Implementation of Bi-directional Evolutionary Structural Optimization (BESO) in the underground excavations and slope stability

A thesis submitted in fulfilment of the requirements for the degree of Doctor of Philosophy

Mohammad Yazdanpanah
B.Sc.

School of Civil Environmental and Chemical Engineering
College of Science Engineering and Health
RMIT University

August 2014
Declaration

I certify that except where due acknowledgement has been made, the work is that of the author alone; the work has not been submitted previously, in whole or in part, to qualify for any other academic award; the content of the thesis/project is the result of work which has been carried out since the official commencement date of the approved research program; any editorial work, paid or unpaid, carried out by a third party is acknowledged; and, ethics procedures and guidelines have been followed.

Mohammad Yazdanpana

August 2014
Acknowledgements

I would like to express my deep gratitude to my parents for their support, encouragement and patience during my life. Without their support, I could not definitely have experienced the blessing of study and pursuing my interests.

I sincerely wish to thank my senior supervisor, Dr. Gang Ren, for his invaluable guidance, support and advice throughout the duration of this research. Also, my warm thanks are due to Prof. Mike Xie for his assistance and support at several stages of this research.

I am grateful to Dr. Zhihao Zuo for his beneficial discussion. In this regard, I also warmly appreciate the support of other members of Centre for Innovative Structures and Materials (CISM).

Finally, I want to gratefully acknowledge the kind advice and support of Mr. Farshid Jalilvand. Without his support, I might not have studied in Australia.
Contents

Abstract.............................................................................................................................................. 11

Introduction ........................................................................................................................................ 13

1.1 Human's needs and optimization .............................................................................................. 13
1.2 Topology optimization in engineering problems ........................................................................ 13
1.3 Topology optimization in geotechnical engineering problems ................................................ 14
1.4 Evolutionary Structural Optimization (ESO) .......................................................................... 16
1.5 Layout of the thesis ..................................................................................................................... 16

Finite element analysis and programming ....................................................................................... 18

2.1 Discretization of the domain ....................................................................................................... 18

2.1.1 Introduction ............................................................................................................................ 18
2.1.2 One dimensional bar element ................................................................................................ 19
2.1.3 Alternative approach to deduce element equilibrium ......................................................... 22
2.1.4 Two-dimensional elements ................................................................................................... 25
2.1.5 Three-dimensional strain and stress ..................................................................................... 28

2.2 General finite element formulation ............................................................................................ 30

2.2.1 Introduction ............................................................................................................................ 30
2.2.2 Local coordinate for quadrilateral elements ......................................................................... 31
2.2.3 Integration for quadrilateral elements .................................................................................. 34
2.2.4 Local coordinate for triangular elements .............................................................................. 36
2.2.5 Higher order 2D elements .................................................................................................... 39
2.2.6 Three-dimensional elements .................................................................................................. 40
2.2.7 Tetrahedral elements ............................................................................................................. 42
2.2.8 Multi-element assemblies ...................................................................................................... 43
2.2.9 Degree of freedom (DOF) of elements .................................................................................. 43
2.2.10 Global degree of freedom .................................................................................................... 43
2.2.11 Modulation of boundary conditions .................................................................................... 45
2.2.12 Multi element assemblies .................................................................................................... 48

2.3 Concluding remarks ..................................................................................................................... 50

Topology optimization methods ......................................................................................................... 52
3.1 Background

3.1.1 Topology optimization

3.1.2 Minimum compliance layout

3.1.3 Sensitivity analysis

3.1.4 Continuous techniques to solve topology optimization problem

3.1.5 Binary techniques to solve topology optimization problems

3.1.6 Summary

3.2 Application of Bi-directional Evolutionary Structural Optimization (BESO)

3.2.1 Introduction

3.2.2 Sensitivity number derivation

3.2.3 Material interpolation scheme

3.2.4 Non-linear interpolation

3.2.5 Mathematical formulation for BESO

3.2.6 Elements substitutions

3.2.7 Shape optimization

3.2.8 Concluding remarks

Underground excavation design

4.1 Introduction

4.2 Stress distribution in rocks

4.2.1 In-situ stresses

4.2.2 Induced stresses

4.3 Mechanical behaviour of rocks

4.3.1 Intact rock

4.3.2 Mohr-Coulomb criteria

4.3.3 Hoek and Brown criteria

4.3.4 Mechanical properties of rock mass

4.4 Mechanical behaviour of soil

4.5 Effect of reinforcement and support

4.6 Displacement due to excavation
4.6.1 Ground and support interaction ........................................... 90
4.7 Rock reinforcement ........................................................................ 92
4.8 Concluding remarks ..................................................................... 93
Coding and Procedure of BESO ....................................................... 95
5.1 Introduction .................................................................................. 95
5.2 Introduction to MATLAB ............................................................ 95
5.3 Defining material and integration properties .................................. 96
5.4 Importing mesh data .................................................................... 97
5.5 Defining boundary conditions and load location ............................ 99
5.6 Assigning the load variable .......................................................... 101
5.7 Calculation of self-weight function .............................................. 102
5.8 Creating force matrix ................................................................... 104
5.9 Cleaning the work place ............................................................... 106
5.10 Assigning variables to store optimization data .............................. 107
5.11 Finding centroid of elements ....................................................... 108
5.12 Computing compliance matrix ................................................... 109
5.13 Defining the parameters of optimization ...................................... 109
5.14 The optimization cycle ............................................................... 110
  5.14.1 Performing finite element analysis ........................................... 112
  5.14.2 Filtering the sensitivity .......................................................... 121
  5.14.3 Finding unique edges ............................................................ 123
  5.14.4 Non-design filter ................................................................. 126
  5.14.5 Nibbling filter ..................................................................... 127
  5.14.6 Changing the design variable ............................................... 129
  5.14.7 Presentation of the results ..................................................... 130
5.15 Concluding remarks ................................................................... 132
Numerical examples and consideration of non-linearity .................... 133
  6.1 Introduction .................................................................................. 133
  6.2 Short cantilever beam optimization and survey of the effect of optimization parameters .................................................. 133
    6.2.1 Impact of filtering radius .................................................... 134
  6.3 Optimizing underground excavation .......................................... 139
6.4 Reinforcement optimization ................................................................. 143
6.5 Optimization in non-linear materials .................................................. 146
6.6 Strain-based reinforcement optimization ............................................. 148
  6.6.1 The procedure to perform strain based optimization ....................... 150
  6.6.2 Numerical implementation .......................................................... 152
  6.6.3 Geotechnical modelling ............................................................... 153
  6.6.4 Example and discussion .............................................................. 154
6.7 Concluding remarks ............................................................................ 159
Conclusions ............................................................................................ 161
References .............................................................................................. 164
List of Figures
Figure 2.1 Equilibrium of a rod element .............................................................. 19
Figure 2.2 Equilibrium in a thin arbitrary section of a bar element ....................... 19
Figure 2.3 Illustration of external and internal forces ........................................... 23
Figure 2.4 Plain rectangular elements ................................................................. 25
Figure 2.5 A typical brick element with 8 nodes ............................................... 29
Figure 2.6 Quadrilateral elements ....................................................................... 31
Figure 2.7 Quadrilateral element in local coordinate system .................................. 32
Figure 2.8 Integration points and their locations layout for a quadrilateral element .... 35
Figure 2.9 Triangular element ............................................................................... 37
Figure 2.10 Integration points and their locations layout for a triangular element ....... 39
Figure 2.11 Higher order elements ...................................................................... 40
Figure 2.12 A general hexahedron element with 8 nodes ....................................... 40
Figure 2.13 A hexahedron element in the local coordinate system ......................... 41
Figure 2.14 A 4-node tetrahedral element in the local coordinate system ............... 42
Figure 2.15 Degree of freedom for rod elements ................................................... 44
Figure 2.16 General solution of the equation ddxEAdu + F = 0 .............................. 46
Figure 2.17 A rod element with boundary conditions ......................................... 47
Figure 2.18 A mesh of element composed of 4 elements ....................................... 49
Figure 3.1 A typical optimization design domain ................................................. 54
Figure 3.2 A General optimization problem with voids size as the design variable .... 56
Figure 3.3 Flowchart of the SIMP method ............................................................. 60
Figure 3.4 Topology optimization problem of a SCB solved by SIMP method ...... 60
Figure 3.5 Filtering scheme showing elements located in the filtering radius r0 ...... 63
Figure 3.6 Flowchart presenting BESO algorithm ................................................. 65
Figure 3.7 A schematic of a procedure for finding elements located on the edges .... 75
Figure 4.1 Mohr-Coulomb failure envelop .............................................................. 81
Figure 4.2 Hoek-Brown failure compared with Mohr-Coulomb failure ................. 82
Figure 4.3 Measuring the RQD from a core recovered .......................................... 83
Figure 4.4 Illustration of elastic-perfectly plastic behaviour .................................. 86
Figure 4.5 Excavation procedure, involved forces and formation of a force free surface ... 88
Figure 4.6 Distribution of the elastic displacement near the face of a circular tunnel ... 90
Figure 4.7 Schematic of excavation procedure in full face ..................................... 91
Figure 5.1 An illustration of first rows of the nodes location s ................................ 97
Figure 5.2 An illustration of first rows of element connectivity. ........................................ 98
Figure 5.3 The complete imported mesh into MATLAB. .................................................. 99
Figure 5.4 Mesh and forces exerted on the nodes ............................................................. 101
Figure 5.5 Mesh and energy distribution in colour bands ................................................ 116
Figure 5.6 Procedure of filtering sensitivity .................................................................... 123
Figure 5.7 Elements located on the external edges of the shape ...................................... 125
Figure 5.8 Elements located on the internal wall of the opening ........................................ 127
Figure 5.9 Switched element illustration and newly formed faces .................................... 128
Figure 5.10 Designable elements after the first iteration depicted in red ............................ 129
Figure 6.1 mesh, boundary condition and applied force in the SCB example .................... 131
Figure 6.2 Optimized shape of an SCB with Rf=0, and ER=0.02 ...................................... 135
Figure 6.3 Optimized shape of an SCB with Rf=1, and ER=0.02 ...................................... 135
Figure 6.4 Optimized shape of an SCB with Rf=2, and ER=0.02 ...................................... 136
Figure 6.5 Optimized shape of an SCB with Rf=3, and ER=0.02 ...................................... 136
Figure 6.6 Optimized shape of an SCB with Rf=4, and ER=0.02 ...................................... 137
Figure 6.7 Optimized shape of an SCB with Rf=5, and ER=0.02 ...................................... 137
Figure 6.8 Optimized shape of an SCB with Rf=15, and ER=0.02 ................................... 138
Figure 6.9 Plot of the final objective functions using different filtering radius. ................. 138
Figure 6.10 Optimized shapes of an opening under different pressure ratio ...................... 140
Figure 6.11 Optimized shapes of an underground intersection ....................................... 141
Figure 6.12 Optimized shapes of an underground opening using developed program ......... 142
Figure 6.13 Optimized shape of a 3D underground opening ............................................ 143
Figure 6.14 Reinforcement optimization of a 2D non-circular underground opening ......... 145
Figure 6.15 Shape optimization of a half of a 2D pillar with non-linear material .............. 147
Figure 6.16 The optimized non-linear pillar ................................................................. 148
Figure 6.17 The initial domain and discretized slope ...................................................... 154
Figure 6.18 Strain distribution of the initial unreinforced slope ...................................... 156
Figure 6.19 Step by step optimization procedure output ............................................... 159
Figure 6.20 Total maximum shear strain changes during the optimization procedure ......... 159
List of Figures

Table 2.1 Low-order integration rules for rectangular elements ..................................34
Table 2.2 Low-order integration rules for a triangular element ..................................38
Table 2.3 Formation of a stiffness matrix for elements depicted in Figure 2.18 ..........49
Table 6.1 Mechanical properties of the applied materials in the slope ......................154
Abstract

Stability of underground excavation is a major concern because any failure may lead to costly damages and possible loss of life. Since the underground material does not have predefined properties, engineers are exposed to a complex situation. In order to avoid adverse effects of failure, researchers and engineers have tried to fix the problem by taking higher factor of safety for the design of external supporting system, reinforcement and optimizing the shape of underground caverns.

During the last 40 years, advances in technology and application of computer made the job for civil and mining engineers much easier. Therefore, the application of finite element method (FEM) gives the engineers the power to simulate any shape and analyse any real case problems. Finally, as a gap was felt in finding the best shapes of design, engineers searched through different fields to solve it. As a result, the development of optimization methods gave researchers a chance to use computer to design the optimum shape.

Although it was primarily proposed to optimize the shape of structural members, topology optimization has been used by few researchers in geotechnical applications. Despite its great potential, in this field, very few and simple studies have been carried out which were limited to two dimensional designs and the presence of joints, discontinuities, materials with various mechanical properties and junctions as well as material non-linearity were not implemented and investigated thoroughly.

In practical projects, most underground excavations are firstly designed according to the experience of engineers and then numerically studied by simulating it by computer programs thereby predicting the weakness and robustness in the design. But here in this research, computer is used to decide how to excavate and cast reinforcement based on different objective functions.
To do so, a computer program is developed to perform the optimization that includes firstly setting up finite element method and then applying the Bi-directional Evolutionary Structural Optimization (ESO). While there are many methods to perform optimization, due to its robustness in producing reasonable results in designing geotechnical problems and ease of linking with other computer programs (finite elements packages), BESO has been chosen.

Secondly two-dimensional patterns are simulated and used as the verification of correctness of the procedure by comparing with other researchers’ results. Lastly 3D models are considered, analysed and the outcomes are presented. Finally, plasticity is considered and methods to implement measures related to stability of the design domain are discussed.
Introduction

1.1 Human's needs and optimization

In the world, the resources are limited and the request for the material is increasing further. However, economical concerns restrain us from expanding the quality of life more. Thus minimizing the costs and increasing the productivity and efficiency has become a major branch of science in which mathematical equations and statistical analysis are frequently deployed. Therefore, optimization can be viewed as the heart of practical application of science in human needs since the ultimate goal of engineering is to satisfy human’s demands according to the current needs and shortage of resources.

Application of optimization in engineering and physical problems can be traced back through history to 17th century when Newton proposed mathematical method to obtain optimums. The advent of high speed computers paved the way for development of shape optimization. Among the proposed optimization methods, one may name Genetic Algorithm and Linear Programming techniques (Hassani and Hinton 1998a).

In geotechnical engineering, many researchers and engineers have sought to minimize the cost and maximize the stability of their designs through empirical approach and trial and error method. Finding the best shape of a tunnel and the best layout of the supports and finding the most cost effective way to stabilize an instable slope are some examples of seeking optimized pattern in geotechnical engineering.

1.2 Topology optimization in engineering problems

While traditional trial-and-error approach is still the main process to improve the quality and seek the objective of the designation, recent enhancement in computation capability of the computers has made the designing process
dramatically easier. These optimization problems simplified the problem to three different classes which are: Sizing Optimization, Shape Optimization and Topology Optimization (Hassani and Hinton 1998b).

Sizing optimization as the first methods, attempts to adjust the dimensions of the discrete members in a structure while keeping the shape fixed completely. Shape Optimization has higher freedom which means the shape of membranes can change while the topology of the shape remains constant. On the other hand, Topology Optimization has the highest level of freedom and can manipulate all the dimensions of the design shape within the design domain to acquire the optimum shape and topology (Hassani and Hinton 1998b). Therefore, due to less number of applied constraints, it is believed that the achieved topology is more ideal compared to other classes of optimization at the cost of more complex techniques.

1.3 Topology optimization in geotechnical engineering problems
Although recent developments in the optimization methods have made them applicable in other fields of engineering including structure and mechanical engineering, very few steps were taken to incorporate these methods in geotechnical problems. These limited researches only considered optimization in linear homogenous isotropic materials oversimplifying the problems. Also the applications were limited to tunnel design and reinforcement while slope stability, dam designation, minimization of a building settlement through optimizing the shape of foundation and the shape of embankments were not discussed. The applied optimization methods in the geotechnical problems are summarized in the followings.

Application of topology optimization in geotechnical problems started by Yin et al. (2000) when homogenisation method was deployed to optimize the reinforcement of a tunnel in an elastic medium. In this article, the reinforced elements were original elements (without reinforcement) enveloped in reinforced material and the element overall properties were estimated through the homogenisation method. Keeping the volume of the reinforcement material constant, the method attempted
to minimize the external work along the tunnel walls. Then Yin and Yang (2000a) expanded the application to include different geological layers structure using Solid Isotropic Microstructures with Penalization (SIMP). Also, tunnel heave minimization problem was addressed by Yin and Yang (2000b) using SIMP and Liu et al. (2008) using Fixed-Grid Bidirectional Evolutionary Structural Optimization (FG BESO) which curbed mesh dependency problem. Furthermore, Ren et al. (2005) investigated the optimum shape of underground excavations using stress as the rejection ratio.

With the advent in high speed computers, the optimization methods were further expanded to consider material non-linearity in geotechnical design problems. In this regard, Nguyen et al. (2014) considered reinforcement distribution optimization by improving the BESO method to incorporate sensitivity analysis of nonlinear material. Recently, Ren et al. (2014) explored the shapes of underground excavations in non-linear domain. A review of their work is explained accompanied with the methodology of the optimization and the numerical problems causing divergence in the procedure. Before that, a chapter has been devoted to Finite Element Method (FEM) to understand the procedure and parameters that affect the output because the FEM is an inseparable part of the proposed optimization method and understanding the detail will provide readers with better insights. Also knowledge about FEM can help the readers to find out the problems in compliance based optimization and why it is better to consider the change in the optimization criterion in order to increase stability of the design.

Understanding the features of underground environment is a significant step toward tailoring the optimization method. Existence of joints, shear zones, geotechnical structural features, beddings, ground-water and discontinuities renders the rock mass inhomogeneous, anisotropic and non-linear. Also different loading sequences and other loading steps make this environment more unpredictable and unknown. Moreover, excavation and support installation sequence have impacts on the stress distribution pattern and therefore change the domain respond to the load. The detail of regarding geomaterial properties has been introduced in chapter 4.

Through this thesis, some measures are taken to address some of the features of geotechnical problems and also propose a new criterion for the optimization to
consider stability. It is noticed that all other optimization methods perform the process to gain the stiffest shape and layout, while here stability of the design, which is the main concern of geotechnical engineers, are discussed and sought to be improved.

It should however be noticed that the proposed method is still very new and much more improvement are needed to make this method applicable for geotechnical engineering problems. But, there is hope that the current research would pave the way for other researchers to incorporate more complicated cases and finally develop a computer program that can design reinforcement and shape proper for the current geotechnical problem in hand.

1.4 Evolutionary Structural Optimization (ESO)
The binary optimization method was firstly introduced by proposition of Evolutionary Structural Optimization (ESO) by Xie and Steven (1993). The concept of this method was inchmeal elimination of elements found to be inefficient in order to find the optimum shape over several iterations. The original ESO used intuitive criterions to judge the materials as inefficient and thus lacked an objective function. Consequently, Chu et al. (1996) presented compliance based objective function which enabled the procedure to be supported by mathematical rationale. By associating both forward and retrieving behaviour into the ESO, BESO was born, allowing both removing inefficient elements and also adhering elements to the vicinity of insufficiently strong areas (Querin et al. 2000). Having clear edges, easy coding and linkage to the finite element solver packages are among the advantages of this method which have resulted in attraction of many researchers to this approach.

1.5 Layout of the thesis
The next chapter presents basics of the finite element method and different steps to formulate it. After focusing on mechanical equilibrium of an object, various element types are discussed. Then assembly of numerous elements are also noticed.
Chapter 3 introduces and provides a brief review of the optimization methods and compares the advantages and shortcomings of the proposed methods. Also numerical modelling and problems with methods to confront these problems are deliberated.

Chapter 4 provides a general description of the conditions in geotechnical design. The material properties, loading characteristics and support interaction with the domain are reviewed.

Chapter 5 deals with implementing the proposed routine in the coding language and provides the reader with step-by-step explanation of the procedure. The details regarding the FEM are also brought up to make the total process an integrated and comprehensive part which enables the reader to follow the same steps for further improvement.

Chapter 6 is allocated to the provided examples, considering material non-linearity and application of the offered approach. Firstly, a Short Cantilever Beam (SCB) is discussed. Then tunnel shape optimization and reinforcement optimization are deliberated. Additionally, rationale about stability is discussed and the optimization is extended to incorporate stability.
Finite element analysis and programming

Many text books specify the fundamentals of finite element analysis and its wide application have made this method an important part of engineering. Since one of the main parts of optimization is performing finite element, knowledge about its details would give us an advantage to manipulate it to obtain needed output, thus a section has been devoted to it.

2.1 Discretization of the domain

2.1.1 Introduction
The FEM is a numerical technique to untangle complex partial differential equations by discretizing the equations to limited space domain. This procedure segregates the domain to small and simple shapes with known physical and mathematical features (Finite Elements) and then approximates the complex equations by connecting these sub-domains together. To do so, a matrix equation is required to link input data at a determined point (node) in each element to the results of that point. Also, the constructed elements must resemble the original shape as closely as possible.

FEM method seeks one of the following two predefined methods to solve the equation over the huge region: summation of matrix equations in each node which results in composition of a large global matrix equation, or solving the equations of nodes in each element and then adding the element results to each other (Zienkiewicz and Taylor 1989).

Since FEM is a huge process, simple single-element shapes will be firstly focused and then attention is turned to shapes with more elements and complexity.
2.1.2 One dimensional bar element

Geometrically, the simplest element is one dimensional element with two nodes at each of its ends (Smith and Griffiths 2004). Although this element might be presented with two or three dimensions, but transitions are calculated only in one dimension.

\[
L \times \delta x \quad F(x_1) \quad F(x_2) \quad u_1 \quad \delta x \quad u_2 \quad \text{nodal displacement}
\]

Figure 2.1 Equilibrium of a rod element.

\[
P = A \sigma = EA \varepsilon = EA \frac{du}{dx}
\] (2.1)

Figure 2.2 Equilibrium in a thin arbitrary section of the element.

Figure 2.1 illustrates an elastic bar element with the length L, surface A, u as axial displacement of each node and E as its elastic modulus. By considering an applied force (P) to a small section of the bar (as depicted in Figure 2.2) and F as applied internal force, the equilibrium equation of the section will yield the following term.
Where $\sigma$ and $\varepsilon$ are stress and strain respectively. Thus the force equilibrium of the thin section can be stated as

$$\frac{dp}{dx} + F = 0$$

(2.2)

Thus, by substituting (2.1) in (2.2), one can write

$$EA \frac{du^2}{dx^2} + F = 0$$

(2.3)

As mentioned earlier, in FEM, all of values and results of each element are calculated and stored at the nodes corresponding to the element. Since the bar element has two nodes, the differential equation should be solved based on its nodal values. In order to do so, a shape function (Ferreira 2009) to approximate the displacement should be applied, which is

$$\ddot{u} = N_1 u_1 + N_2 u_2$$

(2.4)

or in its matrix form

$$\ddot{u} = [N_1 \ N_2] \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

(2.5)

where

$$N_1 = \frac{x}{L}, N_2 = 1 - \frac{x}{L}$$

(2.6)

By replacing (2.5) in (2.3), it can be concluded

$$EA \frac{d^2}{dx^2} \left[ N_1 \ N_2 \right] \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} + F = 0$$

(2.7)

However, it should be considered that this equation is an approximation and it always has a residual. Application of higher order shape functions or smaller elements can boost accuracy of the calculation. Now it is only required to find appropriate values for $u_1$ and $u_2$.

Galerkin’s method is one of the best and common ways to solve the above Equation (Griffiths and Smith 1991) and in order to find $u_1$ and $u_2$, all terms should be
multiplied by shape function, and then integrated over the length of the element, as follows (Strang 1973)

\[
\int_0^L \left\{ \begin{aligned}
N_1 \\
N_2
\end{aligned} \right\} \frac{d^2}{dx^2} [N_1 \ N_2] dx \left\{ \begin{aligned}
u_1 \\
u_2
\end{aligned} \right\} + \int_0^L \left\{ \begin{aligned}
N_1 \\
N_2
\end{aligned} \right\} F dx = \left( \begin{aligned}0 \\
0\end{aligned} \right) \tag{2.8}
\]

It should be noticed that double differentiation over linear shape functions will result in disappearing them, thus Green’s theorem (Kaplan 1991) should be implemented which is

\[
\int N_i \frac{\partial^2 N_j}{\partial x^2} dx \left\{ \begin{aligned}u_1 \\
u_2\end{aligned} \right\} \simeq - \int \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} dx \tag{2.9}
\]

By supposing applied force acting only on the nodes and E and A to be constant over the length of element, the Equation (2.8) can be written as (Szabo and Lee 1969)

\[-EA \int_0^L \begin{bmatrix}
\frac{\partial N_2}{\partial x} & \frac{\partial N_1}{\partial x} \\
\frac{\partial N_2}{\partial x} & \frac{\partial N_2}{\partial x}
\end{bmatrix} \begin{bmatrix} \frac{\partial N_1}{\partial x} \\
\frac{\partial N_2}{\partial x}
\end{bmatrix} dx \left\{ \begin{aligned}u_1 \\
u_2\end{aligned} \right\} + F \int_0^L \left\{ \begin{aligned}
N_1 \\
N_2
\end{aligned} \right\} dx = \left( \begin{aligned}0 \\
0\end{aligned} \right) \tag{2.10}
\]

When \( N_1 \) and \( N_2 \) from (2.5) are substituted in Equation (2.10), it can be deduced

\[-EA \int_0^L \begin{bmatrix}
-\frac{1}{L} & \frac{1}{L} \\
\frac{1}{L} & -\frac{1}{L}
\end{bmatrix} dx \left\{ \begin{aligned}u_1 \\
u_2\end{aligned} \right\} + F \begin{bmatrix} \frac{L}{2} \\
\frac{L}{2}
\end{bmatrix} = \left( \begin{aligned}0 \\
0\end{aligned} \right) \tag{2.11}
\]

or by simplification

\[
\frac{EA}{L} \begin{bmatrix}
-1 & 1 \\
1 & -1
\end{bmatrix} \left\{ \begin{aligned}u_1 \\
u_2\end{aligned} \right\} = \left\{ F(x_1) \right\} \tag{2.12}
\]

where \( F(x_1) \) and \( F(x_2) \) denote forces exerted at node 1 and 2 respectively. By assuming the first matrix as element stiffness matrix,

\[
[k_m] = \frac{EA}{L} \begin{bmatrix}
-1 & 1 \\
1 & -1
\end{bmatrix} \tag{2.13}
\]

and using element force and displacement vector, one will have

\[
[k_m](u) = (F) \tag{2.14}
\]
which is generalized equation of elastic displacement in finite element (Rao 2004).

