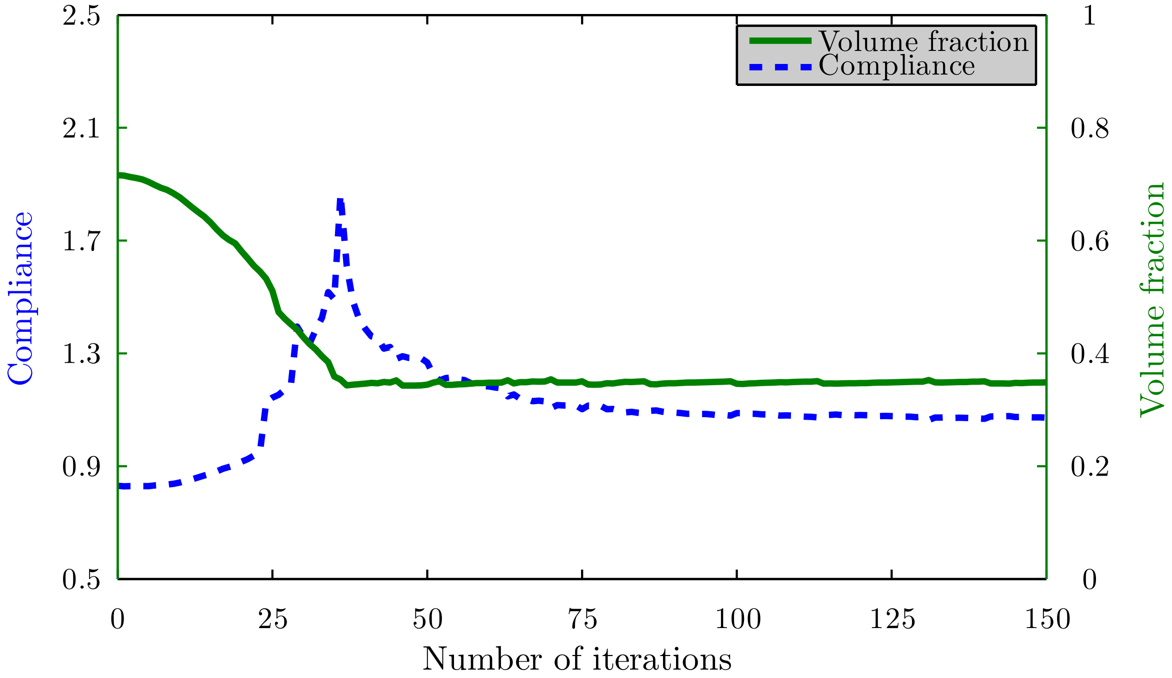


Figure 9.10: Evolution of compliance and volume for Example-2, using LSM-ESO

The evolution of the compliance and the volume constraint functions at each optimisation iteration are depicted in Figure 9.11 with the LSM-Sensitivity method. In the initial iterations, a slow decrease in the volume results in a decrease in the compliance of the structure. In the following iterations, the compliance gradually increases until $\alpha = 0.45$ is reached and the bisectioning algorithm calculates ℓ which exactly satisfies the volume constraint. This results in a reduction in the volume and a corresponding increase in the compliance of the structure. Once the volume constraint is satisfied, the topological changes take place in the subsequent iterations, and this gradually decreases the compliance to 1.06 and remains stable up to the end of the optimisation process.

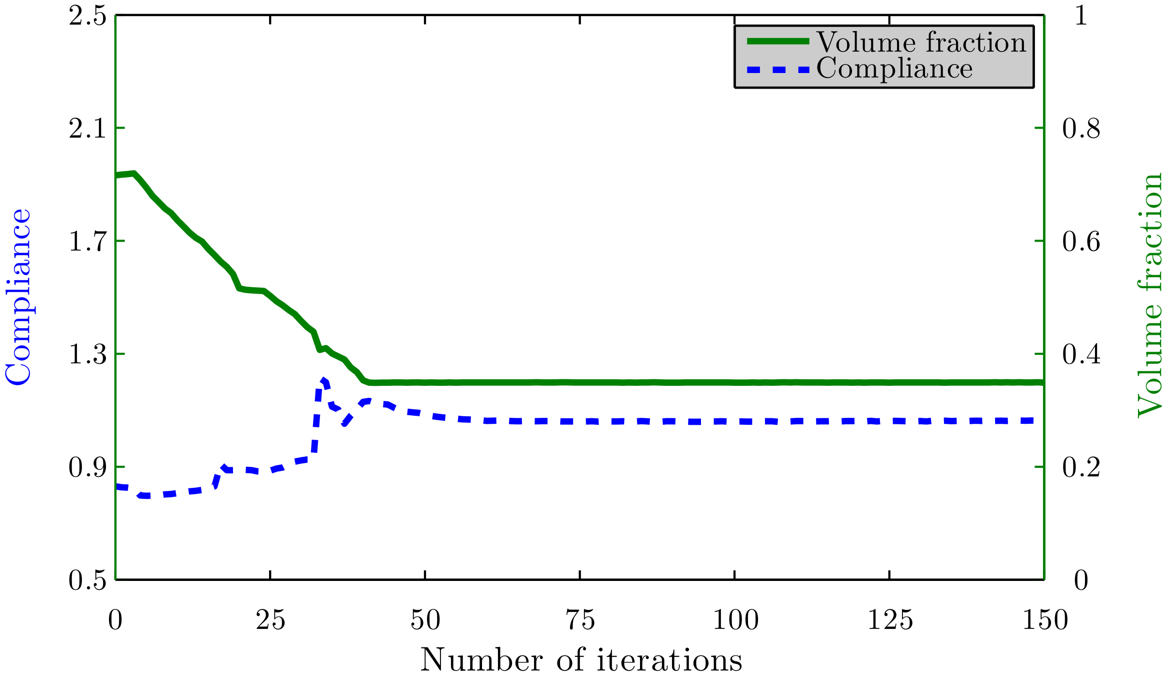


Figure 9.11: Evolution of compliance and volume for Example-2, using LSM-Sensitivity

The final optima obtained in this example have the same number of cavities; however the LSM-ESO provides a better and symmetric final optimum than that with the LSM-Sensitivity. Figure 9.12 and Table 9.2 display a direct comparison of the evolutions of compliances during the optimisation process for Example-2. Similar trends of compliances can be observed in the initial iterations. Afterwards, the compliance with the LSM-ESO diverges with a high peak, which results from hole merging within the design domain. Once the target volume fraction has been achieved in the subsequent iterations, the compliances converge with similar trends. Finally, the optimisation processes terminate with a 1% difference in the final compliances.

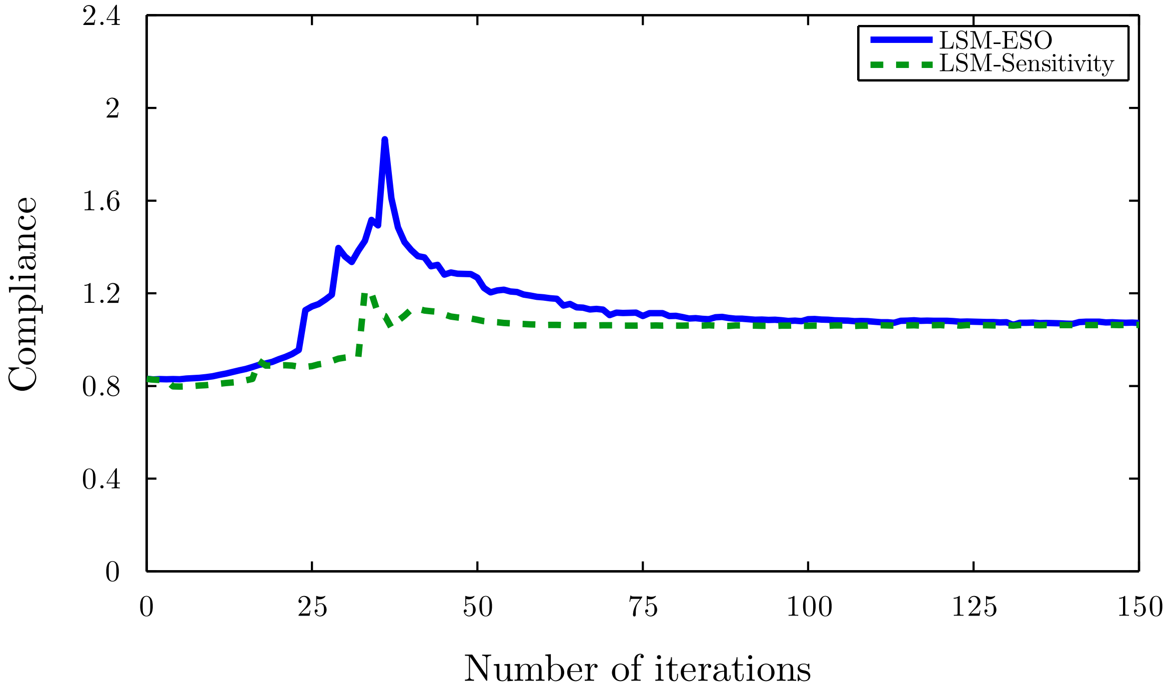


Figure 9.12: Comparison of evolutions of compliances using LSM-ESO and LSM-Sensitivity, for Example-2

	LSM-ESO	LSM-Sensitivity
Target volume fraction	0.35	0.35
No of iterations used to reach the target volume fraction	40	40
Compliance at target volume fraction	1.32	1.10
Total number of iterations	150	150
Final compliance	1.06	1.06

Table 9.2: Comparison of LSM-ESO and LSM-Sensitivity for Example-2

9.3.3 Example-3

The minimum compliance problem for the third example is solved for a cantilever beam with an aspect ratio of 1.5:1 as shown in Figure 9.13. The zero displacement boundary conditions are described at the top and bottom portions of the left hand edge and the structure is loaded with $P = 100$N at the right of the bottom edge. The specified target volume fraction for this example is $\alpha = 0.35$. The level set design domain is discretised into 60×40 square cells.

In this example, the optimisation problem is first solved with the LSM-ESO and then

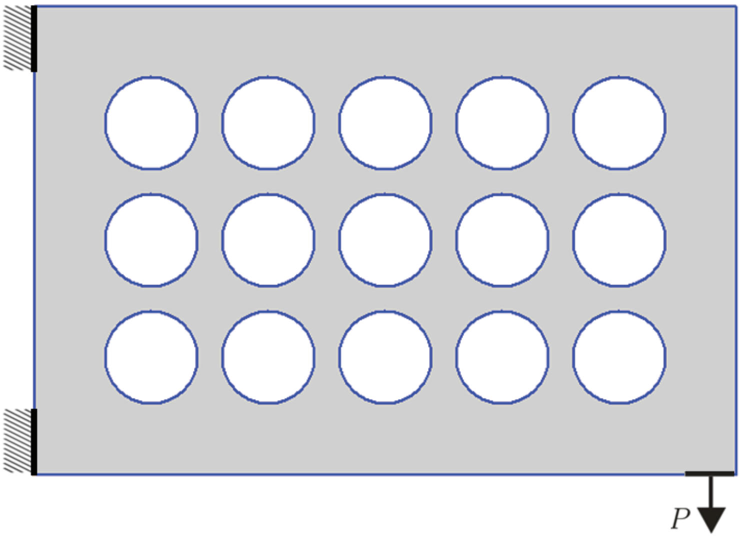
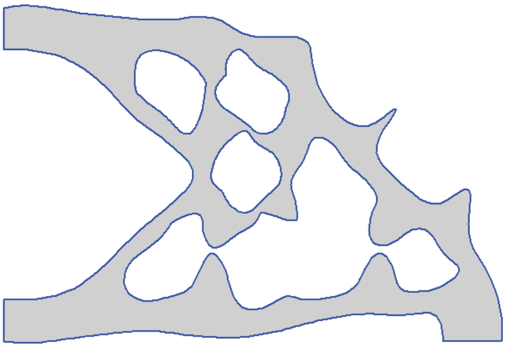


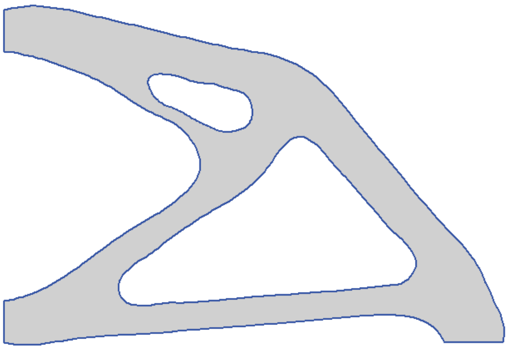
Figure 9.13: Initial geometry with applied loads and constraints for Example-3

with the LSM-Sensitivity approach. The LSM-ESO uses the maximum σ_V as criterion to add and remove material during the optimisation process. Once the structural geometry reached the target volume fraction the constant volume algorithm is used afterwards. However, a fixed value of $\ell = 250$ is used up to $\alpha = 0.50$ and the bisectioning algorithm is then used to carry out the optimisation process at constant volume.

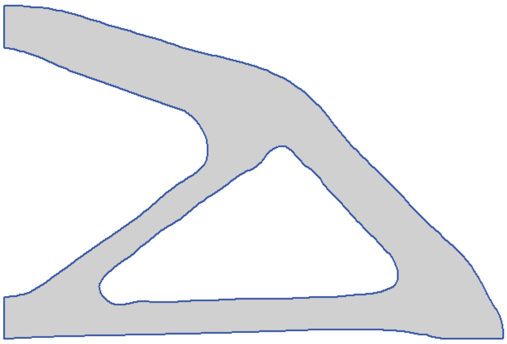
Figures 9.14 and 9.15 display the evolution of the structural geometry at different stages of the optimisation process. The optimal designs obtained with the two different approaches are quite similar to each other. Figure 9.14(a) shows the intermediate geometry at iteration 46 at the target volume fraction, i.e. $\alpha = 0.35$ using the LSM-ESO method. In the subsequent iterations, material addition and removal at constant volume redistributes material within the design domain and thus provides optimal design with better geometrical description than that at iteration 46. During this process hole merging also takes place at constant volume as evident from Figure 9.14(b). Similarly, Figure 9.15(b) shows the evolving geometry near to the target volume fraction with the LSM-Sensitivity approach. It can be seen that both methods are equally capable of adding and removing material at constant volume, and results in optimal designs with better material distribution within the final design than that in Figures 9.14(b) and 9.15(b), respectively.



(a) Iteration 46

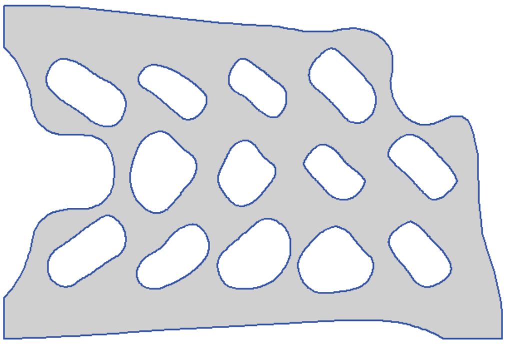


(b) Iteration 120

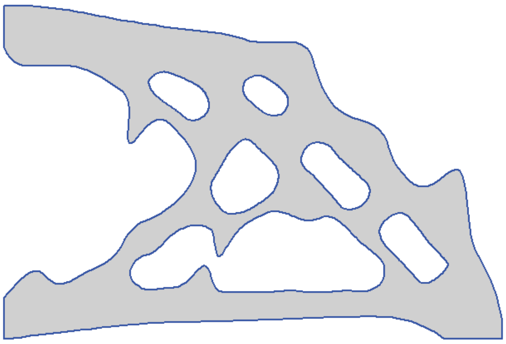


(c) Iteration 250

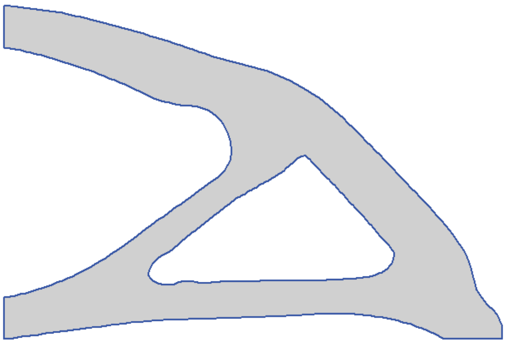
Figure 9.14: Evolution of structural geometry for Example-3, using LSM-ESO method



(a) Iteration 33



(b) Iteration 55



(c) Iteration 250

Figure 9.15: Evolution of structural geometry for Example-3, using LSM-Sensitivity method

The evolution of compliance and the volume with both optimisation methods are depicted in Figures 9.16 and 9.17, respectively. In both cases, material removal takes place in the initial iterations and results in an increase in the compliance of the structure. Figure 9.16 shows high peak at iteration 46 which is related to the geometry shown in Figure 9.14(a). At this stage, the structural geometry is at the target volume fraction and in the following iterations, material addition and removal take place at constant volume and considerably minimises the effect of high peak observed at iteration 46.

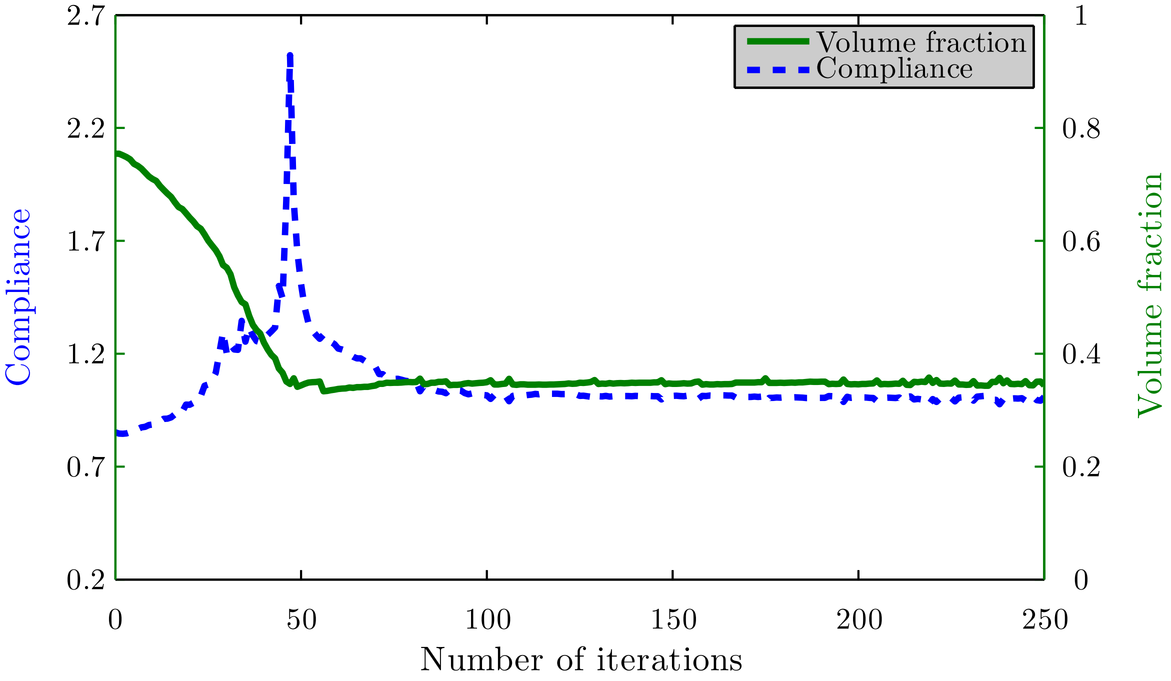


Figure 9.16: Evolution of compliance and volume for Example-3, using LSM-ESO

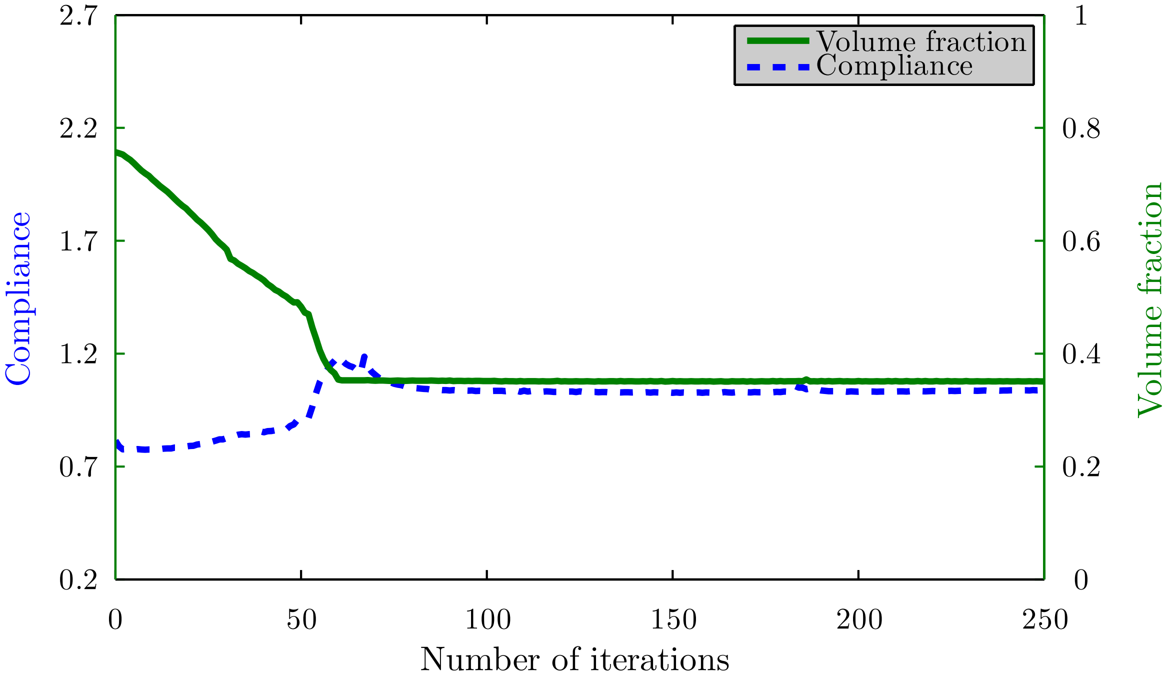


Figure 9.17: Evolution of compliance and volume for Example-3, using LSM-Sensitivity

A direct comparison of the evolutions of compliances is depicted in Figure 9.18 and also shown in Table 9.3. In the initial iterations, both compliances evolves with different trends as a result of different material removal rate. However, similar behavior of compliances can be observed once the volume constraint is satisfied during the optimisation process. The final compliance recorded for both methods are 1.00 and 1.03, respectively. This indicates that optimal designs obtained with similar topologies and with a 3% difference in the final compliances.

