A FRAMEWORK FOR AUTOMATIC MODELING OF UNDERGROUND EXCAVATIONS IN HOMOGENEOUS ROCK MASS

By

Mohammad Hazegh Fetratjoo

A Thesis in the Department of

Building, Civil and Environmental Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy at

Concordia University
Montreal, Quebec, Canada

September 2011

CONCORDIA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Mohammad Hazegh Fetratjoo

Entitled: A framework for automatic modeling of underground excavations in homogeneous rock mass

and submitted in partial fulfillment of the requirements for the degree of

PhD

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

Chair

Dr. Meguid External Examiner

Dr. Dolatabadi External to Program

Dr. Bagchi Examiner

Dr. Haghighat Examiner

Dr. Zsaki Thesis Supervisor

Approved by

Chair of Department or Graduate Program Director

Dean of Faculty

ABSTRACT

A framework for automatic modeling of underground excavations in homogeneous rock mass

Mohammad Hazegh Fetratjoo, PhD
Concordia University, 2011

Determining the optimum excavation sequence in mining or civil engineering requires using stress analysis methods to repeatedly solve large models. Time consuming preparation of the model and lengthy computations, often measured in days, can have major impacts on successful ongoing operation of an underground mine, where stope failures can cost millions of dollars and perhaps result in closure of the mine. Widespread acceptance of new tunneling methods such as NATM which depend heavily on numerical stress analysis tools and the fact that the effects of excavation at the face of the tunnel are distinctively three dimensional necessitates the use of 3D numerical analysis of these problems.

A framework was developed to facilitate efficient modeling of underground excavations and to create an optimal 3D mesh by reducing the number of surface and volume elements while keeping the result of stress analysis accurate enough at the *region of interest*, where a solution is sought. Fewer surface and volume elements means fewer *degrees of freedom* in the numerical model. The reduction in number of degrees of freedom directly translates to savings in computational time and resources. The mesh refinement algorithm is driven by a set of criteria that are functions of distance and visibility of points from the *region of interest* and the framework can be easily extended by adding new types of criteria. A software application was developed to realize the proposed framework and it was applied to a number of mining and civil engineering problems to investigate the applicability, accuracy and efficiency of the framework. The optimized mesh produced by the framework reduced the *time to solution* significantly and the accuracy of the results obtained from the optimized mesh is comparable to the accuracy of the input data for mining engineering problems.

ACKNOWLEDGMENT

I would like to thank my supervisor, Prof. Attila M. Zsaki, for his foresight and help, without which this research work would not have been possible. I am especially grateful to Prof. Kinh H. Ha for his wisdom and all his support during my graduate studies. A special thanks to Prof. Hormoz B. Poorooshasb for his guidance and help along the way.

I would also like to express my love and gratitude to my parents and beloved family for their understanding, endless love and support throughout the duration of my studies. And finally to my brother, Reza, who provided the necessary encouragement and support to get it all done, I really appreciate it.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
Introduction

1.1 Introduction

Underground excavations and tunnels are constructed under the surface of the earth in a rock mass for different purposes. Tunnels improve transportation by moving traffic underground and improving the quality of life above ground. Tunnels also are used for extraction of minerals from underground as part of a mine infrastructure. Other usages for underground excavations are storage, power and water treatment plants, civil defense and other activities. The use of underground excavations often becomes a necessity when there is need for safe operation and environmental protection [1]. However, tunneling projects are expensive and risky and successful execution of these projects requires a high level of knowledge and technical skills in design and construction.

In 1960's a new improved method of tunneling was introduced by Ladislaus von Rabcewicz, Leopold Müller and Franz Pacher [2] which is known as the *New Austrian Tunneling Method* (NATM). In this method load carrying capacity of the ground around the excavation is mobilized to stabilize the excavation. This means that the ground around the excavation forms a load carrying ring and only a thin layer of shotcrete is enough for the support (see section A.6 in appendix A). Today, NATM is widely accepted as a standard tunneling method and is being used in tunneling projects around the world. Modern tunneling techniques such as NATM depend heavily on computer based simulation methods for numerical stress analysis and designing of the underground excavations [3].

In numerical stress analysis of underground excavations, under certain conditions (e.g. plane strain or plane stress fields), one can reduce the dimensions of a 3D problem by one and simplify the solution significantly by solving only a 2D problem. Solution of a 2D problem is easier and faster but has its limitations. This assumption is valid only when the tunnel is relatively long and straight and the *Region Of Interest* (ROI), where a solution to the problem is sought, is far enough

from the edges of the model and the excavation face, which is not the case in most real world tunneling practices.

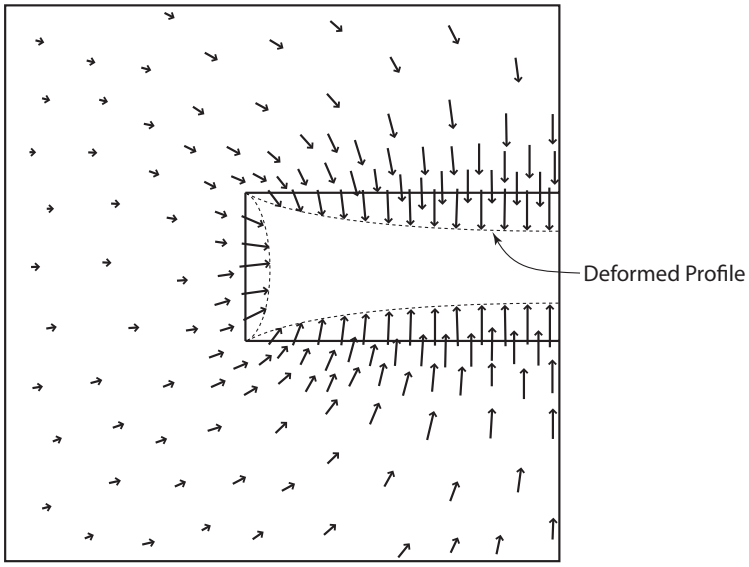

Figure 1.1: Vertical section through a three-dimensional finite element model of the failure and deformation of the rock mass surrounding the face of an advancing circular tunnel [4].

In tunnel construction it is essential to ensure that the tunnel is stable at a tunnel face where the excavation is in progress. Therefore one of the regions of interest in the analysis of underground excavations is the vicinity of a tunnel face. Figure 1.1 shows the results of a three-dimensional finite element analysis of the deformation of the rock mass surrounding a circular tunnel advancing through a weak rock mass subjected to hydro-static stress [4]. The plot shows displacement vectors in the rock mass as well as the shape of the deformed tunnel profile. Figure 1.2 gives a graphical summary of the most important features of this analysis. It is clearly seen that the tunnel end-effects at the tunnel face are distinctively three-dimensional, therefore a 3D analysis of the problem domain is necessary.

But creating 3D geometric models of large scale mining problems, meshing these 3D geometries and performing numerical analysis on these models is very time consuming and challenging. The reason is that the amount of details in these problems is often staggering and if the whole domain with all the details is to be modeled the requirements for the computational resources will exceed resources that

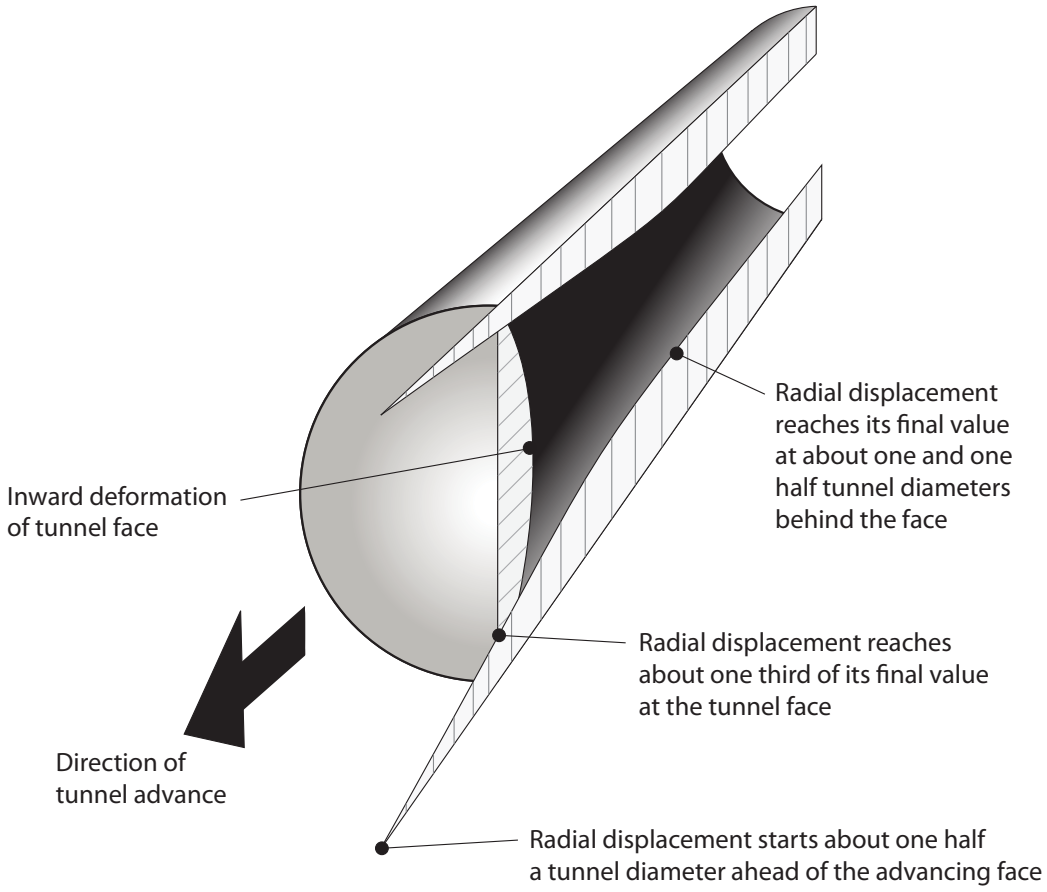

Figure 1.2: Pattern of deformation in the rock mass surrounding an advancing tunnel [4].

are usually available in the engineering firms. Furthermore, to use state of the art constitutive models for the rock mass and to carry out non-linear analysis of the problem will require repetitive numerical solution of the model. These requirements all add up to the complexity of the problem.

1.2 New Austrian Tunneling Method (NATM)

The NATM is defined as a tunneling method where the surrounding rock mass formations of an excavation form a bearing ring that acts as a support structure. In an article by Brown [5] it is noted that NATM can refer to both a design philosophy and a construction method. The characteristics of NATM as a design philosophy are: (a) the strength of the rock mass around a tunnel is mobilized to the maximum

extent possible by allowing controlled deformation of the rock mass, (b) to ensure safe tunneling operations, initial support is designed according to ground conditions to have enough load bearing and deformation capacity and support installation is timed and executed according to ground deformations and (c) the ability to monitor deformation of the initial support system during construction is crucial to success of the method therefore it is important to choose the right instrumentation during the design process. NATM as a construction method has the following characteristics: (a) it allows for sequential excavations that can be varied at each sequence, (b) the initial ground support is usually a layer of shotcrete combined with reinforcements such as fiber, welded wire-mesh and steel arches and (c) the permanent support is provided usually as cast in place concrete lining.

NATM is widely accepted and used worldwide but there have been several unfortunate collapses and stability issues where the method was implemented. The most famous of these incidents is perhaps the Heathrow Airport collapse in October 1994. These incidents triggered a thorough review of the NATM by the British Health and Safety Executive and resulted in a report that identified the heading collapse as the main cause of the failures [6]. As mentioned earlier, the heading is one of the regions that is of great importance in the design and construction of tunnels.

Several causes contribute to these collapses including but not limited to the unexpected ground conditions, errors in design process (especially underestimating the stress and strain in critical regions of the problem domain), poor management and quality control problems during the construction. While great care must be given to eliminate problems that stem from each and every of these causes, this research aims to alleviate the challenges and errors in the design process by introducing a framework that simplifies creation of the optimized 3D models of underground excavations. The optimized model yields more accurate results in the ROI and cuts the time required for numerical analysis of the model. It is worth noting that here the term optimization does not refer to a mathematical optimization technique, rather it refers to finding a better solution by try and error.

1.3 Numerical Methods in Geomechanics and Tunneling

When designing an underground excavation, there are design objectives that must be met. These objectives may be identified as follows:

- Local and overall stability of the excavation and its support system must be ensured.
- The induced displacements for the excavation being designed and any neighboring excavations, structures and services must be within an acceptable range.

Stress analysis of the underground excavation provides an assessment of these important aspects of the design. If the excavation has a circular or oval section with no irregularities and if there are no nearby structures or tunnels that have significant effects on the stress and strain fields around the excavation, then a continuum mechanics based closed form solution might exists for the problem which will be used to solve it. When there exist no closed form solution, numerical methods are used to solve the problem. Today, the availability of inexpensive sophisticated computer hardware has made it possible for engineering firms to deploy computationally intensive numerical methods to solve problems with complex domains.

In geomechanics, *constitutive models* are used to formulate the behavior of rock mass. There is a large number of publications available on constitutive models. To name just a few of these models we may refer to *elasticity models* (linear and piecewise linear), *hyper-elasticity and hypo-elasticity models*, *plasticity models* and *hypo-plasticity models*. Having the constitutive model of the rock mass, the problem domain is formulated by partial differential equations (PDEs). Numerical methods approximate the solution of a linear or non-linear PDE by replacing the continuous system with a finite number of coupled linear or non-linear algebraic equations. This process of *discretization* associates a variable with each of a finite number of points, called *nodes*, in the problem domain. Depending on whether a scalar, vector or tensor quantity is being calculated at nodes, there might be one or more *degrees of freedom* (DOF) at each node.

Some of the numerical methods used in geomechanics to solve boundary value problems are the *finite element method* (FEM), the *boundary element method* (BEM), the *finite difference method* (FDM) and the *discrete element method* (DEM). Elaborative discussions of each of these methods may be found in textbooks (e.g., Zienkiewicz, 1967 [7]; Desai and Abel, 1972 [8]; Britto and Gunn, 1987 [9]; Smith and Griffiths, 1988 [10]; Beer and Watson, 1992 [11]; Potts and Zdravkovic, 1999 [12, 13]). However, as mentioned earlier, to use the FEM, BEM, FDM and DEM methods, one must consider the entire problem domain, break it up into a finite number of discretized sub-regions or elements.

After discretization, the governing equations of the problem are applied separately and approximately within each of the elements, translating the governing differential equations into matrix equations for each element. Compatibility, equilibrium and the boundary conditions are enforced at the interfaces between elements and at the boundaries of the problem. The outcome of this process is a system of equations. The solution to this system of equations is the value of the sought after quantity at nodal points. The result at any point within problem domain can then be approximated based on the available results at the nodes. The number of nodes (and hence the number of DOF) directly affects the accuracy of the numerical model. Usually the larger the number of DOF, the better the accuracy of the result will be. On the other hand, using a large number of DOF results in a large system of equations that requires more time and computational resources to get solved. Because the number of DOF in a practical mining problem can be very large, it is desirable to keep the number of DOF at a minimum in the regions of the model that have less influence on the accuracy of the solution where the results are needed. In this research a framework is proposed that addresses this issue.

1.4 Time to Solution

The design process in engineering is usually an iterative process and consists of three distinct phases: *pre-processing*, *numerical solution* and *post-processing* (see figure 1.3). One starts the design with a series of assumptions and a discrete model is created in the pre-processing phase, based on these assumptions. A numerical

method is used to solve the discrete model. In the post-processing phase the results are compared to the design objectives. If current assumptions do not satisfy the design objectives, new assumptions are made based on previous assumptions and a new round of analysis starts. This loop continues until all assumptions satisfy the design objectives.

Time to solution is defined as the cumulative time required for each of these phases in all iterations that takes to reach the design goals.

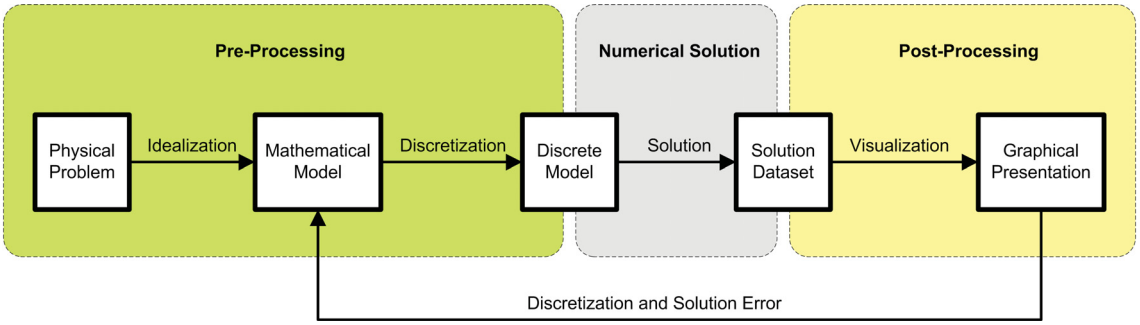

Figure 1.3: Iterative design process

The first step in modeling a physical phenomenon is to idealize it. In the *idealization* process a simplified version of the real problem is created and a mathematical formulation is developed that describes the problem domain. To solve the governing equations using numerical methods, a discretized model (i.e. mesh) of the domain is required. Preparing the mesh involves defining geometry of the excavations, geological properties of the rock mass (e.g. modulus of elasticity and Poisson's ratio) and geological features of the rock mass (e.g. faults and dykes). The sizing of the elements of the mesh has a great impact on the rest of the design process (i.e. numerical solution and post-processing).

A mesh with a large number of elements and degrees of freedom potentially yields more precise results but the numerical analysis can be lengthy and for a large mining model it can take days or weeks [14, 15]. Therefore, it is important to refine the mesh just to the point that the required precision for the problem is satisfied and keep the resource requirements for solving the problem within the available means. The focus of this research is reducing time to solution by simplifying the creation of geometry of tunnels and producing an optimized mesh of the model.

1.5 Objectives and Contributions

Finding the optimum excavation sequence in an underground mining operation involves decision making on different levels: mine planners should plan to meet production requirement (amount of ore extracted per day) and rock engineers must ensure safe operations, overall integrity of mine and stability of each stope during excavation [16]. The rock engineer bases his recommendations on studying the behavior of rock mass around excavation by performing some form of stress analysis in the ROI while the mining operation is in progress. Because of the lengthy calculations and the short time frame for decision making, the results of stress analysis often can only confirm the stability of excavation after mining operation has taken place [14].

The central focus of this research is to reduce *time to solution* and make use of existing computational resources in the mining and civil engineering firms to solve large scale problems. The goal is to reduce time to solution by automation and simplification of the process related to preparing the geometry of the 3D models for underground excavations and by creating optimized meshes. These optimized meshes should produce accurate results in the vicinity of the ROI while minimize the time required for numerical stress analysis. A review of the existing tools (see section 2.7) shows there is need for a specialized framework to address the requirements of this research.

A framework is proposed to address these issues. The framework contributes to reduction of time to solution at two levels: (a) it facilitates creation of geometry of tunnels using minimal input data such as tunnel path and tunnel profile for tunnels with regular shapes and by importing existing surface triangulation for existing excavations with complicated geometry, and (b) it cuts the time required for numerical analysis by reducing the number of DOF in the mesh while keeping the results accurate enough at the ROI.

The simulation methods used for numerical analysis of underground excavations consist of three distinct phases: pre-processing, numerical solution and post-processing. Achieving a high level of automation among these three phases will greatly enhance the efficiency and usability of the simulation method. The impor-

tance of automation becomes evident, considering the fact that the tunnel engineer often has to respond to unexpected ground conditions on site. Therefore, rapid results from numerical simulation performed on site could serve as a tool which assists critical decisions. In a study on the application of numerical simulation at the tunnel site carried out by Golser and Schubert [17] it was found out that there is great need for pre-processing tools with higher degree of automation and simplicity.

This automation brings along a lot of advantages and opportunities. One possibility is to use a parametric representation of the model for analysis. In a parametric model, one can specify different quantities as parameters that can vary between a minimum and maximum value. For example, the tunnel path can be defined using parametric formulation and then be optimized subject to existing geological features and other constraints.

Another possibility would be the ability to run sensitivity analysis for different parameters that define the model. The classical approach used in engineering design is to calculate the capacity C (strength) of the load bearing element or structure and the demand D (stress or disturbing force). The factor of safety of the structure is defined as $F = \frac{C}{D}$ and failure is assumed to occur when $F < 1$. In this method the design decision is based on only one calculated factor of safety.

An approach which is frequently used to give a more rational assessment of the risks associated with a particular design is to carry out a sensitivity study. This involves a series of calculations in which each significant parameter is varied systematically over its maximum credible range in order to determine its influence upon the factor of safety. It provides a useful means of exploring a range of possibilities and reaching practical decisions on some difficult problems [4].

1.6 Results Obtained from Application of the Framework

To make better predictions and obtain more realistic results from numerical analysis of the problems in rock engineering, non-linear constitutive models were developed and now personal computers have reached a point that they can provide enough computational power to carry out 3D non-linear analysis of large scale problems, if the computational resources are used optimally. To reach this goal, the size

of the model to be numerically solved must be reduced as much as possible while the accuracy of the result is kept at a reasonable range in the vicinity of an ROI where a solution to the problem is sought.

The framework that was developed in this research was applied to a number of mining engineering problems and efficiency and accuracy of it was studied. To reach a certain degree of accuracy, the framework was able to reduce the size of the problem by 14 folds compared to a uniform mesh. The time required for a linear finite element analysis was reduced by an incredible amount of 57 times, from 14.5 minutes to 15 seconds. The considerable improvement is because current personal computers must use *disk swapping* to solve problems that their requirements for the RAM (Random Access Memory) surpasses the available amount of RAM. The framework optimizes the model so that it fits within the available RAM and improves the performance of the finite element solver dramatically.

In certain instances, the size of the problem is so large that even disk swapping can not help and the solver simply fails and refuses to solve the finite element problem. To overcome this issue, the framework can be utilized to reduce the size of these problems so that they fit the specifications of the available hardware and computational resources while the accuracy of the numerical analysis is kept within an acceptable range. Chapter 4 covers detailed discussions about the accuracy, efficiency and applicability of the the framework.

1.7 Outline of the Thesis

The following is a summary of the material covered in each chapter. Chapter 2 contains a literature review of the meshing techniques as well as review of existing meshing tools and related work. The objectives of effective meshing techniques such as accurate modeling of the geometry, mesh gradation and quality of mesh structures are discussed. Meshes are classified as structured and unstructured as well as surface and volume meshes. Afterwards, the most important meshing algorithms in 2D and 3D are reviewed. Delaunay triangulation technique is given special attention as it is the preferred method for meshing in this research. Post-processing of meshes is also discussed. Finally, a survey of notable existing meshing tools is provided and

the advantages and shortcomings of each tool is discussed.

Chapter 3 contains the development of contributions that are made in this research. A framework is developed to facilitate efficient modeling of underground excavations and to create an optimal 3D mesh by reducing the number of surface and volume elements while keeping the result of stress analysis accurate enough at the ROI, where a solution is sought. Fewer surface and volume elements means fewer DOF in the numerical model which directly translates into savings in computational time and resources. The mesh refinement algorithm is driven by a set of criteria that are functions of distance and visibility of points from the ROI and the framework can be easily extended by adding new types of criteria.

Chapter 4 is dedicated to the study of applicability, efficiency and accuracy of the proposed framework. To illustrate the accuracy and efficiency of the framework, it was applied to a few mining engineering problems. The error introduced by optimizing the mesh and the time taken for mesh generation and stress analysis were measured and presented. Finally, chapter 5, provides recommendations for future research to be conducted. It also discusses the areas in which the framework can be improved and the research work is summed up and conclusions are provided.

There are also two appendices that provide complementary information about rock mechanics in general and numerical analysis in rock mechanics which are relevant to subject under study but are not the focus of the research.

Appendix A is a review of rock mechanics and standard tunneling practices. First the development of rock mechanics as a discipline is reviewed. Then some of the basic and main terminology in rock mechanics are given and techniques to collect and present geotechnical data are discussed. At the end, tunneling terminology and different excavation techniques are reviewed. In appendix B different numerical methods like Finite Element Method, Boundary Element Method, Coupled Finite-Boundary Element Method and other methods are reviewed and their advantages and disadvantages in numerical stress analysis of geotechnical design are highlighted. Pre-processing which is the focus of this research is discussed as one of the three important phases in numerical analysis.

CHAPTER 2
Review of Meshing Techniques

2.1 Introduction

Meshing is the process of dividing a geometric entity into smaller parts. It is part of a broader branch in mathematics called *computational geometry*. Different applications of meshing in engineering includes computer animation and numerical solution of partial differential equations (PDEs), the latter being our main focus in this research. The first step, namely the pre-processing phase, in numerical solution of a PDE is to break up the problem domain into smaller sub-domains or *elements*. In this chapter the existing meshing techniques and tools are reviewed.

2.2 General Objectives

2.2.1 Correct Modeling of Geometry

One of the objectives of a meshing framework is to correctly model the geometry of the problem domain. In geomechanics one usually faces problems with complex and possibly curved boundaries. Boundaries can appear in the exterior or interior of the problem domain and are represented by a collection of edges or faces in two or three dimensions respectively. Curved boundaries can be approximated with piecewise linear boundaries. In this research linear boundaries will be considered only.

Exterior boundaries separate the meshed region from the unmeshed region of space. They appear on the surface or in the internal holes of the model. *Interior boundaries* enforce constraints that elements must conform to and may not pierce through them. These boundaries separate regions with different physical properties inside model; for example, zones of rock mass with different geological properties.

2.2.2 Mesh Gradation

Another goal of a meshing framework is to offer control over the size of elements in the mesh. Ideally, this control includes the ability to grade from small to large

elements over a relatively short distance. The reason for this requirement is that element size has two effects on a finite element simulation, namely the *accuracy* and the *speed* of the numerical solution. Small, densely packed elements offer more accuracy than larger, sparsely packed elements; but the computation time required to solve a problem is proportional to the number of elements. Therefore, there is a trade off between speed and accuracy when choosing an element size. Also, the element size required to achieve a given accuracy depends upon the behavior of the physical phenomena being modeled, and may vary throughout the problem domain. If elements of uniform size are used throughout the mesh, one must choose a size small enough to guarantee sufficient accuracy in the most demanding portion of the problem domain, and thereby possibly incur excessively large computational demands. To avoid this pitfall, a mesh generator should offer rapid gradation from small to large sizes.

2.2.3 Mesh Quality

The meshing framework should be capable of producing quality elements. This is one of the most difficult goals that a meshing framework should attain. The quality of elements are measured with *mesh quality metrics*. For the most part, mesh quality metrics are based on geometric criteria. For example, does a given element possess positive volume and a good (i.e. relatively "round") shape? Element volume, aspect ratio, skew, angles, stretching, and orientation are common geometric quality metrics [18, 19]. Figure 2.1 shows an example of a well shaped and a degenerate triangular element.

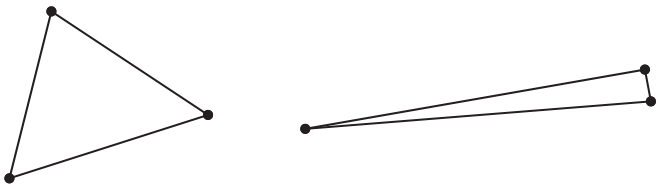

Figure 2.1: Well shaped (left) and degenerate (right) triangles.

2.3 Classification of Meshes

2.3.1 Surface and Volume Mesh

The surface of a model is decomposed using 2D elements. 2D elements are usually either triangular or quadrilateral. The volume of the model is decomposed with tetrahedra or hexahedra. In this research triangles and tetrahedra are used for 2D and 3D meshing respectively. Figure 2.2 shows examples of 2D and 3D meshes.

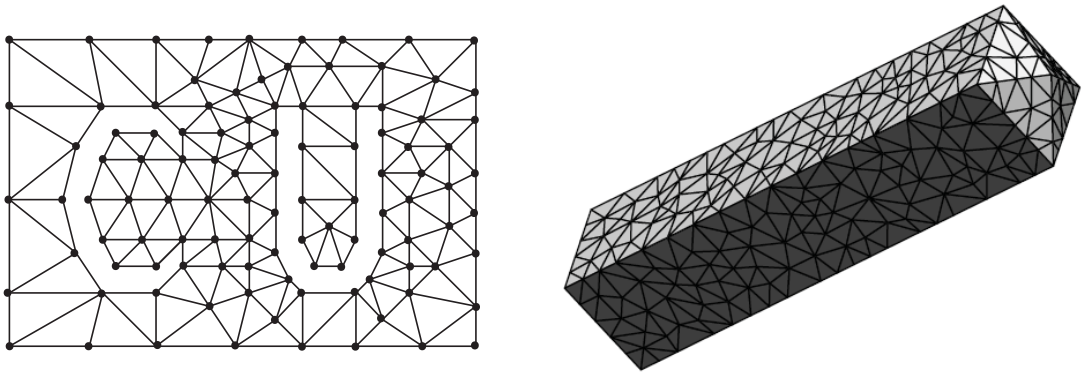

Figure 2.2: Two and three-dimensional meshes. At left, each triangle is an element. At right, each tetrahedron is an element.

2.3.2 Structured versus Unstructured Mesh

Meshes can be categorized as *structured* and *unstructured*. Figure 2.3 shows an example of each. Structured meshes have a uniform topological structure while unstructured meshes lack this uniformity. Another definition could be that in a structured mesh the indexes of the neighbors of any node can be calculated by a simple linear formula. In an unstructured mesh, the only way to know about the index of the neighboring nodes is to store a list of each node's neighbors.

Advantages and disadvantages of each are discussed bellow:

- For the same number of elements and nodes, solution of linear and non-linear system of equations yielded by FEM and BEM is simpler and faster on structured meshes, because of the ease of determining each node's neighbors. On the other hand, unstructured meshes require more storage space and the memory

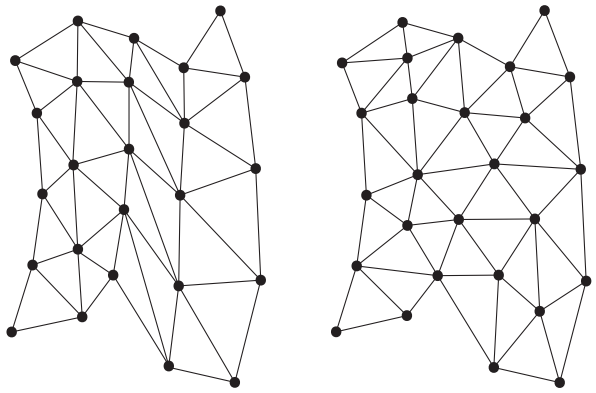

Figure 2.3: Structured (left) and unstructured (right) meshes.

traffic is higher (hence slower) because they need to keep a list of the pointers to each node's neighbors in memory.

- The regularity of the node and element indexes in structured meshes makes it straightforward to apply parallel computing algorithms to them, whereas the complexity in the unstructured meshes necessitates the use of sophisticated partitioning algorithms and parallel unstructured solvers [20].

- Many problems are defined on domains with irregularities that impose lots of restrictions, making it very difficult (and sometimes impossible) to form a structured mesh on them. To create a structured mesh that fully conforms to the exterior and interior boundaries of the domain, one needs a significantly larger number of elements compared to those of an unstructured mesh. Larger number of elements means more memory space requirements and lower speeds. To achieve a solution with the same degree of accuracy throughout the problem domain, one can apply an unstructured mesh that can be flexibly tailored to the physics of the problem resulting in fewer number of elements. Unstructured meshes, far better than structured meshes, can provide multi-scale resolution and conformity to complex geometries.

The disparity between structured and unstructured meshes is of more significance in 3D domains and where the scales of the physical problem vary more [20]. In this research a combination of both structured and unstructured meshes are used.

Individual geometric entities that have a regular and well defined shape are created using a structured mesh. When a geometric entity intersects with another one, an unstructured mesh is used near the area of intersection between the entities.

2.4 Meshing Algorithms in 2D and 3D

Although there is certainly a difference in complexity when moving from 2D to 3D, the algorithms discussed here are for the most part applicable for both 2D and 3D with some restrictions. Triangular and quadrilateral elements are the most common forms of elements used in two dimensional meshing algorithms. Quadrilateral element are better suited for structured mesh (grid) generation but are also used in unstructured meshing. These algorithms are classified under the following major groups Quadtree/Octree, Advancing front and Delaunay triangulation/tetrahedralization and are discussed bellow.

2.4.1 Quadtree/Octree

In this method quadrilaterals containing the geometric model are recursively subdivided until the desired resolution is reached. Figure 2.4 shows the two dimensional quadtree decomposition of a model. Irregular cells are then created where quadrilaterals intersect the boundary, often requiring a significant number of boundary intersection calculations. Quadrilateral elements are generated from both the irregular cells on the boundary and the internal regular cells. The resulting mesh will change as the orientation of the cells in the quadtree structure is changed. To ensure element sizes do not change too dramatically, maximum difference in quadtree subdivision level between adjacent cells can be limited to one. Smoothing and cleanup operations can also be employed to improve element shapes. The disadvantage of this method is that it does not provide flexible control over the size of elements (i.e. grading).