2.1.3 Alternative approach to deduce element equilibrium

An alternative approach in discovering attributes of element is noticing energy equilibrium. Through the following formulas, it will be shown how to attain the element stiffness.

Consider a small length δx of an elastic bar illustrated in Figure 2.1. The strain energy (internal energy hoarded in element) due to axial displacement is

\[ W_i = \frac{1}{2} \int_{\text{vol}} d\varepsilon \, d\sigma \, dV \]  

or in its matrix formation

\[ W_i = \frac{1}{2} \int_{\text{vol}} \{d\varepsilon\}^T \{d\sigma\} \, dV \]  

And the external work applied to the body can be written as

\[ W_e = \frac{1}{2} \, du \, dF \]  

or in its matrix formation

\[ W_e = \frac{1}{2} \{du\}^T \{dF\} \]

Generally, the force exerted on an element would either be externally exerted load or internal loads including the element self-weight. As mentioned earlier, the core theory of FEM is calculation of force, either external or internal, and also displacements at the nodes of the corresponding element and substituting the distributed values with the values at these nodes by means of the integration (Sherif 2012). Thus, a function is demanded to convert these distributed values to nodal amounts which can be concluded as
\[
\{F\} = \int_{vol} [N]^T \{\omega\} \, dV + \int_{s} [N]^T \{T\} \, dS
\]  
(2.19)

\[
\{\omega\} = \begin{bmatrix} 0 \\ -\gamma \end{bmatrix}
\]  
(2.20)

where \( \gamma \) is the material density of element. Since the force is only pulling the element down, it has no component in lateral direction (x-direction). Since this force has opposite direction to the assumed direction of the force and displacement, the negative sign is used to represent it. The parameter T represents surface traction or externally applied force on surface of element and [N] is the shape function matrix of element.

Figure 2.3 illustrates the external and internal forces, depicted by red and blue arrows respectively.

Strain which is the relative deformation of the element to the original length of it can be expressed as differentiation of the displacement to the lateral component (in
the one dimensional example). Thus, by assuming \( O \) as the operator \( d/dx \), one will have

\[
\{d\varepsilon\} = ON\{d\varepsilon\}
\]  

(2.21)

We can assume \( B \) as

\[
B = ON
\]  

(2.22)

to make the equations simpler. Also general relation of stress and strain should be considered which is

\[
\{d\sigma\} = C\{d\varepsilon\}
\]  

(2.23)

where the coefficient \( C \) represents compliance matrix. When substituting (2.22) and (2.23) into (2.16), it will become

\[
W_i = \frac{1}{2} \int_{vol} \{du\}^T [B]^T [C] [B] \{du\} \, dV
\]  

(2.24)

By similar calculation of external energy, and considering state of equilibrium the whole formula gets the form of (Bathe and Wilson 1976)

\[
W_i = W_e
\]  

(2.25)

or

\[
\frac{1}{2} \int_{vol} \{du\}^T [B]^T [C] [B] \{du\} \, dV = \frac{1}{2} \{du\} \{dF\}
\]  

(2.26)

It should be noted again that displacements have nodal values and can be easily excluded from integration, thus

\[
\{dF\} = \int_{vol} [B]^T [C] [B] dV \{du\}
\]  

(2.27)

If stiffness is assumed as
\[
[K_e] = \int_{vol} [B]^T[C][B]dV
\]  

(2.28)

the traditional linear equation is obtained

\[
\{dF\} = [K_e]\{du\}
\]

(2.29)

### 2.1.4 Two-dimensional elements

The previous example was in one dimension while the real world problems are 3 dimensional or by some simplification, in 2 dimensions. In this manner, elements can be connected to each other at the nodes and results in more accurate solution for the equations. The simplest form of two-dimensional element is rectangular element in which 4 nodes are located at its 4 corners. Now, consider a wall discretized into rectangular element with unit thickness and forces acting on its top as shown in Figure 2.4.

![Figure 2.4 Plain rectangular elements: a) A wall composed of rectangular elements; b) a single 4-node rectangular element.](image)
As it was mentioned in one dimensional example, potential energy approach can be used to derive element stiffness matrix (Livesley 1975). The stored energy is

$$W_i = \frac{1}{2} \int \int \{\sigma\}^T \{\varepsilon\} \ dx \ dy$$

(2.30)

which can be written as

$$W_i = \frac{1}{2} \{u\}^T \int \int ([O][N])^T \ C \ ([O][N]) \ dx \ dy \ \{u\}$$

(2.31)

or

$$W_i = \frac{1}{2} \{u\}^T \int \int [B]^T \ C \ [B] \ dx \ dy \ \{u\}$$

(2.32)

By assuming plain stress mode, one will have (Timoshenko and Goodier 1982)

$$[O] = \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial y} \end{bmatrix}, \quad [C] = \frac{E(1-v)}{(1+v)(1-2v)}$$

(2.33)

and as usual in finite element method, shape function can be defined as

$$\tilde{u} = [N_1 \ N_2 \ N_3 \ N_4] \ \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = [N] \{u\}$$

$$\tilde{v} = [N_1 \ N_2 \ N_3 \ N_4] \ \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = [N] \{v\}$$

(2.34)

with

$$[U] = \begin{bmatrix} N_1 & 0 \\ 0 & N_1 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}$$

(2.35)

in which \(u\) and \(v\) are the x and y components of the displacement and \(U\) is total displacement. Thus, the shape function matrix for the element can be written as
\[
[N] = \begin{bmatrix}
N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4 & 0 \\
0 & N_1 & 0 & N_2 & 0 & N_3 & 0 & N_4
\end{bmatrix}
\] (2.36)

According to rectangular element illustrated in Figure 2.4b, shape function N can be derived as

\[
N_1 = \left(1 - \frac{x}{a}\right)\left(1 - \frac{y}{b}\right)
\]

\[
N_2 = \frac{x}{a} \left(1 - \frac{y}{b}\right)
\]

\[
N_3 = \frac{xy}{ab}
\]

\[
N_4 = \left(1 - \frac{x}{a}\right)\frac{y}{b}
\]

Thus, \([B]=[O][N]\), results in

\[
[B] = \begin{bmatrix}
\frac{\partial N_1}{\partial x} & 0 & \frac{\partial N_2}{\partial x} & 0 & \frac{\partial N_3}{\partial x} & 0 & \frac{\partial N_4}{\partial x} & 0 \\
0 & \frac{\partial N_1}{\partial y} & 0 & \frac{\partial N_2}{\partial y} & 0 & \frac{\partial N_3}{\partial y} & 0 & \frac{\partial N_4}{\partial y} \\
\frac{\partial N_1}{\partial y} & \frac{\partial N_2}{\partial y} & \frac{\partial N_2}{\partial y} & \frac{\partial N_3}{\partial y} & \frac{\partial N_3}{\partial y} & \frac{\partial N_4}{\partial y} & \frac{\partial N_4}{\partial y}
\end{bmatrix}
\] (2.38)

Also as expected, the stiffness of the element can be stated as

\[
[K_e] = \int \int [B]^T \ [C] \ [B] \ dx \ dy
\] (2.39)

with strain-displacement relationship (Timoshenko and Goodier 1982)

\[
\varepsilon = B \ U,
\]

\[
\begin{bmatrix}
\varepsilon_x \\
\varepsilon_y \\
\gamma_{xy}
\end{bmatrix} =
\begin{bmatrix}
\frac{\partial}{\partial x} & 0 \\
0 & \frac{\partial}{\partial y}
\end{bmatrix}
\begin{bmatrix}
U \\
V
\end{bmatrix}
\] (2.40)

Entirely the same terms are held for plain stress mode, but compliance matrix \([C]\) should be replaced by
\[ [C] = \frac{E}{(1 - v^2)} \begin{bmatrix} 1 & v & 0 \\ v & 1 & 0 \\ 0 & 0 & \frac{1 - v}{2} \end{bmatrix} \] (2.41)

### 2.1.5 Three-dimensional strain and stress

When another dimension is added to the stiffness equations, integration should contain this dimension and the stiffness takes the form of

\[ [K_e] = \int \int \int [B]^T [C] [B] \, dx \, dy \, dz \] (2.42)

where the strain-displacement matrix (Timoshenko and Goodier 1982) is

\[ \varepsilon = BU, \begin{bmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \] (2.43)

The simplest type of 3D element, is a 8-node brick element with length of a, b and c, as shown in Figure 2.5, and the shape function is given by

---

28
The full shape matrix is

$$
[N] = \begin{bmatrix}
N_1 & N_2 & N_3 & N_4 & N_5 & N_6 & N_7 & N_8 \\
\end{bmatrix}
$$

(2.44)

$$
N_1 = \left(1 - \frac{x}{a}\right)\left(1 - \frac{y}{b}\right)\left(1 - \frac{z}{c}\right)
$$

$$
N_2 = \frac{x}{a}\left(1 - \frac{y}{b}\right)\left(1 - \frac{z}{c}\right)
$$

$$
N_3 = \frac{x}{a}\left(1 - \frac{y}{b}\right)\frac{z}{c}
$$

$$
N_4 = \left(1 - \frac{x}{a}\right)\left(1 - \frac{y}{b}\right)\frac{z}{c}
$$

(2.45)

$$
N_5 = \left(1 - \frac{x}{a}\right)\frac{y}{b}\left(1 - \frac{z}{c}\right)
$$

$$
N_6 = \frac{xy}{ab}\left(1 - \frac{z}{c}\right)
$$

$$
N_7 = \frac{xyz}{abc}
$$

$$
N_8 = \left(1 - \frac{x}{a}\right)\frac{yz}{bc}
$$
\[ [N] = \begin{bmatrix}
N_1 & 0 & 0 & N_2 & 0 & 0 & \cdots & N_8 & 0 & 0 \\
0 & N_1 & 0 & 0 & N_2 & 0 & \cdots & 0 & N_8 & 0 \\
0 & 0 & N_1 & 0 & 0 & N_2 & \cdots & 0 & 0 & N_8
\end{bmatrix} \] (2.46)

and the elastic stress-strain matrix is in the form

\[
[C] = \frac{E(1-v)}{(1+v)(1-2v)} \begin{bmatrix}
1 & \frac{v}{1-v} & \frac{v}{1-v} & 0 & 0 & 0 \\
\frac{v}{1-v} & 1 & \frac{1-v}{v} & 0 & 0 & 0 \\
\frac{1-v}{v} & \frac{v}{1-v} & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1-2v}{2(1-v)} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{1-2v}{2(1-v)} & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1-2v}{2(1-v)}
\end{bmatrix}
\] (2.47)

### 2.2 General finite element formulation

#### 2.2.1 Introduction

In the previous section, the governing equations of simple elements were discussed. In this section, by attaching new attributes, the application will be expanded to solve more complex equations. Firstly, due to their simplicity, basic aforementioned elements are not capable of presenting complex shapes with irregular formations. Hence, to arrive at a more general solution, it is needed to introduce general shapes (including triangular elements in 2D and tetrahedral elements in 3D) with regard to implication of local coordinate. Secondly, so far, examples included only one element, while an appropriate analysis should compromise many elements to resemble desired shape. Also, boundary conditions as an essential parts to solve problems needs to be discussed.
2.2.2 Local coordinate for quadrilateral elements

In section 2.1.4 a shape function was constructed for a rectangular element, which was \( N_1 = (1-x/a)(1-y/b) \) and so on. Formation of these functions for a general quadrilateral element in global coordinate, \( x \) and \( y \) is complicated and requires complex algebraic processes.

Instead, as shown in Figure 2.6, denoting the location of each point within the element \( p(\lambda,\mu) \), using local coordinate is an appropriate way to deal with this problem (Taig 1961). It is notable that in this way, the coordinates should be normalized to make numerical integration easier (refer to Equation (2.55)). So, with regard to the Figure 2.7, the edge 14 has a value of \( \mu=-1 \), edge 23 has \( \mu=1 \), edge 12 has \( \lambda=-1 \) and edge 34 has \( \lambda=1 \). Thus, it can be seen in the Figure 2.7 (b) that by application of local coordinate system, the quadrilateral element has become a simple rectangular elements allowing easier integration.

Figure 2.6 Quadrilateral elements: a) Rectangular element; b) general quadrilateral element
Figure 2.7 Quadrilateral element in local coordinate system: a) local coordinate system; b) deformed quadrilateral element.

Now, according to local coordinate system, the shape functions of this quadrilateral element become

\[ N_1 = \frac{1}{4} (1 - \mu)(1 - \lambda) \]
\[ N_2 = \frac{1}{4} (1 + \mu)(1 - \lambda) \]
\[ N_3 = \frac{1}{4} (1 + \mu)(1 + \lambda) \]
\[ N_4 = \frac{1}{4} (1 - \mu)(1 + \lambda) \]

which can be used to define changes in displacement and force at any point throughout element. Surely, there is a necessity to convert these local coordinates, namely \( \lambda \) and \( \mu \), to global coordinates \( x \) and \( y \). Because of their isoparametric nature (Zienkiewicz 1989), these functions also can be used to determine the relation between local and global coordinates. So the conversion equations are

\[ x = N_1 x_1 + N_2 x_2 + N_3 x_3 + N_4 x_4 \]
\[ y = N_1 y_1 + N_2 y_2 + N_3 y_3 + N_4 y_4 \]

(2.48)

(2.49)

or in their matrix form
\[ x = [N] \{x\} \]
\[ y = [N] \{y\} \tag{2.50} \]

where \( \{x\} \) and \( \{y\} \) are the nodal coordinates (which are also the global coordinates).

Also, to obtain \([B]\), there is need for derivatives of these functions. However these functions have their local variable \( \lambda \) and \( \mu \), and taking their derivative should be done by means of chain rule as follows

\[
\frac{\partial N}{\partial x} = \frac{\partial N}{\partial \lambda} \frac{\partial \lambda}{\partial x} + \frac{\partial N}{\partial \mu} \frac{\partial \mu}{\partial x} \]
\[
\frac{\partial N}{\partial y} = \frac{\partial N}{\partial \lambda} \frac{\partial \lambda}{\partial y} + \frac{\partial N}{\partial \mu} \frac{\partial \mu}{\partial y} \tag{2.51} \]

or alternatively in their matrix forms

\[
\begin{bmatrix} \frac{\partial N}{\partial x} \\ \frac{\partial N}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial \lambda}{\partial x} & \frac{\partial \mu}{\partial x} \\ \frac{\partial \lambda}{\partial y} & \frac{\partial \mu}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial N}{\partial \lambda} \\ \frac{\partial N}{\partial \mu} \end{bmatrix} \tag{2.52} \]

If Jacobian matrix is set as the relation between derivatives in local coordinate to global coordinate, as shown in Equation (2.53)

\[
[J] = \begin{bmatrix} \frac{\partial x}{\partial \lambda} & \frac{\partial x}{\partial \mu} \\ \frac{\partial y}{\partial \lambda} & \frac{\partial y}{\partial \mu} \end{bmatrix} \tag{2.53} \]

it can be written

\[
\begin{bmatrix} \frac{\partial N}{\partial x} \\ \frac{\partial N}{\partial y} \end{bmatrix} = [J]^{-1} \begin{bmatrix} \frac{\partial N}{\partial \lambda} \\ \frac{\partial N}{\partial \mu} \end{bmatrix} \tag{2.54} \]

It should be noted that transformation of integration from local to global coordinates requires determinant of Jacobian matrix, recognized as ‘The Jacobian’, which can be expressed as
\[
\int \int dx \, dy = \int_{-1}^{1} \int_{-1}^{1} \det |J| \, d\lambda \, d\mu
\]  
\(2.55\)

### 2.2.3 Integration for quadrilateral elements

To solve the integral, it is convenient and common in most of FEM packages to apply numerical methods including Gauss-Legendre quadrature. The quadrature rules provide an accurate approximation of an integral of a function, generally remarked as a weighted sum of function values at particular points in the interval of the integral. Also, as long as the function is continues and without singularity, the approximation results in accurate output. One can write weighted sum function as

\[
\int_{-1}^{1} f(x) \, dx \approx \sum_{i=1}^{n} w_i \, f(x_j)
\]  
\(2.56\)

and in the case of double integration on two variables, it can be presented as

\[
\int_{-1}^{1} \int_{-1}^{1} f(\lambda, \mu) \, d\lambda \, d\mu \approx \sum_{i=1}^{n} \sum_{j=1}^{n} w_i \, w_j \, f(\lambda_i, \mu_j)
\]

\[
\approx \sum_{i=1}^{nip} W_i \, f(\lambda_i, \mu_i)
\]  
\(2.57\)

where \(nip=n^2\) represents number of integration points, \(w_i, w_j\) and \(W_i = w_i w_j\) are weighting coefficients and \(f(\lambda_i, \mu_i)\) is the value of function at the sampling points.

Some low-order integration rules for a rectangle are listed below (Kopal 1961).

<table>
<thead>
<tr>
<th>N</th>
<th>((\lambda_i, \mu_i))</th>
<th>(W_i = w_i w_j)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(0,0)</td>
<td>4=2*2</td>
</tr>
<tr>
<td>2</td>
<td>((\pm \frac{1}{\sqrt{3}}, \pm \frac{1}{\sqrt{3}}))</td>
<td>1=1*1</td>
</tr>
</tbody>
</table>

Table 2.1 Low-order integration rules for rectangular elements.
<table>
<thead>
<tr>
<th>3</th>
<th>4 corner points:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(\pm \sqrt{\frac{3}{5}}, \pm \sqrt{\frac{3}{5}})$</td>
<td></td>
</tr>
</tbody>
</table>

2 vertical points:

$(0, \pm \sqrt{\frac{3}{5}})$

2 horizontal points:

$(\pm \sqrt{\frac{3}{5}}, 0)$

1 centre point

$(0,0)$

<table>
<thead>
<tr>
<th>25</th>
<th>$\frac{5}{9} \times \frac{5}{9}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>81</td>
<td>$\frac{5}{9} \times \frac{5}{9}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>40</th>
<th>$\frac{5}{9} \times \frac{8}{9}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>81</td>
<td>$\frac{5}{9} \times \frac{8}{9}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>40</th>
<th>$\frac{8}{9} \times \frac{5}{9}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>81</td>
<td>$\frac{8}{9} \times \frac{5}{9}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>64</th>
<th>$\frac{8}{9} \times \frac{8}{9}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>81</td>
<td>$\frac{8}{9} \times \frac{8}{9}$</td>
</tr>
</tbody>
</table>

(a) (b) (c)
Figure 2.8 Integration points and their locations layout for a quadrilateral element with: a) nip=1; b) nip=4; c) nip=9.

Increasing the number of integration points will result in higher accuracy, but at the cost of computer memory usage. Also, depending on the element type chosen, the number of integration points should be adjusted.Attributing a high integration order to a simple element like Q4 which is 4-node quadrilateral (a quadrilateral element with 4 nodes at its corners, as discussed above) is a waste of memory and computation time. Instead, to achieve more precise results, it is recommended to use higher order element like Q8 or Q9 (Bhavikatti 2005).

Analysing a quadrilateral element in plain strain, like rectangular element discussed in section 2.1.4, requires the following integral to be formed and solved

$$ [K_e] = \int \int [B]^T [C] [B] \, dx \, dy $$

(2.58)

In most cases, this equation can be accurately estimated by using 4-points Gaussian quadrature for a 4-node quadrilateral element. In this approach, the contribution of each integration points is calculated and then added together as shown here

$$ [K_e] \approx \sum_{i=1}^{4} W_i \det |J_i| ([B]^T [C] [B])_i $$

(2.59)

Similarly, this technique can be applied to other types of element.

### 2.2.4 Local coordinate for triangular elements

Shape function for a triangular element is easily assumed to be based on a right angle triangle with equal length of a unit for each of its sides, as depicted in Figure 2.9.
Figure 2.9 Triangular element: a) in global coordinate system; b) deformed triangular element according to its local coordinate system.

While any point in the triangle can be located using two variables of local coordinate \((L_1, L_2)\) (Zienkiewicz et al. 1971), application of a third coordinate makes the calculation simpler

\[
L_3 = 1 - L_1 - L_2
\]  

(2.60)

So the shape functions can be given by

\[
N_1 = 1 - L_1 - L_2
\]

\[
N_2 = L_1
\]

\[
N_3 = L_2
\]

(2.61)

Similarly, coordinates convertor is written as

\[
x = N_1x_1 + N_2x_2 + N_3x_3
\]

\[
y = N_1y_1 + N_2y_2 + N_3y_3
\]

(2.62)

or in its matrix form

\[
x = [N]\{x\}
\]

\[
y = [N]\{y\}
\]

(2.63)
where \( \{x\} \) and \( \{y\} \) represent the nodal coordinates.

It is worth mentioning that derivative equations and Jacobian matrices are still viable, however as variable changed, Equation (2.59) should also be changed to

\[
\int \int dx \, dy = \int_0^1 \int_0^{1-L_1} \det |J| \, dL_1 \, dL_2
\]

(2.64)

to incorporate the new transformation equation.

Similar to quadrilateral element, the application of numerical integration for triangle is in form of

\[
\int_0^1 \int_0^{1-L_1} f(L_1, L_2) \, dL_1 \, dL_2 \approx \sum_{i=1}^{nip} w_i \, f(L_1, L_2)_i
\]

(2.65)

Where \( nip \) is number of integration points and \( w_i \) is the weighting coefficient for sampling point. \( f(L_1, L_2)_i \) is the value of the function at the i-th sampling point. The following table contains the location of the integration points and the associated weighting coefficient for some low-order integration rules in a triangular element.

Table 2.2 Low-order integration rules for a triangular element.

<table>
<thead>
<tr>
<th>Nip</th>
<th>((L_1, L_2)_i)</th>
<th>(w_i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(\left(\frac{1}{3}, \frac{1}{3}\right))</td>
<td>(\frac{1}{2})</td>
</tr>
<tr>
<td>3</td>
<td>(\left(\frac{1}{2}, 0\right))</td>
<td>(\frac{1}{6})</td>
</tr>
<tr>
<td></td>
<td>(\left(0, \frac{1}{2}\right))</td>
<td>(\frac{1}{6})</td>
</tr>
<tr>
<td></td>
<td>(\left(\frac{1}{2}, \frac{1}{2}\right))</td>
<td>(\frac{1}{6})</td>
</tr>
</tbody>
</table>
Figure 2.10 Integration points and their locations layout for a triangular element with: a) nip=1; b) nip=3.

2.2.5 Higher order 2D elements
In considering 2D elements, up to now, the focus was on the simplest rectangular and triangular elements. However to increase capability of elements to deform, it is required to increase the number of nodes on each element or use smaller mesh of elements which can be quite time consuming. Thus, to remedy the coarseness problem, nodes are added to elements resulting in easier deformation of the elements.

Generally speaking, external nodes (nodes on the surfaces of elements) can be divided into two categories. Primary nodes which are the ones located at corner of the shapes (as discussed so far), and secondary nodes which are nodes located on the edges but not at the corners, as shown in Figure 2.11.