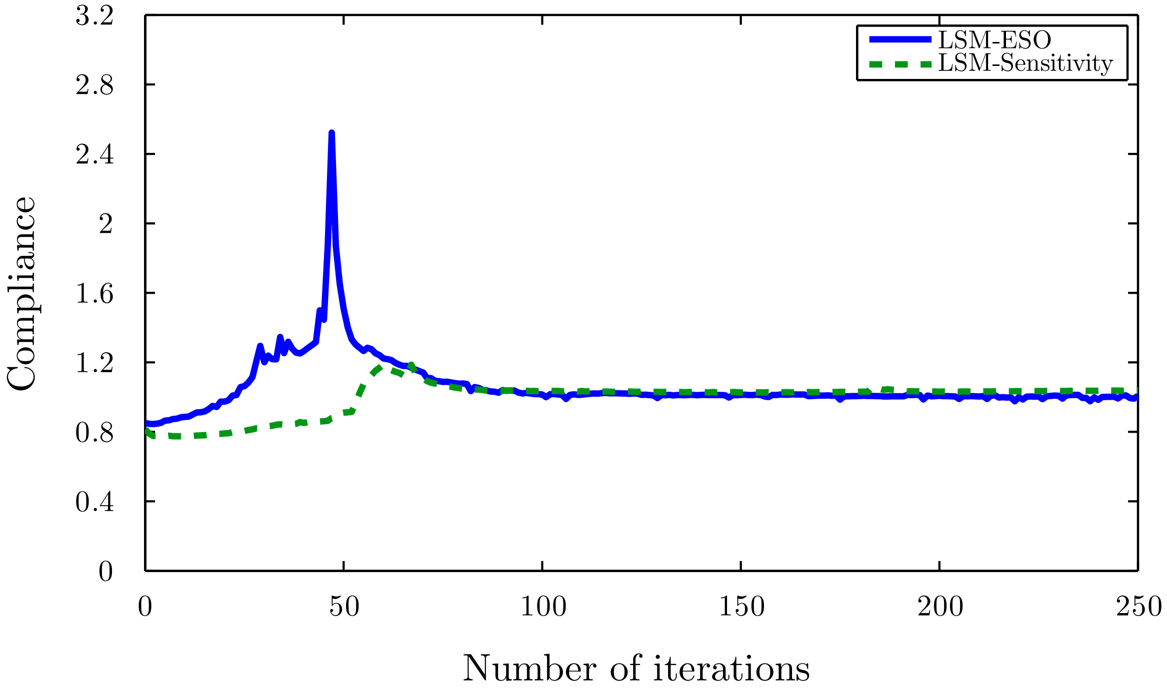


Figure 9.18: Comparison of evolutions of compliances using LSM-ESO and LSM-Sensitivity, for Example-3

	LSM-ESO	LSM-Sensitivity
Target volume fraction	0.35	0.35
No of iterations used to reach the target volume fraction	66	60
Compliance at target volume fraction	1.17	1.18
Total number of iterations	250	250
Final compliance	1.00	1.03

Table 9.3: Comparison of LSM-ESO and LSM-Sensitivity for Example-2

9.4 Conclusions

A constant volume based LSM-ESO method has been presented in this chapter. During the optimisation process once the target volume fraction has been achieved, the proposed method effectively preserves volume in the remaining iterations with the use of bisectioning algorithm. This new implementation allows material addition and removal at constant volume, mainly through boundary movements. Thus, only shape optimisation takes place and as a result efficient material is added to the high stressed regions while the same amount of inefficient material is removed from the low stressed regions. During the evolution of the structural geometry at constant volume, material is redistributed within the design domain and interestingly compliance is either minimised or remain stable. The proposed method is tested with three different benchmark examples and results obtained are in close agreement to those available in the literature. In order to make a direct comparison of the proposed method, each example is further solved with the LSM based sensitivity approach presented in Chapter 8. For each example, it has been observed that the optimal designs obtained with the two different methods, and the corresponding evolutions of the compliances are similar to each other. In some cases, the LSM-ESO produces better compliance solutions than the LSM-Sensitivity approach driven by compliance based objective function. In comparison with the LSM-Sensitivity approach, all problems solved with the LSM-ESO are based on the same settings of different parameters used. However, different Lagrange multipliers have been used for each problem solved with the LSM-Sensitivity approach. Further, additional efforts are always required to use an appropriate value of the Lagrange multiplier. This indicates that the LSM-ESO approach is more user friendly than the LSM-Sensitivity method. Furthermore, the LSM-ESO provides a better geometrical description of the optimal designs obtained in each example.

Chapter 10

A 3D implementation of BEM and LSM based structural optimisation

10.1 Overview

This chapter presents a three-dimensional evolutionary structural optimisation approach based on the level set and boundary element methods. The proposed optimisation method extends the two-dimensional approach presented in Section 5.2 to three-dimensions. During the optimisation process, the LSM evolves the structural boundary into an optimal design using the material removal or addition criterion specified by the optimisation method. The boundary movements in 3D LSM allow automatic hole nucleation by the intersection of two surfaces moving towards each other. This eliminates the need of an additional hole nucleation mechanism as used in the 2D LSM based optimisation. At each optimisation iteration, the Marching Cubes (MC) algorithm extracts the new zero level set contours in the form of a triangular mesh. As the BEM is based on a boundary discretisation approach, the extracted geometry (in the form of a triangular mesh) can be directly analysed within it. This eliminates the need for an additional discretisation tool and provides a natural link between the LSM and the BEM in a three-dimensional structural optimisation. This suggests that only the boundary perturbation guarantees changes in both shape and topology, and there is no need to calculate stresses and displacements within the design domain. Therefore, there is no need to use internal points, which further enhances the computational capabilities of the proposed optimisation method.

10.2 Optimisation algorithm

The 3D optimisation algorithm proposed in this chapter is an extension of the 2D approach presented in Section 5.3. During the optimisation process, the structural geometry evolves into an optimal topology through the progressive removal of inefficient material from the low stressed regions and addition it to the high stressed regions. The performance of the optimisation process is monitored through the specific strain energy (i.e. Equation 5.6) and the target volume fraction is used as a stopping criterion.

The proposed optimisation algorithm is illustrated in Figure 10.1 and summarised as follows:

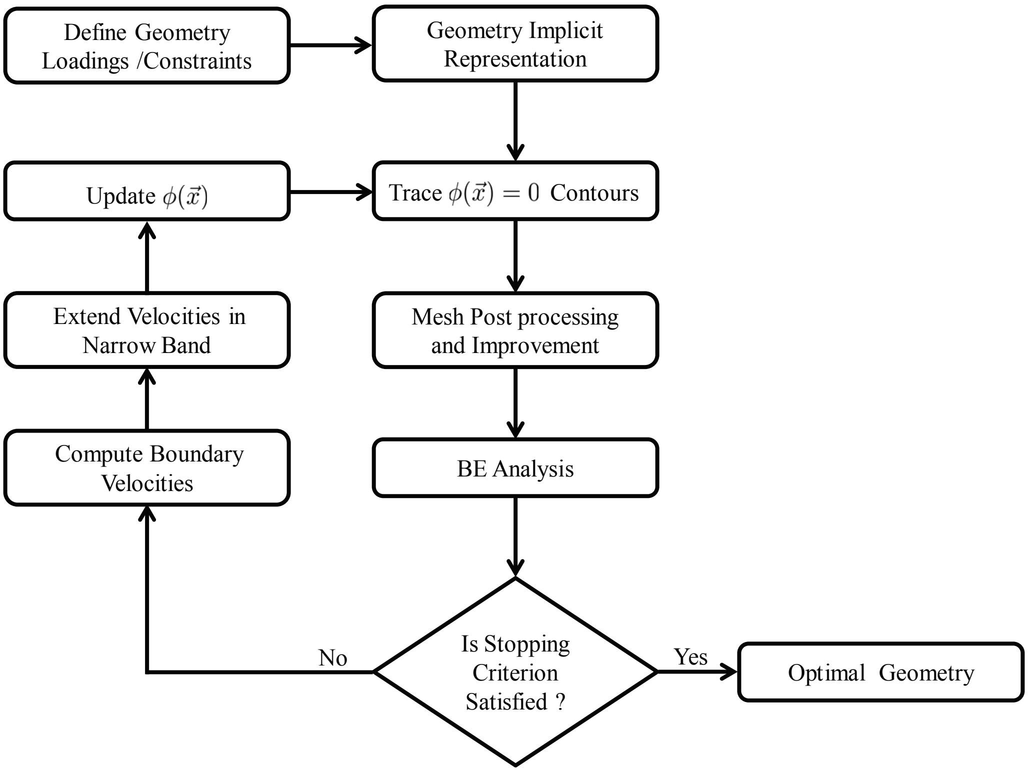


Figure 10.1: Optimisation flow chart

1. Define structural geometry with applied loads and constraints.
2. Initialize level set grid with signed distance function to represent structural geometry implicitly.
3. Trace the zero level set contours.

4. Perform mesh postprocessing and improvement.
5. Carry out boundary element analysis.
6. Compute velocity at each node point of the structural boundary using the BE analysis results.
7. Extend boundary velocities to level set grid points in the narrow band.
8. Solve Equation (2.3) to update the level set function.
9. Repeat the above procedure from step 3 to 8, until the stopping criterion is satisfied.

Most of the above steps are based on the simple extension of the steps followed in the 2D approach and has already been discussed in detail in Section 5.3. However, the extraction of the zero level set contours in 3D is different than that used in the 2D approach. In 3D LSM , the zero level set contours can be extracted from the cubic cell based level set grid with the MC algorithm (explained in Section 10.2.1). This extraction results into an iso-surface in the form of triangular mesh. As explained in Section 3.4.3, a 3D structure can be analysed with the BEM by first discretising its boundary into either triangular or quadrilateral elements. This is then followed by the solution of the equilibrium equation, i.e. Equation (3.80), and the calculation of the required properties at the nodal points. Therefore, with the proposed 3D optimisation method, at each iteration the modified structural geometry is already extracted in the form of a triangular mesh and this can be directly used for the BE analysis. However, mesh postprocessing (see Section 10.2.2) is always needed to make it consistent with the BE analysis requirements. The proposed method uses the 3D version of the BEM analysis software, i.e. 3D concept analyst [37]. Moreover, during the optimisation process, the structural geometry is continuously modified and this may result into some low quality triangular elements which can affect the accuracy of the BE solution. Therefore, in the current implementation, a mesh improvement step (see Section 10.2.3) is used to improve mesh quality. The following sections discuss the implementation details of step 3 and 4 of the proposed optimisation approach.

10.2.1 Algorithm for tracing the zero level set contours

In the literature there are various approaches to the surface generation problems [63]. The MC is the most popular algorithm for extracting iso-surfaces from implicit functions due to its simplicity, efficiency and robustness. It has been widely studied, improved, and extended. The initial MC algorithm described by Lorensen and Cline [67] constructs a piecewise linear approximation of the level set $\{\vec{x}(x, y, z) : \phi(\vec{x}) = \gamma\}$ [94], where γ represents the user specified iso-value. Bloomenthal [16] independently presented a numerical technique that approximates an implicit surface with a polygonal representation and is integrated with an octree approach to facilitate adaptive subdivisions. The surface which satisfies $\phi(\vec{x}) = \gamma$ is called the iso-surface (usually composed of a collection of triangles) [75].

The 3D level set grid is composed of cubic cells (voxels); Figure 10.2 shows a cubic cell with 8 vertices and 12 edges. The scalar values at the eight corners (grid points) of the cubic cell are used to decide whether the iso-surface is inside or outside of the cell. Grid points with scalar value less than or equal to γ are assumed inside or on the surface, and are assigned negative ($-$) label. Similarly, vertices with scalar value greater than γ are assumed outside, and are assigned positive ($+$) label. Based on this categorization one can easily determine edges of the cube intersected by the isosurface as explained in the following section.

(a) Triangle generation in a cubic cell

In the implementation of the MC algorithm the indexing convention used for vertices and edges is shown in Figure 10.2. A cubic cell having negative scalar value at vertex v_4 (represented with a solid sphere) and positive values at the other vertices is shown in Figure 10.3. In this particular case a triangular facet is generated which cuts edges e_3, e_4 and e_{12}, respectively.

Since each of the eight vertices of a cube can be labeled either ($-$) or ($+$), each cube has $2^8 = 256$ possible configurations. For a consistent facet combination, in each cube,

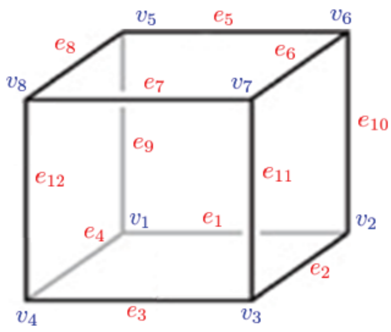


Figure 10.2: Vertices and edges indexing

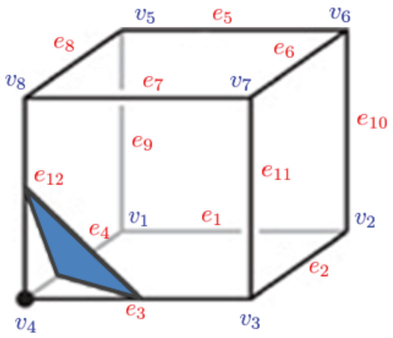


Figure 10.3: Cube triangular facet

the configuration of the triangular facets is determined from the intersection topology look-up table [17], which contains the edges intersected in each case. In the standard MC implementation, the 256 cases are derived from the 15 basic configurations as illustrated in Figure 10.4. Based on these configurations, i.e. from case 1 to 15, the MC algorithm can produce at least one and as many as four triangles per cube.

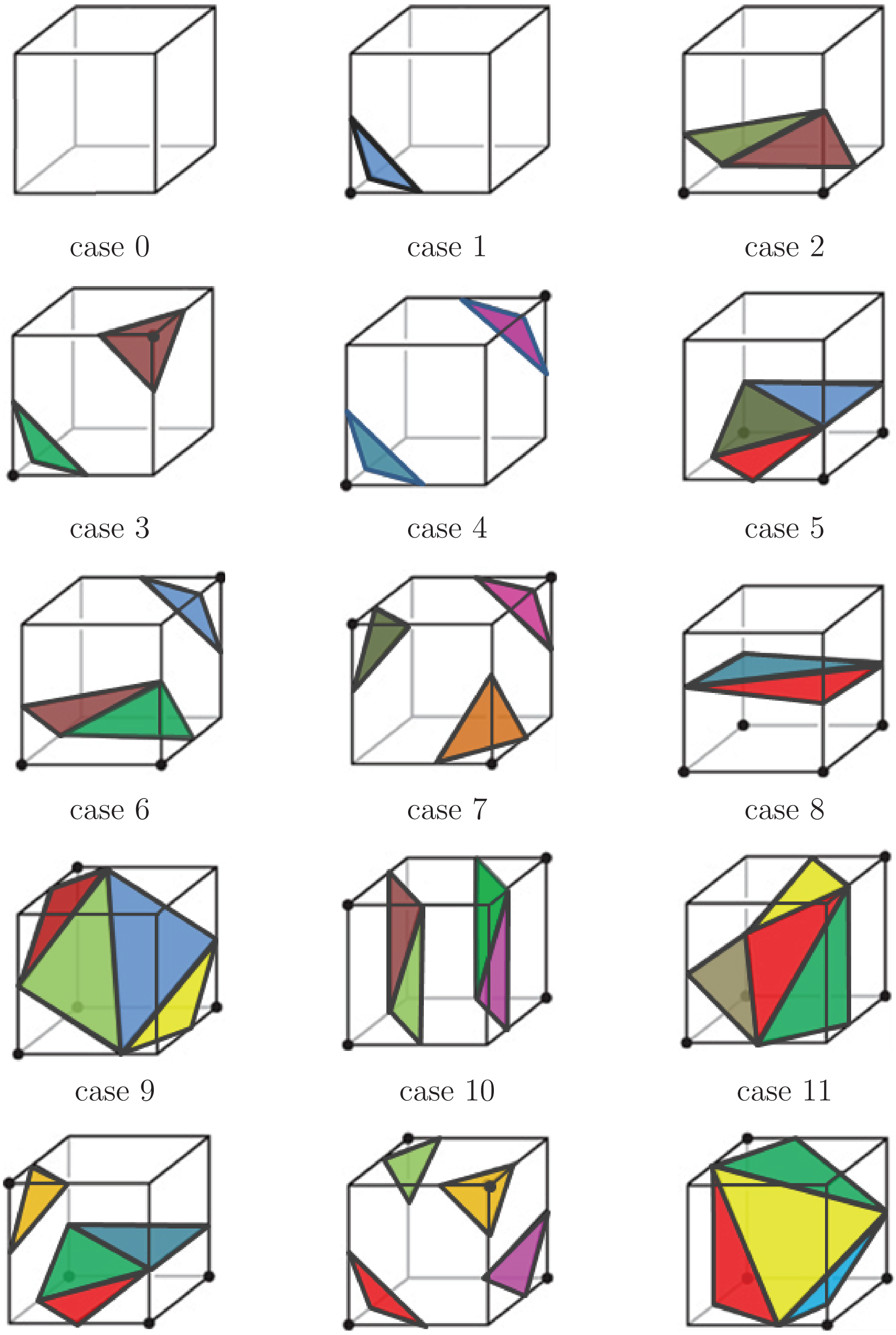
A postprocessing step is used at the end to form a closed iso-surface by connecting the common edges of the triangles in the entire volume. More details on the development, computational properties, extensions and limitations of the MC can be found in [75].

10.2.2 Mesh postprocessing

As explained in the previous step, the MC algorithm generates triangular facets (consisting of up to four) in each cube crossed by the iso-surface. In order to analyse the reconstructed geometry with the BEM, these individual facets need to be combined into a single closed iso-surface. Therefore, mesh postprocessing is used to convert the output mesh into a suitable form which can be directly used in the BE analysis. The main postprocessing steps are discussed below.

(a) Vertex connectivity

In the original MC implementation, the algorithm checks each cube of the level set grid crossed by the iso-surface. Triangular facets are then generated based on the iso-values at the cell vertices. In this step, common triangular vertices in adjacent cells are identified resulting into a closed iso-surface in the form of a triangular mesh M with vertex set



case 0
case 1
case 2
case 3
case 4
case 5
case 6
case 7
case 8
case 9
case 10
case 11
case 12
case 13
case 14

Figure 10.4: The 15 basic cases for Marching cubes

$V = \{1, 2, ..., n\}$. Moreover, the numbers of adjacent vertices of a given triangle vertex are also determined in this step. In the following steps, the vertex connectivity details are used to obtain other required properties of the triangular mesh.

(b) Find the adjacent triangles

Once we have the vertex connectivity details, the adjacent triangles(or the neighbours) to each triangle of the iso-surface are identified in this step.

(c) Correction of triangles orientation

The 3D concept analyst [37] is based on a counter clockwise orientation of the boundary elements. During the iso-surface extraction, some of the triangles generated, may be oriented clockwise. In this step, we use a counter clockwise oriented triangle as a reference and check all triangles of the iso-surface and correct those oriented clockwise.

(d) Calculate mid side nodes

Once properly oriented triangles are obtained, the mid side nodes are calculated for each of the triangle edges. The mid side node is calculated once for all triangles sharing an edge.

(e) Boundary conditions mapping

In the start of the optimisation process, the boundary conditions are assigned to the surface facets of the structural geometry. After the extraction of the modified structural geometry, the optimisation algorithm automatically maps the boundary conditions to those faces which overlap the bounding box for each set of the boundary conditions.

(f) Fix the constraint locations

The mesh improvement strategy (discussed below) used in this study modifies the positions of the vertices. Therefore, once the boundary conditions are assigned to M, the set of vertices are split into two groups. Vertices are marked as fixed, V_f, on which boundary conditions are prescribed and the remaining are marked as movable, V_m. Therefore, only the set of vertices in V_m are allowed to be modified during the mesh improvement step.

10.2.3 Mesh improvements

The quality of the surface representation extracted through the MC algorithm can be measured through many parameters [96]. A good surface mesh would contain mostly equilateral triangles. The most desirable one would reproduce the approximated surfaces with high accuracy and efficiency. As explained in Section 10.2.1 the MC algorithm generates up to four triangles within each cell. As a result, the extracted surface models usually contain many small triangles with poor quality (i.e. aspect ratio) [10]. In order to improve the quality of the output mesh (extracted through an MC algorithm) some additional measures are always required. This can be seen as an enhancement of the MC capabilities, and it can be effectively incorporated into the computational codes.