2.4.2 Advancing front

A popular method for triangular mesh generation is the advancing front, or moving front method. Two of the main contributors to this method are Löhner [21,

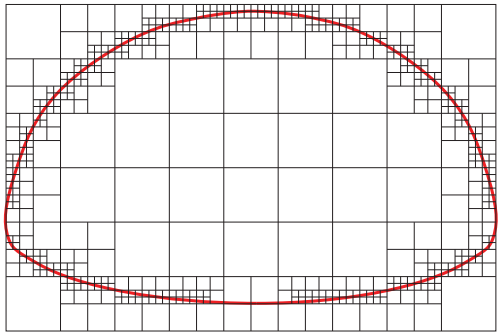

Figure 2.4: Quadtree decomposition of a tunnel cross-section.

22] and Lo [23, 24]. Advancing front methods [25] begin by dividing the boundaries of the mesh into edges (in two dimensions) or triangular faces (in three dimensions). These discretized boundaries form the initial front. Triangles or tetrahedra are generated one at a time, starting from the boundary edges or faces, and moving toward the center of the region being meshed as shown in Figure 2.5. An active front is maintained where new triangles or tetrahedra are formed.

Advancing front methods require a good deal of second-guessing, first to ensure that the initial division of the boundaries is prudent, and second to ensure that when the advancing front of elements meet at the center of the mesh, they are merged together in a manner that does not compromise the quality of the elements. In both cases, a poor choice of element sizes may result in situation where a front of small elements meets a front of large elements, making it impossible to fill the space between with well shaped elements. Advancing front methods typically create astonishingly good triangles or tetrahedra near the boundaries of the mesh, but are much less effective where fronts meet.

A sizing function can be defined in this method to control element sizes. Löhner [21] proposed using a course Delaunay mesh of selected boundary nodes over which the sizing function could be quickly interpolated.

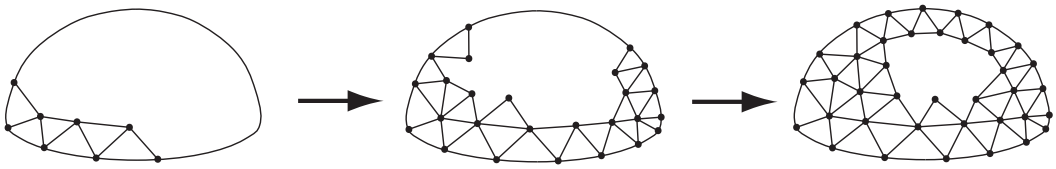

Figure 2.5: Stages in the progression of an advancing front algorithm for triangulation of a tunnel cross-section.

2.4.3 Delaunay Triangulation/Tetrahedralization

The most popular meshing techniques that use triangular elements for meshing the domain follow *Delaunay criteria* to generate a well formed mesh. Delaunay refinement has theoretical guarantees that back up its good performance in practice.

It is not easy to trace who first used Delaunay triangulations for solving PDEs. These ideas have been intensively studied in the engineering community since the mid-1980s, and began to attract interest from the computational geometry community in the early 1990s.

Many of the earliest papers suggest performing vertex placement as a separate step, typically using structured grid techniques, prior to Delaunay triangulation. For instance, Cavendish, Field and Frey [26] generate grids of vertices from cross-sections of a three-dimensional object, then form their Delaunay tetrahedralization. The idea of using the triangulation itself as a guide for vertex placement followed quickly; for instance, Frey [27] removes poor quality elements from a triangulation by inserting new vertices at their circumcenters –the centers of their circumcircles– while maintaining the Delaunay property of the triangulation.

Delaunay triangulation of a vertex set may be unsatisfactory for two reasons: elements of poor quality may appear, and input boundaries may fail to appear. Both problems have been addressed in the literature. The former problem is typically treated by inserting new vertices at the circumcenters [27] or centroids [28] of poor quality elements. It is sometimes also treated with an advancing front approach, discussed briefly in Section 2.4.2.

The problem of the recovery of missing boundaries may be treated in several

ways. These approaches have in common that boundaries may have to be broken up into smaller pieces. For instance, each input segment is divided into a sequence of triangulation edges which is referred to as subsegments, with a vertex inserted at each division point. In three dimensions, each facet of an object to be meshed is divided into triangular faces which is referred to as sub-facets. Vertices of the tetrahedralization lie at the corners of these sub-facets.

In the earliest publications, boundary integrity was assured simply by spacing vertices sufficiently closely together on the boundary prior to forming a triangulation [27], surely an error-prone approach. A better way to ensure the presence of input segments is to first form the triangulation, and then check whether any input segments are missing.

2.4.3.1 Voronoi Diagrams

Given a finite set of points in the plane, the objective is to assign to each point a region of influence in such a way that regions decompose the plane. Let $S \subseteq \mathbb{R}^2$ be a set of points and define the *Voronoi region* of $p \in S$ as the set of points $x \in \mathbb{R}^2$ that are at least as close to p as to any other point in S, i.e.,

$$V_p = \{x \in \mathbb{R}^2 \mid \|x - p\| \leq \|x - q\|,\ \forall q \in S\}.$$

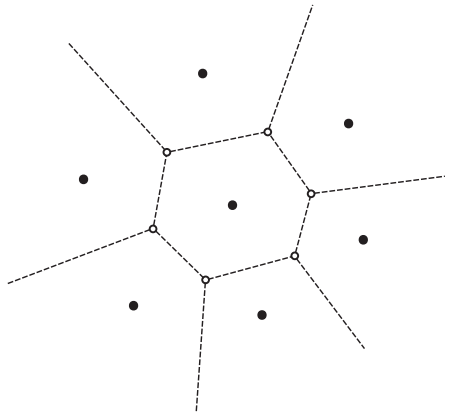

Figure 2.6: n points define the same number of Voronoi regions.

For an illustration of this definition look at Figure 2.6. Now, let the half-plane

of points at least as close to p as to q be $H_{pq} = \{x \in \mathbb{R}^2 \mid \|x - p\| \leq \|x - q\|\}$. The Voronoi region of p, V_p, is the intersection of half-plane H_{pq}, for all $q \in S - \{p\}$. It follows that V_p is a convex polygonal region, possibly unbounded, with at most $n - 1$ edges.

Each point $x \in \mathbb{R}^2$ has at least one nearest point in S, so it lies in at least one Voronoi region. It follows that the Voronoi regions cover the entire plane. Two Voronoi regions line on opposite sides of the perpendicular bisector separating the two generating points. It follows that Voronoi do not share interior points, and if a point x belongs to two Voronoi regions, then it lies on the bisector of the two regions. The Voronoi regions together with their shared edges and vertices form the *Voronoi diagram* of S.

2.4.3.2 Delaunay Triangles

If we connect the points $p, q \in S$ whose Voronoi regions intersect along a common line segment, a *Delaunay edge* is formed. The Delaunay edges decompose the convex hull of S into triangular regions, which are referred to as *Delaunay triangles*. Delaunay triangles are duals of of Voronoi diagrams. See Figure 2.7.

A *triangulation* is a collection of triangles together with their edges and vertices. A triangulation K *triangulates* S if the triangles decompose the convex hull of S and the set of vertices is S. An edge $ab \in K$ is *locally Delaunay* if,

- It belongs to only one triangle and therefore bounds the convex hull, or
- It belongs to two triangles, abc and abd, and d lies outside the circumcircle of abc.

This definition is shown in Figure 2.8. If every edge of K is locally Delaunay, then K is Delaunay triangulation of S.

2.4.4 Meshing by Sweeping

To create a 3D mesh, a surface mesh on a bounded surface is swept through space along a curve. This technique can be generalized to mesh certain classes of volumes by defining *source* and *target* surfaces. Provided that the source and target

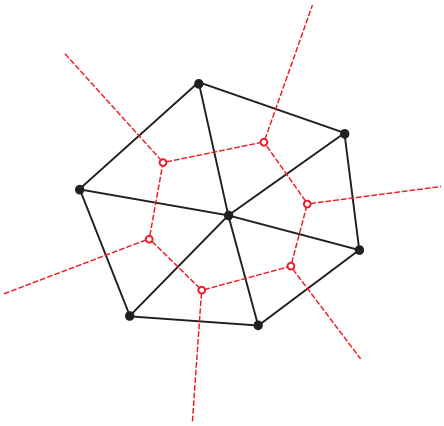

Figure 2.7: The Voronoi edges (dotted) and their dual Delaunay edges (solid).

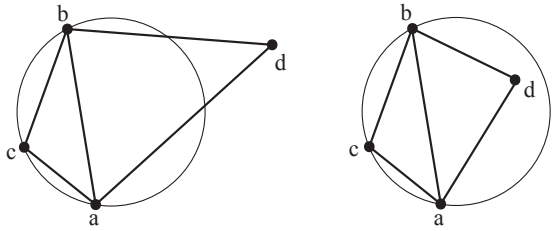

Figure 2.8: To the left *ab* is locally Delaunay and to the right it is not.

surface have similar topology, the elements of the source area can be swept through the volume to generate 3D elements. Different techniques and issues to find internal points along the sweeping path are discussed in [29] and [30].

Blacker [31] generalizes and extends the applicability of sweeping by introducing the *Cooper Tool*. The Cooper tool allows for multiple source and target surfaces while still requiring a single sweep direction. With this tool, the topology is allowed to branch or split along the sweep direction. In addition, the topology of source and target surfaces are not required to be similar. With these requirements relaxed, a greater subset of geometry may be meshed with generally very high quality elements. There is obviously a great potential for using this technique in creating 3D models for tunnels.

2.4.5 3D Surface Meshing

There are different approaches for meshing a 3D curved surface. The following is a review of these methods.

2.4.5.1 Mapping 2D Planar Meshes to 3D Curved Surfaces

NURBS[1] surfaces are widely used in modeling the geometry of objects. Each point on a NURBS surface has an underlying two-dimensional (u, v) representation as well as a global (x, y, z) representation in the Cartesian coordinate system. The algorithms in this method form the mesh in the (u, v) parametric space of the surface and then map the (u, v) coordinates to the global (x, y, z) coordinate.

The drawback to this method is that the elements formed in parametric space may not always form well-shaped elements in three dimensions once mapped to the surface. To address this issue, the following solutions exist:

1. Modify the underlying parametric representation of the surface so there is a reasonable mapping from (u, v) space to (x, y, z) space.

2. Modify the mesh generation algorithm so that stretched or anisotropic elements created in the (u, v) space will map to well-shaped, isotropic elements in (x, y, z) space.

The first method requires that in order to have a good parameterization, the surface derivatives, (D_u, D_v), should not have enormous changes over the domain. An exact arc-length re-parameterization was suggested by Farouki [32]. An approximate arc-length parameterization (i.e. warped parameter space) can be defined by selectively evaluating surface derivatives over the domain and adjusting (u, v) values to hold the magnitude of (D_u, D_v) roughly constant. A warped parametric space can generate reasonable surface meshes for many cases, but there are many problems that the re-parameterization cannot provide an adequate solution. That's why much of the literature on surface meshing focuses on the second solution of forming anisotropic elements in 2D and then mapping them to isotropic elements in 3D.

[1]The word NURBS is an acronym for nonuniform rational B-spline. Non uniform rational B-splines can represent 3D geometry.

In the second method, one can take advantage of the surface derivatives, (D_u, D_v). These are computed from the NURBS surface if the geometry is defined using NURBS. In the case that the geometry is defined using an existing set of elements that should be altered or refined, the surface derivatives are computed using the nodes in the neighborhood of point (u, v). George and Borouchaki [33] use a Delaunay based approach and propose the use of a metric derived from the first fundamental form of the surface. The metric is in the form of a 2×2 matrix and is used to transform vectors and distances in parametric space. In their Delaunay approach, the *empty circle* property for delaunay triangulation, becomes an *empty ellipse* property. Also included with the metric is the option to incorporate element sizing and stretching properties. Similar research is carried out on the subject by Chen and Bishop [34]. Equivalent advancing front surface mesh generation algorithms, which utilize a metric derived from the first fundamental form of the surface are presented independently by Cuilliere [35] and Tristano [36].

2.4.5.2 Direct 3D Surface Meshing

Direct 3D surface mesh generators form elements directly on the geometry regardless of the parametric representation of the underlying geometry. In some cases where a parametric representation is not available or where the surface parameterization is very poor, direct 3D surface mesh generators can be useful. Lau and Lo [37, 38] present an advancing front approach for arbitrary 3D surfaces. In this method surface normals and tangents must be computed in order to compute the direction of the advancing front. In addition, a significant number of surface projections are required to ensure that new nodes remain on the surface. Also of significance is the increased complexity of the intersection calculations required to ensure that triangles on the surface do not overlap.

Dey, Li and Ray [39] proposed a method that recovers the topology from an input polygonal surface and then creates a 3D refined mesh for the recovered topology using Delaunay triangles. The assumption about the input surface is that it actually approximates a smooth surface both point-wise and normal-wise. This means if the given polygonal surface has a very sharp edge (dihedral angle less than

90 degrees), the algorithm may not work.

2.5 Mesh Post-Processing

It rarely happens that a meshing algorithms can directly produce a mesh that is optimal without using any kind of post-processing that improves the quality of the generated elements. The three main categories of mesh improvement include *smoothing*, *clean-up* and *refinement*. Smoothing includes any method that adjusts node locations while maintaining the element connectivity. Clean-up generally refers to any process that changes the element connectivity. Refinement refers to any operation that reduces element size locally.

2.5.1 Smoothing

Smoothing involve some form of iterative process that repositions individual nodes to improve the local quality of the elements. Usually corner nodes of the element are chosen for smoothing but it is also possible to relocate internal nodes of the element. Salem [40] introduced a method providing criteria for repositioning mid-nodes on quadratic elements to improve element quality. This method computes a region surrounding the mid-node known as the *mid-node admissible space* where the mid-node can safely be moved to maintain or improve element quality. There are variety of smoothing techniques that will be discussed below.

2.5.1.1 Averaging Methods

Laplacian smoothing [41] is the simplest and the most straight forward method among other smoothing algorithms. This method relocates the internal nodes of the mesh to the average location of the nodes connected to it. This technique can be used for any element shape with small modifications to the method. The smoothing algorithms will iterate through all the nodes of the mesh several times until they converge (i.e. no nodes move more than a predetermined tolerance). A drawback of Laplacian smoothing is that it can position nodes outside of the boundaries. Similar to Laplacian, there are a variety of other smoothing techniques, which iteratively reposition nodes based on a weighted average of the geometric properties of the

surrounding nodes and elements. Canann [42] provides an overview of some of the common methods in use.

In the constrained Laplacian smoothing a comparison of local element quality is made before and after the proposed move and the node is moved only if element quality is improved. Canaan [42] shows the criteria for the movement of the nodes using this method.

2.5.1.2 Optimization-Based Methods

Rather than relying on heuristic averaging methods, some methods use optimization techniques to improve element quality. Optimization-based smoothing techniques measure the quality of the surrounding elements to a node and attempt to optimize by computing the local gradient of the element quality with respect to the node location. The node is moved in the direction of the increasing gradient until an optimum is reached. Canann [42] and Freitag [43] both present optimization-based smoothing algorithms.

While maintaining that optimization-based smoothing techniques provide superior mesh quality, the computational time involved is generally too excessive to use in standard practice. Canann [42] and Freitag [44] both recommend a combined Laplacian/optimzation-based approach. Laplacian smoothing is done for the majority of the time, reverting to optimization based smoothing only when local element shape metrics drop below a certain threshold.

2.5.2 Clean-up

Like smoothing, there are a wide variety of methods currently employed to improve the quality of the mesh by making local changes to the element connectivities. Cleanup methods generally apply some criteria that must be met in order to perform a local operation. The criteria in general can be defined as: shape improvement or topological improvement.

In addition, cleanup operations are generally not done alone, but are used in conjunction with smoothing. Freitag [45] describes how smoothing and cleanup may be combined to efficiently improve overall element quality.

2.5.2.1 Shape Improvement

For triangle meshes, simple diagonal swaps are often performed. For each interior edge in the triangulation a check can be made to determine at what position the edge would effectively improve the overall or minimum shape metric of its two adjacent triangles. The Delaunay criteria can also be used to determine the position of an edge. For Tetrahedral meshes, Joe [46] presents a series of local transformations that are designed to improve the element quality. These include swapping two adjacent interior tetrahedra sharing the same face for three tetrahedra. Likewise, three tetrahedra can be replaced with two. Other more complex transformations are also defined.

In some applications where mixed element meshes are supported, the element quality of two adjacent triangles may be preferable to a single poor quality quadrilateral. When this is the case, selected quadrilaterals may be split.

In some cases, particularly with curved surfaces, the elements resulting from the mesh generator may deviate significantly from the underlying geometry. For a triangle mesh, edge swaps can be performed based on which local position of the edge will deviate least from the surface. Although not strictly a cleanup operation, local refinement of the mesh may also be considered to capture surface features.

2.5.2.2 Topological Improvement

A common method for improving meshes is to attempt to optimize the number of edges sharing a single node. This is sometimes referred to as node valence or degree. In doing so, it is assumed that the local element shapes will improve. For a triangle mesh there should optimally be 6 edges at a node and four edges at a node surrounded by quads. Whenever there is a node that does not have an ideal valence, the quality of the elements surrounding it will also be less than optimal. Performing local transformations to the elements can improve topology and hence element quality. Several methods have been proposed for improving node valence for both triangle [47] and quadrilateral [48, 49] meshes.

For volumetric meshes, valence optimization becomes more complex. In addition to optimizing the number of edges at a node, the number of faces at an edge

can also be considered. For tetrahedral meshes this can involve a complex series of local transformations. For hexahedral elements, valence optimization is generally not considered tractable. The reason for this is that local modifications to a hex mesh will typically propagate themselves to more than the immediate vicinity. One special case of cleanup in hex meshes used in conjunction with the whisker weaving algorithm is presented by Mitchell [50] .

2.5.3 Refinement

Element refinement procedures are numerous. For our purposes, refinement is defined as any operation performed on the mesh that effectively reduces the local element size. The reduction in size may be required in order to capture a local physical phenomenon, or it may be done simply to improve the local element quality. Some refinement methods in themselves can be considered mesh generation algorithms. Starting with a coarse mesh, a refinement procedure can be applied until the desired nodal density has been achieved. Quite frequently, refinement algorithms are used as part of an adaptive solution process, where the results from a previous solution provide criteria for mesh refinement.

2.5.3.1 Point Insertion

A simple approach to refinement is to insert a single node at the centroid of an existing element, dividing the triangle into three or a tetrahedron into four. This method does not generally provide good quality elements, particularly after several iterations of the scheme. To improve upon the scheme, a Delaunay approach can be used that will delete the local triangles or tetrahedra and connect the node to the triangulation maintaining the Delaunay criterion. Any of the Delaunay point insertion methods discussed previously could effectively be used for refinement. See Figure 2.9.

2.5.3.2 Edge Bisection

Edge bisection involves splitting individual edges in the triangulation. As a result, the two triangles adjacent the edge are split into two. The same concept can be extended to volumetric meshing, any tetrahedron sharing the edge to be split

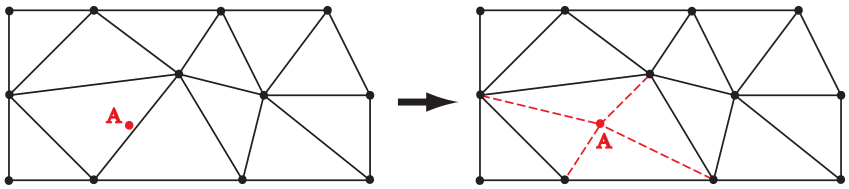

Figure 2.9: Delaunay refinement by point insertion.

must also be split. Rivara [51] proposes criteria for the splitting of edges based on the longest edge of a triangle or tetrahedron.

2.5.3.3 Templates

A template refers to a specific decomposition of the triangle. One example is to decompose a single triangle into four similar triangles by inserting a new node at each of its edges as show in Figure 2.10. The equivalent tetrahedron template would decompose it into eight tetrahedra where each face of the tetrahedron has been decomposed into 4 similar triangles. To maintain a conforming mesh, additional templates can also be defined based on the number of edges that have been split. Staten [52] outlines the various templates needed to locally refine tetrahedra while maintaining a conforming mesh. See Figure 2.10.

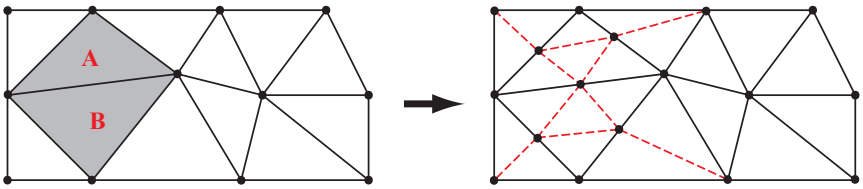

Figure 2.10: Triangles A and B are refined using the template method.

2.6 Previous and Related Work in Mesh Optimization

Because the time to solution increases by complexity of the problem in all engineering disciplines, there have been several initiatives to reduce mesh size in order to reduce time to solution. In structural and mechanical stress analysis, most

of the problems deal with finding a solution for inside of the model and in the entire domain and can be classified as *interior* problems. In rock engineering on the other hand, the solution is sought in an ROI outside the excavations, therefore these problems can be classified as *exterior* problems [53]. For interior problems, the geometry details can affect the results in the domain. As a result, only small features in the model can be simplified or removed [54, 55]. Work has also been done on removing holes and modifying the topology of the so called interior models [56]. For exterior problems, a recent study addresses geometry simplification and optimization of surface meshes by adopting and extending algorithms that originated in computer graphics field. In this study an existing surface mesh is simplified by defining a cost function that drives the mesh simplification process [16].

In geomechanics, it is generally not necessary to compute field quantities everywhere in the problem domain. Rather, the stresses and strains are only sought at specific regions that are close to the stope or regions that interact with other adjacent facilities and substructures [53]. Therefore it is acceptable for the regions that are farther from the ROI to have a coarser mesh. Currently, to optimize the model, human expert knowledge is required to simplify the mesh. The mesh is coarsened in the regions that, based on the expert's estimate, will not affect the field quantities where the results are required. Because this form of mesh optimization requires human intervention, it is a very slow process. Also, the decisions made by the expert are subjective and not based on quantitative measures. This research investigates methods and measures that provide a framework for automating the creation of the model and the mesh by taking into account the region of influence (see section 3.2), principles of continuum mechanics and incorporating other problem specific knowledge.

2.7 A Survey of Existing Meshing Tools

Here a survey of notable exiting meshing tools is provided and their capabilities and limitations are discussed.

2.7.1 ANSYS, ANSYS Inc.

ANSYS [57] is a commercial product with closed source code. It provides finite element solvers for problems in the fields of structural mechanics, explicit dynamics, fluid dynamics and electromagnetics.

ANSYS is capable of producing tetrahedral and hexahedral meshes and provides two types of size functions. The first type of size function is designed to capture the geometry while minimizing the number of elements in the model and is used for mechanical applications. The second type of size function is designed to capture the geometry while maintaining a smooth growth rate between the regions of curvature and/or proximity and is used for fluids dynamics problems [57].

While both of these size functions are useful in reducing the total number of elements and DOF and improve the accuracy of the results for the problem globally, ANSYS does not provide the capability to define regions of interest with specific mesh grading requirements to control the mesh sizing in the regions of interest. Since there is no concept of ROI, ANSYS also does not account for visibility of regions that are hidden behind other geometrical features from the ROI to optimize mesh based on this criterion.

ANSYS is a closed source code application so it is not possible to extend the meshing capabilities of the software to adapt to one's particular needs.

2.7.2 CUBIT, Sandia National Lab.

CUBIT [58] is a two and three dimensional solid modeler and mesh generation tool that is developed by Sandia National Laboratories. CUBIT can produce surface and volume meshes for finite element analysis and uses a combination of techniques including paving, mapping, sweeping and various other algorithms for discretizing the geometry into a finite element mesh. CUBIT provides a few methods for controlling the size of the surface mesh and volume [58].

2.7.2.1 Constant sizing function

The *constant sizing function* specifies that a uniform element size be used over the interior of the surface or volume and is applied to the whole surface or volume

part.

2.7.2.2 Geometry adaptive sizing function

The *geometry adaptive sizing function* (also referred to as the *skeleton sizing function*), automatically generates a mesh sizing function based on geometric properties of the model. In this method, sizing function is created to satisfy these properties: (a) The sizes of the mesh elements vary smoothly throughout the mesh, (b) the mesh elements resolve the geometry to a sufficient degree and (c) the mesh elements do not over-resolve the geometry.

This sizing function uses geometric properties to influence mesh size. The scheme calculates or estimates:

- 3D-proximity (thickness though the volume)
- 2D-proximity (thickness across a surface)
- 1D-proximity (curve length)
- Surface curvature
- Curve curvature

Regions of relatively high complexity will have a fine mesh size, while regions of relatively low complexity will have a coarse mesh size. This method results in meshes that represent the geometry of the model accurately on a global scale.

2.7.2.3 Other sizing functions

CUBIT provides an experimental sizing function which is still under research and development and is not reliable yet. In this method a periodic sizing function can be specified and the mesh will be sized in periodical intervals on the surface and volume.

The meshing capabilities of CUBIT are very flexible and can be used in a variety of applications but it lacks the ability to plug in user defined mesh sizing functions to take full control of meshing process. The fact that it is a closed source application does not allow further customization of the application to address certain

needs such as accounting for ROI and geometry simplification based on location of ROI.

2.7.3 TetGen

TetGen generates tetrahedral meshes that are suitable for finite element and finite volume methods. It can generate tetrahedral mesh that cover of the interior of a piecewise linear boundary of a 3D domain and preserve the boundary.

TetGen provides mesh a few mesh refinement controls. The mesh can be sized so that the sizes of the tetrahedra are graded with respect to the input boundary. In other words, the mesh size is small close to the boundary and is gradually increased towards the interior of the domain [59].

It also supports mesh refinement through these methods that can be used for adaptive meshing:

- Using a .vol file, one can specify a maximum volume for each tetrahedron. Each tetrahedron's volume constraint is applied to that tetrahedron.
- Using a .node file, one can specify a list of additional nodes that must be included in the mesh.
- Using a nodal size map, one can specify the desired mesh edge size at each node.

TetGen is an open source project. It is written in C++ and the source code is available. While the mesh refinement capabilities of TetGen are very powerful, it does not have the ability to differentiate between the edge size of surface facets and volume cells. Also, since it conforms to the exact input geometry, it is not suitable for generating meshes that simplify parts of the boundary of the domain. TetGen also lacks the concept of ROI so it is not possible to define meshing criteria to account for visibility from ROI.

2.7.4 Gmsh

Gmsh [60] is a a three-dimensional finite element mesh generator with built-in pre-processing and post-processing facilities and is build around four modules:

geometry, mesh, solver and post-processing. It is able to create 2D and 3D structured and unstructured meshes [60].

Gmsh provides different methods to control mesh sizing.

- Characteristic length: The characteristic lengths at the geometrical points of the model can be specified. The size of the mesh elements will then be computed by linearly interpolating these characteristic lengths on the initial mesh.
- Curvature: the mesh will be adapted with respect to the curvature of the geometrical entities
- Box: the size of the elements inside and outside of a parallelipipedic region
- Explicit: the size of the mesh is specified using an explicit mathematical function.

Gmsh is written in C++ and the source code is available for modification. It provides very flexible control over the mesh sizing but lacks the concept of ROI. Because of this, it can not account for visibility of regions that are hidden behind geometrical features from the ROI to optimize mesh based on this criterion.

2.8 Conclusions

All domain decomposition methods should satisfy the following major goals: accurate modeling of geometry, mesh gradation, mesh quality. Meshes can be classified with respect to dimension (surface and volume mesh) or structure (structured and unstructured mesh). Structured mesh is numerically more efficient in solving some numerical problems but are hard to establish in complicated domains. Unstructured meshes, on the other hand, might be less efficient but are a lot more flexible and are applicable to complex domains.

The techniques used for 2D and 3D meshing were reviewed. Voronoi diagrams and Delaunay triangles proved to be very useful and popular for generating good quality meshes. Also some of the mesh post-processing methods were discussed that put the final touch on the generated mesh to improve mesh quality.

Finally a survey of notable exiting meshing tools was provided and their capabilities and limitations were discussed.

CHAPTER 3
The Proposed Framework

3.1 Introduction

Time to solution is defined as the total time involved in preparing the model, carrying out the numerical stress analysis computation and post processing the result (see section 1.4). The goal of this research is to reduce time to solution by automation and simplification of the process in preparing the models for underground excavations and by creating optimized meshes that produce accurate results in the vicinity of an ROI while minimizing the time required for numerical stress analysis. A review of the existing tools (see section 2.7) shows there is need for a specialized framework to address the requirements of this research.

A framework is proposed that contributes to reduction in *time to solution* at two levels: (a) it facilitates creation of geometry of the tunnels using minimal input data such as tunnel path and tunnel profile for tunnels with regular shapes or by importing existing surface triangulation of an excavation with complicated geometry, and (b) it cuts the time required for numerical analysis by reducing the number of surface and volume elements in the mesh (which in turn results in fewer DOF) while keeping the accuracy of results within a predetermined range at the ROI.

Figure 3.1 shows the overall concept of the framework and how each component relates to others. A brief description of the components shown in figure 3.1 follows. Detailed discussion about each component is provided throughout the rest of this chapter.

First and foremost, the geometry of a problem is defined by specifying the geometry of each individual excavation or tunnel segment. Each tunnel segment is defined using the minimal input data provided by the engineer. For excavations with regular shapes, this data consists of tunnel path and tunnel profile. For excavations with irregular shapes, if surface geometry is available in the form of a polyhedron, it can be directly used as input.

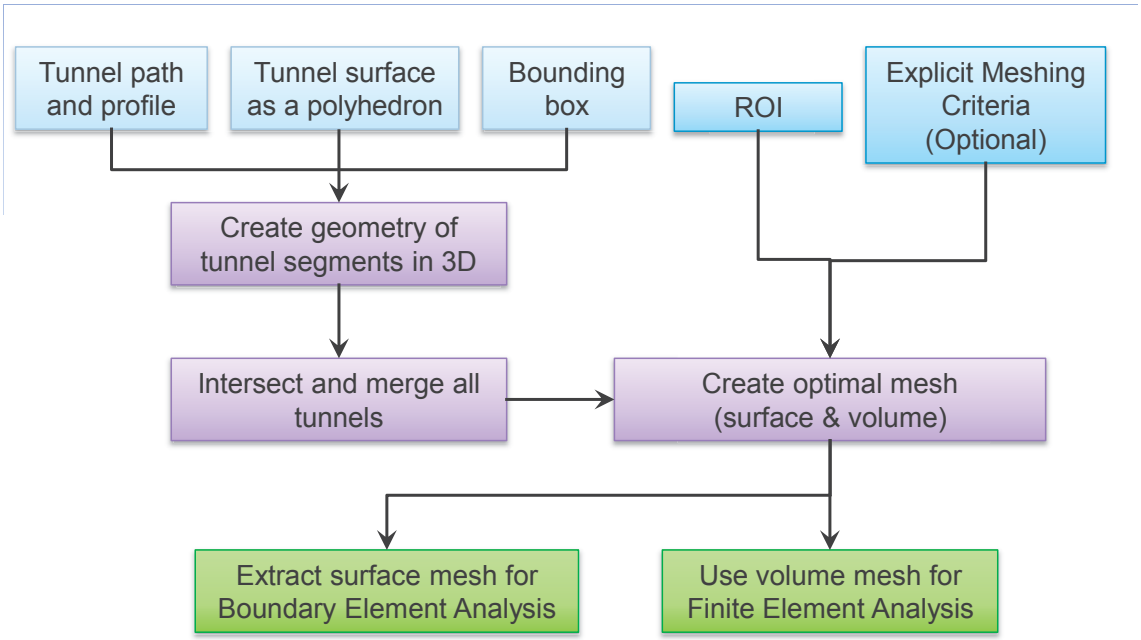

Figure 3.1: Concept of the proposed framework

Having the geometry of individual tunnels as polyhedra, it is possible to apply geometrical boolean operations on them and obtain the geometry of the whole problem domain as a polyhedron. Then a 3D Delaunay meshing algorithm is employed to generate a well formed 3D tetrahedral mesh that respects a set of predefined criteria to obtain the desired mesh. To improve the quality of the tetrahedra, a few mesh smoothing techniques are applied to the mesh. Now, the tetrahedra can be directly used for finite element analysis. For boundary element analysis, the surface mesh can be extracted from the generated 3D mesh.

3.2 Region of Interest

The recommendations that a rock engineer makes, come from stress analysis of the problem. In rock engineering, the domain of the problem is usually vast and out of the whole domain only a specific region is of importance and subject to study at a time. This region, where a solution to the problem is sought, is called the *region of interest* (ROI).

For an example of an ROI, consider a case where the rock engineer must asses the stability of the area around the face of excavation to ensure safe operations in

an ongoing tunneling project (see figure 3.2). The result of the stress analysis is required to be accurate in this region and the accuracy of the result in the rest of the domain is of no importance at this time. In this case, the region close to the face of excavation is the ROI. ROI is one of the key concepts in this research and the mesh optimization techniques discussed later on depend on it.

The ROI can be a point or a volume that is reasonably small with respect to the bounding box. There may be one or multiple ROI's for a particular problem. Throughout this work only one ROI is assumed for developing the framework but multiple ROI's can be accommodated in the framework by following the same principles laid out for one ROI.

