Scholars' Mine

Masters Theses

Student Theses and Dissertations

Spring 2024

A Binary Integer Linear Programming Model for Optimizing Underground Stope Layout

Theophilus Mensah
Missouri University of Science and Technology

Follow this and additional works at: https://scholarsmine.mst.edu/masters_theses

Part of the Mining Engineering Commons

Department:

Recommended Citation

Mensah, Theophilus, "A Binary Integer Linear Programming Model for Optimizing Underground Stope Layout" (2024). *Masters Theses*. 8176.
https://scholarsmine.mst.edu/masters_theses/8176

This thesis is brought to you by Scholars' Mine, a service of the Missouri S&T Library and Learning Resources. This work is protected by U. S. Copyright Law. Unauthorized use including reproduction for redistribution requires the permission of the copyright holder. For more information, please contact scholarsmine@mst.edu.

A BINARY INTEGER LINEAR PROGRAMMING MODEL FOR OPTIMIZING UNDERGROUND STOPE LAYOUT

by

THEOPHILUS MENSAH

A THESIS

Presented to the Graduate Faculty of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE

in

MINING ENGINEERING

2023

Approved by:

Kwame Awuah Offei, Advisor
Eugene Ben-Awuah
Samuel Frimpong

© 2023

Theophilus Mensah

All Rights Reserved

ABSTRACT

Underground mine planning engineers face significant challenges when determining what geometry provides the most profitable and safe stope for extraction. Several techniques and optimization algorithms have been developed in recent years, but most fail to find optimal solutions because they are heuristic or LP-based without efficient geometric constraints. This thesis work proposes a two-dimensional binary linear programming (BILP) model for determining the optimal combination of blocks in a stope that maximizes the economic value of the layout of stopes for a sublevel deposit. The work draws from Queyranne and Wolsey's (2017 & 2018) formulations of tight constraints for bounded up/down times in production planning problems to formulate novel and efficient geometric constraints along with geotechnical and grade constraints for the stope layout optimization problem. Results from the model indicate that it is possible to formulate efficient shape constraints in LP-based approaches. The model used for the numerical example contained 144 valuable blocks out of 774 blocks. The BILP model selected 60 valuable blocks and 13 waste blocks that met all constraints translating into a maximum economic value of $34.4M in 1.83 hours within a gap tolerance of 0.00%. A series of experiments show that the model is sensitive to cutoff grade, stope frame size, pillar size, the number of stopes, and the optimization problem size. Depending on the input values for these key parameters selected, they impact the objective function value, the solution time and the final layout of stopes generated by the algorithm. The main limitation of the proposed model is that pillar constraints are implemented in the Z – X or Z – Y directions but not implemented diagonally.

ACKNOWLEDGMENTS

I would like to express my deepest gratitude to my academic advisor, Dr. Kwame Awuah-Offei, for his invaluable guidance, support, and encouragement throughout the course of this research. His expertise and wisdom have been instrumental in shaping the direction and outcome of this thesis. I would also like to thank the members of my thesis committee, Dr Samuel Frimpong and Dr Eugene Ben-Awuah, for their valuable feedback and insights, which have greatly contributed to the improvement of this thesis.

I am also grateful to the lovely people who made my stay in Rolla feel like home. Edie Carly Smith, Hussam Altalhi, Noah Adekunle Aluko, Eugene Gyawu, Albert Amponsem, Joy Amponsah and Jeffery Kwarteng, your unwavering support and encouragement throughout this journey is unmatched.

I would like to acknowledge the support and resources provided by the Mining and Explosives Department of the Missouri University of Science and Technology, without which this research would not have been possible.

Lastly, I thank my parents, Perpetual Oduro and John Mensah (RIP) as well as my sister Esmiranda Mensah for their unending love, prayers, and support towards and throughout my master's program.

TABLE OF CONTENTS

Page

LIST OF ILLUSTRATIONS

Page

LIST OF TABLES

Page

NOMENCLATURE

Symbol	Description
BILP	Binary integer linear programming
UG	Underground
SLO	Sublevel layout optimization
2D	Two (2) dimensional space
LHD	Load -Haul -Dump
MILP	Mixed-integer linear programming
MLP	Mixed-integer programming
FS	Floating stope
MVN	Maximum value neighborhood
MAs	Metaheuristic algorithms
EAs	Evolutionary algorithms
GAs	Genetic algorithms
SIAs	Swarm intelligence algorithms
PSO	Particle swarm optimization
Z	Vertical direction in blockmodel
X	Horizontal direction in blockmodel
NPV	Net present value
COG	Cutoff grade

Sets

I	Number of blocks in the Z direction in block model
J	Number of blocks in the X direction in block model
K	Number of stopes
W	Number of pillar blocks

Indices

i $= 1, 2, 3, \ldots, \mathrm{I}$	index for blocks in the Z direction in model
j $= 1, 2, 3, \ldots, \mathrm{J}$	index for blocks in the X direction in model
k $= 1, 2, 3, \ldots, \mathrm{K}$	index for stopes in layout
w $= 1, 2, 3, \ldots, \mathrm{W}$	index for pillar blocks

Decision Variables

$x_{ijk} \in [0, 1]$	Binary integer variable; equals one (1) if block (*i*, *j*) is mined in stope k; zero (0) otherwise
$z_{ijk}^{1} \in [0, 1]$	Binary integer variable; equals one (1) if block (*i*, *j*) is the topmost block in stope k; zero (0) otherwise
$z_{ijk}^{2} \in [0, 1]$	Binary integer variable; equals one (1) if block (*i*, *j*) is the leftmost block in stope k; zero (0) otherwise
$w_{ij}^{1} \in [0, 1]$	Binary integer variable; equals one (1) if block (*i*, *j*) is the topmost block of a pillar; zero (0) otherwise
$w_{ij}^{2} \in [0, 1]$	Binary integer variable; equals one (1) if block (*i*, *j*) is the leftmost block of a pillar; zero (0) otherwise

Parameters

P	Unit Price of metal
C_s	Unit cost of refining and selling metal
C_{min}	Unit cost of mining tonne of rock in block (*i*, *j*)
C_{pro}	Unit cost of processing a tonne of ore in block (i, j)
Rec	Processing recovery of metal in block (*i*, *j*)
g_{ij}	Average grade of metal in block (*i*, *j*)
T_{ij}	Tonnage in block (*i*, *j*)
EBV_{ij}	Economic value of a block (*i*, *j*)
G_{off}	Stope cutoff grade
α_1	Minimum mining height in Z-direction

α_2	Minimum mining width in X-direction
β_1	Maximum mining height in Z-direction
β_2	Maximum mining width in X-direction
γ_1	Minimum pillar length in Z-direction
γ_2	Minimum pillar length in X-direction

1. INTRODUCTION

1.1. BACKGROUND

The underground mining industry exploits deposits that are deeply seated and economically infeasible to extract with open-pit mining techniques as illustrated below in Figure 1.1. Complexities in geology and geotechnical characteristics of the deposits require underground strategic mine planning engineers to select and create mine designs that are safe, cost-effective to operate, and yield the maximum value [1]–[3].

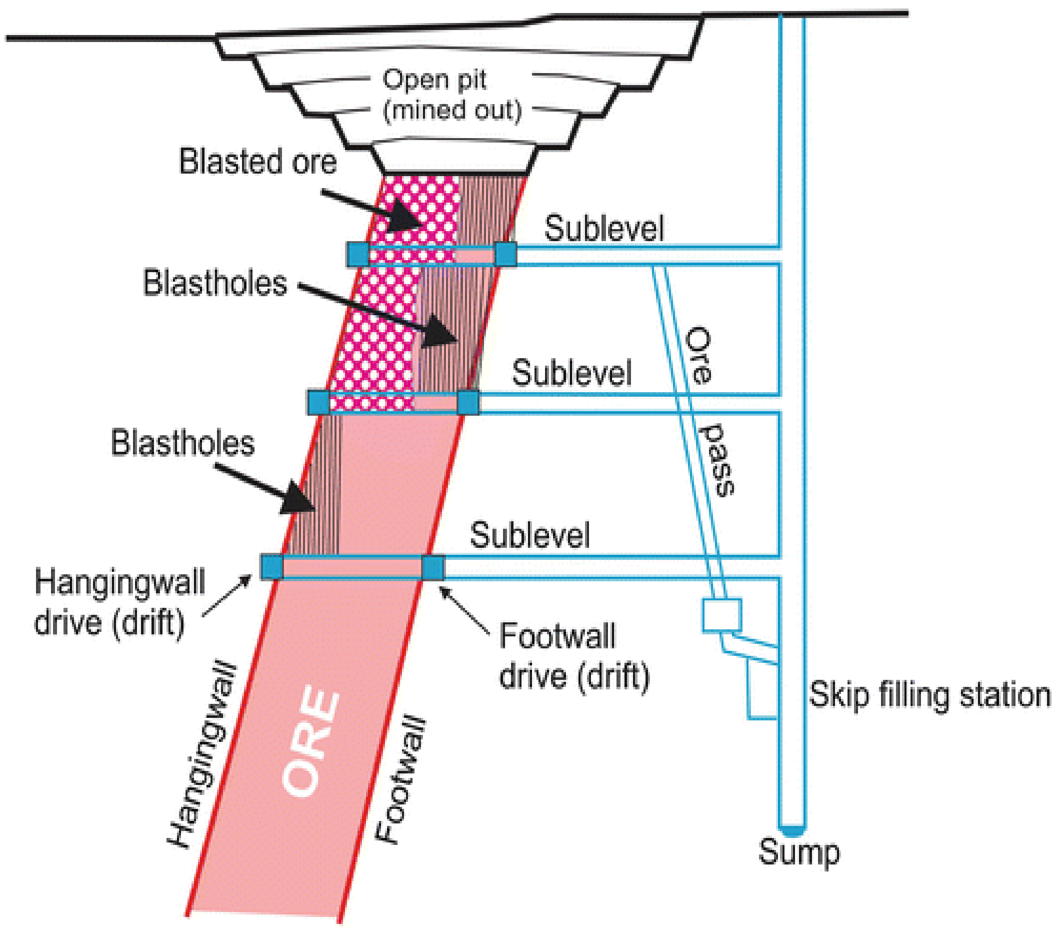

Figure 1.1 Typical Underground Mine [3]

Strategic underground mine planning involves a series of directly related activities such as orebody modeling, stope layout design, main access network design (e.g., main decline, ventilation raises, levels, primary drifts, and crosscuts), equipment selection, and stope scheduling and sequencing. The first stage in value creation is the determination of an intuitive optimal layout of extraction zones (optimal stope layout) in the deposit by strategic mine planning engineers. This is done by using computer-aided design software. Figure 1.2 shows the layout of stopes in an underground mine.

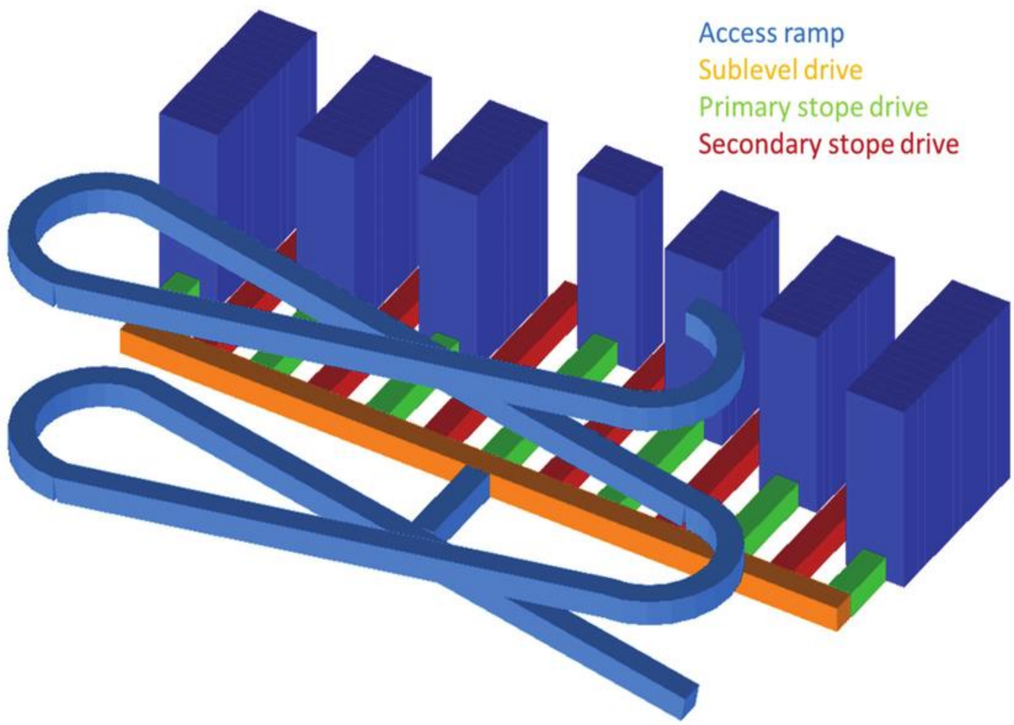

Figure 1.2 Typical Stope Layout in an Underground Mine

In stope-based methods (e.g., open stoping, sublevel stoping, and long hole stoping) that require stopes for the extraction of the minerals of interest, strategic planning engineers design several scenarios of mineable stopes based on 3D dtms/wireframe outputs from a commercial mining software (e.g., Maptek, Datamine, and Deswik) shown in Figure 1.2. This mining software considers economic as well as technical input parameters to perform multiple iterations that maximize the value of the mine for the investor. These input parameters include geologic models (block models), cutoff grades, economic parameters (e.g., metal prices, mining, and processing cost), geotechnical parameters (e.g., extraction ratio, hydraulic ratios, and pillar widths), and geometric parameters (e.g., minimum mining width and mining height) [4], [5].

One of the challenges of this approach is which 3D dtm/wireframe shapes and sizes maximize value (often, at this stage, just the present value of all blocks). Another challenge is the engines that drive this software are built off a heuristic programming technique. Shapes are simple to model using heuristic or non-linear methods, hence most of this commercial software includes heuristic algorithms in their optimization packages, although these algorithms do not guarantee an optimal solution [6], [7]. However mathematical optimization techniques such as LP-based algorithms (linear programming and mixed integer programming) have the benefit of generating an optimal result but modeling shapes in LP is relatively challenging due to LP's requirement that all formulations (objective function and constraints) must be linear.

However, Queyranne [26] has shown that with the proper formulation, it is possible to define efficient shape constraints in the LP-based algorithms that ensure contiguity and respect rectangular shapes in the determination of the optimal stope layout. This presents an optimization decision-making problem.

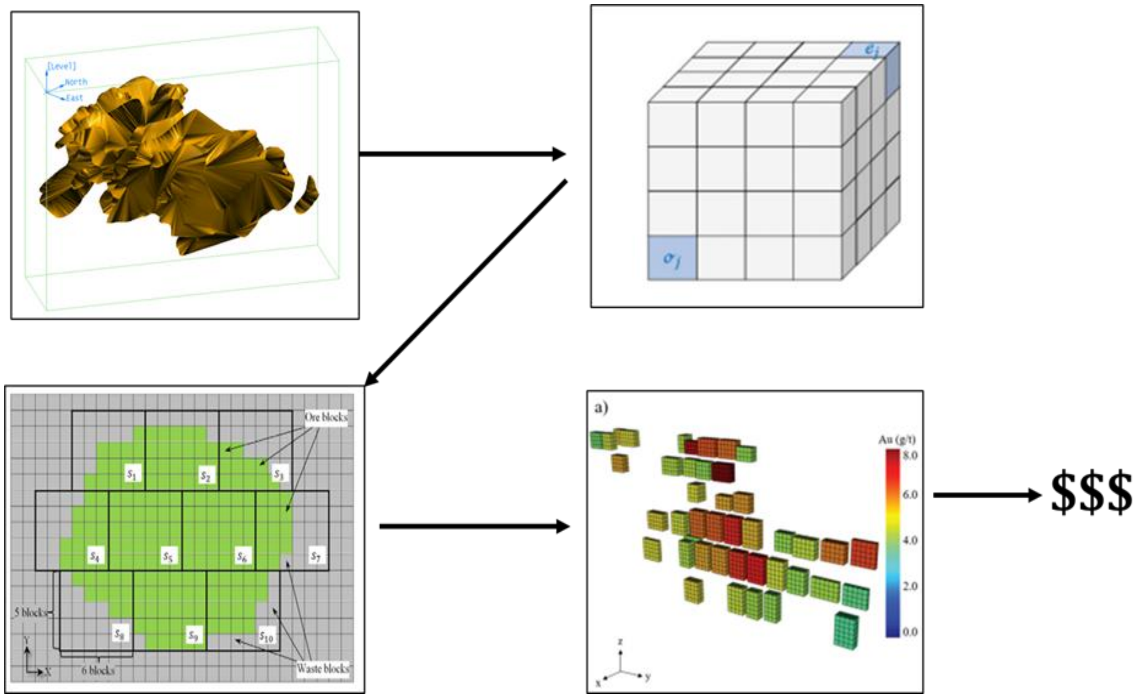

Figure 1.3 Strategic Planning Process

1.2. PROBLEM STATEMENT

Mathematical programming models have been used to assist strategic planning engineers in mine planning since the early 1960s. Compared to surface mining where there are many models and solution algorithms [8]–[10], the underground mining environment has fewer optimization models and solution algorithms thus the underground mine plan optimization problem remains largely unsolved due to its complexity [10]–[12]. Studies conducted on underground mine plan optimization suggest there are three main problems to consider [13]–[17]:

1. Stope layout optimization
2. Access and development network optimization and
3. Stope production schedules and sequence optimization.