In order to obtain accurate stress results using the BEM, it is expected that the extracted mesh should be of suitable quality. Therefore, once the triangular mesh has been extracted, the element quality check is carried out in the next step. According to [116], the quality of an element of the triangular mesh can be assessed by the radius ratio, and it is defined as the ratio of the in-circle to the circum-circle of the element. Mathematically it can be written as,

$$Q = \frac{16A^2}{L_1 \times L_2 \times L_3(L_1 + L_2 + L_3)} \tag{10.1}$$

where A is the area and L_1, L_2 and L_3 are the side lengths of the element. The value of Q varies between 0 and 1, where 1 indicates the highest quality (or in other words, an equilateral triangle) and 0 a fully collapsed element. As per the accuracy of the structural analysis results it is required that every element is of quality $Q > Q_{min}$, where Q_{min} is the minimum acceptable quality of the element. In addition to the element quality, high curvature values have also been observed at some portions of the mesh which may result into stress concentrations at those regions. In addition, the overall quality can be assessed using the mean quality, $\bar{Q}$ [37],

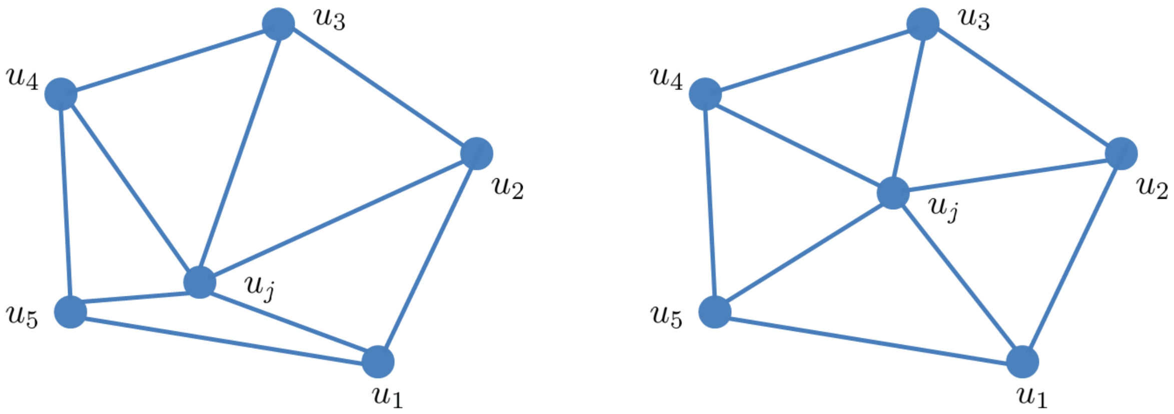
$$\bar{Q} = \frac{1}{N_E} \sum_{E=1}^{N_E} Q_E \tag{10.2}$$

In order to improve the mesh quality, as well as to minimise the stress concentration effects, smoothing techniques may be required [10]. The most commonly used techniques are Laplacian and HC-Laplacian smoothing. The following subsections discuss in detail the relevant theory, implementation and the associated advantages and disadvantages of these methods.

(a) Laplacian smoothing

A simple approach of mesh improvement often used is the Laplacian smoothing. Based on this approach, in a surface mesh, the new position of a vertex u_j is computed by averaging the location of the neighbouring vertices, i.e. $Adj(j)$ as shown in Figure 10.5 and given as,

$$u_j = \begin{cases} \frac{1}{|Adj(j)|} \sum\limits_{i \in Adj(j)} u_i & i \in V_m \\ u_i & i \in V_f \end{cases} \tag{10.3}$$

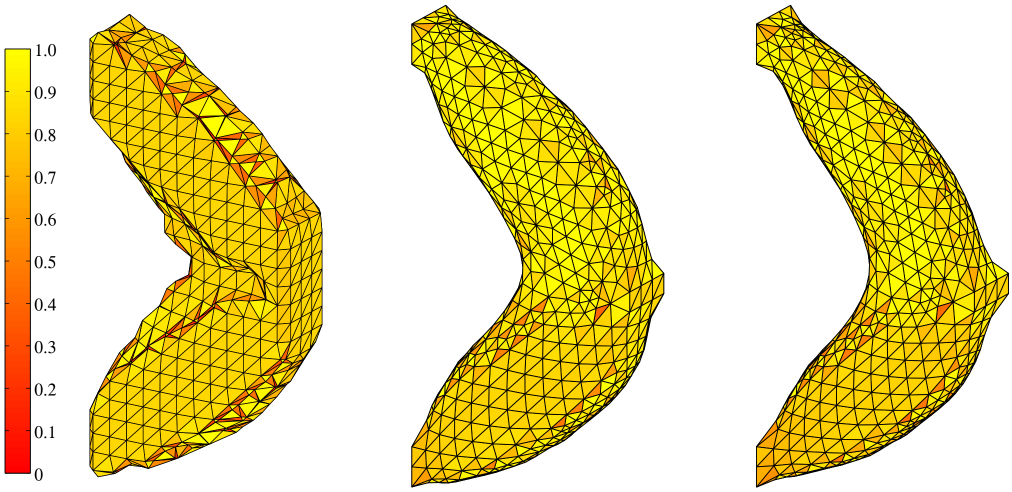


(a) Original mesh (b) Smoothed mesh

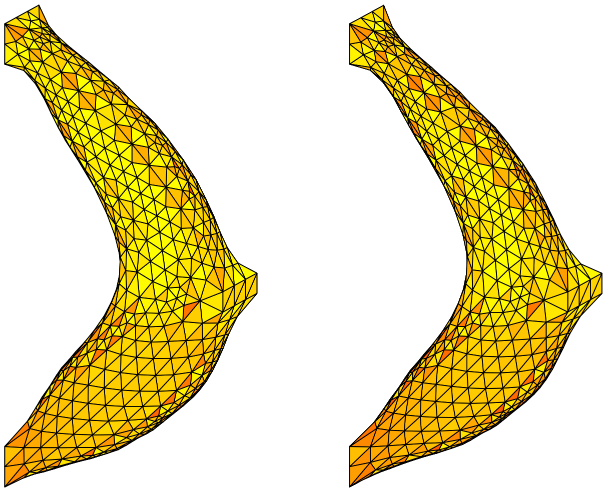
Figure 10.5: Laplacian smoothing

The new position of u_j can be calculated by two methods [120]. In the first method, known as the simultaneous version, the new positions are calculated for all u_i, i.e. using the same set of positions. The second method updates the u_i immediately after the new position calculation. This method is known as the sequential version. Therefore, in the second method the new position of u_i depends on both old and new positions. The results of the simultaneous version are better than the sequential one but requires more storage space for holding the old positions.

In order to evaluate the performance of the Laplacian smoothing algorithm the extracted mesh of a short cantilever beam (see Section 10.3.1) at iteration 10 is considered. This initial mesh has fixed vertices at the constrained locations and movable everywhere else. Figure 10.6 shows contours of the mesh quality Q calculated using Equation (10.1) after 0, 5,10, 15 and 20 iterations (smoothing steps) using the simultaneous version of the Laplacian algorithm. Low quality elements in the initial mesh depicted in Figure 10.6(a) can be clearly observed. This particular mesh consists of some fully collapsed elements with $Q = 0$. The results for 5,10, 15 and 20 smoothing steps demonstrate considerable volume shrinkage and therefore, it is not recommended to use the results of the Laplacian smoothing for further mesh quality checks. In addition, an increase in smoothing steps decreases the element quality, though the curvature across the adjacent elements is minimised.



(a) Step 0 (volume = 100%) (b) Step 5 (volume = 75.28%) (c) Step 10 (volume = 59.16%)



(d) Step 15 (volume = 47.68%) (e) Step 20 (volume = 39.19%)

Figure 10.6: Mesh quality Q after 0, 5, 10, 15 and 20 Laplacian smoothing steps

(b) HC-Laplacian smoothing

In order to reduce the shrinkage effects of Laplacian smoothing, Vollmer *et al.* [120] introduced the HC-Laplacian algorithm. Based on this modified approach, the vertices moved by the Laplacian smoothing are pushed back towards their previous positions u_i. The algorithm calculates the magnitude and direction of the backward movements from the original and previous vertex location using the weight ψ and the mean displacement vector in the neighborhood using vector β. Details of the HC-Laplacian algorithm are given in Figure 10.7.

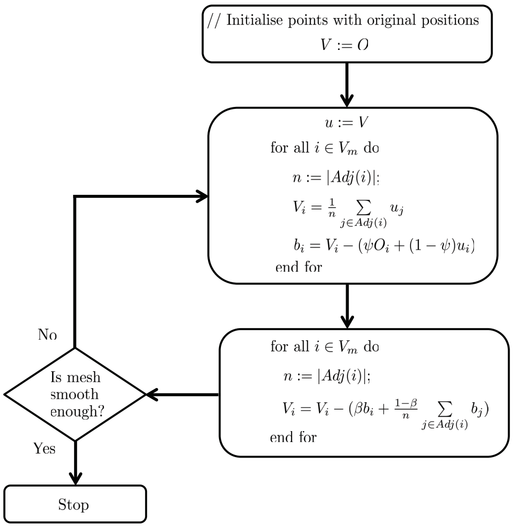
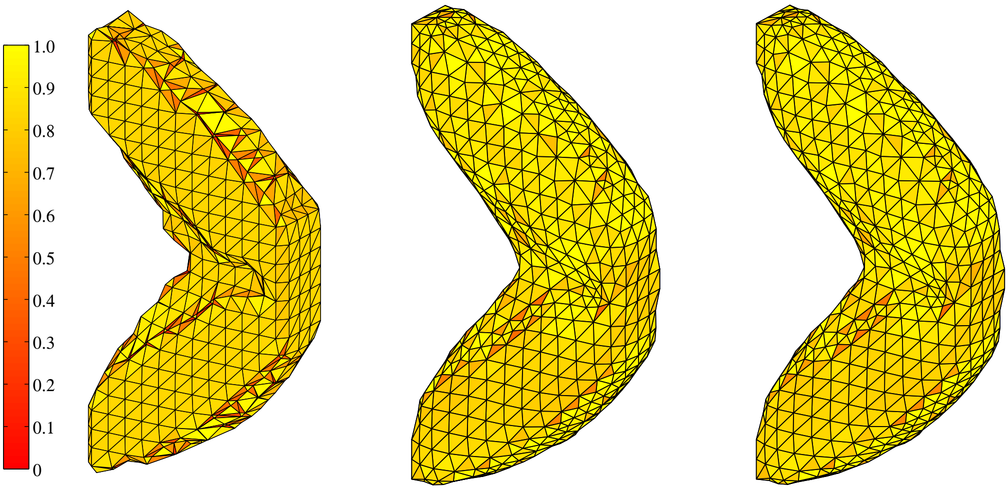


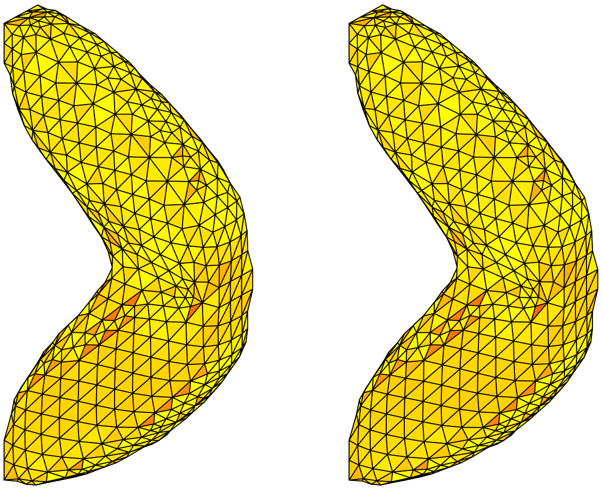
Figure 10.7: HC-Laplacian smoothing

Several iterations of the HC-Laplacian algorithms results in a sufficiently smoother mesh with no or low volume shrinkage. The factors ψ and β used during the smoothing steps are 0.1 and 0.2 (according to [120]), respectively. Figure 10.8 shows the mesh quality Q calculated using Equation (10.1) after 0, 5,10, 15 and 20 smoothing steps of the HC-Laplacian algorithm. It is evident from the results that the HC-Laplacian algorithm improves the element quality with little volume shrinkage. Additional mesh quality measurements with the HC-Laplacian algorithm with different smoothing steps are displayed in Table 10.1.

It is evident from the data presented in Table 10.1 and Figure 10.8, that the HC-Laplacian algorithm significantly improves the individual element quality as well as the



(a) Step 0 (volume = 100%) (b) Step 5 (volume = 98.56%) (c) Step 10 (volume = 97.91%)



(d) Step 15 (volume = 97.55%) (e) Step 20 (volume = 97.36%)

Figure 10.8: Mesh quality Q after 0, 5, 10, 15 and 20 HC-Laplacian smoothing steps

Steps	Total elements	Number of elements with Q_E				Q_{min}	$\bar{Q}$
		$0 - 0.5$	0.51-0.7	$0.71 - 0.8$	$0.81 - 1.0$		
0	1424	272	98	141	913	0.0	0.718
5	1424	16	66	114	1228	0.377	0.878
10	1424	8	73	89	1254	0.413	0.884
15	1424	8	71	81	1264	0.415	0.885
20	1424	7	73	80	1264	0.417	0.886

Table 10.1: Element quality data for HC-Laplacian smoothing

overall mesh quality. Bade *et al.* [10] also used the reduction in the mean curvature of the mesh as a smoothing criterion. Based on their results the maximum mean curvature of the surface decreases with an increase in the smoothing steps. On the other hand, the HC-

Laplacian algorithm is not completely shrinkage free. Therefore, an optimum number of smoothing steps should be selected to obtain a good quality mesh with smoother geometry and low volume shrinkage. Based on the results presented in Table 10.1, Figure 10.8 and in [10], a good choice would be to use 15 smoothing steps in each optimisation iteration. It should be noted that the low volume shrinkage does not affect the structural geometry during the optimisation process. This is because that the mesh smoothing only modifies the extracted geometry and does not alter the level set function from which this geometry has been extracted.

10.2.4 Three-dimensional shape optimisation

The use of the level set method in three-dimensional optimisation has several advantages. The first one is related to its natural extension for two to three-dimensional space [103]. Another important advantage is that, during the optimisation process new hole automatically appears by the intersection of two surfaces moving towards each other [8]. Therefore, the hole insertion mechanism can be eliminated in 3D LSM based optimisation. In a BEM based shape optimisation, all the design variables are available at the structural boundary, and there is no need to use internal points within the design domain or to use the volumetric mesh. This considerably reduces the computational efforts and accelerates the optimisation process. Moreover, the BE analysis provides more accurate boundary stress calculations as compared to the FEM [105, 109].

The proposed 3D optimisation approach is based on the extension of the 2D shape optimisation approach presented in Section 5.3. In the first step boundary velocities are calculated for each node point using the stress velocity relationship as depicted in Figure 5.12. The velocities are then extended to the grid points in the narrow band around the boundary, using the methods developed by Adalsteinsson and Sethian [4, 5]. The boundary segments with constraints and loads, are assigned with zero velocity before the velocity extension; this prevents these locations from movement during the optimisation process. The level set function is re-initialised by the substitution of the temporary signed distance function (computed during the velocity extension method) for the current level set function. This provides a very fast and accurate way of re-initialisation of the level set function in the narrow band [103].

After the velocity extension, the level set function is updated by the solution of Equation 2.3 with an upwind finite difference approximation. The value of the time step used in Equation 2.3 is based on the CFL condition.

10.3 Examples

The validity and efficiency of the proposed optimisation method are tested against some benchmarking problems in the field of structural optimisation. The material properties used in these examples are: Poisson's ratio = 0.3, Young's modulus = 210 GPa, Yield stress = 280 MPa. In all examples, the optimisation process starts with evolutionary parameters, $RR = 0.01$ and $RR_i = 0.01$, unless otherwise stated. In order to capture all possible boundary movements during the numerical implementation of the LSM, an additional row of cells is provided on each side of the level set grid.

10.3.1 Example-1

In the first example, a short cantilever beam has been considered with dimensions, $L = 24$, $W = 8$ and $H = 48$. The geometry of the structure shown in Figure 10.9, is constrained at the top and bottom portions of the left face, and a load $P = 1.2$KN is applied at the middle of the right face. The level set design domain is discretised into $12 \times 4 \times 24$ cubic cells with edge length $d = 2$. The target volume fraction α used in this example is 0.30.

In order to investigate the effect of stress criterion values on the final optimal solution, four different cases are considered in this example. In case 1, the short cantilever beam is optimised with the maximum von Mises stress in the initial design, i.e. $\sigma_{Vmax} = 178$ MPa. The evolution of the structural geometry during the optimisation process is depicted in Figure 10.10. It can be seen that during the initial iterations, the structural geometry evolves through boundary movements caused by the incremental removal of inefficient material from the low stressed regions of the structure. In contrast to the 2D LSM based optimisation method, the present 3D method is capable of automatically nucleating holes

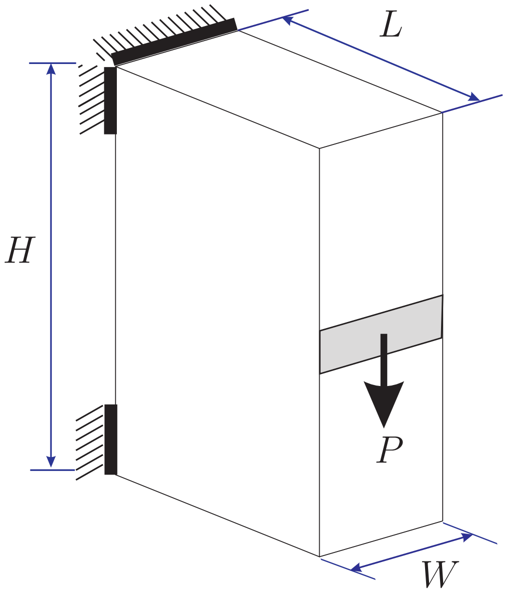
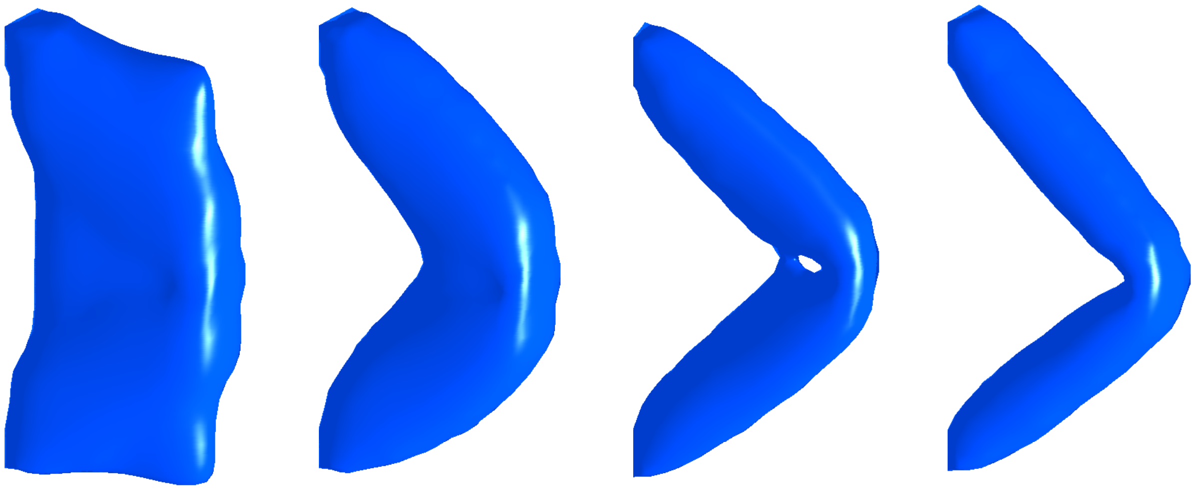


Figure 10.9: Design domain, loading and boundary conditions for Example-1

at low stressed regions as shown in Figure 10.10(c). In the following iterations, the geometry evolves further and as a result the hole merges with the outer boundary leading the structural geometry into an optimal design. The optimal design obtained closely resemble to that obtained in [20].



(a) Iteration 2 ($\alpha = 0.77$)

(b) Iteration 10 ($\alpha = 0.52$)

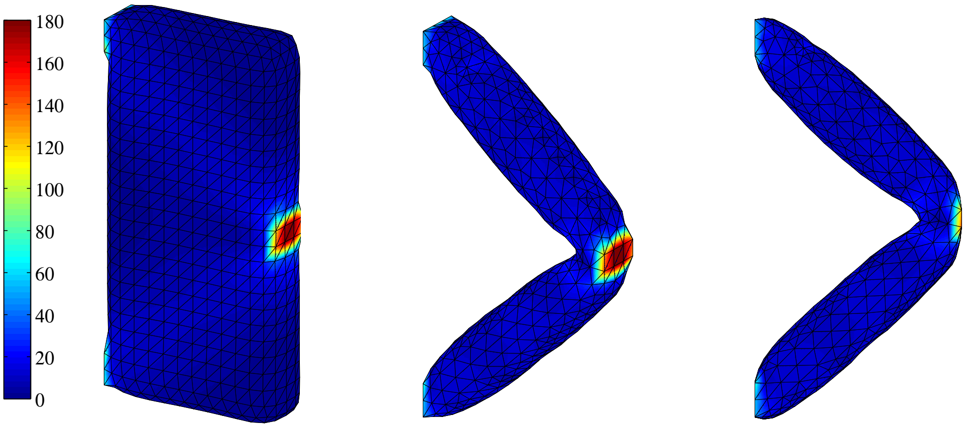
(c) Iteration 24 ($\alpha = 0.37$)

(d) Iteration 37 ($\alpha = 0.30$)

Figure 10.10: Evolution of structural geometry for Example-1, case 1

Figure 10.11 shows the von Mises stress distribution in the initial and optimal design.

There are 1724 six-noded triangular elements in the initial and 1200 in the optimal design. Comparison of these plots shows that the stress contours are more uniform in the optimal design than the initial design. This indicates that the optimisation method efficiently redistributes material within the design domain and results in an optimal which is approaching a fully stressed design.



(a) Initial design (b) Optimal design (c) Optimal design front view

Figure 10.11: von Mises stress contours of initial and final optimal geometry for Example-1, case 1

During the optimisation process the evolution of the specific strain energy is recorded at each optimisation iteration and is depicted in Figure 10.12. It is evident from the results displayed that in the initial iterations the optimisation method removes the inefficient material rapidly and this results in a rapid decrease in f_U up to iteration 7. In the following iterations, the specific strain energy further decreases slowly until the target volume fraction is reached and the optimisation process terminates.