3.3 The Bounding Box

The bounding box defines the bounding limits of the model and must include the ROI (see figure 3.2). It can be as large as the whole problem domain or be limited to just a region that contains a subset of features of the problem. In either case the engineer will make the decision based on the available data and the engineering knowledge. The proposed framework produces optimized meshes that result in low computational costs even if the whole problem domain is modeled. Here, the bounding box is represented by a closed polyhedron with oriented triangular facets.

3.4 Tunnel Geometry

In an underground excavation, depending on the purpose of the excavation and the chosen method of construction, the tunnel geometry can have a regular or irregular shape. Here, each case is considered individually and a proper solution is proposed to facilitate generating the model and reduce the time required for modeling the problem.

3.4.1 Model Representation

There are two major representation schemes that are used to describe a solid model: constructive solid geometry (CSG) and boundary representations (B-rep) [61], each having their own advantages and disadvantages.

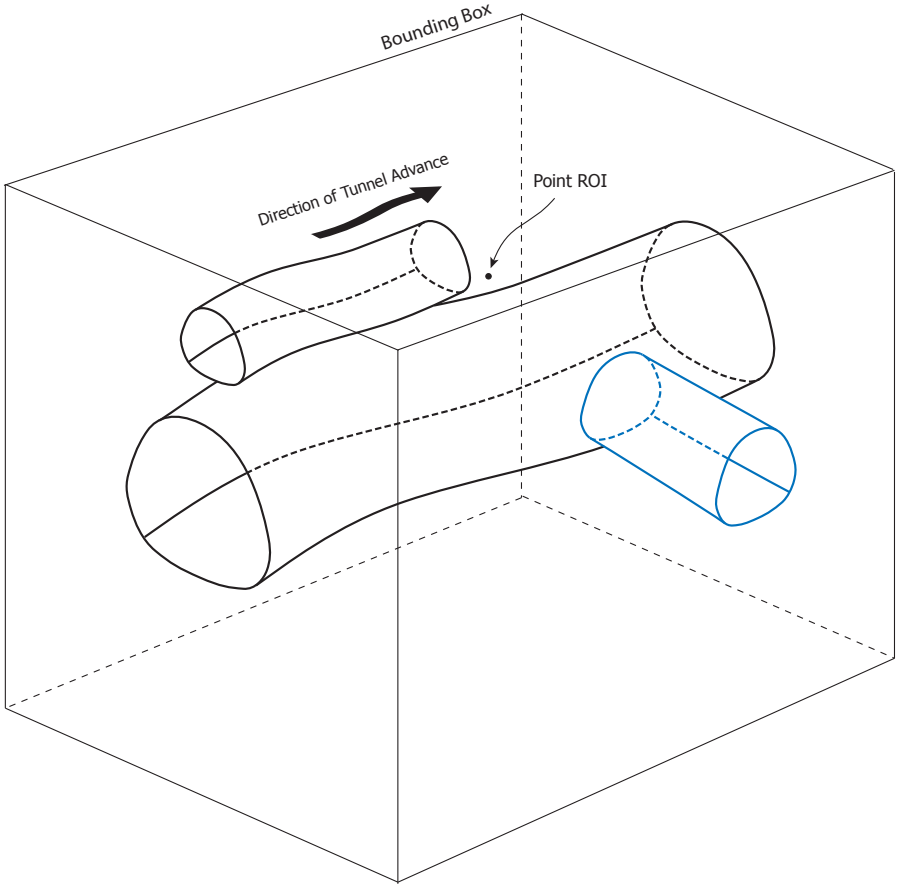

Figure 3.2: Bounding box and ROI

In *CSG* a solid is specified as boolean combination of a set of primitive objects. The solid objects are represented implicitly with a tree structure that consists of *leaves*, representing the primitive objects and *interior nodes*, representing boolean operations. The algorithms that operate on a CSG tree, evaluate properties on the primitive objects and propagate the results using the tree structure. CSG is suited for solids that have a regular shape and can be exactly described with simple primitives.

A *B-rep*, on the other hand, describes a solid in terms of the incidence structure and the geometric properties of all lower-dimensional features of its boundary. These features include faces, edges and vertices. Faces are oriented to determine the interior of the solid. B-rep is a proper choice for representing solids with highly irregular geometry.

By nature, underground excavations can be very irregular in shape therefore in this study B-rep is the chosen scheme for modeling the problem. In particular, closed polyhedral surfaces that consist of triangular facets are used to represent the boundary of the geometry.

3.4.2 Tunnels with Regular Shapes

The idea is to use the information about the tunnel path and cross-sections to automatically create the geometry of the tunnel.

3.4.2.1 Tunnels as Generalized Cylinders

A generalized cylinder is a representation of an elongated object that has a main axis (*directrix or spine*) and a smoothly varying cross-section (*generatrix*) [62]. See Figure 3.3. Directrix and generatrix can both be open or closed curves.

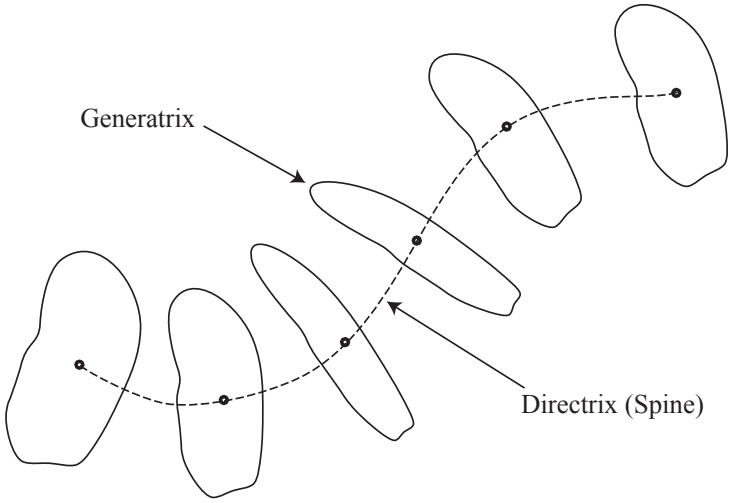

Figure 3.3: A generalized cylinder

In tunneling, spine is analogous to tunnel path and generatrix is tunnel cross-section. Spine is a bounded 3D curve serving as the tunnel path and is mathematically defined as:

$$\mathbf{A} = \mathbf{A}(s),\ 0 \leq s \leq 1 \tag{3.1}$$

Generatrix or tunnel cross-section is a closed curve that is defined as:

$$\mathbf{C} = \mathbf{C}(t; s) = [x(t; s), y(t; s), 0],\ 0 \leq t \leq 1 \tag{3.2}$$

and the surface of the generalized cylinder or tunnel is given by:

$$\mathbf{R}(t; s) = \mathbf{A}(s) + x(t; s)\mathbf{X}(s) + y(t; s)\mathbf{Y}(s) \tag{3.3}$$

where $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ are orthogonal 3D unit vectors and $\mathbf{Z}$ is tangent to $\mathbf{A}(s)$, i.e. $\mathbf{Z}(s) = \frac{\mathbf{A}'(s)}{\|\mathbf{A}'(s)\|}$.

$\mathbf{X}(s)$ and $\mathbf{Y}(s)$ can be chosen as the normal and binormal vectors of spine curve $\mathbf{A}(s)$ or by rotation of those by a small angle. See figure 3.4.

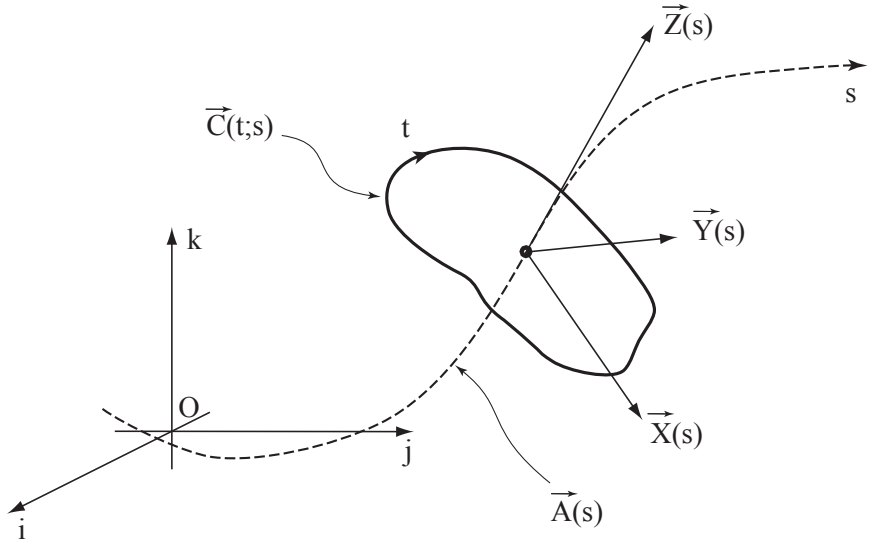

Figure 3.4: Mathematical representation of a generalized cylinder

3.4.2.2 Degeneracy of Generalized Cylinders

There are two kinds of degeneracies of generalized cylinders: local self-intersection and global self-intersection. See figure 3.5 for illustration.

As illustrated in figure 3.6, a condition to avoid local self-intersection of generalized cylinders is [62]:

$$max_t(x^2 + y^2) \leq \rho^2(s) \tag{3.4}$$

for all s, where $\rho(s)$ is the radius of curvature of the spine.

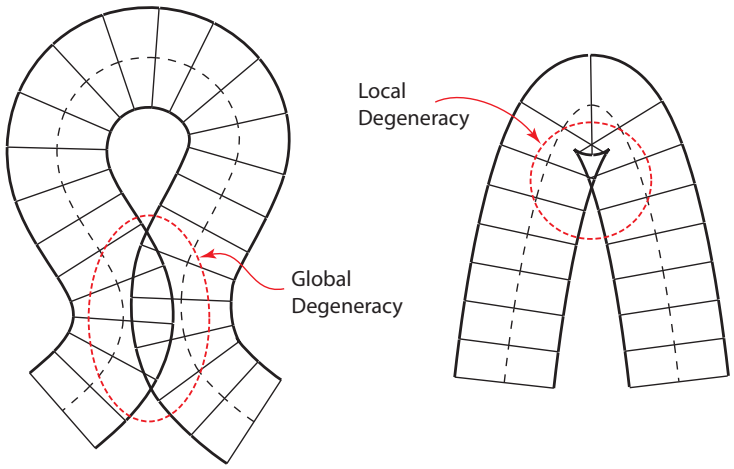

Figure 3.5: (Left) Global self-intersection. (Right) Local self-intersection

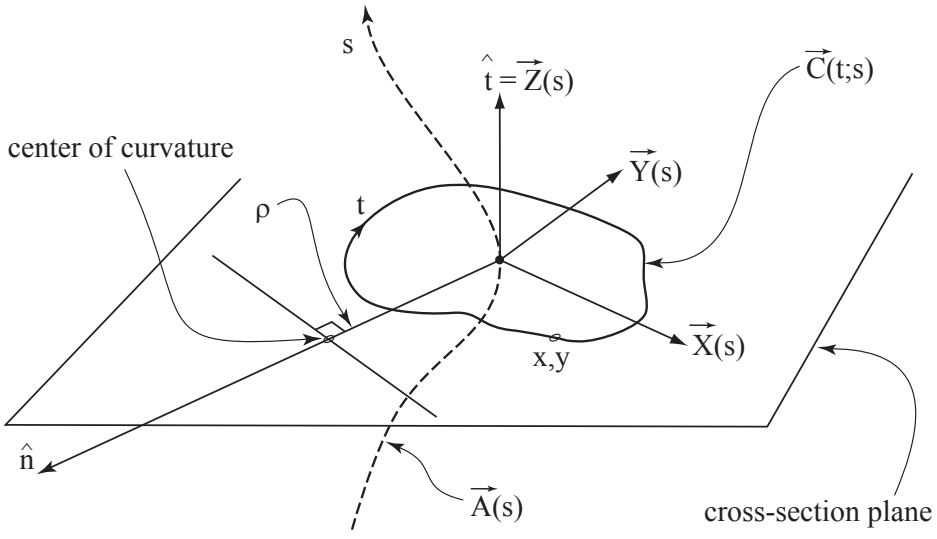

Figure 3.6: Criterion to avoid local self-intersection

3.4.2.3 Tunnel Cross-Section

The shape of the cross-section is approximated with a polygon. This approach is very flexible and can be used to represent very complex shapes. See figure 3.7. Larger number of sides in the polygon will result in a more accurate representation of the geometry of the tunnel.

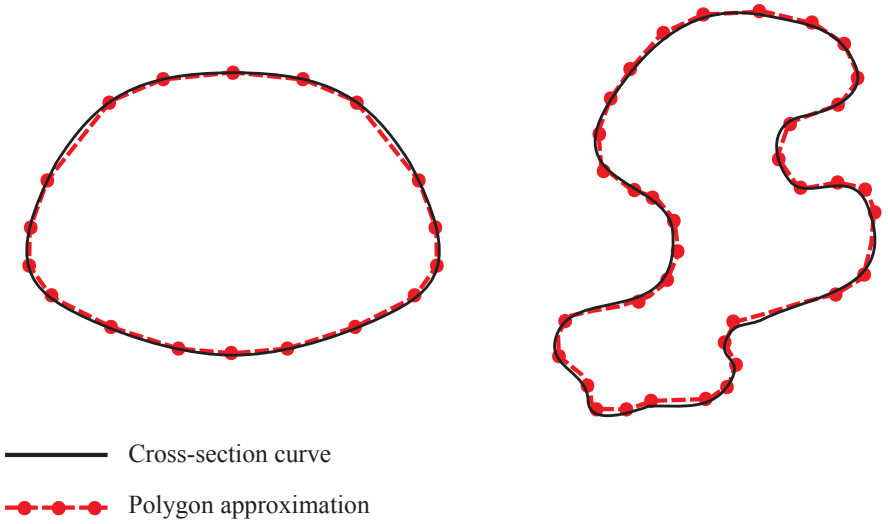

Figure 3.7: A mouth cross-section (left) and an rather complex cross-section (right) approximated with polygons

3.4.2.4 3D Tunnel Path

The 3D tunnel path is approximated using piecewise linear line-segments. The cross section can be scaled along tunnel path if required. See figure 3.8.

3.4.2.5 Tunnel Surface by Sweeping

One of the methods for creating the polyhedral surface of a tunnel is the extrusion of the two dimensional polygon that defines the cross-section. This method works well for a number of problems in civil and mining engineering. In mining, many of the excavations can be modeled by the extrusion process. For example, shafts, drifts and crosscuts are just a few types of structures which can easily be modeled using this process. As well, many underground civil engineering structures such as subway tunnels and hydroelectric power caverns can be modeled by

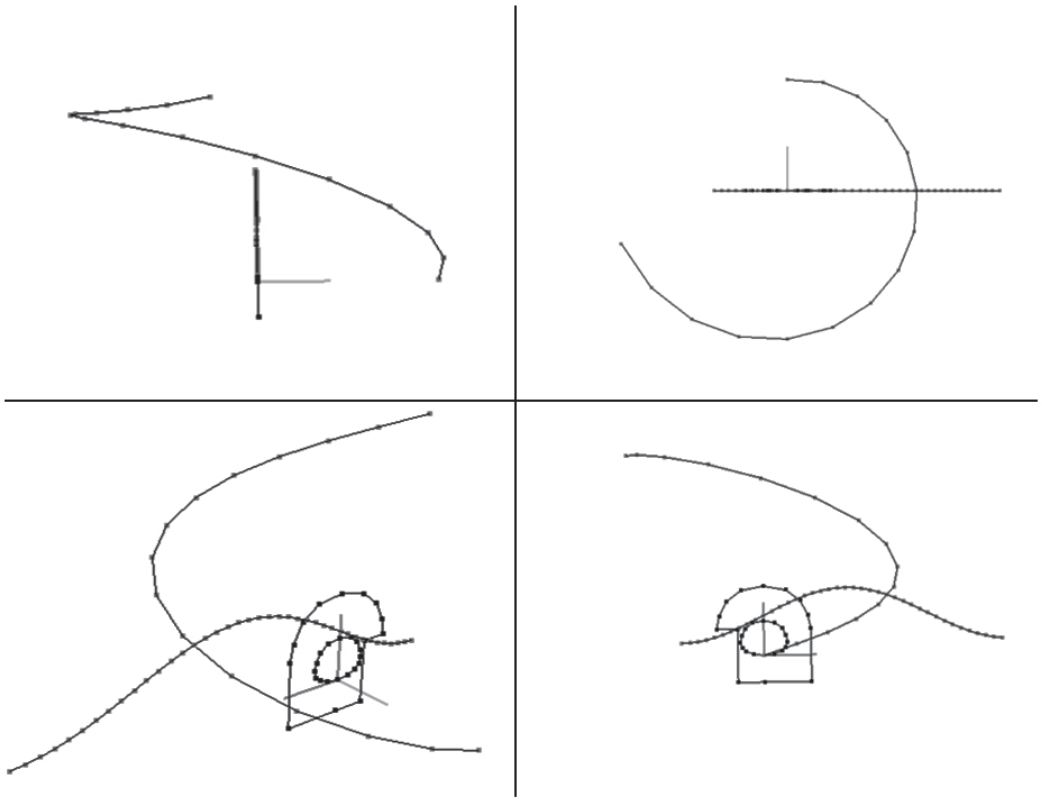

Figure 3.8: Two tunnel paths and profiles as seen from 4 different views

extrusion [63].

The algorithm implemented in this research is a generalized form of extrusion that is based on the concept of a generalized cylinder (see section 3.4.2.1). The outcome of the process is a closed polyhedron with triangular facets. All facets are oriented and have coherent normal directions. It is crucial that all facets have coherent normal directions because this information is used to determine if a point is topologically inside or outside of the polyhedron. In figure 3.9 three consecutive points, P_1, P_2 and P_3, on a tunnel path are shown with their corresponding cross-sections and the normal vector n of triangle abc points toward outside of the polyhedron.

Figure 3.10 shows the surface of two tunnels, one with a spiral path and the other with a wave like path, generated by the application that was developed for this research in order to implement the framework.

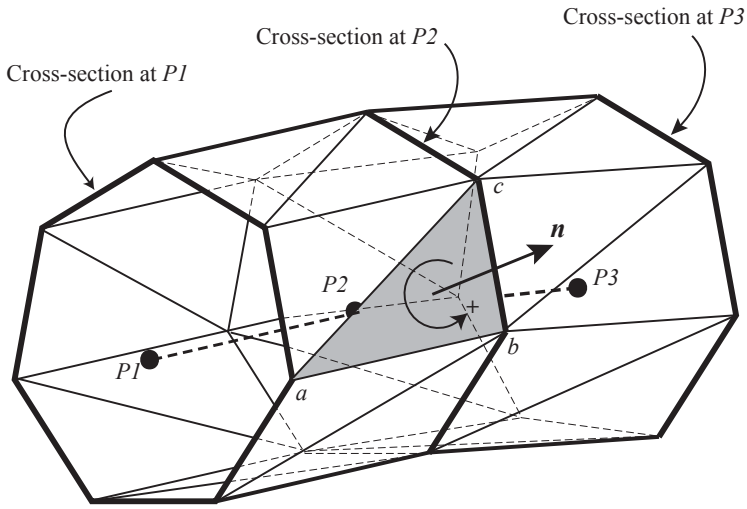

Figure 3.9: Creating the surface of a tunnel by sweeping the cross-section along the path.

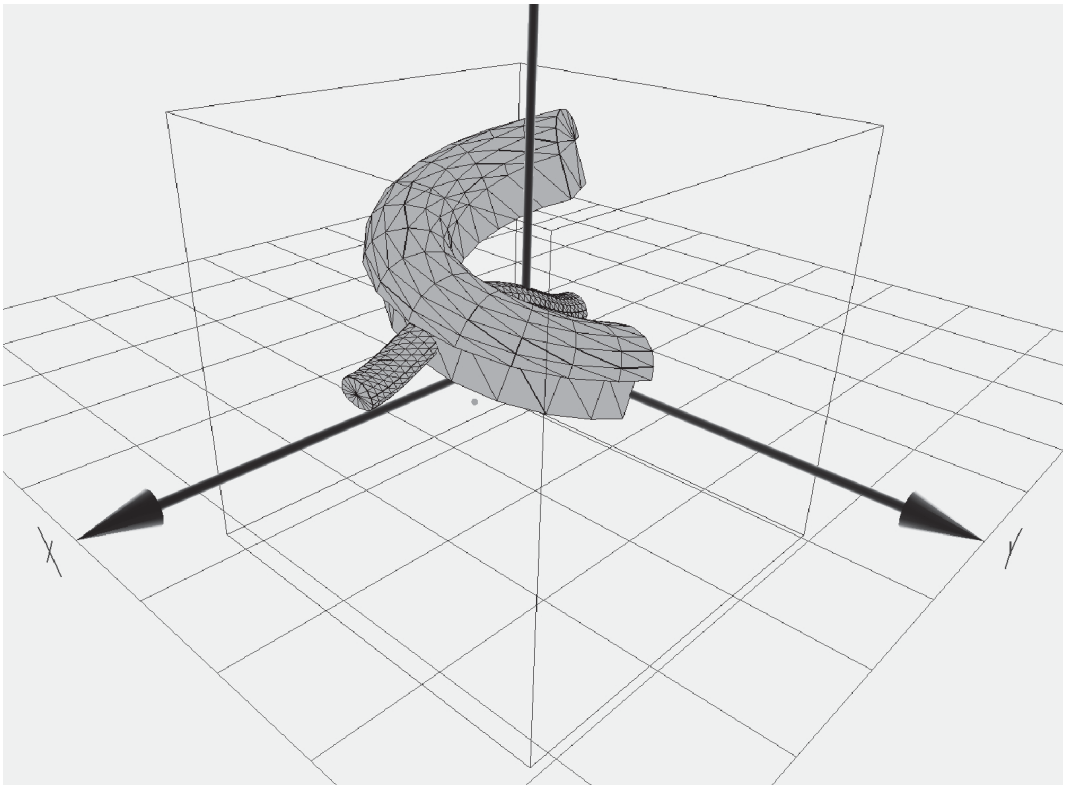

Figure 3.10: Geometry of the two tunnels created by the framework.

3.4.3 Tunnels with Irregular Shapes

The data that represents the shape of an irregular excavation can be gathered using laser tunnel scanners. A tunnel scanner captures the surface of the tunnels in the form of a *point cloud* which can then be converted to a polyhedral surface with triangular facets using a surface reconstruction algorithm. This polyhedron, which represents the exact geometry of the excavation, contains too much detail and if all these details are to be considered for meshing, the numerical stress analysis will be computationally expensive.

In order to achieve an acceptable computational cost, only the significant features of the tunnel surface should be kept and the rest should be ignored while creating the mesh. The meshing algorithm used in this study automatically discards the insignificant details in the excavation surface and produces a mesh that closely approximates the excavation surface according to a predetermined tolerance.

3.5 Intersection of Geometric Entities

To produce a mesh of the domain, the meshing algorithm requires a polyhedron that defines the whole boundary of the domain as input. All facets of this polyhedron must have coherent normal vectors to topologically identify the inside and outside space of the excavation.

To obtain this polyhedron, boolean operations, such as union and subtraction, must be performed on polyhedra that represent each tunnel and the polyhedron that represents the bounding box. See section 3.5.2 for more details about boolean operations on geometric entities.

3.5.1 Exact versus Inexact Arithmetic

Geometric objects are expressed in terms of a 3D coordinate system that is based on real numbers. Arithmetic operations done by digital computers are usually performed by using *floating point* numbers and suffer from a problem called *roundoff error*. Predicate operations such as intersection of geometric entities can return incorrect results if roundoff error occurs in the computations [20]. There are numerous algorithms and software libraries that are designed to remedy this

problem. Some use integer numbers altogether instead of floating point numbers, others use different techniques to get around this problem [20].

In this research, exact arithmetic operations [64] that are available through CGAL library [65] are used. Therefore, the framework is robust and never returns false results that could originate from roundoff errors. Of course this robustness comes at a cost: there is some speed penalty for using exact arithmetic but it is inevitable when robustness and correctness have a high priority.

3.5.2 Boolean Operations on Geometric Entities

In this research B-rep is chosen to represent geometric entities. In particular, polyhedra with oriented triangular facets are used to model the geometric entities. Boolean operations on polyhedra are a set of operations that result in a new polyhedron. To perform boolean operations a specific kind of polyhedron, named Nef-polyhedron, is employed.

The theory of Nef-polyhedra has been developed for arbitrary dimensions [66] but here an implementation of Nef-polyhedra for 3 dimensions is used. A Nef-polyhedron in dimension 3 is a point set $P \subseteq R^3$ generated from a finite number of open halfspaces by set complement and set intersection operations. Consider two Nef-polyhedra P_1 and P_2. The union of P_1 and P_2, produces a new Nef-polyhedron, P_U, that contains the point set that exist in both P_1 and P_2:

$$P_U = P_1 \cup P_2 \tag{3.5}$$

$$P_U = \{x \subseteq \mathbb{R}^3 : x \in P_1 \; or \; x \in P_2\} \tag{3.6}$$

The relative complement (i.e. set difference) of P_1 and P_2, produces a new Nef-polyhedron, P_C, that contains the point set that exist in P_1 but not in P_2:

$$P_C = P_1 \setminus P_2 \tag{3.7}$$

$$P_C = \{x \subseteq \mathbb{R}^3 : x \in P_1 \mid x \notin P_2\} \tag{3.8}$$

To get the polyhedron that represents the whole model, first the union of all

tunnel segment polyhedra is found:

$$P_{union} = P_1 \cup P_2 \cup P_3 \cup \cdots \cup P_n \tag{3.9}$$

where P_i is the i^{th} tunnel segment. Then the set difference of this union and the bounding box is found:

$$P_{final} = P_{BBox} \setminus P_{union} \tag{3.10}$$

where P_{BBox} is the bounding box polyhedron and P_{union} is obtained from equation 3.9. P_{final} is the polyhedron that contains all the points inside the bounding box and outside of the tunnel segments and it is used by the meshing algorithm to create the mesh.

3.6 The Meshing Algorithm

The meshing algorithm is the core component of the framework. It is responsible for producing a mesh of the domain subject to certain criteria. These criteria are either given explicitly or are inferred from other information such as location of the ROI.

To mesh the domain, a 3D Delaunay meshing algorithm is used. Delaunay based algorithms are capable of producing quality meshes and provide control over mesh sizing throughout the domain. They have solid mathematical background and are guaranteed to terminate [20].

In numerical stress analysis, fine meshes generally produce more accurate results compared to coarse meshes. A finer mesh means more DOF which translates into longer numerical analysis and requires more computational resources. The general idea is to refine the mesh in areas that have more contribution to the results at ROI and keep a coarse mesh in other areas of the problem domain. This will result in an optimum mesh that significantly reduces the time required for numerical analysis.

The proposed algorithm extends the 3D Delaunay algorithm that was developed by Rineau et al. [67] to enable applying an ROI based mesh sizing function over the domain. The output mesh is an optimized 3D triangulation of the domain

that can be used for finite element or boundary element analysis.

3.6.1 How the Meshing Algorithm Works

The major components of information required by the meshing algorithm are:

- The input domain which is a polyhedron that defines the geometry of the problem
- The ROI
- A set of optional explicit criteria

Figure 3.11 shows an overview of the meshing algorithm. The actual implementation of the algorithm is based on a generic software design for Delaunay refinement meshing that uses a *recursive* technique [68].

3.6.2 The Input Domain

For the meshing algorithm to work properly, the domain to be meshed must be representable as a pure 3D complex. A 3D complex is a set of faces with dimension 0 (vertices), 1 (edges), 2 (facets) and 3 (cells) such that all faces are pairwise interior disjoint, and the boundary of each face of the complex is the union of faces of the complex. The 3D complex is *pure*, meaning that each face is included in a face of dimension 3, so that the complex is entirely described as a set of 3D cells. The set of faces with dimension lower or equal than 2 form a 2D subcomplex. By this definition, P_{final}, the polyhedron that represents the model geometry (equation 3.10) is an acceptable form of input for the meshing algorithm.

3.6.3 Meshing Criteria

To refine the mesh, the algorithm is driven by five criteria: three conditions for mesh surface facets and two conditions for mesh volume cells.

1. The criteria for surface facets are:
   (a) *Angular bound* which controls the facet shape,
   (b) *Radius bound* which controls facet size, and

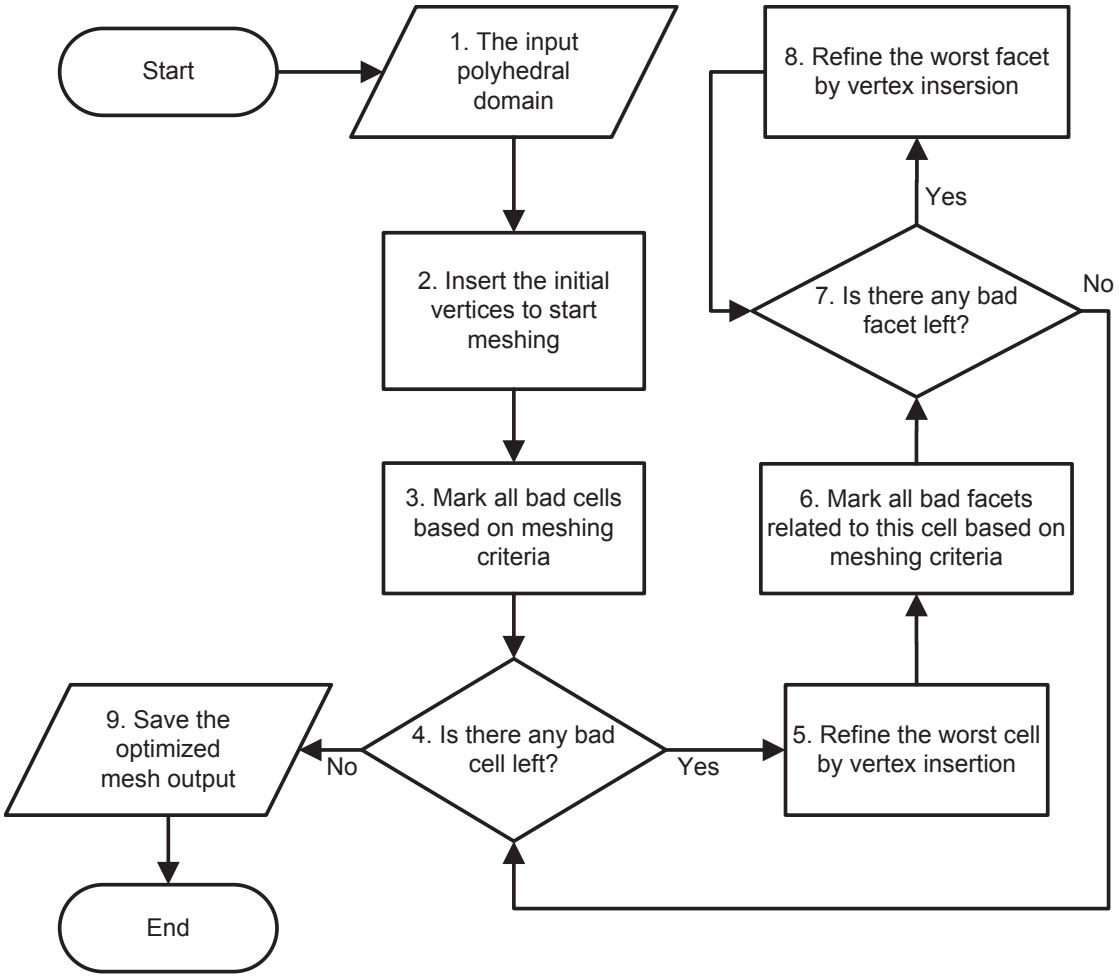

Figure 3.11: A simplistic flowchart of the meshing algorithm

(c) *Distance bound* which controls how closely the surface facets approximate the geometry of the problem.

2. The criteria for cells are:

   (a) *Radius-edge bound* which controls the cell shape, and

   (b) *Radius bound* which controls the cell size.

Each of these five criteria can be individually controlled throughout the problem domain. Each criteria can be evaluated at any point (x, y, z) in the domain. A detailed discussion of these criteria follows.

Facet Radius Bound

Radius bound of the facets, Q_{fr}, controls the size of the surface facets. For a triangular facet $\triangle ABC$ the radius bound is equal to the circumradius of the triangle. The meshing algorithm guarantees every facet in the triangulation has a radius bound smaller than a specified value.

Facet Distance Bound

The distance bound, Q_{fd}, controls how closely the surface facets approximate the geometry of the problem. For a triangular facet $\triangle ABC$ the radius bound is calculated as follows. Let P be the circumcenter of $\triangle ABC$ and H be the intersection of a line perpendicular to the plane of the triangle at P and the surface of the boundary surface (i.e. the input polyhedron). The distance between P and H is defined as the distance bound.

By manipulating the designated maximum distance bound for different regions in the model it is possible to control the refinement of the surface facets in the mesh. By choosing a larger facet distance bound in the regions that do not have significant impact on the amount of stress at the ROI, the amount of detail in the surface mesh can be reduced in those regions. This is particularly important when creating optimized surface meshes for boundary element analysis.