It is a convention that elements are named by their type and number of their points. For example, a triangle with 3 nodes is called ‘T3’ in which ‘T’ is abbreviation of Triangle and ‘3’ is number of the points. Also a quadrilateral element has ‘Q’ representing ‘Quadrilateral’ and it is followed by number points like ‘Q4’. Conventionally, triangle with 3, 6, 10 or 15 nodes (T3, T6, T10, T15) and quadrilateral element with 4, 8 or 9 nodes are admissible and common.
2.2.6 Three-dimensional elements

When extending application of the 3D brick element to a more general form, application of local coordinate through introduction of shape function is essential. Thus, as depicted in Figure 2.12, by considering the local coordinate system \((\mu, \lambda, \xi)\), the cuboidal element can be formed as

![Figure 2.11 Higher order elements: a) a 6-node triangular element; b) a 8-node quadrilateral element.](image)

![Figure 2.12 A general hexahedron element with 8 nodes.](image)
Figure 2.13 A hexahedron element in the local coordinate system.

with the shape function of

\[
[N] = \begin{bmatrix}
\frac{1}{8}(1 - \mu)(1 + \lambda)(1 + \xi)
\frac{1}{8}(1 - \mu)(1 - \lambda)(1 + \xi)
\frac{1}{8}(1 - \mu)(1 - \lambda)(1 - \xi)
\frac{1}{8}(1 - \mu)(1 + \lambda)(1 - \xi)
\frac{1}{8}(1 + \mu)(1 + \lambda)(1 + \xi)
\frac{1}{8}(1 + \mu)(1 - \lambda)(1 + \xi)
\frac{1}{8}(1 + \mu)(1 - \lambda)(1 - \xi)
\frac{1}{8}(1 + \mu)(1 + \lambda)(1 - \xi)
\end{bmatrix} \tag{2.66}
\]

All other properties of the functions remain unchanged, but stiffness of element can be calculated through

\[
[K_e] = \int \int \int [B]^T [C] [B] \, dx \, dy \, dz \tag{2.67}
\]
2.2.7 Tetrahedral elements
Other optional element in 3D coordinate is tetrahedron, which in its simples form is composed of 4 nodes at its corners. In this element, local system consist a right angled tetrahedron with its nodes located at unit length of each axis and one at the origin of the coordinates.

![Diagram of a 4-node tetrahedral element in its local coordinate system.]

Figure 2.14 A 4-node tetrahedral element in its local coordinate system.

Although this method can map all points within the element using three coordinate ($L_1$, $L_2$, $L_3$), inclusion of the fourth coordinate $L_4$ given by

$$ L_4 = 1 - L_1 - L_2 - L_3 \quad (2.68) $$

provides simplicity in mathematical calculation. Therefore, the shape function becomes

$$ [N] = \begin{bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \end{bmatrix} \quad (2.69) $$

And all other governing equations remain similar as given in the previous sections.
2.2.8 Multi-element assemblies
Assembly of multi-element stiffness firstly requires formation of each element stiffness matrix (what was explained up to now) and then mapping them in global stiffness matrix according to location allocated to them. The augmentation is carried out in a way that properties of each element directly sit in global stiffness matrix. The mapping concept which transforms element local properties to global matrix is called degree of freedom (Felippa 2004) and will be explained in the proceeding section.

2.2.9 Degree of freedom (DOF) of elements
Degree of freedom indicates number of parameters allowed to vary in a mechanical system. Since loads and movement are calculated at each node, their degree of freedom is of significance. Literally, DOF defines ability of movement or rotation in each node due to exerted load. In 2 dimensions, any element or point has 3 degree of freedom, which is movement along x direction, y direction and rotation around the z axis. In three dimension, any point has 6 degree of freedom which are translations in directions of x, y, z and rotations about the x, y or z axis.

DOF also describes how loads and moments are transferred to neighbouring elements. For example, an ideal door hinge does not transfer any rotational moment whereas it is efficient in carrying tension, pressure and shear. As another instance, a rope only transfers tension, not pressure, shear or rotation. Figure 1.15a shows local degree of freedom. DOF of each element is dependent on DOF of its nodes.

2.2.10 Global degree of freedom
Since each node has 3 DOFs in 2D and 6 DOFs in 3D, 3 and 6 spaces for numbering are allocated to them respectively. For instance in 2D, as shown in Figure 2.15c, node number one has global DOF of 1 (g1) for its x-direction translation, global DOF of 2 (g2) for y-direction movement and global DOF number 3 (g3) for its rotational about z axis. Then node number 2 has global DOF 4 (g4) as its x-direction movement, global DOF 5 (g5) for y direction and global DOF 6 (g6) for its rotation and so on. This method of numbering does not apply to each
element individually, instead for all nodes, three numbers are reserved (in 2D and 6 in 3D). So totally, 12 spaces for global DOF should be allocated to the system. It should be noticed that the number of elements does not have any effect on DOFs and the DOF is only dependent to the number of nodes.

![Diagram](image1)

![Diagram](image2)

Figure 2.15 Degree of freedom for rod elements: a) in local coordinates; b) in global coordinates; c) global coordinates of a system with 4 elements.

It should be noted that in finite element method, rotation of nodes might be neglected as the DOF of a bar element (or any other element) can be expressed
through the movement and the relative position of the corresponding nodes. So, in
FEM, 2 and 3 DOFs suffice analysis of each node in 2D and 3D respectively.

### 2.2.11 Modulation of boundary conditions

In FEM, solving equations always requires boundary conditions to restrain some
nodes from movements or rotation (Dixit 2007). Here, a bar element is discussed to
clarify the need. On element basis, each equation is composed of following term

$$\frac{d}{dx}(EA \frac{du}{dx}) + F = 0$$  \hspace{1cm} (2.70)

and the solution demands at least one of these two conditions to be satisfied.

1) \(u\) is specified at both ends
2) \(u\) is specified at one of the ends and \(du/dx\) is specified at the other end

Without meeting one of these conditions, the equation is not solvable and is
singular.

To solve the above equation, the equilibrium equation of a bar element can be
written as

$$\frac{EA}{L} \begin{bmatrix} -1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} F(x_1) \\ F(x_2) \end{bmatrix}$$  \hspace{1cm} (2.71)

If this problem is solved using matrix multiplication, the result is

$$\frac{EA}{L} (-u_1 + u_2) = F(x_1)$$

$$\frac{EA}{L} (u_1 - u_2) = F(x_1)$$  \hspace{1cm} (2.72)

or

$$u_1 = u_2 - \frac{L}{EA} F(x_1)$$

$$u_1 = u_2 + \frac{L}{EA} F(x_2)$$  \hspace{1cm} (2.73)
which presents a set of two parallel lines without any crossing, thus it can be deduced that no answer exists for this equation, as shown in Figure 2.16.

![Figure 2.16 General solution of the equation $\frac{d}{dx}(EA \frac{du}{dx}) + F = 0$.](image)

It can be readily comprehended from Equation (2.73) that even if acting force diminishes, or $F(x_1) = F(x_2) = 0$, again $u_1 = u_2$. This means without any applied force, the element will slide and both ends of element will have exactly the same amount of displacement which is called rigid body motion. However, if any of the displacement variables in this equation are specified, the singularity stops. Physically, it is similar to bounding one end of the element with boundary conditions and restraining it from movement, then calculating the displacement in other end. If this happens and the $u_2$, for example, gets the value of zero, the equation becomes

$$u_1 = \frac{L}{EA} F(x_1)$$

$$u_1 = \frac{L}{EA} F(x_2)$$  \hspace{1cm} (2.74)

However, force acting on node 2 should be neglected because it has no effect on element and only the first equation in (2.74) is viable.
By curbing movement and rotation of some nodes, corresponding DOF of these nodes become zero and are exempt from being mapped in the final system because they are predefined and fixed. Therefore, only non-zero values need to be solved and volume of calculation is reduced.

\[ F(x) = u_1 u_2 = 0 \]

Figure 2.17 A rod element with boundary conditions (fixed end).

In mathematics and matrix algebra, finding force in the following equation

\[ \{f\} = [K_e]\{u\} \quad (2.75) \]

requires formation of the inverse matrix of the stiffness \([K_e]\) as shown

\[ \{u\} = [K_e]^{-1}\{f\} \quad (2.76) \]

In order to explain matrix singularity, a simple example is brought up. There is a general matrix A with size of (2x2) as shown

\[ A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \quad (2.77) \]

Then inverse of this matrix is

\[ A^{-1} = \frac{1}{\det(A)} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} = \frac{1}{(ad - bc)} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} \quad (2.78) \]

in which determinant of the matrix is in denominator of the coefficient and should not become zero. If the determinant of a matrix becomes zero then this matrix is non-invertible (Szabo and Lee 1969).

If the value of determinant in Equation (2.71) is sought, then it can be seen that it is zero thus the matrix is non-invertible as it was explained in physical terms.
2.2.12 Multi element assemblies

In previous sections, each lonely element was focused and constructed stiffness matrices for them, which was in form of

\[
[k_e] \{u\} = \{f\}
\]  

(2.79)

Figure 2.18 shows a grid comprised of four quadrilateral elements. Each element property is described by the Equation (2.79). In element 1 for example, the equation can be written

\[
\begin{bmatrix}
k_{11} & k_{12} & k_{14} & k_{15} \\
k_{21} & k_{22} & k_{24} & k_{25} \\
k_{41} & k_{42} & k_{44} & k_{45} \\
k_{51} & k_{52} & k_{54} & k_{55}
\end{bmatrix}
\begin{bmatrix}
u_1 \\
u_2 \\
u_4 \\
u_5
\end{bmatrix} =
\begin{bmatrix}
f_1 \\
f_2 \\
f_4 \\
f_5
\end{bmatrix}
\]  

(2.80)

If the stiffness matrix for each element is constructed, because of repetitiveness of some nodes in various elements, namely node 5 in Figure 2.18, the process becomes inefficient. Instead, by adopting strategy of viewing calculation on nodal basis, the process can be easily addressed (Smith and Griffiths 2004). Therefore, when a point is shared between two or more element, stiffness in that node is an augmentation of stiffness values of that node in each element. For instance, in the example provided, stiffness of node 2 is a summation of stiffness of node 2 in element 1 and 2. It should be noted that in this example, for simplicity, every point is assigned one variable, while there are 2 and 3 variables in 2D and 3D respectively. As a result, if assuming a true 2D version of this example, the stiffness matrix would be twice the length and twice the height because each node has 2 DOFs. Also, the displacement and force matrices would be twice the height regarding to displacement and force acting on each of two DOFs of each node. The outcome of the summation is presented in Table 2.3.
Figure 2.18 A mesh of element composed of 4 elements
Table 2.3 Formation of a stiffness matrix for elements depicted in Figure 2.18.

<table>
<thead>
<tr>
<th>$k_{11}$</th>
<th>$k_{12}$</th>
<th>0</th>
<th>$k_{14}$</th>
<th>$k_{15}$</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k_{21}$</td>
<td>$k_{12}^2+k_{22}^2$</td>
<td>$k_{23}$</td>
<td>$k_{24}$</td>
<td>$k_{12}^4+k_{22}^4$</td>
<td>$k_{25}$</td>
<td>$k_{26}$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>$k_{32}$</td>
<td>$k_{33}$</td>
<td>0</td>
<td>$k_{35}$</td>
<td>$k_{36}$</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>$k_{41}$</td>
<td>$k_{42}$</td>
<td>0</td>
<td>$k_{44}^2+k_{44}^2$</td>
<td>$k_{45}^2+k_{45}^2$</td>
<td>0</td>
<td>$k_{47}$</td>
<td>$k_{48}$</td>
<td>0</td>
</tr>
<tr>
<td>$k_{51}$</td>
<td>$k_{52}^2+k_{52}^2$</td>
<td>$k_{53}$</td>
<td>$k_{54}^2+k_{54}^2$</td>
<td>$k_{55}^2+k_{55}^2$</td>
<td>$k_{56}^2+k_{56}^2$</td>
<td>$k_{57}$</td>
<td>$k_{58}^2+k_{58}^2$</td>
<td>$k_{59}$</td>
</tr>
<tr>
<td>0</td>
<td>$k_{62}$</td>
<td>$k_{63}$</td>
<td>0</td>
<td>$k_{65}^2+k_{65}^2$</td>
<td>$k_{66}$</td>
<td>0</td>
<td>$k_{68}$</td>
<td>$k_{69}$</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$k_{74}$</td>
<td>$k_{75}$</td>
<td>0</td>
<td>$k_{77}$</td>
<td>$k_{78}$</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$k_{84}$</td>
<td>$k_{85}^2+k_{85}^2$</td>
<td>$k_{86}$</td>
<td>$k_{87}$</td>
<td>$k_{88}$</td>
<td>$k_{89}$</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$k_{95}$</td>
<td>$k_{96}$</td>
<td>0</td>
<td>$k_{98}$</td>
<td>$k_{99}$</td>
</tr>
</tbody>
</table>

### 2.3 Concluding remarks

In this chapter, a brief introduction of the finite element method and its performance is provided. Firstly, a simple bar is considered and equation to approach equilibrium (force and energy) in it, is sought. Then the stiffness matrix is explained and the method to obtain it is brought up. Then simple two dimensional rectangular and triangular elements and the correspondent stiffness matrix are mentioned. By addition of another dimension, the basic elements to satisfy the need to simulate object with volume is covered and the procedure to gain the stiffness matrix is elaborated.
However, simple elements cannot simulate all the shapes desirably. As an example, one may not occupy a circle with small rectangles and there will be spaces that will not be covered at the corners. Thus, the requirement for more general elements is sensed. By application of quadrilateral elements with different sizes of each of their edges, the constraint of their coverage is removed and they become more practical. In order to make general elements easier to be calculated, local coordination systems are deployed and their conversion system to the global coordination system is sought.

Complex shapes require numerous elements to simulate the original shape and bear reasonable results. Therefore, a system of addition for the stiffness matrix of each element is needed and one should consider how the global stiffness matrix is built up. Here, Degree Of Freedom (FOD) is introduced which places each component in the global stiffness matrix. Furthermore, as a system should be constrained to acquire static equilibrium, FODs are assigned to limit the displacement and thus strains in the system of elements and has been explained in this chapter.
Chapter 3

Topology optimization methods

3.1 Background
Topology optimization methods date back to one and a half centuries ago when Culmann (1866), a pioneer structural engineer, sought the best design of trusses to withstand predefined load (Prager 1974). Almost 40 years later, Michell (1904) published a paper about optimum shapes of a structure where fundamentals of optimization were established. Although these principles were credible, they did not attract major attentions before advent of computer. Prager (1969, 1974) and Rozvani (1972) expanded the theories and laid the principles of truss-like medium.

On the other hand, with broad application of computers, design optimization was invented. In this approach, instead of relocation of trusses and bars to obtain optimum, the initial continuum plate was perforated in a way that the final shape must have the desired properties (Cheng and Olhoff 1981; Kohn and Strang 1986).

Finally, it was 1988 when Bendsoe and Kikuchi (1988) proposed a practical method for topology optimization by using finite element and mathematical methods. In their approach, optimization problem was simplified to sizing features of microstructures in medium. Alternatively, Xie and Steven (1993) introduced a distinct and simpler method in which an evolutionary manner was deployed to transform the primary formation into optimized shape. By gradually eliminating inefficient parts, only efficient parts remain and the outcome possesses favourable specification. These methods fascinated numerous researchers and research on this subject got accelerated.

3.1.1 Topology optimization
Among the main procedures to specify optimized shape, material distribution has gained popularity in research. Here, a fixed domain is set to the material and some variables of material properties (usually modulus of elasticity) are modified. If the
method distinguishes a part as inefficient, modulus of elasticity of that part is reduced and it can be inferred that no (or less dense) material exists there (Bendsoe and Kikuchi 1988; Xie and Steven 1993). As explained, no re-meshing happens and by manipulation of modulus of elasticity, perforation is performed and topology of design is altered.

An option to solve this problem is using statistics and to find the chance of incidentally finding the answer. This method converges to an answer only for small problems, however for complex shape, where a medium is discretized to thousands of particles, it does not seem feasible (Rozvani 2001; Bendsoe and Sigmund 2003).

As a way to overcome this hurdle, **optimality criteria method** is used, which is an indirect approach. A criterion is set and all the elements are judged based on the criteria and those that do not satisfy the criteria undergo changes (Hassani and Hinton 1998c). This method was proposed by Karush-Kuhn-Tucker (KKT) optimality conditions which can be expressed as (Kuhn and Tucker 1951; Kaurush 1939)

\[
\begin{align*}
\min_{x \in \mathbb{R}^n} f(x) \\
g_i(x) &\leq 0, \quad i = 1,2,...,n \\
h_j(x) &\leq 0, \quad j = 1,2,...,m
\end{align*}
\]

in which \( f : \mathbb{R}^n \rightarrow \mathbb{R}, \ g_i : \mathbb{R}^n \rightarrow \mathbb{R} \) and \( h_i : \mathbb{R}^n \rightarrow \mathbb{R} \). It is mandatory in these equations that all functions are differentiable and gradient of variables are linearly independent. The KKT equations explain that with the existence of a local minimum in \( x \), an equation with constants \( \lambda_i \) and \( \xi_i \) is defined that

\[
\begin{align*}
\nabla f(x) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x) + \sum_{j=1}^{m} \xi_j \nabla h_j(x) &= 0; \\
h_j(x) &= 0, \quad \forall j = 1,2,...,m, \\
g_i(x) &\leq 0, \quad \lambda_i g_i(x) = 0, \quad \lambda_i \geq 0, \quad \forall i = 1,2,...,n
\end{align*}
\]
3.1.2 Minimum compliance layout

One of widespread objective functions for optimization is mean compliance which is equivalent to external work and can be written as

\[ E(u) = \int_\Gamma t u d\Gamma + \int_\Omega f u d\Omega \]  \hspace{1cm} (3.3)

where \( u \) represents displacement and \( t \) and \( f \) are surface traction and body forces respectively, \( \Omega \) stands for the whole domain and \( \Gamma \) is surface of the object. \( \Gamma_t \) and \( \Gamma_u \) also represent part of the surface where traction and boundary conditions are located. The mean compliance formed due to applied forces is presented in Figure 3.1.

![Figure 3.1 A typical optimization design domain.](image)

The internally stored energy because of an imaginary displacement \( v \) at the equilibrium \( u \) can be stated as

\[ a(u, v) = \int_\Omega K_{ijkl}(\lambda) \varepsilon_{ij}(u) \varepsilon_{kl}(v) d\Omega \]  \hspace{1cm} (3.4)

with \( K_{ijkl}(\lambda) \) as the stiffness tensor varying with location \( \lambda \) and \( \varepsilon \) representing strain.
By substituting mean compliance in KKT condition, one will have

$$\min_{x \in \chi} E(u)$$

(3.5)

$$a_{\chi} (u, v) = E(u), \quad \forall v \in \chi$$

$$x \in D$$

where $x$ defines field variables. $\chi$ represents space of mechanically acceptable displacement and $D$ represents acceptable design variables.

If the domain is discretized to a mesh of elements, the Equation (3.5) gets the form (Bendsoe and Sigmund 2003)

$$\min_{x,u} c(x) = f^T u$$

(3.6)

$$K(x)u = f,$$

$$x_i \in D, \quad \forall i$$

Here, $f$ and $u$ define nodal force and displacement. $c(x)$ denotes the mean compliance, $x$ as vector of design variables and $K$ as global stiffness matrix.

### 3.1.3 Sensitivity analysis

Sensitivity analysis assesses how changing variables in inputs change outputs and since the aim is deriving optimal conditions, sensitivity analysis can be implemented in optimization process (Bendsoe and Sigmund 2003). By using an auxiliary term, the mean compliance can be reshaped to

$$c(x) = f^T u - \bar{u}^T (K(x)u - f)$$

(3.7)

and by differentiation, the result would be

$$\frac{\partial c}{\partial x_i} = (f^T - \bar{u} K) \frac{\partial u}{\partial x_i} - \bar{u}^T \frac{\partial K}{\partial x_i} u$$

(3.8)
The auxiliary equation obtained for mean compliance is $f^T - \bar{u}K = 0$. This equation can become naive by imbedding $\bar{u} = u$ as

$$\frac{\partial c}{\partial x_i} = -\bar{u}^T \frac{\partial K_i}{\partial x_i} u$$

(3.9)

with $K_i$ and $u_i$ denoting local stiffness matrix and displacement vector for the i-th element. While the Equation (3.6) was proposed for the whole design domain, here, the sensitivity of the mean compliance is founded on each element individually.

Indirect technique of solving topology optimization can be divided to two main branches, namely continuous or binary and will be discussed in the following sections.

### 3.1.4 Continuous techniques to solve topology optimization problem

This approach was presented by Bendsoe and Kikuchi (1988) as the first step toward optimization. This method supposed the medium to possess staggering microstructural voids and dimensions of these voids are variables of the optimization. The two extreme extents of these voids can be imagined when the design variable has its maximum or minimum value representing a completely void element or completely solid material respectively.

![Figure 3.2 A General optimization problem with microstructures voids size as the design variable.](image)
As shown in Figure 3.2, size and shape of voids in microcells are parameters varying with location of the microcells and these variables are manipulated by the topology optimization program. Once the program identifies a location as a redundantly strong part, by increasing size of voids in that part, it tries to weaken that part. As the result, voids get enlarged and the amount of material used for each microstructure decreases and the microstructure loses its stiffness.

Among shortcomings of this method, one may notice its complexity in analysing effective homogenised nature of these microcells. Thus, to define properties of the microstructures there is a need for use of numerical technique such as FEM as or alternatively referring to available tables with required information about microstructures.

To keep the explanation concise and general, details about mathematical explanations of homogenisation method are omitted and researcher are recommended to refer to Bendsoe and Kikuchi (1988), Hassani and Hinton (1988a,b) and Eschenauer and Olhoff (2001) for further information.

For bypassing this intricacy, Bendsoe offered a direct approach in homogenisation method called SIMP (or Solid Isotropic Microstructures with Penalization) by Rozvany et al. (1992). Opposed to previous method where demand for predefined homogenisation properties existed, here, microstructures are substituted with a virtually synthetic material whose density varies. Actually, the material density changes in a continuous manner resulting in changes in the stiffness. Since material with faded density (with fake properties) exists in this approach, some researchers (e.g. Hassani and Hinton 1998c) called it as ‘artificial material model’.

### 3.1.4.1 Density scheme

As discussed in the proceeding section, changing material density leads to change in its stiffness. A mathematical model is needed to quantify this feature preferably in a continuous manner (Bendsoe and Sigmund 1999). Originally, Bendsoe (1989) utilised a quadratic interpolation method which is

\[ E_\lambda(\rho) = [\rho(\lambda)]^p \bar{E} \quad \lambda \in D \]  

(3.10)
with $E_{i}(\rho)$ as the interpolated stiffness tensor. The subscript $\lambda$ denotes its local value for the arbitrary location $\lambda$. $\rho(\lambda)$ represents relative density which $0 \leq \rho(\lambda) \leq 1$ and $D$ is admissible domain of design. The factor $p$ in the equation is penalisation parameter that biases middle values of $\rho$ toward maximum and minimum poles ($\rho = 0$ and $\rho = 1$ representing solid and empty elements respectively).

One should perpect that when $p >> 3$, the outcome resembles binary methods at the cost of numerical instabilities and convergence problems. Also when $\rho = 0$, the stiffness of some elements are omitted and the calculation might be exposed to singularity, thus a soft material $\rho_{\varepsilon}$ is applied instead of zero.

### 3.1.4.2 Deducing optimization criteria

The minimum compliance design problem using SIMP method can be expressed as

$$\min_{\rho, u} c(\rho) = f^T u$$

where $K(\rho)u = f$,

$$\rho_{\varepsilon} \leq \rho_i \leq 1, \quad i = 1,2,...,N$$

$$\sum_{i=1}^{N} \rho_i V_i - \bar{V} \leq 0$$

Here $N$ is number of elements, $V_i$ is the volume of $i$-th element and $\bar{V}$ is preferred final volume of topology. The affiliated Lagrangian function is

$$\zeta = f^T u - \tilde{u}^T (K u - f) + \sum_{i=1}^{N} \left( \alpha_i (\rho_i - 1) + \beta_i (\rho_{\varepsilon} - \rho_i) \right) + \Lambda \left( \sum_{i=1}^{N} \rho_i V_i - \bar{V} \right)$$

with $\tilde{u}$ as the vector of Lagrangian multipliers for equilibrium condition. $\alpha$ and $\beta$ are Lagrangian multipliers for upper and lower limit conditions and $\Lambda$ is Lagrangian multiplier for volume constraints. Stationarity of the Lagrangian with respect to $\rho_i$ conveys
\[
\frac{\partial c}{\partial \rho_i} + \alpha_i - \beta_i + V_i \Lambda = 0 \quad \forall i = 1, 2, \ldots, n \quad (3.13)
\]

By proposing a parameter \( \Psi_i \) as

\[
\Psi_i = \frac{-\left(\frac{\partial c}{\partial \rho_i}\right)}{V_i \Lambda} \quad (3.14)
\]

the following altering scheme can be deduced

\[
\rho_i^{K+1} = \begin{cases} 
\max\{(1 - \zeta)\rho_i^K, \rho_e\} & \text{if } \rho_i^K(\Psi_i^K)^\eta \leq \max\{(1 - \zeta)\rho_i^K, \rho_e\} \\
\min\{(1 + \zeta)\rho_i^K, \rho_e\} & \text{if } \rho_i^K(\Psi_i^K)^\eta \geq \min\{(1 + \zeta)\rho_i^K, \rho_e\} \\
\rho_i^K(\Psi_i^K)^\eta & \text{otherwise}
\end{cases} \quad (3.15)
\]

By using Equation (3.13) in (3.15), the partial derivative of mean compliance with respect to \( \rho \) is

\[
\frac{\partial c}{\partial \rho_i} = -p \rho_i^{p-1} u_i^T K_i u_i, \quad i = 1, 2, \ldots, N \quad (3.16)
\]

Also Lagrangian multiplier \( \Lambda \) should be calculated in each iteration.

The following flowchart concludes the procedure of SIMP. The results of optimization using SIMP method is depicted in Figure 3.3 (Sigmund 2001).
Figure 3.3 Flowchart of the SIMP method.