In cases 2 and 3, the optimisation problem is solved for $\sigma_{Vmax} = 100$ MPa and $\sigma_{Vmax} = 50$ MPa, respectively. Figures 10.13 and 10.14 display the evolutions of the structural geometry in both cases. It is evident from the results obtained that the evolution of the structural geometry in these two cases is identical to that presented in case 1. However, it also shows that a reduction in the stress criterion value decreases the material removal rate, and as a result more optimisation iterations are required to achieve the same target volume fraction (i.e 54 and 107 in case 2 and 3, respectively). Comparison of the results

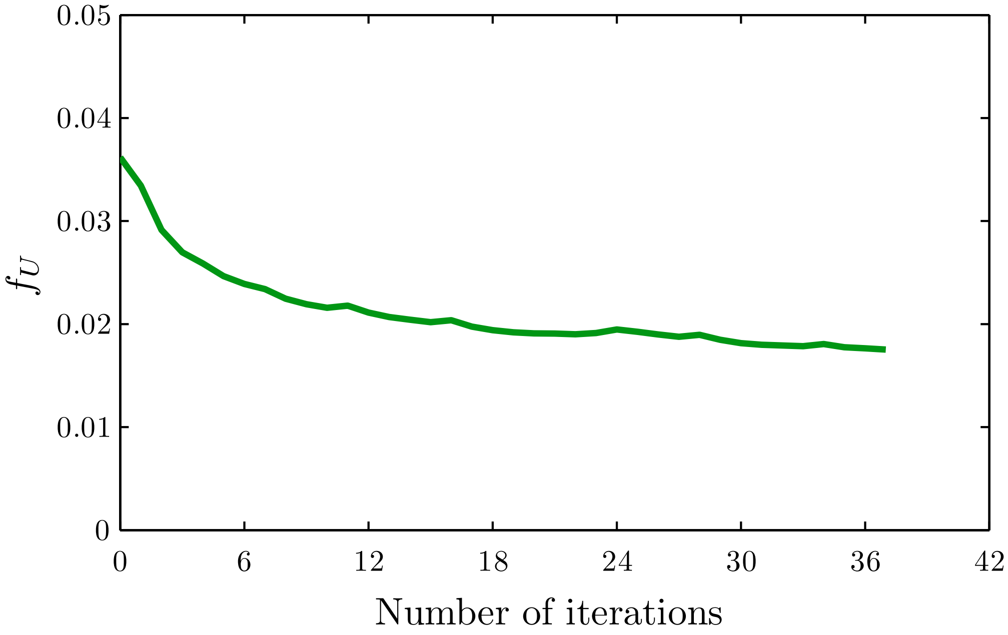


Figure 10.12: Evolution of f_U for Example-1, case 1.

obtained in all three cases suggests that the optimal designs are identical and in addition, independent of the stress criterion values.

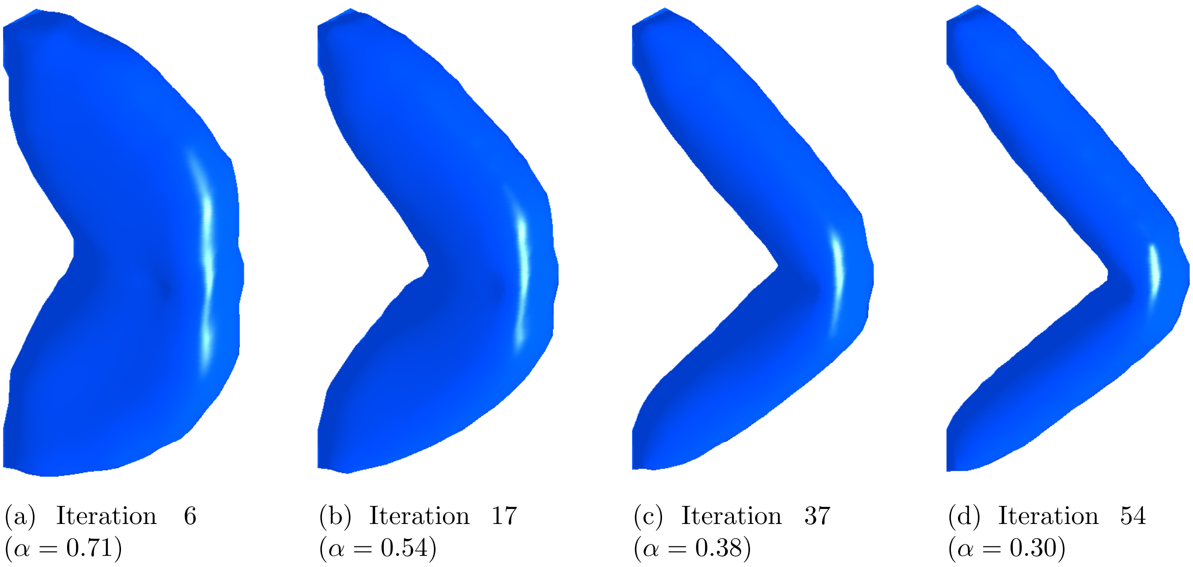
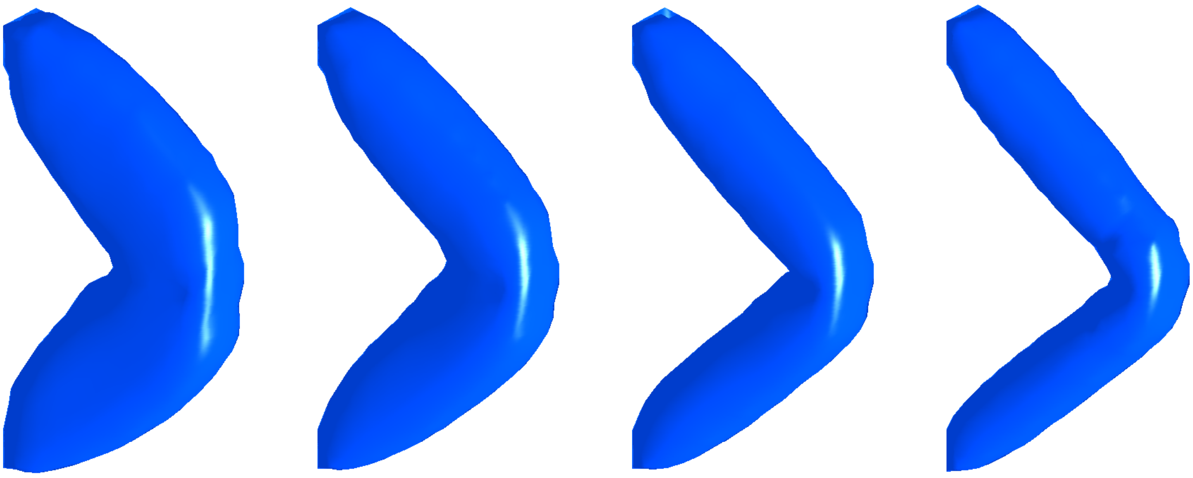

(a) Iteration 6 ($\alpha = 0.71$)
(b) Iteration 17 ($\alpha = 0.54$)
(c) Iteration 37 ($\alpha = 0.38$)
(d) Iteration 54 ($\alpha = 0.30$)

Figure 10.13: Evolution of structural geometry for Example-1, case 2

Figures 10.15 and 10.16 show the von Mises stress contours plots of the initial and optimal designs for case 2 and 3, respectively. In both cases, the optimisation process



(a) Iteration 20 ($\alpha = 0.56$)

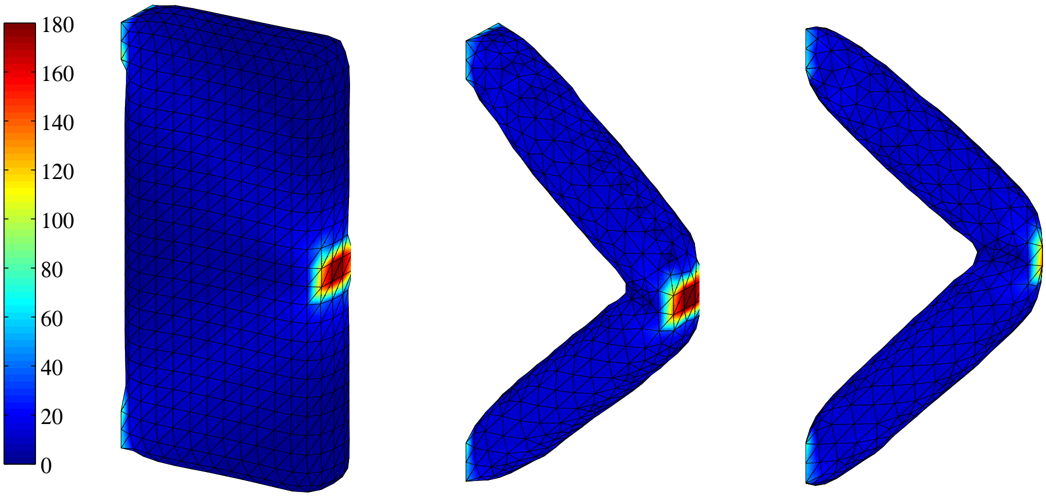
(b) Iteration 35 ($\alpha = 0.44$)

(c) Iteration 68 ($\alpha = 0.37$)

(d) Iteration 107 ($\alpha = 0.30$)

Figure 10.14: Evolution of structural geometry for Example-1, case 3

starts from the same initial design having 1724 six-noded triangular elements. The optimal designs obtained in the previous two cases have 1192 and 1188 six-noded triangular elements, respectively. The results obtained with different stress criterion levels demonstrate that the stress contours are uniform in both cases, and the optimal designs are approaching towards the fully stressed design.

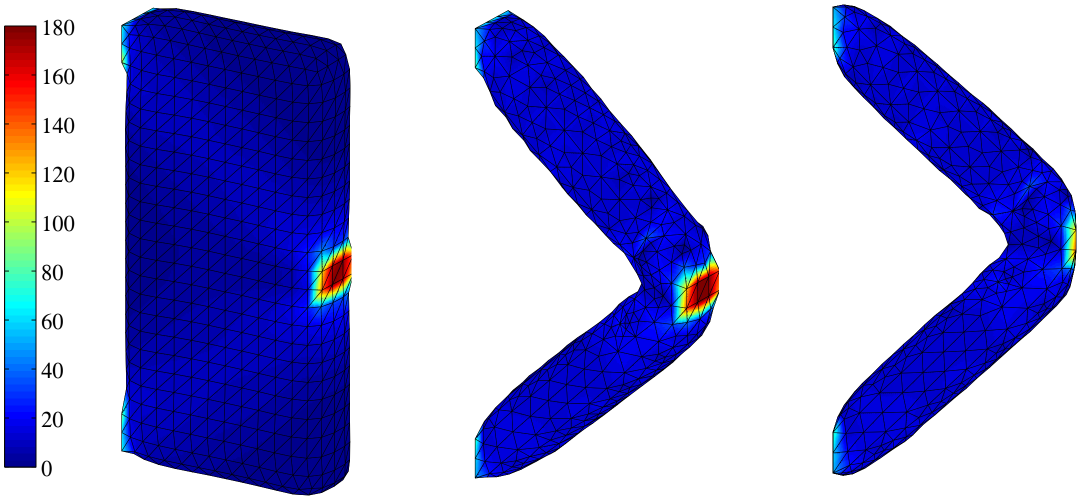


(a) Initial design

(b) Optimal design

(c) Optimal design front view

Figure 10.15: von Mises stress contours of initial and final optimal geometry for Example-1, case 2



(a) Initial design (b) Optimal design (c) Optimal design front view

Figure 10.16: von Mises stress contours of initial and final optimal geometry for Example-1, case 3

Similarly, Figures 10.17 and 10.18 display the specific strain energy recorded during the evolution of cases 2 and 3. It is evident from the results displayed that in both cases, f_U follows the same trend as observed in the first case. Furthermore, in all cases, the optimisation process terminates at nearly the same values as shown in Table 10.2.

Case	Total iterations	Final f_U
1	37	0.0175
2	54	0.0188
3	107	0.0175

Table 10.2: Comparison of f_U for Example-1

Finally, in case 4, the optimisation problem has been solved with a load $P = 100$N. The von Mises stress contours shown in Figure 10.19 demonstrate a similar optimum design as obtained in the previous three cases.

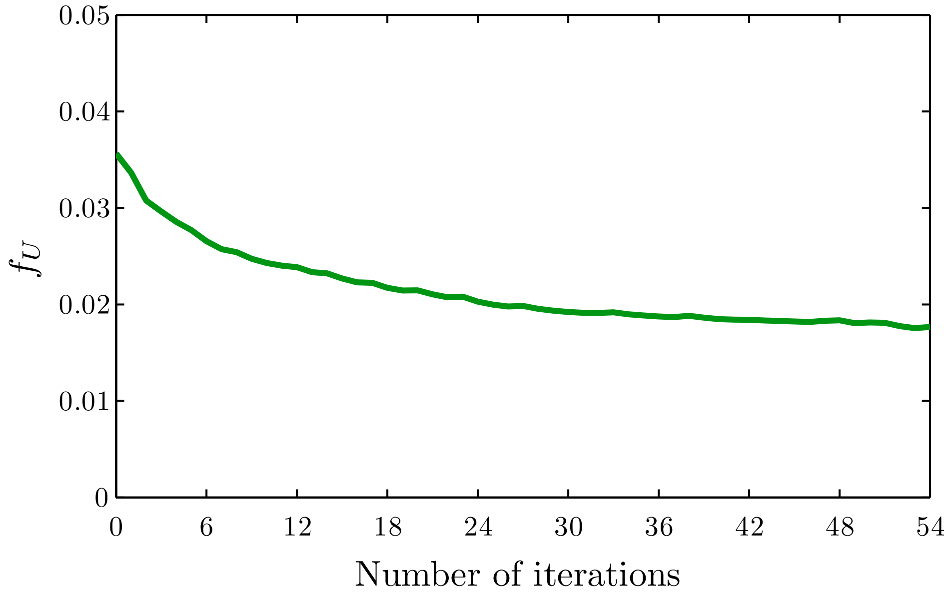


Figure 10.17: Evolution of f_U for Example-1, case 2

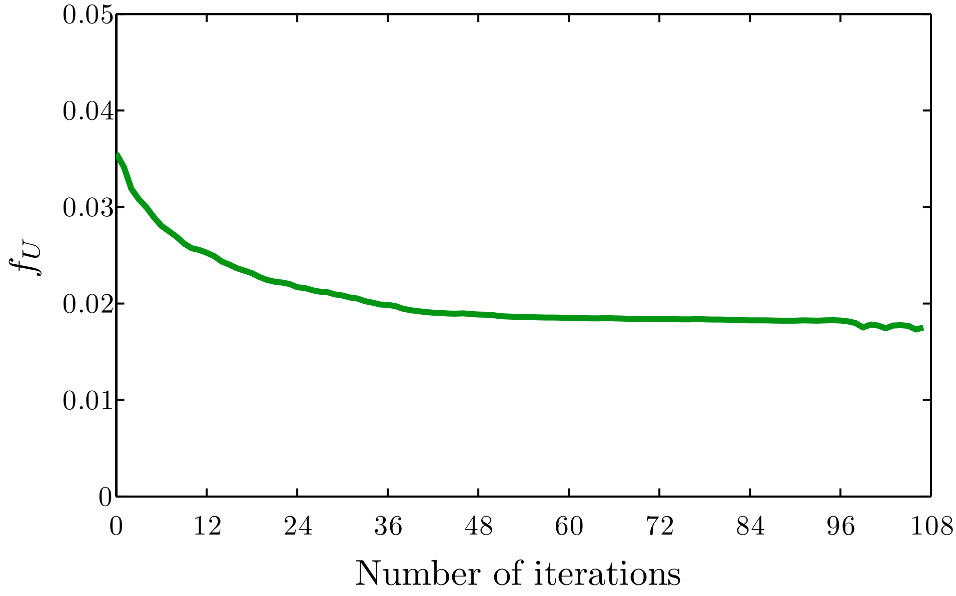
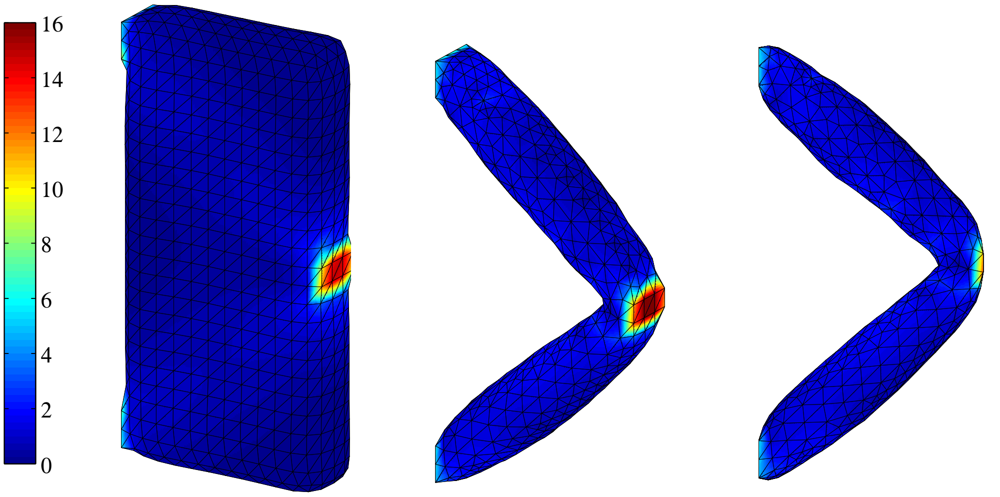


Figure 10.18: Evolution of f_U for Example-1, case 3



(a) Initial design (b) Optimal design (c) Optimal design front view

Figure 10.19: von Mises stress contours of initial and final optimal geometry for Example-1, case 4

10.3.2 Example-2

The second example considered in this study is a short cantilever beam with dimensions $L = 40$, $W = 8$ and $H = 40$. The structural geometry shown in Figure 10.20 is constrained at the top and bottom portions of the left face, and a load $P = 2.4$KN is applied at the end of the bottom face. The level set design domain is discretised into $20 \times 4 \times 20$ cubic cells with edge length $d = 2$. The optimisation problem is solved for $\alpha = 0.35$.

Similar to Example-1, the effect of stress criterion values has been investigated with two cases in this example. In case 1, the optimisation problem is first solved with the maximum von Mises stress in the initial design, i.e. $\sigma_{Vmax} = 260$ MPa. The evolution of structural geometry is depicted in Figure 10.21. The use of ESO and LSM allows progressive removal of inefficient material through boundary movements only and automatically nucleate holes at the low stressed regions of the structure. Throughout the optimisation process, holes appear, evolve and merge together and in this way leading the structural geometry towards the target volume fraction. The optimal design obtained is similar to that available in [20].

The von Mises stress contour plots of the short cantilever beam for the initial and final

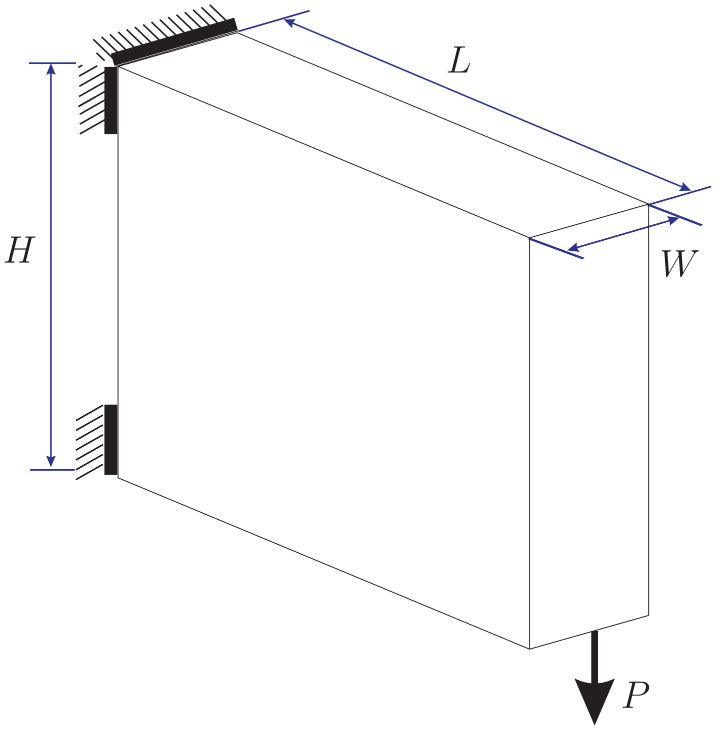
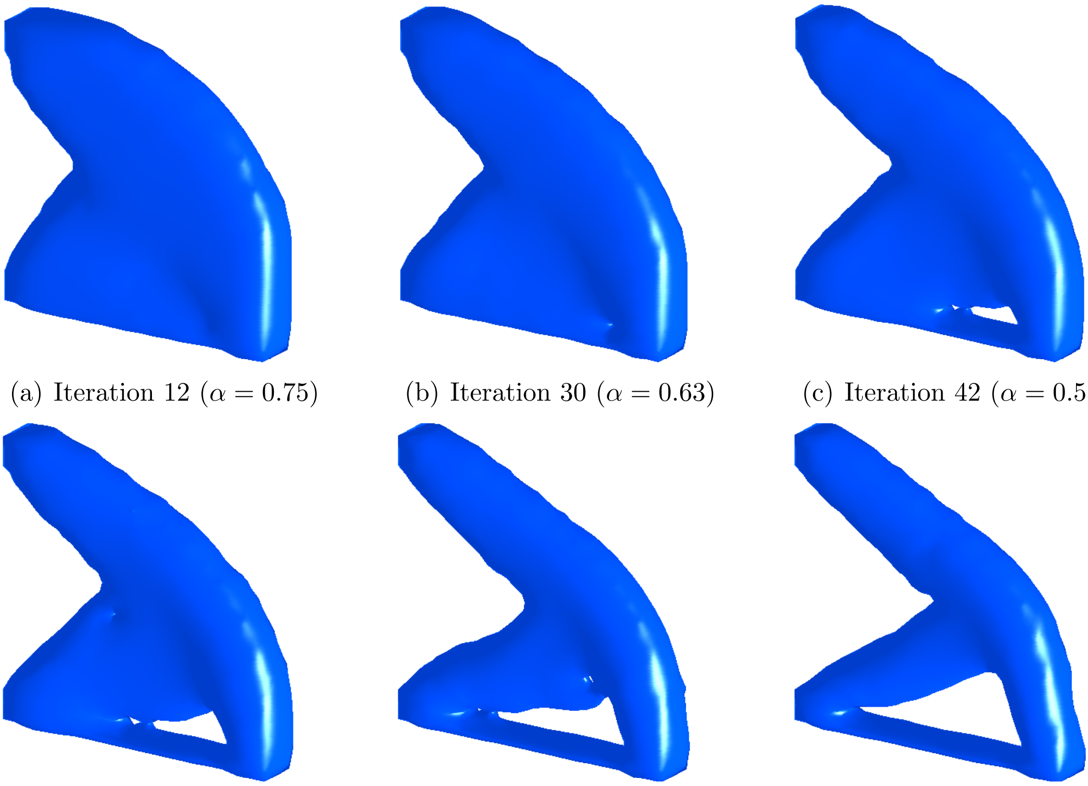


Figure 10.20: Design domain, loading and boundary conditions for Example-2

designs are displayed in Figure 10.22. There are 2236 six-noded triangular elements in the initial design and 1900 in the optimal design, respectively. It is evident from the results displayed that the proposed optimisation method efficiently redistributes material within the design domain and thus provides an optimal geometry with consistent and uniform stress distribution. In addition, the optimal design is approaching towards a fully stressed design.