Facet Angular Bound

Angular bound of the facets, Q_{fa}, controls the shape of the surface facets of the mesh. For the triangular facet $\triangle ABC$ with sides a, b and c the angular bound is defined as:

$$Q_{fa} = \frac{4S \cdot d_{min}}{abc} \tag{3.11}$$

where

$$d_{min} = min\,(a, b, c) \tag{3.12}$$

S is the area of the triangle and R is its circumradius. The meshing algorithm guarantees every facet in the triangulation has an angular bound larger than a minimum specified value.

Cell Radius Bound

Cell radius bound, Q_{cr}, controls the size of tetrahedra in the mesh. The radius bound of a tetrahedron is equal to radius of the circumsphere of that tetrahedron. The meshing algorithm guarantees every cell in the mesh has a radius bound smaller than a specified value. This parameter plays an important role is optimizing the volume mesh. By manipulating the designated value of the cell radius bound for different regions in the model smaller tetrahedra are used in areas that have significant influence on the amount of stress at ROI and larger tetrahedra are used in other areas resulting in an optimized mesh with lower number of cells which translates into fewer DOF.

Cell Radius-edge Bound

For accuracy in the finite element analysis, it is generally necessary that the shape of elements have bounded aspect ratio. The aspect ratio of an element is the ratio of the maximum side length to the minimum altitude. For a quality mesh, this value should be as small as possible. For example *thin and flat* tetrahedra tend to have large aspect ratios. In this study, radius-edge ratio which is a similar but weaker quality measure and is more suitable for the Delaunay algorithm [69] is used.

The radius-edge bound, Q_{ce}, controls the shape of the cells. For tetrahedron $ABCD$ with edges e_1, e_2, e_3, e_4, e_5 and e_6 the radius-edge is defined as:

$$Q_{ce} = \frac{R}{e_{min}} \tag{3.13}$$

where

$$e_{min} = min\,(e_1, e_2, e_3, e_4, e_5, e_6) \tag{3.14}$$

and R is the radius of the circumsphere of tetrahedron $ABCD$.

For all well-shaped tetrahedra, the radius-edge ratio is small (figure 3.12), while for most of badly-shaped tetrahedra, this value is large (figure 3.13). Hence, in a quality mesh, this value should be bounded as small as possible. However, the ratio is minimized by the regular tetrahedron (in which the six edges have equal

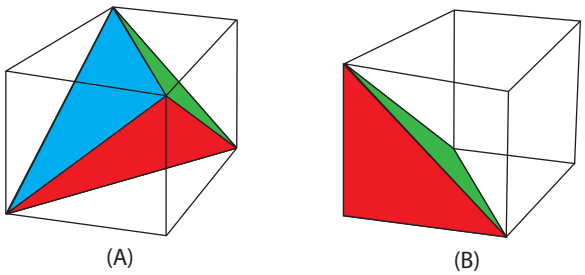

Figure 3.12: Radius-edge for well shaped tetrahedra. For tetrahedron A (left): $Q_{ce} \approx 0.612$ and for tetrahedron B (right): $Q_{ce} \approx 0.866$

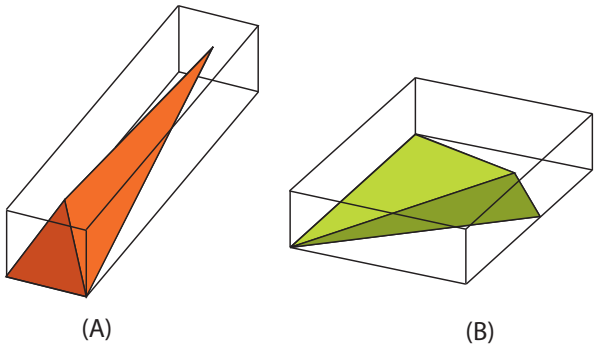

Figure 3.13: Radius-edge for badly shaped tetrahedra. For tetrahedron A (left): $Q_{ce} \approx 2.51$ and for tetrahedron B (right): $Q_{ce} \approx 2.5$

lengths and the circumcenter is the barycenter), that is:

$$Q_{ce} \geq \sqrt{6}/4 \approx 0.612 \tag{3.15}$$

A special type of badly-shaped tetrahedron is called *sliver* (see figure 3.14). This type of tetrahedron is very flat and nearly degenerate. Slivers can have radius-edge ratio as small as $\sqrt{2}/2 \approx 0.707$ thus the radius-edge ratio is not a proper measure for weeding out the slivers. However, Miller et al. [69] have pointed out that it is the most natural and elegant measure for using in Delaunay refinement algorithms. To remove slivers from the final mesh a few mesh smoothing techniques have been used which are discussed in section 3.6.6.

Applying Multiple Criteria

The meshing algorithm supports applying multiple criteria when refining the mesh subject to each of the five conditions mentioned previously. For example, to

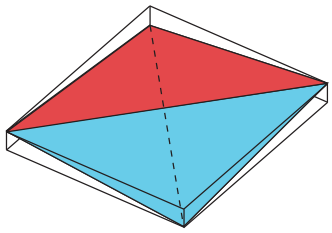

Figure 3.14: The radius-edge ratio of a sliver: $Q_{ce} \approx 0.707$

control the cell size when refining the cells, it is possible to have the algorithm to do the cell *badness* check against multiple criteria. If any of those criteria indicates badness of a cell, it will be marked for refinement. This behavior is very well suited for extending the current implementation by adding new criteria to the existing ones.

3.6.4 ROI Based Mesh Sizing

To drive the refinement process automatically, a mesh sizing function is needed. This function should account for the ROI in order to decide on the sizing of the elements in each region of the domain. The following measure is proposed to estimate upper bound of the meshing criteria:

$$S = S_{min} + (S_{max} - S_{min}) \cdot K \tag{3.16}$$

$$K = \frac{(w_{proximity} \cdot C_1 + w_{visibility} \cdot C_2)}{w_{proximity} + w_{visibility}} \tag{3.17}$$

and

$$S_{min} \leq S \leq Min\,(S_{cap}, S_{max}) \tag{3.18}$$

where S is the upper bound of the quantity that is being measured for element refinement (e.g. Q_{fr}, Q_{fd}, Q_{fa}, Q_{cr} or Q_{ce}) and S_{min} and S_{max} are the minimum and maximum desired upper bounds of S and are provided before the meshing begins. S_{cap} is a predetermined constant and is the final upper bound for the sizing function.

The mesh sizing function is used to evaluate the badness of facets and cells. For example, when applying the cell radius bound criterion(Q_{cr}), S is evaluated for

a cell and if $Q_{cr} > S$ then the cell is marked as *bad* to be refined further by the meshing algorithm.

3.6.4.1 Proximity Factor

The concept of *region of influence* [70] is used in developing the framework. It is known from the continuum mechanics that the degree of interaction between regions in the continuum depends on the inverse of the squared distance between them [3]. A direct result of this fact is that elements farther from the ROI have less contribution to the amount of stress at ROI therefore these elements can be coarsened without affecting the accuracy of results at ROI.

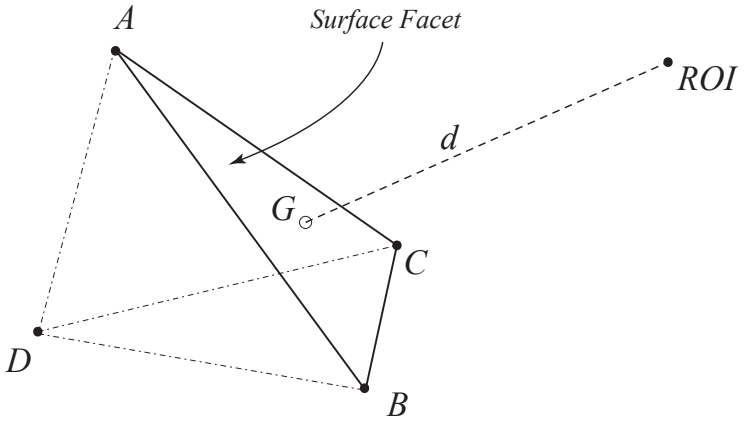

Figure 3.15: Proximity for facet ABC

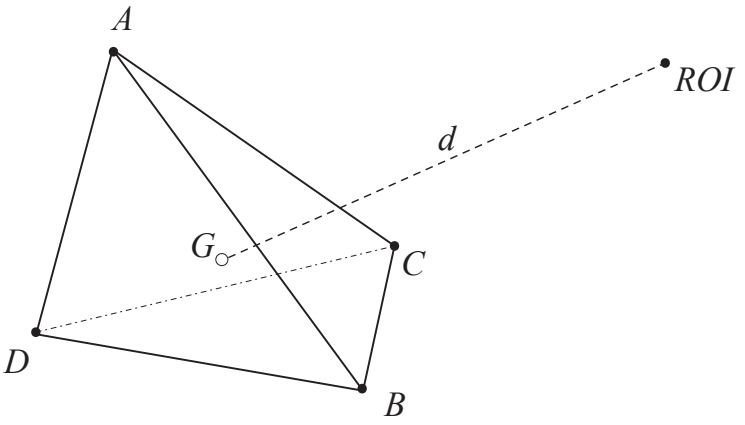

Figure 3.16: Proximity for cell $ABCD$

In equation 3.17, $w_{proximity} \cdot C_1$ is the contribution of proximity of the element to the ROI. The $w_{proximity}$ is the weight of proximity contribution factor and C_1 is defined as:

$$C_1 = \frac{d^2}{D^2} \tag{3.19}$$

$$0 \leq d \leq D \tag{3.20}$$

where d is the distance between ROI and centroid of the current element being refined and D is the maximum possible value for d (see figures 3.15 and 3.16.

The sizing function can be set up for each of the five criteria discussed in section 3.6.3 and be activated for every one of them in the meshing process.

3.6.4.2 Visibility Factor

Regions of the domain that are not directly eying the ROI will have less influence on the results at the ROI. That is because the excavations that are laying between the refinement point and the ROI disturb the stress field distribution in the domain and act as a blockage that prevent direct influence of that point on ROI. The second term in equation 3.17, $w_{visibility} \cdot C_2$, accounts for the contribution of this phenomenon. The $w_{visibility}$ is the weight of visibility factor and C_2 is defined as:

$$C_2 = \frac{i}{N} \tag{3.21}$$

$$0 \leq i \leq N \tag{3.22}$$

where i is the number of intersections between the surface of excavations and the line segment stretching from ROI to the centroid of the element currently being refined. The larger the calculated value of i is for a point, the lesser influence it has on the ROI. N is a predetermined positive integer and is the upper bound for i.

Figure 3.17 shows cross section of two excavations and three refinement points A, B and C with $i = 0$ for point A, $i = 2$ for point B and $i = 4$ for point C.

To calculate i in equation 3.21, a line segment is constructed from the point ROI to the refinement point and then the number of intersections of this line segment with the input polyhedron is calculated. To test for the intersections, the simplest method would be to use a *brute force search*. This method is computationally

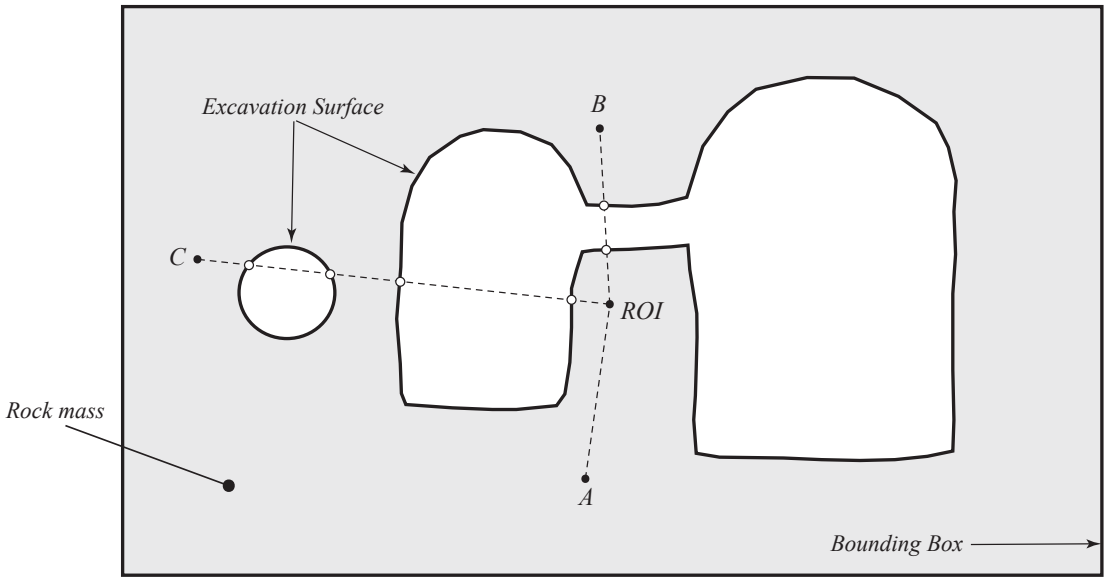

Figure 3.17: Cross section of an excavation showing point visibility from the ROI

expensive because this intersection check is performed each time a facet or cell is being refined and for a polyhedron with a large number of triangles this will be inefficient.

Another approach which is more efficient is using the *Axis-Aligned Bounding Box Tree* (AABB Tree) method [71]. The AABB tree provides the means to perform efficient intersection and distance queries against sets of finite 3D geometric objects stored in a static data structure. The data structure is created once and the same data structure is used for all future queries. The AABB tree construction is initialized by computing the AABB of the whole set of triangles that make up the input polyhedron (i.e. P_{final}). All triangles are then sorted along the longest coordinate axis of this box, and the triangles are separated into two equal size sets. This procedure is applied recursively until an AABB contains a single triangle. An intersection query traverses the tree by computing intersection tests only with respect to the AABB's during traversal, and with respect to the input triangle at the end of traversal (in the leafs of the tree).

Sizing Function Diagram

A closer look at the definition of the sizing function in equations 3.16 and 3.17 shows that it is a parabolic function (figure 3.18) and the upper and lower bounds of S are as follows:

$$S_{min} \leq S \leq Min\left(S_{cap}, S_{max}\right) \tag{3.23}$$

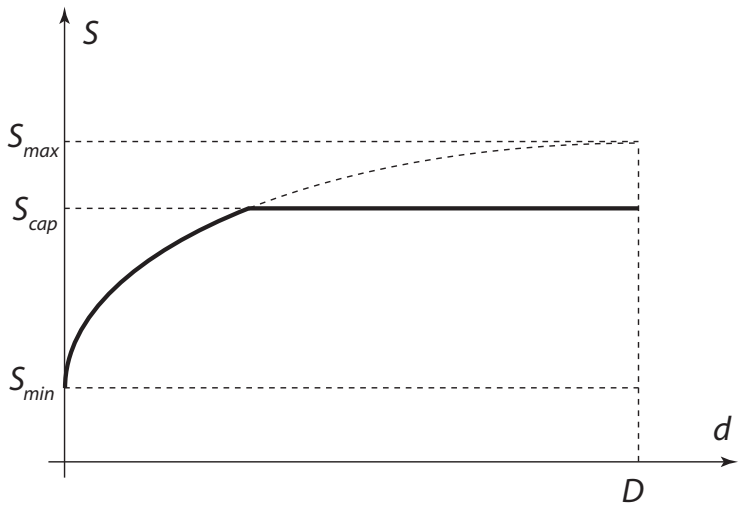

Figure 3.18: The solid line is the ROI based mesh sizing function according to equation 3.16

3.6.5 Explicit criteria

In addition to the criteria that are calculated based on equation 3.16, it is also possible to use an explicit algebraic sizing function, $S_f = f\left(x, y, z\right)$, to control the mesh refinement process. An independent sizing function can be applied to each of the five criteria mentioned in section 3.6.3.

3.6.6 Mesh Smoothing

At some point the meshing algorithm will stop the refinement process and guarantee that all elements in the output mesh satisfy all five criteria that drive the meshing process (see section 3.6.3). For better accuracy in the finite element analysis, the tetrahedra in the mesh must have a small aspect ratio. As it was discussed earlier, the radius-edge which is a weaker criteria similar to aspect ratio is

used in the meshing algorithm. The radius-edge is not a good measure for *slivers*. As a result, there will be some slivers in the output mesh. To eliminate the slivers and improve mesh quality the following process is applied to the mesh.

3.6.6.1 Lloyd Smoother

The Lloyd smoother is a global mesh optimizer and improves the mesh by moving vertices in order to minimize a global mesh energy. The mesh energy is the error resulting from interpolation of function $f(x) = x^2$ by a piecewise linear function [72]. Lloyd improves the whole mesh rather than focusing on removing slivers but it is known empirically to be very efficient as a preliminary mesh optimization phase that will enhance the efficiency of Perturber and Exuder that are applied after.

3.6.6.2 Perturber

Perturber aims to remove slivers by relocating the vertices of the mesh while keeping the mesh Delaunay. There are methods that explicitly perturb the slivers through random vertex relocation. These methods are effective but slow. Here a more efficient and effective method that favors deterministic over random perturbation is used [73].

3.6.6.3 Exuder

Exuder chases down the remaining slivers and removes them. To do so, the Exuder turns the Delaunay mesh into a weighted Delaunay mesh with optimal weights applied to vertices [74]. The Exuder must be the last optimization process that is run on the mesh because it changes the weights of the Delaunay mesh.

3.7 The Meshed Model

After the mesh smoothing process is finished, the final mesh can now be used for stress analysis. For finite element analysis the tetrahedra are extracted, rock mass properties are assigned to the cells, boundary conditions and loading are applied and then a finite element analysis tool is used to run the stress analysis process. To use

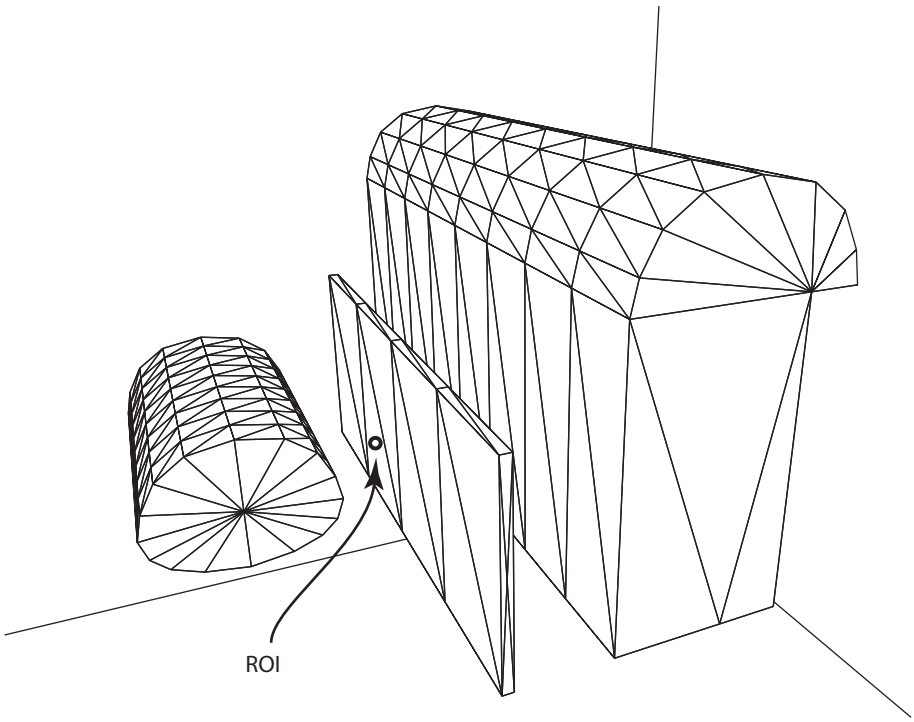

Figure 3.19: Geometry of the underground excavation.

the output mesh for boundary element analysis, the surface facets are extracted from the volume mesh. After boundary conditions and loading are applied, a boundary element analysis tool can be used for stress analysis.

Figure 3.19 shows a 3D view of an underground excavation along with the location of the ROI. The framework was used to create the mesh of the model. A uniform mesh created by the framework with $Q_{cr} = 0.4$ has a total number of DOF equal to $287,000$ and a linear finite element analysis for this uniform mesh takes about 875 seconds to finish.

Figure 3.20 is an optimized mesh of the same geometry produced by applying proximity and visibility factor to control cell radius (Q_{cr}) and facet distance (Q_{fd}) criteria. Number of DOF in the optimized mesh is $19,400$ and the linear finite element solution takes only 15 seconds. The results obtained for the optimized mesh shows 93% reduction in the number of DOF and 98% reduction in the stress analysis time.

It is worth noting that this significant improvement introduces only 20% of error to the amount of principal stress (σ_1) at ROI compared to the uniform mesh.

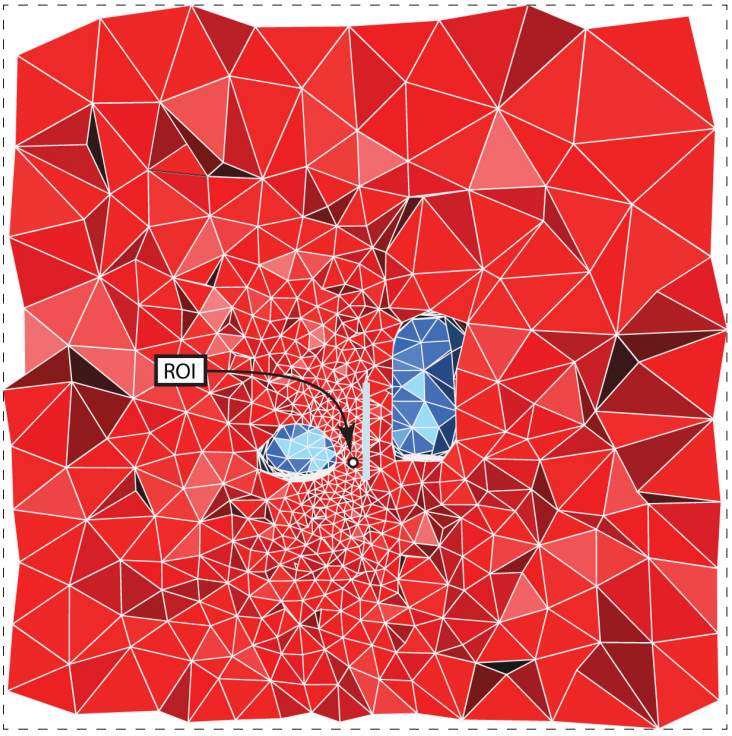

Figure 3.20: Cross section of the 3D mesh optimized with cell radius bound and facet distance bound: $w_{proximity} = 1.0$ **and** $w_{visibility} = 0.1$

This amount of error is acceptable because the rock mass property seldom has an accuracy better than $10 - 20\%$ [53]. These reductions in time to solution are for a linear stress analysis. For a non-linear stress analysis the savings in time and resources will become even more significant. In chapter 4 the efficiency of the method and the effects of each of the five cell and facet criteria on the number of DOF and the accuracy of the stress analysis results at the ROI are investigated.

3.8 Implementation of the Framework

To realize the proposed framework and study the application of the framework on some problems in mining and civil engineering a software application was developed. The software application is designed to be cross platform, meaning that it can be run under different operating systems including *Windows*, *Mac OS X* and *Linux* family of operating systems. It is developed using the C++ programming language

and uses advanced C++ programming techniques like *template programming*. It also leverages several C++ libraries such as *Nokia Qt*, *Boost*, *OpenGL* and *CGAL*.

The software application operates both from command line and through a Graphical User Interface (GUI). In the command line mode, it takes a text file containing the description of the problem in form of Extensible Markup Language (XML), produces the output mesh and then saves the resulting mesh along with some statistical information into designated output files. In the command line mode, no human intervention is required to create the mesh from the input file which makes it ideal for integration as part of an automated finite element analysis tool-chain. The GUI mode is used to provide visual insight into different stages of mesh generation and help in debugging and finding out problems in the process of modeling.

3.9 Conclusions

Successful employment of new tunneling methods such as NATM to design and execute underground excavation projects depends on accurate estimation of stresses and strains for a 3D model of the excavation. Creating the 3D geometry of the problem and preparing an optimal mesh for numerical stress analysis is a time consuming process. Right now this process requires human intervention and depends on expert knowledge to simplify the model in certain regions and keep the details in other areas and the decisions made by the expert are subjective. Because of the sheer size of the mining problems the numerical solution can be very lengthy. Besides, because of non-linear constitutive models that govern the behavior of rock mass the numerical solution must be run iteratively which means even longer computational times (often measured in days or weeks).

A framework was introduced that reduces time to solution by simplifying creation of the geometry of underground excavations and by creating optimal meshes. The framework contributes to reduction in time to solution at two levels: (a) it facilitates creation of geometry of the tunnels and (b) it cuts the time required for numerical analysis by reducing number of surface and volume elements in the mesh while keeping the results accurate enough at the ROI.

The concept of the framework was first depicted in a diagram that shows

the components involved and their interrelationship. Then each component in the framework was discussed individually. A generic *sizing function* (equation 3.16) was proposed that is capable of incorporating the effects of proximity and visibility to predict a proper value for a given meshing criteria (such as cell radius bound or facet distance bound).

Mesh smoothing operations such as Lloyd Smoother, Perturber and Exuder were discussed. These operations are required for producing high quality meshes that are to be used for numerical analysis. Finally a brief overview of the technology used in developing the software application that realizes the proposed framework was provided.

CHAPTER 4
Applicability, Accuracy and Efficiency of the Framework

4.1 Introduction

A framework for automated modeling of underground excavations was developed in chapter 3. The framework produces an optimized mesh of the problem by accounting for the location of ROI and produces a mesh that is coarser in the regions that have little or no influence on the solution at the ROI.

Since the input data such as rock mass property has seldom an accuracy better than 10-20% [53], the accuracy of the solution is also allowed to be about the same. The optimized mesh maintains the accuracy of the solution at the ROI within this range while reduces the time to solution by an average of 90%.

The optimized mesh produced by the framework should be appropriate (i.e. the elements must be well shaped) in order to be suited for use in numerical stress analysis. The mesh refinement process is governed by five criteria: cell radius bound, cell radius-edge bound, facet distance bound, facet radius bound and facet angular bound. In this chapter, the effects of each of these five criteria on the optimized mesh are studied.

This chapter is dedicated to evaluating the applicability, accuracy and efficiency of the proposed framework. To illustrate this, the framework was applied to a few practical mining problems and the reduction in time to solution was measured for each case. A strict measure for accuracy of results of a stress analysis is the principal stress induced in the rock mass. Therefore, in order to determine the accuracy of the results produced by the optimized meshes, principal stress was used as a metric. Detailed comparison of the results produced by the framework are provided for different scenarios in the following sections of this chapter.

4.2 Meshing Criteria

The Delaunay meshing algorithm incorporated in the proposed framework refines the mesh subject to five criteria: Q_{cr}, Q_{ce}, Q_{fr}, Q_{fd} or Q_{fa} (see section 3.6.3).

Each of these five criteria can be evaluated using the ROI based equation suggested earlier (see equation 3.16) or can be explicitly calculated from a scalar function $f(x, y, z)$ for each point in the domain. Table 4.1 shows a template of the matrix of parameters that control the meshing algorithm ($w_{proximity}$, $w_{visibility}$, S_{min}, S_{max} and S_{cap} are defined in section 3.6.4).

	$w_{proximity}$	$w_{visibility}$	S_{min}	S_{max}	S_{cap}
Q_{cr} (cell radius)					
Q_{ce} (cell radius-edge)					
Q_{fd} (facet distance)					
Q_{fr} (facet radius)					
Q_{fa} (facet angle)					

Table 4.1: Template of the matrix of parameters that control the meshing algorithm.

The framework uses equation 3.16 to calculate each criterion if the parameters are provided for it. If no parameters are specified for a criterion, the framework will not account for that criterion when refining the mesh.

Depending on what numerical method is going to be used for stress analysis, one or more criteria are the key criteria for controlling the meshing process while the others have no significant influence on it. For example, when creating 3D volume meshes for finite element analysis, *cell radius*, *cell radius-edge* and *facet distance* are the key criteria and the other criteria are not important because they are indirectly influenced by these three criteria and imposing them as extra conditions to control the mesh will not create a better mesh, only prolongs the mesh refinement process. When creating 3D surface meshes for boundary element analysis, *facet distance*, *facet radius* and *facet angle* are the key criteria and the other criteria can safely be discarded. A more detailed discussion about each criterion follows.

4.2.1 Cell Radius Bound (Q_{cr})

Cell radius bound controls the size of the tetrahedra in the mesh. Assuming all tetrahedra in the mesh are well shaped and have an aspect ratio that is suitable for numerical stress analysis, the smaller tetrahedra potentially provide a better accuracy for finite element analysis of a continuum. On the other hand, employing

small tetrahedra throughout the mesh will increase the number of DOF and the number of the equations that should be solved by the finite element solver. The larger the number of equations, the longer it takes to solve the system of equations and the amount of resources like CPU cycles and RAM to solve the problem increases. Therefore it is best to refine the mesh in a way that while the result remains accurate at ROI, the number of DOF is kept to a minimum.

For boundary element analysis, only the surface mesh is required and the size of the tetrahedra in the volume mesh have no effect on the resulting surface mesh. Therefore, when creating the surface mesh for the boundary element analysis, this criterion is not enforced or it is relaxed by setting it to a value larger than the dimensions of the bounding box. Because all cells have a smaller cell radius than the dimensions of the domain, the meshing algorithm will not insert any new vertices based on cell radius criterion.

4.2.2 Cell Radius-edge Bound (Q_{ce})

To obtain meshes that are proper for finite element analysis, the aspect ratio of the tetrahedra in the mesh must be as low as possible. In practice, radius-edge which is a similar but weaker criterion is used instead of the aspect ratio because it is a more natural and elegant measure to use in a Delaunay meshing [69]. Cell radius-edge bound controls the quality of the tetrahedra in the mesh and its impact on the solution time is insignificant but to avoid unnecessary introduction of errors into the solution it is bounded to a maximum value. The maximum value for radius-edge bound depends on the type of the element used in the finite element stress analysis of the problem. In this study a 4-node tetrahedron element with one integration point is used and a maximum radius-edge bound of 2.0 yielded appropriate results.

On the other hand when creating surface meshes for boundary element analysis, since all tetrahedra will be ignored and only the surface facets are extracted to represent the surface mesh, this criterion is not enforced.

4.2.3 Facet Distance Bound (Q_{fd})

Facet distance is a key criterion in mesh generation when creating meshes for either finite element analysis or boundary element analysis. Facet distance bound

can be leveraged as a means to remove unnecessary details in the boundary of the model. It comes very handy when there is an staggering amount of detail in the geometry of the problem.

When creating volume meshes for the finite element analysis, this criterion can be used along with the cell radius bound to further simplify the mesh in the boundary of the domain. By increasing the value of facet distance in the areas of the model that have little or no influence on the results at the ROI, the number of elements in the vicinity of these regions is reduced, hence the the number of DOF is lowered and the efficiency of the mesh for the numerical analysis is improved. As it is shown in a case study in section 4.4.5, this criterion, when evaluated using the ROI based equation introduced earlier in section 3.6.4, reduced the number of DOF by 40% compared to when facet distance bound was assumed constant throughout the domain.

4.2.4 Facet Radius Bound (Q_{fr})

When generating surface meshes for boundary element analysis, facet radius bound is used to control the sizing of the facets of the mesh. The facets are refined further in the areas that have significant influence on the solution at ROI and coarsened elsewhere.

In mesh generation for finite element analysis, the size of the tetrahedra in the volume mesh are the major factor in improving the efficiency of the numerical analysis and surface facets have no impact of the solution accuracy or efficiency. Furthermore, facet radius bound is indirectly related to *cell radius* and *cell radius-edge* criteria and by imposing the latter two criteria, the facet radius is confined indirectly as a result. Therefore this criterion is relaxed by setting it to a value larger than the dimensions of the bounding box. This will ensure that the meshing algorithm will not insert any vertices to refine the mesh based on this criterion.

4.2.5 Facet Angular Bound (Q_{fa})

Facet angular bound controls the shape of the surface facets and is useful to control the quality of the surface mesh when creating meshes for boundary element

analysis. The appropriate minimum value of this criterion depends on the type of element that will be used in the boundary element stress analysis of the problem.

When creating volume meshes for finite element analysis, facet angular bound is not a determining factor in refining the mesh cells so it can be safely ignored.

4.3 Accuracy and Efficiency of the Framework

To determine the accuracy of the optimized model, the results obtained form the numerical solution of the optimized model should be compared with the *reference* solution. This comparison is established by measuring the relative difference between the results from the optimized and the results from the reference solution.

A rigorous measure to assess the accuracy of the results of a stress analysis is the principal stress. Principal stresses are found at the ROI and the difference between optimized and reference solutions are evaluated as follows. Stress at any point of the domain of a continuum can be defined by the nine components of a second-order tensor, $\boldsymbol{\sigma}$:

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ & \sigma_{yy} & \sigma_{yz} \\ sym & & \sigma_{zz} \end{bmatrix} \tag{4.1}$$

The principal stresses, σ_i $(i = 1, 2, 3)$, for any point in the domain are found by calculating the eigenvalues of the above matrix [3]. To provide a quantitative measure for the error introduced in the solution, the percent difference of the principal stress at a given point is calculated based on the following equation:

$$\Delta\sigma_i = \left(\frac{\sigma_i^{ref} - \sigma_i^{optimized}}{\sigma_i^{ref}} \right) \times 100 \tag{4.2}$$

where $i = 1, 2, 3$ and σ_i^{ref} is the reference principal stress, $\sigma_i^{optimized}$ is the principal stress obtained from the model with the optimized mesh and $\Delta\sigma_i$ is the percent difference of the principal stress at a point.