Figure 3.4 Application of SIMP to solve a Short Cantilever Beam problem and the objective function value in each iteration (after Ghabraie 2009).
### 3.1.5 Binary techniques to solve topology optimization problems

Xie and Steven (1993) proposed Evolutionary Structural Optimization (ESO) method in which inefficient elements are gradually eliminated and the structure is reformed to an optimal shape. After loading elements, physical reaction of them are ranked and assessed with rejection criterion and those whose values are lower than the criterion, are not marked as efficient and get eliminated. This method is commonly cited as Hard-kill method because of the nature of complete removal of inefficient elements in it, while previous programs (which are called Soft-kill) replaced elements with soft material in any inefficient parts.

Originally, von Mises stress was chosen as scale of performance in each element (Xie and Steven 1993) and element whose von Mises stress did not exceed a predefined percentage of maximum stress were omitted. It is interesting that redistribution of stress caused by the element removal, adds further pressure to the remaining elements and the rejection ratio (RR) should be updated and lifted in each iteration of the process. In each evolution, FE model, assessment, removal and updating RR happens sequentially till a termination condition (including the minimum volume of optimized shape or maximum RR or etc.) is met.

Although the manner seams intuitively inferring structure with full stress design, but factually, it lacks mathematical basis and support. Hence, Chu et al. (1996) replaced von Mises stress with compliance of structure and introduced sensitivity number as a scale for measuring changes in efficiency of structure due to elimination of each element. The objective in this procedure is minimising the compliance as the consequence of manipulating the sensitivity numbers. Imagine the i-th element is eliminated from a structure with N elements and stiffness K. As the result of elimination, K will alter to $K^{-i}$. Mutation of stiffness matrix takes the form of $\Delta K = K^{-i} - K = \tilde{K}_i$ with $\tilde{K}_i$ denoting stiffness matrix of the element when regarded globally. Force vector f remains constant during the evolution, therefore differentiation of the compliance matrix can be stated as $\Delta c = f^T \Delta u$ which described as sensitivity number of i-th element, $\alpha_i$. By considering change in displacement as

$$\Delta u = -K^{-1}\Delta Ku$$  \hspace{1cm} (3.17)
one can write sensitivity number for the i-th element as

\[ \alpha^i = u_i^T K_i u_i \]  

where \( u_i \) and \( K_i \) are displacement vector and stiffness of i-th element. The sensitivity number has relevance with efficiency, meaning that lowest sensitivity numbers belong to the least efficient elements.

The problem of ESO is that once an element is identified as insufficient and is omitted from the scene, because of hard-kill essence of ESO, it cannot come back even the lowest possible compliance option has not been gained.

This defect was treated when an extended version of ESO was proposed by Querin (1997), Querin et al. (1998) and Yang et al. (1999). The modified method not only removes inefficient elements, but also fortifies weak part by adhering extra material to them when needed. This bi-directional algorithm resembling ‘trial and error’ was given the name ‘Bi-directional Evolutionary Structural Optimization’ or BESO.

In contrast to ESO, the BESO should keep records of all the elements even after elimination of the inefficient element from domain design. When the program decides to revive the removed element, the BESO turns on that element according to already saved geometrical information.

The minimum compliance problem can be defined as

\[
\min_{x,u} c(x) = f^T u
\]  

where \( K(x) u = f \),

\[ x_i \in \{0,1\}, \quad i = 1,2, ..., N \]

\[
\sum_{i=1}^{N} x_i V_i \leq \bar{V}
\]

in which \( V_i \) expresses volume of the i-th element; \( \bar{V} \) as the desired volume of final topology and N is the number of elements. The parameter \( x_i \) might get the value of either 0 or 1 representing void or solid respectively.
In BESO, when the algorithm finds a part mechanically insufficient, it urges to attaches extra material to vicinity of that part. Hence, in the vicinity, elements with higher sensitivity number are prioritised to be reinforced. However, surrounding void elements have the sensitivity number of zero because they were excluded from FEM analysis. As one of the numerous ways to overcome this problem, Huang and Xie (2007) applied a linear extrapolation method that assigns the sensitivity number to the surrounding voids. This linear extrapolation filtering is defined as

\[
\hat{a}_i = \frac{\sum_{j=1}^{N} w_{ij} a_j}{\sum_{j=1}^{N} w_{ij}} \tag{3.21}
\]

where \( w_{ij} \) is weighting factor described as

\[
w_{ij} = \max\{0, r_f - r_{ij}\} \tag{3.22}
\]

with \( r_f \) and \( r_{ij} \) remarking filtering radius and the distance between centre of \( i \) and \( j \) elements as shown in Figure 3.5. The mentioned figure also shows a green shading to display the weighting value which becomes lesser when distance from the centre of the circle increases.

Figure 3.5 Filtering scheme showing elements located within the distance of filtering radius \( r_f \) from the element \( i \).
One of the other merits of filtering sensitivity is preventing formation of checkerboard instability (Diaz and Sigmund 1995; Jog and Haber 1996). When filtering is not exploited in BESO method, due to mixed FEM formulations, solid and void are arranged intermittently, resembling checkerboard-like patterns. However, when filtering technique is employed in the process, these irregularities are faded and smoothed propelling a gentle transition from hard material to voids (Sigmund 1994).

Attaching and detaching mechanism in the new BESO is based on defining volume fraction in each iteration and gradually increasing the volume of elements needed to be removed. This method, proposed by Huang et al. (2006) and Huang and Xie (2007) enjoys an explicit parameter named Evolutionary Volume Ratio (EVR) which determines the volume of next iteration by

\[
V_{k+1} = V_k(1 + \text{sign}(\bar{V} - V^k)\text{EVR})
\]  

(3.23)

in which \(\bar{V}\) expresses desired final volume, \(V_k\) and \(V_{k+1}\) are the volume of topology at k and k+1 iterations respectively.

It is clear that the BESO does not impose restrictions on volume of addition or removal separately, instead, total volume of elements to be added or removed are counted. To clarify the procedure, explanation will be given through following steps (note that here, elements volume are assumed to be equal) (Huang and Xie 2007)

1. Sort sensitivity number of all the elements in ascending order
2. Find sensitivity number of an element whose value equals to desired volume \(V_{k+1}\) and set it as marginal value \(\alpha_{th}\)
3. Find Admission Ratio as

\[
AR = \frac{V_{add}}{V}
\]

(3.24)

where \(V\) is total volume and \(V_{add}\) is the volume of void elements with \(\alpha\) greater than \(\alpha_{th}\)
4. If \(AR \geq AR_{max}\), then calculate the \(V_{add}\) as \(V_{add} = AR_{max} \times V\)
5. Sort sensitivity of void elements and find sensitivity number of an element whose value equals to volume \(V_{add}\) and set it as \(\alpha_{th-add}\)
6. Find the removal volume threshold as $V_{del} = V_{add} + V_{k+1} - V_k$ and find sensitivity number of an element whose value equals to volume $V_{del}$ and set it as $\alpha_{th-del}$

7. Void elements with the sensitivity number greater than $\alpha_{th-add}$ should be added (turned on) and solid elements with sensitivity lower than $\alpha_{th-del}$ should be eliminated (turned off).

The flowchart below concludes the BESO algorithm.

![Flowchart presenting BESO algorithm.](image)

Figure 3.6 Flowchart presenting BESO algorithm.
3.1.6 Summary
This section has provided a brief review of history of topology optimization and brought forward four methods of optimization including the homogenisation, the SIMP, the ESO and the BESO. Each method has its own advantages and disadvantages and when using them, one should understand their capabilities and choose them according to the current needs. Here, negative and positive factors of each method will be briefly explained and finally by expressing the specification for geotechnical optimization, the appropriate method should be chosen accordingly.

Beginning with homogenisation technique, existence of microstructures requires numerous variables producing complexity for modellings. Also connecting the homogenisation method with external FEM programs needs sophisticated programming.

The SIMP technique, compared with homogenisation technique, enjoys substantially less variable and complexity making this technique comparably naive for coding and linking to external FEM programs. On the other hand, presence of blurred edges in final results is unfavourable in designing physical shapes.

The ESO technique, in contrast to two aforementioned techniques, has a binary outcome, meaning no microstructures dominate in the design nor blurred or faded boundary. Coding and linkage to external FEM is significantly easy and the user has both options of using intuitive or mathematically derived optimization criteria to adopt. The major drawback of this technique is its one directional approach toward optimum. In other words, if for any reason (like choosing high ER) it makes an inappropriate change, there is no way to retrieve that undesired step.

Finally the BESO technique compensates the unrecoverable changes with the bi-directional performance making it robust to changes in the optimization parameters. Similar to ESO, this method supplies the user with clear edges (because of its binary state) and also adopting either intuitive or rigorous optimization criteria. Having a mathematical support, possessing filtering technique to prevail numerical instabilities and having the capability of using soft-kill to overcome unstable solutions are among important advantages of this method.
Regarding the underground design domain, as the final goal for the optimization, an optimization technique, which can relatively be easy to be implemented in codes and linked to external FEM and also does not want a post-processing scheme to convert the blurred edges to sharp margins, should be chosen. Considering these properties, homogenisation and SIMP techniques are excluded, leaving ESO and BESO more suitable to be mentioned in this thesis.

Speed of BESO in convergence surpasses the ESO and results are more reliable because of its recovering ability. In contrast, BESO’s bi-directionality does not match non-linear material usage as they are history dependent. Since, here, the modelling is limited to elastic material, only BESO’s application will be discussed and then it will be expanded for non-linear material. In the next chapter application of BESO will be discussed and then step by step, by specifying the needs for modelling underground excavation, some minor modification will be introduced.

3.2 Application of Bi-directional Evolutionary Structural Optimization (BESO)

3.2.1 Introduction
In this section, application of BESO will be presented. Also, each component of the process will be explained individually. Sensitivity number derivation, effect of filtering scheme, mathematical support for BESO, application of multi-material and interpolation needed for multi-material are described.

3.2.2 Sensitivity number derivation
Changes in the topology of a shape should be stated through switching its element from void to solid or vice versa. BESO’s mathematical core is minimization of the objective function and this function considers mean compliance of elements. Assessing changes of mean compliance in whole the system is dependent of compliance in each element and thus effect of change in each element should be evaluated by a mean called sensitivity. Sensitivity analysis should be deployed to measure how a change in the i-th element alters the mean compliance.
As mentioned, objective function $f$ is a function of each element, $f = f(x)$ and the goal is to minimize it. Changes in each element can be stated as either of two following options, when a void element changes to solid or when a solid element changes to void.

If $i$-th element is switched to void state, the variable $x$ is changed from 1 to 0 and tensor of design variable is converted from $x$ to $x^{-i}$. According to Taylor series, when applying first order approximation, the function $f$ can be expanded to (Ghabraie 2009)

$$f(x) = f(x^{-i}) + \frac{\partial f}{\partial x_i} (x_i - x_i^{-i})$$

(3.25)

or by reordering it, one can write

$$\Delta f_{-i} = f(x^{-i}) - f(x) = -\frac{\partial f}{\partial x_i}$$

(3.26)

where $\Delta f_{-i}$ is the change in the objective function due to the shift in the $i$-th element and is therefore sensitivity number of the $i$-th element.

Alternatively, if an element changes from void state to solid, the design variable $x$ changes to $x^{+i}$ with change in the value $x_i = 0$ to $x_i^{+i} = 1$ for the $i$-th element. Similarly, after application of first-order Taylor series and then reordering it, the sensitivity can be written as

$$\Delta f_{+i} = f(x^{+i}) - f(x) = \frac{\partial f}{\partial x_i}$$

(3.27)

Comprehensively, the change in the objection function can be expressed as

$$\Delta f = \sum_{i=1}^{n} \frac{\partial f}{\partial x_i} - \sum_{j=1}^{m} \frac{\partial f}{\partial x_j}$$

(3.28)

in which first term $\left(\sum_{i=1}^{n} \frac{\partial f}{\partial x_i}\right)$ is allocated for the elements which needs to be strengthened (by substituting void with solid) and the second term $\left(\sum_{j=1}^{m} \frac{\partial f}{\partial x_j}\right)$ is for elements needed to be weakened (by deleting solid to create void instead). Here, $n$
is the number of elements requiring to be strengthened and \( m \) counts void elements requiring to be subsided.

It can be concluded that elements with highest sensitivity numbers are prone to be strengthened (as the most efficient elements) and elements with lowest sensitivity number are more apt to be weakened (because of their low efficiency).

According to what has been stated, sensitivity in each element can be proposed as

\[
\alpha_i = -\frac{\partial f}{\partial x_i}, \quad i = 1, 2, ..., n
\]  

(3.29)

And by substituting the compliance in the objective function, the compliance sensitivity of each element can be achieved as

\[
\alpha_i = u_i^T \frac{\partial K_i}{\partial x_i} u_i, \quad i = 1, 2, ..., n
\]  

(3.30)

It means that sensitivity of each element is a function of displacement in each element and difference of the element stiffness when it changes from element to void or vice versa. The change in stiffness value needs elaboration which is given in the next section.

### 3.2.3 Material interpolation scheme

Material interpolation methods are used in BESO to identify the changes of stiffness upon switching elements and can be divided to two groups, linear and non-linear.

Linear material interpolation is the simplest model of interpolation and can be written as (Ghabraie 2009)

\[
E_i(x_i) = E_1 + x_i(E_2 - E_1), \quad i = 1, 2, ..., n
\]  

(3.31)

where \( E_i \) is the interpolated stiffness of the i-th element, \( E_1 \) and \( E_2 \) are stiffness of two constant materials. Also it is supposed that \( E_2 \) is the stiffness of stronger material and \( E_1 \) is the stiffness of softer material. \( x_i \) is the design variable and \( n \) represents number of elements. It should be noted that when isotropic material is
considered, material stiffness is dependent on two variables, namely Poisson’s ratio \( \nu \) and Young’s modulus of elasticity \( E \).

Supposing the materials to be in linear elastic domain, one can derive level stiffness matrix \( K_i \) of the i-th element as

\[
K_i(x_i) = \frac{E_i(x_i)}{E_i} \bar{K}_i
\]

(3.32)

where \( E \) is elasticity of the original material and \( \bar{K}_i \) is the stiffness matrix of the element when original material comprises that element. Finding the changes of this function due to changes of design variable requires differentiation of level stiffness matrix with respect to \( x_i \), resulting in

\[
\frac{\partial K_i}{\partial x_i} = \frac{\partial E_i}{\partial x_i} \bar{K}_i = \frac{\partial E_i}{\partial x_i} \bar{K}_i
\]

(3.33)

and by using Equation (3.31), one can write

\[
\frac{\partial K_i}{\partial x_i} = \frac{\bar{K}_i}{E_i}(E_2 - E_1)
\]

(3.34)

Here, if Equation (3.30) is substituted in Equation (3.34), sensitivity number of minimization problem in compliance can be expressed as

\[
\alpha_i = \frac{(E_2 - E_1)}{E_{orig}} u_i^T K_i u_i
\]

(3.35)

where \( E_{orig} \) might have either value of \( E_2 \) or \( E_1 \) depending on the original state of element. If the element was weak before the switching, \( E_1 \) should be applied otherwise \( E_2 \) should be used. Here, the coefficient \( \frac{(E_2 - E_1)}{E_{orig}} \) moderates the bias between compliance in void and solid elements in a way that without it, compliance in solid elements are all greater than their counterparts in void elements. Based on the assumptions of \( E_2 > E_1 \), the value of the coefficient \( \frac{(E_2 - E_1)}{E_{orig}} \) is increased when material is weak and \( \frac{(E_2 - E_1)}{E_{orig}} \) is decreased when material is strong.
However, this mechanism gets embroiled when filtering is applied. When filtering gets involved in the calculations, the surrounding elements affect each other and their values are merged assigning higher values to void elements around solid ones compared to void elements with void neighbours. This phenomenon also happens oppositely when solid elements are at the edges and they get devaluated because of the presence of voids around them.

Therefore, linear interpolation is only proper for optimization without filtering otherwise application of non-linear interpolation schemes is recommended.

### 3.2.4 Non-linear interpolation

There have been many non-linear methods of interpolation from which one of the most common schemes, namely power-law interpolation will be discussed.

SIMP method enjoys a power-law interpolation scheme and this idea was brought to BESO by Huang and Xie (2009) to overcome interference of filtering and dual reaction of linear interpolation. This scheme has the form of

\[ E_i(x_i) = x_i^p \overline{E}, \quad i = 1, 2, ..., n \]  

(3.36)

in which \( p \) stands for penalty factor with the value greater than or equal to 1 and \( \overline{E} \) is the stiffness of the base material.

Again, assuming that material is isotropic and the Poisson’s ratio remains constant during switching elements, one can write

\[ E_i(x_i) = E_1 + x_i^p(E_2 - E_1), \quad i = 1, 2, ..., n \]  

(3.37)

Finding the changes of this function due to changes of design variable requires differentiation of level stiffness matrix with respect to \( x_i \), resulting in

\[ \frac{\partial K_i}{\partial x_i} = px_i^{p-1}(E_2 - E_1) \frac{\overline{K_i}}{E_i} \]  

(3.38)

Similarly, by using (3.35) in (3.38), the sensitivity number for compliance minimization problem can be stated as
\[ \alpha_i = px_i^{p-1} \frac{(E_2 - E_1)}{E_{\text{orig}}} u_i^T K_i u_i \] (3.39)

in which \( E_{\text{orig}} \) might have either value of \( E_2 \) or \( E_1 \) depending on the original state of element. Noticing the binary nature of BESO with design variable values of either 1 or 0, as long as \( p > 1 \), the equation can be written as

\[ \alpha_i = \begin{cases} 
  p \left( \frac{E_2 - E_1}{E_2} \right) u_i^T K_i u_i & \text{if } x_i = 1 \\
  0 & \text{if } x_i = 0 
\end{cases} \] (3.40)

It should be noted that this equation completely neglects sensitivity number of weak elements, and thus as a treatment for this malfunction, application of soft-kill BESO is recommended in which design variable for weak material is assumed to be a small positive value \( x_{\text{min}} > 0 \). After implementation of this assumption, the Equation (3.39) can be redefined as

\[ \alpha_i = \begin{cases} 
  p \left( \frac{E_2 - E_1}{E_2} \right) u_i^T K_i u_i & \text{if } x_i = 1 \\
  p \frac{x_i^{p-1}(E_2 - E_1)}{E_1 + x_i^p(E_2 - E_1)} u_i^T K_i u_i & \text{if } x_i = x_{\text{min}} 
\end{cases} \] (3.41)

Throughout this thesis, this equation would be used.

### 3.2.5 Mathematical formulation for BESO

One may formulate the compliance minimization problem as

\[ \min_{x,u} c(x) = f^T u \quad (3.42) \]

where \( K(x)u = f \),

\[ E_i = E_i(x_i), \]

\[ x_i \in \{0,1\}, \quad i = 1,2, \ldots, N \]

\[ \sum_{i=1}^{N} x_i V_i = \bar{V} \]
Here the $\bar{V}$ is the desired volume of solid material and $E_i(x_i)$ is the material interpolation layout. Note that because of the volume restriction, this problem is classified as constrained optimization problem. If the volume constraint is dismissed, the ‘relaxed version’ of this formulation is produced and has more generality leading to altering of the whole design domain.

The whole process of optimization can be viewed as two phases, one updating the design in the relax version and then exerting constraint on it.

The first phase comprises discovering gradient of the objective function to find the optimum by applying saddle-point method. If $f$ stands for objective function, then this gradient would be stated as

$$ g = -\nabla f = -\sum_{i=1}^{n} \frac{\partial f}{\partial x_i} = -\sum_{i=1}^{n} u_i^T \frac{\partial K_i}{\partial x_i} u_i $$ \hspace{1cm} (3.43)

The second phase consists of modification factor to satisfy the volume constraints. The modification factor can be stated as

$$ \hat{d}_K = x_{K+1} - x_K $$ \hspace{1cm} (3.44)

where subscripts represent the number of iterations. Since in each element the design variable has two values, or $x_i \in \{0,1\}$, then the modification factor will be assigned -1,0 or 1, corresponding to removing, steady-state or addition respectively.

Assuming the move limit (move limit is the number of changes in design variable during each iteration of optimization) is chosen as $m = 1$, then the relevance between $\hat{d}$ and $g$ gets the form of (Ghabraie 2009)

$$ \hat{d}_i = \text{sign}[(1 - x_i)g_i - \max\{(1 - x_i)d_i\}] + $$

$$ \text{sign}[x_i g_i - \min\{x_i d_i\}], \hspace{0.5cm} i = 1,2,\ldots,n $$ \hspace{1cm} (3.45)

Higher order of approximation requires sophisticated mathematical algorithms, thus for the sake of brevity, they are neglected, but interested readers should refer to (Hertskovits 1995).
3.2.6 Elements substitutions
Considering the compliance minimization problem and sensitivity numbers discussed in (3.2.2), the alteration of objective function can be rewritten as (Ghabraie 2009)

$$\Delta c = \alpha_s - \alpha_w$$  \hspace{1cm} (3.46)

Where $\alpha_s$ and $\alpha_w$ are the sensitivity of strong and weak element respectively. According to this equation, when weak elements with the highest sensitivity numbers or strong elements with the lowest sensitivity numbers are switched, the objective function dives to the minimum.

Hindering sudden changes in the design requires setting a maximum for switching of elements and this predefined value is called move limit. Setting a large move limit (i.e. too many elements be allowed to be switched) usually reduces the accuracy of the optimization but enhance the speed. On the contrary, with setting a small move limit chance of approaching the optimum is higher at the cost of slow progress.

The method also should feature a criterion to identify when the progress has converged and also when the same changes are happening in two successive iterations. Furthermore, a maximum number of iteration should be defined in order to curb limitless loops.

3.2.7 Shape optimization
Having sharp and clear boundaries makes binary optimization methods, namely BESO and ESO, useful when shape optimization is the purpose. During the shape optimization, boundaries between material phases get altered in each iterations. Thus an algorithm to identify elements on the edges and the corners is essential.

Boundary elements can be categorized into two classes, each of which requires different methodology to identify. These classes are elements on the edges of domain design and those around voids.

When design domain is discretized by a mesh, some elements undoubtedly get situated at the corners and on edges of the shape. One option to recognise these
elements is counting elements surrounding each of these elements through application of a filtering radius. By assuming a uniform mesh to be applied, when number of neighbouring elements is below a logical and predefined value, it is implied that the element is not located in middle of other elements. As an example, Figure 3.7 illustrates a quadrilateral element which has 8 elements around it located in the filtering radius and when the number of elements surrounding is less than 8, the element can be included to be located at the corner or on the edge of the shape.

![Figure 3.7 A schematic presentation of a procedure for finding elements located on the edges of the shape.](image)

However, the problem with this scheme is that when the applied mesh is not uniform, the counting of elements within the filtering zone results in different numbers and thus this method is not reliable any more.

Another option is finding edges (for 2D and surfaces for 3D elements) that are not shared between elements. When an edge is repeated in formation of two elements, it is perceived that two elements are sharing the same wall between each other. On the other hand, when an edge is uniquely used in formation of one and only one element, that edge is not a wall in between of two elements and that edge is located on the boundary of mesh of elements, thus the element containing the edge is a boundary element.
3.2.8 Concluding remarks
Optimizations of topology and also rock reinforcement are two main steps in underground excavation design optimization, which after being tailored would be capable of resolving these kinds of problems. Since the homogenization and the SIMP method do not produce definable shape, they are not employed here and focus is redirected to BESO because of its rapidity and more chance of finding optimum.

To find changes of stiffness due to alteration of design variable, material interpolation schemes are introduced and also the process of obtaining sensitivity number for minimization of mean compliance is discussed and then procedure of switching element is discussed. Some mathematical support for the optimization is also specified in which gradient and saddle point solution is deployed. Also as optimizing of the shape of an opening requires identification of boundary elements, through a brief explanation, appropriate method to cover this issue is proposed.
Chapter 4

Underground excavation design

4.1 Introduction

Unlike constructed materials with predefined properties, underground environments have unknown behaviour. This uncertainty can be attributed to different sequence of loading and also various behaviours and proportion of composing material, namely rock pieces, geological masses and water. The major factors governing the material responses which are stress distribution and geological features in rock masses are studied in this chapter.

4.2 Stress distribution in rocks

Stress in rock masses is the main concern in excavation designs and engineers should design excavation in such way that these stresses do not lead to stability failures. Regarding their origin, stresses in rocks can be classified into in-situ and induced stresses.

4.2.1 In-situ stresses

In-situ stresses which are the stresses found prior to excavation include gravitation stresses, tectonic stresses, residual stresses and thermal stresses of which only the first one will be discussed (Hegret 1988).

Gravitation stresses result from the weight of overlying column of substances and the vertical component of this pressure can be safely approximated as (Terzaghi et al. 1996; Hoek and Brown 1980)

$$\sigma_v = \int_0^z \rho g \, dz$$

(4.1)
where \( \rho \), \( g \) and \( z \) are density of the overlying materials, gravitational acceleration and depth respectively. Common rocks have the density value around 2650 kg/m\(^3\). Although the vertical component was readily estimated, finding the horizontal component seems more complicated due to effect of rock mass properties and different boundary conditions.