During the evolution of structural geometry, the specific strain energy recorded at each iteration is depicted in Figure 10.23. In the initial iterations, a relatively high material removal rate rapidly decreases f_U. Once most of the inefficient material is removed, a slow decrease can be observed after iteration 40. Finally, the optimisation process terminates at the target volume fraction with $f_U = 0.290$.

In case 2, the optimisation problem is solved with $\sigma_{Vmax} = 100$MPa which is lower than the maximum von Mises stress of the initial design. Figure 10.24 displays the evolution of the structural geometry at different stages of the optimisation process. It can be seen that the structural geometry evolves in a similar way as observed in the previous case of



(a) Iteration 12 ($\alpha = 0.75$) (b) Iteration 30 ($\alpha = 0.63$) (c) Iteration 42 ($\alpha = 0.55$)

(d) Iteration 52 ($\alpha = 0.48$) (e) Iteration 62 ($\alpha = 0.39$) (f) Iteration 70 ($\alpha = 0.35$)

Figure 10.21: Evolution of structural geometry for Example-2, case 1

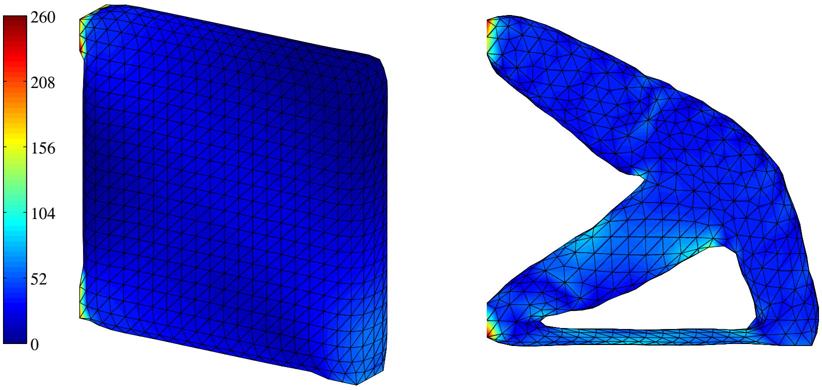
this example. However, with a lower value of the stress criterion, the same target volume fraction has been achieved in more iterations than than that in the case 1 of this example.

The evolution of f_U has also been recorded in the case 2 of this example and displayed in Figure 10.25. The results exhibit a similar trend in the evolution of f_U as observed in the previous case (Table 10.3), and the optimisation process terminates with $f_U = 0.281$, which is 3% lower than that recorded in the first case.

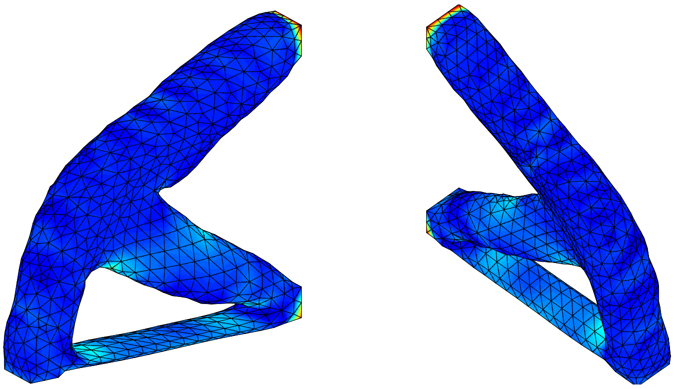
Case	Total iterations	Final f_U
1	70	0.290
2	114	0.280

Table 10.3: Comparison of f_U for Example-2

Figure 10.26 displays the von Mises stress contour plots with 2236 six-noded triangular elements in the initial and 1952 in the optimal design. Similar to the previous case of this



(a) Initial design (b) Optimal design view 1



(c) Optimal design view 2 (d) Optimal design view 3

Figure 10.22: von Mises stress contours of initial and optimal geometry for Example-2, case 1

example, the present case also exhibits a uniform stress distribution within the optimal design domain.

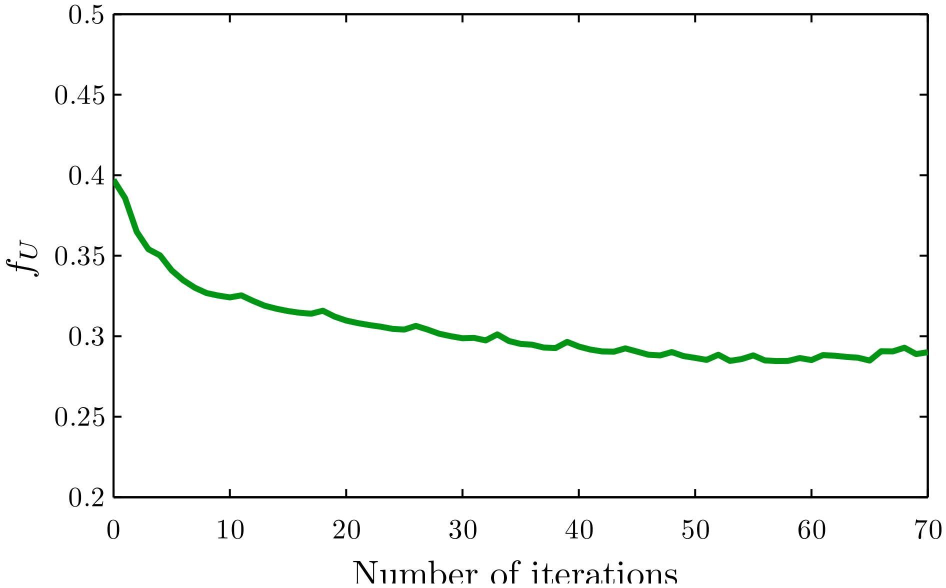


Figure 10.23: Evolution of f_U for Example-2, case 1

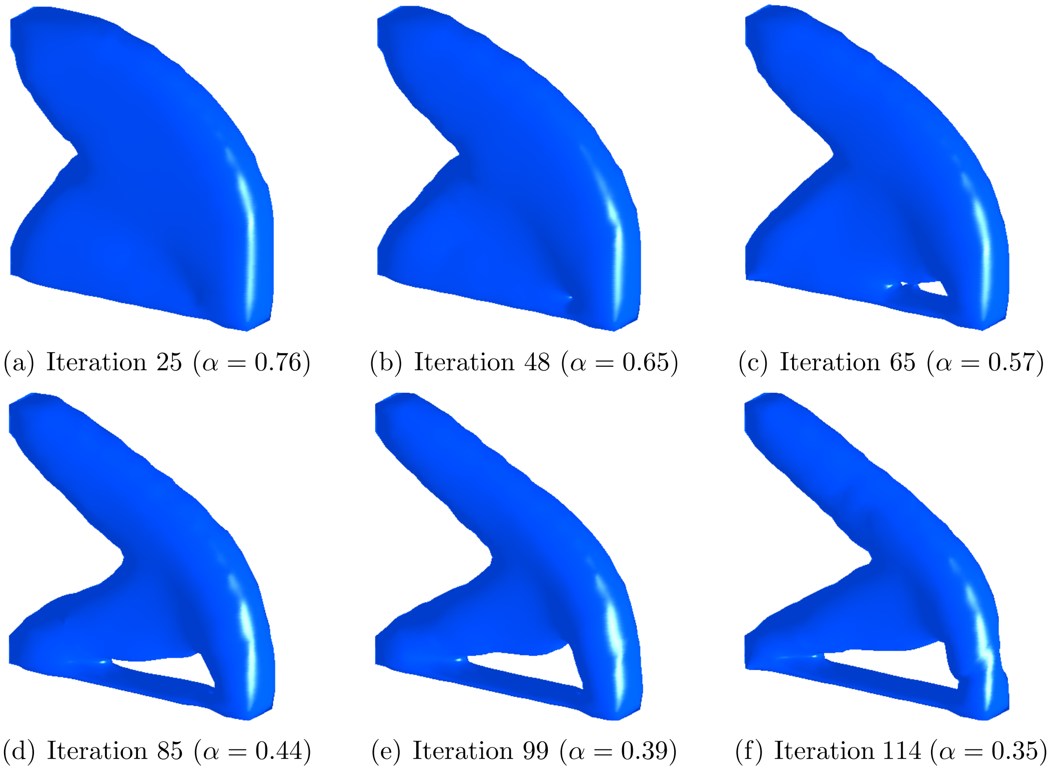

(a) Iteration 25 ($\alpha = 0.76$)
(b) Iteration 48 ($\alpha = 0.65$)
(c) Iteration 65 ($\alpha = 0.57$)
(d) Iteration 85 ($\alpha = 0.44$)
(e) Iteration 99 ($\alpha = 0.39$)
(f) Iteration 114 ($\alpha = 0.35$)

Figure 10.24: Evolution of structural geometry for Example-2, case 2

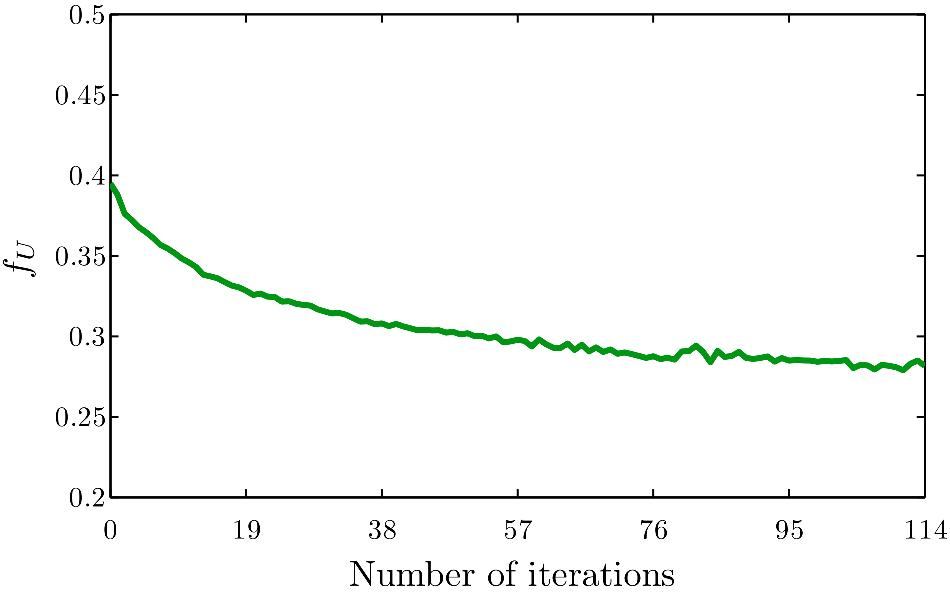
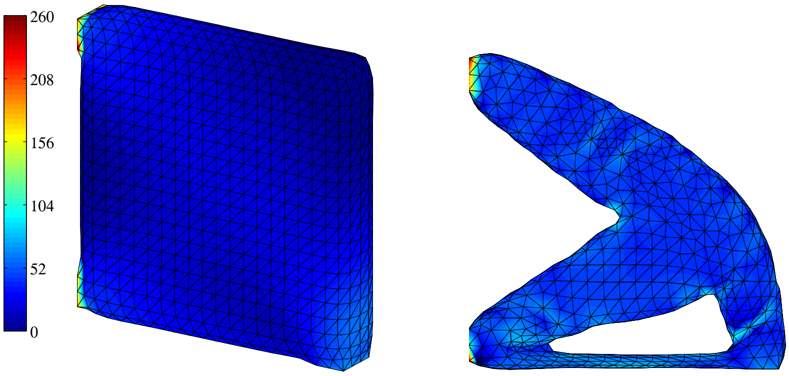
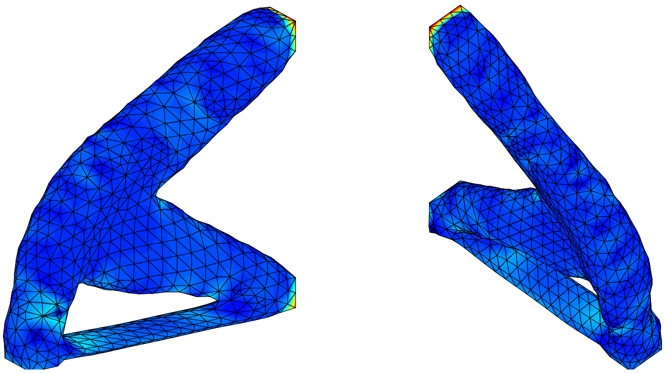


Figure 10.25: Evolution of f_U for Example-2, case 2



(a) Initial design (b) Optimal design view 1



(c) Optimal design view 2 (d) Optimal design view 3

Figure 10.26: von Mises stress contours of initial and optimal geometry for Example-2, case 2

10.3.3 Example-3

The proposed optimisation method is further tested against another benchmark problem of a short cantilever beam with dimensions $L = 40$, $W = 8$ and $H = 24$. The structural geometry and loading/boundary conditions are shown in Figure 10.27. The optimisation problem is solved with $P = 1.2$KN and $\alpha = 0.30$. The level set design domain is discretised into $20 \times 4 \times 12$ cubic cells with edge length $d = 2$.

Similarly, two different cases are considered in this example. In case 1, the optimisation problem is first solved with a maximum von Mises stress in the initial design, i.e. $\sigma_{Vmax} =$ 185 MPa. Figure 10.28 shows the evolution of the structural geometry at different stages of the optimisation process. Inefficient material is progressively removed from the design domain in all dimensions through boundary movements, which allows automatic hole

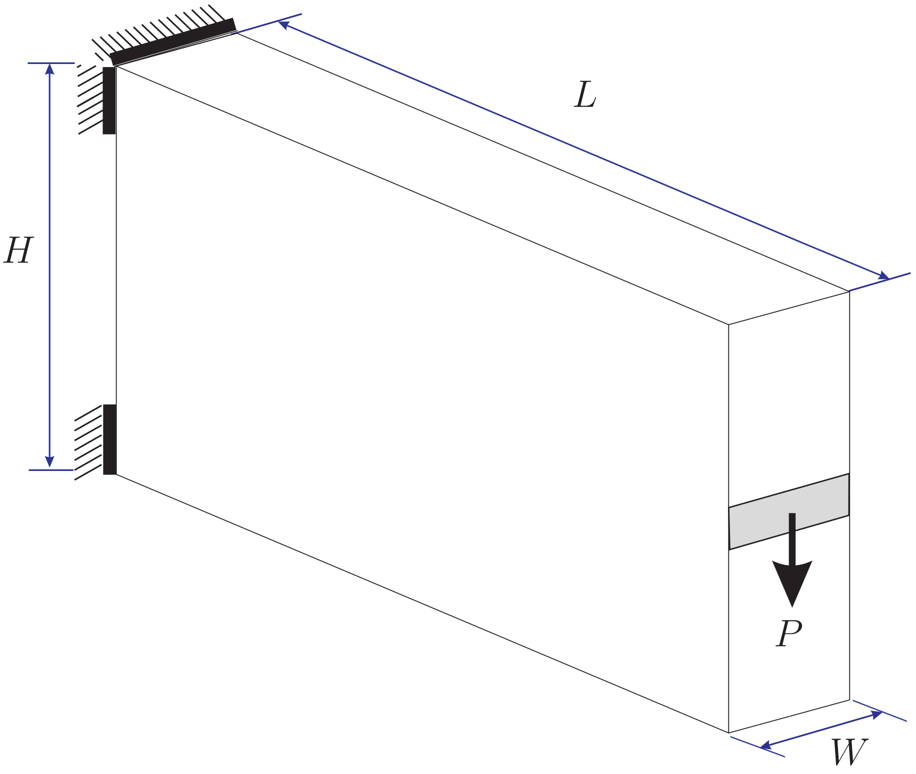
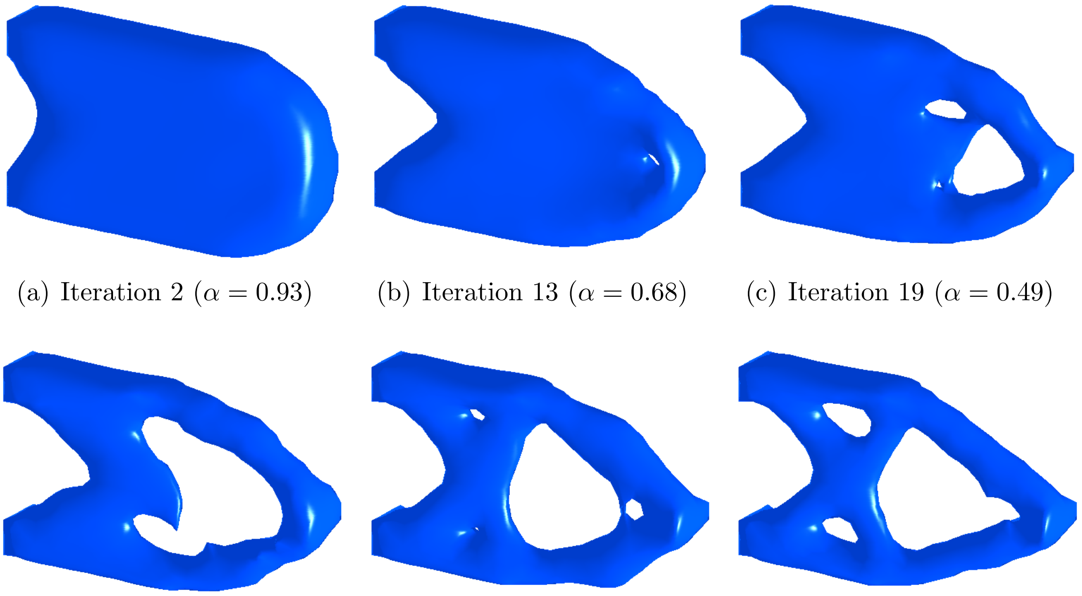


Figure 10.27: Design domain, loading and boundary conditions for Example-3

nucleation at the low stressed regions of the structure. During the optimisation process, holes appear, evolve and merge together and in this way leading the structural geometry towards the target volume fraction.

The von Mises stress contour plots for the initial and final designs are displayed in Figure 10.29. There are 1468 six-noded triangular elements in the initial design and 1332 in the optimal design, respectively. It is evident from the results that the proposed optimisation method efficiently redistributes material within the design domain and thus provides an optimal geometry, which is approaching a fully stressed design.

During the evolution of structural geometry, the specific strain energy recorded at each iteration is depicted in Figure 10.30. In the initial iterations, a slow decrease can be observed up to iteration 16. In the following iterations hole insertion takes place, which increases f_U and results in high peak around iteration 22. The effect of this peak dies out with the evolution of structural geometry in the subsequent iterations. Finally, the optimisation process terminates at the target volume fraction with $f_U = 0.144$.



(a) Iteration 2 ($\alpha = 0.93$) (b) Iteration 13 ($\alpha = 0.68$) (c) Iteration 19 ($\alpha = 0.49$)

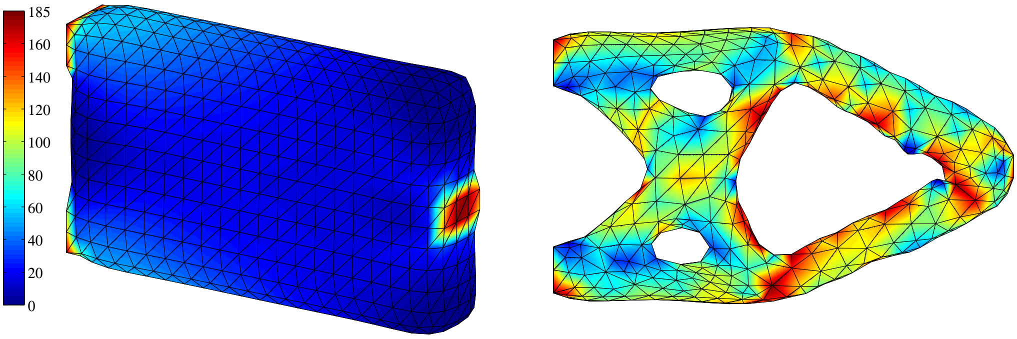
(d) Iteration 22 ($\alpha = 0.40$) (e) Iteration 25 ($\alpha = 0.34$) (f) Iteration 28 ($\alpha = 0.28$)

Figure 10.28: Evolution of structural geometry for Example-3, case 1

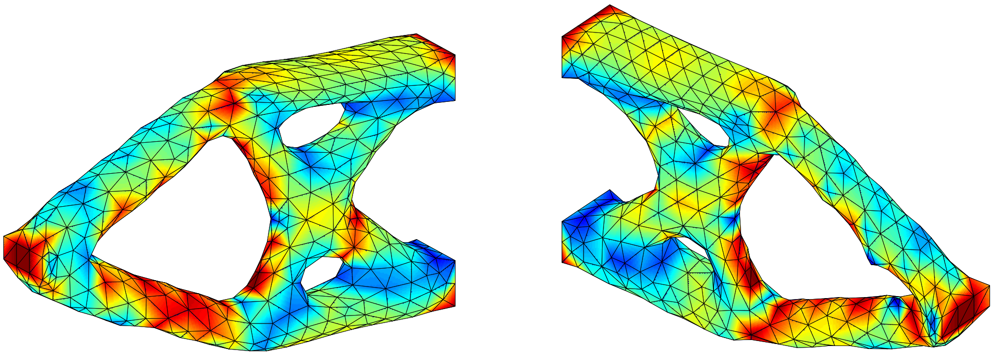
In case 2, the optimisation problem is further solved with $\sigma_{Vmax} = 165$ MPa and the results obtained at different stages of the optimisation process are depicted in Figure 10.31. Throughout the optimisation iterations, the structural boundary evolves in all directions and this allows hole nucleation at the low stressed regions of the structure. Following this, holes appear, evolve and merge together until the stopping criterion is satisfied and optimisation process terminates. The optimal design obtained closely resembles that available in the literature of this type of bench mark example.

Figure 10.32 shows the evolution of f_U at each iteration of the optimisation process. Apart from the interval between iteration 18 to 30, a smooth progression can be observed during the evolution of f_U. The interval between iteration 18 to 30 is mainly related to the hole insertion and afterwards merging (as shown in 10.31(d) and (e)), and this resulted in high peaks during this interval.