To obtain the reference results, the problem domain was meshed using a uniform fine mesh (Q_{cr} about 2% of the size of the bounding box) and the solution to the problem was found at the ROI. Then, in order to ensure the convergence of the

solution, the mesh was refined further and the solution at the ROI was evaluated for each refined mesh. The results from each refinement indicated the convergence of the solution (see figures 4.7 and 4.8).

To measure the effect of each meshing criterion on the resulting mesh, individual and combined effect of those criteria were studied. First the effect of *cell radius* criterion (evaluated using equation 3.16) was studied when (a) under influence of proximity factor, (b) under influence of visibility factor and (c) under combined influence of proximity and visibility factors. In this study the best result was obtained from the combined influence of proximity and visibility factors on the cell radius criterion by reducing the number of DOF from 287,022 to 32,682. That is a reduction of 89% in the number of DOF while the maximum principal stress, σ_1, was only 3.5% different from the reference result obtained from a fine uniform mesh.

In another case study, to further optimize the mesh, the effect of the *facet distance* combined with *cell radius* from the previous study was examined. This reduced the number of DOF from 287,022 to 19,404. That is a reduction of 93% in the number of DOF compared to the uniform mesh used to obtain the reference result. In this case, the principal stress, σ_1, was 19% different from the reference result which is within the acceptable range for a rock engineering problem.

To measure the efficiency of the framework, the time required for meshing, the time required for the numerical stress analysis and the total time were measured for each case study. The percentage of time savings was calculated using the following equation:

$$\Delta t = \left(\frac{t_{ref} - t_{optimized}}{t_{ref}} \right) \times 100 \tag{4.3}$$

As it is shown in section 4.4.5, applying both cell radius and facet distance criteria improved the total time required for mesh generation and finite element analysis by 95% compared to a uniform mesh.

4.4 Application of the Framework

4.4.1 Uniform Mesh (Constant Cell Radius Bound)

In this study, the cell radius bound is kept constant throughout the domain. The model is meshed several times independently using uniform meshes with differ-

ent cell radius bounds. Effect of the cell size on the number of DOF, time to solve the finite element model and the accuracy of the solution around ROI is studied. By using the *h-refinement* approach (i.e. reducing the cell radius of the uniform mesh each time), it was demonstrated that the solution is convergent.

4.4.1.1 The Model

A power cavern which is adjacent and parallel to an access tunnel was chosen for this case study. The tunnel path is straight for both tunnels. The bounding box which defines the confinement of the model is a $20m \times 20m \times 20m$ cube. The point ROI is located at $(0.0, -5.0, -0.5)$ (see figure 4.1).

	$w_{proximity}$	$w_{visibility}$	S_{min}	S_{max}	S_{cap}
Q_{cr}	–	–	–	–	$0.4 \leq S_{cap} \leq 4.0$
Q_{ce}	–	–	–	–	3.0
Q_{fd}	–	–	–	–	0.1
Q_{fr}	–	–	–	–	5.0
Q_{fa}	–	–	–	–	25°

Table 4.2: The set of parameters used for uniform meshes.

Several meshes were created using the criteria shown in table 4.2. Q_{cr}, Q_{ce}, Q_{fd}, Q_{fr} and Q_{fa} are constant in the whole domain. Q_{cr} changes from 0.4 to 4.0 for each individual mesh. Figure 4.2 shows a 3D cross section of the mesh produced by the framework when $Q_{cr} = 0.4$. Figures 4.3 and 4.4 shows cross section of two other meshes made from the same geometry using different values for Q_{cr}.

For finite element stress analysis, the model is assumed to be under gravity loading (the weight of the rock mass). As for the boundary conditions, the horizontal DOF (i.e. horizontal displacement of nodes) on the vertical sides of the bounding box are restricted and nodes on the bottom of the bounding box are encastré.

A static perturbation load-type is used to apply the loading and *SIMULIA Abaqus FEA* [75] software was used to perform the finite element analysis. A 4-node tetrahedron element was used to model the problem. Poisson's ratio of the rock mass was 0.2, modulus of elasticity was equal to 40,120 MPa and density of the rock mass was 1,980 kg/m^3.

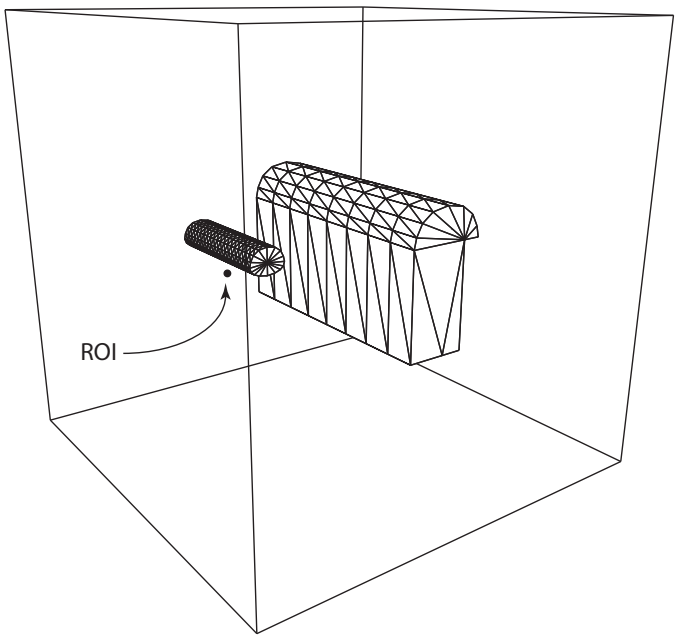

Figure 4.1: Geometry of the excavations and the bounding box

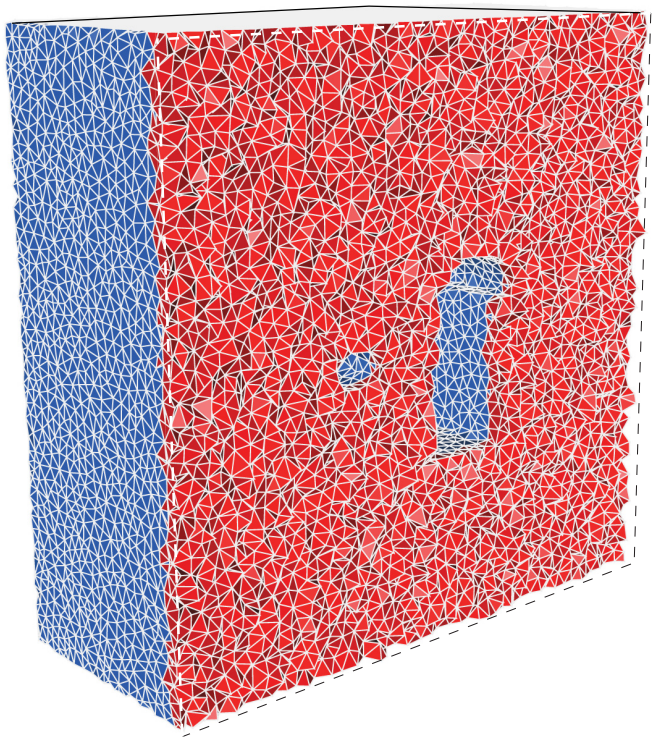

Figure 4.2: 3D cross section of the uniform mesh produced by the framework ($Q_{cr} = 0.4$)

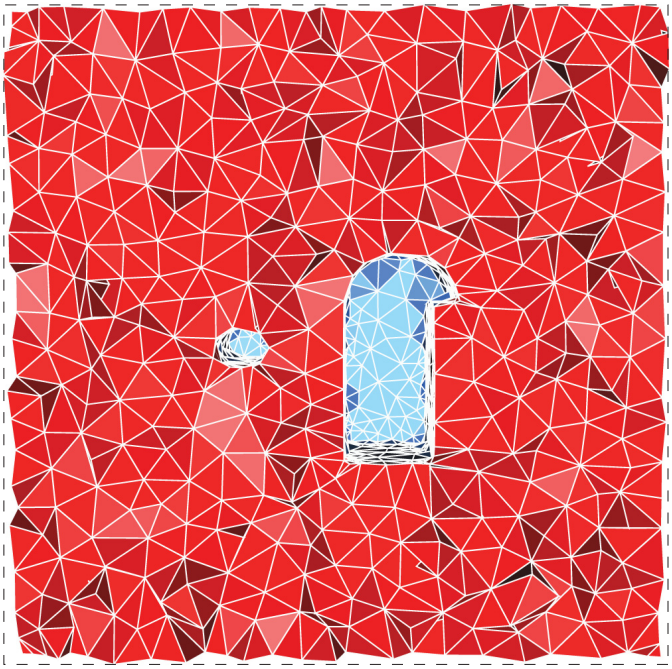

Figure 4.3: Cross section of the uniform mesh ($Q_{cr} = 1.0$)

Figure 4.4: Cross section of the uniform mesh ($Q_{cr} = 4.0$)

4.4.1.2 Time Required for Finite Element Analysis

To study the effect of cell radius on the time required for finite element analysis, the domain was meshed several times using different cell sizes. Cell radius, Q_{cr}, was kept uniform across the domain and the same loading and boundary conditions were applied to the model each time. Other meshing criteria were kept constant across the domain as seen in table 4.2.

Figure 4.5 shows the relation between cell radius bound and number of generated cells, vertices and DOF.

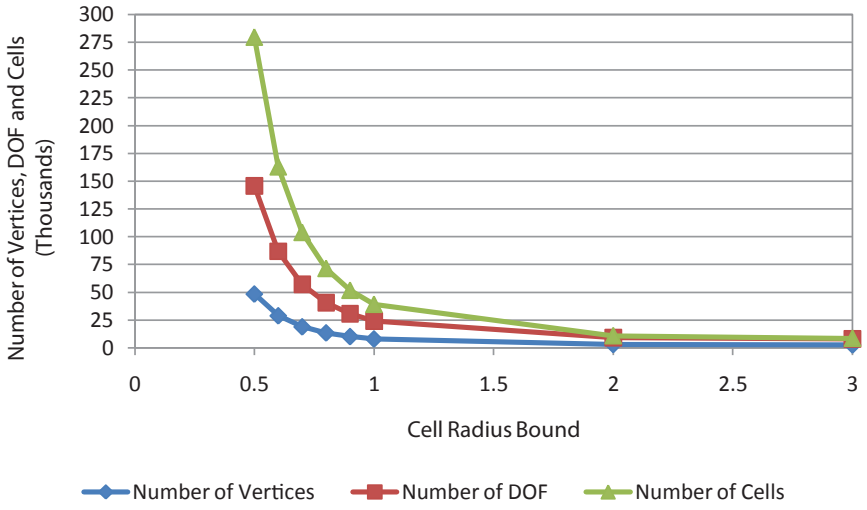

Figure 4.5: Relation between the cell radius bound and the number of generated cells, vertices and DOF

The trend lines are:

$$N_{cell} = 51858 \left(\frac{1}{Q_{cr}}\right)^{1.952} \text{ and } R^2 = 0.9588 \tag{4.4}$$

and

$$N_{vertices} = 32954 \left(\frac{1}{Q_{cr}}\right)^{1.596} \text{ and } R^2 = 0.9305 \tag{4.5}$$

and

$$N_{DOF} = 10985 \left(\frac{1}{Q_{cr}}\right)^{1.596} \text{ and } R^2 = 0.9305 \tag{4.6}$$

where Q_{cr} is the cell radius bound, N_{cell} is number of cells, $N_{vertices}$ is number of vertices and N_{DOF} is number of DOF.

These relations would be more generic if cell radius is expressed relative to the dimensions of the whole problem. For this purpose, we divide cell radius bound by the space diagonal of the bounding box to get the dimensionless quantity, *relative cell radius*:

$$q = Q_{cr}/D \tag{4.7}$$

For a bounding box with sides a, b and c, the space diagonal is equal to:

$$D = \sqrt{a^2 + b^2 + c^2} \tag{4.8}$$

In this problem $a = b = c = 20.0$ therefore $D = 34.641$. Substituting equation 4.7 into equation 4.6 we get:

$$N_{DOF} = 38.33 \left(\frac{1}{q}\right)^{1.596} \tag{4.9}$$

where $0 < q < 1$. Equation 4.9 is useful for estimating the number of DOF in terms of q, the relative cell radius. In section 4.4.2 it is shown how reaching a desirable accuracy at the ROI impacts choosing a proper cell size for the mesh.

Figure 4.6 shows the relation between the number of DOF and the time required for finite element analysis. The trend line for this diagram is:

$$T_{FEA} = 2 \times 10^{-6} \cdot {N_{DOF}}^{1.579} \text{ and } R^2 = 0.9996 \tag{4.10}$$

Since the main goal is to reduce time to solution when creating the mesh, it is desirable to find a relation between cell radius bound and the time required for finite element analysis. Substituting equation 4.9 into equation 4.6 we get:

$$T_{FEA} = 6.33 \times 10^{-4} \left(\frac{1}{q}\right)^{2.52} \tag{4.11}$$

Equation 4.11 predicts that if the cell radius is about 0.1% of the diameter of the bounding box (i.e. $q = 0.001$), there will be roughly 2,350,000 degrees of freedom in a uniform mesh. Solving a problem of this magnitude can take considerably longer that the 6 hours predicted by equation 4.11 because the problem description (i.e. the stiffness matrix in the case of a finite element analysis) does not fit in the RAM

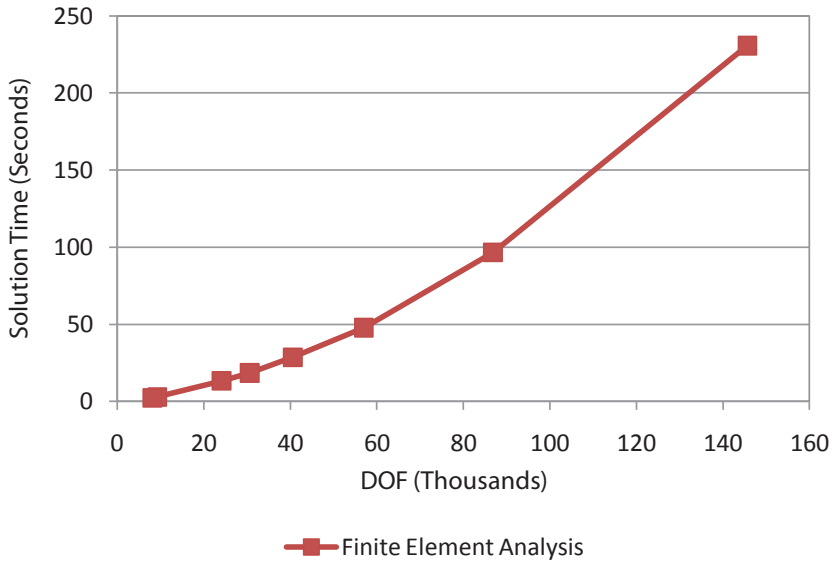

Figure 4.6: Relation between the DOF and the time required for finite element analysis

available on a computer used in today's average engineering firm. A more detailed discussion about scalability of solution is provided in section 4.4.2. It is worth noting that the equations derived here cannot be applied to an arbitrary geometry and are used to illustrate how the number of DOF affects time to solution for this problem.

4.4.1.3 Accuracy of the Results

The smaller the cell radius, the better the accuracy of the finite element results will be. To find out how cell size affects the accuracy of the results, the model was meshed using different cell radiuses ranging between $0.4 \leq Q_{cr} \leq 3.0$. The finite element analysis was repeated for each mesh using the same loading and boundary conditions. Then the value of stress was measured at the ROI to conduct a comparison. Figure 4.7 shows the result of this comparison. It is observed that by reducing the cell radius, the stress converges at a value about 920 kPa.

To Measure the accuracy of the solution, equation 4.2 was used to evaluate the percent difference in stress values at the ROI for each mesh. Figure 4.8 shows the result of this comparison and the trend line is:

$$\Delta\sigma_{vm} = 0.0518 Q_{cr} + 0.0135 \text{ and } R^2 = 0.5911 \tag{4.12}$$

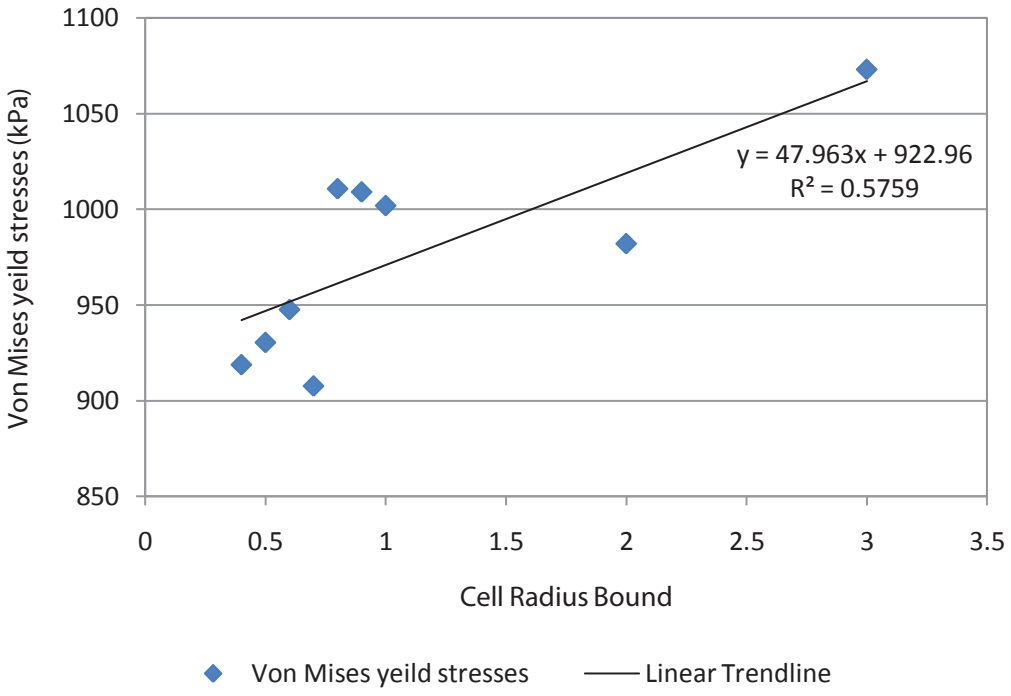

Figure 4.7: Relation between the cell radius and the σ_{vm} at the ROI

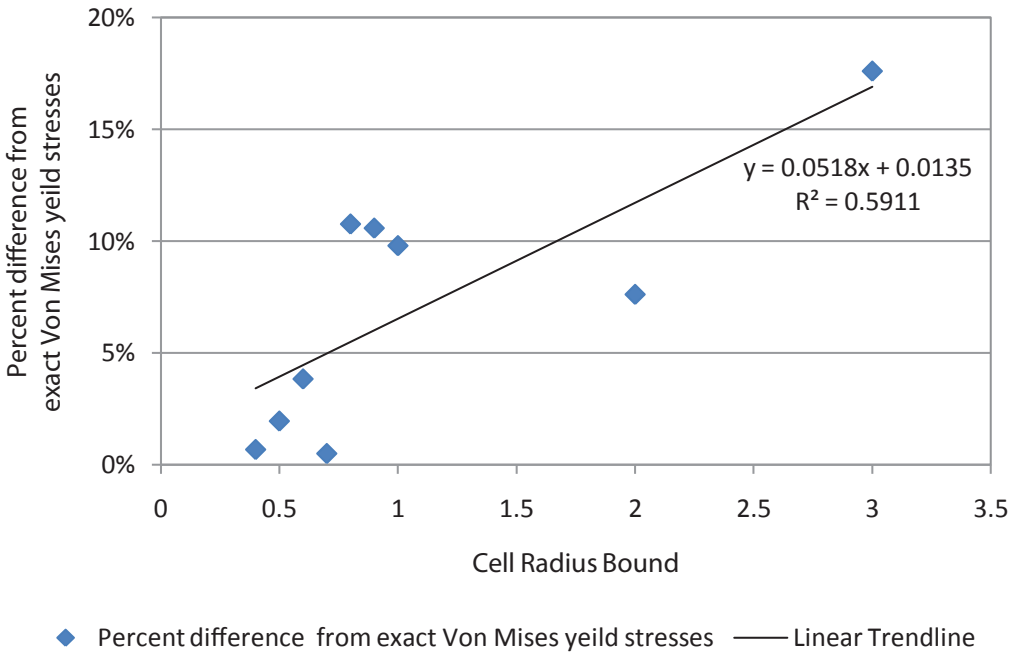

Figure 4.8: Relation between cell radius and percent difference from the reference stress value at the ROI

where $\Delta\sigma_{vm}$ is the percent difference from the reference value of Von Mises stress.

4.4.2 Scalability of a Solution Using Uniform Meshes

Consider a mining engineering problem with a bounding box size of $500m \times 500m \times 500m$. To reach an accuracy of 5% ($\Delta\sigma_{vm} = 0.05$) in the stress analysis, the approximate number of DOF in a uniform mesh can be roughly estimated using equations 4.9, 4.11 and 4.12 as: $N_{DOF} \approx 2,000,000$. If all the information required for solving this problem could fit in the RAM of the computer, it would take about 5 hours for the finite element solver to solve the linear version of the problem. As it is shown subsequently, this is not the case. The size of this problem is orders of magnitude larger than what an ordinary computer used at an engineering firm can handle.

The number of equations to be solved for this problem is ${N_{DOF}}^2 = 4 \times 10^{12}$. According to IEEE 754 standard for floating-point arithmetic [76], a *double precision floating point* number requires 64 bits (i.e. 8 bytes) of memory space. Therefore, the space required for storing the system of equations can be estimated to about $4 \times 10^{12} \times 8 = 3.2 \times 10^{13}$ bytes or 29 terabytes which is about 10 times of the size of the largest hard disk drive you can find in the consumer market today. Therefore persisting the data that represent a problem of this size on a hard disk would be the first obstacle in everyday practices in an engineering firm.

On the other hand, the amount of memory available on computers used in engineering firms can be around 32 gigabytes at best (the average amount of RAM available on a personal computer is about 4 gigabytes today). So, in the best scenario, the available memory is about 1000 times smaller than the size of the problem. In these situations a frontal solver [77] or a multifrontal solver [78] is used. Frontal solver is a variant of Gauss elimination approach for solving system of equations and it automatically avoids a large number of operations involving zero terms that usually appear in the system of equations formed for finite element analysis.

Frontal solvers break down the problem into smaller blocks that fit in the available RAM and load just enough data from the hard disk into memory at each

time. This process enables solution of very large system of equations but causes *disk swapping* and dramatically slows down the speed of the solver [53] and increases the *time to solution*. It is worth mentioning that these numbers are for the linear analysis of the problem. For a non-linear analysis, the time to solution will increase significantly as the problem needs to be solved iteratively.

4.4.3 Cell Radius Bound Influenced by Proximity Factor

In this section the effects of proximity (as defined in section 3.6.4.1) on cell radius, number of DOF and the solution time are studied. The cell radius is evaluated based on the proposed sizing function in equation 3.16. The goal is to reduce the number of DOF while keeping the results accurate enough at the ROI.

4.4.3.1 The Model

The model consists of 3 excavations: a tunnel that intersects with a power cavern and a third access tunnel that has a smaller profile and is parallel to the power cavern (see figures 4.9, 4.10 and 4.11). The bounding box which defines the confinement of the model is chosen to be a $30m \times 30m \times 30m$ cube. The point ROI is located at $(-1.0, -0.5, -3.7)$.

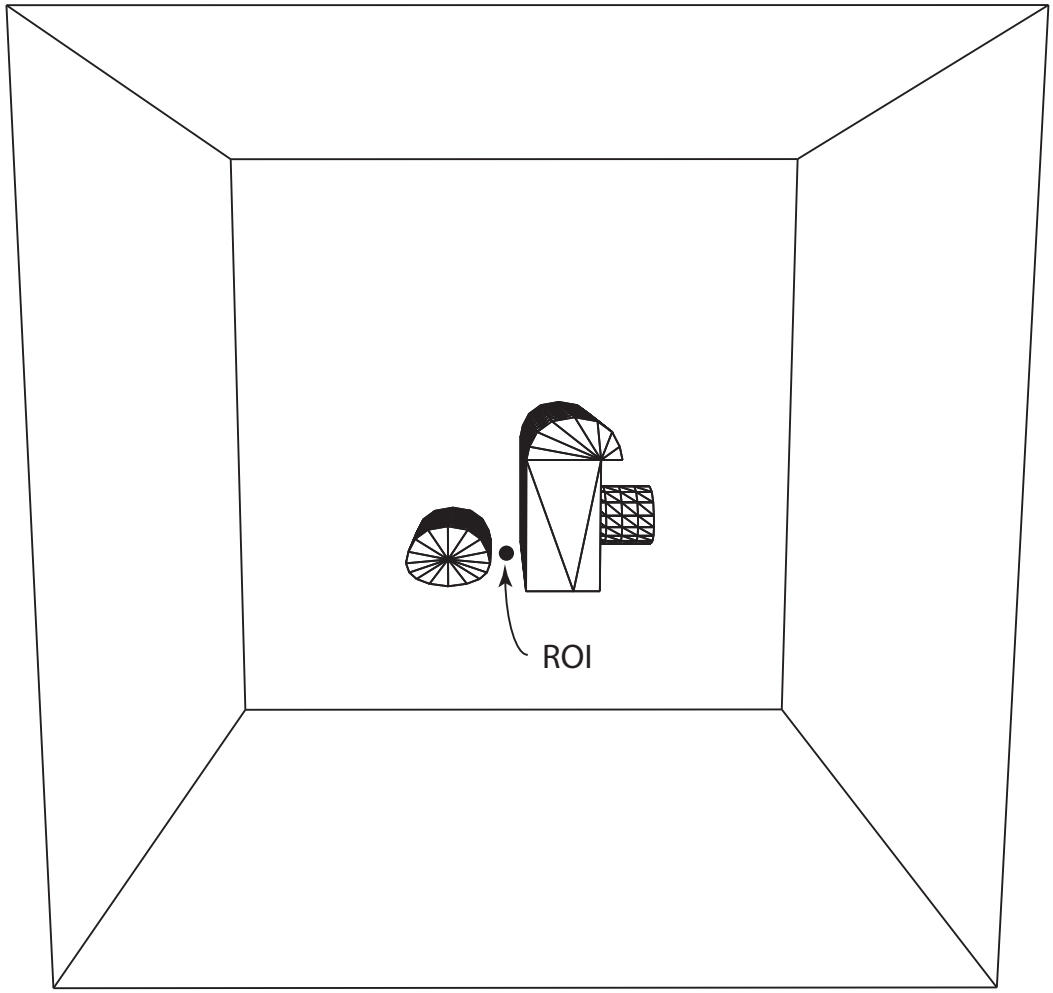

Figure 4.9: The geometry of the tunnels and the bounding box: front view

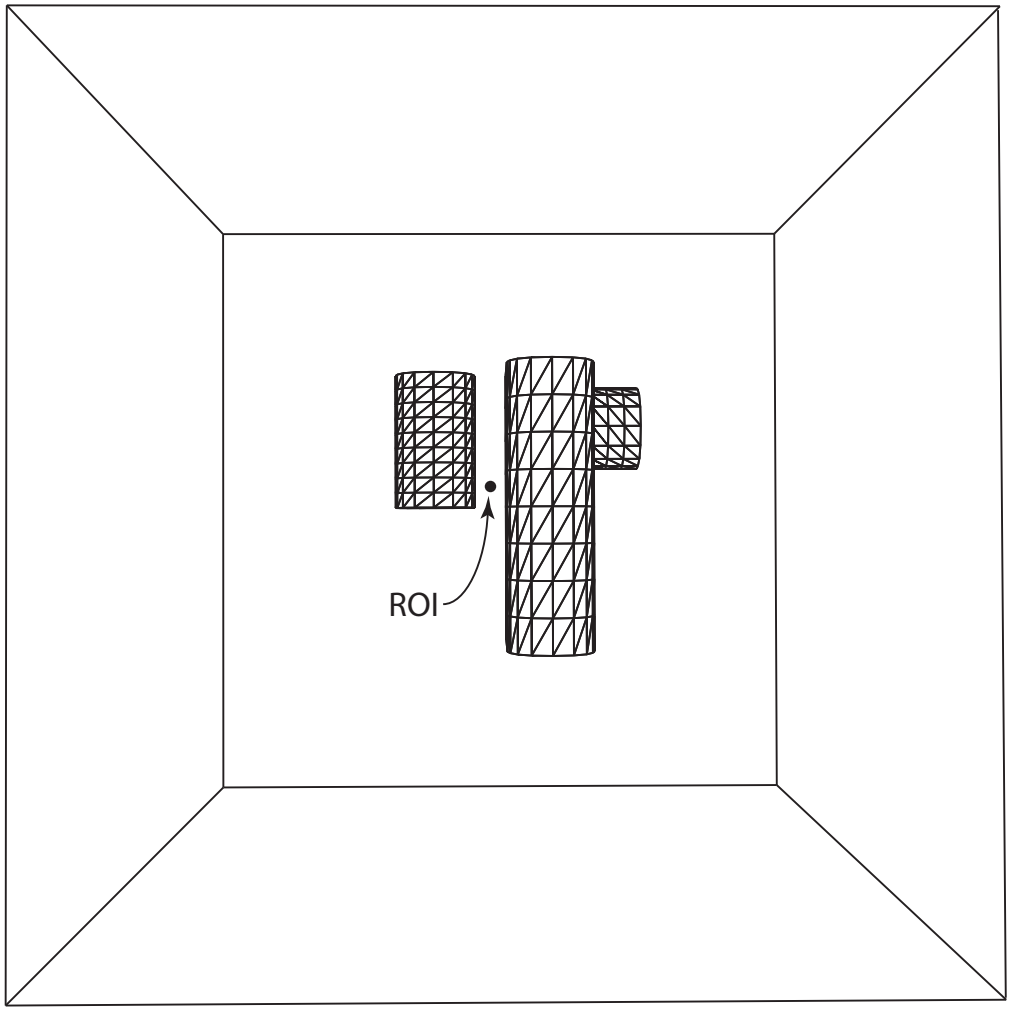

Figure 4.10: The geometry of the tunnels and the bounding box: top view

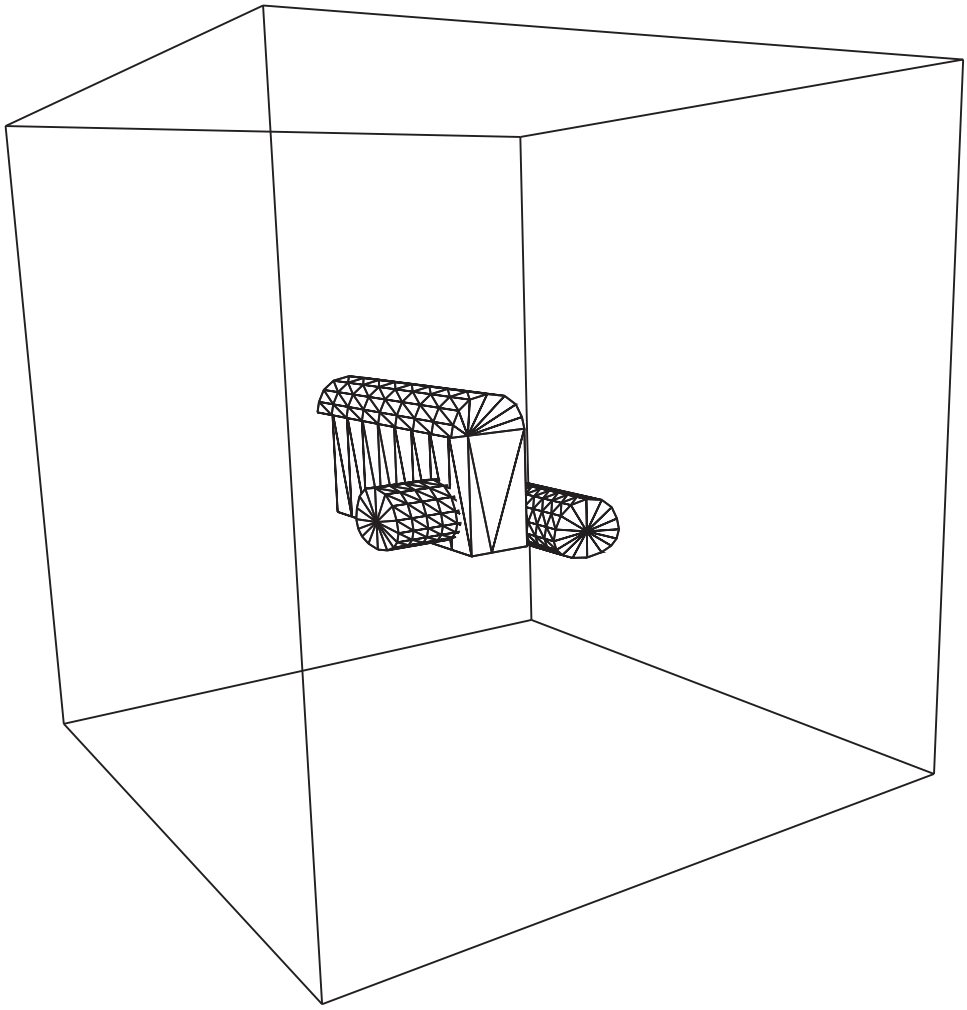

Figure 4.11: The geometry of the tunnels and the bounding box: side view

4.4.3.2 Meshing and Finite Element Analysis

The domain was meshed first using a uniform cell radius bound (Q_{cr}) and then using a variable cell radius bound calculated from equation 3.16 to obtain an optimum mesh.