The ratio of horizontal stress to vertical stress is defined by the ratio \( k \) in a way that (Terzaghi et al. 1996; Craig 2004)

\[
\sigma_h = k\sigma_v
\]

(4.2)

Generally, for rocks, the value of \( k \) for the in-situ stress should satisfy (Hoek and Brown 1980)

\[
\frac{100}{z} + 0.3 < k < \frac{1500}{z} + 0.5
\]

(4.3)

It is interesting that the ratio in the rocks is only dependent on the depth \( z \), not material properties. Also, for soils, the ratio depends on friction angle and consolidation condition and can be written as (Jaky 1944; Mayne and Kulhawy 1982)

\[
k = \begin{cases} 
1 - \sin \phi' & \text{for consolidated soils} \\
(1 - \sin \phi')(OCR)^{\sin \phi'} & \text{for overconsolidated soils}
\end{cases}
\]

(4.4)

where \( \phi' \) is the effective angle of friction for soils, \( OCR \) is the overconsolidation ratio. The effective friction angle \( \phi' \) normally has the value of 18\(^\circ\) to 43\(^\circ\) (Terzaghi et al. 1996).

### 4.2.2 Induced stresses

Induced stresses are caused by disturbance of in-situ stresses due to excavation activity and generally happen in the surrounding area around the opening of excavation. After the excavation, stress distribution around the opening is different from the stress that existed before digging process because when material is excavated from an underground environment, the burden it was carrying should be transited to the rock left standing. And it means that the extra pressure exerted to
the neighbouring parts causes stress redistribution. Specification of induced stresses is therefore significant and can be carried out with application of one of the three following ways, which are field tests, analytical methods and numerical simulations (Hegret 1988).

Various field tests are available to assess the ground stress in rocks. These methods enjoy application of measurement facilities (including over-coring, flat-jack, hydraulic fracturing, etc.) and are mainly based on elastic responses of isotropic rocks or soil (Hegret 1988).

Analytical methods determine the pressure around opening using mathematical formulation and assuming that elastic or elasto-plastic materials are used (Yu 2000). However, many underground shapes are too intricate to be analysed using closed form solutions.

For general cases, induced stresses are analysed using numerical simulation with the computer assistance. These methods are sorted into two groups: continuous methods and discontinuous methods. The former one including Finite Element Method (FEM), Finite Difference Method (FDM) and Boundary Element Method (BEM) discretizes a continuous model to elements and analyses their reaction to the applied force. This group of simulation are appropriate for homogenous continuum where discontinuities do not exist or discontinuities effect can be considered as isotropic behaviour. Performance of this group has been extended to rock masses with few joints and discontinuities provided the responds from each of these discontinuities have been considered in calculation.

The latter group including Discrete Element Method (DEM) and Discrete Fracture Network (DFN) discretizes the domain to small particles and the effect of motions of large number of particles are analysed (Jing and Hudson 2002; Jing 2003). To illustrate the idea better, it can be imagined that the model is considered as sand particles and then reactions of forces on the surfaces of sand grains and their motions are studied. This group has been recently become popular for geotechnical analysis and is more suitable for simulation of rock masses with modest number of discontinuities.
### 4.3 Mechanical behaviour of rocks

Discontinuities and joints are one of the main characteristics of rock masses and neglecting them is oversimplification of the true behaviour of rocks leading to severe stability problems in most cases. The ability of rock to withstand tensions perpendicular to joints or discontinuities plain is so smaller than contact rock that it is mostly neglected and assumed to be zero (Hoek et al. 1997). This zero tensile strength commonly associated with low shear strength (compared to intact rock) imposes anisotropic properties to the rock masses. Thus, direction, location, orientation, length, spacing and mechanical properties of these discontinuities are of great emphasis for civil and mining engineers (Hegret 1988).

To address mechanical properties of rock masses, firstly intact rock is studied.

#### 4.3.1 Intact rock

Similar to many other materials, when exposed to excessive pressure, rocks undergo development of fractures and finally collapse (Hoek et al. 1997). Determining the mechanisms of rock failure has been one of the major concerns for researchers and different theories have been offered of which some classify rocks as a brittle material and other estimate the behaviour to be proportionally ductile. Here two important models for failure are reviewed.

#### 4.3.2 Mohr-Coulomb criteria

In 1773, Coulomb suggested that shear stresses tend to cause failure. To maintain the material in its elastic zone, the shear should be resisted by cohesion of the material and also a material constant which relates normal stress to the shear. Therefore, the shear resistance is formulated as

\[ \tau = c + \sigma_n \mu \]  

(4.5)

where \( c \) is the cohesion (or inherent shear strength); \( \sigma_n \) is the normal stress and \( \mu = \tan(\varphi) \) with \( \varphi \) as the internal angle of friction.

In 1900 Mohr proposed Mohr circle which was a transformation formula to calculate stresses acting on rotated coordinate systems and it can be expressed as...
\[
\sigma_n = \frac{1}{2}(\sigma_1 + \sigma_2) + \frac{1}{2}(\sigma_1 - \sigma_2)\cos 2\theta \tag{4.6}
\]

\[
\tau = -\frac{1}{2}(\sigma_1 - \sigma_2)\sin 2\theta
\]

Here, \(\sigma_1\) and \(\sigma_2\) are the principal stresses such that \(\sigma_1 > \sigma_2\); \(\tau\) is the shear on the rotated surface and \(\theta\) is the angle between \(\sigma_1\) and rotated coordinate. By adding the Mohr’s circle to the Coulomb’s criteria, a more general failure criteria was developed. Here, when one of the Mohr’s circles touches the failure envelop, the material experiences plastic deformation. The Mohr-Coulomb failure envelop is depicted in Figure 3.1.

![Mohr-Coulomb failure envelop](image)

**Figure 4.1** Mohr-Coulomb failure envelop.

### 4.3.3 Hoek and Brown criteria

Based on numerous field tests, Hoek and Brown (1980) showed that relation between shear strength and applied stress is not linear. So they offered a comprehensive empirical failure mechanism as

\[
\sigma'_1 = \sigma'_3 + \sigma_c \left( m_i \frac{\sigma'_3}{\sigma_c} + 1 \right)^{0.5} \tag{4.7}
\]
in which $\sigma_1'$ and $\sigma_3'$ are the major and minor effective stresses respectively. The uniaxial compressive strength (UCS) is denoted by $\sigma_c$ and $m_i$ is rock properties factor and is specified in rock laboratories (Hoek et al. 1997).

Figure 4.2 Hoek-Brown failure envelop compared with Mohr-Coulomb failure envelop and Mohr’s circles located under the yield line (after Eberhardt 2012).

Uniaxial compression stress $\sigma_c$ is one of the most basic parameters of the rock and can be by applying stress on a rock specimen with standard properties. This factor is measured when a standard rock piece is subjected to only vertical load and other sides of the specimen are free (Hoek and Brown 1982). Also other indirect methods for determining UCS exists which are beyond scope of this thesis.

When discontinuities get involved, the behaviour is altered and strength along it follows the Mohr-Coulomb criterion, but here cohesion can be neglected (Hudson and Harrison 1997), thus the shear strength gets simplified to $\tau = \sigma_n \tan \varphi$.

Rock masses containing rock pieces, discontinuities and water react to loading in a complicated manner. This condition combined with changes of properties due to scale, makes the process even more dire. To address these problems, an engineer requires numerical specification of the discontinuity surfaces to analyse their effect.
in distribution of stresses from which mechanical responds were discussed. Moreover, geometrical properties of discontinuities significantly affect the rock mass.

To predict general behaviour of rock masses, Terzaghi (1964) published a system of rock classification and the corresponding steel support required to sustain the system. This classification divides rocks into intact, stratified, moderately jointed, blocky, crushed but contact by chemical bonds, squeezing or swelling.

Deere et al. (1967) suggested a measurement to quantify occurrence of discontinuities and their spacing. This value, named Rock Quality Designation (RQD), is the ratio of core recovered in continuous pieces of 100 mm or more to total length of core when diamond drilling is used.

Bieniawski (1976) introduced a geo-mechanical classification system to determine rock masses grouping called RMR. Bieniawski’s classification scheme uses five parameters which are

![Figure 4.3 Measuring the RQD from a core recovered (after Deer and Deer 1988).](image-url)
1. Rock Quality Designation (RQD)
2. Spacing of joints
3. Conditions of joints
4. Ground water conditions
5. Strength of intact rock

RMR ranging from 0 to 100 implies the quality of rock and those with higher values can stand up for longer time without support.

Barton et al. (1974) offered a comprehensive index for defining rock mass properties called Tunnelling Quality Index (Q) as

$$Q = \left( \frac{RQD}{J_n} \times \frac{J_r}{J_a} \right) \times \left( \frac{J_w}{SRF} \right)$$

in which $RQD$ is the rock quality designation; $J_n$ is the joint set number; $J_r$ is the joint roughness number; $J_a$ is the joint alteration number; $J_w$ is the joint water reduction number and finally, $SRF$ is the Stress Reduction Factor. Details can be found in Barton (1974) and Hoek et al. (1997).

### 4.3.4 Mechanical properties of rock mass

Rock mass properties are mostly affected by the discontinuities occurring in the inspection domain. If the inspection area is small compared to discontinuity spacing, then intact rock behaviour dominates the formulation for the rock mass. By choosing a larger examination area, a few discontinuities happen which need to be modelled explicitly. In this spectrum, discontinuities impose un-isotropic behaviour to the model and their mechanical characteristic along the surface of discontinuity can be formed using Mohr-Coulomb criterion and total response of the domain design can be simulated using numerical models.

If the inspection area is broaden, one might inspect numerous joints. And for heavily jointed rock masses, joints will be considered as a random characteristic of rock masses and their overall respond will be implicitly formulated.

Hoek and Brown (1980) proposed a failure criteria to predict the behaviour of rock masses which has become popular and can be expressed as
\[ \sigma'_1 = \sigma'_3 + \sigma_c \left( m_b \frac{\sigma'_3}{\sigma_c} + s \right)^a \]  \hspace{1cm} (4.9)

with \( m_b \) as the rock property factor, and \( s \) and \( a \) as constants related to specification of rock mass (Hoek et al. 1997).

One should notice that this formula is the same as (4.7), but it is more general to consider effects of discontinuities.

The rock property factor \( (m_b) \) is the reduced form of material constant in (4.7) and is given by (Hoek et al. 1997)

\[ m_b = m_i \exp \left( \frac{\text{GSI} - 100}{28 - 14D} \right) \]  \hspace{1cm} (4.10)

and \( s \) and \( a \) are estimated by

\[ s = \exp \left( \frac{\text{GSI} - 100}{9 - 3D} \right) \]  \hspace{1cm} (4.11)

\[ a = \frac{1}{2} + \frac{1}{6} \left( e^{-\frac{\text{GSI}}{15}} - e^{-\frac{20}{3}} \right) \]  \hspace{1cm} (4.12)

Here \( D \) is degree of disturbance, showing how much rock has been disturbed by blasting damages or excavation relaxation.

Also, Geological Strength Index (GSI) is a system of rock-mass classification which can be defined using RMR or Q index by the following formula

\[ \text{GSI} = \begin{cases} RMR - 5, & \text{for } \text{RMR} > 23 \\ 9 \ln \left( \frac{RQD}{J_n} \times \frac{J_c}{J_a} \right) + 44, & \text{for } \text{RMR} > 23 \end{cases} \]  \hspace{1cm} (4.13)

GSI can range from 0 to 100, with 0-10 representing poor condition rock masses, and 100 for intact rocks.

By omitting the minor effective stress \( \sigma'_3 \) from the Equation (4.9), one can have uniaxial compressive strength, as

\[ \sigma'_c = \sigma_c s^a \]  \hspace{1cm} (4.14)
and tensile strength as

\[
\sigma'_t = - \frac{s\sigma_c}{m_b}
\]  

(4.15)

Also, when cohesion of rock mass becomes zero (rock mass without tensile strength) and \(s = 0\), the criteria can be modified to

\[
\sigma'_1 = \sigma'_3 + \sigma_c \left( m_b \frac{\sigma'_3}{\sigma_c} \right)^a
\]  

(4.16)

4.4 Mechanical behaviour of soil

Soil is commonly composed of cohesive grains with friction and is usually assumed to be isotropic and continuous. One of the most widely used schemes to predict mechanical respond of soil is elastic-perfectly plastic (Yu 2006) as depicted in Figure 4.4.

![Figure 4.4 Illustration of elastic-perfectly plastic behaviour and the similarity with a frictional block connected in series with a spring.](image)

According to Figure 4.4, the behaviour of soil can be broken down to three phases: linear elastic stress-strain relationship, yield and post failure flow. This reaction resembles a system consisting a friction block connected in series with a spring, as illustrated in Figure 4.4.
Within limited pressure, soil reacts elastically, meaning that after loading of soil being removed, the displacement will be recovered. This phase of reaction can be defined using two values namely Young’s modulus of elasticity \(E\) and Poisson’s ratio \(\nu\). These values are estimated by triaxial tests or in-situ tests.

During load increment of soil, it approaches a point where displacement is not recoverable anymore and yield happens. Among several yield criteria, Mohr-Coulomb is broadly applied for soil. The Mohr-Coulomb yield function can be stated as

\[
\begin{align*}
    f &= (\sigma_1 - \sigma_3) + (\sigma_1 + \sigma_3)\sin\varphi - 2c\cos\varphi = 0 \\
    (4.17)
\end{align*}
\]

where \(\sigma_1 > \sigma_2 > \sigma_3\) are principal stresses; \(c\) and \(\varphi\) represent cohesion and friction angle respectively and are defined through triaxial or direct shear tests.

Once the yield happens, without any increase in the pressure, strain occurs and the soil starts deforming permanently. Thus here, instead of strain \((\varepsilon)\), the rate of changes in strain \((d\varepsilon)\) can be formulated and in each point can be expressed as (Yu 2006)

\[
\begin{align*}
    d\varepsilon^p &= d\lambda \frac{\partial g}{\partial \sigma} \\
    (4.18)
\end{align*}
\]

in which \(d\varepsilon^p\) is plastic deformation rate; \(d\lambda\) is the increment of plastic multiplier and \(g(\sigma) = 0\) is plastic potential function. In case of associate flow rule, this function is equal to yield function \((g = f)\), otherwise, the flow will be non-associative.

### 4.5 Effect of reinforcement and support

During excavation, the material which was resisting the burden is removed. One way to elaborate stress redistribution is through considering the approach of changes in stress and displacement induced by excavation and also by superposition of stresses to attain the final stage (Sejnoha 2009). In this approach, as shown schematically in Figure (4.5), when material is excavated, excavated surface must experience an imaginary force such that the surrounding material goes through an appropriate unloading effect and a stress free surface to be formed. Assume that
before excavation in the initial state, the material is loaded and stresses are formed in imaginary zone A and B in a way that \( \sigma_a \) and \( \sigma_b \) oppose each other and provide static equilibrium for zone A. This assumption can be stated as the superposition of these two internal forces. Now, if the material from zone A is removed, preserving the initial stress condition \( \sigma_B \) produced in body B requires the new free surface to be loaded by force \( F_{AB} \) applied by zone A on the zone B. Likewise, the force \( F_{BA} \) with the same magnitude but opposite direction must be exerted to zone A to fulfil the static equilibrium state. Here, it was shown that for finishing excavation action, undesirable layer of force \( F_{AB} \) has to be omitted using force \( F_{BA} \) to zone B and therefore obtaining the favourable stress free surface (both in terms of shear and normal pressure). In other words, the surrounding materials around excavation surface carry more loads.

![Diagram](image)

Figure 4.5 Excavation procedure, involved forces and formation of a force-free surface (after Sejnoha, 2009)

The removal of stress \( F_{AB} \) from excavation surface leads to expansion and squeeze in the vicinity of cavern which might be problematic in case of weak rocks resulting in local or global failure (Hudson and Harrison 1997). In order to counteract this instability, the taken measures involve enhancing integrity and rigidity by application of external facilities or consolidation of the neighbouring medium. External facilities or ‘supports’ are structural elements exerting force on the surface to counteract loads on the surface totally or partially and thus prevent probable failure. These measures includes wooden posts, timber cribs, steel support, concrete...
blocks, yielding arcs, rigid ring set, hydraulic props and powered supports (Brady and Brown 2006; Hudson and Harrison 1997).

On the other hand, consolidation, or better said ‘reinforcement’, conserves and enhances the overall properties of the rock mass from inside of the rock mass so that rock becomes able to support itself. Reinforcement includes bolts, trusses, wire mesh and shotcrete which provide bonds to integrate loose parts (rock masses) through application of pressure or filling the joints with sticky pastes (i.e. concrete).

4.6 Displacement due to excavation
During the excavation, digging material causes stress relief in remaining part which can be explained through stored energy. When stress is relieved from the remaining part, the potential energy in the material is converted to kinematic energy or displacement. If the amount of energy released is bigger than a specific value, this displacement causes plastic (unrecoverable) displacement and yield happens (Brady and Brown 2006). Therefore, understanding the mechanism of displacement occurrence is necessary.

Assume a tunnel is advancing into rock masses using either drilling or blasting. The in-situ stress is supposed to be hydrostatic and of magnitude $P_0$. By monitoring changes in displacement and stresses in a point ahead of the tunnel front, it can be inferred that advancement of the tunnel causes significant redistribution of stresses.

Figure 4.6 illustrates a tunnel and a point located in front of excavation plan and which its displacement is monitored during excavation progress. The Figure 4.6 also contains diagram of elastic radial displacement with the direction to centre line of the tunnel. It is seen that the radial displacement starts almost at the distance of twice the radius of the tunnel. Also when tunnel face moves toward the point P, the displacement increases and when the face passes the point, it experiences one third of the maximum value. Further advancement causes further increase of displacement and eventually when the face passes a distance of 3 radii of the tunnel, the point this value approaches its maximum and the no further elastic displacement happens.
4.6.1 Ground and support interaction

This section concerns about interaction of excavated tunnel with the supporting system during a step by step procedure of underground advancement. Assume the example shown schematically in Figure 4.7 where the tunnel front progresses using drilling and blasting method and the hydrostatic pressure with the magnitude of $P_0$ is exerted (Brady and Brown 2006). Steel ribs are installed after each cycle of advancement in the tunnel. Here, the radial displacement of the tunnel and the support pressure of point as advancing face passes the observation section X-X, will be discussed. It should be noted that although steel ribs were used here, this discussion could be extended to other forms of support and reinforcement with minimal modification. Also, the support pressure is the equivalent normal pressure enforced by supports to reciprocate the pressure of the ground periphery.
Figure 4.7 a) Schematic of excavation procedure in full face with installation of external support after each cycle of excavation; b) the radial support pressure-displacement curves around the tunnel and the support interaction (after Brady and Brown 2006).
In step 1, the tunnel front has not approached the section X-X therefore, the rock mass inside the tunnel exerts the same pressure as $P_0$ but with the opposite direction, to maintain the profile at equilibrium.

By step 2, the tunnel front has passed X-X and the internal pressure formerly exerted by rock inside the tunnel has been completely diminished. In this case, the displacement on unsupported part between the last steel rib and the front is limited because in previous section (4.6), it was seen that immediately behind the excavation face, the displacement does not increase to its maximum and it requires 3 radii to approach that value.

Figure 4.7b depicts radial support pressure needed to restrain radial displacement of the tunnel. If the pressure increased immediately behind the excavation face, the support pressure of point B was required to restrain the displacement.

In step 3, the steel ribs have progressed forward and are installed close to the tunnel surface. But they do not experience any pressure since no radial displacement happened yet and the pressure is still presented by point B.

By step 4, the face has progressed to about 3 radii beyond X-X. Here, the displacement approaches its maximum and steel ribs are responsible to carry full load and once the support reaction meets the ground pressure the equilibrium occurs. As illustrated in the Figure 4.7b, for reasonable pressure, the steel ribs show linear elastic behaviour of displacement and pressure, and further pressure (projected by dash lines) causes yield and plastic respond of still ribs.

4.7 Rock reinforcement

Another measure to increase stability of the excavated periphery is through application of rock reinforcement. The reinforcement is usually referred to as active support because the rock itself carries the pressure not external support. Regarding to the implication on the original rock, reinforcement can be classified into two types of systems. The first group exerts curbing pressure to the rock blocks to attach them together by increasing shear on the joints and thus stopping any movement. This group includes un-grouted rock bolts. The second type enhances
the quality of the host rock by adding stiffer and stronger material to seal them. In this category, grouting, application of pre-tensioned steel bars and dowels apply pressure to attach rock blocks together.

The simulation of reinforcement in computer programs depends on the approach chosen. One way is explicitly simulating and modelling all the details rigorously which leads to precise results at the cost of remarkable time and effort. On the other hand, implicit simulation in which reinforced rock is considered as a stiffer material and the distribution of the stronger material is deliberated (Bernaud et al. 1995, 2009).

Through usage of the implicit approach, the reinforcement optimization problem can be simplified to finding the distribution of the stronger material using optimization technique.

4.8 Concluding remarks
This thesis seeks to apply an optimization method for a geotechnical environment. Without enough knowledge about how this medium is composed and behaves, one cannot perform the optimization in the shape and reinforcement.

The applied forces due the weight of geometerals are the main cause of pressure in the underground medium. However, due to sequential loading and movement of the earth crust over billions of years, these pressures are now more complex and needs consideration. Furthermore, when a cavern is excavated, the material which was carrying load is removed, thus the pressure should find another path which in its new form is called induced pressure.

One should notice that earth materials are composed of various components including rock, joints, voids and water. This material composition cause complex physical response to pressure and numerous empirical criteria have been introduced to predict it.

The underground cavern causes redistribution of pressure through a sequence in which the remaining material tries to produce a force-free surface. Application of
reinforcement can result in limitation of this displacement and thus they are used to carry the load in the caverns. This procedure is also mentioned in this chapter.
Coding and Procedure of BESO

5.1 Introduction
In previous chapters, detail of finite element method, optimization procedure and underground material behaviours were discussed. This chapter provides practical coding for finite element and BESO optimization method and provide the readers the means to use for optimization of geotechnical problems. Here, detail of coding and how the program progress is discussed. The optimization process seeks to maximize the compliance of the shape through modifying the designable areas, and in this chapter, means to approach this goal is explained.

In this chapter, detailed procedure of soft-kill BESO engine using MATHWORK MATLAB coding language is provided. Firstly, a mesh is imported from ABAQUS program, and then material properties and matrices are made. Then finite element analysis is performed. Then optimization is run and the process will be repeated in loops resulting in gradual optimization of the shape of the original design domain. Since the main focus is on finite element analysis and optimization process, the basics of coding in MATLAB language is neglected and readers are assumed to have a general knowledge and understanding of coding in this language. Interested readers are recommended to refer to (Ferreira 2009; Kwon and Bang, 1996) for the coding language and further knowledge about finite element analysis.

5.2 Introduction to MATLAB
MathWorks MATLAB is a commercial program for numerical computing, programming and visualization. This high-level language contains various predefined functions which provide the user with fast mathematical operations with ability to solve complex problems. Also all the data is stored in matrices (the name MATLAB comes from Matrix Laboratory) and matrix operations are easily computed while other languages such as C++ and FORTRAN require manual coding for these functions. As an example, matrix multiplication and division can be obtained using operator * and / operators (A*B and A/B respectively represent multiplication and division of matrix A in (to) matrix B) while this process requires
substantially long codes in other language. Since finite element analysis includes matrix handling and operations, MATLAB has been chosen as the platform for the analysis because of its agility and ease in presentation of data. Also the interactive environment and ability to communicate with other Microsoft Windows programs can be regarded as a significant assistance to the analysis.

The procedure is divided into steps and each step is explained separately. Also associated coding is included in each section and is presented with different font and contained in two lines of green ‘%’ as the starting and finishing lines indicators.

One should bear in mind that the lines written in green font coming after ‘%’ are not read by the program MATLAB and only add comments and explanation to the codes to clarify the procedure components.

### 5.3 Defining material and integration properties

The first step in analysis is defining properties of materials and required information for the engine to run, so one should provide Young’s modulus and then Poisson’s ratio as shown in the following lines

```matlab
%%%%%%%%%%%%%%%%%% Defining material properties %%%%%%%%%%%%%%%%%
E=10000; nu=0.3;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Here the E has unit of kPa and also nu is without unit.

Also finite element analysis function demands setting the number of integration in each element and also element type which depending to the design dimensions might be triangular with 3 or 6 nodes (‘T3’ and ‘T6’ respectively), quadrilateral with 4, 8 or 9 nodes (‘Q4’, ‘Q8’ and ‘Q9’ respectively), tetrahedral with 4 nodes (‘Tet 4’), hexahedron with 8 nodes (‘H8’) or other element types.

```matlab
%%%%%%%%%%%%%%%%%%%% Element shape variables %%%%%%%%%%%
number_integ=2;
Type_element='Q4';
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```
5.4 Importing mesh data

Because of its ease in formation and also less complexity and memory use, ‘Q4’
element was chosen. Having set element type, element mesh and nodes should be
formed. There are different programs to produce mesh set in Matlab, however, in
this project meshes were created in a commercial FEM program, ABAQUS, and
then the node and elements information were imported to MATLAB. This
technique not only caters fast formation of element (compared to other program
developed in MATLAB with consumption of considerable amount of time), but the
model can also be run by the ABAQUS itself to verify the authenticity of the finite
element procedure at the first iteration of optimization procedure. In this example,
the tunnel is located at the depth of 50 meters from the surface. Also in order to
make the calculations easier, a local coordination system has been introduced with
the centre of the tunnel as its coordinate centre. Thus the conversion of local
coordinate to the depth can be written as \( D = 50(1 - p_2) \) where \( D \) and \( p_2 \)
represent depth and the y-coordinate of the nodes locations respectively. Also lateral
coordination of the tunnel can be written as \( L = 50p_1 \), in which \( p_1 \) represents the x-
coordinate of the nodes locations.