However, the complexities surrounding the underground mine such as orebody orientations, mining method, geotechnical characteristics, the large number of variables and constraints to consider as well as the computational requirements, make the underground mine plan a challenge to solve wholistically with the existing algorithms [18], [19]. As a result, the developed algorithms only tackle one or two combinations of the underground mine planning problem using rigorous or heuristic methodologies [20] as shown in Figure 1.4. This thesis work will thus focus on the stope layout optimization problem.

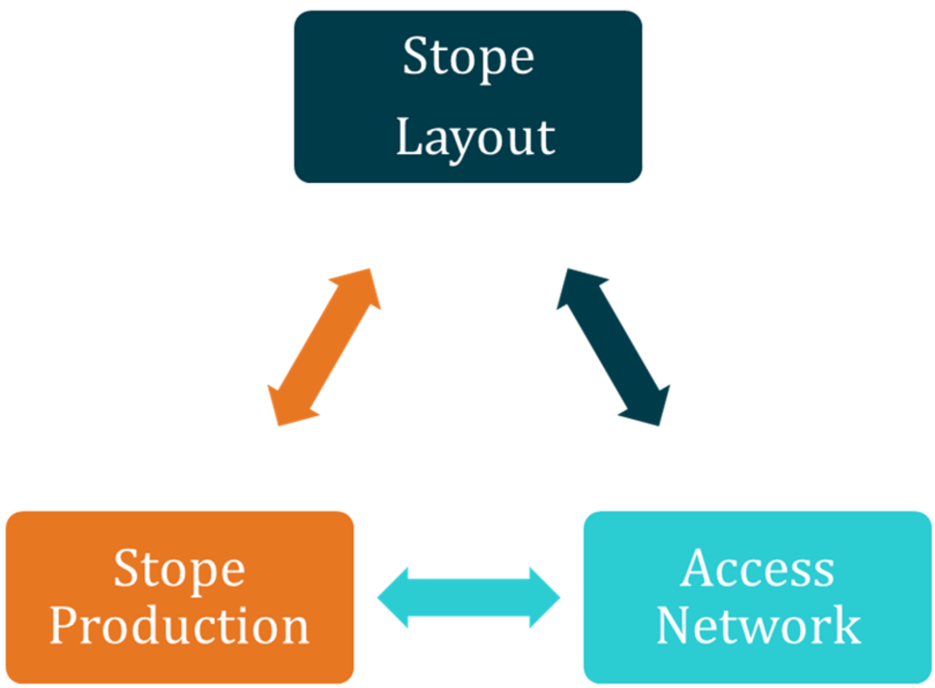

Figure 1.4 The Underground Mine Planning Problem

The optimal stope layout that maximizes a deposit's value while accounting for geotechnical requirements and grade quality constraints is the first step in the mine planning process for stoping methods. In recent years, several heuristic and linear programming (LP) techniques and optimization algorithms have been developed in both three-dimensional and two-dimensional space [5], [21]–[23] but most LP-based ones fail to account for efficient shape constraints that satisfy stability and operability constraints which leads to suboptimal solutions (i.e., even if the solution is optimal for the problem posed, if mine engineers have to adjust that solution to implement, the implemented solution is likely to be suboptimal). However, Queyranne's [25], [26] work has shown that it is possible to define efficient shape constraints that ensure continuity and respect rectangular shapes in LP with the proper formulation.

Therefore, it is essential to develop linear programming formulations of the stope layout problem that account for or adequately approximate such shape constraints [7], [24]. This thesis work, therefore, applies efficient shape constraints in a binary integer linear programming formulation of the stope layout optimization problem in two-dimensional space.

1.3. OBJECTIVES AND SCOPE OF RESEARCH

The overall objective of this thesis is to formulate the stope layout optimization problem (SLOP) as a binary linear problem that maximizes the value of the generated stopes subject to novel grade, geotechnical (minimum pillar sizes), and allowable mining (minimum and maximum stope width and height) constraints in two-dimensional space.

To achieve this goal, the author:

1. Draws from Queyranne and Wolsey's [25], [26] formulations of tight constraints for bounded up/down times in production planning problems to

formulate novel and efficient geometric constraints along with geotechnical and grade constraints for the BILP stope layout optimization problem.

2. Illustrates the novel BILP model with a sample gold mining data set to verify the model. The original geological model of the orebody was regularized to generate equal-sized blocks ideal for conversion into an economic model which serves as the primary input for the 5-experimental 15-scenario runs to verify the BILP model as a model that applies efficient shape constraints in solving the SLOP in two-dimensional space.

1.4. STRUCTURE OF THESIS

A comprehensive literature review on mathematical programming algorithms for optimizing underground stope layout is covered in Section 2 of this research thesis, with an emphasis on gaps. Section 3 presents a detailed explanation of the objective function and constraints of the proposed BILP model for optimizing underground stope layout. Section 4 illustrates how the BILP model was implemented on the sample data set and discusses the findings and results deduced from the implementation. Section 5, which is the last section of this thesis work, gives the conclusions of the study and recommendations for future work.

2. LITERATURE REVIEW

2.1. UNDERGROUND STRATEGIC MINE PLANNING

Underground strategic mine planning (SMP) is a challenging iterative process involving a series of directly related activities including orebody modeling, stope layout design, main access network design (e.g., main decline, ventilation raises, levels, primary drifts, and crosscuts), equipment selection, and stope scheduling. The strategic mine planning process for underground mining focuses on addressing three important criteria; operational safety, profitability, and environmental stewardship. A detailed technical and economic assessment of the deposit is required for every stage. Figure 2.1 illustrates the iterative and cyclical nature of the underground strategic mine planning process.

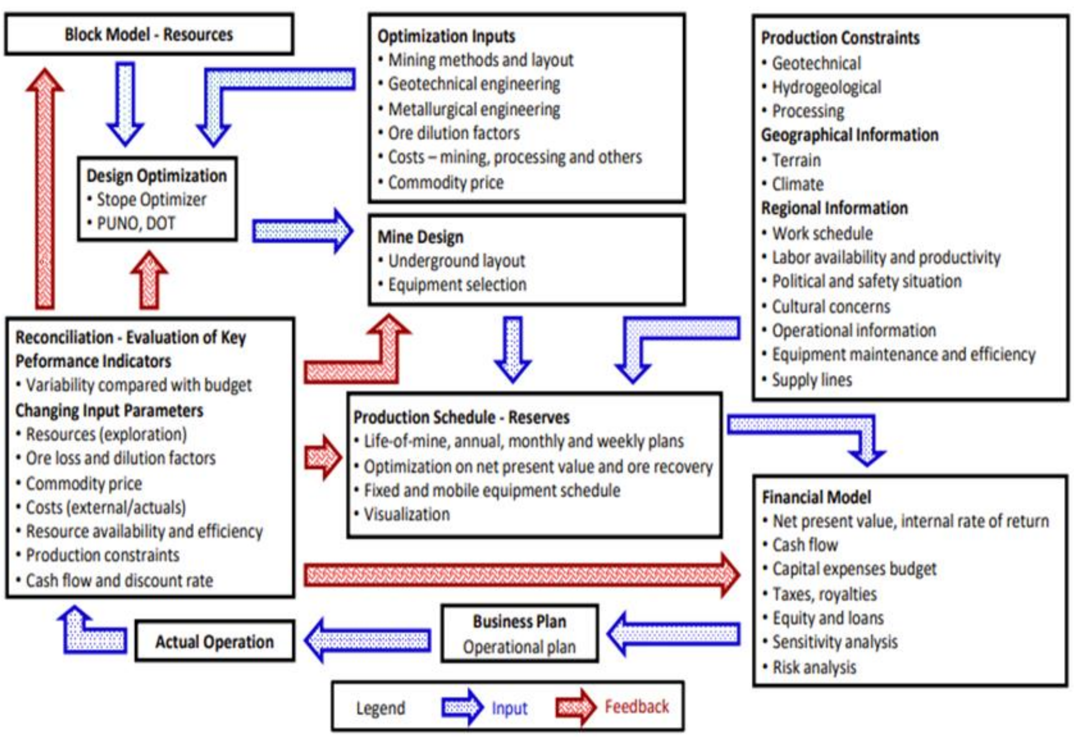

Figure 2.1 Underground Strategic Mine Planning Process [27]

The traditional workflow for mine planning is unable to evaluate a wide range of options (mine plans) within a reasonable amount of time. Furthermore, with traditional workflows, where only a few options can be considered, optimal profit is, often, not possible. Thus, current underground strategic mine planning processes preferably rely on the use of mathematical algorithms to handle large datasets, multiple constraints, and numerous variables to find the optimal mine plan that maximizes value [18], [28].

Due to the complexities surrounding the underground as well as the computational requirements, the underground strategic mine plan remains a challenge to solve wholistically with the existing algorithms [19], [28], [29]. Consequently, the underground strategic mine plan is subdivided into three (3) sub-problems; (1) Stope layout optimization, (2) Access and development network optimization, and (3) Stope production (scheduling and sequencing) optimization. The existing algorithms can tackle one or two combinations of the underground strategic mine planning problem using rigorous or heuristic techniques [8] as shown in Figure 1.4.

This section of the thesis focuses on reviewing the literature on heuristics, meta-heuristic and LP-based algorithms developed in the underground mine planning space for the stope layout optimization problem with a particular emphasis on gaps in geometric constraints included in the formulation of these algorithms. The author reviews literature gathered using a variety of abstracting indices including Google Scholar and OneMine.

2.2. MINE PLANNING FOR UNDERGROUND STOPING METHODS

The primary objective of mine planning is to maximize the recovery of ore while minimizing waste rock production, ensuring the safety of workers, and minimizing the environmental impact [30]. There are multiple underground mining methods an engineer can choose from to exploit any deposit [31]. To optimize value from the exploitation of mineral reserves, strategic and tactical decisions regarding the most appropriate mining method need to be made. Key considerations made in the selection of a mining method includes host and country rock geomechanical characteristics, mineralization style, orebody orientation, production scale, equipment, ground support systems and costs [3], [10], [32].

2.2.1. Underground Mining Method Selection. The choice of an underground mining method is an extremely important decision that affects the entire mining project. The selection of a suitable mining method relies on sound technical and economic evaluations of the deposit. These technical and economic analyses take into consideration geological characteristics (dip, size, quality, and shape of the orebody), geomechanical and geotechnical characteristics (strength of ore and host rock mass) as well as the economics (NPV, IRR, Payback Period) of the deposit [3], [11], [33]–[36]. Generally, the definition of a mining method permits:

- Mining equipment selection
- Stoping rate analysis
- Stope design configuration
- Mine layout configuration

Mining engineers can apply various underground mining methods to extract mineral reserves that are located at significant depths (Figure 2.2). According to the SME Handbook [2], some of the factors that must be considered when choosing an underground mining method include:

- Geological characteristics (extent, shape, and depth of the deposit)
- Mineralization (quality and distribution)
- Geomechanical/ Geotechnical conditions of host and country rocks.

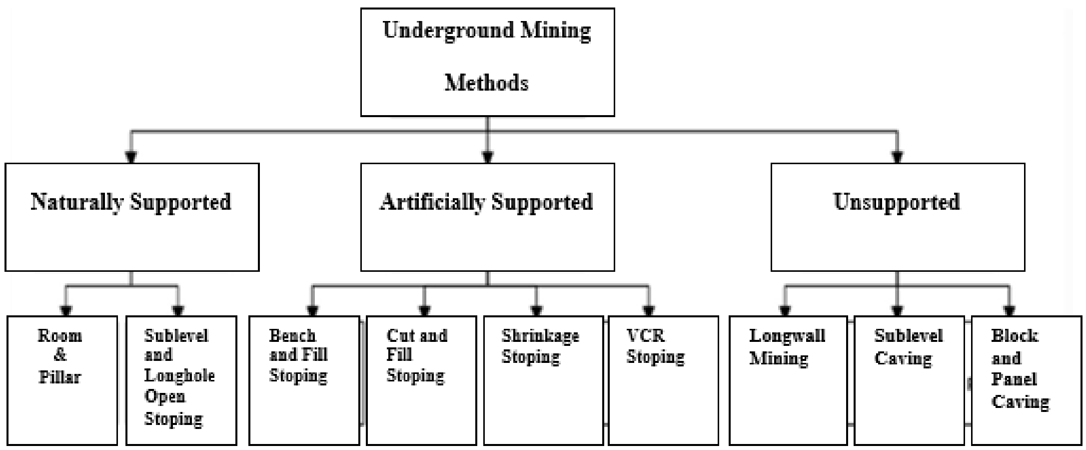

Figure 2.2 Underground Mining Methods

The interested reader can consult mining engineering resources [2], [31], [34], [37] to learn about the mining methods listed in Figure 2.2. In the next subsection of this work, the thesis describes naturally supported mining methods because the stope optimization algorithm developed in this work applies to these methods. Consequently, a background in stoping methods allows the reader to understand the context of the work.

2.2.2. Naturally Supported Underground Methods. Naturally supported underground mining methods are mining techniques that rely on the inherent stability and strength of the surrounding rock mass to safely extract valuable minerals from beneath the surface. These methods minimize the need for extensive artificial support systems and instead leverage the natural characteristics of the geological formations [2], [36]. Sublevel stoping (SLS) technique for ore extraction is a prominent self-supported, selective, and non-entry naturally supported underground mining method that is commonly used for extracting steeply dipping, ore bodies with a thickness greater than 10 meters [2]. The method involves splitting the orebody into horizontal tunnels known as production levels (primary levels) with stopes separated by pillars on these levels. These primary levels can be subdivided into sublevels for more control on production and ore selectivity. The stopes are then mined using a mechanized system of drill & blast and haulage. The method is particularly well-suited for mining large, low-grade deposits where high production rates are required [37]. Figure 2.3 shows a typical layout of a sublevel stoping method.

2.2.2.1. Primary level placement. In naturally supported methods such as sublevel stoping method, primary levels refer to the horizontal mining levels that are established within the orebody to extract the mineralized material. These levels serve as access points for mining activities, providing a platform for drilling, blasting, and mucking operations [2], [37]–[39]. Here are the key considerations for determining primary levels in the sublevel stoping method:

- Orebody geometry: Analyze the orebody's shape, size, and dip to determine the number and spacing of primary levels. The primary levels should be positioned at regular intervals to efficiently cover the entire orebody and ensure maximum ore recovery.
- Vertical interval: Determine the vertical spacing between primary levels based on the desired stope height and the mining equipment's capabilities. The vertical interval should provide enough room for the mining operations within each level while maintaining safe working conditions.

- Access and egress: Establish primary levels at suitable locations to facilitate efficient access to the orebody. Consider existing infrastructure, such as ramps, declines, or shafts, to minimize the distance and cost of accessing each level. Ensure that there are sufficient entrances and exits to accommodate personnel, equipment, and ore transportation.
- Hanging wall and footwall stability: Consider the stability of the hanging wall and footwall when determining the primary levels. Assess the rock mass quality, presence of geological structures, and potential for ground instability. Position the levels in stable rock formations to ensure the safety of workers and equipment.
- Ventilation: Account for ventilation requirements by establishing primary levels to allow for adequate airflow throughout the mine. Consider the direction of airflow and position the levels to facilitate efficient ventilation and the removal of gases, dust, and fumes generated during mining operations.
- Safety and emergency response: Ensure that escape routes, emergency exits, and refuge chambers are appropriately positioned and easily accessible from each level choosing the primary levels.

It's essential to note that the specific determination of primary levels may vary based on the specific characteristics of the orebody, mining regulations, and operational constraints. It is advisable to consult with mining professionals and engineers experienced in sublevel stoping to develop an optimized and safe primary level layout for a particular mining project [40], [41].