	$w_{proximity}$	$w_{visibility}$	S_{min}	S_{max}	S_{cap}
Q_{cr}	–	–	–	–	0.7
Q_{ce}	–	–	–	–	2.0
Q_{fd}	–	–	–	–	0.1
Q_{fr}	–	–	–	–	5.0
Q_{fa}	–	–	–	–	25°

Table 4.3: The set of parameters used for the uniform mesh.

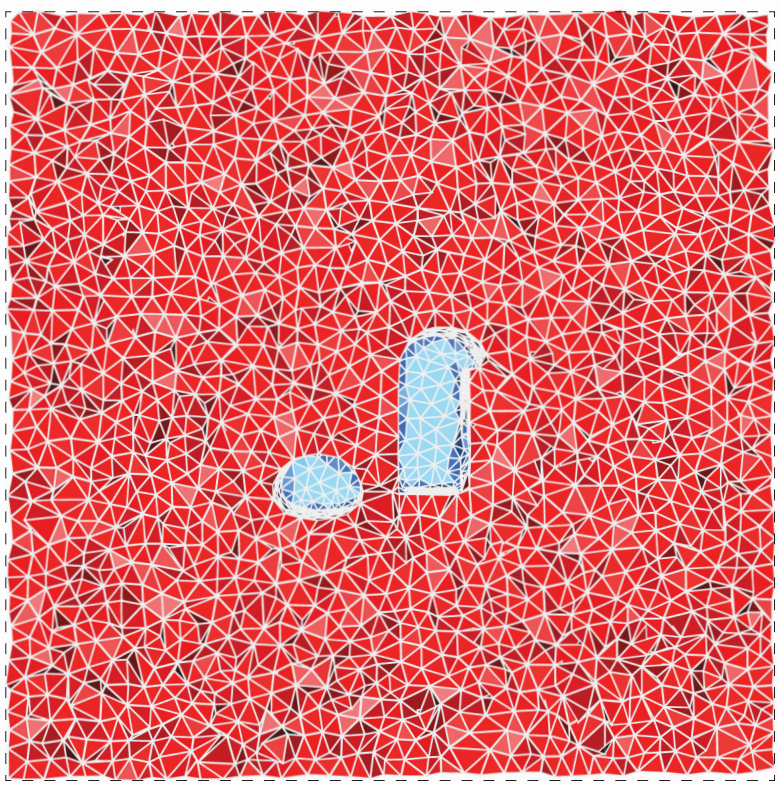

Figure 4.12: Cross section of the uniform mesh (meshing parameters chosen according to table 4.3)

The parameters used for meshing are shown in table 4.3 and table 4.4 respectively. Cell radius is constant $Q_{cr} = 0.7$ for the uniform mesh and for the optimized mesh, Q_{cr} is evaluated using the ROI based formula with the following parameters:

$S_{min} = 0.4$, $S_{max} = 10.0$, $S_{cap} = 4.0$, $w_{proximity} = 1.0$ and $w_{visibility} = 0.0$. Other meshing criteria were kept constant across the domain as follows: cell radius-edge $Q_{ce} = 2.0$, facet radius $Q_{fr} = 5.0$, facet distance $Q_{fd} = 0.1$, facet angle $Q_{fa} = 25°$. In both cases the mesh was smoothened to remove slivers and improve the mesh quality for the finite element analysis.

	$w_{proximity}$	$w_{visibility}$	S_{min}	S_{max}	S_{cap}
Q_{cr}	1.0	0.0	0.4	10.0	4.0
Q_{ce}	–	–	–	–	2.0
Q_{fd}	–	–	–	–	0.1
Q_{fr}	–	–	–	–	5.0
Q_{fa}	–	–	–	–	25°

Table 4.4: The set of parameters used for the mesh with variable cell radius.

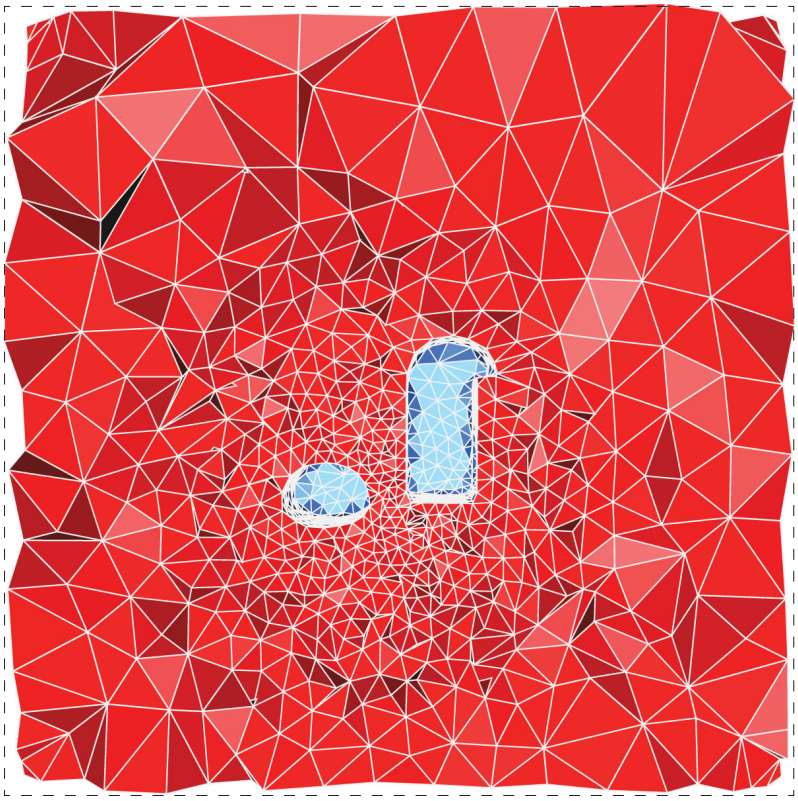

Figure 4.13: Cross section of the optimized mesh (meshing parameters chosen according to table 4.4)

Figures 4.12 and 4.13 show cross sections of the mesh with plane $x = -1$ that passes through the point ROI. The model is assumed to be under geostatic loading that varies linearly from top to bottom of the bounding box. The vertical component of the load starts at $B_{z,top} = -2.0MPa$ on the top and reaches $B_{z,btm} = -2.6MPa$ at the bottom of the bounding box (negative value indicates that direction of the force is in the opposite direction of the z-axis). The horizontal components of the load are functions of the vertical load: $B_x = K_x \cdot B_z$ and $B_y = K_y \cdot B_z$ where $K_x = K_y = 0.85$. For the boundary conditions, the horizontal translation of the nodes on the vertical sides of the cube are restricted and the nodes on the bottom of the cube are encastré. *SIMULIA Abaqus FEA* software was used to perform the finite element analysis. A 4-node tetrahedron element was used to model the problem. Poisson's ratio was 0.2 and modulus of elasticity of the rock mass was equal to 40,120 MPa. Density of the rock mass was 1,980 kg/m^3.

Figure 4.14 and figure 4.15 show distribution of maximum principal stress on cross section of plane $x = -1$ that passes through ROI. The stress distribution in the model with optimized mesh generally follows the same trend as the model with uniform mesh, specially in the vicinity of the ROI which is the desired behavior. A quantitative comparison is given in table 4.5. Figure 4.16 and figure 4.17 show distribution of minimum principal stress on cross section of plane $x = -1$ that passes through ROI. Again, the stress distribution in the model with optimized mesh the same trend as the model with uniform mesh.

4.4.3.3 Accuracy and Efficiency

Table 4.5 shows a side by side comparison of important values extracted for each model. In the optimized model, number of DOF is only 14% of the model with uniform mesh and the stress analysis of the optimized model clearly finishes faster. In fact it is 20 times faster than the model with uniform mesh. The computations are performed on a PC with 3GB of RAM which is normal for personal computers that are used in the consulting engineering firms.

The difference in maximum principal stresses (σ_1), minimum principal stresses (σ_3) and Von Mises yield stress (σ_{vm}) are 21%, 7% and 0.8% respectively. This much

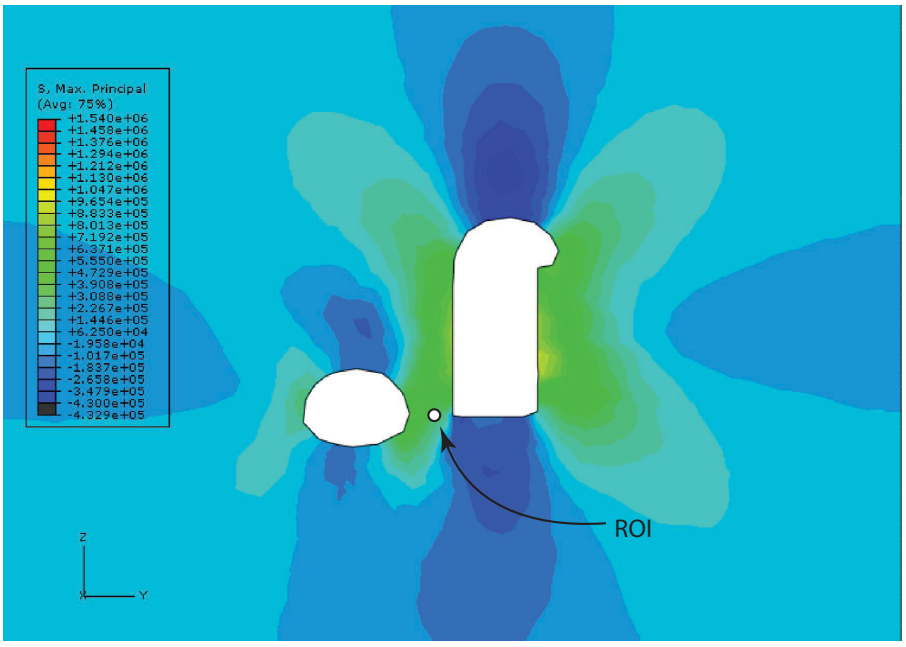

Figure 4.14: Distribution of the maximum principal stress in the model with *uniform mesh*

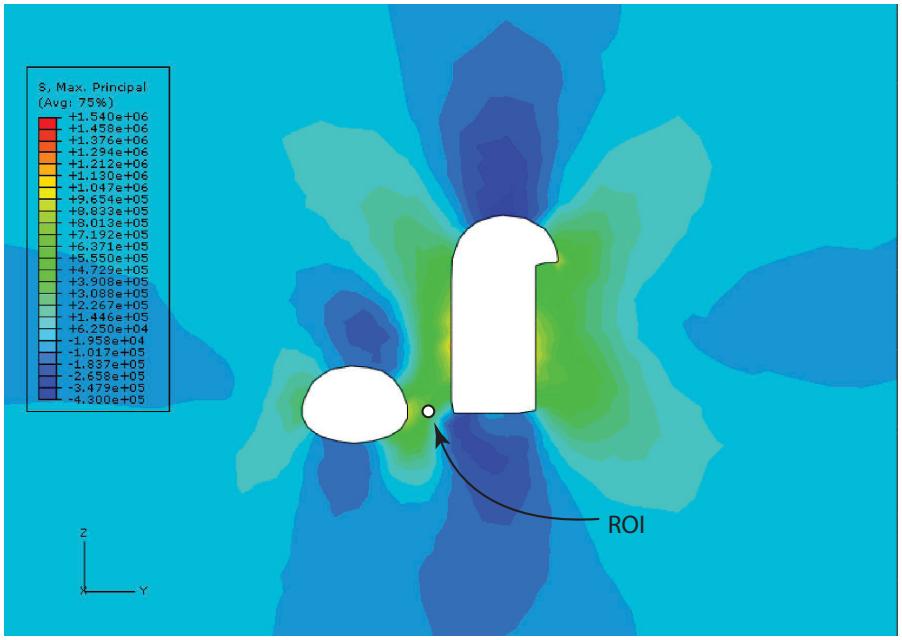

Figure 4.15: Distribution of the maximum principal stress in the model with *optimized mesh*

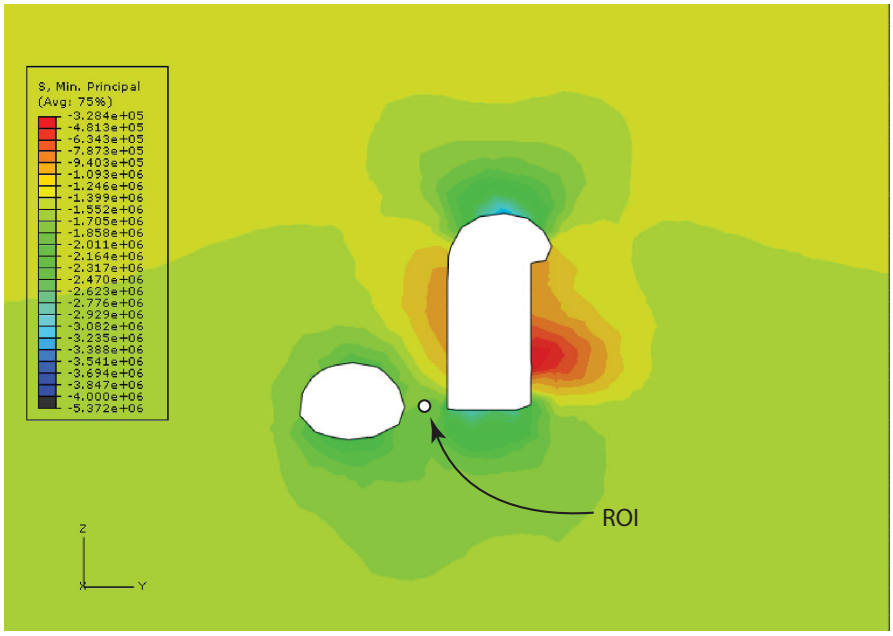

Figure 4.16: Distribution of the minimum principal stress in the model with *uniform mesh*

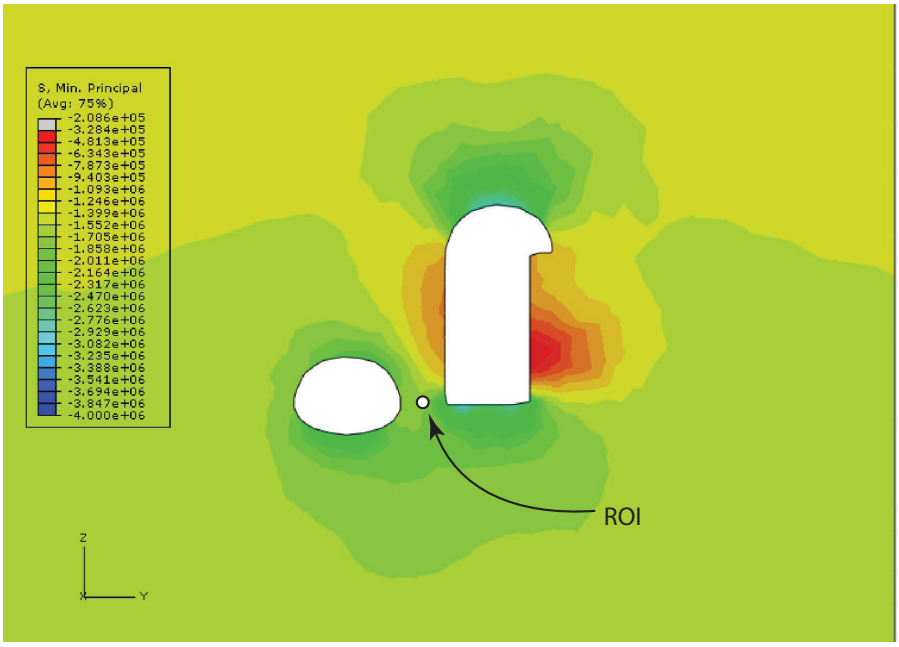

Figure 4.17: Distribution of the minimum principal stress in the model with *optimized mesh*

accuracy is quite acceptable for rock engineering problems where accuracy of the input data are within the same range. In this case, compared to the uniform mesh, the results of the optimized mesh are even more accurate in the vicinity of the ROI because the mesh is about 40% finer in that area.

Table 4.6 also shows the time savings made by optimizing the mesh. The optimized mesh took 79.5% less time to get created and the finite element analysis of it took 95.4% less time to finish. The total time saving, Δt_{total}, compared to the uniform mesh was 85.2%.

	# DOF	t_{mesh}(sec)	t_{fea}(sec)	σ_{vm}(kPa)	σ_1(kPa)	σ_3(kPa)
Uniform mesh	181755	688	392	1915.7	440.3	-1710.3
Optimized mesh	26154	141	18	1931.2	562.4	-1600.0

Table 4.5: Comparison of the results for uniform and optimized mesh

	Δt_{mesh}	Δt_{fea}	Δt_{total}	$\Delta\sigma_{vm}$	$\Delta\sigma_1$	$\Delta\sigma_3$
Optimized mesh	79.5%	95.4%	85.2%	0.8%	21.0%	7.0%

Table 4.6: Percent difference relative to the uniform mesh

4.4.4 Cell Radius Bound Influenced by Visibility Factor

The effects of the visibility (as defined in section 3.6.4.2) on cell radius bound are studied. The intent is to visually demonstrate how the visibility factor affects the final mesh.

Cell radius bound (Q_{cr}) is evaluated based on equation 3.16. The parameters used to set up the formula were: $S_{min} = 1.0$, $S_{max} = 10.0$, $S_{cap} = 5.0$, $w_{proximity} = 0.0$ and $w_{visibility} = 1.0$. Other meshing criteria were kept constant across the domain as follows: cell radius-edge $Q_{ce} = 2.0$, facet radius $Q_{fr} = 5.0$, facet distance $Q_{fd} = 0.1$, facet angle $Q_{fa} = 25^\circ$ (see table 4.7).

Figure 4.18 show cross section of the mesh at plane $x = -1$ which passes through the point ROI. It can be observed that the mesh in areas of the model that are hidden from ROI is radically coarsened and the areas that are visible from ROI are refined further which is the expected behavior.

	$w_{proximity}$	$w_{visibility}$	S_{min}	S_{max}	S_{cap}
Q_{cr}	0.0	1.0	1.0	10.0	5.0
Q_{ce}	–	–	–	–	2.0
Q_{fd}	–	–	–	–	0.1
Q_{fr}	–	–	–	–	5.0
Q_{fa}	–	–	–	–	25°

Table 4.7: The set of parameters used for meshing.

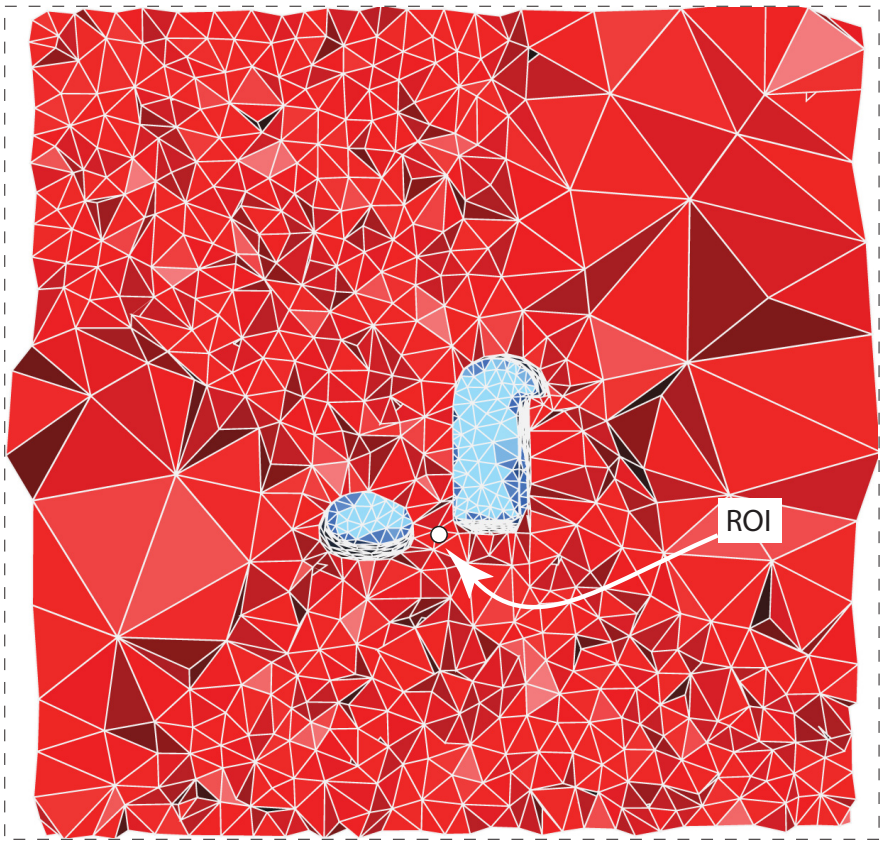

Figure 4.18: Cross section of mesh at the ROI. Cell radius is influenced by the visibility factor

Although this helps in legitimately reducing the number of DOF globally, if it is accompanied by the proximity factor it will produce a more balanced and effective mesh (see section 4.4.5). Visibility factor is particularly helpful in detecting areas that are geometrically in proximity of the ROI but are hidden behind an excavation (i.e. not visible to the ROI) and have little influence on the stresses at the ROI. Therefore, visibility factor assists in coarsening the mesh in these areas, hence reducing the number of DOF.

4.4.5 Optimized Mesh

To overcome the obstacles mentioned in section 4.4.2, the proposed framework is used to produce optimized meshes that fit within the available computational resources. Optimized meshes allow solution of large size problems using the currently available computational resources that previously were not able to tackle problems of this size. A combination of proximity and visibility factors affecting cell radius bound, Q_{cr}, and facet distance, Q_{fd}, are used for producing the optimized meshes. Several meshes are produced for the same model using different sets of parameters and the effects of the choice of parameters is discussed for each case.

4.4.5.1 The Model

The model helps to better observe the combined effects of proximity and visibility factors. Figure 4.19 shows the model. A narrow excavation, located in between the power cavern (right) and the adjacent tunnel (left), hides the power cavern from the ROI located at $(-1.0, -0.5, -3.7)$.

The model is under geostatic loading that varies linearly from top to the bottom of the bounding box. The vertical component of the load starts at $B_{z,top} = -2 \times 10^6 Pa$ on the top and reaches $B_{z,btm} = -2.6 \times 10^6 Pa$ at the bottom of the bounding box (negative value indicates that direction of the force is in the opposite direction of the z-axis). The horizontal components of the load are factors of the vertical load: $B_x = K_x \cdot B_z$ and $B_y = K_y \cdot B_z$ where $K_x = K_y = 0.85$. For the boundary conditions, the horizontal translation of the nodes on the vertical sides of the cube are restricted and the nodes on the bottom of the cube are encastré. *SIMULIA Abaqus FEA* software was used to perform the finite element analysis. A

4-node tetrahedron element was used to model the problem. Poisson's ratio is 0.2 and modulus of elasticity of the rock mass is equal to 40,120 MPa. Density of the rock mass is 1,980 kg/m^3.

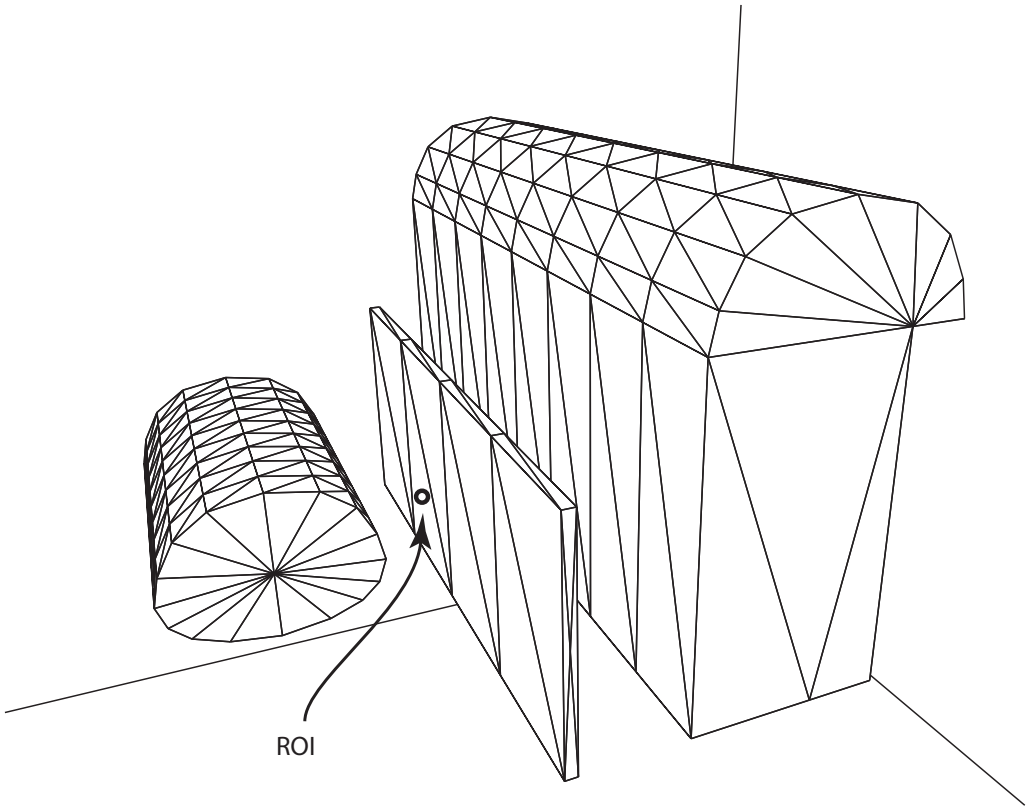

Figure 4.19: Geometry of the model. An narrow excavation (middle) hides the power cavern (right) from ROI

4.4.5.2 Meshing

Figures 4.20, 4.21, 4.22 and 4.23 show cross sections of the three different meshes created from the same model based on different criteria.

In the first three cases the mesh is optimized subject to the cell radius bound, Q_{cr}, computed from the ROI based equation (equation 3.16) by varying values for proximity ($w_{proximity}$) and visibility ($w_{visibility}$) while the rest of the criteria were kept constant. In case #4, the mesh is optimized subject to both cell radius bound, Q_{cr}, and facet distance bound Q_{fd} evaluated from equation 3.16. The meshing parameters are tabulated in tables 4.8, 4.9, 4.10 and 4.11.

Case #1: In the first case, the mesh is optimized based on cell radius bound, Q_{cr}, evaluated from the ROI based equation where $w_{proximity} = 1.0$ and $w_{visibility} = 0.0$. The proximity factor is dominant and contribution of the visibility factor is null. It can be seen that the cell radius is increasing at a quadratic rate regardless of the visibility from the ROI (see table 4.8 and figure 4.20).

Case #2: In the second case, the mesh is optimized based on cell radius bound, Q_{cr}, evaluated from the ROI based equation where $w_{proximity} = 1.0$ and $w_{visibility} = 0.5$. The visibility factor has significantly influenced the mesh. It can be observed that the mesh is radically coarsened in areas of the model that are hidden from ROI and it is refined further in the areas that are visible from the ROI which is the expected behavior. As it is shown in case #3, when the proximity and visibility are used in a balanced way they produce a more efficient mesh.

Visibility factor is particularly helpful in detecting areas that are geometrically in proximity of the ROI but are hidden behind an excavation (i.e. not visible to the ROI) and have little influence on the stresses at the ROI. Therefore, visibility factor assists in coarsening the mesh in these areas, hence reducing the number of DOF (see table 4.9 and figure 4.21).

Case #3: Here the mesh is again optimized subject to cell radius bound, Q_{cr}, and $w_{proximity} = 1.0$ and $w_{visibility} = 0.1$. It can be observed that there is a better balance between the influence of the proximity factor and visibility factor. There are no abrupt changes in the size of cells as there were in case #2. The produced mesh has the least number of DOF among the first three cases (see table 4.10 and figure 4.22).

Case #4: Facet distance criteria can help reduce the details on the boundary of the geometry. To investigate this hypothesis, the mesh optimized in case #3 using the cell radius criterion, Q_{cr}, is further optimized by incorporating the effects of facet distance, Q_{fd}, and evaluating it using equation 3.16. Table 4.11 shows the values used for each parameter to drive the meshing algorithm and figure 4.23 shows the resulting mesh.

	$w_{proximity}$	$w_{visibility}$	S_{min}	S_{max}	S_{cap}
Q_{cr}	1.0	0.0	0.3	10.0	5.0
Q_{ce}	–	–	–	–	2.0
Q_{fd}	–	–	–	–	0.1
Q_{fr}	–	–	–	–	5.0
Q_{fa}	–	–	–	–	25°

Table 4.8: Parameters used for case #1

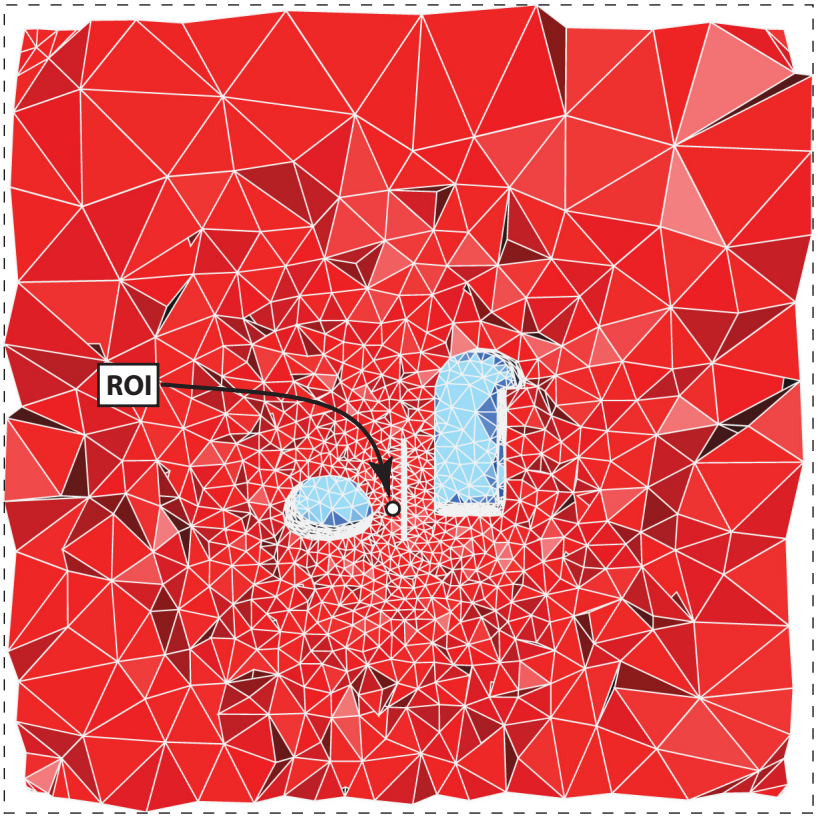

Figure 4.20: Case #1: Cross section of the mesh optimized subject to cell radius bound, $w_{proximity} = 1.0$ and $w_{visibility} = 0.0$

	$w_{proximity}$	$w_{visibility}$	S_{min}	S_{max}	S_{cap}
Q_{cr}	1.0	0.5	0.3	10.0	5.0
Q_{ce}	–	–	–	–	2.0
Q_{fd}	–	–	–	–	0.1
Q_{fr}	–	–	–	–	5.0
Q_{fa}	–	–	–	–	25°

Table 4.9: Parameters used for case #2

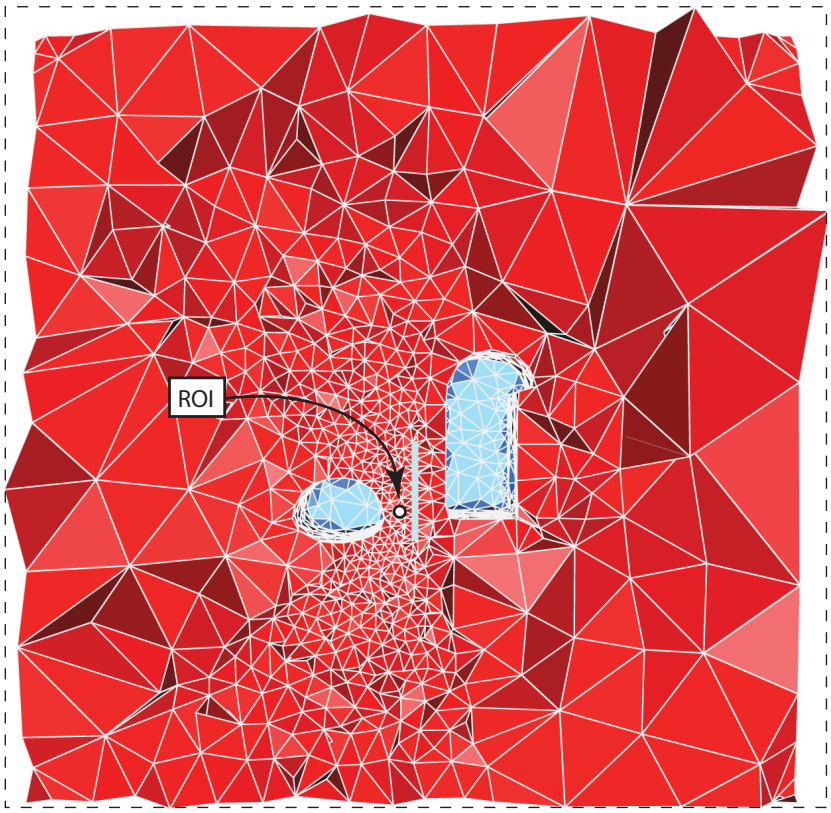

Figure 4.21: Case #2: Cross section of the mesh optimized subject to cell radius bound, $w_{proximity} = 1.0$ and $w_{visibility} = 0.5$