Figure 10.33 shows the von Mises stress distribution in the initial and optimal design. There are 1468 six-noded triangular elements in the initial and 1384 in the optimal design. Comparison of these plots shows that the stress contours are more uniform in the



(a) Initial design (b) Optimal design view 1



(c) Optimal design view 2 (d) Optimal design view 3

Figure 10.29: von Mises stress contours of initial and optimal geometry for Example-3, case 1

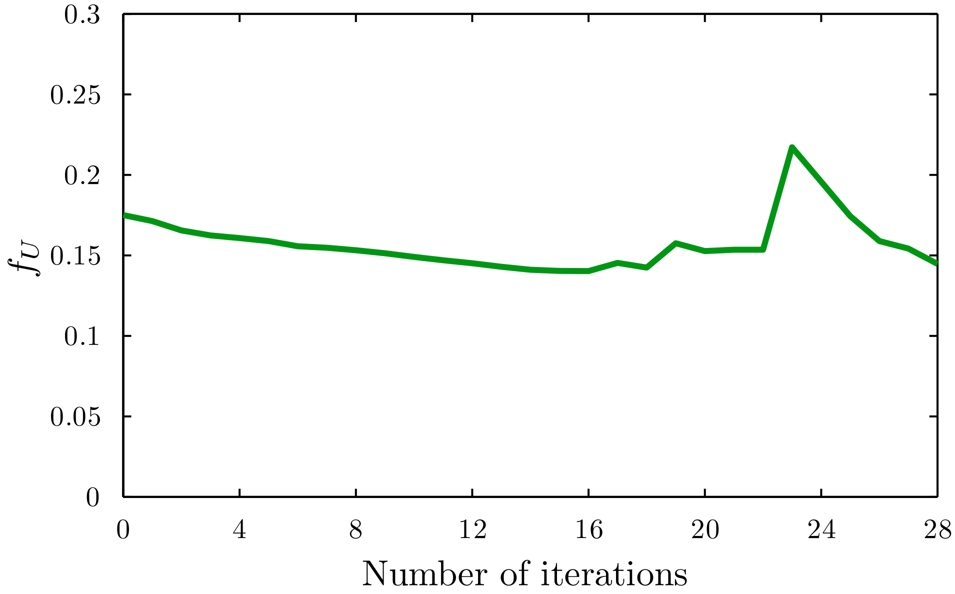
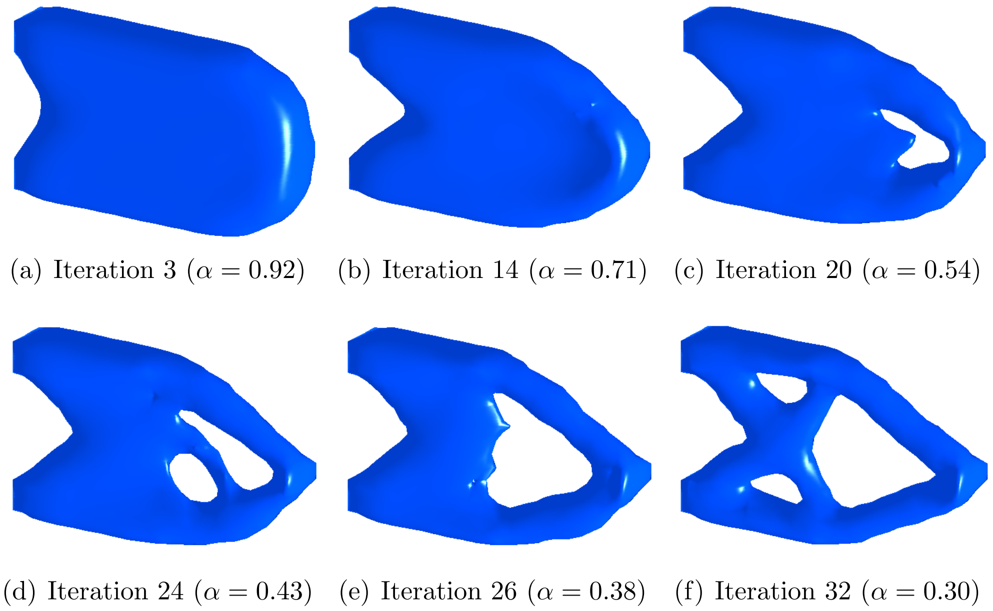


Figure 10.30: Evolution of f_U for Example-3, case 1

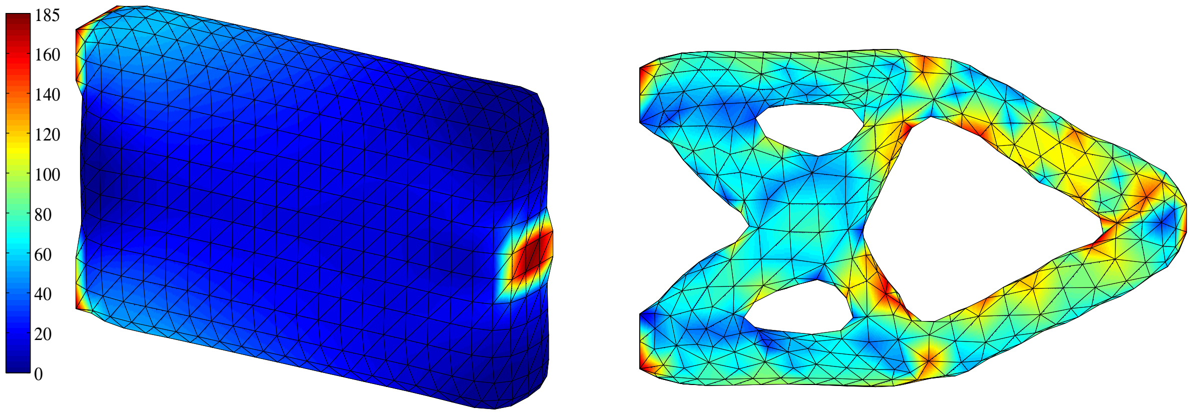

(a) Iteration 3 ($\alpha = 0.92$) (b) Iteration 14 ($\alpha = 0.71$) (c) Iteration 20 ($\alpha = 0.54$)

(d) Iteration 24 ($\alpha = 0.43$) (e) Iteration 26 ($\alpha = 0.38$) (f) Iteration 32 ($\alpha = 0.30$)

Figure 10.31: Evolution of structural geometry for Example-3, case 2

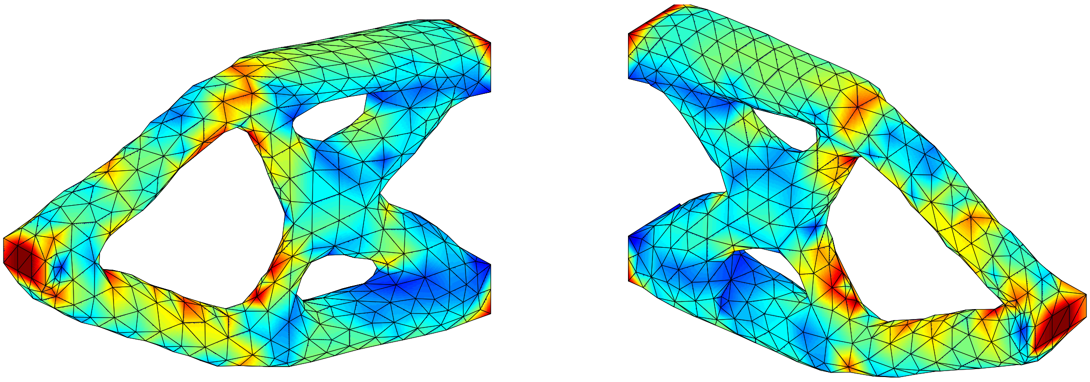
Figure 10.32: Evolution of f_U for Example-3, case 2

optimal design than in the initial design. This indicates that the optimisation method efficiently redistributes material within the design domain and results in an optimum which is approaching towards a fully stressed design.



(a) Initial design

(b) Optimal design view 1



(c) Optimal design view 2

(d) Optimal design view 3

Figure 10.33: von Mises stress contours of initial and optimal geometry for Example-3, case 2

10.3.4 Example-4

The final example solved in this study is a cube with dimensions, $L = 26$, $W = 26$ and $H = 26$, as shown in Figure 10.34. Based on the boundary conditions, three different cases are considered for this example. In the first case, a load $P = 1.6$KN is applied at the centre of the top face, and the bottom face is constrained in all directions at the four corners. In the second case, three of the fixed constraints are replaced by roller supports. The third case is similar to the first one; however, the load is applied at the whole area of the top face instead at the centre. In all cases, the level set design domain is discretised into $13 \times 13 \times 13$ cubic cells with edge length $d = 2$.

In case 1, the optimisation problem is solved using the maximum von Mises stress in

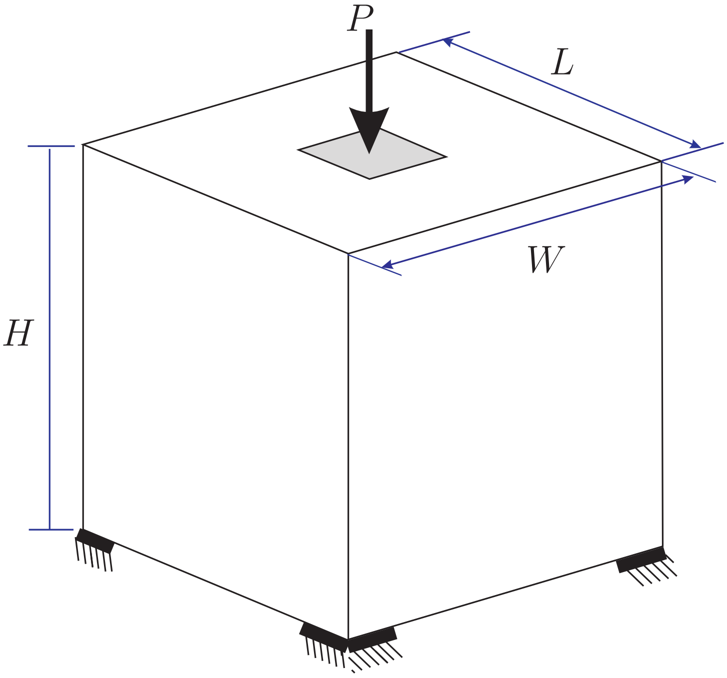


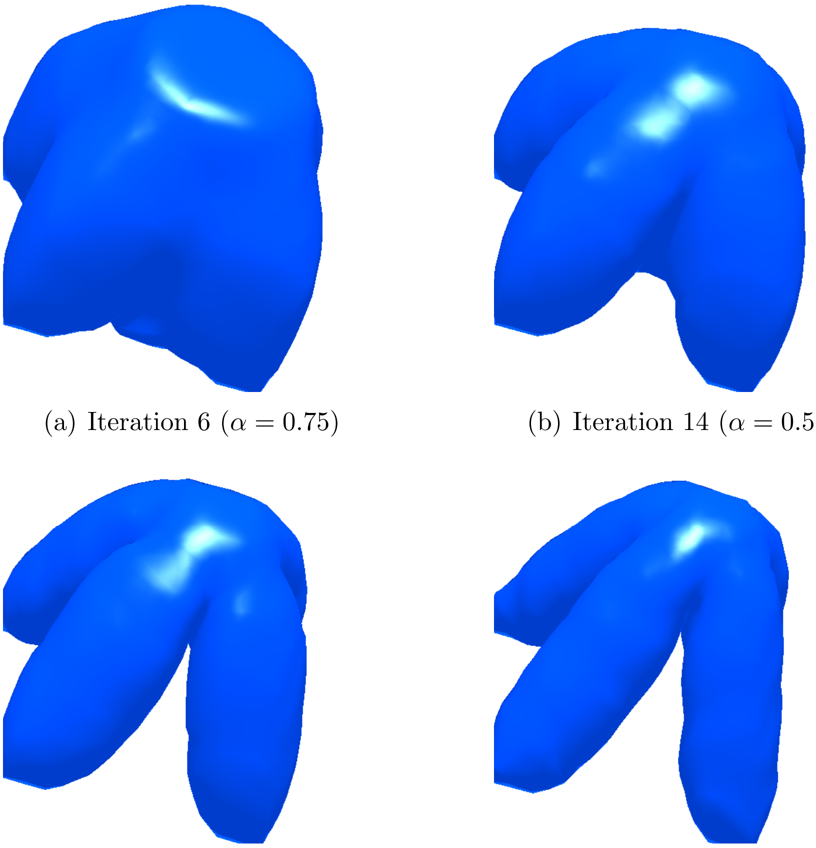
Figure 10.34: Design domain, loading and boundary conditions for Example-4

the initial design, i.e. $\sigma_{Vmax} = 55$ MPa and with a target volume fraction of $0.30V_0$. The evolution of the structural geometry during the optimisation iterations is depicted in Figure 10.35. The optimum obtained closely resembles that presented in [20], and hence further validates the proposed optimisation method with different design domain and boundary conditions.

The von Mises stress contours for the initial and optimal designs are depicted in Figure 10.36. It is evident from the comparison of the contour plots, that the optimisation algorithm efficiently removes low stressed material, and hence the optimum obtained has more uniform stress contours than the initial design.

Figure 10.37 displays the evolution of the specific strain energy during the optimisation process. It can be seen that in the initial iterations the specific strain energy decreases rapidly due to high material removal, and finally settles down near the end of the optimisation process.

As stated above, in the case 2, three fixed constraints are replaced with roller supports whereas the initial design and loading are kept the same as used case 1. The optimisation



(a) Iteration 6 ($\alpha = 0.75$)

(b) Iteration 14 ($\alpha = 0.55$)

(c) Iteration 32 ($\alpha = 0.41$)

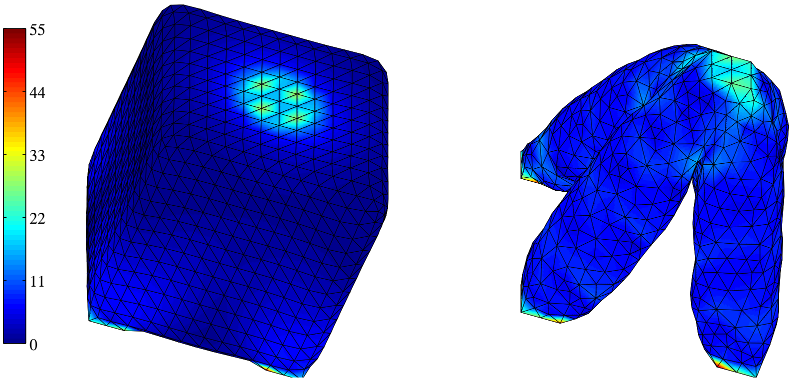
(d) Iteration 58 ($\alpha = 0.30$)

Figure 10.35: Evolution of structural geometry for Example-4, case 1

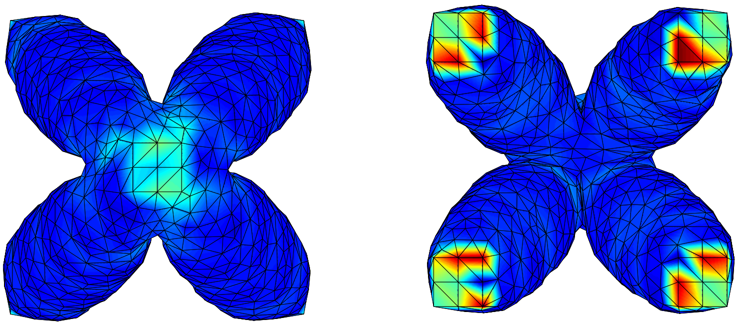
problem is solved with the maximum von Mises, i.e $\sigma_{Vmax} = 53$ MPa and with a target volume fraction of $0.30V_0$. The evolution of the structural geometry at different stages of the optimisation process is displayed in Figure 10.38. In comparison with the previous case of this example, as expected the use of roller supports resulted in interconnecting bars between the four supporting members of the structure. The results obtained are in close agreement with those presented in [15, 20, 77].

Figure 10.39 presents the von Mises stress contour plots of the initial and optimal designs. It can be seen that due to the cross-section variation within the structural members, the interconnecting bars are more highly stressed than the main supporting bars. A similar trend of the specific strain energy (as shown in Figure 10.40) has also been observed in the second case of this example.

In the final case of this example, the boundary conditions are those used in the first



(a) Initial design (b) Optimal design iso view



(c) Optimal design top view (d) Optimal design bottom view

Figure 10.36: von Mises stress contours of initial and optimal geometry for Example-4, case 1

case. However, the load is now applied at the whole face. Figure 10.41 shows several intermediate results obtained during the optimisation process. In the previous two cases, the optimum designs obtained have inclined supporting members. However,in the present case, with different loading conditions the optimisation method resulted in a final design with straight supporting members as shown in Figure 10.41(d). This final case demonstrates that the proposed optimisation method can be efficiently used for the solution of a range of optimisation problems. Figure 10.42 shows the evolution of f_U at each iteration.

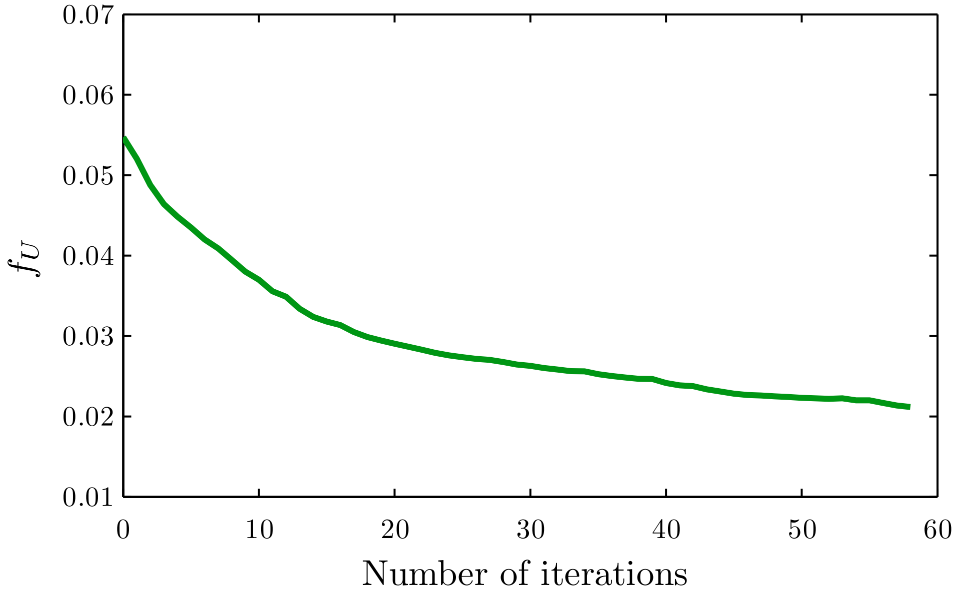
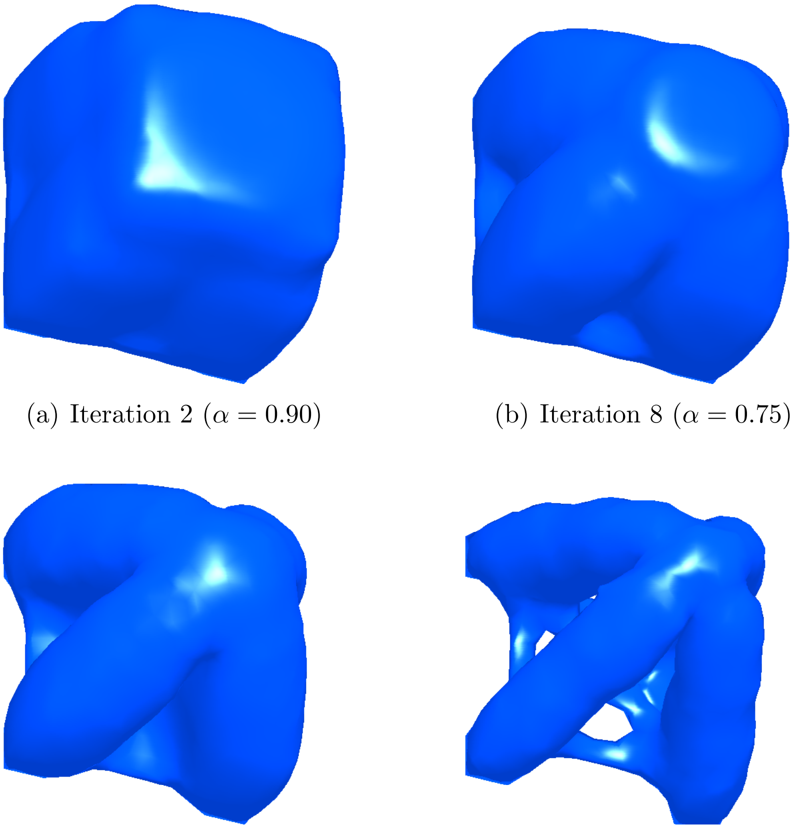


Figure 10.37: Evolution of f_U for Example-4, case 1



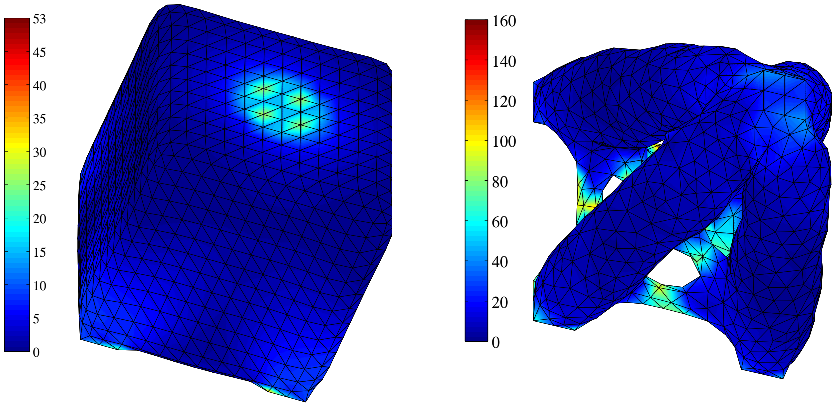
(a) Iteration 2 ($\alpha = 0.90$)

(b) Iteration 8 ($\alpha = 0.75$)

(c) Iteration 19 ($\alpha = 0.57$)