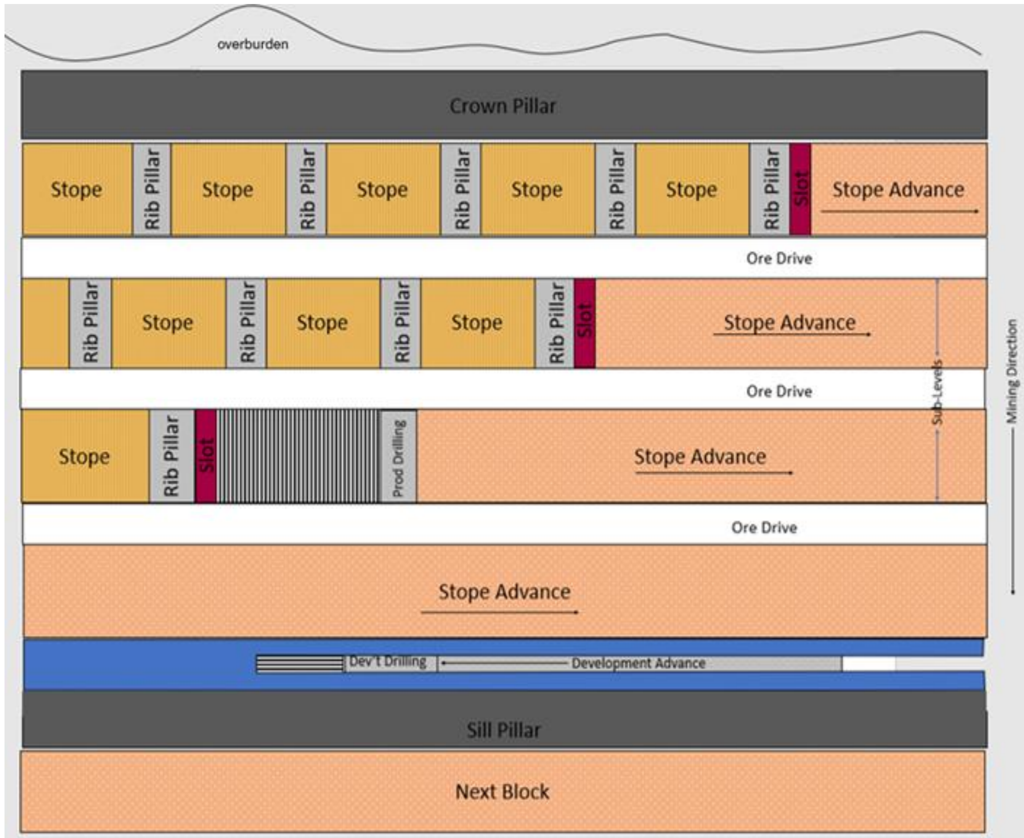

Figure 2.3 Underground Sublevel Stoping Method

2.2.2.2. Stope layout. The design of the layout of a sublevel stoping method is generally influenced by the geotechnical conditions of host and country rocks, production scale, equipment size, and the grade of material to deliver to the processing plant [21]. These decisions determine the size, shape, and location of the stopes with respect to the orebody. The production levels in sublevel stoping are constructed after the development of all accesses to the stoping areas. Stopes are generated with heights ranging from 30 to 120 meters in the orebody. Stope width is generally dependent on the equipment and orebody thickness. A raise or winze is operated into one corner of the stope from one sublevel to the next, followed by the provision of draw points. The method can be customized to limit the number of sublevels while increasing the height of stopes between sublevels to reduce development cost and time. A slot is then constructed for drilling and blasting within the stope to extract the ore. Extracted ore is transported by load-haul-dump (LHD) loaders and transferred to the underground crusher through an ore pass or to the surface crushers using trucks, conveyor, bin, or skip. The levels above the stope crown are protected, while neighboring stopes are separated by pillars. [37], [42].

A complete layout design of these stopes is the basis for evaluating the economic potential of a deposit and thus the reason to ensure that an optimal layout that maximizes value is designed [17], [30].

2.2.2.3. Pillar support. Pillar design is a crucial aspect of sublevel stoping mining as it plays a significant role in ensuring the stability and safety of underground mining operations. The primary purpose of pillars in sublevel stoping is to provide support to the overlying rock mass and prevent the collapse of the stope or caving of the hanging wall [42]. In a typical sublevel stoping operation, numerous pillars are utilized for ground control. Rib pillars are placed as support dividers between stopes that are horizontally contiguous. Some vertical slices are left behind as support pillars during production to help prevent subsidence within the stope. Another important support advancement in sublevel stopping is the use of sill and crown pillars. They are employed as a sill pillar between vertical stopes and as the crown pillar for the transition between surface and underground activities s [2], [33], [43], [44]. The design of pillars in sublevel stoping is influenced by a range of factors, including rock mass quality, seismicity, ore grade, stope geometry, and mining method [45]. The pillar size is determined based on the minimum required pillar dimensions to ensure the stability of the overlying rock mass and prevent excessive deformation. The ratio of pillar size to stope width (hydraulic radius) is typically between 0.2 and 0.4, depending on the rock mass quality, depth of the deposit, and mining method [45]. In general, larger pillar sizes with higher rock mass quality are necessary to maintain stability in deep-seated deposits. In sublevel stopping mining, the optimal pillar size and layout are commonly determined using numerical modeling and empirical techniques.

2.2.2.4. Stope grade. This Stope grade in sublevel stoping methods is a principal factor that determines the economic viability of the mining operation. It refers to the quality of the ore reserve material mined from a stope to meet the milling or processing requirement., the distribution of mineralization within the deposit, the quality of the geological and mineralization models, and The stope grade is influenced by various factors such as the geological characteristics of the orebody (thickness, dip, strike) the distribution of mineralization within the deposit, the quality of the geological and mineralization models as well as dilution. Generally, the aim is to maximize the value of the project by extracting as much ore from the deposit that exceeds the cut-off grade, while minimizing dilution. This is achieved by selecting the best location, size, and shape of the stopes within the orebody while ensuring geotechnical stability [2], [46], [47].

2.3. UNDERGROUND STOPE LAYOUT OPTIMIZATION

The stope layout problem involves determining the optimal arrangement of stopes in an underground mining operation, subject to various technical, economic, and safety constraints. In the stope layout problem, the objective is typically to maximize the net value (NV) of the mine, which is a function of the revenue generated by the ore mined and the costs incurred in mining and processing that ore. To do this, engineers must take into account several constraints, such as geotechnical, economic, and operational limitations [24], [48], [49]. Traditional approaches involve manually drawing stope shapes around ore blocks that meet the cutoff grade. This is tedious and leads to suboptimal solutions and it is not reasonable to do all the required iterations to evaluate all possible solutions [24], [50]–[52]. The underground stope layout problem is still difficult to solve holistically with the current algorithms due to the complexity of underground mines, the numerous variables to consider, and the computational requirements [19], [28], [29].

Mathematical programming models for mine planning have existed since the early 1960s with considerable development made in the surface mining space as opposed to the underground space particularly on stope layout optimization [7], [19], [20], [53], [54]. These algorithms involve formulating an objective function that seeks to maximize value and constraints such as ore grade, stope shape, and pillars, to find the optimal solution. The following literature review provides an overview of the methods and techniques used for optimizing underground stope layouts. Table 2.1 shows the main algorithms for solving the underground stope layout problem.

Table 2.1 Limitations of Stope Layout Optimization Algorithms

Classification	**Algorithm/Author(s)**	**Dimensional Space**	**Optimality**
Heuristics	Octree Division (1989) [55]	3D	No
	Floating Stope (FS) (1995)[50]	3D	No
	Maximum Value Neighborhood (MVN) (2000)[30]	3D	No
	Ataee-Pour (2000)[56]	3D	No
	Multiple Pass Floating Stope Process (MPFS) (2001)[57]	3D	No
	Topal and Sens (2010)[58]	3D	No
	Sandanayake and Topal (2010)[59]	3D	No
	Matamoros and Kumral (2017)[60]	3D	No
	Nikbin et. al (2017)[61]	3D	No
	Sari and Kumral (2021)[44]	3D	No
	Clustering-Based Algorithm (2021)[63]	3D	No
	Dual Interchange Algorithm (2022)[46]	3D	No
Rigorous	Dynamic Programming (1977)[64]	2D	No
	Downstream Geostatistics (1984)[13]	2D	No
	Branch and Bound (MIP) (1995,1999)[55]	1D	Yes
	Probable Stope (2004)[65]	2D	No
	Grieco and Dimitrakopoulos (MIP) (2007)[65], [66]	Not Indicated	Yes
	Network Flow (2013)[67]	3D	No
	OLIPS (2007)[65]	2D	Yes
	GOUMA (2015)[13], [65]	2D	Yes
	Samanta and Suranjan (2021)[5]	3D	No

2.3.1. The Stope Layout Optimization Problem (SLOP). From literature review, most early algorithms were purely heuristic. Heuristic algorithms, such as floating stope (FS), maximum value neighborhood (MVN), multiple pass floating stope (MPFS), and simulated annealing (SA), have been used to optimize stope layouts in several studies [50], [64], [65]. Most of the early developed algorithms utilized heuristic approaches because shapes are simple to model using heuristics or non-linear methods. Heuristic algorithms also have the advantage of finding a good near optimality solution quickly and are based on simple concepts or guidelines that are inspired by natural processes or engineers' intuition. However, heuristic stope layout optimization algorithms have several disadvantages including they result in sub-optimal solutions, they are sensitive to parameters changes and algorithm are often complex [18], [65] [30].

Most recent algorithms are meta-heuristic to overcome the limitations of the earlier developed heuristic algorithms. Examples include pattern search method algorithm [21], clustering-based iterative approach [63], greedy heuristic approach [6], dual interchange algorithm [46], simulated annealing [68], [69], and genetic algorithms [28], [70]–[72]. However, these algorithms do not guarantee optimality.

Mathematical optimization techniques, such as linear programming with its variations such as mixed-integer linear programming (MILP) and binary integer linear programming (BILP), provide a more efficient approach for stope layout optimization. [20], [23], [65]. One of the main advantages of LP-based approaches is that they can handle large-scale problems with many decision variables and constraints. They can also provide a globally optimal solution, if the problem satisfies certain conditions, such as convexity [73]. Several models have been developed such as mixed-integer programming (MIP) [7],

[8], [53], [58], [61], [63], Network flow models [74] [73] and integer programming (IP) [75]–[77].

However, LP-based approaches have some limitations when it comes to stope layout optimization. One of the main limitations is that they are limited by shape constraints [78]. In stope layout optimization, the shape of the stopes is often constrained by geotechnical considerations, such as stability and fragmentation. These constraints can take various forms, such as minimum width, maximum length, minimum height, and minimum distance between adjacent stopes. Thus, it is imperative that more research is conducted to develop effective and efficient mathematical algorithms to solve the underground stope layout problem.

2.3.2. Formulation of the SLOP with Heuristic Algorithms. As stated in section 2.3.1, earlier algorithms developed to solve the SLO problem were purely heuristic with limitations. Thus, to overcome these limitations, meta-heuristic algorithms have been applied to most recent model developments to solve the SLO problem [79]. Metaheuristic algorithms (MAs) are optimization algorithms that are designed to solve complex optimization problems that are difficult to tackle with traditional optimization methods. These algorithms are inspired by natural mechanisms and abstract concepts that aim to efficiently explore large solution spaces to find near-optimal solutions [62], [79], [80].

MAs methods can be categorized according to various factors, including the search strategy employed, the number of candidate solutions considered, and the extent of hybridization or memetic techniques utilized There are several kinds of MAs, such as Evolutionary Algorithms (EAs) and Swarm Intelligence Algorithms (SIAs).

Formally, the general procedure for MAs adapted from [79] is given in the following form:

Let X be a set of possible solutions
Find $x_o \in X$ such that maximize/minimize f
(i.e. $f(x_o) = max_{x \in X}(f(x))$ *or* $f(x_o) = min_{x \in X}(f(x))$
where $f : X \rightarrow R$.

EAs the most well-known population-based, global search Mas [70], [81]. EAs are meta-heuristic techniques that draw their inspiration from biological evolution-dependent phenomena including reproduction, mutation, and natural selection. In EA, the search space X is a set of chromosomes (i.e., DNA strings) regarded as candidate solutions for a given problem. Their fitness is evaluated by objective function (*f*). To discover the best solution, the fitness value of $x \in X$ should be kept as high or low as possible (i.e., the fittest chromosome). Evolutionary algorithms (EAs) including genetic algorithms (GAs), genetic programming (GP), evolutionary programming (EP) are examples of popular techniques developed under the EAs.

Swarm Intelligence Algorithms (SIAs) are optimization algorithms inspired by nature, specifically the collective behavior and interactions observed in animal colonies. The concept of "swarm intelligence" was coined by G. Beni and J. Wang in 1989 to describe these algorithms. Swarm Intelligence Algorithms aim to mimic the cooperative and adaptive behavior observed in natural swarms to solve complex optimization problems. [82].

The reason behind the widespread recognition and effectiveness of these algorithms lies in their inherent ability to learn autonomously, their flexibility, and their capability to adapt to changes originating from both external and internal factors [82]. In the space of stope layout optimization, several MAs have been developed to tackle the complex constraints, large solution spaces, and the need to find a good solution within a reasonable amount of time that characterizes the SLO problem. Examples include pattern search method algorithm [21], clustering-based iterative approach [63], greedy heuristic approach [6], dual interchange algorithm [46], simulated annealing [59], [60], and genetic algorithms [28], [70]–[72]. However, these algorithms do not guarantee optimality.

In the next subsections that follows, this work will provide a good overview of the application of these methods to the stope layout optimization problem, the general principle behind it, their application to the SLO problem and some limitations of using these approaches.

2.3.2.1. Simulated annealing process. Simulated annealing (SA) is a meta-heuristic algorithm that can be used to find the global optimum solution of a non-convex and non-linear optimization problem. The algorithm is based on the idea of simulating the physical process of annealing in metals, where the metal is heated and then slowly cooled to remove any defects in its structure and obtain a high-quality crystal lattice [83].

During the annealing process, a solid material, such as metal, is subjected to high temperatures to transform it into a liquid or molten state. This elevated temperature allows the atoms within the molten metal to move more freely. However, as the temperature is gradually decreased, the motion of atoms becomes increasingly constrained [84].

Krikpatrik et al. [85] introduced the concept of the simulated annealing (SA) algorithm, which can be applied to search for the global optimum of a complex function (combinatorial problem).

The algorithm starts with an initial solution and iteratively improves it by perturbing the solution and accepting or rejecting the new solution based on a probabilistic criterion that allows for escaping local optima [15], [69]. During each iteration of the simulation, a fresh configuration of the system is generated by introducing a random displacement to a randomly chosen particle from the current state. If the energy of the new state is equal to or lower than that of the current state, the new state is immediately adopted as the current state. However, if the energy of the new state is higher, it is still considered for acceptance, but with a probability determined by Boltzmann's probability distribution, Figure 2.4 shows the general SA algorithm [86].

The temperature parameter decreases over time, allowing the algorithm to escape from local minima and converge to a near-optimal solution [69]. Although an objective function and constraints can be used to define a problem that is addressed by simulated annealing, in practice the constraints are incorporated into the objective function as penalties [68], [87]. The SA algorithm can be used to optimize the stope layout by treating the problem as a combinatorial optimization problem. In this approach, the stope layout is represented as a binary string of 0's and 1's, where a 1 indicates that a stope is present and a 0 indicates that it is absent. The objective function is typically the net present value (NPV) of the mine, which is a function of the revenue generated by the extracted ore and the costs incurred in extracting the ore and waste (that is necessary to ensure feasible stopes) as well as the cost of processing ore.

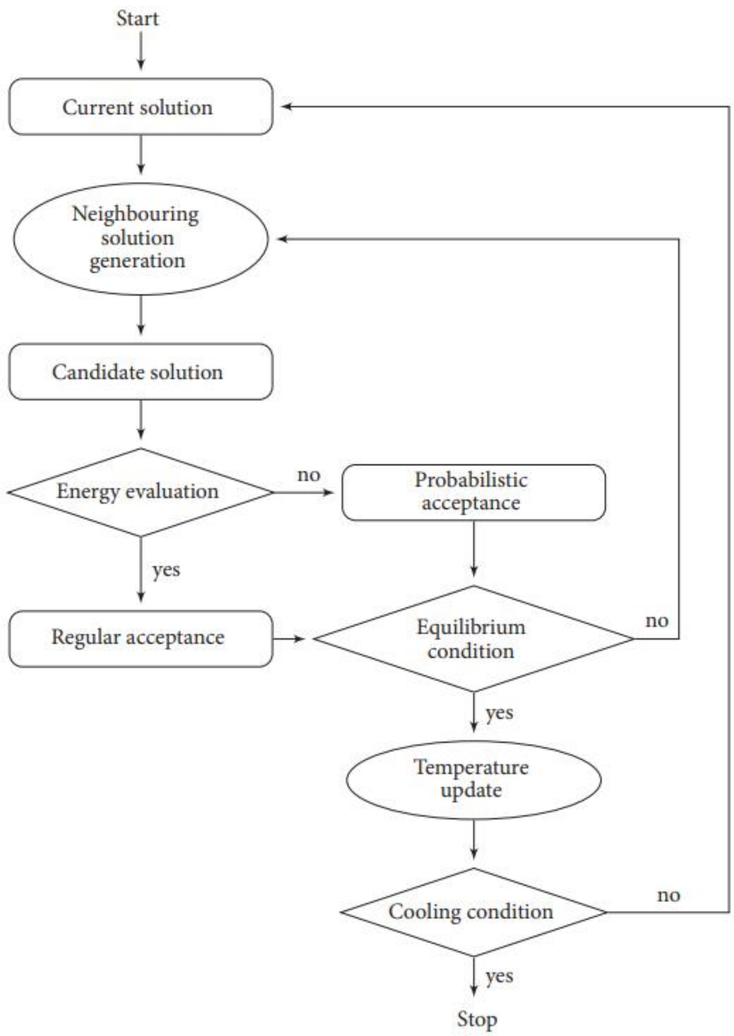

Figure 2.4 General Illustration of The Simulated Annealing Algorithm [86]

The constraints in the stope layout problem can be incorporated into the algorithm by using penalty functions. For example, geotechnical constraints such as minimum width, maximum length, and minimum height can be enforced by applying penalties to solutions that violate these constraints. Similarly, the economic and operational constraints can be incorporated into the algorithm using penalty functions [88]–[90].

Simulated annealing is known for its robustness and ability to find good solutions even when dealing with complex and non-convex optimization problems. This makes it particularly useful in situations where other optimization methods may struggle. Also simulated annealing is a global optimization method, meaning that it can find the global optimum of an objective function, rather than just a local optimum. This is important in many real-world optimization problems where finding the best possible solution is critical [68], [91]. However, simulated annealing can be slow to converge to a satisfactory solution, particularly for complex optimization problems such as the stope optimization problem, which deals with many variables. This can be a disadvantage in mine planning situations where time is a critical factor. Simulated annealing requires the user to set several parameters, including the initial temperature and cooling rate. These parameters can be difficult to set correctly and can have a significant impact on the performance of the algorithm. Although simulated annealing is a global optimization method, it can still get trapped in local minima, particularly for complex optimization problems. This can result in suboptimal solutions. Lastly, simulated annealing can be computationally intensive, particularly for large-scale optimization problems. This can make it impractical for certain applications.