	$w_{proximity}$	$w_{visibility}$	S_{min}	S_{max}	S_{cap}
Q_{cr}	1.0	0.1	0.3	10.0	5.0
Q_{ce}	–	–	–	–	2.0
Q_{fd}	–	–	–	–	0.1
Q_{fr}	–	–	–	–	5.0
Q_{fa}	–	–	–	–	25°

Table 4.10: Parameters used for case #3

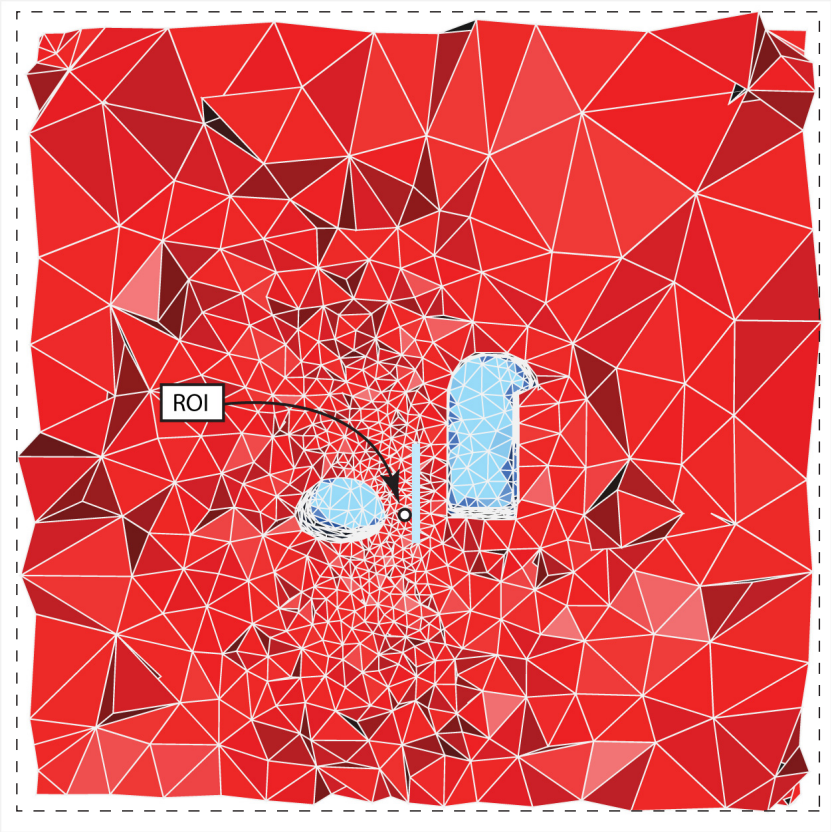

Figure 4.22: Case #3: Cross section of the mesh optimized subject to cell radius bound, $w_{proximity} = 1.0$ and $w_{visibility} = 0.1$

	$w_{proximity}$	$w_{visibility}$	S_{min}	S_{max}	S_{cap}
Q_{cr}	1.0	0.1	0.3	10.0	5.0
Q_{ce}	–	–	–	–	2.0
Q_{fd}	1.0	0.1	0.05	2.0	1.0
Q_{fr}	–	–	–	–	5.0
Q_{fa}	–	–	–	–	25°

Table 4.11: Parameters used for case #4

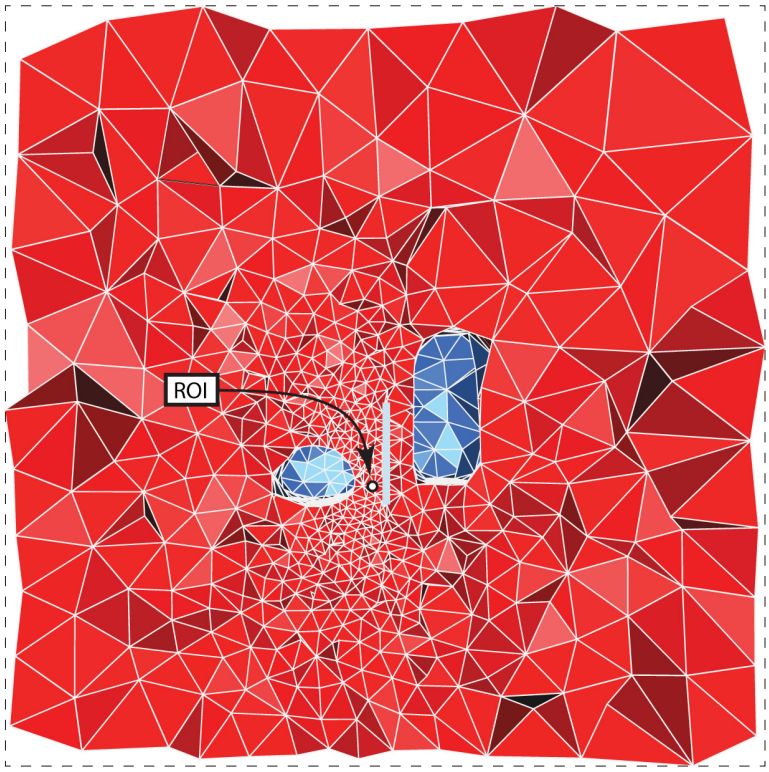

Figure 4.23: Case #4: Cross section of the mesh optimized subject to cell radius bound and facet distance bound, $w_{proximity} = 1.0$ and $w_{visibility} = 0.1$

Table 4.12 shows the comparison of the results for the uniform mesh and the four optimized meshes. t_{mesh} is the time required to generate the mesh, t_{fea} is the time required for finite element analysis, σ_1 is the maximum principal stress, σ_3 is the minimum principal stress and σ_{vm} is the Von Mises yield stress.

	# DOF	t_{mesh}(sec)	t_{fea}(sec)	σ_{vm}(kPa)	σ_1(kPa)	σ_3(kPa)
Uniform Mesh	287022	1125	875.0	1698.7	568.1	-1372.8
Case #1	37794	220	33.3	1638.3	551.2	-1320.7
Case #2	38850	218	34.9	1680.4	487.0	-1430.9
Case #3	32682	200	25.1	1648.2	548.4	-1332.6
Case #4	19404	84.5	15.3	1781.6	458.9	-1574.3

Table 4.12: Comparison of the results for different meshes derived from the same model

The lowest number of DOF belongs to case #4 where there is a good balance between $w_{proximity}$ and $w_{visibility}$. By combining the effects of cell radius bound, Q_{cr}, and facet distance, Q_{fd}, the number of DOF is reduced by 40% and the time required for finite element analysis is reduced by 39% compared to case #3 where the mesh optimized was only subject to cell radius criterion.

	Δt_{mesh}	Δt_{fea}	Δt_{total}	$\Delta\sigma_{vm}$	$\Delta\sigma_1$	$\Delta\sigma_3$
Case #1	80.4%	96.2%	87.3%	3.6%	3.0%	3.8%
Case #2	80.6%	96.2%	87.3%	1.1%	14.3%	4.2%
Case #3	82.2%	97.1%	88.7%	3.0%	3.5%	2.9%
Case #4	92.5%	98.3%	95.0%	4.9%	19.2%	14.7%

Table 4.13: Percent difference relative to the uniform mesh

Table 4.13 shows the percent difference of time and stresses relative to the uniform mesh. Δt_{mesh}, Δt_{fea} and Δt_{total} calculated using equation 4.3 are percent difference of the time required for meshing, finite element analysis and the total time respectively. $\Delta\sigma_1$, $\Delta\sigma_3$ and $\Delta\sigma_{vm}$ are evaluated based on equation 4.2 and are percent difference of maximum principal stress, minimum principal stress and Von Mises yield stress respectively. The tabulated results show that the mesh optimized by applying both cell radius and facet distance criteria (case #4) improves the total time required for mesh generation and finite element analysis, Δt_{total}, by

95% compared to the uniform mesh. Also, the error in principal stresses at the ROI compared to the uniform mesh remains within the acceptable range of under 20% (i.e. the accuracy of the input data for rock engineering problems).

4.5 Conclusions

For large scale problems in mining and civil engineering, it is often impossible to use a uniform mesh for the finite element analysis because the hardware requirements for solving the problem surpasses the available resources available on the computers used at consulting engineering firms. Therefore a model with less number of DOF is required. To obtain such a model, an optimized mesh with fewer number of nodes and elements should be created.

The framework was applied to several practical mining and civil engineering problems. The effects of each meshing criteria was studied on the solution time and the accuracy of the results. Using a combination of cell radius bound and facet distance bound calculated based on equation 3.16 proved to produce an optimized 3D volume mesh that provides results with accuracy comparable to the accuracy of the input data in the vicinity of the ROI. The optimized mesh significantly reduces the number of DOF and time to solution.

CHAPTER 5
Future Research and Conclusion

5.1 Future Research

This research provides the necessary framework to simplify defining the geometry of underground excavations and to generate optimized volume and surface meshes for finite element analysis and boundary element analysis respectively. One ROI was assumed in developing the principals that govern the framework. In some mining and civil engineering problems there might be multiple regions that are of importance while carrying out stress analysis. In those cases, it would be useful to define multiple ROI. Even though the framework was developed using only one ROI, the same principles apply to multiple ROI so it should be fairly simple to extend the framework in order to support multiple ROI.

Another area that can be improved is the ability to mesh input domains that have multiple regions (i.e. subdomains), for example rock mass with different material property. In its current form, the framework takes one polyhedral domain and creates the mesh. The Delaunay meshing algorithm that is used is generic enough to handle domains with multiple regions because it takes an *oracle* as input. This oracle must be able to answer certain queries. For example, it must be able to tell whether a point is inside the domain or not and if the point is inside the domain, it must return the index of the subdomain. A new oracle that can answer these questions can be constructed and added to the existing framework to support multiple domains.

To further optimize the mesh, more criteria can be added to provide more information to the framework and drive the meshing process more effectively. For example, the effects of the direction of normal vector of the surface facets can be taken into consideration. A facet that point away from the ROI has less influence on the results of the numerical analysis than one that is directly facing the ROI. Once the framework supports multiple regions, domains with different material property (e.g. modulus of elasticity and Poisson's ratio) can be used to study the effect of

material property on the mesh optimization process.

Right now, exact arithmetic is being used for some of the operations like intersection of geometrical entities to ensure stability of the geometric algorithms. Using exact arithmetic operations will provide the user with a robust tool that can be trusted. The disadvantage is that the tool will be slower compared to a similar tool that uses floating point arithmetic operations. Of course using floating point operations for geometric algorithms will result in less reliable applications that might fail on many occasions. Even though the time required by the framework for producing an optimized mesh of a typical problem is reasonable (e.g. producing a mesh of 6500 vertices, 3000 facets and 37000 cells takes only about 1.5 minutes), there is still room for improving the performance of the algorithm. This will be useful in certain use cases. Imagine a software application that enables the user to define the ROI and then control values of the meshing criteria with sliders made available through the GUI to observe the changes to the output mesh in real time. In this scenario, having the mesh produced in seconds (rather than minutes) will greatly improve the user experience.

5.2 Conclusions

There is no doubt that the computational power of the personal computers that are used in engineering firms is rapidly increasing over time but so does the complexity of the problems solved by engineers. Until a few years ago, most of the computations in the mining and geotechnical engineering were performed using 2D methods because producing a 3D model of the problem was very time consuming and the existing computational tools did not allow practical use of 3D models. Over time more sophisticated construction methods (such as NATM) were developed for tunneling that depend heavily on numerical stress analysis tools because the tunnel end-effects at the tunnel face are distinctively three-dimensional. Therefore the need for 3D stress analysis has increased recently.

To make better predictions and obtain more realistic results from numerical analysis of the problems in rock engineering, non-linear constitutive models were developed. Now personal computers have reached a point that they can provide

enough computational power to carry out 3D non-linear analysis of the large scale problems, if the computational resources are used optimally. To reach this goal, the size of the model to be numerically solved must be reduced as much as possible while the accuracy of the result is kept at a reasonable range in the vicinity of ROI where a solution to the problem is sought.

This thesis makes an organized attempt to address these issues. A framework was proposed that reduces *time to solution*. The framework contributes to reduction in time to solution at two levels: (a) it facilitates creation of geometry of the underground excavations and (b) it cuts the time required for numerical analysis by reducing number of surface and volume elements in the mesh while keeping the results accurate enough at the ROI.

To investigate the applicability and effectiveness of the framework, a software application was developed and the it was applied to a few mining and civil engineering problems. To reach a certain degree of accuracy, the framework was able to reduce the size of the problem by 14 folds compared to a uniform mesh. The time required for a linear finite element analysis was reduced by an incredible amount of 57 times, from 14.5 minutes to 15 seconds. The considerable improvement is because current personal computers must use *disk swapping* to solve problems that their requirements for the RAM surpasses the available amount of RAM. The framework optimizes the model so that it fits within the available RAM and improves the performance of the finite element solver dramatically.

In certain instances, the size of the problem is so large that even disk swapping can not help and the solver simply fails and refuses to solve the finite element problem. To overcome this issue, the framework can be utilized to reduce the size of these problems so that they fit the specifications of the available hardware and computational resources while the accuracy of the numerical analysis is kept within an acceptable range.

In view of these facts, the proposed framework can be of paramount importance and a great help in solving everyday problems solved by engineers in the field of mining and civil engineering.

APPENDIX A
Rock Mechanics and Standard Tunneling Practices

A.1 Introduction

This appendix provides a review of rock mechanics and standard tunneling practices. First the development of rock mechanics as a discipline is reviewed. Then tunneling terminology and different excavation techniques are reviewed.

A.2 Rock Mechanics

Rock mechanics is concerned with the application of principles of engineering mechanics in design and construction of underground excavations in rock mass. Rock mechanics itself is part of a broader subject named *geomechanics* which is concerned with the mechanical responses of all geomaterials including soils [3]. A widely accepted definition of rock mechanics is given by the US National Committee on Rock Mechanics in 1964 and subsequently modified in 1974 [3]:

> *"Rock mechanics is the theoretical and applied science of the mechanical behavior of rock and and rock masses; it is that branch of mechanics concerned with the response of rock and rock masses to the force fields of their physical environment."*

The earliest academic paper on rock mechanics was published by Coulomb. In 1773, Coulomb included results of tests on rocks from Bordeaux in a paper read before the French Academy in Paris [79]. French engineers started construction of the Panama Canal in 1884 and this task was taken over by the US Army Corps of Engineers in 1908. In the half century between 1910 and 1964, 60 slides were recorded in cuts along the canal. In discussing the Panama Canal slides in his Presidential Address to the first international conference on Soil Mechanics and Foundation Engineering in 1936, Karl Terzaghi [80, 81] said:

"The catastrophic descent of the slopes of the deepest cut of the Panama Canal issued a warning that we were overstepping the limits of our ability to predict the consequences of our actions ..."

In 1920 Josef Stini started teaching *Technical Geology* at the Vienna Technical University. He was probably the first to emphasize the importance of structural discontinuities on the engineering behavior of rock masses. Other notable scientists and engineers from a variety of disciplines did some interesting work on rock behavior during the early part of the 20th century. Von Karman, 1911 [82]; King, 1912 [83]; Griggs, 1936 [84]; Ide, 1936 [85]; and Terzaghi, 1945 [86] all worked on the failure of rock materials.

The principles of rock mechanics has long been known and used in practice by civil engineers. Rock mechanics is simply a formal expression of some of these principles and it is only during the past few decades that the theory and practice in this subject have come together in the discipline which we know today as rock mechanics. The formal development of rock mechanics as an engineering discipline in its own dates back to early 1960s.

Rockbursts are explosive failures of rock which occur when very high stress concentrations are induced around underground openings. A characteristic of almost all rockbursts is that they usually occur in deep level excavations that are highly stressed and consist of brittle rock.

Analysis of stresses induced around underground excavations can be carried out by means of the theory of elasticity. In the first edition of Jaeger and Cook's book *Fundamentals of Rock Mechanics* [87], elastic theory is the dominating approach in solving rock mechanics problems in deep excavations. Books by Coates [88] and by Obert and Duvall [89] reflect the same emphasis on elastic theory. Stini, one of the pioneers of rock mechanics, emphasized the importance of structural discontinuities in controlling the behavior of rock masses [90].

An important event in the development of the rock mechanics was the merging of elastic theory with the discontinuum approach. The gradual recognition that rock could act both as an elastic material and a discontinuous mass resulted in a much more mature approach to the subject than before.

A.3 Rock Mass Structure

Rock material is the term used to describe the intact rock with no discontinuity. The collection of the intact rock material, groundwater, as well as joints, faults and other natural planes of weakness that divide the rock into interlocking blocks of varying sizes and shapes is called *rock mass*. Rock masses are discontinuous and often have anisotropic and heterogeneous engineering properties. *Rock structure* is the distribution of this discontinuous structure throughout the rock mass [3]. Rock structure has a significant effect on the underground excavation operations. Since the rock structure controls the stability of excavation spans, support requirements, subsidence and fragmentation, it will influence the choice of excavation method and designing of tunneling layouts.

A.3.1 Major Geological Features

Faults: Faults are fractures in the rock that are the result of shear displacement. They are recognized by the relative displacement of rock on opposite sides of the *fault plane*. The sense of this displacement is used to classify faults. The two sides of a fault are called the *hanging wall* and *footwall*. By definition, the fault always dips away from the footwall. Faults can be categorized into three groups based on the sense of slip. A fault where the main sense of movement (or slip) on the fault plane is vertical is known as a *dip-slip* fault. Where the main sense of slip is horizontal the fault is known as a *strike-slip (or transform)* fault. *Oblique-slip* faults have significant components of both strike and dip-slip.

Dip-slip faults include both *normal* and *reverse*. A *normal fault* occurs when the crust is in tension. The hanging wall moves downwards relative to the footwall. A *reverse fault* is the opposite of a normal fault - the hanging wall moves up relative to the footwall. Reverse faults are indicative of compressional forces and shortening of the local crust. The dip of a reverse fault is relatively steep, greater than 45 degrees.

Strike-slip fault surface is usually near vertical and the footwall moves either left or right or laterally with very small vertical motion. Strike-slip faults with

left-lateral motion are also known as *sinistral* faults. Those with right-lateral motion are also known as *dextral* faults.

Fault thickness might vary from meters to millimeters in different cases and might contain weak material such as fault gouge (clay), fault breccia (re-cemented), rock flour or angular fragments. The wall rock may be coated with low friction minerals such as graphite and chlorite. These factors make fault zones, areas of low shear strength that slip may readily occur in them [3].

Dykes: A dyke is an *intrusion* of generally fine-grained igneous rock into a cross-cutting fissure. In geology, intrusion is usually a body of igneous rock that has crystallized from a molten magma below the surface of the Earth. A dyke cuts across other pre-existing layers or bodies of rock, meaning a dyke is always younger than the rocks that contain it. Dykes are usually high angle to near vertical in orientation, but subsequent tectonic deformation may rotate the including sequence. The thickness is usually much smaller than the other two dimensions. Thickness can vary from sub-centimeter scale to many meters in thickness and the lateral dimensions can extend over many kilometers.

Joints: Joints are fractures in rock along which no appreciable movement has occurred. A group of parallel joints is called a *joint set*. The intersection of joint sets forms a *joint system*. Joints may be open, filled or healed. Commonly, streams develop along zones of weakness caused by joints in rocks, and thus the regional pattern of joint orientation often exerts a strong control on the development of drainage patterns.

Discontinuity: It is common in rock mechanics to use the term *discontinuity* as a collective term for all fractures or features in a rock mass such as joints, faults, shears, weak bedding planes and contacts that have zero or relatively low tensile strength.

A.3.2 Geomechanical Properties of Discontinuities

In this section the geomechanical properties of discontinuities that influence the engineering behavior of rock mass are discussed briefly. For a comprehensive

review of these properties, the reader may refer to *Suggested methods for the qualitative description of discontinuities in rock masses* prepared by the Commission on Standardization of Laboratory and Field Tests [3].

Spacing: The perpendicular distance between adjacent discontinuities is called *spacing*. It is usually expressed in terms of the mean value of the spacing of a *joint set*. The size of the blocks that make up the rock mass is determined by the discontinuity spacing. Discontinuity spacing is a factor that is used in classifying rock masses for engineering purposes. To quantify discontinuity spacing, *Rock Quality Designation* (RQD) is defined as:

$$RQD = \frac{100\Sigma x_i}{L} \tag{A.1}$$

where x_i are the length of individual pieces of core that have a length of 0.1 meter or greater in a drill run and L is the total length of the drill run.

Persistence: Persistence is the term used to describe the size and extent of discontinuities within a plane. It can be quantified by observing the trace lengths of discontinuities on exposed surfaces. It is one of the most important rock mass parameters yet one of the most difficult to determine.

Roughness: Roughness is a measure of surface unevenness and waviness of the discontinuity relative to its mean plane. The wall roughness of a discontinuity has an important effect on its shear strength, especially in the case of undisplaced and interlocked features (e.g. unfilled joints). The importance of roughness declines with increasing aperture, filling thickness or previous shear displacement.

Aperture: *Aperture* is the perpendicular distance between the adjacent rock walls of an open discontinuity. The intervening space in an open discontinuity is filled with weather or water therefore it is distinguished from the *width* of a filled discontinuity. Large aperture may result from outwash of filling materials (e.g. clay) or other causes. In most subsurface rock masses, however, aperture will be small and in order of millimeters but varies over the extent of

the discontinuity. Aperture and its spatial variation will influence the shear strength and permeability (i.e. hydraulic conductivity) of the discontinuity and the rock mass.

A.4 Tunneling

A.4.1 Geometry of Tunnels and Related Terminology

Considering the cross and longitudinal sections of tunnels in Figure A.1 and A.2, the various parts are referred to by the the names shown in the diagrams. The word *chainage* is used to identify a point along the axis of a tunnel defined by its distance from a fixed reference point.

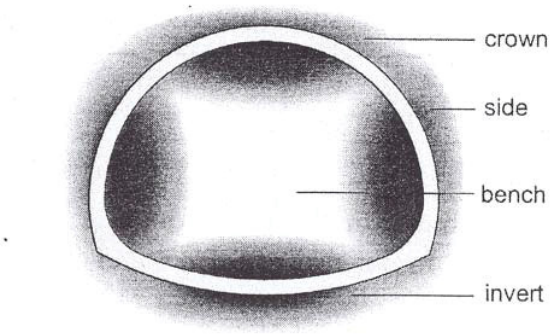

Figure A.1: Parts of the tunnel cross-section [1].

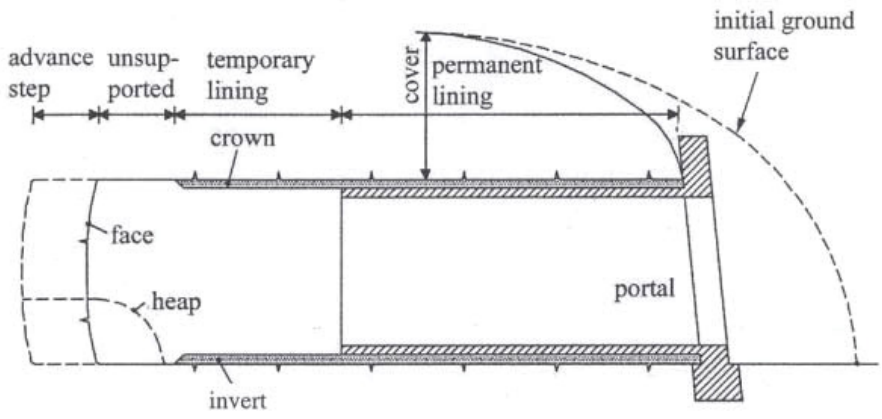

Figure A.2: Longitudinal section of heading [1].

A.4.2 Cross-Sections

The shape of a tunnel cross-section is also called profile. Various profiles are possible, e.g. rectangular ones. The most popular ones, however, are circular and mouth profiles (Figure A.1). The choice of a profile follows the performance requirements of the tunnel. Moreover it should minimize bending moments in the lining as well as costs for excavation and lining. Further aspects for the choice of the profile are: ventilation, maintenance, risk management and avoidance of claustrophobia[2] of users [1].

The size of a tunnel is often given by its cross-sectional area. Typical values for tunnel cross-section areas are given in table A.1.

Type of Tunnel	Area (m^2)
Sewer	10
Hydropower tunnels	10 - 30
Motorway (one lane)	75
Rail (one track)	50
Metro (one track)	35
High speed rail (one track)	50
High speed rail (two tracks)	80 - 100

Table A.1: Typical values for tunnel cross-section areas [1]

A.5 Heading

The heading of a tunnel comprises the following actions: excavation, support of the cavity and removal of the excavated earth (mucking). Two different heading methods are distinguishable: conventional (also called incremental or cyclic) heading and continuous heading. A rigorous classification of heading methods is difficult since these methods are often combined.

A.5.1 Core Heading

This is also known as German heading method (although it was first used in France). It consists of excavating and supporting first the side and top parts of the cross-section and subsequently the central part (core). The ring closure at the invert

[2] An abnormal fear of being in narrow or enclosed spaces.

comes at the end. The first gallery also serves for exploration. The crown arch is founded on the side galleries thereby keeping the related settlements small.

A.5.2 Old Austrian Tunneling Method

This method is schematically represented in Figure A.3. Its characteristic feature is the crown slot. The simultaneous work in several excavation faces allows a fast advance.

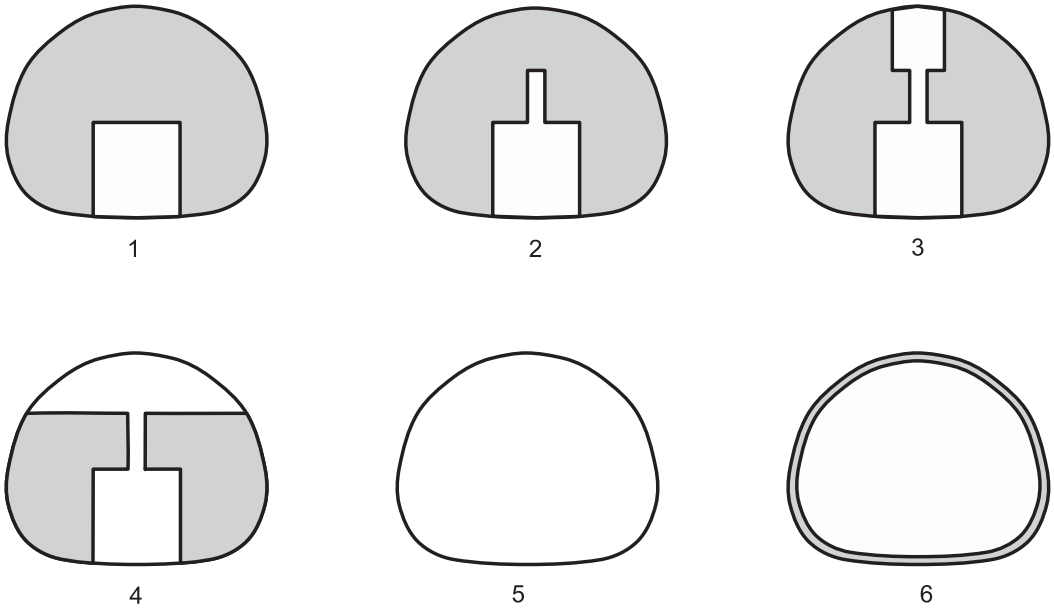

Figure A.3: Excavation Sequence of the Old Austrian Method [1]

A.5.3 Top Heading

The crown is excavated before the bench (Figure A.4 and A.5). The temporary support of the crown with shotcrete can be conceived as a sort of arch bridge. This explains why the abutments are prone to settlements, which include settlements of the ground surface. Countermeasures are to enlarge the abutments (so-called elephant feet) or the construction of a temporary invert. The latter must be constructed soon after the heading of the crown. A soon construction of the crown section or better, the soon excavation and support of the bench and invert helps avoiding large settlements of the abutments of the crown arch [1]. This means that

the length $a = a_1 + a_2$ (Figure A.5) should be kept as small as possible. On the other hand, a_1 should be sufficiently large to enable efficient excavation and support works in the crown.

If the crown and the bench are excavated simultaneously, then the ramp must be continuously moved forward (i.e. every now and then). Alternatively, the ramp is not placed at the center, as shown in Figure A.5, but on the side of the bench. Then, the other side of the bench can be excavated over a longer distance. If the excavation of a ramp may cause instability, then the ramp must be heaped up after excavation and support of the bench.

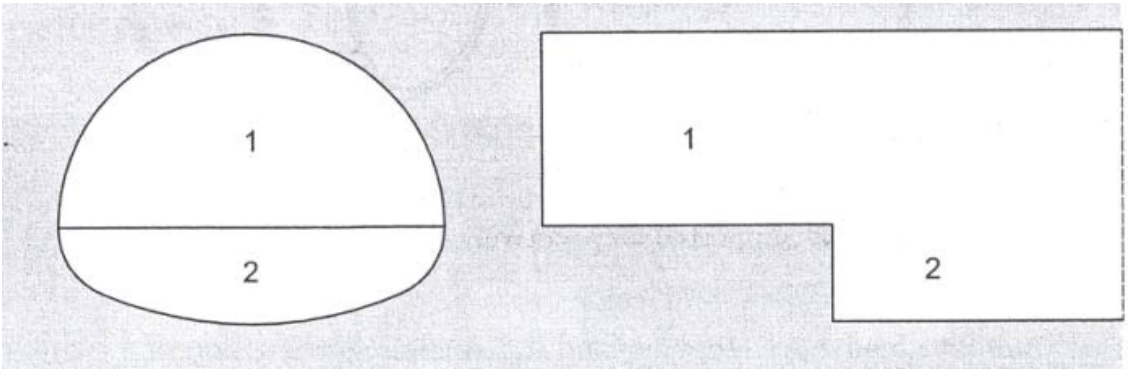

Figure A.4: Top heading, cross and longitudinal sections. 1: calotte, 2: bench [1].

A.5.4 Sidewall Drift

The side galleries are excavated and supported first. They serve as abutment for the support of the crown, which is subsequently excavated (Figure A.6). This type of heading is approximately 50% more expensive and slower than top heading. Therefore it is preferred in soil/rock masses with low strength. Note that a change from top heading to sidewall drift is difficult to accomplish.

A.6 New Austrian Tunneling Method (NATM)

The New Austrian Tunneling Method (NATM), emerged in the years 1957 to 1965 [91]. The NATM was developed by Austrian tunneling specialists von Rabcewicz, Pacher and Müller-Salzburg. As defined by the Austrian Society of

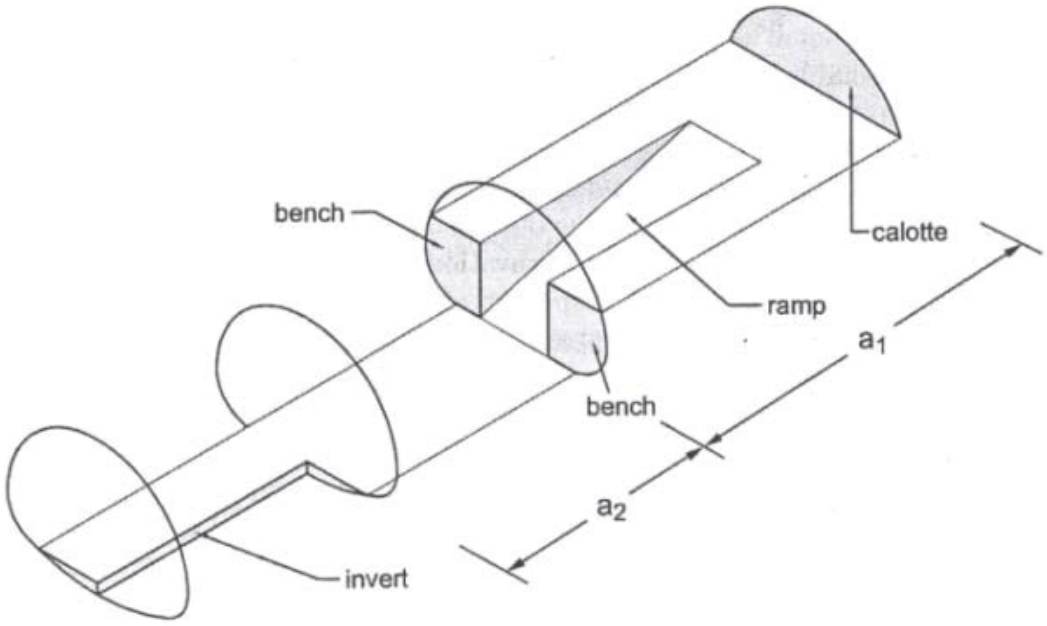

Figure A.5: Schematic representation of top heading [1].

Engineers and Architects, the NATM constitutes a method where the surrounding ground (rock and/or soil) formations of a tunnel form a bearing ring that acts as a support structure. Thus the supporting formations will themselves be part of this supporting structure.