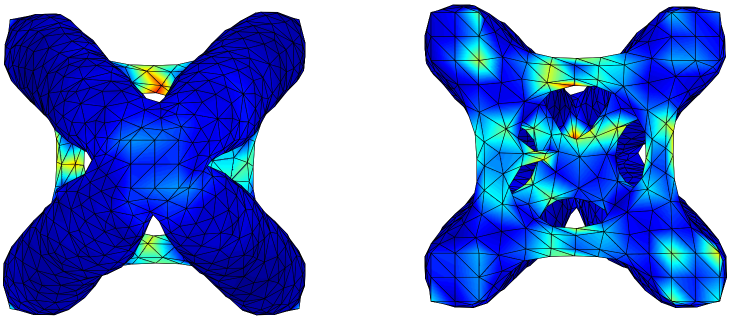
(d) Iteration 34 ($\alpha = 0.30$)

Figure 10.38: Evolution of structural geometry for Example-4, case 2



(a) Initial design

(b) Optimal design iso view



(c) Optimal design top view

(d) Optimal design bottom view

Figure 10.39: von Mises stress contours of initial and optimal geometry for Example-4, case 2

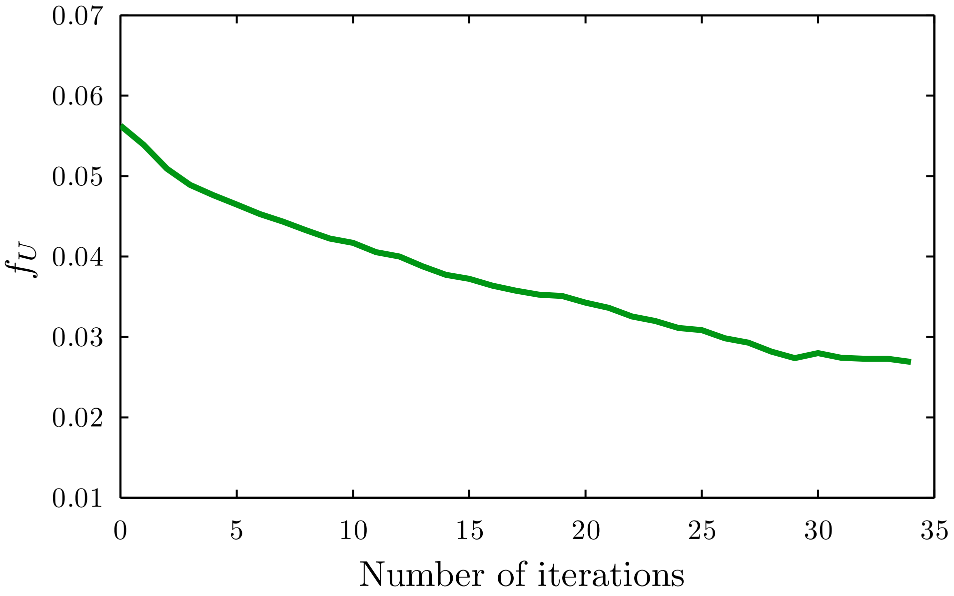
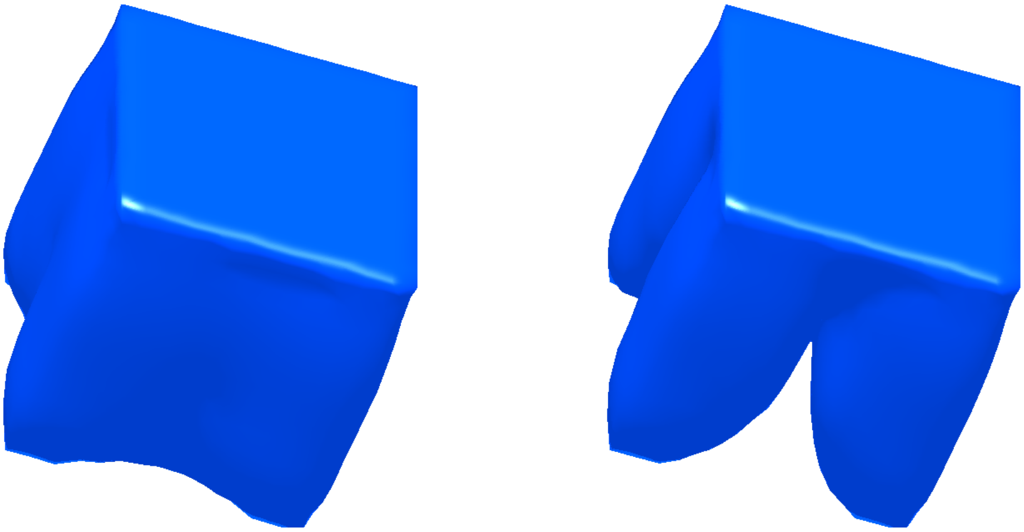
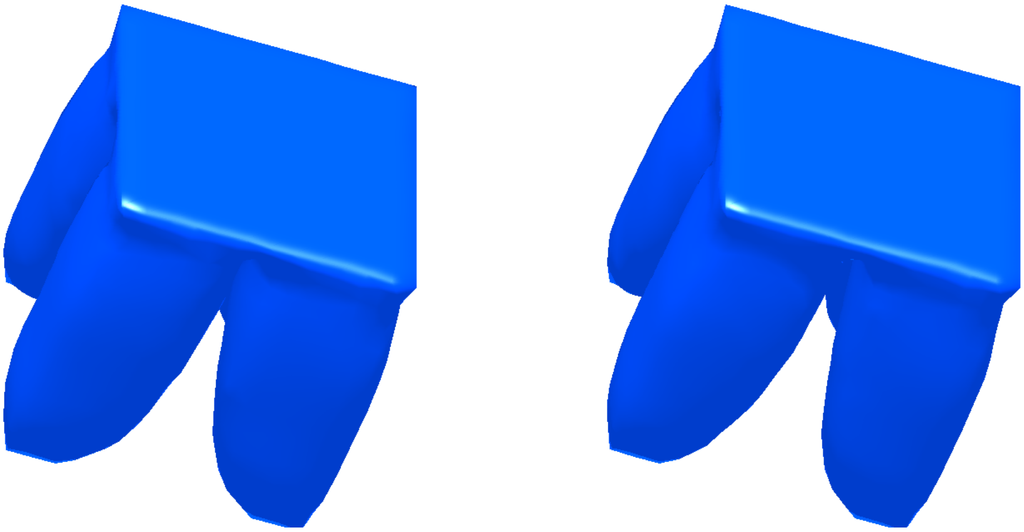


Figure 10.40: Evolution of f_U for Example-4, case 2



(a) Iteration 3 ($\alpha = 0.90$)

(b) Iteration 20 ($\alpha = 0.80$)



(c) Iteration 35 ($\alpha = 0.70$)

(d) Iteration 55 ($\alpha = 0.60$)

Figure 10.41: Evolution of structural geometry for Example-4, case 3

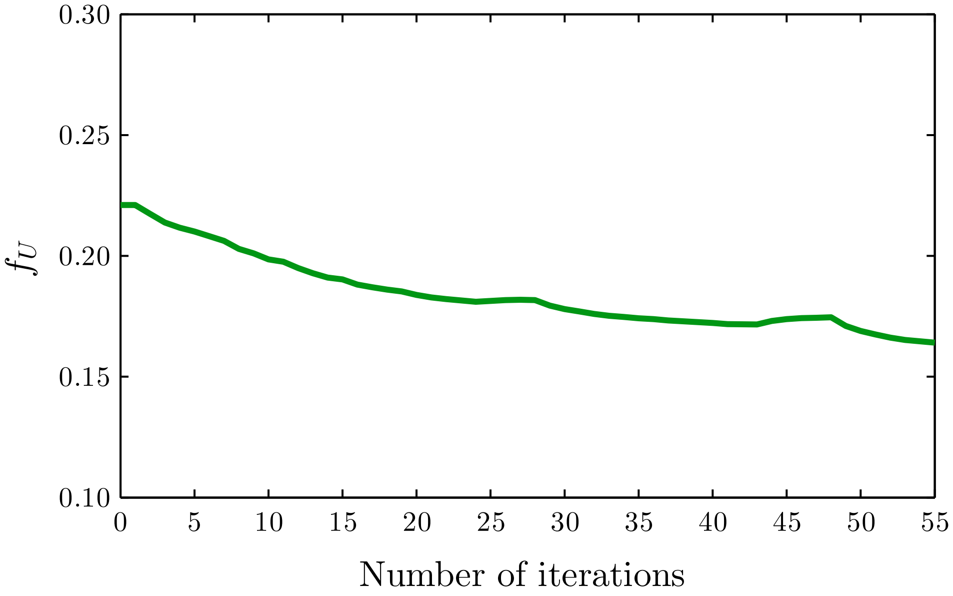


Figure 10.42: Evolution of f_U for Example-4, case 3

10.4 Conclusions

A three-dimensional LSM based structural topology optimisation method has been successfully implemented in this Chapter. The proposed optimisation method extends the two-dimensional optimisation approach to three-dimensions. During the optimisation process, the structural geometry evolves into an optimal design through the progressive removal of inefficient material from the low stressed regions and addition to the high stressed regions of the structure. This evolutionary approach is integrated with the boundary element and level set methods. The BEM is used to analyze the modified structural geometry at each iteration. The optimisation method then identifies the potential regions within the structure to add and remove material. The LSM is then used to modify those regions of the structure identified by the optimisation method. As demonstrated through the examples presented, in 3D LSM, holes appear automatically through the intersection of two surfaces moving towards each other. Therefore, in 3D, the use of LSM eliminates the use of an additional hole insertion mechanism as both shape and topology optimisation can be performed at the same time. During the optimisation iterations, the MC algorithm extracts the new zero level set contours in the form of a triangular mesh. As the BEM is based on a boundary discretisation approach; therefore, the extracted geometry can be directly used to analyse the modified geometry. However, there may exist some poor quality elements in the extracted mesh, which can reduce the accuracy of the BE analysis. As demonstrated, the mesh postprocessing and improvement methods significantly enhance the quality of the individual elements as well as the overall mesh. Therefore, in the present implementation, integration of LSM and BEM eliminates the use of additional mesh generation tools. Furthermore, the LSM handles both shape and topology optimisation and there is no need to use internal points within the design domain. Hence, this greatly enhances the computational capabilities of the proposed optimisation method. In order to validate the proposed optimisation method, four different benchmark examples are considered in this study. Each example is solved with different stress criteria, and similar optimal designs are obtained for each case. Furthermore, the optimal designs obtained for each example closely resemble the optima published within the field of structural optimisation.

Chapter 11

Conclusions

11.1 Conclusions

The research work presented in this thesis is related to the development of structural optimisations algorithms based on the boundary element and level set methods for two and three-dimensional linear elastic problems. In the initial implementation, a stress based ESO approach has been used to add and removal material for the solution of two-dimensional optimisation problems. At each iteration, NURBS are fitted to a set of points lying on the zero level set contour, and these are automatically meshed with boundary elements. The von Mises stress results from the boundary element analysis are mapped to a distribution of the level set velocity function, which is then used to update the design geometry in preparation for the next iteration. During the numerical implementation, the level set function is updated with a HJ type level set equation. However, an update of the level set function with the HJ equation does not nucleate holes in two-dimensional optimisation probelms. In order to evaluate the proposed method initial guessed designs with pre-existing holes have been used for the solution of the optimisation problems. The optimal design topologies and geometries obtained from the proposed method closely resemble the optima published in the literature of the structural optimisation. The unique combination of BEM, evolutionary approach, LSM and NURBS provides an optimisation technique with fast and accurate structural analysis and with the added advantage of a smooth geometry both from the structural analysis as well as the manufacturing point of view.

The research work presented to date using BEM and LSM based optimisation methods are dependent on initially guessed designs with pre-existing holes. Therefore, the initially proposed method has been further improved with the integration of a hole insertion mechanism based on the von Mises stress criterion, providing an LSM and BEM based optimisation technique which does not rely on an initial guess topology with pre-existing holes. Further, the use of a von Mises stress based hole insertion mechanism allows both shape and topological changes with a single criterion. However, the hole insertion is based on the internal points distribution and in the initial implementation there is some randomness in the algorithm that distributes these internal points in the design domain. Therefore, some of the results appeared to be slightly asymmetric at intermediate iterations when the problem was symmetric, and the final solutions seem to recovery symmetry. However, the successful integration of BEM and LSM provides an optimisation approach which is no more dependent on an initial guess topology with pre-existing holes.

In the above discussion, a von Mises stress based hole insertion criterion has been successfully implemented within a BEM and LSM based framework. However, most of the level set based optimisation methods use the topological derivative as a criterion for hole insertion. Therefore, a detailed study has also been carried out to investigate the relationship between the von Mises stress and topological derivative based hole insertion criteria in a BEM and LSM based structural optimisation approach. During the numerical implementation, it has been found that there exists an approximately linear relationship between the square of the von Mises stress and topological derivative. The interesting correlation found between the two hole insertion criteria has been tested for four different benchmark examples. The results presented for these examples show (i) a close resemblance to optima published in the literature for those cases (ii) the robustness of the proposed optimisation method, and (iii) validation of the correlation between the two hole insertion criteria.

The BEM and LSM are further combined with a shape sensitivity approach for the solution of two-dimensional minimum compliance problems. The proposed sensitivity based method is capable of automatically inserting holes during the optimisation process using a topological derivative approach. It has been observed from the results obtained with the proposed method, that for a given problem different initial designs can be used and the optimisation method provides quite similar optimal solutions. In the current implementation internal points can be eliminated, and the computational efficiency can be enhanced if an initial guess design with pre-existing holes is used. The results obtained show that the topological derivative approach for hole insertion (previously studied with an ESO approach) also fits well within the current implementation. Therefore, during the optimisation process, both shape and topology optimisation can be performed at the same time. In the sensitivity based approach the normal velocity is a function of the strain energy density and the Lagrange multiplier. The strain energy at structural boundary can be directly calculated from the BE analysis. In the present implementation, a bisectioning algorithm has been implemented to maintain a constant volume during the optimisation process. The use of the bisectioning algorithm allows calculation of the exact value of the Lagrange multiplier in accordance with the volume constraint. An exact Lagrange multiplier can be used throughout the optimisation process. However, in situations, where the current volume fraction is far away from the target volume a fixed Lagrange multiplier can be used around the target volume fraction and the exact value can be used later on to perform the optimisation process at constant volume. However, the fixed value used during the numerical implementation is based on numerical experience and additional efforts will always be required to select an optimum value of the Lagrange multiplier. As a whole, the sensitivity based approach provides optimal solutions with fast convergence rate and in addition the optimal topology closely resemble those available in the literature.

The initially proposed ESO based optimisation approach has been further improved with the implementation of an additional mechanism which can exactly satisfy the volume constraint. During the optimisation process once the target volume fraction has been achieved, the proposed method effectively preserves volume in the remaining iterations with the use of a bisectioning like algorithm. This new implementation allows material addition and removal at constant volume, mainly through boundary movements. Therefore, material redistribution takes place within the design domain and as a result the compliance of the structure either slightly decreases or remains stable. The implementation of a constant volume mechanism also allows a direct comparison of the results obtained with the evolutionary and sensitivity based approaches. It has been observed that the optimal designs obtained with the two different methods, and the corresponding evolutions of the objective functions are similar to each other. This suggests that in the absence of an objective function based minimisation mechanism the constant volume based evolutionary approach is still capable of minimising the compliance of the structure.

Finally, the two-dimensional evolutionary based optimisation method has also been extended for the solution of three-dimensional optimisation problems. There are two advantages associated with the use of LSM in three-dimensional topology optimisation. Firstly, the LSM may readily be applied to three-dimensional space. Secondly, holes appear automatically through the intersection of two surfaces moving towards each other. Therefore, the use of LSM eliminates the need of an additional hole insertion mechanism as both shape and topology optimisation can be performed at the same time. During the optimisation process, the structural geometry evolves into an optimal design through the progressive removal of inefficient material from the low stressed regions and addition to the high stressed regions of the structure. As demonstrated through the examples presented, in three-dimensions holes appear automatically with the use of LSM.

During the optimisation process, the MC algorithm extracts the new zero level set contours in the form of a triangular mesh. As the BEM is based on a boundary discretisation approach; therefore, the extracted geometry can be directly used to analyse the modified geometry. However, there may exist some poor quality elements in the extracted mesh, which can reduce the accuracy of the BE analysis. As demonstrated, the mesh post-processing and improvement methods significantly enhance the quality of the individual elements as well as the overall mesh. Therefore, in the present implementation, integration of LSM and BEM eliminates the use of additional mesh generation tools. Furthermore, the LSM handles both shape and topology optimisation and there is no need to use internal points within the design domain. Hence, this greatly enhances the computational capabilities of the proposed optimisation method. The unique combination of BEM, LSM and evolutionary approach provides a fully integrated three-dimensional structural optimisation approach, which has been successfully verified through the solution of different types of optimisation problems.

11.2 Recommendations for future work

The proposed structural optimisation algorithms have been successfully implemented for the solution of both two and three-dimensional structural optimisation problems. This initial research work was mainly focused on the development of an optimisation approach using boundary element and level set methods. This has been mainly accomplished by the development of ESO based two and three-dimensional, and a sensitivity based two-dimensional optimisation approaches. The objective functions in these approaches are in general based on the compliance minimisation (or stiffness maximisation).

Future work might involve extending these methods to include other objective functions, such as stress minimisation, natural frequency and buckling. The three-dimensional implementation can be further improved with the use of a volume preserving mechanism as outlined for the the two-dimensional optimisation approach. Shape sensitivities can also be incorporated in three-dimensions to compute the velocity function at the structural boundary.

In the present implementation, the NURBS based geometry representation is limited to two-dimensions only. The incorporation of a NURBS based surface fitting technique will provide a final optimal design in a standard CAD representation which can be directly used in further design processes. This will greatly reduce the lead time between the conceptual design and final product. However, the integration of a surface fitting algorithm in an optimisation process requires a number of additional steps, making it computationally expensive. Therefore, future research work should be focused on the selection and implementation of an efficient surface fitting algorithm within a three-dimensional optimisation algorithm.

There are different methods available in the literature of surface fitting or surface reconstruction. The first detailed work has been carried out by Hoppe *et al.* [46] for automatic surface reconstruction from unorganised three-dimensional data points. The surface extracted with the proposed method contained large number of triangles with fixed vertex connectivity. Hoppe *et al.* [47] proposed a mesh optimisation technique to vary the number of vertices and their connectivity. The proposed mesh optimisation technique has been utilised by Hoppe *et al.* [45] for the fitting of a piecewise smooth surface to the unorganised data points. Eck and Hoppe [34] further extended the research work in [45] for automatic reconstruction of B-spline surfaces. Another notable contribution towards the surface reconstruction can be found in the work of Peters [84–86]. This work has been further extended towards a surface reconstruction algorithm for topology optimisation by Koguchi and Kikuchi [59]. Krishnamurthy and Levoy [60] presented a B-spline fitting method for dense polygon meshes of arbitrary topology. Park *et al.* [81–83] proposed an algorithm to produce a surface model from range data, based on the NURBS surface fitting techniques. Dong *et al.* [30, 31] presented an approach for building a quadrangular base complex over a triangular manifold using the Morse-Smale complex [18]. Some level set based surface reconstruction techniques can also be found in references [139, 141].

References

[1] *Computational Mechanics, BEASY user guide, volume 1, 1998.*

[2] K. Abe, T. Fujiu, and K. Koro. A be-based shape optimization method enhanced by topological derivative for sound scattering problems. *Engineering Analysis with Boundary Elements*, 34(12):1082 – 1091, 2010.

[3] K. Abe, S. Kazama, and K. Koro. A boundary element approach for topology optimization problem using the level set method. *Communications in Numerical Methods in Engineering*, 23(5):405 – 416, 2007.

[4] D. Adalsteinsson and J. A. Sethian. A fast level set method for propagating interfaces. *Journal of Computational Physics*, 118(2):269–277, 1995.

[5] D. Adalsteinsson and J. A. Sethian. The fast construction of extension velocities in level set methods. *Journal of Computational Physics*, 148(1):2–22, 1999.

[6] G. Allaire and F. Jouve. A level-set method for vibration and multiple loads structural optimization. *Computer Methods in Applied Mechanics and Engineering*, 194(3033):3269 – 3290, 2005.

[7] G. Allaire and F. Jouve. Minimum stress optimal design with the level set method. *Engineering Analysis with Boundary Elements*, 32(11):909–918, 2008.

[8] G. Allaire, F. Jouve, and A. M. Toader. Structural optimization using sensitivity analysis and a level-set method. *Journal of Computational Physics*, 194(1):363–393, 2004.

[9] S. Amstutz and H. Andreä. A new algorithm for topology optimization using a level-set method. *Journal of Computational Physics*, 216(2):573–588, 2006.

[10] R. Bade, J. Haase, and B. Preim. Comparison of fundamental mesh smoothing algorithms for medical surface models. In *Simulations and Visualisierung*, volume 6, pages 289–304, 2006.

[11] A. A Becker. *The Boundary Element Methods in Engineering: A complete course.* McGRAW - HILL BOOK COMPANY, 1992.

[12] T. Belytschko, S. P. Xiao, and C. Parimi. Topology optimization with implicit functions and regularization. *International Journal for Numerical Methods in Engineering*, 57(8):1177–1196, 2003.

[13] M. P. Bendsøe. *Optimization of structural topology, shape, and material.* Springer. Berlin, 1995.

[14] M. P. Bendsøe and N. Kikuchi. Generating optimal topologies in structural design using a homogenization method. *Computatr Methods in Applied Mechanics and Engineering*, 71(2):197–224, 1988.

[15] C. Bertsch, A. P. Cisilino, and N. Calvo. Topology optimization of three-dimensional load-bearing structures using boundary elements. *Advances in Engineering Software*, 41(5):694–704, 2010.

[16] J. Bloomenthal. Polygonization of implicit surfaces. *Computer Aided Geometric Design*, 5(4):341 – 355, 1988.

[17] P. Bourke, 1994. Polygonising a scalar field. [Online].http://paulbourke.net/geometry/polygonise/.

[18] P. T. Bremer, B. Hamann, H. Edelsbrunner, and V. Pascucci. A topological hierarchy for functions on triangulated surfaces. *IEEE Transactions on Visualization and Computer Graphics*, 10(4):385–396, 2004.