In summary, simulated annealing is a metaheuristic optimization method that has found many applications in many fields including stope optimization, energy and many more[87], [90]. However, because of its drawbacks such as the inability for SA to guarantee an optimal solution, inability to model efficient shape constraints on the stope geometry and its computational intensity, it has not fully addressed the problem of stope layout optimization.

2.3.2.2. Genetic algorithm approach. Another popular meta-heuristic algorithm for stope layout optimization is the genetic algorithm (GA). Genetic algorithm (GA) is an evolutionary inspired meta heuristic algorithm, based on the mechanism of natural selection and biological processes of generating the fittest individual from a population [28], [71], [92]. Applied to search for an optimal solution, GAs have the capacity to improve solutions produced in the search space iteratively until a near optimal solution is generated. In the 1960s and 1970s at the University of Michigan, John Holland, his students, and colleagues pioneered and made popular the GAs [28], [81]. Since then, GAs have grown in popularity and the diversity of applications [81], [93]–[97]

The principle behind GA mimics the process of natural selection that works on a population consisting of competing individuals (i.e., chromosomes) where only the strongest individuals survive. GA (Genetic Algorithms) selects a pool of parents from the population using certain criteria, without relying on strict mathematical formulations, to generate the next generation. As a result, GAs are considered nonlinear, discrete event, and stochastic algorithms rather than being solely guided by mathematical rules. Crossover and mutation operators introduce new candidates into the population. The crossover operator creates offspring by exchanging parts of genetic information between two parents, while the mutation operator may modify certain genes in the offspring. The elitism operator merges the new population with the previous population and selects superior solutions from the combined population, ensuring that performance does not deteriorate. GA assesses the fitness of each individual using a fitness function. In the final generation, the fittest individual is regarded as the optimal solution [81], [98]–[100].. Figure 2.5 shows the workflow for the general principle of the GA algorithm [79], [98], [101]–[103].

The GAs have been widely applied to solve complex combinatorial optimization problems, including stope layout optimization in underground mining operations. In the GA approach, the stope layout is represented as a chromosome in a genetic population, and the objective function is the NPV of the mine. The geotechnical, economic, and operational constraints can be incorporated into the algorithm using penalty functions, as in the SA approach. Figure 2.6 shows the application of the GA to the stope layout problem [28].

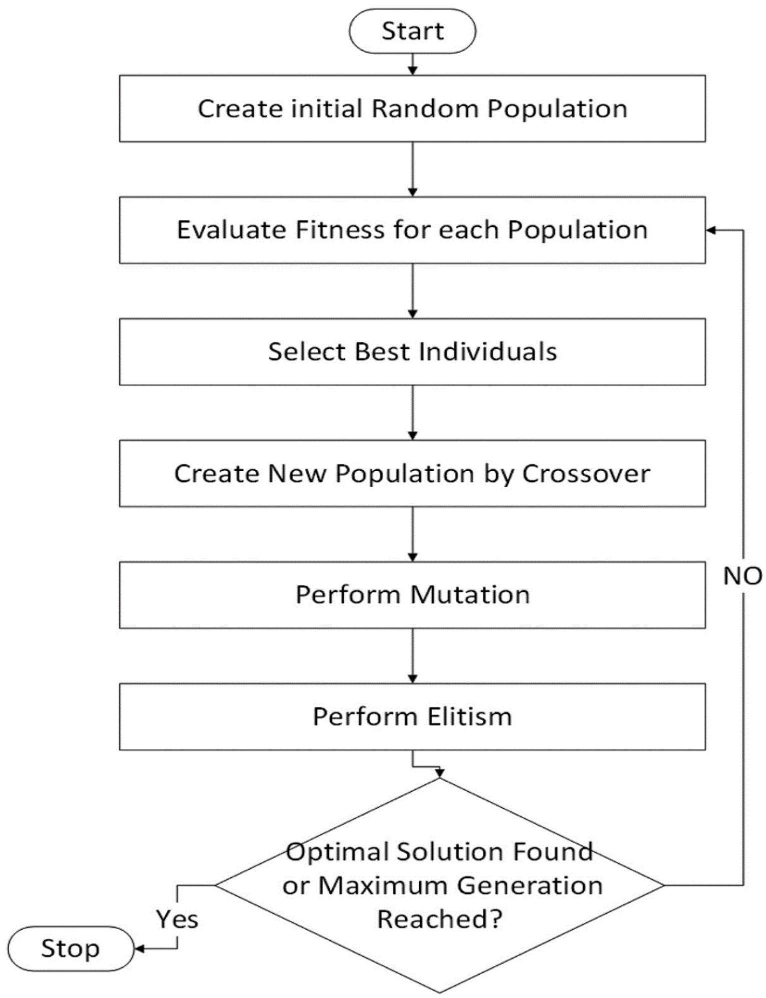

Figure 2.5 Genetic Algorithm Workflow Chart [71]

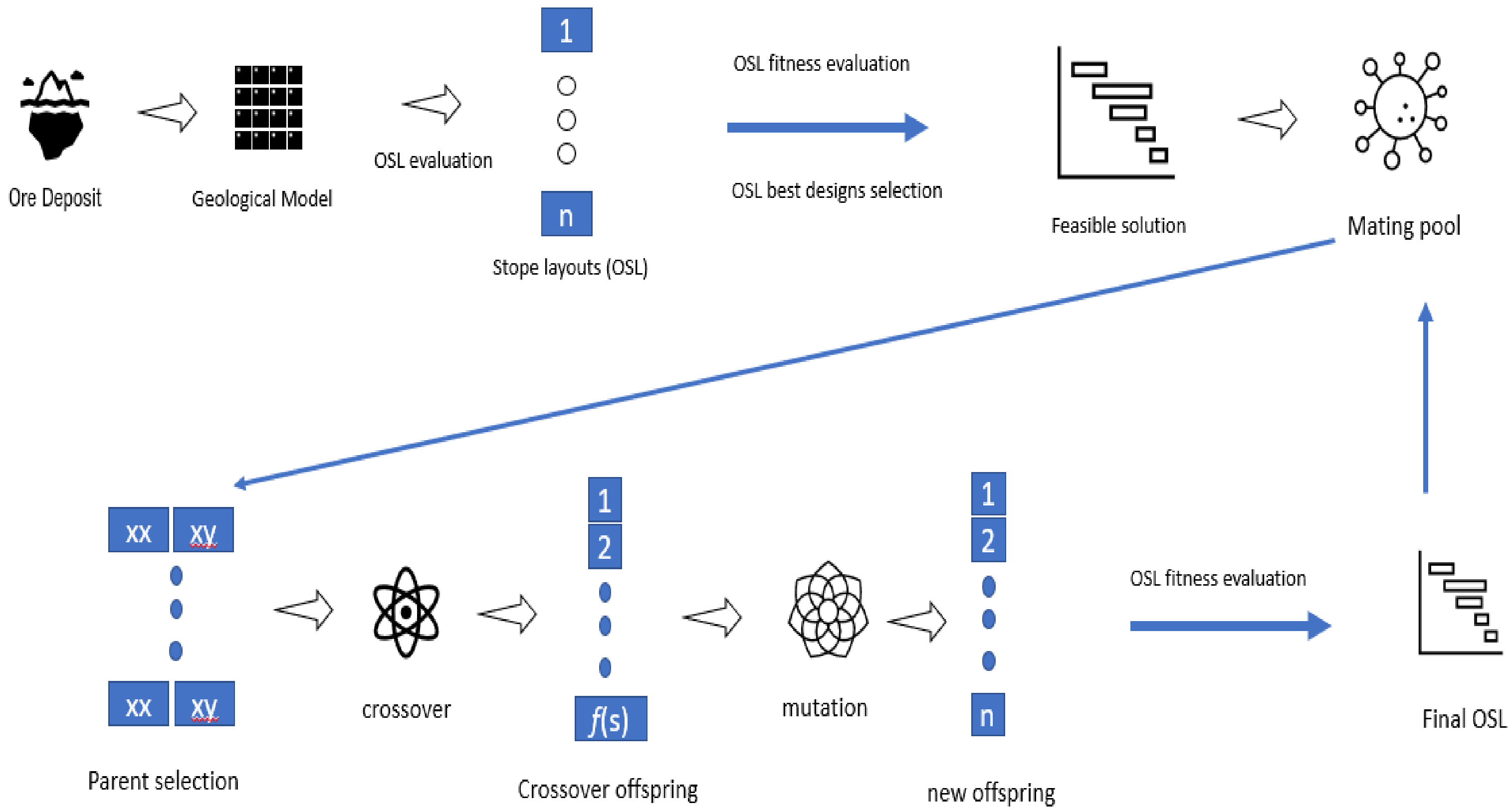

Figure 2.6 A Workflow of GA Applied to Stope Layout Optimization Problem [28]

GA is a population-based metaheuristic algorithm that can effectively explore the search space, allowing for a global search for optimal or near-optimal solutions. It has the potential to overcome local optima and converge towards better solutions, making it suitable for complex and multimodal optimization problems like stope layout optimization. GA can naturally lend itself to parallel implementations, as multiple solutions can be evaluated and evolved simultaneously. This parallelism can leverage modern computing architectures, speeding up the optimization process and providing opportunities for efficient utilization of computational resources.

GA can handle various constraints in stope layout optimization problems. Constraints related to stope geometry, operational requirements, geological considerations, and others can be incorporated into the fitness function or through customized genetic operators. GA's ability to maintain a diverse population enhances the chance of generating feasible solutions that adhere to the constraints.

GA also has several limitations some of which includes, tunning parameters. GA applied to the SLOP involves several parameters that need to be carefully tuned to achieve good performance. These parameters include the population size, crossover and mutation rates, selection strategies, and termination criteria. Finding the optimal values for these parameters can be challenging and often requires multiple trial-and-error iterations. Like any metaheuristic optimization algorithm, GA is susceptible to premature convergence, where the algorithm settles on suboptimal solutions without exploring the entire search space. Lastly GA's optimization process may yield optimal or near-optimal solutions, but the resulting stope layouts can be challenging to interpret and understand. The evolved solutions may not offer clear insights into the underlying reasons for their effectiveness, making it difficult to extract actionable knowledge or recommendations for the mining operation.

In summary, using the GA for stope optimization offers advantages such as global search capability, solution diversity, constraint handling, flexibility, and customization. However, it has drawbacks including parameter tuning challenges, potential for premature convergence leading sub-optimal solution and difficulties in interpreting the results. In SLO achieving an optimal solution is paramount to making decisions on investment as such algorithms developed needs to ensure optimal solutions are guaranteed.

2.3.2.3. Particle swarm optimization approach. This Another meta-heuristic technique that researchers have applied to the stope layout optimization problem is the particle swarm optimization algorithm (PSO) [62], [80]. Particle swarm optimization (PSO) is a metaheuristic optimization algorithm developed from swarm intelligence and the social behavior of bird flocking or fish schooling. The first PSO was presented by Kennedy and Eberhart to solve non-linear continuous optimization problems [104]. Particle swarm optimization is a search strategy that operates based on a population of flying particles. These particles dynamically adjust their velocities according to their own historical performance and the collective historical performance of the entire group, aiming to efficiently converge towards optimal solutions in the search space[82], [105], [106].

In recent years, PSO has been widely used as an effective tool to deal with many practical, real-life application problems such as stope layout optimization due to its ability to handle increasingly complex problems. PSOs' popularity and success have been linked to their capacity for self-learning, flexibility, and adaptability to both internal and external changes. as well as their ability to handle complex multi-objective problems and search for global optimal solutions [80], [107], [82], [108].

In PSO, the technique initializes a swarm of n random particles in the search space at random positions and velocities. The limits, the inertia factor (w), the cognitive and social characteristics (cs_1, cs_2), and the maximum number of iterations that will be carried out are also set during the initialization process of the algorithm. At each iteration, the objective function value for each particle at their current position is evaluated to determine its fitness. The particle's best position (x_{pbest}) as well as the swarm's global best particle position (x_{gbest}) is found at this step. To move closer to the *gbest* and *pbest* particles, the particle's current velocity and location are updated[62], [80], [82]. If any particle in the swarm turns out to be in a position that is better than the present position of the swarm's *gbest* particle, the index of the swarm's *gbest* particle is modified before an iteration ends. When the stopping requirement is satisfied, that is, when the maximum number of iterations have been finished, a good enough fitness value has been reached, or the algorithm has been producing the same result for a period of consecutive iterations, the iterative process is halted. The optimized function value is taken to be the fitness value of the *gbest* particle at the conclusion of the process [104], [106], [108]. The formula for updating the velocity and position [82] is, respectively, given by Equations (2.1) and (2.2):

$$v^{n+1} = \left(w * v^n\right) + \left(cs_1 * r_1 * \left(x_{pbest} - x^n\right)\right) + \left(cs_2 * r_2 * \left(x_{gbest} - x^n\right)\right) \tag{2.1}$$

$$x^{n+1} = v^{n+1} + x^n \tag{2.2}$$

Where:

v^{n+1} = velocity of the succeeding particle.

x^{n+1} = position of the succeeding particle.

x^{n} = particle's current position.

x_{pbest} = personal best position of the particle.

x_{gbest} = position of the global best particle of the swarm.

w = inertia factor, which controls the exploration capabilities of the algorithm

r_1 and r_2 are random numbers uniformly generated within the range [0,1].

cs_1 *and* cs_2 are positive parameters called the cognitive and social parameters respectively.

SI algorithms such as PSO have continuously developed over the years, leading to a surge in research demonstrating their rapid evolution and successful implementation in real-world optimization problems. Computational modeling of swarms using SI algorithms has expanded beyond operations research [109] to various domains like machine learning [110], business, and finance[110]. PSO has also been applied in engineering optimization problems such as stope layout problem [62], scheduling and routing problems [111]. SI algorithms are thought of as very promising optimization strategies due to the following traits:

1. PSO is known for its the capacity to search the whole optimization problem space. Finding global optimum or nearly optimum solutions is made possible by the extensive range of potential solutions it investigates [112].
2. PSO's implementation and understanding are both rather straightforward. In comparison to other optimization methods, it contains fewer tuning parameters, making it usable even by people with little optimization experience [111].

3. PSO has shown effectiveness in solving optimization problems with a high number of dimensions. It can handle problems with a large number of variables or decision parameters efficiently [113].
4. Fast Convergence: PSO has the potential to converge to good solutions quickly, especially in problems where the fitness landscape is relatively smooth and devoid of sharp local optima [80].

However, like any other metaheuristic algorithm, the PSO has some limitations:

1. PSO does not guarantee finding the global optimum in every optimization problem. Depending on the problem and parameter settings, PSO may converge to suboptimal or local optima instead of the global optimum.
2. The performance of PSO is highly sensitive to its parameter settings, such as the swarm size, inertia weight, cognitive, and social parameters. Fine-tuning these parameters to achieve good performance can be challenging and time-consuming.
3. PSO struggles with incorporating constraints in optimization problems. Ensuring that solutions adhere to problem-specific constraints can be challenging, requiring additional mechanisms such as penalty functions or repair strategies.

Overall, the use of meta-heuristic algorithms in stope layout optimization can be advantageous due to their ability to handle complex and non-linear problems, their efficiency in searching large solution spaces, and their ability to escape local optima. However, they have limitations in comprehensively solving the stope layout optimization problems such as the need for tuning algorithm parameters, and the possibility that they get stuck in a local optimum and do not find the global optimum solution.

2.3.3. Formulation of The SLOP as a Linear Programming Problem. Linear programming optimizes a linear objective function subject to linear equality and linear inequality constraints. The feasible region is a convex polytope, which is the intersection of half spaces defined by linear inequalities. The objective function is a real-valued linear function on this polyhedron. An algorithm for linear programming identifies a point in the polytope where the function has the smallest or largest value, if such a point exists [114] [115], [116]. Equation (2.3) shows the general form of the LP problem with decision variable **y**, and "cost coefficients" **c**.

$$\text{Maximize } \mathbf{c}^T\mathbf{y}$$
$$\textit{subject to } \begin{cases} \mathbf{Ay} \leq \mathbf{b} \\ \mathbf{y} \geq \mathbf{0} \end{cases} \tag{2.3}$$

$\boldsymbol{c} \in \mathbb{R}^{\boldsymbol{n}}, \boldsymbol{b} \in \mathbb{R}^{\boldsymbol{m}}$ are vectors and $\boldsymbol{A} \in \mathbb{R}^{\boldsymbol{m}\times\boldsymbol{n}}$ is a matrix. $\mathbf{y} \geq \mathbf{0}$ means that each component of **y** is non-negative. Several variations of this problem are possible; for example, instead of maximizing, we can minimize, or the constraints may be in the form of equalities, such as **Ay = b**.

An instance of Equation (2.3) where all the variables and constraints are restricted to integers is called integer linear programming (ILP) problem. A variation to this case is when all the decision variables must be binary (i.e., 0 or 1) is called the binary integer linear programming (BILP) problem. Mixed-integer linear programming (MILP), another variation to the ILP is when some of the variables are restricted to integers and some allowed to be non-integer variables [78], [117], [118].