In world-wide practice, however, when shotcrete is proposed for initial ground support of an open-face tunnel, it is often referred to as NATM. The term NATM with reference to soft ground, however, can be misleading. As noted in a very thoughtful article by Emit Brown [5], NATM can refer to both a design philosophy and a construction method. Key features of the NATM design philosophy are:

- The strength of the rock mass/soil around a tunnel is deliberately mobilized to the maximum extent possible.
- Mobilization of rock mass/soil strength is achieved by allowing controlled deformation of the ground.
- Initial primary support is installed having load-deformation characteristics appropriate to the ground conditions, and installation is timed with respect to ground deformations.

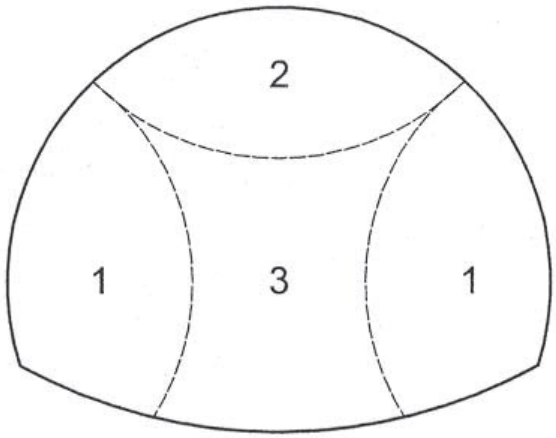

Figure A.6: Sidewall drift [1]

- Instrumentation is installed to monitor deformations in the initial support system, as well as to form the basis of varying the initial support design and the sequence of excavation.

Key features of NATM construction methods are:

- The tunnel is sequentially excavated and supported, and the excavation sequences can be varied.
- The initial ground support is provided by shotcrete in combination with fiber or welded-wire fabric reinforcement, steel arches (usually lattice girders), and sometimes ground reinforcement (e.g., soil nails, spiles[3]).
- The permanent support is usually (but not always) a cast-in-place concrete lining.

It should be noted that many of the construction methods described above were in widespread use in the US and elsewhere in soft-ground applications before NATM was described in the literature. In current practice, for soft-ground tunnels which are referred to as NATM tunnels, initial ground support in the form of shotcrete

[3]A column of wood or steel or concrete that is driven into the ground to provide support for a structure.

(usually with lattice girders and some form of ground reinforcement) is installed as excavation proceeds, followed by installation of a final lining at a later date [6].

Soft ground can be described as any type of geomaterial requiring support as soon as possible after excavation in order to maintain stability of the excavation. For tunnels in dense urban areas, it is very important to control the settlements in order to avoid damage to overlying structures. In order to limit settlement and ensure a safe work environment, soft ground tunnels must employ the following measures:

- Dimensions and duration of excavation stages must be adequately short.
- Formation of the *full ring* of initial ground support must be completed immediately after excavation.

A.6.1 NATM for Soft Ground

In soft-ground tunneling, safety dictates that the ground support be placed immediately after excavation. As long as the ground is properly supported, NATM construction methods are appropriate for soft-ground conditions. However, there are cases where soft-ground conditions do not favor an open face with a short length of uncompleted lining immediately next to it, such as in flowing ground or ground with short stand-up time (i.e., failure to develop a ground arch). Unless such unstable conditions can be modified by dewatering, spiling, grouting, or other methods of ground improvement, then NATM may be inappropriate. In these cases, close-face shield tunneling methods may be more appropriate for safe tunnel construction.

A.6.2 NATM and Numerical Modeling Frameworks

Numerical modeling frameworks are useful tools for design of sequentially excavated, shotcrete-lined tunnels. They are used to evaluate stresses and strains in the ground and tunnel support (i.e. lining). These frameworks usually use finite or boundary element methods for numerical solution of the problem and the results are strongly dependent on the geotechnical input parameters and the constitutive models used for analysis. The engineer should clearly understand the limitations of numerical modeling. Most importantly, field observations and measurements should be used to confirm assumptions and calibrate future models.

In particular, numerical modeling is useful where interacting tunnels, unusual geometries, discontinuities or adjacent structures are present [6]. However, if a shotcrete-lined tunnel has a section that is nearly circular or oval with no irregularities, and if there are no adjacent surface or subsurface structure interacting with the tunnel, then approximate or closed-form solutions for interaction between ground and lining can be used if they exist. The closed-form solutions can also be used for a prudent check on the results from numerical modeling.

A.6.3 Instrumentation and Monitoring

As noted above, instrumentation and monitoring play a key role in verifying design assumptions and calibrating numerical models. More importantly, however, monitoring serves to alert the designer and the constructor if the lining is not performing as intended, or is in danger of collapse. In this respect instrumentation of NATM construction is no different from other types of geotechnical construction. Therefore the following geotechnical instrumentation rules equally apply to NATM [6]:

1. Predict mechanisms that control behavior, and define the geotechnical questions to be answered.
2. Define the purpose of instrumentation, and select parameters to be monitored.
3. Predict magnitudes of change, and determine threshold limits and remedial actions.
4. Assign tasks and responsibilities.
5. Select instruments and locations.
6. Devise methods to ensure correctness.
7. Plan data collection, processing, presentation, interpretation, and reporting.

If these steps are correctly followed in a systematic instrumentation and monitoring approach then there is a chance of getting good data that can be relied upon, in order to make decisions during construction.

A.6.4 Tunnel Collapses

Unfortunately there have been several collapses or other stability failures of NATM projects around the world including, most recently in Turkey and the US. Perhaps the most famous is the Heathrow Airport collapse in October 1994, which triggered a thorough review of the NATM by the British Health and Safety Executive (HSE). In a 1996 report, the HSE examined 39 NATM failures, categorizing the location (in the tunnel) of the failure. In most cases, the failure was a result of heading collapse.

Broadly speaking the causes of these failures were varied, from unanticipated geologic conditions, to design errors, to construction quality problems, to poor management. Nevertheless NATM failures, or for that matter any tunnel failure, have one thing in common: most are caused by human error. Its not the fault of the method, but misapplication of the method.

A.7 Conclusions

The development of rock mechanics as a discipline was reviewed. In the early stages of rock mechanics development, theory of elasticity was being used for analysis of stress induced around underground excavations but the work of pioneers like Josef Stini resulted in gradual recognition that rock could act both as an elastic material and a discontinuous mass which resulted in the merging of elastic theory with the discontinuum approach.

Major geological features such as bedding planes, folds, faults, dykes, joints and other kinds of discontinuities were defined and data collection using mapping exposure, drilling and core logging were visited briefly. After an overview of tunneling terminology, heading methods such as core heading, top heading and sidewall drift were introduced. Finally New Austrian Tunneling Method was introduced and the need for a modeling framework was emphasized.

APPENDIX B
Numerical Methods in Geomechanics and Tunneling

B.1 Introduction

The desire to understand the physical world and to describe it using mathematical concepts has long been a goal of scientists and engineers. After a physical phenomenon is formulated mathematically, an in-depth analysis of it is made possible through studying the governing equations.

When designing a tunnel or excavation, there are design objectives that must be met. These objectives may be identified as follows:

- Local stability of the underground structure and its support system as well as overall stability should be ensured.
- The induced displacements must be tolerable, not only for the structure being designed but also for any neighboring structures and services.

Analysis of the mathematical model of a geotechnical problem provides an assessment of these important aspects of the design. Traditionally, geotechnical design has been carried out using simple analysis or empirical approaches but these methods have their limitations and are not sufficient for problems with complex domains. Today, the availability of inexpensive sophisticated computer hardware has made it possible for engineers to deploy computationally intensive numerical methods to solve problems with complex domains.

In geomechanics, *constitutive models* are used to formulate the behavior of geomaterial (soil or rock mass). There is a large number of publications available on constitutive models. To name just a few of these models we may refer to *elasticity models* (linear and piecewise linear), *hyper-elasticity and hypo-elasticity models*, *plasticity models* and *hypo-plasticity models*.

Having the constitutive model of the geomaterial, it is possible to formulate the problem domain by partial differential equations (PDEs). When these equations have complicated boundary conditions or are posed on irregularly shaped objects

or domains, they usually do not admit closed-form solutions. A numerical approximation of the solution is thus necessary. These methods numerically approximate the solution of a linear or non-linear PDE by replacing the continuous system with a finite number of coupled linear or non-linear algebraic equations. This process of *discretization* associates a variable with each of a finite number of points, called *nodes*, in the problem domain. A brief review of the numerical methods used in geomechanics is given here.

B.2 Numerical Methods in Geomechanics

Some of the numerical analysis methods used in geomechanics to solve boundary value problems are the *finite element method* (FEM), the *boundary element method* (BEM), the *finite difference method* (FDM) and the *discrete element method* (DEM). Detailed descriptions of each of these numerical methods may be found in a large number of textbooks (e.g., Zienkiewicz, 1967 [7]; Desai and Abel, 1972 [8]; Britto and Gunn, 1987 [9]; Smith and Griffiths, 1988 [10]; Beer and Watson, 1992 [11]; Potts and Zdravkovic, 1999 [12, 13]). However, it is probably worth noting that to use the FEM, BEM, FDM and DEM methods, one must consider the entire problem domain, break it up into a finite number of discretized sub-regions or elements.

The governing equations of the problem are applied separately and approximately within each of these elements, translating the governing differential equations into matrix equations for each element. Compatibility, equilibrium and the boundary conditions are enforced at the interfaces between elements and at the boundaries of the problem. On the other hand, in the BEM only the boundary of the problem domain under consideration is discretized, thus providing a computational efficiency by reducing the dimensions of the problem by one. The BEM is particularly suited to linear problems. For this reason, and because it is well suited to modeling infinite or semi-infinite domains, the BEM is sometimes combined with the finite element technique.

B.2.1 Finite Element Method

The finite element method is still the most widely used and probably the most versatile method for analyzing boundary value problems in geomechanics. The main advantages and disadvantages for geotechnical analysis may be summarized as follows.

Advantages

- Non-linear material behavior can be considered for the entire domain analyzed.
- Modeling of excavation sequences including the installation of reinforcement and structural support systems is possible.
- Structural features in the soil or rock mass, such as closely spaced parallel sets of joints can be efficiently modeled.
- Time-dependent material behavior may be introduced.
- The system of equations is symmetric (except for non-associated flow rules in elasto-plastic problems using tangent stiffness methods).
- The conventional displacement formulation may be used for most load-path analysis.
- Special formulations are now available for other types of geotechnical problem, e.g., seepage analysis.
- The method has been extensively applied to solve practical problems and thus a lot of experience is already available.

Disadvantages

- The entire volume of the domain analyzed has to be discretized, i.e., large pre- and post-processing efforts are required.
- Due to the large system of equations, run times and disk storage requirements may be excessive (depending on the general structure and the implemented algorithms of the finite element code).

- Sophisticated algorithms are needed for strain hardening and softening constitutive models.
- The method is generally not suitable for highly jointed rocks or highly fissured soils when these defects are randomly distributed and dominate the mechanical behavior.

B.2.2 Boundary Element Method

Significant advances have been made in the development of the boundary element method and as a consequence this technique provides an alternative to the finite element method under certain circumstances, particularly for some problems in rock engineering [11]. The main advantages and disadvantages may be summarized as follows.

Advantages

- Pre- and post-processing efforts are reduced by an order of magnitude (as a result of surface discretization rather than volume discretization).
- The surface discretization leads to smaller system of equations and less disk storage requirements, thus computation time is generally decreased.
- Distinct structural features such as faults and interfaces located in arbitrary positions can be modeled very efficiently, and the non-linear behavior of the fault can be readily included in the analysis [92].

Disadvantages

- In general, non-symmetric and often fully-populated system of equations is obtained.
- Detailed modeling of excavation sequences and support measures is still a problem that is being studied and is not completely solved [93].
- The standard formulation is not suitable for highly jointed rocks when the joints are randomly distributed.

- The method has only been used for solving a limited class of problems, e.g., tunneling problems, and thus less experience is available than with finite element models.

B.2.3 Coupled Finite Element - Boundary Element Method

It follows from the arguments given above that it should be possible to minimize the respective disadvantages of both methods by combining them. This is in fact true and very efficient numerical models can be obtained by discretizing the soil or rock around the region of particular interest, e.g., representing the region around a tunnel by finite elements and the far field by boundary elements [11, 94]. Two disadvantages however remain, namely the cumbersome modeling of major discontinuities intercepting the ROI in an arbitrary direction, e.g., a tunnel axis, and the non-symmetric system of equations that is generated by the combined model. The latter problem may be resolved by applying the principle of minimum potential energy for establishing the stiffness matrix of the boundary element region [11]. If this is done, then after assembling with the finite element stiffness matrix, the resulting system of equations remains symmetric.

B.2.4 Finite Difference Method

The finite difference method is not as popular as finite element or boundary element methods in geotechnical engineering but is used in analyzing flow problems including those involving contaminant transport [95].

B.2.5 Discrete Element Method

The methods described so far are based on continuum mechanics principles and are therefore restricted to problems where the mechanical behavior is not governed to a large extent by the effects of joints and cracks. If this is the case, discrete element methods are much better suited for numerical solution. These methods may be characterized as follows:

- Finite deformations and rotations of discrete blocks (deformable or rigid) are calculated.

- Blocks that are originally in contact may separate during the analysis.
- New contacts which develop between blocks due to displacements and rotations are detected automatically.

Due to the different nature of a discontinuum analysis, as compared to continuum techniques, a direct comparison seems to be not appropriate. The major strength of the distinct element method is certainly the fact that a large number of irregular joints can be taken into account in a physically rational way. The drawbacks associated with the technique are that establishing the model, taking into account all relevant construction stages, is still very time consuming, at least for 3D analysis. In addition, a lot of experience is necessary in determining the most appropriate values of input parameters such as stiffness of joints. These values are not always available from experiments and specification of inappropriate values for these parameters may lead to computational problems. In addition, run-times for 3D analysis are usually quite high.

B.3 Processing Phases in Numerical Analysis

Numerical analysis of a problem consists of three distinct phases: *pre-processing*, *numerical solution* and *post-processing*. Achieving a high level of automation among these three phases will greatly enhance the efficiency of the numerical analysis. A brief description of each phase follows.

B.3.1 Pre-Processing

The first step in modeling a physical phenomenon is to idealize it. In the *idealization* process a simplified version of the real problem is created. Only the most essential aspects of the problem are considered in the simplified version to create a mathematical formulation of the problem. This is one of the most important phases in problem solving. This step needs human intervention and engineering judgment. It can not be easily automated.

If the problem is complicated and there exist no closed form solution, then a numerical method is chosen to solve the idealized version of the problem. FEM,

BEM or a hybrid of them has successfully been used in geomechanics. To use any of the aforementioned methods one needs to establish a discretized version of the problem. Therefore the next step after idealization is *discretization* of the problem domain. Automation of the discretization process (i.e mesh generation) is the central focus of this research.

In FEM, the problem domain (or in the BEM, the boundary of the problem domain) must be partitioned into small pieces of simple shape. These pieces are called *elements*, and are usually triangles or quadrilaterals (in two dimensions), or tetrahedra or hexahedral bricks (in three dimensions). FEM and BEM employ a *node* at every element vertex (and sometimes at other locations); each node is typically shared among several elements.

A *mesh* is the collection of these nodes and elements and conforms to the geometry and boundaries of the physical problem one wishes to model. This mesh should be composed of elements whose sizes possibly vary throughout the mesh and are well shaped. Reconciling these constraints is not easy. Historically, the automation of mesh generation has proven to be more challenging than the entire remainder of the simulation process [20]. For a review of model discretization techniques see chapter 2.

B.3.2 Numerical Solution

Once model discretization is complete, one can solve the problem with a numerical method that best suits that specific problem. FEM, BEM or a hybrid of them are the most commonly used methods to solve problems in geomechanics. A brief review of these methods is given in the sections B.2.1 through B.2.3.

B.3.3 Post-Processing

The result of a numerical solution is a dataset that describes the behavior of the model at a finite number of points, called *nodes*, in the problem domain. The size of this dataset depends on the size of the problem and the number of nodes used in the discretized model. In the real world problems with large domains, this is usually an enormous amount of data that is prohibitively large for human observation alone.

Data mining and data visualization techniques respond to this problem. Data visualization is the graphical presentation of these datasets, with the goal of providing the viewer with a qualitative understanding of the behavior of the system.

B.4 Conclusions

The pros and cons of different numerical methods used in geomechanics were discussed. Finite and boundary element methods or a hybrid of them are the most popular methods in geomechanics. All these methods need the problem domain be subdivided into elements of regular shapes which is one of the processing phases in numerical analysis of a problem. Finally, the three distinct phases in numerical methods were identified as pre-processing, numerical solution and post-processing and each were discussed individually.

BIBLIOGRAPHY

[1] Kolymbas D. *Tunneling and Tunnel Mechanics: A Rational Approach to Tunnelling*. Springer: Berlin; New York, 2005. Dimitrios Kolymbas. Tunnelling and tunnel mechanics ill.; 24 cm. Includes index.

[2] Karakus M, Fowell RJ. An insight into the new austrian tunnelling method (NATM). *ROCKMEC'2004-VIIth Regional Rock Mechanics Symposium, 2004, Sivas, Turkey*, 2004.

[3] Brady BHG, Brown ET. *Rock Mechanics for Underground Mining*. 2nd edn., Kluwer Academic: Dordrecht; Boston, 1999. B.H.G. Brady and E.T. Brown. – Rock mechanics ill. Reprinted with corrections. This edition originally published: London: Chapman & Hall, 1993.

[4] Hoek E. *Practical Rock Engineering*. 2007. URL `http://www.rocscience.com/hoek/corner/Practical_Rock_Engineering.pdf`.

[5] Brown ET. Putting the NATM into perspective. *Tunnels & Tunnelling* November 1981; **13**(10):13–17.

[6] Romero V. NATM in soft-ground: A contradiction of terms? *World Tunneling* 2002; **15**:338–343.

[7] Zienkiewicz OC, Cheung YK. *The Finite Element Method in Structural and Continuum Mechanics*. McGraw-Hill Publ., London, 1967.

[8] Desai CS, Abel JF. *Introduction to the Finite Element Method: A Numerical Method for Engineering Analysis*. Van Nostrand Reinhold, New York, 1972.

[9] Britto AM, Gunn MJ. *Critical State Soil Mechanics via Finite Elements*. Ellis Horwood, Chichester, 1987.

[10] Smith IM, Griffith DV. *Programming the Finite Element Method*. John Wiley & Sons, Chichester, 2004.

[11] Beer G, Watson JO. *Introduction to Finite and Boundary Element Methods for Engineers*. J. Wiley & Sons, New York, 1992.

[12] Potts D, Zdravkovic L. *Finite Element Analysis in Geotechnical Engineering: Volume I- Theory*. Telford Publishing, London, 1999.

[13] Potts D, Zdravkovic L. *Finite Element Analysis in Geotechnical Engineering: Volume II- Application*. Telford Publishing, London, 1999.

[14] Watson J, Cowling R. Application of three-dimensional boundary element method to modelling of large mining excavations at depth. *Proceedings of the Fifth International Conference on Numerical Methods in Geomechanics*, A.A. Balkema: Nagoya, 1985.

[15] Zsaki AM. Innovative techniques in large-scale stress analysis of underground excavations. PhD Thesis, University of Toronto 2003.

[16] Zsaki AM, Curran JH. A continuum mechanics-based framework for boundary and finite element mesh optimization in two dimensions for application in excavation analysis. *International Journal for Numerical and Analytical Methods in Geomechanics* 2005; **29**:369–393.

[17] Golser H, Schubert W. *Numerical Simulation in Tunneling*, chap. Application of Numerical Simulation at the Tunnel Site. Springer, Wien, NewYork, 2003; 427–473.

[18] Knupp PM. Algebraic mesh quality metrics. *SIAM Journal on Scientific Computing* 2001; **23**(1):193–218.

[19] Knupp PM. Algebraic mesh quality metrics for unstructured initial meshes. *Finite Elements in Analysis and Design* 2003; **39**:217–241.

[20] Shewchuk JR. Delaunay refinement mesh generation. PhD Thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania May 1997.

[21] Löhner R. Progress in grid generation via the advancing front technique. *Engineering with Computers* 1996; **12**:186–210.

[22] Löhner R, Parikh P, Gumbert C. *Numerical Grid Generation in Computational Fluid Mechanics '88*, chap. Interactive Generation of Unstructured Grid for Three Dimensional Problems. Pineridge Press, 1988; 687–697.

[23] Lo SH. Volume discretization into tetrahedra – I. verification and orientation of boundary surfaces. *Computers and Structures, Pergamon* 1991; **35**(5):493–500.

[24] Lo SH. Volume discretization into tetrahedra – II. 3D triangulation by advancing front approach. *Computers and Structures, Pergamon* 1991; **39**(5):501–511.

[25] Löhner R. Generation of three-dimensional unstructured grids by the advancing front method. *Proceedings of the 26th AIAA Aerospace Sciences Meeting (Reno, Nevada)*, 1988.

[26] Cavendish JC, Field DA, Frey WH. An approach to automatic three-dimensional finite element mesh generation. *International Journal for Numerical Methods in Engineering* 1985; **21**(2):329–347.

[27] Frey WH. Selective refinement: A new strategy for automatic node placement in graded triangular meshes. *International Journal for Numerical Methods in Engineering* 1987; **24**(11):2183–2200.

[28] Weatherill NP. Delaunay triangulation in computational fluid dynamics. *Computers and Mathematics with Applications* 1992; **24**(5/6):129–150.

[29] Mingwu L, Benzley SE, Sjaardema G, Tautges T. A multiple source and target sweeping method for generating all-hexahedral finite element meshes. *Proceedings of 5th International Meshing Roundtable, Sandia National Laboratories*, 1996; 217–228.

[30] Staten ML, Canann SA, Owen SJ. BMSWEEP: Locating interior nodes during sweeping. *Proceedings of 7th International Meshing Roundtable, Sandia National Laboratories*, 1998; 7–18.

[31] Blacker T. The cooper tool. *Proceedings of 5th International Meshing Roundtable, Sandia National Laboratories*, 1996; 13–30.

[32] Farouki RT. Optimal parameterizations. *Computer Aided Geometric Design* February 1997; **14**(2):153–168.

[33] George P, Borouchaki H. *Delaunay Triangulation and Meshing: Application to Finite Elements*. Hermes, 1998.

[34] Chen H, Bishop J. Delaunay triangulation for curved surfaces. *Proceedings of 6th International Meshing Roundtable, Sandia National Laboratories*, 1997; 115–127.

[35] Cuilliere JC. An adaptive method for the automatic triangulation of 3D parametric surfaces. *Computer-Aided Design, Elsevier* 1998; **30**(2):139–149.

[36] Tristano JR, Owen SJ, Canann SA. Advancing front surface mesh generation in parametric space using a riemanniansurface definition. *Proceedings of 7th International Meshing Roundtable, Sandia National Labs*, Sandia National Labs, 1998; 429–445.

[37] Lau TS, Lo SH. Finite element mesh generation over analytical surfaces. *Computers and Structures, Elsevier* 1996; **59**(2):301–309.

[38] Lau TS, Lo SH, Lee CK. Generation of quadrilateral mesh over analytical curved surfaces. *Finite Elements in Analysis and Design, Elsevier* 1997; **27**:251–272.

[39] Dey TK, Li G, Ray T. Polygonal surface remeshing with delaunay refinement. *Proceedings of 14th International Meshing Roundtable, Sandia National Laboratories*, 2005; 343–361.

[40] Salem AZI, Canann SA, Saigal S. Robust distortion metric for quadratic triangular 2d finite elements. *ASME Trends in Unstructured Mesh Generation - AMD* July 1997; **220**:73–80.

[41] Field DA. Laplacian smoothing and delaunay triangulations. *Commuications in Applied Numerical Methods* 1988; **4**:709–712.

[42] Canann SA, Tristano JR, Staten ML. An approach to combined laplacian and optimization-based smoothing for triangular, quadrilateral, and quad-dominant meshes. *Proceedings of 7th International Meshing Roundtable, Sandia National Laboratories*, 1998; 479–494.

[43] A FL, Mark J, Plassmann P. An efficient parallel algorithm for mesh smoothing. *Proceedings of 4th International Meshing Roundtable, Sandia National Laboratories*, 1995; 47–58.

[44] Freitag LA. On combining laplacian and optimization-based mesh smoothing techniques. *ASME Trends in Unstructured Mesh Generation - AMD* 1997; **220**:37–43.

[45] Freitag LA, Ollivier-Gooch C. Tetrahedral mesh improvement using swapping and smoothing. *International Journal for Numerical Methods in Engineering, Wiley* 1997; **40**:3979–4002.

[46] Joe B. Construction of three-dimensional improved-quality triangulations using local transformations. *SIAM Journal on Scientific Computing* 1995; **16**:1292–1307.

[47] Canann SA, Muthukrishnan SN, Phillips RK. Topological refinement procedures for triangular finite element meshes. *Engineering with Computers, Springer-Verlag* December 1996; **12**:243–255.

[48] Staten ML, Canann SA. Post refinement element shape improvement for quadrilateral meshes. *ASME Trends in Unstructured Mesh Generation - AMD* July 1997; **220**:9–16.

[49] Kinney P. Cleanup: Improving quadrilateral finite element meshes. *Proceedings of 6th International Meshing Roundtable, Sandia National Laboratories*, 1997; 437–447.

[50] Mitchell SA, Tautges TJ. Pillowing doublets: Refining a mesh to ensure that faces share at most one edge. *Proceedings of 4th International Meshing Roundtable, Sandia National Laboratories*, 1995; 231–240.

[51] Rivara MC. New longest-edge algorithms for the refinement and/or improvement of unstructured triangulations. *International Journal for Numerical Methods in Engineering, Wiley* 1997; **40**:3313–3324.

[52] Staten ML, Jones NL. Local refinement of three-dimensional finite element meshes. *Engineering with Computers, Springer-Verlag* 1997; **13**:165–174.

[53] Zsaki AM, Curran JH. A continuum mechanics-based framework for optimizing boundary and finite element meshes associated withunderground excavations - framework. *International Journal for Numerical and Analytical Methods in Geomechanics* November 2005; **29**:1271–1298.

[54] Saikat D, Shephard MS, Georges MK. Elimination of the adverse effects of small model features by the local modification of automatically generated meshes. *Technical Report SCOREC Report #30-1995*, Rensselaer Polytechnic Institute, Troy, NY 1995.

[55] Tautges TJ. Automatic detail reduction for mesh generation applications. *10th International Meshing Roundtable*, Sandia National Laboratories, 2001; 407–418.

[56] El-sana J, Varshney A. Topology simplification for polygonal virtual environments. *IEEE Transactions on Visualization and Computer Graphics* 1998; **4**:133–144.

[57] ANSYS Workbench Platform. URL `http://ansys.com/Products/Workflow+Technology/ANSYS+Workbench+Platform`, [Online; accessed 15-April-2010].

[58] Morris R. *CUBIT User Documentation*. Sandia National Laboratories April 2010. URL `http://cubit.sandia.gov/help-version12.1/cubithelp.htm`, version 12.1.

[59] Si H. *TetGen: A Quality Tetrahedral Mesh Generator and a 3D Delaunay Triangulator*. Weierstrass Institute for Applied Analysis and Stochastics, Mohrenstr. 39, 10117 Berlin, Germany January 2006. URL `http://tetgen.berlios.de/files/tetgen-manual.pdf`, version 1.4.

[60] Geuzaine C, Remacle JF. *Gmsh Reference Manual* November 2009. URL `http://www.geuz.org/gmsh/`, version 2.4.

[61] Hoffmann CM. *Geometric and solid modeling: an introduction*. Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 1989.

[62] Ko K, Maekawa T, Patrikalakis N. Computational geometry. MIT Open Courseware 2003.

[63] Corkum BT. Three-dimentional triangulated boundary element meshing of underground excavations and visualization of analysis data. PhD Thesis, University of Toronto 1997.

[64] Hemmer M, Hert S, Kettner L, Pion S, Schirra S. Number types. *CGAL User and Reference Manual*. 3.8 edn., CGAL Editorial Board, 2011; 147–222.

[65] The CGAL Project. *CGAL User and Reference Manual*. 3.8 edn., CGAL Editorial Board, 2011.

[66] Hachenberger P, Kettner L. 3D Boolean operations on Nef polyhedra. *CGAL User and Reference Manual*. 3.8 edn., CGAL Editorial Board, 2011; 1871–1920.

[67] Rineau L, Tayeb S, Yvinec M. 3D mesh generation. *CGAL User and Reference Manual*. 3.8 edn., CGAL Editorial Board, 2011.

[68] Rineau L, Yvinec M. A generic software design for delaunay refinement meshing. *Comput. Geom. Theory Appl.* 2007; **38**(1-2):100–110.

[69] Miller GL, Talmor D, Teng SH, Walkington N. A delaunay based numerical method for three dimensions: generation, formulation, and partition. *STOC '95: Proceedings of the twenty-seventh annual ACM symposium on Theory of computing*, ACM: New York, NY, USA, 1995; 683–692.

[70] Brown ET, Bray J. *Analytical and computational methods in engineering rock mechanics*. Allen & Unwin, London ; Boston :, 1987.

[71] Alliez P, Tayeb S, Wormser C. AABB Tree. *CGAL User and Reference Manual*. 3.8 edn., CGAL Editorial Board, 2011; 3583–3616.

[72] Du Q, Wang D. Tetrahedral mesh generation and optimization based on centroidal voronoi tessellations. *International Journal for Numerical Methods in Engineering* 2003; **56**(9):1355–1373.

[73] Tournois J, Srinivasan R, Pierre A. Perturbing slivers in 3D Delaunay meshes. *Proceedings of the 18th International Meshing Roundtable*, Clark BW (ed.), Springer Berlin Heidelberg, 2009; 157–173.

[74] Cheng SW, Dey TK, Edelsbrunner H, Facello MA, Teng SH. Sliver exudation. *J. ACM* 2000; **47**(5):883–904.

[75] SIMULIA, Rising Sun Mills, 166 Valley Street, Providence, RI 02909-2499, USA. *Abaqus Analysis User's Manual*. 6.9 edn.

[76] IEEE 754-2008 standard for floating-point arithmetic August 2008, doi:10.1109/IEEESTD.2008.4610935.

[77] Irons BM. A frontal solution program for finite element analysis. *International Journal for Numerical Methods in Engineering* 1970; **2**(1):5–32.

[78] Duff IS, Reid JK. The multifrontal solution of indefinite sparse symmetric linear. *ACM Trans. Math. Softw.* September 1983; **9**:302–325.

[79] Coulomb CA. Essai sur une application des regles de maximis et minimis a quelques problemes de statique, relatifs a l'architecture. *Memoires de Mathematique & de Physique* 1776; **7**:343–82.

[80] Terzaghi K. Presidential address. *Proceedings of 1st International Conference for Soil Mechanics and Foundations Engineering, Cambridge, Mass.*, vol. 1, 1936; 22–23.

[81] Terzaghi R, Voight B. *Karl Terzaghi on Rockslides: The Perspective of a Halfcentury (Part 2)*. New York: Elsevier, 1979; 111–131.

[82] Karman Tv. Festigkeitsversuche unter allseitigem druck. *Zeitschrift des Vereins Deutscher Ingenieure* 1911; **55**:1749–1757.

[83] King LV. On the limiting strength of rocks under conditions of stress existing in the earth's interior. *Journal of Geology* 1912; **20**:119–138.

[84] Griggs DT. Deformation of rocks under high confining pressures. *Journal of Geology* 1936; **44**:514–577.

[85] Ide JM. Comparison of statically and dynamically determined Young's modulus of rock. *Proceedings of the National Academy of Sciences*, vol. 22, 1936; 81–92.

[86] Terzaghi K. Stress conditions for the failure of saturated concrete and rock. *Proceedings of American Society for Testing and Materials, 45*, 1945; 777–801.

[87] Jaeger JC, Cook NGW. *Fundamentals of Rock Mechanics*. Chapman and Hall, London, 1969.

[88] Coates D. *Rock Mechanics Principles*. Pergamon: Ottawa: Dept. Mines and Technical Surveys, 1966.

[89] Obert L, Duvall WI. *Rock Mechanics and the Design of Structures in Rock*. New York: Wiley, 1967.

[90] Muller J. *Rockslides and Avalanches*, vol. 2, chap. Josef Stini. Contributions to Engineering Geology and Slope Movement Investigations. New York: Elsevier, 1979; 95–109.

[91] Müller-Salzburg L, Fecker E. Grundgedanken und grandsätze der 'neuen österreichischen tunnelbauweise'. *Felsmechanik Kolloquium Karlsruhe 1978*, Trans Tech Publications, 1978; 247–262.

[92] Beer G. Modelling of inelastic behaviour of jointed media with the boundary element method. *Proceedings of 4th International Conference on Computational Plasticity, Swansea*, Pineridge Press, 1995; 1761–1772.

[93] Dünser C. Simulation of sequential tunnel excavation with the boundary element method. PhD Thesis, Institute for Structural Analysis, Graz University of Technology 2001.

[94] Carter JP, Xiao B. Coupled finite element and boundary element method for the analysis of anisotropic rock masses. *Proceedings of International Symposium on Application of Computer Methods in Rock Mechanics and Engineering, Xian, China*, vol. 1, 1993; 249–258.

[95] Zheng C, Bennett GD. *Applied Contaminant Transport Modeling.* Second edn., John Wiley & Sons, 2002.