[19] L. Carretero Neches and A. P. Cisilino. Topology optimization of 2D elastic structures using boundary elements. *Engineering Analysis with Boundary Elements*, 32(7):533–544, 2008.

[20] J. Céa, S. Garreau, P. Guillaume, and M. Masmoudi. The shape and topological optimizations connection. *Computer Methods in Applied Mechanics and Engineering*, 188(4):713–726, 2000.

[21] E. Cervera and J. Trevelyan. Evolutionary structural optimisation based on boundary representation of NURBS. Part I: 2D algorithms. *Computers and Structures*, 83:1902–1916, 2005.

[22] E. Cervera and J. Trevelyan. Evolutionary structural optimisation based on boundary representation of NURBS. Part II: 3D algorithms. *Computers and Structures*, 83(23):1917–1929, 2005.

[23] J. H. Choi and B. M. Kwak. Boundary integral equation method for shape optimization of elastic structures. *International Journal for Numerical Methods in Engineering*, 26(7):1579–1595, 1988.

[24] D. N. Chu, Y. M. Xie, A. Hira, and G. P. Steven. Evolutionary structural optimization for problems with stiffness constraints. *Finite Elements in Analysis and Design*, 21(4):239 – 251, 1996.

[25] D. N. Chu, Y. M. Xie, A. Hira, and G. P. Steven. On various aspects of evolutionary structural optimization for problems with stiffness constraints. *Finite Elements in Analysis and Design*, 24(4):197–212, 1997.

[26] A. P. Cisilino. Topology optimization of 2D potential problems using boundary elements. *Computer Modeling in Engineering and Sciences*, 15(2):99–106, 2006.

[27] T. A Cruse. Numerical solutions in three dimensional elastostatics. *International Journal of Solids and Structures*, 5(12):1259–1274, 1969.

[28] J. Deaton and R. Grandhi. A survey of structural and multidisciplinary continuum topology optimization: post 2000. *Structural and Multidisciplinary Optimization*, pages 1–38, 2013.

[29] N. P. Dijk, K. Maute, M. Langelaar, and F. Keulen. Level-set methods for structural topology optimization: a review. *Structural and Multidisciplinary Optimization*, 48(3):437–472, 2013.

[30] S. Dong, P. T. Bremer, V. Garland, M. Pascucci, and J. C. Hart. Spectral surface quadrangulation. In *3SIGGRAPH, ACM Transactions on Graphics*, pages 1057–1066, 2006.

[31] S. Dong, S. Kircher, and M. Garland. Harmonic functions for quadrilateral remeshing of arbitrary manifolds. *Computer-Aided Geometric Design*, 22:392–423, 2005.

[32] P. D. Dunning and H. A. Kim. A new hole insertion method for level set based structural topology optimization. *International Journal for Numerical Methods in Engineering*, 93(1):118–134, 2013.

[33] P. D. Dunning, H. A. Kim, and G. Mullineux. Introducing loading uncertainty in topology optimization. *AIAA Journal*, 49:760–768, 2011.

[34] M. Eck and H. Hoppe. Automatic reconstruction of b-spline surfaces of arbitrary topological type. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, SIGGRAPH '96, pages 325–334, 1996.

[35] H. A. Eschenauer, V. V. Kobelev, and A. Schumacher. Bubble method for topology and shape optimization of structures. *Structural and Multidisciplinary Optimization*, 8(1):42–51, 1994.

[36] G.E. Farin. *Curves and surfaces for computer aided geometric design: A practical guide.* Boston; London: Academic Press, 1988.

[37] T. M. Foster, M. S. Mohamed, J. Trevelyan, and G. Coates. Rapid re-meshing and re-solution of three-dimensional boundary element problems for interactive stress analysis. *Engineering Analysis with Boundary Elements*, 36(9):1331–1343, 2012.

[38] T. Fries and T. Belytschko. The extended/generalized finite element method: An overview of the method and its applications. *International Journal for Numerical Methods in Engineering*, 84(3):253–304, 2010.

[39] M. J. Garcia and G. P. Steven. Fixed grid finite elements in elasticity problems. *Engineering Computations*, 16(2):145–164, 1999.

[40] D. E. Goldberg and J. H. Holland. Genetic algorithms and machine learning. *Machine learning*, 3(2):95–99, 1988.

[41] X. Guo, W. S. Zhang, M. Y. Wang, and P. Wei. Stress-related topology optimization via level set approach. *Computer Methods in Applied Mechanics and Engineering*, 200(4748):3439 – 3452, 2011.

[42] S. Ha and S. Cho. Topological shape optimization of heat conduction problems using level set approach. *Numerical Heat Transfer, Part B: Fundamentals*, 48(1):67–88, 2005.

[43] S. Ha and S Cho. Level set based topological shape optimization of geometrically nonlinear structures using unstructured mesh. *Computers and Structures*, 86(13–14):1447–1455, 2008.

[44] R. T. Haftka and R. V. Grandhi. Structural shape optimization - A survey. *Computer Methods In Applied Mechanics And Engineering*, 57:91–106, 1986.

[45] H. Hoppe, T. DeRose, T. Duchamp, M. Halstead, H. Jin, J. McDonald, J. Schweitzer, and W. Stuetzle. Piecewise smooth surface reconstruction. In *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, SIGGRAPH '94, pages 295–302, 1994.

[46] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. *SIGGRAPH Computer Graphics*, 26:71–78, July 1992.

[47] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Mesh optimization. In *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '93, pages 19–26, 1993.

[48] X. Huang and Y. M. Xie. Convergent and mesh-independent solutions for the bi-directional evolutionary structural optimization method. *Finite Elements in Analysis and Design*, 43(14):1039–1049, 2007.

[49] X. Huang and Y. M. Xie. *Evolutionary Topology Optimization of Continuum Structures: Methods and Applications.* New York: John Wiley & Sons, 2010.

[50] S. H. Jacobson, K. A. Sullivan, and A. W. Johnson. Discrete manufacturing process design optimization using computer simulation and generalized hill climbing algorithms. *Engineering Optimization*, 31(2):247–260, 1998.

[51] G. W. Jang and Y. Y. Kim. Sensitivity analysis for fixed-grid shape optimization by using oblique boundary curve approximation. *International Journal of Solids and Structures*, 42(11–12):3591–3609, 2005.

[52] M. A. Jaswon. Integral equation methods in potential theory. I. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences,*, 275:23–32, 1963.

[53] H. Jia, HG Beom, Y. Wang, S. Lin, and B. Liu. Evolutionary level set method for structural topology optimization. *Computers and Structures*, 89(5-6):445–454, 2011.

[54] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon. Optimization by simulated annealing: An experimental evaluation; part i, graph partitioning. *Operations research*, 37(6):865–892, 1989.

[55] J.H. Kane. *Boundary Element Analysis in Engineering Continuum Mechanics*. Prentice Hall, 1994.

[56] H. A. Kim, M. J. Garcia, O. M. Querin, G. P. Steven, and Y. M. Xie. Introduction of fixed grid in evolutionary structural optimisation. *Engineering Computations*, 17(4):427–439, 2000.

[57] H. A. Kim, O. M. Querin, G. P. Stevem, and Y. M. Xie. A method for varying the number of cavities in an optimised topology using evolutionary structural optimisation. *Structural and Multidisciplinary Optimization*, 19(2):140–147, 2000.

[58] H. A. Kim, O. M. Querin, G. P. Stevem, and Y. M. Xie. Improving efficiency of evolutionary structural optimization by implementing fixed grid mesh. *Structural and Multidisciplinary Optimization*, 24(6):441–448, 2003.

[59] A. Koguchi and N. Kikuchi. A surface reconstruction algorithm for topology optimization. *Engineering with Computers*, 22(1):1–10, 2006.

[60] V. Krishnamurthy and M. Levoy. Fitting smooth surfaces to dense polygon meshes. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pages 313–324, New York, NY, USA, 1996.

[61] J. Kwak and S. Cho. Topological shape optimization of geometrically nonlinear structures using level set method. *Computers and Structures*, 83(27):2257 – 2268, 2005.

[62] J. C. Lachat and J. O Watson. Effective numerical treatment of boundary integral equations. *International Journal for Numerical Methods in Engineering*, 10:991–1005, 1976.

[63] J. Li and P. Agathoklis. An efficiency enhanced isosurface generation algorithm for volume visualization. *The Visual Computer*, 13(9-10):391–400, 1998.

[64] Q. Li, G. P. Steven, and Y. M. Xie. On equivalence between stress criterion and stiffness criterion in evolutionary structural optimization. *Structural Optimization*, 18(1):67–73, 1999.

[65] Q. Li, Y. M. Xie, and G. P. Steven. Evolutionary topology optimization for temperature reduction of heat conducting fields. *International Journal of Heat and Mass Transfer*, 47(23):5071–5083, 2004.

[66] Z. Liu, J. G. Korvink, and R. Huang. Structure topology optimization: fully coupled level set method via FEMLAB. *Structural and Multidisciplinary Optimization*, 29(6):407–417, 2005.

[67] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *Computer Graphics*, 21(4):163–169, 1987.

[68] J. Luo, Z. Luo, L. Chen, L. Tong, and M. Y. Wang. A semi-implicit level set method for structural shape and topology optimization. *Journal of Computational Physics*, 227(11):5561–5581, May 2008.

[69] D. Manickarajah, Y. M. Xie, and G. P. Steven. An evolutionary method for optimization of plate buckling resistance. *Finite Elements in Analysis and Design*, 29(3–4):205–230, 1998.

[70] R. J. Marczak. Topology optimization and boundary elements: A preliminary implementation for linear heat transfer. *Engineering Analysis with Boundary Elements*, 31(9):793–802, 2007.

[71] R. J. Marczak. Optimization of elastic structures using boundary elements and a topological-shape sensivity formulation. *Latin American Journal of Solids and Structures*, 5:99–117, 2008.

[72] A. G. M. Michell. The limits of economy of material in frame-structures. *Philosophical Magazine Series 6*, 8(47):589–597, 1904.

[73] S. G. Mikhlin. *Integral Equations.* Pergamon Press. New York, 1957.

[74] N. I. Muskhelishvili. *Some Basic Problems of the Mathematical Theory of Elasticity.* Noordhoff. Holland, 1953.

[75] T. S. Newman and H. Yi. A survey of the marching cubes algorithm. *Computers and Graphics*, 30(5):854 – 879, 2006.

[76] A. A. Novotny, R. A. Feijóo, E. Taroco, and C. Padra. Topological sensitivity analysis. *Computer Methods in Applied Mechanics and Engineering*, 192(7):803–829, 2003.

[77] A. A. Novotny, R. A. Feijóo, E. Taroco, and C. Padra. Topological sensitivity analysis for three-dimensional linear elasticity problem. *Computer Methods in Applied Mechanics and Engineering*, 196(41):4354–4364, 2007.

[78] S. Osher and J. A. Sethian. Front propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations. *Journal of Computational Physics*, 79(1):12–49, 1988.

[79] S. J. Osher and F. Santosa. Level set methods for optimization problems involving geometry and constraints: I. frequencies of a two-density inhomogeneous drum. *Journal of Computational Physics*, 171(1):272 – 288, 2001.

[80] G. Papanicolau, A. Bensoussan, and J. L. Lions. *Asymptotic analysis for periodic structures.* North-Holland, Amsterdam-New York-Oxford, 1978.

[81] I. K. Park and S. U. Lee. Geometric modeling from scattered 3-d range data. In *Image Processing, 1997. Proceedings., International Conference on*, volume 2, pages 712–715 vol.2, 1997.

[82] I. K Park, I. D. Yun, and S. U Lee. Constructing nurbs surface model from scattered and unorganized range data. In *Proceedings of Second International Conference on 3-D Digital Imaging and Modeling, 1999*, pages 312–320, 1999.

[83] I. K Park, I. D. Yun, and S. U Lee. Automatic 3-d model synthesis from measured range data. *IEEE Transactions on Circuits and Systems for Video Technology*, 10(2):293–301, 2000.

[84] J. Peters. Constructing c1 surfaces of arbitrary topology using biquadratic and bicubic splines. *Designing fair curves and surfaces*, pages 277–293, 1994.

[85] J. Peters. Biquartic c1-surface splines over irregular meshes. *Computer-Aided Design*, 27(12):895–903, 1995.

[86] J. Peters. C1-surface splines. *SIAM Journal of Numerical Analysis*, 32:645–666, April 1995.

[87] L. Piegl and W. Tiller. *The NURBS book*. Springer-Verlag, 1995.

[88] P. Pons and M. Latapy. Computing communities in large networks using random walks. In *Computer and Information Sciences-ISCIS 2005*, pages 284–293. 2005.

[89] W. Prager. A note on discretized michell structures. *Computer Methods in Applied Mechanics and Engineering*, 3(3):349 – 355, 1974.

[90] W. Prager and G. I. N. Rozvany. Optimization of structural geometry. *Dynamical System Proceedings International Symposium Florida*, pages 265–293, 1977.

[91] B. N. Pshenichny and J. M. Danilin. *Numerical Methods in External Problems.* Moscow: MIR Publishers, 1978.

[92] O. M. Querin, G. P. Steven, and Y. M. Xie. Evolutionary structural optimisation (ESO) using a bidirectional algorithm. *Engineering Computations*, 15(8):1031–1048, 1998.

[93] O. M. Querin, V. Young, G. P. Steven, and Y. M. Xie. Computational efficiency and validation of bi-directional evolutionary structural optimisation. *Computer Methods in Applied Mechanics and Engineering*, 189(2):559 – 573, 2000.

[94] S. Raman and R. Wenger. Quality isosurface mesh generation using an extended marching cubes lookup table. *Computer Graphics Forum*, 27(3):2008, 2008.

[95] F. J. Rizzo. An integral equation approach to boundary value problems of classical elastostatics. *Quaterly of applied mathematics*, 25:83–95, 1967.

[96] B. Rodrigues de Araújo and J. Armando Pires Jorge. Adaptive polygonization of implicit surfaces. *Computers and Graphics*, 29(5):686–696, 2005.

[97] D. F. Rogers. *An Introduction to NURBS: With Historical Perspective.* Morgan Kaufmann, 2001.

[98] J. H. Rong and Q. Q. Liang. A level set method for topology optimization of continuum structures with bounded design domains. *Computer Methods in Applied Mechanics and Engineering*, 197(17-18):1447–1465, 2008.

[99] G. I. N. Rozvany. Grillages of maximum strength and maximum stiffness. *International Journal of Mechanical Sciences*, 14(10):651 – 666, 1972.

[100] G. I. N. Rozvany. Aims, scope, methods, history and unified terminology of computer-aided topology optimization in structural mechanics. *Structural and Multidisciplinary Optimization*, 21(2):90–108, 2001.

[101] G. I. N. Rozvany, M. Zhou, and T. Birker. Generalized shape optimization without homogenization. *Structural and Multidisciplinary Optimization*, 4(3):250–252, 1992.

[102] E. Sanchez-Palencia. *Non-homogeneous media and vibration theory.* Springer-Verlag, Berlin, 1980.

[103] J. A. Sethian. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science.* Cambridge University Press, 2 edition, 1999.

[104] J. A. Sethian and A. Wiegmann. Structural boundary design via level set and immersed interface methods. *Journal of Computational Physics*, 163(2):489–528, 2000.

[105] X. Shi and S. Mukherjee. Shape optimization in three-dimensional linear elasticity by the boundary contour method. *Engineering Analysis with Boundary Elements*, 23(8):627 – 637, 1999.

[106] Y. Shi and R. C. Eberhart. Empirical study of particle swarm optimization. In *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, volume 3, 1999.

[107] O. Sigmund. A 99 line topology optimization code written in Matlab. *Structural and Multidisciplinary Optimization*, 21(2):120–127, 2001.

[108] O. Sigmund and J. Petersson. Numerical instabilities in topology optimization: A survey on procedures dealing with checkerboards, mesh-dependencies and local minima. *Structural optimization*, 16(1):68–75, 1998.

[109] C .A. M. Soares and K. K. Choi. Boundary elements in shape optimal design of structures. In *The Optimum Shape*, General Motors Research Laboratories Symposia Series, pages 199–231. Springer US, 1986.

[110] C. A. M. Soares, H. C. Rodrigues, L. M. O. Faria, and E. J. Haug. Optimization of the geometry of shafts using boundary elements. *Journal of Mechanical Transmission and Automotive Design*, 106:199–202, 1984.

[111] J. Sokolowski and A. Zochowski. On the topological derivative in shape optimization. *SIAM journal on control and optimization*, 37(4):1251–1272, 1999.

[112] G. P. Steven, , L. Qing, K. Proos, and Y. M. Xie. The role of physical sensitivity in evolutionary topology design optimisation with multi-criteria and multi-physics. In *Proceedings of Fifth World Congress on Computational Mechanics (WCCM V). Vienna, Austria.*, 2002.

[113] K. Svanberg. The method of moving asymptotesa new method for structural optimization. *International Journal for Numerical Methods in Engineering*, 24(2):359–373, 1987.

[114] G. T. Symm. Integral equation methods in potential theory. II. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences,*, 275:33–46, 1963.

[115] P. Tanskanen. The evolutionary structural optimization method: theoretical aspects. *Computer Methods in Applied Mechanics and Engineering*, 191:5485–5498, 2002.

[116] B. H. V. Topping, J. Muylle, and P. Ivanyi. *Finite element mesh generation.* Saxe-Coburg publications on computational engineering. Saxe-Coburg Publications, Stirling, Scotland, 2004.

[117] J. Trevelyan. *Boundary Elements for Engineers: Theory and Applications.* WIT Press, 1994.

[118] J. Trevelyan, 2006. "Concept Analyst Ltd". [Online] .www.conceptanalyst.com.

[119] L. Van Miegroet and P. Duysinx. Stress concentration minimization of 2d filets using x-fem and level set description. *Structural and Multidisciplinary Optimization*, 33(4-5):425–438, 2007.

[120] J. Vollmer, R. Mencl, and H. Mueller. Improved laplacian smoothing of noisy surface meshes. In *Computer Graphics Forum*, volume 18, pages 131–138. Wiley Online Library, 1999.

[121] M. Y. Wang and X. Wang. color level sets: a multi-phase method for structural topology optimization with multiple materials. *Computer Methods in Applied Mechanics and Engineering*, 193(68):469 – 496, 2004.

[122] M. Y. Wang, X. Wang, and D. Guo. A level set method for structural topology optimization. *Computer Methods in Applied Mechanics and Engineering*, 192(1-2):227–246, 2003.

[123] M. Y. Wang, X. Wang, Y. Mei, and S. Chen. Design of multimaterial compliant mechanisms using level-set methods. *Journal of Mechanical Design*, 127(5):941–956, 2005.

[124] S. Wang and M. Y. Wang. A moving superimposed finite element method for structural topology optimization. *International Journal for Numerical Methods in Engineering*, 65(11):1892–1922, 2006.

[125] S. Wang and M. Y. Wang. Radial basis functions and level set method for structural topology optimization. *International Journal for Numerical Methods in Engineering*, 65(12):2060–2090, 2006.

[126] S. Y. Wang, K. M. Lim, B. C. Khoo, and M. Y. Wang. An extended level set method for shape and topology optimization. *Journal of Computational Physics*, 221(1):395–421, 2007.

[127] P. Wei and M. Y. Wang. Piecewise constant level set method for structural topology optimization. *International Journal for Numerical Methods in Engineering*, 78(4):379–402, 2009.

[128] P. Wei, M. Y. Wang, and X. Xing. A study on X-FEM in continuum structural optimization using a level set model. *Comput-Aided Design*, 42(8):708–719, 2010.

[129] Q. Xia, T. Shi, S. Liu, and M. Y. Wang. A level set solution to the stress-based structural shape and topology optimization. *Computers and Structures*, 90–91:55–64, 2012.

[130] Q. Xia and M. Y. Wang. Topology optimization of thermoelastic structures using level set method. *Computational Mechanics*, 42(6):837–857, 2008.

[131] Y. M. Xie and G. P. Steven. A simple evolutionary procedure for structural optimization. *Computers and Structures*, 49:885–96, 1993.

[132] Y. M. Xie and G. P. Steven. Evolutionary structural optimization for dynamic problems. *Computers and Structures*, 58(6):1067 – 1073, 1996.

[133] T. Yamada, K. Izui, S. Nishiwaki, and A. Takezawa. A topology optimization method based on the level set method incorporating a fictitious interface energy. *Computer Methods in Applied Mechanics and Engineering*, 199(45-48):2876–2891, 2010.

[134] S. Yamasaki, S. Nishiwaki, T. Yamada, K. Izui, and M. Yoshimura. A structural optimization method based on the level set method using a new geometry-based re-initialization scheme. *International Journal for Numerical Methods in Engineering*, 83(12):1580–1624, 2010.

[135] S. Yamasaki, T. Nomura, A. Kawamoto, K. Sato, and S. Nishiwaki. A level set-based topology optimization method targeting metallic waveguide design problems. *International Journal for Numerical Methods in Engineering*, 87(9):844–868, 2011.

[136] S. Yamasaki, T. Yamada, and T. Matsumoto. An immersed boundary element method for level-set based topology optimization. *International Journal for Numerical Methods in Engineering*, 93(9):960–988, 2013.

[137] K. Yamazaki, J. Sakamoto, and M. Kitano. An efficient shape optimization technique of a two-dimensional body based on the boundary element method. *Comput Struct*, 48(6):1073–1081, 1993.

[138] R. J. Yang and C. H. Chuang. Optimal topology design using linear programming. *Computers and Structures*, 52(2):265 – 275, 1994.

[139] H. Yoshihara, T. Yoshii, T. Shibutani, and T. Maekawa. Topologically robust b-spline surface reconstruction from point clouds using level set methods and iterative

geometric fitting algorithms. *Computer Aided Geometric Design*, 29(7):422 – 434, 2012.

[140] M. Yulin, W. Xiaoming, M. Yulin, and W. Xiaoming. A level set method for structural topology optimization and its applications. *Advances in Engineering Software*, 35(7):415–441, 2004.

[141] H. K. Zhao, S. Osher, and R. Fedkiw. Fast surface reconstruction using the level set method. In *In VLSM 01: Proceedings of the IEEE Workshop on Variational and Level Set Methods*, 2001.