![Variables - p]

\[
\begin{array}{cccc}
1 & 2 & 3 & 4 \\
1 & 0 & -0.1400 & \\
2 & 0 & -1 & \\
3 & 1 & -1 & \\
4 & 1 & 1 & \\
5 & 0 & 1 & \\
6 & 0 & 0.1750 & \\
7 & 0.1400 & 0.0350 & \\
8 & -0.1400 & -0.1400 & \\
9 & -0.1400 & -0.1400 & \\
10 & -0.1400 & 0.0350 & \\
11 & -1 & 1 & \\
12 & -1 & -1 & \\
13 & 0 & -0.1500 & \\
14 & 0 & -0.1600 & \\
15 & 0 & -0.1700 & \\
16 & 0 & -0.1800 & \\
\end{array}
\]

Figure 5.1 An illustration of first rows of the nodes location.
Figure 5.1 illustrates first rows of nodes data with first column of variables as its x-coordinate location and second column of variables as the y-coordinate location. Totally, 49128 nodes are involved in formation of this model.

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5569</td>
<td>5570</td>
<td>627</td>
<td>628</td>
</tr>
<tr>
<td>2</td>
<td>6023</td>
<td>6248</td>
<td>6393</td>
<td>1128</td>
</tr>
<tr>
<td>3</td>
<td>7684</td>
<td>7594</td>
<td>1211</td>
<td>20539</td>
</tr>
<tr>
<td>4</td>
<td>8586</td>
<td>8061</td>
<td>20680</td>
<td>8324</td>
</tr>
<tr>
<td>5</td>
<td>21276</td>
<td>9547</td>
<td>9593</td>
<td>8922</td>
</tr>
<tr>
<td>6</td>
<td>9477</td>
<td>9307</td>
<td>9403</td>
<td>21380</td>
</tr>
<tr>
<td>7</td>
<td>9886</td>
<td>9612</td>
<td>1396</td>
<td>9723</td>
</tr>
<tr>
<td>8</td>
<td>10769</td>
<td>10884</td>
<td>10702</td>
<td>1445</td>
</tr>
<tr>
<td>9</td>
<td>11307</td>
<td>1537</td>
<td>22563</td>
<td>11252</td>
</tr>
<tr>
<td>10</td>
<td>12040</td>
<td>11827</td>
<td>11944</td>
<td>22816</td>
</tr>
<tr>
<td>11</td>
<td>11373</td>
<td>18137</td>
<td>11656</td>
<td>11787</td>
</tr>
<tr>
<td>12</td>
<td>12092</td>
<td>12291</td>
<td>12208</td>
<td>23378</td>
</tr>
<tr>
<td>13</td>
<td>1725</td>
<td>17599</td>
<td>12785</td>
<td>23563</td>
</tr>
<tr>
<td>14</td>
<td>24017</td>
<td>24016</td>
<td>13131</td>
<td>24092</td>
</tr>
<tr>
<td>15</td>
<td>13769</td>
<td>13858</td>
<td>13981</td>
<td>1867</td>
</tr>
<tr>
<td>16</td>
<td>14308</td>
<td>14412</td>
<td>17850</td>
<td>1970</td>
</tr>
</tbody>
</table>

Figure 5.2 An illustration of first rows of element connectivity.

Figure 5.2 shows elements data. One should note that each row of this table has 4 variables corresponding to 4 nodes building up each ‘Q4’ element. So according to the table, the first element has the connectivity of node number 5569, 5570, 627 and 628. And in order to find the required information about the location of each node, one should refer to nodes data table. Totally 48647 elements has been used to make this set of mesh.

Although it is common to use finer mesh near excavation cavern and coarser mesh on the farther parts, but because in BESO, a constant value for radius in filtering sensitivity is defined, an almost uniform mesh is tried to be applied.
Also one need to define the normal size of elements needed for filtering sensitivity. This number is acquired from the process of mesh generation in ABAQUS.

\[
\text{Defining size of elements} \quad r=0.01;
\]

5.5 Defining boundary conditions and load location

The next step is creating boundary conditions and defining load locations or performing the FEM. This step requires finding the nodes that are located on the edges and middle of the shape.

\[
\text{Defining support conditions}
\]

\[
p_x=p(:,1);
\]
\[
p_y=p(:,2);
\]
LeftEdge = find(p_x < (-1 + 1e-3));
RightEdge = find(p_x > (1 - 1e-3));
midEdge1 = find(p_x > (-1e-3));
midEdge2 = find(p_x < (1e-3));
fixed_x = intersect(midEdge1, midEdge2);
fixed_y = find(p_y < (-1 + 1e-3));
TopEdge = find(p_y > (1 - 1e-3));
node = p;
element = t;
clear p_x p_y p t midEdge1 midEdge2

Primarily, p_x and p_y are allocated to x-coordinate and y-coordinate data of each node separately in a way that p_x shows lateral position and p_y shows vertical location of each node. Then, nodes located on the left edge are identified using their lateral location which is within small distance to x=1 line (small range of error in meshing is considered, this error range is one tenth of an element size). Also nodes situated on the right edge are those whose distance to line x=1 is less than the error range. Moreover, nodes on the line x=0 are specified as the fixed_x and this nodes act as the boundary condition restricting the movement in x-direction. Top_edge defines nodes on the top edge of the shape hitting the y=1 line and fixed_y as the boundary condition and restricting the displacement in y-direction passes on line y=1.

Commonly, in simulation of underground excavation, there exists a burden pressure on top (here, the nodes situated on the TopEdge will be loaded by vertical pressure), horizontal pressure exerted on the LeftEdge and RightEdge and also in some cases an upward pressure acting on the bottom edge of the shape (because of the hydrostatic pressure in deep excavation sites and presence of water). But assumption in this thesis is that the simulation domain is located in shallow depth and dry condition.

The formed mesh and the constrained nodes are presented in Figure 5.4. In this figure, red nodes are fixed nodes in a way that nodes located in the vertical line in the middle are restricted in x-direction movement and nodes on the bottom edge are
fixed in y-direction. Yellow nodes are nodes on top edge and receive vertical force and nodes depicted in green are enforced by horizontal force.

Figure 5.4 Mesh and forces exerted on the nodes.

5.6 Assigning the load variable
Having determined the supporting conditions and load locations, it is time to assign load on the shape and the amount of the pressure should be created based on the predefined values as shown in the following code.

```plaintext
%Pressure values
rou=0.3;
sigmay=-33;
gamma=26.5;
```
In the top codes the \( \rho \) represents horizontal to vertical pressure \( (\rho = \frac{\sigma_h}{\sigma_v}) \), \( \sigma_y \) is amount of vertical pressure in kPa and \( \gamma \) is the unit weight of the material in kN/m\(^3\).

It is supposed that the design domain is located in depth of 20 meters from the ground surface and using the unit weight of 26.5 kN/m\(^3\), one can conclude the pressure of 33 kN/m\(^2\) or 33 kPa (which can be calculated as 20m x 26.5 kN/m\(^3\)=33 kN/m\(^2\)).

### 5.7 Calculation of self-weight function

Each element has a weight which requires obtaining the volume of the element and then calculating the forces caused by gravitation. Finally, this force should be properly distributed to each of composing nodes of element as shown in the following codes.

```
%% Calculation of the self-weight of the elements %

elemSize = size(element,1);
nodeSize = size(node,1);
Ununknowns = nodeSize * 2;
self_weight = zeros(Ununknowns,1);

Q=[1/sqrt(3), 1/sqrt(3);
   1/sqrt(3), -1/sqrt(3);
   -1/sqrt(3), -1/sqrt(3);
   -1/sqrt(3), 1/sqrt(3)];

W=[1; 1; 1; 1];

for iel = 1 : elemSize
    % Looping over elements to calculate ...
    % their weight
    elcon = element(iel,:);
    % Holding connectivity of each element ...
    % in elcon
    eldof_sw=elcon.*2;
    % Degree of freedom of nodes associated with the elements
    % Looping over the 4 points of each ...
    % element starts here
```
point = Q(q,:);  % Using the location of q-th Gauss point
wt = W(q);  % Weight of the q-th Gauss point
eps= point(1);  % eps is x-coordinate of the quad. point
eta= point(2);  % eta is y-coordinate of the quad. point
N=(1/4)*[(1-eps)*(1-eta);  % Formation of shape function of 'Q4'
         (1+eps)*(1-eta);  % element
         (1+eps)*(1+eta);
         (1-eps)*(1+eta)];
dNdx= (1/4)*[-(1-eta), -(1-eps);  % Formation of derivatives ...
          1-eta, -(1+eps);  % of the shape function
          1+eta, 1+eps;
          -(1+eta), 1-eps];
J0 = node(elcon,:)'*dNdx;  % Jacobian matrix of the element

self_weight(eldof_sw)=self_weight(eldof_sw)+N*(1*gamma)*det(J0)*wt;
end  % End of loop on points in elements
end  % End of loop of elements

In the top set of codes, after defining size number of nodes and elements, the variable ‘unknown’ is used to hold the number of total unknown variable and is twice the size of node matrix size because each node has two unknown variable, namely $x_i$ and $y_i$. The variable self Weight is constructed and filled with zeros at the first moment and then zeros are substituted with the results of calculation in each element and node according to element degree of freedom or eldof_sw. This technique firstly allocate a specific amount of memory to the self_weight and then change data in it and is significantly faster than changing the size of the matrices in each loop. The procedure comprises finding Gauss points and the weightings, then looping over elements and calculating their DOF. Then looping over Gauss points and deducing shape function, its derivatives, the Jacobian and finally computing the load on each node is performed (Sherif 2012).
5.8 Creating force matrix

Because force is acting on three edges, namely TopEdge, RightEdge and LeftEdge, it is needed to use three different loops to calculate pressure on these three different edges.


\\% Generates the force matrix due to externally applied loads
f = sparse(Uknown,1);

Q= [1/sqrt(3), 1/sqrt(3);
   1/sqrt(3), -1/sqrt(3); % number_integ=2
   -1/sqrt(3), -1/sqrt(3);
   -1/sqrt(3), 1/sqrt(3)];

W= [1; 1; 1; 1]; % Gauss weight for a 'Q4' element ...
   % with number_integ=2


\\% Vertical force acting on the TopEdge
for e =1: size(topEdge)
   elcon = topEdge(e,:);
   elcony = elcon.*2 ;
   %elcony = elcon.*2-1;
   for q= 1:4
      point = Q(q,:);
      weight = W(q);
      eps=pt(1); % eps is x-coordinate of quad. point
      N = ([1- eps,1+ eps)/2]; % Shape function of the 'L2'element
      dNdx=[-1;1]/2; % Derivatives of shape func. 'L2' elem.
      J0 = abs( node(elcon(2))-node(elcon(1)) )/2;
      % Define the length of edges to which force is exerted
      f(elcony) = f(elcony) + N*sigmay*det(J0)*weight;
   end % End of loop on points in elements
end % End of loop of elements


\\% Horizontal force acting on the RightEdge
re(:,1)= RightEdge(1:(size(RightEdge,1)-1));
re(:,2)= RightEdge(2:size(RightEdge,1));
RightEdge = re;
clear re

for e = 1: size(RightEdge)
elcon = RightEdge(e,:);
elconx = elcon.*2-1;
%elcony = elcon.*2;
for q = 1:4
    point = Q(q,:);
    weight = W(q);
    N = ([1-eps,1+eps]/2)'; % Shape function of the 'L2' element
dNdx = [-1;1]/2; % Derivatives of shape func. 'L2' elem.
J0 = abs(node(elcon(2))-node(elcon(1)))/2;
    % Defining the length of edges to which force ... exerted
    f(elcony) = f(elcony) - N*sigma*y*rou*det(J0)*weight;
end % End of loop on points in elements
end % End of loop of elements

%%%%%%%%%%%%%%%% Horizontal force acting on the LeftEdge %%%%%%%%%%%%%%%
le(:,1)=LeftEdge(1:(size(LeftEdge,1)-1));
le(:,2)=LeftEdge(2:size(LeftEdge,1));
LeftEdge=le;
clear le

for e = 1: size(LeftEdge)
elcon = LeftEdge(e,:);
%elcony = elcon.*2;
elconx = elcon.*2-1;
for q=1:size(W,1)
    point = Q(q,:);
    weight = W(q);
    N = ([1-eps,1+eps]/2)'; % Shape function of the 'L2' element
dNdx = [-1;1]/2; % Derivatives of shape func. 'L2' elem.
J0 = abs( node(elcon(2))-node(elcon(1)) )/2;
    % Defining the length of edges to which force ... exerted
    f(elcony) = f(elcony) + N*sigma*y*rou*det(J0)*weight;
end % End of loop on points in elements
end % End of loop of elements
In the above set of codes, force matrix \( f \) is a sparse matrix instead of full matrix because this method consumes far less memory compared to full matrix. Here, the code is constituted of three parts each of which is designed to apply force on each edge of TopEdge, LeftEdge and RightEdge separately. In each of these parts, after finding the edges connected to the node, these edges are sorted to make the process easier. One should notice that at first section where TopEdge is calculated, only vertical force component of each node (\( elcony \) which represents degree of freedom in y-direction for node of elements) is considered and for section corresponding to specifying forces on LeftEdge and RightEdge only horizontal components are created.

Also, one may notice that defining the pressure requires finding shape function of bar element with 2 nodes at its ends (‘L2’ element), while 4-node quadrilateral elements (‘Q4’) has been chosen. This is because the pressure is only acting on one of its sides, thus it demands calculation of related shape function. Moreover, horizontal pressure is \( rou \) times of \( sigmay \) as discussed in chapter 4.2.1 and \( sigmax \) on the TopEdge is eliminated because of neglecting any shear force acting on the TopEdge. Furthermore, the reason for devoting two separate parts for LeftEdge and RightEdge is the difference in sign of the pressure applied to them. This means since pressure on LeftEdge is toward right and on the RightEdge is toward left, they will have opposite values. The procedure of the calculation is not very different from self-weight function, thus explanation is the same.

### 5.9 Cleaning the work place

Since the preconditioning of the FEM process is finished, some of the unusable variables are better to be cleaned.

```
clear gamma topEdge LeftEdge RightEdge sigmatoy rou
```

---

 básica en el documento.
This procedure not only provides more memory available for other functions, but mainly because the process of optimization is a huge process, it is appropriate to remove unnecessary variables and keep the workplace neat and tidy.

5.10 Assigning variables to store optimization data

Now, it is time to start optimization process and before going through explanation of the optimization, it is required to allocate a matrix for saving data about elements as shown here.

```
%%%% Assigning a matrix to store data for BESO %%%%%
X= zeros(elemSize,7);
%X(:,1)= the optimization design variable
%X(:,2)= sensitivity of each element
%X(:,3)= element elastic energy after filtering
%X(:,4)= sensitivity of last iteration
%X(:,5)= indicator of designable elements
%X(:,6)= centroid of elements in x direction
%X(:,7)= centroid of elements in y direction
X(:,1)= 1;
X(:,5)= 1;
```

Here, the matrix X with the number of rows equal to number of elements and 7 columns has been assigned for storing data of elements. The first column (written as X(:,1) in MATLAB programming language) represents the optimization design variable which can be assigned the value of either 0.001 or 1 representing soft and hard material respectively.

It is noticeable, in this thesis, the BESO uses two materials, namely weak and strong materials, thus two phases are needed, while in other programs, there might be three or more materials. And since this method is soft-kill optimization the inefficient elements are not completely removed and are replaced with softer material having one thousandth (0.001) of the original material stiffness.

The second column (or X(:,2)) is the sensitivity results needed for the optimization process to assess the elements efficiency. The third column (or X(:,3)) is devoted to
the results of filtering of the sensitivity. The fourth column (or \( X(:,4) \)) represents the sensitivity of the last iteration of optimization process and it is used for history averaging to make the process more stable. The fifth column (or \( X(:,5) \)) represents whether elements are designable or not. For example, if the nibbling process is applied, then only elements located on the surface of the current shape are set to be changed. Other elements are inactive and are not included in removal and addition process. If the optimization is not in nibbling mode, all the elements are designable. In this column, 1 represents designable and 0 represents inactive and un-designable. Finally, the last two columns (\( X(:,6) \) and \( X(:,7) \)) are used for filtering purpose and save the centroid location of each element. Since each element needs its sensitivity number to be smoothed with its neighbours, finding the neighbours requires their location defined as the elements centroid.

Because in this example the optimization starts with strong material and this material gradually is substituted with weaker material, all of the elements, in their initial state are filled with strong material, thus the first column has one as its value as its primary value (or \( X(:,1)=1 \)). Also since all the elements are assumed to be designable, the initial value for all of them in the \( X \) matrix is 1 (or \( X(:,5)=1 \)).

### 5.11 Finding centroid of elements

Filtering of sensitivity needs the centre of each element to be found because sensitivity of neighbours of each element gets involved in calculation of averaged sensitivity of that element and for distinguishing whether an element is located within the filtering range, the location of them should be stored.

```
%%%%%%%%%%%%%%%% Finding centroid of each element %%%%%%%%%%%%%%%%%
for iel = 1 : elemSize
    elcon = element(iel,:);  % Element connectivity
    nn = length(elcon);      % Number of nodes per element
    x_ave= 0;
    y_ave= 0;
    for n = 1:nn
        x_ave= x_ave+node(elcon(n),1);
        y_ave= y_ave+node(elcon(n),2);
```

```
The process of finding the elements’ centre consists looping on each element to add the location components of each node belonging to that element. In the current example, the element has 4 nodes (because the elements are ‘Q4’ type), so the x-coordinate and y-coordinate of the nodes are accumulated in \( x_{\text{ave}} \) and \( y_{\text{ave}} \) respectively and averaged. Finally the unrequired variables are cleared from the memory.

### 5.12 Computing compliance matrix

Since the example of optimization is performed in plain strain mode, the corresponding compliance matrix is constituted as shown below

\[
\begin{align*}
d &= \frac{E}{(1+\nu)(1-2\nu)}; \\
C &= d \times \begin{bmatrix} 1-\nu, & \nu, & 0, & \nu; \\
\nu, & 1-\nu, & 0, & \nu; \\
0, & 0, & 0.5-\nu, & 0; \\
\nu, & \nu, & 0, & 1-\nu; \end{bmatrix};
\end{align*}
\]

### 5.13 Defining the parameters of optimization

As the last step before commencement of the optimization cycle, the parameters for optimization need to be set.

\[
\text{vol}=1.; \quad % \text{The volume of the design domain} \\
i = 0; \quad % \text{The counter of the iterations}
\]
change = 1.; % The indicator of the change in compliance
penal = 3.; % Penalty factor in the optimization process
ER= 0.005; % Evolution Rate
volfrac= 0.5; % The desired remaining
rmin= 2.5*r; % Defining the filter radius
iter_max= 100; % Maximum number of iteration

Optimization parameters are defined and explained in above set of codes. Also it should be noted that rmin which is 2.5 times of r represents filter radius of 2.5 sizes of the elements.

### 5.14 The optimization cycle

Since the optimization cycle includes various and long code, thus they are separated to sub-functions which will be called during the procedure. Here, the general properties of the process is firstly discussed, and then explain each sub-function in other sections.

```
% Optimization process
% START of i-th iteration
while change > 0.001

X(:,2)=0;
i = i + 1;
vol = max(vol*(1-ER),volfrac);

if i >1; X(:,4) = X(:,3); end

% FEA & producing objective function
[u_x,u_y,X,c]=
elFEM(X,node,element,Type_elem,number_integ,C,fixed_y, ...
fixed_x,f,self_weight,i,penal);

% Filtering of sensitivities
[X]= filterMesh(elemSize,rmin,X);

% Stablization of evolutionary process
if i > 1; X(:,3) = (X(:,3)+X(:,4))/2.; end
```

110
The BESO cycle (Huang and Xie 2010) starts with a condition of the variable change being larger than a constant. This means as long as the change is larger than 0.001, the cycle will be repeated. The variable change represents change in mean compliance and when this amount becomes small, it can be deduced that the cycle has converged and further modification through optimization does not have significant effect.

Firstly, the sensitivity of element is reset. With this method, sensitivity of the last iterations does not get involved in the current calculations. The variable i counts the number of iterations and this value is updated in each iteration. The i value is 1 at the first iteration and during each cycle the amount is increased by 1 unit.

The volume of unchanged material is indicated with the use of the variable vol. Having the value of one in the first iteration, the vol is reduced step by step, with the step size of Evolution Rate (er). This results in the conversion of the high volume shape to optimized low volume shape. Whenever the optimization approaches the optimum desired volume volfrac (which is also the minimum possible value for vol), the change in the volume is stopped and only reshaping of material is performed with the constant value till the cycle is terminated.
To stabilize the process of optimization, history averaging is needed and unless the process is in its first cycle (and thus sensitivity of previous step does not exist) the value is copied and saved in the fourth column of the matrix $X$ ($X(:,4)$ in MATLAB code) and then the rest of process, involving change of sensitivity, starts.

Then, the finite element analysis should be performed and the sensitivity of each element should be calculated. This function will be explained in one of the following sections devoted to it. The result of sensitivity analysis also should be filtered which will be deliberated in a separate section.

Next, considering the sensitivity results of the previous iteration and the current iteration, history averaging is performed. As the next step, Nibbling and the NonDesing functions modify design-ability of each element and decide whether an element’s design variable can be changed or not. NonDesing is activated in the first iteration to find all the elements located near the sides of original shapes, while in the succeeding iterations Nibbling finds the new surfaces formed due to optimization process. It is anticipated that Nibbling leaves the results of NonDesign without any alteration and only adds newly formed edges to it. These functions are explained in the succeeding sections.

The function ADDDEL is responsible for modification of the design variables with adding or deleting of elements based on their efficiency or sensitivity.

As a way to measure the convergence of the optimization cycle, the variable change assesses the amount of alteration to the compliance and this measuring is activated after the 10-th iteration.

Finally, with the taken measures, the number of maximum number of iterations is limited to $\text{iter}_\text{max}$ and if the iterations counter becomes equal to this value, the cycle stops.

### 5.14.1 Performing finite element analysis

The function $\text{elFEM}$ performs finite element analysis in elastic materials and computes the strain compliance of each element as shown here.
%%%%%%%%%%%%%%%%%%%%%% Finite Element Analysis %%%%%%%%%%%%%%%%%%%%%

function [u_x,u_y,X,c]= elFEM(X,node,element,elemType, ...
    normal_order,C, fixed_y, fixed_x,f,self_weight,i,penal)

nodeSize= size(node,1); % Specifying the number of nodes
elemSize= size(element,1); % Specifying the number of elements

K0= stiff_mat(node,element,Type_elem,number_integ,C,X,penal);
[U,u_x,u_y]=displacement(fixed_y,fixed_x,nodeSize,K0,f,self_weight)

clear K0 % Stiffness matrix is a huge matrix ...
% and should be deleted A.S.A.P

stress_points=[ -1 -1;1 -1;1 1];

c(i)=0.; % Compliance matrix for each iteration

for iel = 1 : numelem % Looping over the elements starts here
    elcon= element(iel,:); % Element connectivity
    nn= length(elcon); % Number of nodes per element
    eldof= [ 2*elcon(1)-1; 2*elcon(1)]; % Element DOFs, each element...
    eldof= [ 2*elcon(2)-1; 2*elcon(2)]; % has 4 nodes each of which...
    eldof= [ 2*elcon(3)-1; 2*elcon(3)]; % has 2 DOF in x,y direction
    eldof= [ 2*elcon(4)-1; 2*elcon(4)];

    elemEnergy= 0; % Element energy resets in...
        % each element loop
    for n = 1: nn % Looping over the node
        point= stress_points(n,:); % stress points in elements
        B_mat= Bmatrix(point, iel,element,node);
        eps= B_mat*U(eldof); % Strain at each node
        eps(4)=0; % Addition of 4-th component to eps
        sigma= C*eps; % Stress at each node
        en= eps'*sigma; % Energy at each node
        elemEnergy= elemEnergy+en; % Storing energy of nodes
    end % End of looping on nodes of elements

    X(iel,2) = 0.5*(X(iel,1)^(penal-1))*elemEnergy/nn; % Storing sensitivity number in each elem

    c(i)= c(i)+0.5*(X(iel,1)^penal)*elemEnergy/nn;
The process of finite element analysis (Sherif 2012) is depicted and explained in above code. After finding the size of node matrix, the stiffness matrix and displacement are calculated (refer to section 4.14.1.1 and 4.14.1.3). Stiffness matrix is a gigantic matrix with the size of (2*nodeSize)*(2*nodeSize) for 2D elements and (3*nodeSize)*(3*nodeSize) for 3D elements. In this example with 49128 nodes, the stiffness matrix is a matrix with (2*49128)=98256 rows and (2*49128)=98256 columns and totally (2*49128)*(2*49128)= 9.65E+09 components and even with this comparably small number of nodes even the fastest available computers hang. While application of sparse matrix has speeded up the calculations, but still the matrix consumes a big fraction of available memory and it needs to be cleared as soon as the related computation has been done and the results were used. The usage of stiffness matrix is restricted to obtaining the displacement matrix in the current iteration and is not used anywhere else because in each iteration the shape of the domain changes and the corresponding stiffness matrix should be reconstructed. It should be noticed that in contrast to stiffness matrix, the force and boundary conditions do not change over iterations and should not be removed or reconstructed in each iteration.

It is interesting that in the FEM, the force and displacement value are calculated at nodes and also forces at the neighbouring elements are transmitted through the nodes of the elements, thus they should be considered as point forces acting on the nodes. Since they are not distributed, there is no necessity for integration, attaining the Jacobian and shape function matrix in these actions.

Then, the size of the mean compliance matrix is extended. When the number of the counter i increases, the size of the compliance matrix also should be boosted accordingly to provide enough room for saving the current mean compliance value. This issue has been address using the appropriate code.
As mentioned in previous chapters, the process of optimization is based on minimizing the compliance matrix. It can be seen in the codes that the compliance matrix is gained through the addition of the sensitivity number of each element. It should be noted that here, a power-law interpolation scheme is deployed.

Then looping on each elements starts, and after obtaining the required data, the loop on the nodes of each element commences.

During the loop over the nodes, the B matrix which is a matrix to convert displacement to strain should be calculated and with using this matrix, strain is obtained. The B matrix is a 3*8 matrix and if this matrix is multiplied in displacement matrix of the current DOF of the element (the size of U(eldof) is 8*1 because of 4 nodes each of which having 2 DOF in x and y directions), the size of the strain would have the size of 3*1 which can be shown as

\[ \varepsilon = \begin{bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{bmatrix} \]  

(5.1)

On the other hand, in plain strain mode, a z-direction force exists which can be imagined to act as the pressure trying to keep the thickness of the section constant.

\[ \sigma = \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \tau_{xy} \\ \sigma_{zz} \end{bmatrix} \]  

(5.2)

The pressure matrix can be obtained using

\[ \sigma = C * \varepsilon \]  

(5.3)

and this multiplication seems impossible because the size of the compliance matrix C is 4*4 and the size of the strain matrix \( \varepsilon \) is 3*1. Thus one needs to add another component to the strain matrix representing the strain in the z-direction. However, as in the plain strain mode, the thickness of in the z-direction remains constant, the strain of this direction is assumed to be zero. Thus the strain matrix becomes
After multiplication of the compliance matrix in the strain matrix, the stress matrix is built up and the z component of this matrix, as mentioned earlier, is not zero.

The node compliance is calculated through multiplication of the transpose of the strain matrix at the stress matrix. Then, in order to calculate the element energy, these amounts are added up and accumulated in the variable `elemEnergy` and then averaged.

Figure 5.5 Mesh and energy distribution in colour bands

BESO judges elements efficiency based on their sensitivity and elements sensitivity is calculated through averaging the sensitivity of the nodes constructing the element.
The last operation in each element cycle is obtaining the sensitivity of each element and compliance of the shape using the appropriate equations (Equation 3.40). The result is shown in the Figure 5.5.

5.14.1.1 Calculation of stiffness matrix
As one of the main components of finite element analysis, here the procedure of gaining stiffness matrix is presented. To calculate displacement, stiffness matrix should be constructed and then multiplied in force matrix. Here, the method of producing stiffness matrix is explained.