The stope layout problem involves determining the optimal arrangement of stopes in an underground mining operation, subject to various technical/operational (allowable mining dimensions), economic (cutoff grade), and safety constraints (pillar requirements). The stope layout problem can, thus, be formulated as a linear programming (LP) problem, which involves maximizing or minimizing a linear objective function subject to a set of linear constraints [8], [14], [17], [24], [48], [55], [73].

An example of stope boundary optimization is represented as an LP formulation as follows by Alochukwu et.al., [119]. In their model, the SLOP is formulated as a 2D mathematical model where a binary decision variable y_{ij} is defined as $y_{ij} = 1$ if block (*i*,*j*) is mined and $y_{ij} = 0$ if block (*i*,*j*) is not mined. The block economic value V_{ij} is preprocessed using a value equation similar to Equation (3.1). These two variables are then used to define a maximization of the objective function that seeks to maximize the economic value of the stope layout generated. The mining area is represented by an $n \times m$ grid while the stope dimensions is a fixed $\alpha \times \beta$. Equations (2.4) – (2.9) summarizes their BILP model where *p* is minimum stope dimension in α direction and *q* is minimum stope dimension in β direction.

$$\text{maximise} \sum_{i=1}^{n-p} \sum_{j=1}^{m-q} V_{ij} x_{ij} \tag{2.4}$$

Subject to:

$$\sum_{i}^{i+p} \sum_{j}^{j+q} x_{ij} \quad \leq 1 \ \forall i \in \{1, \ldots n-p\}, \forall j \in \{1, \ldots m-q\} \tag{2.5}$$

$$x_{ij} - \sum_{j'=j+1}^{j+q} x_{ij'} = 1 \ \forall i \in \{1, \ldots n-p\}, \forall j \in \{1, \ldots m-q\} \tag{2.6}$$

$$x_{ij} - \sum_{i'=i+1}^{i+p} x_{i'j} = 1 \ \forall i \in \{1,...n-p\}, \forall j \in \{1,...m-q\} \tag{2.7}$$

$$x_{ij} - \sum_{i'=i+1}^{i+p} \sum_{j'=j+1}^{j+q} x_{i'j'} = 1 \ \forall i \in \{1,...n-p\}, \forall j \in \{1,...m-q\} \tag{2.8}$$

$$p = \alpha - 1 \ and \ q = \beta - 1, \ x_{ij} \in \{0,1\} \tag{2.9}$$

This approach by Alochukwu et.al., [119] has some limitations including the use of fixed stope dimensions, lack of flexibility to adapt to the peripheries of the deposit, and the fact that they are not well formulated. The authors define fixed stope dimensions ($\alpha \times \beta$), which indicates the inability of this model to generate variable stope dimensions. Models that do not give flexibility to adopt to the deposit's peripheries will most likely generate suboptimal shapes. The model does not integrate explicit formulations for level constraints or geotechnical constraints, which are essential factors in effectively implementing naturally supported stoping methods. The absence of these constraints limits the model's ability to effectively account for the specific requirements and considerations related to maintaining stable mining levels and addressing geotechnical challenges associated with underground mining operations. Finally, the main problem with Alochuku et al.'s model is that these constraints are not well formulated. Equation (2.6) – (2.8) are the constraints that control how the model generates the stopes in the $n \times m$ grid. Equation (2.10) ensures a block (*i, j*) is mined at most once in a stope. Equations (2.11) – (2.12) control the selection of blocks in a stope (the mining constraint). The problem with these formulations for these constraints are that they are forward looking only. Consequently if x_{ij} is mined, the blocks ahead ((*i*+1) or (*j*+1) onwards) cannot be mined. But blocks behind it can be mined because those are not constrained, which will lead to solutions that always include the "left-hand" side blocks only. But even then, since each block has this same constraint, only one block can be mined in each index. Another limitation of this model is that the constraints are formulated in the “natural” decision variables, which will lead to an exponential growth in constraint equations leading to more computational time and resources is needed to solve this optimization model. Time is essential in mine planning where large scale models are used as input. Thus, such a model can negatively impact the optimization process significantly. Finally, this model does not incorporate stope grades into the model. Determining the material to include in a stope relies heavily on the required stope grade. The objective is to meet the processing plant requirements by including material that ensures the stope’s average grade meets or exceeds a specific cutoff grade. However, this model does not incorporate stope grade constraints. As a result, some stopes formed by the model may contain significant amounts of low-grade material since there is no control over the threshold of grades to include in a stope.

To enhance LP models, one can introduce efficient shape constraints, tighten the formulation of the constraints, and incorporate geotechnical and stope grade constraints. These additions will improve the practicality and effectiveness of the model in solving the stope layout optimization problem. This LP formulation of the stope layout problem can be solved using standard LP solvers, such as CPLEX or Gurobi. The solution provides the optimal selection of stopes subject to the given constraints. The LP formulation can also be extended to incorporate additional features, such as uncertainty, multiple objectives, and discrete variables, using appropriate modeling techniques [14], [49], [120].

2.4. FORMULATION OF GEOMETRIC CONSTRAINTS IN LP OPTIMIZATION PROBLEM

The Geometric constraints in linear programming (LP) optimization problems are constraints that restrict the feasible region of the problem to a certain shape by imposing limits on the decision variables. These constraints are usually applied to ensure that the solutions to the optimization problem are physically feasible and satisfy engineering requirements[121], [122]. The geometric constraints in LP problems can be formulated using linear equations or non-linear equations which can be linearized [115], [123], [124]. Non-linear equations cannot be directly solved within the framework of linear programming (LP) because LP models are based on linear relationships between decision variables and constraints. However, non-linear equations can sometimes be linearized or approximated to enable their inclusion in LP formulations using techniques such as the *Piecewise Linear Approximation (PLA)* [115], [123]. Inequalities can also be used depending on the specific problem and the nature of the constraints [114], [117]. Some general methods for formulating geometric constraints in LP optimization problems are described below.

1. Linear Inequalities: One approach to formulating geometric constraints in LP optimization problems is to use linear inequalities[5], [24], [73]. This approach involves specifying upper and lower bounds on the decision variables that reflect the geometric constraints. For example, in a stope layout optimization problem, the decision variable might be the size of a stope, and the geometric constraint might be a minimum height and width requirement[13], [49]. This constraint could be formulated as a linear inequality of the form:

$$\sum \varpi_i y_i \geq \psi^{\min} \tag{2.13}$$

Where:

y_i = binary decision indicating if a block is included (y_i =1) or not (y_i = 0)

ϖ_i = decision variable representing stope width

$\Psi^{\min}$ = minimum stope width requirement

2. Nonlinear Inequalities: The geometric constraints in LP optimization problems may be nonlinear and cannot be expressed as simple linear inequalities [125]–[127]. However, the feasible region must still be a convex set. In such instances, nonlinear constraints can be transformed into linear constraints using techniques such as piecewise linearization approximation [25], [123], [128]. Linearization involves approximating a non-linear function by a linear function in a particular region of the function's domain so the resulting linear equations can then be used to formulate linear constraints in the LP problem [123], [124].

More complex geometric constraints, such as those involving non-rectangular stope shapes or irregular boundaries, may require more complex formulations, such as nonlinear equations or more complex inequalities. The specific formulation depends on the specific problem and the nature of the geometric constraint [26], [75], [129].

Nhleko et.al [20] show that stope layout optimization problems consider several constraints in their formulation (Figure 2.7). Their study shows none of the algorithms developed include shape or geometry constraints in their models. LP problems do not contain shape constraints because they are nonlinear as such the heuristics techniques tend to be the ones that include shape constraints. Hence, most of the algorithms use heuristics techniques [18], [65].

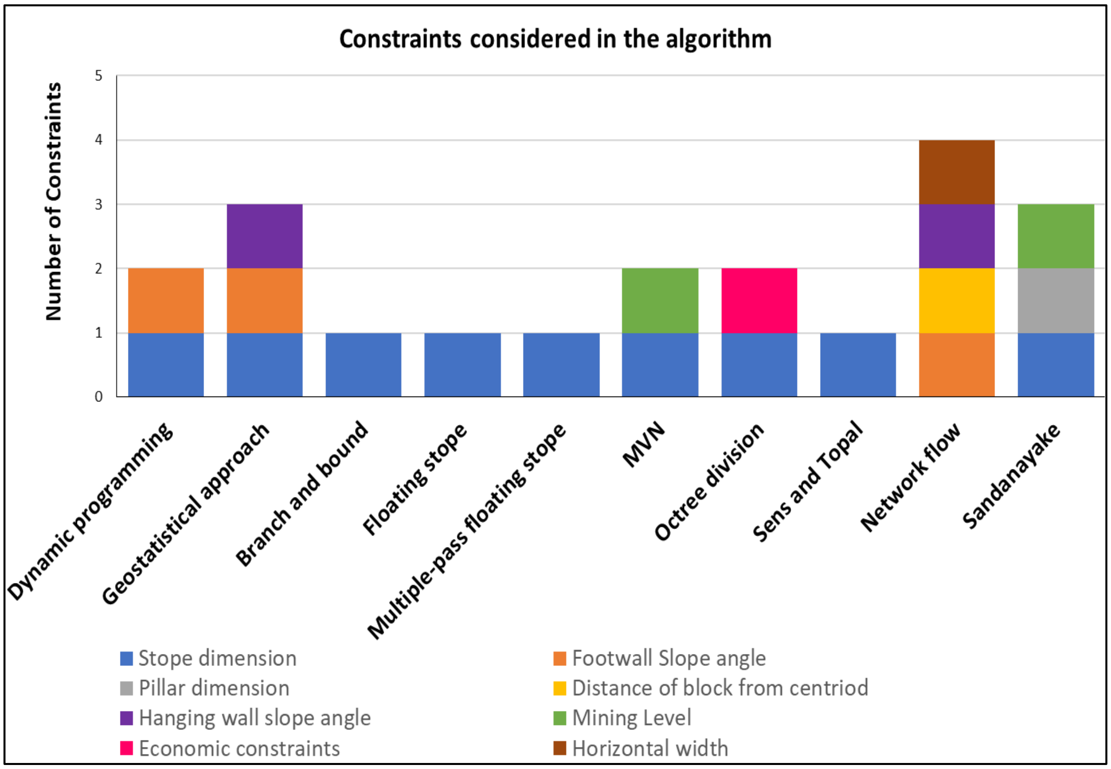

Figure 2.7 Constraints Considered in the SLO Algorithms [57]

However, Queyranne, in his work on production sequencing and mine production sequencing, has shown that with the proper formulation, it is possible to define efficient shape constraints in LP-based models that ensure contiguity and respect rectangular shapes [26]. Queyranne and Wolsey [25], [26] propose a unique approach to incorporating shape constraints into mine planning optimization models using extended formulations. The extended formulation approach proposed by Queyranne involves introducing additional decision variables to identify the first block in a sequence of mined blocks. The key idea is to reformulate the original problem in a higher-dimensional space ("natural" decision variables), where the shape constraints can be represented by linear constraints. The approach also provides a more compact and efficient representation of the problem. One of the main advantages of the extended formulation approach is that the resulting optimization problem is solved using LP or MIP techniques using standard LP and MIP solvers.

2.5. EFFICIENT LINEAR SHAPE CONSTRAINTS

Shape constraints are essential for underground mine planning because they play an essential role in ensuring that the shapes of the stopes or mine layouts in the solutions, meet operational and technical requirements for practical and safe extraction of the orebody. For instance, the LHD equipment must be able to maneuver inside the stopes while mining. Thus, not accounting for these constraints can lead to a loss of valuable ore material, increase stope dilution, stability issues and present a suboptimal mining operation [26]. As stated in Section 2.4 above, Queyranne and Wolsey [25], [26] presented an approach to incorporate linear shape constraints into mine planning optimization models using extended formulations.

The approach by Queyranne and Wolsey assumes a discrete (1D) series of blocks as in Figure 2.8. A stope starting with block *t* can have length at least α_t and at most β_t. Similarly, a pillar starting with block *t* can have length at least γ_t and at most δ_t. Their model defines the binary decision variables:

- γ_t = 1, if block *t* is a stope block; 0 otherwise.
- Z_t = 1, if block *t* is the leftmost of a stope. Z_t = 1 if γ_{t-1} = 0 and γ_t = 1.
- w_t = 1, if block *t* is the leftmost of a pillar. w_t = 1 if γ_{t-1} = 1 and γ_t = 0.

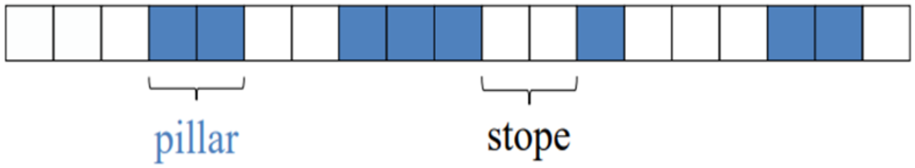

Figure 2.8 Series of Blocks for 1D Room and Pillar [26]

Based on these decision variables, Queyranne and Wolsey proposed the constraints described by Equations (2.14 – 2.19) as tight MIP formulations of 1D constraints [25].

$$Z_t \geq y_t - y_{t-1} \qquad t \in [1,n] \tag{2.14}$$

$$\sum_{\substack{u\in[0,t]:\\ u+\alpha_u>t}} Z_u \leq y_t \qquad t \in [1,n] \tag{2.15}$$

$$Z_t \leq \sum_{u=t+1}^{t+\beta_t} (1 - y_u) \qquad t: t \geq 0 \text{ and } t + \beta_t \tag{2.16}$$

$$w_t \leq \sum_{u=t+1}^{t+\delta_t} y_u \qquad t: t \geq 0\ and t + \delta_t \leq n \tag{2.17}$$

$$\sum_{\substack{u\in[0,t]:\\ u+\gamma_u>t}} w_u \leq 1 - y_t \qquad t \in [1,n] \tag{2.18}$$

$$y_t - y_{t-1} = z_t - w_t \qquad t \in [1, n] \tag{2.19}$$

Equations (2.14) and (2.15) establishes the formation of the leftmost block. The Equation (2.14) ensures that block *t* must be mined to be the leftmost point and Equation (2.15) ensures if block *t* is mine but block $t-1$ is not, then block *t* must be leftmost. Equation (2.16) ensures the contiguity control on block selection after the formation of the leftmost block Z_t. Equations (2.17) and Equation (2.18) ensure the formation of pillars between the stopes formed. Equation (2.19) establishes the link between the variables.

This 1D approach can be extended to 2D space. To do this, we can define two sets of 1D constraints to control blocks in each dimension. However, this approach will require twice the number of constraints and variables. Queyranne and Wolsey [25], [26] proposed a relaxation with two sets of 1D constraints, to reduce the complexity of the model. Each of these 1D problems is a special case of the bounded on/off interval (pillar placement) problem. However, this 1D relaxation leads to the formation of some blocks that are not covered by the shape template as illustrated in Figure 2.9. In Figure 2.9 Arrows indicate blocks where variables $Z_{ivw} = 1$ while dark blue blocks are those that would not be in a solution with two sets of 1D constraints but are contained in the relaxation constraints. These relaxed constraints do not lead to stopes that are operational as stopes in those solutions are “connected” rather than separated by pillars. Hence, the relaxed constraints, while more computationally efficient, cannot be used to generate feasible stopes. Consequently, in this thesis, the author uses the approach of two sets of 1D constraints.

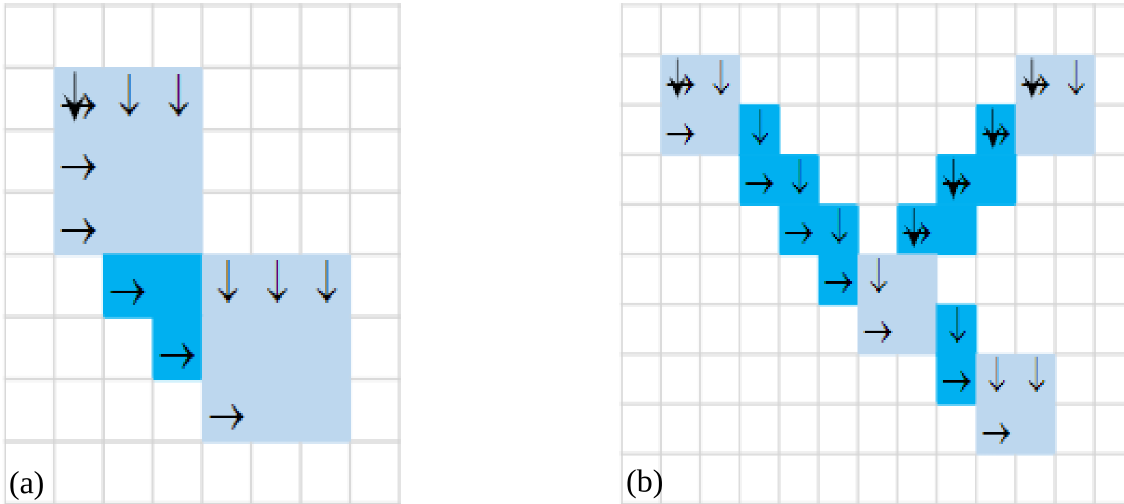

Figure 2.9 (a) 3×3 Rectangles and (b) 2×2 Rectangles. Arrows indicate blocks where variables $Z_{ivw} = 1$ while dark blue blocks are those that would not be in a solution with two sets of 1D constraints but are contained in the relaxation constraints

2.6. SUMMARY

This section of the thesis focused on reviewing the literature on heuristics, meta-heuristic, and LP-based algorithms for the stope layout optimization problem. The section also identified gaps in the literature regarding geometric constraints included in the formulation of these algorithms.

Heuristic models include the floating stope (FS) algorithm and the maximum value neighborhood (MVN) algorithm. The FS algorithm is limited because it produces overlapping stopes, which does not guarantee optimality. The MVN algorithm was developed to address this limitation, but it generates different optimal solutions based on the chosen starting point, and hence it does not guarantee optimality.

Recent algorithms in the literature are meta-heuristic algorithms such as the pattern search method algorithm, clustering-based iterative approach, greedy heuristic approach, dual interchange algorithm, and genetic algorithms. However, these algorithms also do not guarantee optimality.

LP-based approaches have also been developed, and one of their main advantages is that they can handle large-scale problems with many decision variables and constraints. They can also provide a globally optimal solution if the problem satisfies certain conditions such as convexity. However, LP-based approaches have some limitations, such as all the objective function and constraints equations must be linear. Shape constraints in stope layout optimization introduce complexity, such as minimum width, maximum length, minimum height, and minimum distance between adjacent stopes. MILP-based approaches have been proposed to deal with shape constraints, but they can be computationally expensive and may not scale well to large problems. However, Queyranne has shown that with the proper formulation, it is possible to define efficient shape constraints in LP-based models that ensure contiguity and respect rectangular shapes.

3. BINARY INTEGER LINEAR PROGRAMMING MODELING OF UNDERGROUND SUBLEVEL STOPE LAYOUT OPTIMIZATION

3.1. OVERVIEW

This section of the work will focus on the framework of the BILP stope layout optimization (SLO) model proposed in this thesis. The author will present the assumptions of the framework of the algorithm as well as a detailed description of the notations, variables, parameters, objective function, and constraints of the model. This section allows the reader to understand the context for the BILP mathematical model applied to SLO.

3.2. BILP MODEL FRAMEWORK

The goal of this thesis work is to formulate the stope layout optimization problem (SLOP) as a binary integer problem that maximizes the value of the mined stopes subject to novel grade, geotechnical (minimum and maximum pillar sizes), and allowable mining (minimum and maximum stope width and height) constraints in two-dimensional space. A key contribution of the work is to account for geotechnical and allowable mining constraints using efficient shape constraints.

The framework of the BILP model, as illustrated in Figure 3.1, starts by converting a geological resource model (blockmodel) into a regularized blockmodel. This is a key primary input for the BILP model. This regularized model contains block attributes such as quality (ore grades), density, geotech (joints, faults), processing (recoveries) as well as block dimensions [130], [131]. The regularized model is then converted into an economic model using technical and economic parameters (metal price, refinery cost, mining, and processing cost) supplied by the engineer. The economic model generated at this stage is the second key input for the BILP Model.

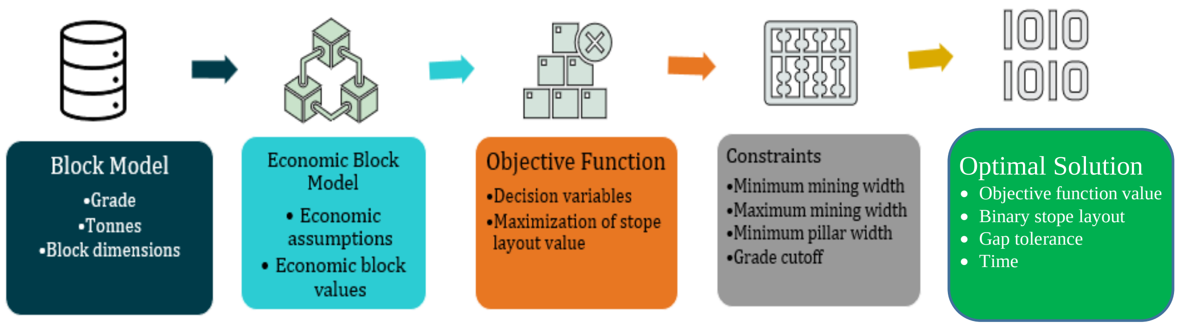

Figure 3.1 Framework of the BILP Model

The next stage in the framework is to model an objective function. As stated in Section 2.3 of this thesis, the objective in an underground stope layout optimization process is typically to maximize the economic viability of the mine to investors [13]. Thus, the goal of the objective function is to maximize the economic value of the deposit from mining and processing the optimal blocks from the entire set of blocks while respecting all constraints. The next stage in the framework is the application and consideration of operational (allowable mining dimensions), geotechnical (pillar requirements), and economic (cutoff grade, stope grade) constraints. This set of equations is modelled to constrain the selected blocks to generate a feasible combination of blocks into stopes that form the optimal layout. The last stage in the framework is the visualization and analysis of the optimal layout of stopes.

The following subsections in this section will provide details of each stage of the framework and provide a comprehensive description of key assumptions, primary inputs, decision variables, constraints as well as mathematical formulations developed for the BILP model.

3.3. MODEL ASSUMPTIONS

The thesis work and the modeling effort makes several assumptions. Some of the most critical assumptions considered are:

1. The model is limited to two-dimensional (2D) space for now to verify the possibility of modeling effective shape constraints in LP- based algorithms used to solve the SLO problem. However, the model is formulated in a way that makes it possible to later extend it into 3- dimensional (3D) space.
2. The model assumes a uniform material density to simplify the formulation although this does not cause any loss of generality (one can simply include a tonnage factor in the formulation to account for varying block densities).
3. Binary variables were used for modeling since it establishes the decision to include a block in the stope or not.
4. There is no cap on the number of stopes for the final design. However, the author assumed a reasonable number to allow for multiple stopes.
5. The block model is the primary input to generate the economic block values and it was regularized to have equal block sizes (no loss of generality because irregular block models can always be reblocked into regular block sizes).

3.4. DECISION VARIABLES, INDICES & SETS, AND PARAMETERS

This subsection provides details of the technical and economic parameters used for the conversion of block models into economic models. It also gives a comprehensive description of the decision variables, indices, and sets as well as the mathematical notations used to develop the BILP model. Tables 3.1 – 3.3 contain the definitions of the notations used for the decision variables the sets and the indices of each block in the model while Table 3.4 contains definitions for the parameters that was used in the economic block value calculation function.

Table 3.1 BILP Model Sets

Set	**Value**	**Definition**
I	$\{1, 2, 3, \dots, I\}$	Number of blocks in the Z direction in block model
J	$\{1, 2, 3, \dots, J\}$	Number of blocks in X direction in block model
K	$\{1, 2, 3, \dots, K\}$	Number of stopes
W	$\{1, 2, 3, \dots, W\}$	Number of pillars

Table 3.2 BILP Model Indices

Index	Value	Definition
i	$i = 1, 2, 3, \dots, I$	index for blocks in the Z direction in model
j	$j = 1, 2, 3, \dots, J$	index for blocks in the X direction in model
k	$k = 1, 2, 3, \dots, K$	index for stopes in the layout
w	$w = 1, 2, 3, \dots, W$	index for the pillar blocks

Table 3.3 BLIP Model Decision Variables

Index	Value	Definition
x_{ijk}	$x_{ijk} \in [0, 1]$	1 if block (*i*, *j*) is mined in stope *k*; 0 otherwise
z_{ijk}^1	$z_{ijk}^1 \in [0, 1]$	1 if block (*i*, *j*) is the topmost block in stope *k*; 0 otherwise
z_{ijk}^2	$z_{ijk}^2 \in [0, 1]$	1 if block (*i*, *j*) is the leftmost block in stope *k*; 0 otherwise
w_{ij}^1	$w_{ij}^1 \in [0, 1]$	1 if block (*i*, *j*) is a topmost block of pillar *w*; 0 otherwise
w_{ij}^2	$w_{ij}^2 \in [0, 1]$	1 if block (*i*, *j*) is a leftmost block of pillar *w*; 0 otherwise

Table 3.4 Technical and Economic Parameters

Parameter	**Unit**	**Definition**
P	$\$/oz$	Price of metal
C_s	$\$/oz$	Cost of selling (refinery) the metal
C_{min}	$\$/t$	Cost of mining a tonne of rock
C_{pro}	$\$/t$	Cost of processing a tonne of rock
Rec	%	Processing recovery of metal
g_{ij}	g/t	Grade of metal in a block (i, j)
T_{ij}	t	Tonnage of block (i, j)
EBV_{ij}	$	Economic value of a block (i, j)
G_{off}	g/t	Stope cutoff grade
α_1	m	Minimum mining height in Z-direction
α_2	m	Minimum mining width in X-direction
β_1	m	Maximum mining height in Z-direction
β_2	m	Maximum mining width in X-direction
γ_1	m	Minimum pillar length in Z-direction
γ_2	m	Minimum pillar length in X-direction

3.5. RESOURCE AND ECONOMIC MODEL

This section provides a detailed description of the geological resource model and the various mineralization domains as well as the block schema and the various attributes that will be useful in the optimization model.

3.5.1. Geological Resource Model.

3.5.1. Geological Resource Model. Block modeling is an essential tool for mineral resource estimation and mine planning in the mining industry. Block models are used to create reliable and accurate estimates of the location, size, and quality of mineral resources in a deposit, which is essential for assessing the economic viability of a mining project [132]. A geological resource model (block model) is a simplified mathematical representation of a geological deposit (ore body) and its surroundings discretized into small, regular-shaped blocks (cells). Each block is assigned attributes such as grade, density, and other geological and/or engineering characteristics of the mineralization distribution within the deposit [2], [131], [133].

The block model is created using a combination of geological, geophysical, and geochemical data collected from exploration activities, such as drilling, sampling, and mapping. The blocks are typically defined by their x, y, and z coordinates in an XYZ grid system, and the blocks may be of equal or of variable sizes depending on the resolution defined by the geologist [131].

The attributes of each block, such as grade, density, and other geological characteristics, are estimated using geostatistical and mathematical techniques, such as kriging or inverse distance weighting, based on the available data. The block model is typically validated using statistical and graphical methods to assess the accuracy of the model predictions and identify any areas of uncertainty [130]. Figure 3.2 illustrates a blockmodel section.

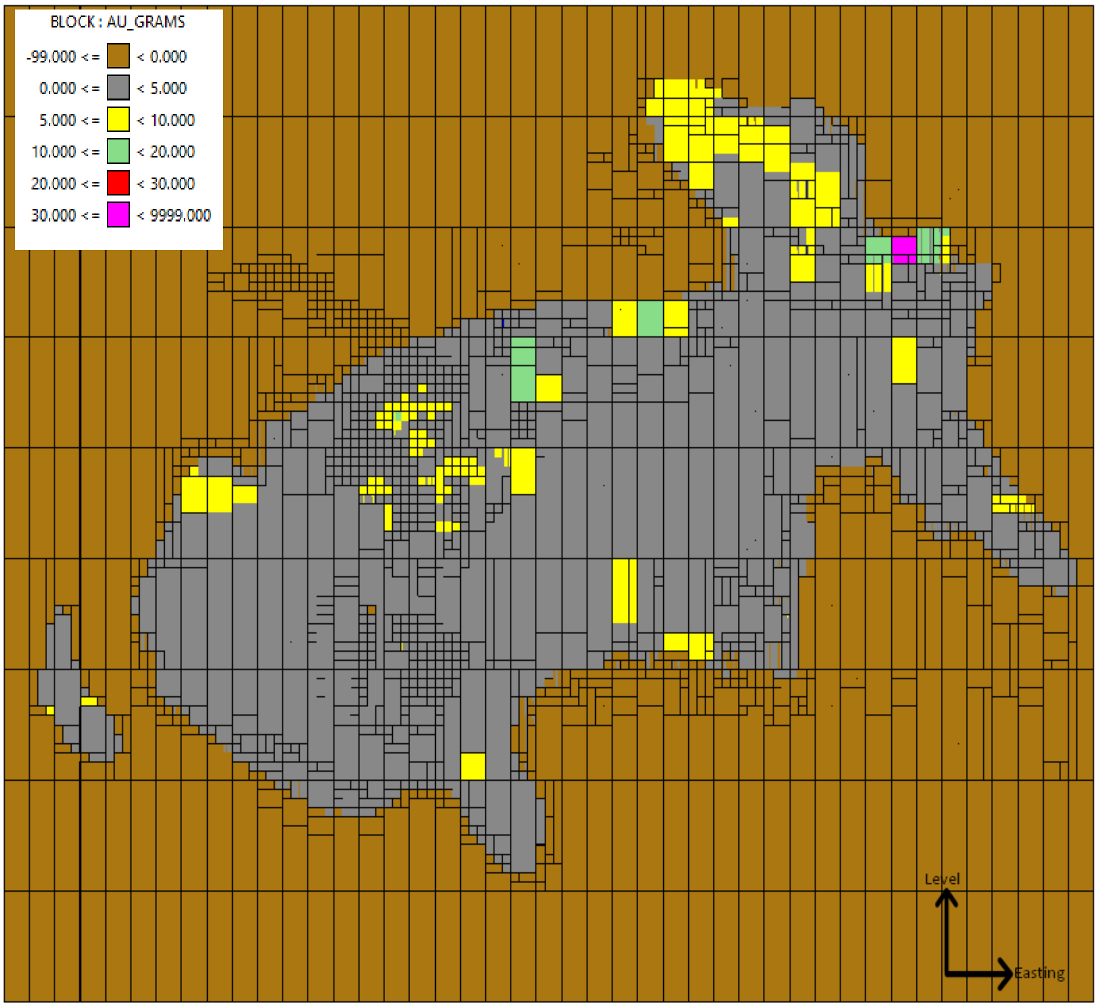

Figure 3.2 Sample Block Model Section

3.5.2. Economic Blockmodel. The economic block value (EBV) is one of the most important parameters considered in mine valuation. This parameter has considerable impact on important decisions like the ultimate open pit (OP) limit, final UG stope layout, the mining sequence and net present value (NPV) of a mining project. Therefore, it is necessary to calculate the EBV_{ij} at the first stage of the mine planning process, correctly. Unrealistic economic block value estimation may cause the mining project's managers to make the wrong decision and may consequently subject investors to unimaginable losses [134].

Each block within the geological block model has specific geological data, such as grade, volume, density, and lithology. The geological data together with technical and economic factors such as metal prices, mining cost, processing cost and mineral processing recovery rate are then used to calculate the economic value of each block (*i*, *j*) called the economic block value, EBV_{ij} thus converting it into an economic model. This economic model is a key input for the BILP model [133], [134]. Figure 3.3 illustrates the conversion of a geological blockmodel into an economic blockmodel.

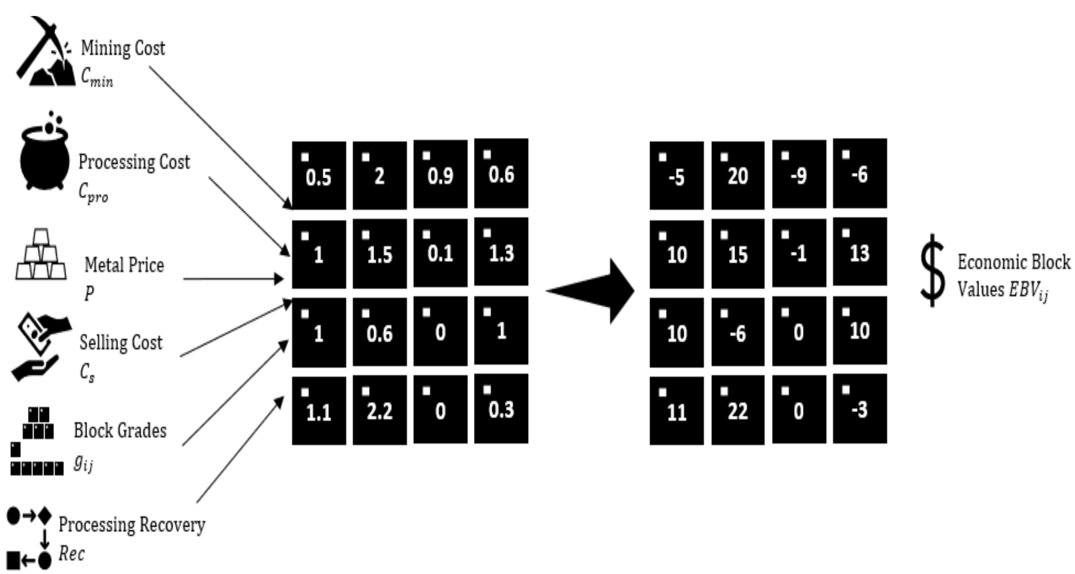

Figure 3.3 Economic Blockmodel Generation

Equation (3.1) is a generic mathematical formula used to estimate the EBV of each block (i, j) in the economic block model. EBV is the undiscounted revenue from mining and processing the block [76].

$$\left(EBV_{ij}\right) = \left[(P - C_s) \times g_{ij} \times Rec - \left(C_{min} + C_{pro}\right)\right] \times T_{ij} \tag{3.1}$$

Where:

$P = Price\ of\ Metal$

$C_s = Refinery\ and\ Selling\ Cost\ of\ Metal$

$g_{ij} = Grade\ of\ Block$

$Rec = Process\ Recovery\ of\ Metal$

$C_{min} = Mining\ Cost$

$C_{pro} = Processing\ Cost$

$T_{ij} = Tonnage\ of\ Block$

3.6. BILP MODEL FORMULATION

The BILP mathematical model applies efficient shape constraints in a binary integer linear programming model to find the optimal combination of mining blocks into stopes yielding the maximum value of a deposit. The work draws from Queyranne and Wolsey's [25], [26] formulations of tight constraints for bounded up/down times in production planning problems to formulate novel and efficient geometric constraints along with geotechnical and grade constraints for the stope layout optimization problem (see Section 2.5). The subsections that follow on in this section will describe the mathematical formulations of the BILP Model. The following subsections present the objective function, the operational/technical constraints, the geotechnical constraints as well as the grade constraints that are modelled using the decisions variables and notations.