```matlab
function K = stiff_mat(node, element, Type_elem, number_integ, C)

nodeSize = size(node,1);
elemSize = size(element,1);
Unknown = nodeSize*2; % Number of total unknown value
K = sparse(Unknown, Unknown); % Formation of empty sparse matrix...

for iel = 1 : nodeSize
    elcon = element(iel,:); % Element connectivity
    nn = length(elcon);
    eldof = [ 2*elcon(1)-1; 2*elcon(1);
             2*elcon(2)-1; 2*elcon(2);
             2*elcon(3)-1; 2*elcon(3);
             2*elcon(4)-1; 2*elcon(4)];
    Q = [1/sqrt(3), 1/sqrt(3);
         1/sqrt(3), -1/sqrt(3);
         -1/sqrt(3), -1/sqrt(3);
         -1/sqrt(3), 1/sqrt(3)];
    W = [1; 1; 1; 1]; % Gauss weight for a ‘Q4’

    for kk = 1 : 4 % Loop on Gauss points
        point = Q(kk,:);
        eps = point(1); % eps is x-coord. of quad. point
        eta = point(2); % eta is y-coord. of quad. point
        N = (1/4)*[(1-eps)*(1-eta);
                 (1+eps)*(1-eta);
                 (1+eps)*(1+eta);
                 (1-eps)*(1+eta)];
```
\[ dNdx = (1/4)[-(1-\eta), -(1-\epsilon); \quad \% \text{Derivatives of the ...} \\
1-\eta, -(1+\epsilon); \quad \% \text{shape function} \\
1+\eta, 1+\epsilon; \\
-(1+\eta), 1-\epsilon]; \]

\[ J_0 = \text{node}(elcom,:)'*dNdx; \quad \% \text{Element Jacobian matrix} \]

\[ Bmat = B\text{matrix}(\text{point,Type}_\text{elem},iel,element,node); \]

\[ K(\text{eldof,eldof}) = \ldots \]

\[ K(\text{eldof,eldof}) + Bmat'*C(1:3,1:3)*Bmat*W(kk)*\text{det}(J_0)*((X(\text{iel},1)).^\text{pen al}); \]

\[ \text{end} \quad \% \text{End of looping on Gauss Points} \]

\[ \text{end} \quad \% \text{End of looping on elements} \]

\[ \text{end} \quad \% \text{End of function} \]

Since most of the procedures were explained at the last sections, they will not be explained again. The number of unknown variable is double the number of the node because each node has two degrees of freedom. As said before, the sparse matrix is applied to store data of the stiffness matrix because the stiffness matrix sparsely occupies the available places and number of unfilled components significantly outweighs the number of occupied spaces. During each loop on the elements, when the corresponding nodes and the DOFs in the elements were found, looping on the quadrature points start. During this cycle, firstly the shape functions and their derivatives on the active quadrature point are formed. Then Jacobian is found and then B matrix is shaped. Finally the DOFs of the element are the navigator for the mapping of the current element stiffness over the global stiffness matrix.

Computation of the stiffness matrix is the most time consuming process during the optimization, taking almost 90% of the total time. The reason is repetitive procedure of looping on the elements, nodes and also calling sub-functions during each loop while a huge amount of data is stored in the matrix. Furthermore, adding data to the global matrix using DOFs as the address for the process, counts for a considerable portion of the used time.
### 5.14.1.2 Calculation of strain-displacement conversion matrix

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Strain displacement matrix %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

**function** Bmat = Bmatrix(point, Type_elem, iel, element, node)

elcom = element(iel,:);

n = length(elcom);

eps = point(1); % eps is x-coordinate of the quad.

eta = point(2); % eta is y-coordinate of the quad.

\[ \frac{\partial N_1}{\partial x} \quad 0 \quad \frac{\partial N_2}{\partial x} \quad 0 \quad \frac{\partial N_3}{\partial x} \quad 0 \quad \frac{\partial N_4}{\partial x} \quad 0 \\
0 \quad \frac{\partial N_1}{\partial y} \quad 0 \quad \frac{\partial N_2}{\partial y} \quad 0 \quad \frac{\partial N_3}{\partial y} \quad 0 \quad \frac{\partial N_4}{\partial y} \\
\frac{\partial N_1}{\partial y} \quad \frac{\partial N_1}{\partial x} \quad \frac{\partial N_2}{\partial y} \quad \frac{\partial N_2}{\partial x} \quad \frac{\partial N_3}{\partial y} \quad \frac{\partial N_3}{\partial x} \quad \frac{\partial N_4}{\partial y} \quad \frac{\partial N_4}{\partial x} \]

(5.5)

As explained in section (2.1.3), the B matrix for a ‘Q4’ element is written as
This function is formed in each element separately and since this function is called repetitively, one may consider that as the heart of finite element. By default, this matrix has three rows and each quadrature point occupies two columns. The components used in this matrix are derivatives of the shape functions and interestingly odd numbers of the first row are filled up with the derivatives of the shape function with respect to x and other components in this row are zero. The even numbers in the second row are filled with the derivatives of shape functions with respect to y then the rest are zero. The last row is alternatively filled with the derivatives with respect to y and x for each quadrature point.

5.14.1.3 Attaining Displacement
The function displacement calculates the displacement according to stiffness matrix, forces and degree of freedom for nodes. The process is shown in below codes.

```matlab
function U = displacement(fixed_y, fixed_x, nodeSize, K, f, self_weight)

Unknown = 2 * nodeSize;

dof_x = (fixed_x .* 2) - 1;  % Predefined displacement in x-dir
dof_y = fixed_y .* 2;        % Predefined displacement in y-dir

dof = union(dof_x(:), dof_y(:));  % All of predefined displacements

Unknown_dof = setdiff((1:Unknown)', dofs);  % Unknown DOF needed to be defined

F = f(Unknown_dof) + self_weight(Unknown_dof);  % Total forces acting on nodes

u = K(Unknown_dof, Unknown_dof) \ F;

U = zeros(Unknown, 1);
U(Unknown_dof) = u;
end  % End of function
```

Here, after defining the size of unknown variables, fixed degrees of freedom are identified. Regarding the idea of having the even numbers in displacement (also the same concept applies for force matrix) as x-direction DOFs and even numbers as y-direction DOFs, `dof_x` and `dof_y` are built up using mentioned equations. Total fixed
displacements are stored in \textit{dof} and the difference between total possible DOFs (which is the variable \textit{Unknown}) and is shown as \textit{Unknown_dof}. Both the external force and the self-weight are added up to form the total force acting on the system. As pointed out in section 2.2.10, the components of stiffness matrix which are related to fixed DOFs are removed from global matrix. Thus, to keep the matrix operation possible, the size of force also should be reduced to the size of \textit{Unknown_dof}. Also the approached displacement matrix is in its reduced version and construction of the full displacement matrix needs placing the reduced version of displacement matrix using the \textit{Unknown_dof} as the address for placement and adding zeros in the place of fixed DOFs.

\textbf{5.14.2 Filtering the sensitivity}

After finishing functions related to finite element analysis, the attention will be focused on functions used in process of BESO. The function \texttt{filterMesh} smooths the sensitivity gathered from \texttt{elFEM} function by using weighted averaging over neighbouring area as discussed here.

```
function \[X\] = \texttt{filterMesh} (\textit{elemSize}, \textit{rmin}, \textit{X})
\textit{X}(\textit{\cdot},3) = 0;
for \textit{iel} = 1 : \textit{elemSize}
    \textit{sum} = 0;
    for \textit{iel1} = 1 : \textit{elemSize}
        \textit{dx} = \textit{X}(\textit{iel},6) - \textit{X}(\textit{iel1},6);
        if \text{abs} (\textit{dx}) <= \textit{rmin} \quad \% \text{Vertical filter}
            \textit{dy} = \textit{X}(\textit{iel},7) - \textit{X}(\textit{iel1},7);
            if \text{abs} (\textit{dy}) <= \textit{rmin} \quad \% \text{Horizontal filter}
                \textit{factor} = \textit{rmin} - \texttt{sqrt}(\textit{dx}^2 + \textit{dy}^2);
                \textit{sum} = \textit{sum} + \text{max}(0, \textit{factor}); \quad \% \text{Circular filter}
                \textit{X}(\textit{iel},3) = \textit{X}(\textit{iel},3) + \text{max}(\textit{factor},0) \times \textit{X}(\textit{iel1},2);
            end
        end
    \end
end
\textit{X}(\textit{iel},3) = \textit{X}(\textit{iel},3)/\textit{sum};
```

121
filterMesh smooths the sensitivity values using a filtering scheme in which the elements sensitivities are averaged with their surrounding elements. In this procedure, the results of the filtering are reset to provide a fresh space for new round of calculation. Originally, the technique includes holding an element and then going through all other elements to see whether their locations are within the filter radius distance from the centroid of the held element. And if any of elements are located in the vicinity of the held element, a factor of closeness to the held element is obtained and multiplied by their sensitivity value, then averaged. Thus, elements that are closer to the held element get larger factors and their sensitivities have more effect on the sensitivity of the held element. After finding the averaged sensitivity of the held element, that element is released and the next element is held. This loop continues until all the elements are held and effects of all other elements are assessed.

This method includes two loops one holding elements, one by one, and the inner loop for assessment of the effect of other elements. Thus, the number of calculation needed to be performed is \((\text{number of elements}) \times (\text{number of elements} - 1)\).

In the modelling with 48674 elements, this number would be\((48674) \times (48674 - 1) = 2,369,109,602\) which is a huge number and the calculation would be really time consuming. In order to speed up the calculation, two more filters have been applied, which weed out the elements located far from the held element. The first filter only accepts the elements having horizontal distance less than the filtering radius, and the second only accepting elements with vertical distance less than the filtering radius as shown in Figure 5.6. As shown in this figure, element that are located in the filtering circle are coloured.
5.14.3 Finding unique edges

Elements located on the edges and surfaces of a shape are highly demanded in shape optimization methods because domain of action in this method is limited to edges and surfaces (in 2D and 3D respectively) of a shape. Thus a way to distinguish which elements are located on the outside of a shape is very important. One of the most interesting and precise way of detecting these elements is to check if these elements have any edges (or surfaces) which are not shared with other elements or literally unique.

The following code, identifies the nodes that are located on the edges (or surfaces) of all the elements (Person and Strang 2004).

```
Finding unique edges

function four=sideEdge(node,element)
edges= [element (:,[1,2]); % The edges on each element
       element (:,[2,3]);
```
element (:,[3,4]);
element (:,[4,1]));
edges= sort(edges,2);
[val,iq,jq]= unique(edges,'rows');
vector= histc(jq,1:max(jq));
qx= find(vector==1);
four= edges(iq(qx),:);

Each ‘Q4’ element has four edges and which can constructed as: Edge connecting 1st node to the 2nd node, edge connecting 2nd node to the 3rd node, edge connecting 3rd node to the 4th node and edge connecting 4th node to the 1st node. With this technique all the edges of elements, will be stored in a matrix. Because the numbering in all of elements is clockwise, an edge shared between two neighbouring elements are sorted differently in edges (for example, in one element, the edge connects node number 100 to 325 while in the neighbouring element sharing this edge, the edge connects node number 325 to 100), so one cannot find unique edges unless the nodes in each edge are sorted from small to large. After sorting the node numbers in edges, one can find the unique edges. Then the edges also get sorted in columns.

By identifying the elements these edges, elements located at the out edges (or surfaces) can be found using the following codes:

for iel=1:2*(size(four,1))
    [row1,col]= find(element==four(iel));
    X(row1,5)= 1;
end
In this set of code, after finding the side edge, elements that own these edges are identified and their designability variable, X(:,5), is turned on by using number 1. The result of this function is presented in Figure 5.7.

Figure 5.7 Elements located on the external edges of the shape a) a general view; b) close up with mesh depicted.
5.14.4 Non-design filter
In previous section, it was seen that with usage of the function sideEdge and some simple codes, one can identify the elements located on the edges of the shape. However, the focus in designing is only the cavern and the excavation site. Thus, a way to limit the designable part to tunnel is required. The following code solves this issue.

function \( X = \) NonDesign(element, node, \( X \))
\( X(:,5) = 0; \)
\( \text{four} = \) sideEdge(node, element);

\[
\begin{align*}
\text{for } \text{iel} & = 1:2*(\text{size(four,1)}) \\
[\text{row1}, \text{col}] & = \text{find}(\text{element} == \text{four(iel)}); \\
\text{if } X(\text{row1},6) & < 0.5 \quad \% \text{Right limit} \\
\text{if } X(\text{row1},6) & > -0.5 \quad \% \text{Left limit} \\
\text{if } X(\text{row1},7) & > -0.5 \quad \% \text{Down limit} \\
\text{if } X(\text{row1},7) & < 0.5 \quad \% \text{Top limit} \\
X(\text{row1},5) & = 1;
\end{align*}
\]

In this code, after finding the edge side, elements that own them are to be marked. However if these elements locations satisfy the following rule,

\[-0.5 < x_c < 0.5 \]
\[-0.5 < y_c < 0.5 \]

(5.6)

are marked as designable elements. In these equations, \( x_c \) and \( y_c \) denote location centroid of an element in \( x \)-coordinate and \( y \)-coordinate respectively. The Figure 4.8 illustrates the output of the NonDesign function.
Figure 5.8 Elements located on the internal walls of the opening.

### 5.14.5 Nibbling filter

Nibbling filter specifies the elements that are surrounded by switched elements and the procedure is presented in the following lines.

```
function [X] = Nibling(element, node, X)

four = sideEdge(node, element((X(:,1)==0.001),:));

for iel = 1:2*size(four,1)
    [row1, col] = find(element==four(iel));
    X(row1,5) = 1;
end
```

end
```
In this set of code, only elements having contact with switched elements are activated for designing. By finding shared edged with switched elements, one can find elements that should be considered designable. The output is presented in Figure 5.9.

![Figure 5.9 Switched element illustration (with green colour) and newly formed faces (red elements) after alteration of the design domain.](image)

This function is not used in the first iteration during which no switched element exists. Since this function does not reset the designablity variable of the elements, this function can be used in all the proceeding iterations while keeping the designable elements from the succeeding iterations. This means that designable elements are firstly found by NonDesign function in the first iteration, and then in each iteration with development of switched area, new designable elements are added to formerly constructed output. This feature shows that elements that have been switched once, still remain designable, providing the chance for the program to include them in the current switching process.
Figure 5.10 Designable elements after the first iteration depicted in red.

5.14.6 Changing the design variable

In this part, the element that deemed to be inefficient, are replaced by weaker material. So a procedure that sorts the sensitivity and softens elements is needed while the restriction on the volume is not breached. Of course, only elements marked as designable should be considered for the changes.

```
function [X]=ADDDEL(vol,X)
[i,j,t]=find(X(:,3).*X(:,5)); % Elements with nonzero sensitivity
clear i j
Lmin = min(t); Lmax = max(t);
while ((Lmax-Lmin)/Lmax > 1.0e-5) %Loop to match constraints with ...
    th = (Lmin+Lmax)/2.0;
    for iel= 1:size(X,1) % Loop over elements
        if X(iel,5)==1 % Finding designable elements
            X(iel,1)=max(0.001,sign(X(iel,3)-th));
        end
    end
    if sum(X(:,1))-vol*(size(X,1)) > 0; % Updating criterion to ...
        % match constraints
        Lmin= th;
```

else
    Lmax= th;
end
end
end

Here, since most of the elements are set un-designable, the removal and addition should be limited to small fraction of elements which are designable. As mentioned earlier, the designable elements are specified with 1 in the 5\textsuperscript{th} column of the X matrix and 0 in that matrix indicates un-designable. If the sensitivity is multiplied in these variables, one will have 0 for un-designable element (0 \times \textit{sensitivity number} = 0) and the sensitivity for designable elements (1 \times \textit{sensitivity number} = \textit{sensitivity number}). If the zeros are filtered out of the numbers, one would only have sensitivity number for designable areas. Here, the \texttt{find} function in MATLAB extracts nonzero values in the sensitivity numbers.

The process commences by acquiring the maximum and minimum of the sensitivities. Then they are averaged and the obtained value is called \( \theta \).

The sensitivity numbers of designable elements are judged base on the variable \( \theta \) and elements with larger value will have the design variable of 1 and those with lesser value of sensitivity number will have 0.001 as their design variables. This process continues until the volume restriction for change is satisfied and the assessment variable \( \theta \) moves to keep the volume of changed material desirable.

5.14.7 Presentation of the results
In each iteration, the optimized shape should be illustrated and also the value of compliance should be printed on the screen to inform the user about changes and gradual movement of the compliance toward its optimal value. The function \texttt{resplot} contains two sections, first of which plots the mesh and the value of the designablity. The second section displays the number of iteration, volume, mean compliance and change in mean compliance.
The first part of the codes, sets up a plot area, and keeps this plot fixed and a cycle to add elements one by one is deployed. In the cycle, location components of the nodes are saved separately and the value of design variable is read and the sizes of all these components are equalized. Then the elements are plotted and filled with a colours corresponding to their value (colour bands).

The second part of the code, print the results of the iteration starting with the iteration number and then compliance (objective function), volume of the unchanged material and the amount of change are displayed on the screen.
5. 15 Concluding remarks

This chapter is devoted to providing the readers with detail of coding for the finite element method and optimization process. The Matlab programming language has been used to perform the procedure due its simplicity and ease of mathematical calculation.

The steps start with defining the material properties and elements type. Here, because of readily accessible outputs, Abaqus package was chosen to produce mesh of elements. After importing the meshes data into the Matlab codes, boundary conditions and applied forces needs to be assigned. Since the geomaterials weight can affect the pressure distribution, if the excavation is located near surface, the weight should be calculated and considered.

The method to produce the weight matrix and force matrix has been elaborated and all the requirements are met to perform the finite element method. Since finite elements require huge storage area, large matrices are allocated to save data related to optimization

Finite elements is consisted of a loop of various actions. The first and most time consuming action is finding stiffness matrix of the current shape. One should note that finite element and optimization process are embodied in a single loop, thus changing the design domain due to the optimization, changes the stiffness of the shape, resulting in a demand to produce the stiffness matrix in each step of the loop. Once the stiffness matrix is produced, stress and strain of the elements can be obtained and then sensitivity of the element, as the objective function, be reached. It should be noticed that all of the elements are not designable and available to be switched, thus in each step of the loop, the designability of elements should be restrained accordingly.

Finally to keep track of the changes in the shape and to illustrate the objective function during each cycle, codes to produce figures are provided.
Chapter 6

Numerical examples and consideration of non-linearity

6.1 Introduction
In this chapter, to clarify the features of the optimization, examples of application of the optimization problem in various conditions are discussed. The mentioned examples include short cantilever beam, underground opening, reinforcement optimization and optimization of pillar with nonlinear material. Also optimization to increase stability of a slope will be considered.

6.2 Short cantilever beam optimization and survey of the effect of optimization parameters
This section discusses Short Cantilever Beam (SCB) problem using various parameters for optimization. The initial condition of the supporting conditions, load and the mesh is depicted in Figure 5.1

Figure 6.1 Mesh, boundary condition and applied force in the SCB example.
The design domain is composed of a 100*150 quadrilateral 4-node (Q4) elements. The node at the left edge are completely fixed in both x and y directions. A concentrated force is exerted on the middle node of the right edge dragging the node to down. Experiment has shown that because of comparative nature of optimization in assessing the efficiency of elements, the amount of force and the Young’s module do not affect the optimization results.

Here, the material is supposed to be made up of strong material and optimization replaces inefficient material with a soft material which is 1000 times softer than the original matter. Through the following examples, the effects of filtering radius and is discussed.

### 6.2.1 Impact of filtering radius

Having identified the initial mesh and properties of the original material, the optimization of the SCB problem using various filtering radii will be considered. The example include of filtering radii of \( r_{\text{min}} = 0, r_{\text{min}} = r, r_{\text{min}} = 2r, r_{\text{min}} = 3r, r_{\text{min}} = 4r \) and \( r_{\text{min}} = 5r \) in which \( r_{\text{min}} \) is the filtering radius and \( r \) is the size of element. The evolution rate for the optimization has been chosen as 2%. Choosing the radius filter of 0, results in formation of the checkerboard pattern as is illustrated in Figure 6.2 and a higher objective function. To remedy the checkerboard pattern and the un-optimized objective function, the filter radius scheme should be applied. However, selecting very high filter radius imposes a restriction on the procedure by banning formation of members thinner than \( r_{\text{min}} \) and thus reducing complexity and higher value for objective function. The results of applications of higher filtering radii in the following examples confirm obtaining lower objective function when using high filter radii, as shown in Figure 6.9.
Figure 6.2 Optimized shape and objective function of an SCB with $R_f=0$ and $ER=0.02$.

Figure 6.3 Optimized shape and objective function of an SCB with $R_f=1$ and $ER=0.02$. 
Figure 6.4 Optimized shape and objective function of an SCB with RF=2 and ER=0.02.

Figure 6.5 Optimized shape and objective function of an SCB with RF=3 and ER=0.02.
Figure 6.6 Optimized shape and objective function of an SCB with Rf=4 and ER=0.02.

Figure 6.7 Optimized shape and objective function of an SCB with Rf=5 and ER=0.02.
Figure 6.8 Optimized shape and objective function of an SCB with Rf=15 and ER=0.02.

Figure 6.9 Plot of the final objective functions for the optimization problems using different filtering radius.

It can be seen in Figure 6.9 that the values of the obtained objective functions are presented in the vertical axis and the filtering radius in the horizontal axis. It can be concluded that the best filtering radius to gain the most optimized shape is almost 2.5 times of the element size. It is understood that using large filtering radius causes restriction in the thickness of the members thus the objective function is not minimized using large filtering radius.
6.3 Optimizing underground excavation

Approaching the optimized shape and reinforcement layout of an underground cavern has been a most challenging issues in the underground design. By application of the optimization method, the problem of shape optimization and reinforcement optimization can be resolved. However, loading procedures and material properties in underground medium are completely different and more intricate and tackling these obstacles requires more consideration.

As presented earlier, the geomaterials are non-linear, anisotropic and non-homogenous. However, focusing on linear material would lighten the way for further research. Here, the linear elastic material will be applied and non-linearity will be introduced in the next sections.

In this section, the method introduced by other researchers will be reviewed and then the results of the program will be discussed.

Ren et al. (2005) brought forward optimization of a cavern using ESO in a linear elastic medium. Exerting remotely distributed forces on a large mesh (to omit boundary effect), they optimized the underground shapes of caverns while weight of ground material was neglected. As the measurement for element efficiency and removal, they used intuitively set criteria which was mean principal compressive stress expressed as

$$\bar{\sigma} = \frac{\sigma_1 + \sigma_2 + \sigma_3}{3}$$  \hspace{1cm} (6.1)

where $\sigma_1$, $\sigma_2$ and $\sigma_3$ are principal stresses. The mean principal of all the elements are compared with the threshold and those whose values are less than this threshold should be removed. Two controlling parameters were used to specify this threshold, namely, Rejection ratio (RR) and Volume Removal rate (VR). RR is the fixed percentage which alters the criterion for switching in a way that

$$\sigma_{th1} = RR \cdot \bar{\sigma}_{max}$$  \hspace{1cm} (6.2)

One should notice that since the amount of mean stresses are not evenly distributed, the rejection ratio might cause significant changes in the topology. Therefore, they
introduced another ratio as VR. VR is a constant number in percentage to define how much of the total volume in each iteration are allowed to be removed and can be written as

\[ \sigma_{th2} = \bar{\sigma}(n(1 - VR)) \] (6.3)

in which \( n \) denotes number of element in the current iteration, \((1 - VR)\) is the percentage of the remaining material in the current iteration and \( \bar{\sigma}(n(1 - VR)) \) is the highest mean stresses corresponding to the remaining volume of that iteration. Since changes in the ESO are irrecoverable, they chose the minimum of these two thresholds for removal of the elements as shown here

\[ \sigma_{th} = \min \{\sigma_{th1}, \sigma_{th2}\} \] (6.4)

According to analytical solution, the best shape of an underground opening is an elliptic with the ratio of height to width equal to ratio of vertical to horizontal pressure which can be expressed as

\[ \frac{\sigma_v}{\sigma_h} = \frac{H}{W} \] (6.5)

The optimized shapes in Figure 6.10 show the consistency with the mentioned pressure ratio. Furthermore, 3D example of a tunnel intersection was studied and the results of optimization using von Mises as the rejection are illustrated in Figure 6.11.

Figure 6.10 Optimized shapes of an opening under different vertical to horizontal pressure ratio: a) \( \sigma_v/\sigma_h=1 \); b) \( \sigma_v/\sigma_h=1/2 \); c) \( \sigma_v/\sigma_h=1/3 \) (after Ren. et al 2005).
To verify the proposed program, for the 2D example, the method was applied to an underground opening with the pressure ratio as $\sigma_v/\sigma_h=3/1$, the optimization parameters of $ER=0.001$ and $r_f=2.5$. It can be seen in Figure 6.12 that the opening shape matches the following ratio.

$$\frac{\sigma_v}{\sigma_h} = \frac{Height}{Width} = 3$$  \hspace{1cm} (6.6)
Figure 6.12 Optimized shapes of a half of an underground opening with the vertical to horizontal pressures ratio of $\sigma_v/\sigma_h=3/1$.

Also application of the optimization method was attempted on 3D Model to obtain the stiffest intersection (lowest objective function). In this model, the pressure ratio was chosen to be $\sigma_v/\sigma_h=1/1$ and the optimization parameters were $r_f=2.5$ and $ER=0.002$. It can be seen that the result is similar to the outcome of previous research.
6.4 Reinforcement optimization

Yin et al. (2000), Yin and Yang (2000a, b), Liu et al. (2008) and Ghabraie (2009) solved reinforcement optimization problem using different objective functions. The proposed BESO technique by Ghabraie (2009), is used here and the application is extended to 3D versions.

Firstly an opening in 2D is assumed here for the reinforcement. In each iteration, the volume of the reinforced material gradually increases and when the volume of reinforced material approaches 5% of the size of the domain, the increase in the volume of reinforced material stops. Finally when the changes of the objective function become nearly zero, the process terminates. Here, materials are assumed to be elastic and the ratio of the module of elasticity in the reinforced material to the original rock is considered to be 4 times or

\[ \frac{E_R}{E_O} = 4 \]  

(6.7)
The filtering radius is chosen to be $r_f=2.5$ size of elements. The Evolution Rate (ER) is equivalent to 0.1% and mean compliance is picked as the objective function. The optimization problem can be expressed as

$$\min_{x,u} \ c(x) = f^T u$$  \hspace{1cm} (6.8)

where $K(x)u = f$,

$$x_i \in \{0,1\}, \quad i = 1,2, ..., N$$

$$\sum_{i=1}^{N} x_i V_i - V_R \leq 0$$

with $V_i$ as the volume of the i-th element and $V_R$ as the desired volume of the reinforced material. Also, $x_i = 0$ represents the original material and $x_i = 1$ represents reinforced material. The obtained results of the optimization are illustrated in Figure 6.14. It can be seen that the results orientation is in a way that two bars tend to be created to withstand the larger pressure and reduce the pressure over the tunnel.
Figure 6.14 Reinforcement optimization of a non-circular underground opening for the pressure ratio of: a) $\sigma_v/\sigma_h=1/1$; b) $\sigma_v/\sigma_h=2/1$; c) $\sigma_v/\sigma_h=3/1$. 
6.5 Optimization in non-linear materials

While many researches in optimization focused on linear material, here, the steps to expand the usage into non-linear material is taken. However due to computational cost of this analysis in non-linear material (which requires hundreds of cycle to obtain the plastic strain), only simple 2D results are mentioned.

For non-linear materials, the optimization problem can be stated as

\[
\min_{x,u} c(x) = \int f^T du \tag{6.9}
\]

where \( R = f - f_{\text{internal}} \),

\[ x_i \in \{0,1\}, \quad i = 1,2, \ldots, N \]

\[ \sum_{i=1}^{N} x_i V_i - \bar{V} \leq 0 \]

The only difference in these equation compared to other equations is that the linear relation between force and displacement does not exist. Instead, the equilibrium procedure is solved through an iterative scheme to minimize the residual force \( R \) between externally applied force \( f \) and the internal response of load carried by elements (Yu 2006). Moreover, the sensitivity number of elements can be stated through derivation of the stiffness with respect to the design variable and can be expressed as

\[
\alpha_i = \frac{\partial c}{\partial x_i} \tag{6.10}
\]

With application of adjoint method, the result of this derivation becomes

\[
\alpha_i = \int f_{\text{internal}}^T du_i \tag{6.11}
\]

One should consider that the above sensitivity number includes both elastic and plastic strain energies.

As explained in section 3.3.2, Mohr-Coulomb yield criterion is one of the most common material models for geomaterials and the strength parameters for different materials are easily accessible. For this model, the assigned material properties are
Young’s Modulus \( E=60 \text{ GPa} \), Poisson’s ratio \( v=0.3 \), cohesion \( c=270 \text{ kPa} \) and internal friction angle \( \varphi=43^\circ \).

A vertical force with the magnitude of 2.7 MPa was applied on the pillar, the parameters for optimization were \( ER=0.01 \) and \( R_f=2.5 \). The initial model (with the forces and boundary conditions) and the results of optimization are illustrated in figure 6.15. It should be noted that only the pillar body was designable area, so the optimization process did not affect other areas.

One of the most severe negative critics about application of optimization in non-linear medium is that some of the elements that are located in the plastic zone are
removed causing destabilization of the medium. Therefore, when the procedure advances, firstly the external low pressurized elements are removed and then the element on the plastic zone, as depicted in figure 6.16, get removed. Thus further pressure on the remaining elements is exerted and many of them undergo plastic deformation resulting in large deformation of the shape and abortion of the finite element procedure after a few optimization iteration. Therefore, this method can be said to seek the stiffest shape, not the most stable one.

Figure 6.16 The optimized pillar with plastic zone coloured with warm colours.

6.6 Strain-based reinforcement optimization
In geotechnics, failure happens when the material is not capable of resisting the stresses and thus deforms permanently leading to large displacement of the shape. Generally, the failure occurs progressively starting at the edges and toes of the exposed surfaces and then it continues gradually by finding its path into the
material mass. The deformation of the whole shape is the results of accumulation of strain in the elements.

Currently, the evaluation of the failure is based on the numerical analysis like FEM, DEM, BEM and DFM. Since all other sections associated the FEM as the main numerical analysis of the optimization method, here, I will consider how FEM performs the analysis; the reasons for its failure to bear results and why strain based optimization will help improve its convergence and stability of the shape.

Non-linear FEM solves the problems through stepwise manner in which a portion of the total load is applied to the mesh of elements and the internal force respond is calculated to keep the domain in a static equilibrium. If the calculated respond does not have considerably lower value compared to the applied load, the step is deemed to be converged. Otherwise, in an iterative approach, Newton-Raphson solver is deployed to find new displacement in the elements to keep the respond of the material within accepted tolerance. If the solver could find any respond that satisfies the equilibrium, the displacement is saved and higher portion of load is applied. When total portion of the load is applied and all the responds of the medium are captured, the non-linear FEM is said to be converged.

In non-linear finite element analysis, distortion of the elements leads to the analysis divergence. This phenomenon can be explained through the fact that, when elements are extensively distorted, their Jacobians (explained in section 2.2.2) might become zero or the surface of the element becomes negative which requires many numerical attempts to confront this problem. Also, it is worth mentioning that highly strained elements have lower stress respond compared to the elements if they were in their elastic state, therefore their contribution to the internal force is lower.

Thus to stabilize the geotechnical problems, it is needed to consider strain as one of critical components for the optimization. Almost all other researchers considered compliance based optimizations, however, for non-linear problems, strain should be utilized. When considering Soft-Kill optimization where elements are not completely removed and the inefficient elements are replaced with softer material, these softer elements undergo large deformation and since compliance based optimization only considers energy as its criterion for the optimization, the
deformation in the elements which might be the source for the numerical instability are not regarded sufficiently.

### 6.6.1 The procedure to perform strain based optimization

Unlike compliance based optimization process, the strain based optimization leads to numerical instability and divergence of the optimization procedure. Through experiments, it has been approved that filtering is an essential part of the optimization process and either compliance or strain should be filtered to overcome local minimums within the outputs.

When an element becomes reinforced, the total compliance in that element increases. As was shown in section (3.2.2), the compliance based optimization utilizes the sensitivity of the strain energy as the criterion which can be expressed as

\[ \alpha_i = u_i^T \frac{\partial K_i}{\partial x_i} u_i, \quad i = 1, 2, ..., n \]  

(6.12)

It is seen that the sensitivity is proportional to the stiffness matrix of that element and using stronger material leads to an increase in the sensitivity of the elements. However, when performing strain based optimization, one may not use the strain as the criterion because the strain for the reinforced material has lower value compared to the original element. Thus, when the objective is to reinforce highly strained elements, the procedure will lead to divergence.

The strain distribution pattern shows that the reinforced elements experience lower strain (due to their stiffness) and when their strain value becomes filtered, the neighbouring element of the reinforced elements get lower value, thus these neighbouring elements have lower chance of being reinforced and it can be said that strain based reinforced elements repel other reinforced elements and the reinforced material gets a scattered shape.

Therefore, there is a need to manipulate data and obtain a value based on the strain that satisfies the numerical needs for the optimization.
With some assumptions, one may develop a criterion which measures the changes in the strain during the replacement of the elements from the original phase to the reinforced phase and vice versa.

If it is assumed that the pressure distribution does not undergo severe changes during the optimization, one may suppose that

\[ \sum d\sigma = \sum D^{ep} d\varepsilon^{ep} \]  \hspace{1cm} (6.13)

which means total stress changes during the non-linear FEM would be equal to the non-linear constitutive model for that element times the total strain. The total strain is composed of two components which are elastic and plastic strain as shown in the following equation

\[ d\varepsilon^{ep} = d\varepsilon^e + d\varepsilon^p \]  \hspace{1cm} (6.14)

The elastic component can be calculated through linear Hook’s law equations and the plastic component is obtained through the following equation

\[ d\varepsilon^p = d\lambda \left( \frac{\partial g}{\partial \sigma} \right) \]  \hspace{1cm} (6.15)

where \( d\lambda \) is a multiplier and \( g \) is the plastic potential function. By considering Drucker-Prager consistency rule, the elasto-plastic constitutive matrix can be written as

\[ D^{ep} = \frac{D^e - D^e \left( \frac{\partial f}{\partial \sigma} \right) \left( \frac{\partial g}{\partial \sigma} \right)^T D^e}{\left( \frac{\partial f}{\partial \sigma} \right)^T D^e \left( \frac{\partial g}{\partial \sigma} \right)} \]  \hspace{1cm} (6.16)

with \( f \) as the yield function of the material, \( D^e \) as the elastic constitutive model (Hook’s law), \( \left( \frac{\partial f}{\partial \sigma} \right) \) and \( \left( \frac{\partial g}{\partial \sigma} \right) \) as the derivative of the yield and plastic potential functions. For the Mohr-Coulomb material, these functions are

\[ f = (\sigma_1 - \sigma_3) - (\sigma_1 + \sigma_3)\sin\varphi - 2ccos\varphi = 0 \]  \hspace{1cm} (6.17)

\[ g = (\sigma_1 - \sigma_3) - (\sigma_1 + \sigma_3)\sin\psi - 2ccos\psi = 0 \]
with \( \sigma_1 \) and \( \sigma_3 \) denoting maximum and minimum principal stresses respectively (compressive stresses are defined as positive). \( c, \varphi \) and \( \psi \) are cohesion, internal friction angle and dilation angle respectively. Also \( f \) and \( g \) are the yield and plastic potential functions. The value of \( f < 0 \) represents material in its elastic behaviour domain and \( f = 0 \) is the yield surface and any positive value for the yield function is not accepted. Also \( g \) is used to define the rate and direction of strain changes when the material is yielding.

### 6.6.2 Numerical implementation

Having limited changes during the optimization process, it can be assumed that the stress distribution remains constant in the elemental level. Thus one can conclude

\[
D_2^{ep} \varepsilon_2 = D_1^{ep} \varepsilon_1
\]

(6.18)

Obtaining the sensitivity of the strain requires two cases to be noticed.

Case 1: The element is made of the original material \( (x_{el} = 0) \), then changes of strain in the element becomes

\[
\alpha = \left( \frac{D_1^{ep}}{D_2^{ep}} \right) \varepsilon_1 - \varepsilon_1
\]

(6.19)

It should be noted when either of the states (original or reinforced) are in their elastic region \( (f_1(\sigma) < 0 \) or \( f_2(\sigma) < 0 \) elastic tangent modulus should be substituted instead of elasto-plastic tangent modulus.

In the Equation 6.18 and 6.19 both of non-linear constitutive models are singular matrices and cannot be placed as denominator. Thus by assuming that the volumetric strain remains constant (the dilation angle for both materials are the same), only shear strain (which is responsible for the deformation of the elements) is regarded. Therefore, in the tangent moduli, the components corresponding to the shear are extracted.

Case 2: The element is composed of the reinforced material \( (x_{el} = 1) \), then for changes of strain, one will have
\[ \alpha = \left( \varepsilon_2 - \frac{D_2^{ep}}{D_2^{ep}} \varepsilon_2 \right) \]  

(6.20)

The same rule mentioned earlier is applied here as well, which is in case of elastic behaviour \( f_1(\sigma) < 0 \) or \( f_2(\sigma) < 0 \), elastic tangent modulus should be deployed.

### 6.6.3 Geotechnical modelling

Generally, geomaterials (rock and soil) show non-linear, inhomogeneous and anisotropic behaviour. While many of empirical assessment of the geotechnical problems considered these materials to be isotropic, homogeneous and linear-elastic, but further understanding and computation power led to expansion of the analysis to adequately consider most aspects of the materials and structures. In this study, due to the broad application and simplicity, Mohr-Coulomb constitutive model has been deployed to adequately represent both original and reinforced materials.

The slope is considered to be straight and long to assure the plane strain assumption. The slope, shown in Figure 6.17, is composed of the homogenous soil and has a height of 7 m and slope angle equal to 54°. The domain is discretized to 7620 quadrilateral elements with mesh refinement around the slope to investigate the behaviour of the failure more precisely. The design has been chosen wide enough to ensure that the boundary conditions do not affect the topology optimization design and stress redistribution tangibly. The applied load arises from the weight of the material in the slope. Nodes on the bottom line are completely fixed while vertical side nodes are only restrained in horizontal direction.

The process of optimization requires the finite element analysis to assess elements respond. The slope stability and strain distribution is carried out by finite element package ABAQUS (version 6.11).
6.6.4 Example and discussion

To demonstrate the proposed method for optimization, an example of slope stability problem is discussed and the developed BESO is deployed. The slope shown in Figure 6.17 is deliberated in the following example. A slope with homogenous and isotropic material is considered and gradual increase in the reinforcement decreases the slope deformation. The material properties of soil and reinforcement are shown in Table 6.1.

Table 6.1 Mechanical properties of the applied materials in the slope.

<table>
<thead>
<tr>
<th>Material</th>
<th>Original soil</th>
<th>Reinforced soil</th>
</tr>
</thead>
<tbody>
<tr>
<td>Young’s Modulus (MPa)</td>
<td>300</td>
<td>900</td>
</tr>
<tr>
<td>Poisson’s Ratio</td>
<td>0.27</td>
<td>0.27</td>
</tr>
<tr>
<td>Cohesion (KPa)</td>
<td>78</td>
<td>250</td>
</tr>
<tr>
<td>Internal friction angle (φ)</td>
<td>22°</td>
<td>42°</td>
</tr>
<tr>
<td>Dilation angle (ψ)</td>
<td>8°</td>
<td>8°</td>
</tr>
<tr>
<td>Material density (Kg/m³)</td>
<td>1900</td>
<td>1900</td>
</tr>
</tbody>
</table>

The optimization procedure commences with reinforcing the elements, located near the toe of the slope where slippage is initiated (Figure 6.18). It should be noted that, reinforcing the slope by stronger material will cause the slippage surface to relocate and find another path, thus when a high Admission Ratio is chosen, the
optimization method will experience fundamental changes in the shape and therefore divergence of the optimization process happens. However by adopting the AR as 5% of the current reinforced volume, the change is reduced. The step by step iterations are illustrated in Figure 6.19. As it can be understood in Figure 6.19b, the reinforced section seems a barrier at toe of the slope preventing formation of highly probable slippage surfaces. As the optimization proceeds, the barrier of reinforced material extends to cover the surface to ban formation of slippage (Figure 6.19c). Subsequently when the developed retaining wall has got enough thickness to catch easy slippage surfaces, the optimization is encountered to the total rotation of the slope face on a deep circular surface shown in Figure 6.19d (analysed strain distribution). To deal with this concern, the optimization pursues to restrain the rotation by formation of toe-down to increase the shear between the sheath and the original material and thus reduce the rotation (Figure 6.19e). By further admission of the reinforced material and swapping of highly strained elements with lower strained elements, lastly, at the iteration 122 the reinforced slope becomes stable.

Upon reaching final iterations, the down tip of the membrane is deep enough to stop rotation of the whole domain. This layout can be explained through the idea that any chance of rotation will be banned by the stiff membrane and the force applied to it by the material beneath the slope and also on the top of the reinforced membrane. Figure 6.20 displays that by addition of the reinforcement to the slope, the total strain of the slope is abridged and the proceeded slope approaches its stable shape, then the optimization stops which indirectly shows that the stability has been increased to desired amount.

Likewise, texts on the topic of dam constructions and slope stability usually recommend extension of the retaining wall where reinforced material is embedded beneath the slope toe to intensify frictional forces, stop lateral movements, deep global slope rotation and toe protection.
Figure 6.18 Strain distribution of the initial unreinforced slope.

Figure 6.19a Optimized shape in the iteration 1 with 8 reinforced elements.
Figure 6.19b Optimized shape in the iteration 15 with 64 reinforced elements.

Figure 6.19c Optimized shape in the iteration 30 with 124 reinforced elements.
Figure 6.19d Optimized shape in the iteration 60 with 244 reinforced elements.

Figure 6.19e Optimized shape in the iteration 90 with 364 reinforced elements.
Figure 6.19f Optimized shape in the iteration 120 with 484 reinforced elements.

Figure 6.19 Step by step optimization procedure showing reinforced parts and strain distribution.

Figure 6.20 Total maximum shear strain changes during the optimization procedure.

6.7 Concluding remarks

This chapter dealt with the application of the developed optimization method through some numerical examples. Firstly, application of the BESO was explored in the underground environment and then the usage was extended to incorporate the non-linearity of the materials. Also a new criterion was chosen to optimize the
slope considering the stability and an example was presented to verify the effectiveness of the offered approach.

In the first example, the optimization method with different parameters was tested on a short cantilever beam. The results were compared and the optimum value for the filtering radius was obtained. Also optimization of shape of a linear elastic tunnel was discussed and the results matched empirical recommendation for the tunnel design. The optimization of an underground tunnel junction was argued and the obtained output reflects the best shape to withstand the major stress acting of the tunnel.

Moreover, the BESO was applied to a problem with non-linear material. It was shown that the shape optimization could result in instability of the shape. Thus a new method considering stability was developed which deployed maximum shear strain as its criterion for the optimization. The optimized reinforcement shape displays an integrated membrane which matches literature regarding slope stability and dam construction recommendation.
Chapter 7

Conclusions

Through this thesis, the author has tried to bring on application of state-of-art shape and reinforcement optimization in the geotechnical problems. Here, attempts are made to explain the procedure of optimization and expand the usage of shape and reinforcement optimization to an underground environment. While usage of the optimization method has become broadly applied in various fields, only few researchers sought pathways to bring this idea to the underground excavation. Through this thesis, by introducing optimization in 3-dimensions and plasticity, further attention to explanation and application of this method was given.

As the first step, a platform for optimization was formed. Through explanation of the procedure of the finite element analysis, a general understanding of the process was made. By focusing on a single element, the governing equations were studied and then further details regarding the shape and dimensions were added. Also, algorithm for calculation for assemblies of elements combined with calculation of displacement, stress and energy were mentioned.

Additionally, optimization was introduced with explanation of its backgrounds through literature review. The algorithm for these optimization methods was reviewed and some modifications to obtain the appropriate topology optimization technique were developed. Moreover, instabilities in the procedure were addressed and ways to remedy them were brought forward and also derivation of objective functions, the process of element switching and technique of shape optimization and reinforcement optimization were discussed to some extent.

Chapter 4 was devoted to presentation of the behaviour of geomaterials and mechanics of underground domain. Firstly, the applied pressures and pressure distribution were described, then mechanical responds of rock masses were briefly discussed staring with intact rock behaviour and then including other complexities in the rock masses like cracks and water effects, also typical mechanical properties of soil were mentioned. Finally displacement due to excavation in a tunnel was surveyed and the effect of support installation was considered.
By means of numerous examples, properties of pillars and excavations in 2D and 3D were investigated. Initially the elastic models were focused then the application was extended to non-linear material with Mohr-Coulomb criteria. Because of requirement of hundreds of finite element analysis in each iteration, only 2D results were presented and more complex simulations in 3D are still due to investigation.

A complete detailed explanation of procedure and codes required to perform optimization were discussed in chapter 5. Here, the data was imported from ABAQUS, and then material properties (this chapter only considered linear elastic materials) and integration parameters were assigned. The procedure continued with introduction of forces and boundary conditions and calculation of the effect of these forces on each node. Then the optimization process started with defining parameters for optimization and the cycle of optimization commenced. The process included performing the finite element (explanation about calculation of stiffness matrix and displacement were given), the technique of filtering sensitivity, finding elements which are located on the edges (for purpose of nibbling), elements on newly formed faces, the procedure of switching element were considered. This section ended with the description of convergence parameters, printing and plotting the results.

Finally, examples of linear and non-linear material were considered. After checking the example with linear material in 2 and 3 dimensions, a new approach to consider stability was sought. Through this section, the BESO method application was broaden to incorporate some measures to overcome stability problems. The proposed extension of the BESO method confirms some of the empirical application of reinforcement in the slope stabilization and dam construction manuals. The author believes that the studied slope stability analysis of this research is novel and will have significant impact on minimization of the applied reinforcement and costs, because stability is the main concern of geotechnical engineers.

One should mention that example and applications discussed in this thesis are only simple cases while in nature complexity and uncertainty in the underground environment have a long way to find their place in the optimization process. However, the solved examples in this thesis prove that the provided method and
coding can actually improve the stability and shape of design in the geotechnical problems. While one might argue that in plastic materials, the respond of geomaterials are dependent to the loading path and the Bi-Directional structural optimization cannot be appropriate for this environment, one can assume that whole the sequence of optimization and the excavation is performed at once and does not interfere with material load dependence respond (non-linear behaviour).

As a comment for interested readers, the author believes that other material models, namely Hoek and Brown criterion, inclusion of crack faces with plastic slippage and application of external support can also be some of the interesting subjects for the optimization in geotechnical problems. Regarding other supporting systems in the underground and slope stability problems, external supports can be an interesting subject for the optimization. Also in the discrete environment (imagine a system of sliding rock slabs on each other), one might need to find the minimum amount of concrete to make a tunnel stable. Thus instead of using finite elements, Discrete Fracture Network (DFN) or other discontinuous method should be applied and the optimization method be expanded to cover discrete methods. There is a hope that the research provided in this thesis can be applied in geotechnical problems and the method of optimization becomes fully applicable in underground designing procedure.
References