3.6.1. Objective Function. The objective function of the BILP model is to maximize the economic value (undiscounted profit) of the optimal stope layout of the deposit. Equation (3.2) shows the objective function of this model, which is the sum of the block values of all blocks that are selected to be included in the optimal stope layout based on the value of the decision variable ($\boldsymbol{x_{ijk}}$).

$$Maximize \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} EBV_{ij} * x_{ijk} \tag{3.2}$$

Where:

$EBV_{ij} = Economic\ vlaue\ of\ block\ (i, j)$

$$x_{ijk} = \begin{cases} 1 \text{ if block } (i,\ j) \text{ is selected to stope } k \\ 0 \text{ otherwise} \end{cases}$$

3.6.2. Constraints. This section presents the constraints related to mining requirements, geotechnical requirements, grade as well as links between the variables. Following the example of Queyranne and Wolsey [25], [26], the author introduces new decision variables z^1_{ijk} and z^2_{ijk} to represent the topmost and/or leftmost block (*i, j*) in a stope *k*. The direct formulations of such constraints generally require exponentially many constraints in the natural decision variables. Using this new variable enables the author to model a compact constraint on the geometry forcing a more efficient stope shape. z^1_{ijk} is assigned to control the blocks along the Z-direction representing the stope height and z^2_{ijk} is assigned to control the blocks along the X- or Y-direction depending on which section of the deposit one uses for the optimization (for the remainder of the thesis, the work refers to the X-direction for simplicity; the reader should note the constraints are applicable to the Y-direction in the same way if the section is in the Z-Y plane). The constraints modelled are repeated along each coordinate direction. Other constraints are similar to previous work and intuitive.

3.6.2.1. Shape constraints.

- *Leftmost and/or Topmost Block Constraint:* To form a tighter and effective geometry the “selected" blocks must be contiguous. This work applies this concept to define constraints that ensure that each stope *k*, contains a set of contiguous blocks. To enforce this a block must be designated as the leftmost block ($z_{ijk}^{2} = 1$) along the X-direction and topmost block ($z_{ijk}^{1} = 1$) along the Z-direction in order to facilitate efficient formulation of the contiguity constraints along each direction in stope *k*. Figure 3.4 illustrates the corner blocks that enforce the contiguous selection of blocks ($x_{ijk} = 1$) into stope *k*. Equations (3.3) and (3.4) ensure that, if block (*i*, *j*) is the leftmost or topmost block of stope *k*, then the block is also mined in the stope. Equations (3.5) and (3.6) ensures that, if block (*i*, *j*) is deemed the leftmost or topmost block in stope *k*, then the preceding block (*i*-1, *j*) or (*i*, *j*-1) is not mined in that stope.

$$Z_{ijk}^{1} \leq 1 - x_{ijk} \quad \forall i, j, k \tag{3.3}$$

$$Z_{ijk}^{2} \leq 1 - x_{ijk} \quad \forall i, j, k \tag{3.4}$$

$$Z_{ijk}^{1} \geq x_{ijk} - x_{(i-1)jk} \quad \forall i, j, k \tag{3.5}$$

$$Z_{ijk}^{2} \geq x_{ijk} - x_{i(j-1)k} \quad \forall i, j, k \tag{3.6}$$

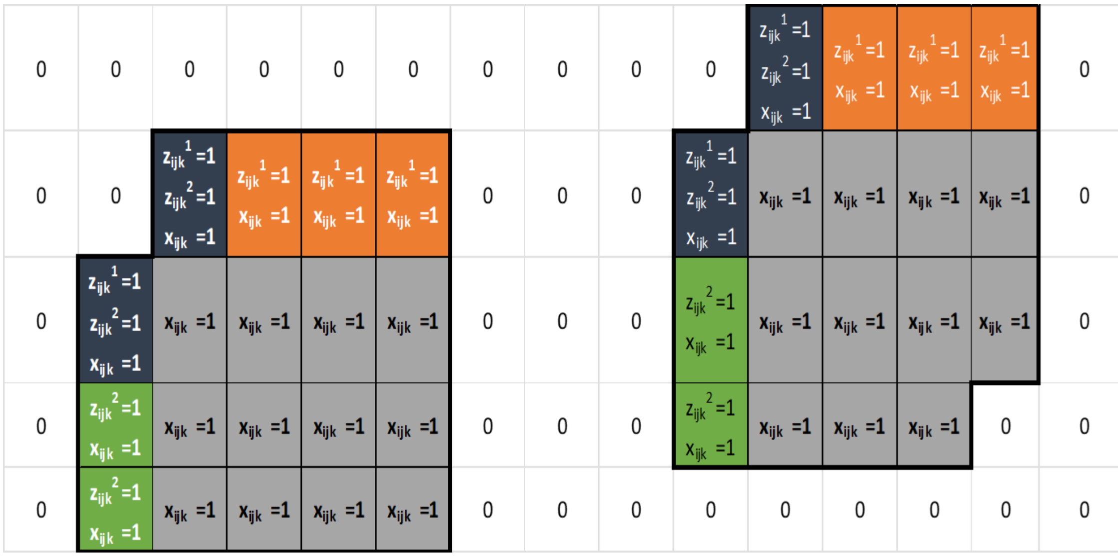

Figure 3.4 Corner Blocks Enforcing Block Contiguity

- *Block Contiguity Constraint (Operational Requirement):* Each stope must meet a minimum and maximum mining requirement for practical extraction of the ore. This is mine specific and varies based on geomechanical properties of the host rock and ore as well as the scale of operation [45]. There are numerous combinations of stope dimensions that can be used to produce a stope layout as seen in Figure 3.4 above. This permits the mining to follow irregular mineral deposit peripheries to minimize dilution, among other key considerations [135]. The following set of equations enables this work to model a constraint on the stope size (height and length dimensions). α_1 and α_2 are the minimum dimensions of the stopes, in the Z and X directions respectively, in units of number of blocks. Similarly, β_1 and β_2 are the maximum dimensions of the stope in the Z and X directions respectively. Equations (3.7) – (3.10) ensure all stopes meet the minimum and maximum dimensions in both directions.

$$\sum_{\varphi=\max(i-\alpha_1+1,1)}^{i} Z^1_{\varphi jk} \leq x_{ijk} \qquad \forall i,j,k \tag{3.7}$$

$$\sum_{\sigma=\max(j-\alpha_2+1,1)}^{j} Z^2_{i\sigma k} \leq x_{ijk} \qquad \forall i,j,k \tag{3.8}$$

$$\beta_1 - \sum_{\delta=\min(I,i+1)}^{\min(I,i+\beta_1)} x_{\delta jk} \geq Z^1_{ijk} \qquad \forall i,j,k \tag{3.9}$$

$$\beta_2 - \sum_{\vartheta=\min(J,j+1)}^{\min(J,j+\beta_2)} x_{i\vartheta k} \geq Z^2_{ijk} \qquad \forall i,j,k \tag{3.10}$$

3.6.2.2. Geological domain constraint.

- *Stope Limit Constraint:* Each stope *k* generated from the combination of block (*i*, *j*) must be spatially unique representing a specific domain in the deposit. Thus, the model needs a constraint to ensure only one stope is generated per slice of blocks from the block model.

  To do this, it is important to ensure the blocks mined in each contiguous selection of blocks is isolated into its own stope so that the resulting stope does not have multiple combinations of blocks within and avoid solutions where the cut-off grade constraints are applied across multiple "stopes". This requires a new set of constraints that are not based directly on Queyranne and Wolsey's [25], [26] work but uses the decision variables designating the leftmost or topmost blocks to ensure efficient constraints. This work proposes Equation (3.11), which limits the number of leftmost z_{ijk}^2 and topmost z_{ijk}^1 blocks (generally, this thesis refers to these as "corner" blocks) to less than the sum of the maximum number of allowable blocks in each direction ($\beta_1 + \beta_2$). While this will, technically, allow multiple stopes that exceed the minimum number of blocks constraint but for which the sum of corner blocks is still below the sum of the maximum blocks (i.e. $\beta_1 + \beta_2$), we find this to be rare. Additionally, Equation (3.11) results in much more efficient constraints than an attempt to write constraints for each individual block.

$$\sum_{k=1}^{K} Z_{ijk}^{1} + \sum_{k=1}^{K} Z_{ijk}^{2} \leq \sum_{n=1}^{2} \beta_n \quad \forall i, j \tag{3.11}$$

- *Stope Overlap Constraint:* As stated in Section 3.5.1, each block representing a portion of the mineralization in the deposit can only be mined in one stope. This restriction prevents overlapping of the stopes that will be formed from solutions that mine one or more blocks in multiple stopes (Figure 3.5 illustrates the types of solution the model should avoid). Equation (3.12) ensures this situation does not arise in the solution.

$$\sum_{k=1}^{K} x_{ijk} \leq 1 \qquad \forall i, j \tag{3.12}$$

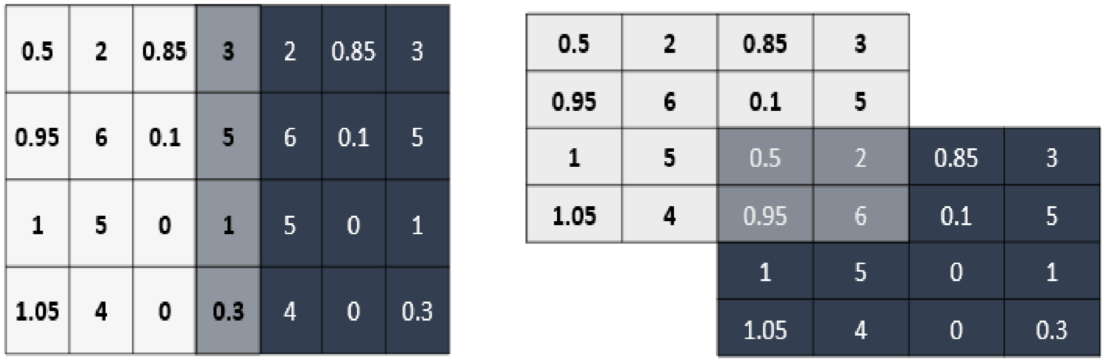

Figure 3.5 Stope Overlap Examples

3.6.2.3. Stope grade constraint. One aspect of stope layout optimization is the desire to ensure that each stope meets a certain cut-off grade (G_{off}) such that material mined from that stope can be sent to the mill to be processed. However, this does not mean every single block in the stope must have a grade above the cut-off grade but only that the average grade of all blocks in a stope exceed the cut-off grade. Equation (3.13) ensures that the average grade of the blocks in each stope *k* meets the set cutoff grade, G_{off}.

$$\sum_{k=1}^{K} \frac{(g_{ijk} \times x_{ijk})}{x_{ijk}} \geq G_{off} \qquad \forall i, j \tag{3.13}$$

3.6.2.4. Geotechnical pillar constraints. As stated earlier in Section 2.2.2.3, the geotechnical characteristics of the rock mass surrounding stopes play a critical role in determining the stability and safety of underground mining operations. The operational activities (blasting and mining) from adjacent stopes, as well as the presence of geological structures can affect the stability of the stopes generated in the layout. Therefore, it is important to incorporate geotechnical constraints into stope layout optimization in the form of minimum and maximum pillar dimensions to ensure a layout that maximizes ore recovery while minimizing the risk of geotechnical failure [45], [135]. A stope layout optimization algorithm should, therefore, include constraints to place rib pillars as support between stopes that are horizontally contiguous and sill pillar between vertical stopes [136].

Equations (3.14 – 3.17) are constraints that ensure the geotechnical requirement of pillars is enforced around the stopes. Additional decision variables, w_{ij}^1 and w_{ij}^2, are introduced that controls the corner blocks of each pillar. That is, these variables become 1 if block (i, j) is the corner block of a pillar. Equations (3.14) and (3.15) controls pillar size (minimum and minimum pillar dimensions in vertical and horizontal directions, respectively, γ_1 and γ_2). Equation (3.16) and (3.17) ensures $w_{ij}^1 = 1$ or $w_{ij}^2 = 1$ if block (i, j) is the leftmost or topmost block is a pillar.

$$\sum_{v=\max(i-\gamma_1+1,1)}^{i} w_{vj}^1 \leq 1 - \sum_{k=1}^{K} x_{ijk} \qquad \forall i,j \tag{3.14}$$

$$\sum_{\mu=\max(j-\gamma_2+1,1)}^{j} w_{i\mu}^2 \leq 1 - \sum_{k=1}^{K} x_{ijk} \qquad \forall i,j \tag{3.15}$$

$$\sum_{k=1}^{K} x_{ijk} - \sum_{k=1}^{K} x_{(i-1)jk} = \sum_{k=1}^{K} z_{ijk}^1 - w_{ij}^1 \quad \forall i,j \tag{3.16}$$

$$\sum_{k=1}^{K} x_{ijk} - \sum_{k=1}^{K} x_{i(j-1)k} = \sum_{k=1}^{K} z_{ijk}^2 - w_{ij}^2 \quad \forall i, j \tag{3.17}$$

Figure 3.6 illustrates the types of pillars generated around the stopes in the layout using these constraints. The pillars are respected around the stopes formed and cause the stope shapes to also respect those pillar blocks. Note that, because the pillar constraints are defined along the directions of the vertical and horizontal directions, pillar widths are not maintained in the diagonal direction. This is a limitation of the formulation, and it is further discussed in Section 4.4.

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$w_{ij}^2=1$	0
0	0	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$w_{ij}^2=1$	0	0	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$w_{ij}^2=1$	0
0	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$w_{ij}^2=1$	0	0	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$w_{ij}^2=1$	0
0	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$w_{ij}^2=1$	0	0	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$w_{ij}^1=1$ $w_{ij}^2=1$	0	0
0	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$w_{ij}^2=1$	0	0	$w_{ij}^1=1$	$w_{ij}^1=1$	$w_{ij}^1=1$	$w_{ij}^1=1$	0	0	0
0	$w_{ij}^1=1$	$w_{ij}^1=1$	$w_{ij}^1=1$	$w_{ij}^1=1$	$w_{ij}^1=1$	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$w_{ij}^2=1$	0	0	0	0
0	0	0	0	0	0	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$w_{ij}^2=1$	0	0	0	0
0	0	0	0	0	0	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$w_{ij}^2=1$	0	0	0	0
0	0	0	0	0	0	$w_{ij}^1=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$x_{ijk}=1$	$w_{ij}^2=1$	0	0	0	0
	0	0	0	0	0	0	$w_{ij}^1=1$	$w_{ij}^1=1$	$w_{ij}^1=1$	$w_{ij}^1=1$	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 3.6 Geotechnical Rib & Sill Pillars in Stope Layout

3.7. BILP MODEL IMPLEMENTATION

The model presented in this section is implemented in MATLAB. The MATLAB code calls Gurobi to solve the optimization problem through Gurobi's MATLAB API. A custom application (App) called SSLO.mlapp was developed in MATLAB using the MATLAB's App Designer Toolkit [137].

The SSLO.mlapp has two (2) upload buttons which enables user to upload the primary input data (BILP_Grade, BILP_EBV) into the algorithm. Users may upload pre-processed Grade & EBV data in comma delimited (CSV) file format or a MATLAB file format or upload the grade data and enter the economic data and allow the algorithm to generate the EBVs of the blocks. The App interface has four (4) panels. Three (3) of the panels (i.e., Block Counter, EBV Generator and Stope Configuration panels) accept user input to configure the BILP algorithm, while the last panel (SSLO Results panel) is for displaying the results of the optimization. A bulb in the lower left corner of the app indicates the status of any ongoing activity. It turns blue when all the input data are correctly keyed by the user and after the optimization completes. A red lamp indicates an error during the input process as well as the optimization run. Figure 3.7 shows the SSLO.mlapp user interface used to configure and run the BILP model for solving the SLO problem.

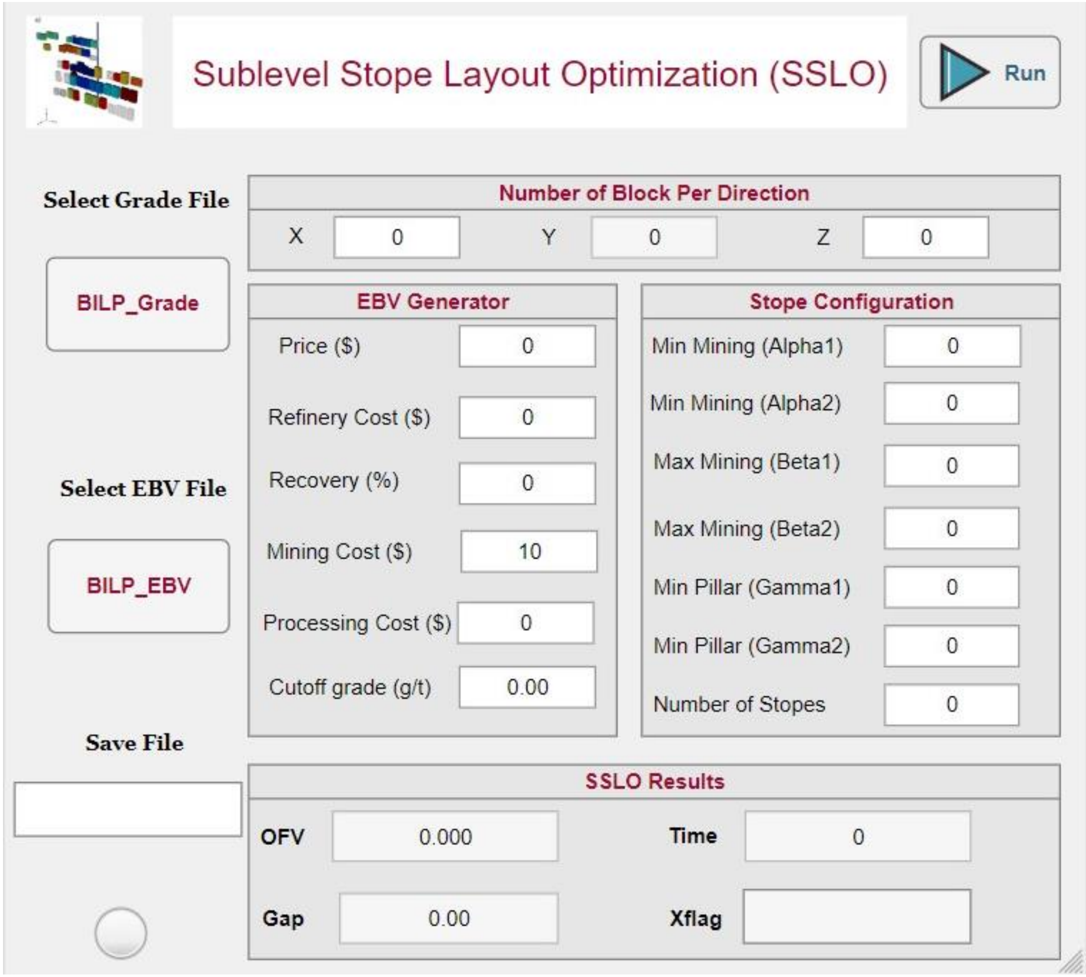

Figure 3.7 The User Interface of the SSLO Application

3.7.1. BILP Model Solution Process. As stated above the BILP Model is the brain behind the SSLO.mlapp. Figure 3.8 illustrates the process for solving a 2D stope layout optimization problem using the algorithm developed in this research.

3.7.2. BILP Model Verification. The formulated BILP model is verified with an experimental gold deposit dataset. Maptek Vulcan software is used to reblock the original resource block model into a regularized 15m x 15m x 30m model to avoid variable block sizes in the resource model. Figure 3.2 and Figure 3.9 shows a section through the original resource model and the regularized resource model respectively. The orebody for this deposit is irregularly shaped occurring between a depth of -560m to -1,100m below mean sea level with a total mineral resource of 2,138,400 t at an average gold grade of 2.62 g/t. Table 3.5 shows summary statistics of the resource models. This base case example used to verify the algorithm includes 43 blocks in the X-direction and 18 blocks in the Z-direction. We specify 20 stopes in the problem (number of specified stopes should always be higher than what the engineer expects). Consequently, the problem results in 140,400 binary decision variables and 1,479,800 constraints.

The algorithm for the BILP model (the engine behind the SSLO.mlapp) was implemented in MATLAB 2022b [137] environment. When the user clicks run in the SSLO.mlapp, the app passes the user configuration to prepare the model in MATLAB (i.e., builds the model in MATLAB using MATLAB's problem-based optimization workflow commands [138]). MATLAB then calls GUROBI OPTIMIZER (Gurobi Optimizer version 10.0.0 build v10.0.0rc2) [120] through its MATLAB API to solve the optimization problem at a gap tolerance of 0.0%. The base case example is run on a Dell Precision T5610 computer with an Intel(R) Xeon(R) Gold 5220R CPU @ 2.20GHz (8 CPUs) with a 32 GB RAM.

3.7.2.1. Block economic model. Table 3.6 shows the mining, processing, and economic data used to convert the reblocked model into the economic block model—the main input for the SSLO.mlapp [52].

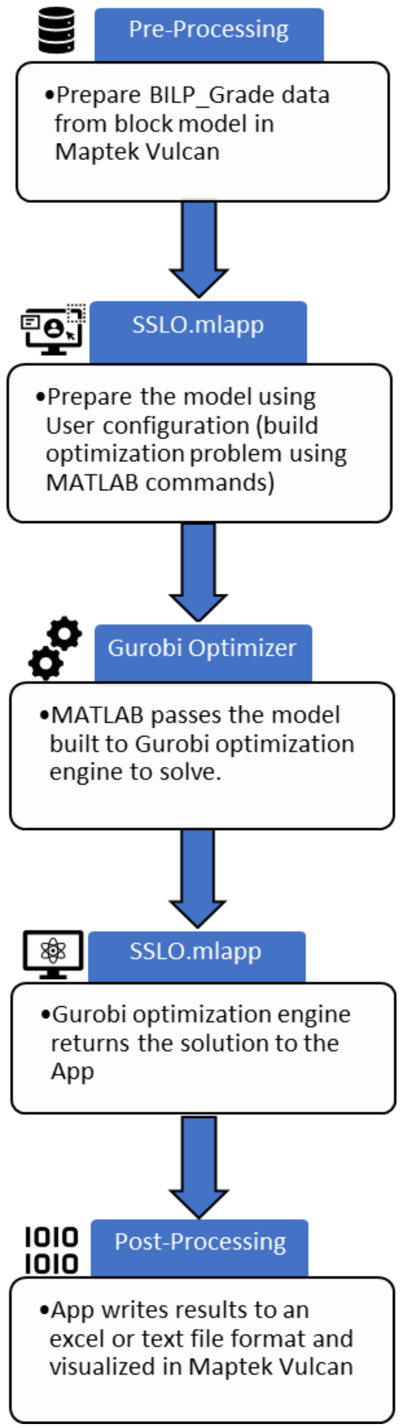

Figure 3.8 Schematic Flowchart of BILP Model Process

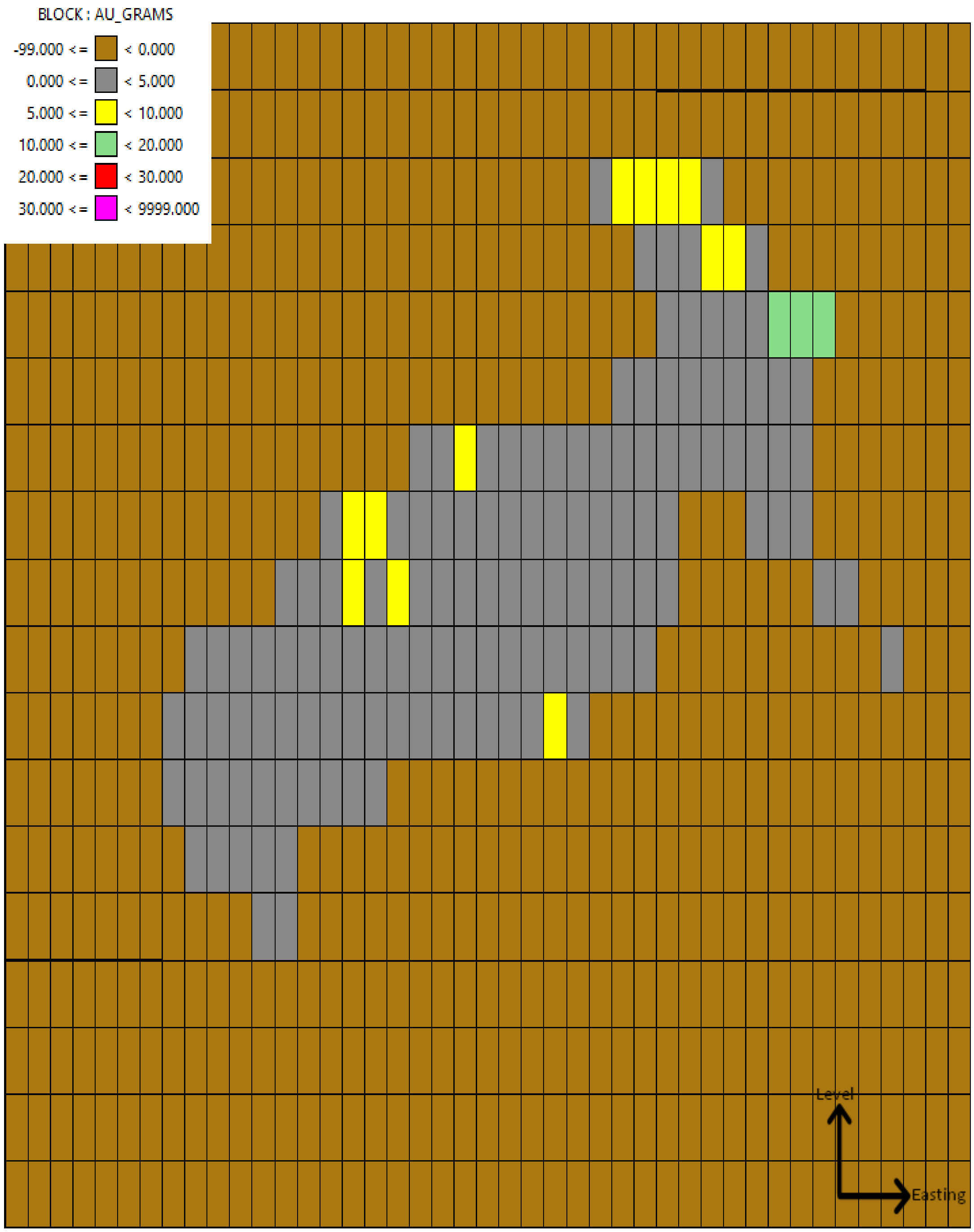

Figure 3.9 The Reblocked Resource Model

Table 3.5 Summary Statistics of Block Models

Attribute	Resource Model	Reblocked Model
Metal	Au	Au
Number of blocks	2132	774
Blocks Au > 0	1344	144
Total mineralized material (Mt)	2,138,400.00 t	2,138,400.00 t
Maximum Au value (g/t)	37.563g/t	14.91g/t
Minimum Au value (g/t)	0.006g/t	0.145g/t
Average Au value (g/t)	2.61 g/t	2.62 g/t
Density	2.2kg/m^3	2.2 kg/m^3
Variance	5.76 %^{2}	4.35 %^{2}
Standard deviation (%)	2.40%	2.09%
Block size	Varied	15m ×15m ×30m
Depth from surface	560m – 1,100m	560m – 1,100m

Table 3.6 List of Economic and Technical Parameters

Parameter	Definition
Price of metal	$1,500/oz
Cost of selling the metal	$5/oz
Cost of mining a tonne of rock	$15/t
Cost of processing a tonne of rock	$10/t
Processing recovery of metal	90%
Tonnage of block$_{ij}$	6,750t
Cutoff grade	1.5g/t

3.7.2.2. Stope design input. Table 3.7 shows the stope design input data used for this numerical example. These are also the main user inputs for the stope configuration panel on the SSLO.mlapp as seen in Figure 3.10, which has the completed configuration for the application to run.

Table 3.7 Parameters for Basecase Scenario

BILP Configuration	Scenario
Minimum Mining Height α_1	2
Minimum Mining Width α_2	3
Maximum Mining Height β_1	3
Maximum Mining Width β_2	4
Minimum Pillar Length γ_1	2
Minimum Pillar Length γ_2	2
Number Of Stopes k	20
Cutoff Grade G_{off}	1.5g/t

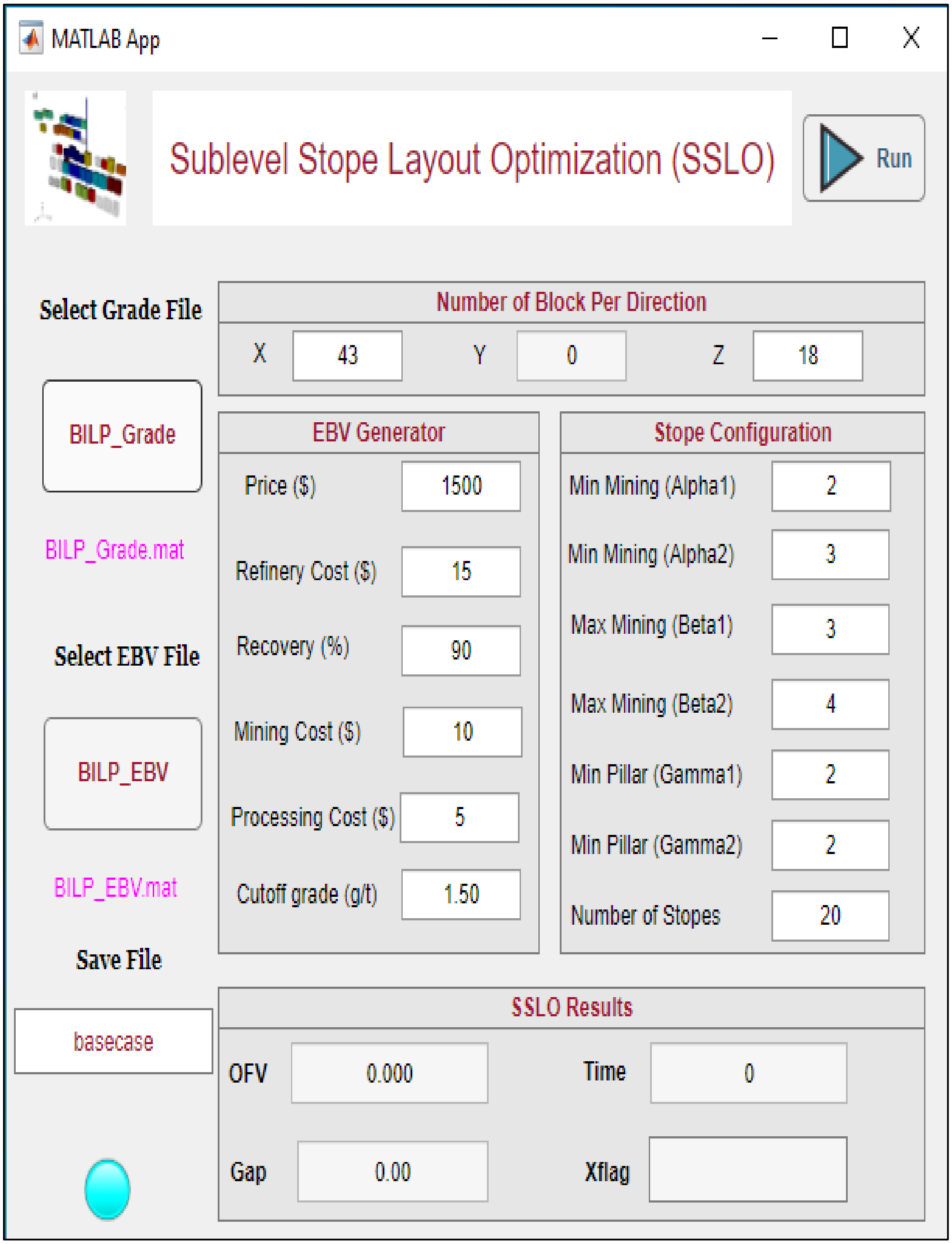

Figure 3.10 SSLO.mlapp App Configuration

3.7.2.3. Results and discussions. Figure 3.11 presents the optimal stope layout obtained by solving the BILP model for the underground deposit using Gurobi optimization engine. Results from the test scenario summarized in Table 3.8 indicates that the model found an optimal solution within an optimality gap tolerance of 0.0% in approximately 1.835 hrs. The model combined 60 ore blocks out of the 144 ore blocks and 13 waste blocks into eight (8) stopes from the 20 stopes. This translated into an undiscounted value of $ 34.37 million with 1,084,050 t of total mineralized material at an average grade of 3.40 g/t. The eight (8) stopes in the final layout of the test scenario, satisfies the operational, technical, and economic requirements. The stopes have variable heights and lengths, indicating the power of the BILP model to adapt the stope shapes to the geological and geotechnical characteristic of the deposit for maximum recovery. The BILP model also selected 13 waste blocks as part of the layout to ensure it conforms with the efficient shape constraints. All the stopes created, had overall stope grades above the cutoff ($G_{off} = 1.5g/t$). As seen in Figure 3.12, the model generated stopes to the right side of the deposit which has the best grades. Majority of the stopes are found in the central portion of the deposit which has medium grades. The left side contains the low grades in the deposit hence a few stopes were created in that zone. This indicates the power of the BILP model to pick out blocks that maximizes the overall profit of the deposit. The scenario also verified that the allowable geotechnical requirement (pillars) was enforced. Figure 3.13 shows the results on the SSLO.mlapp.

Table 3.8 Optimization Results for Basecase Scenario

Parameter	**Scenario**
Number Of Stopes	8
Number Of Mined Blocks	73
Number Of Pillar Blocks	65
Objective Function Value ($)	34,373,085.19
Solution Time (Hrs.)	1.83
Gap Tolerance (%)	0.0
Minimum Stope Grade (g/t)	1.71
Maximum Stope Grade (g/t)	5.23
Average Layout Grade (g/t)	3.40
Total Layout Tonnage (tonnes)	1,084,050

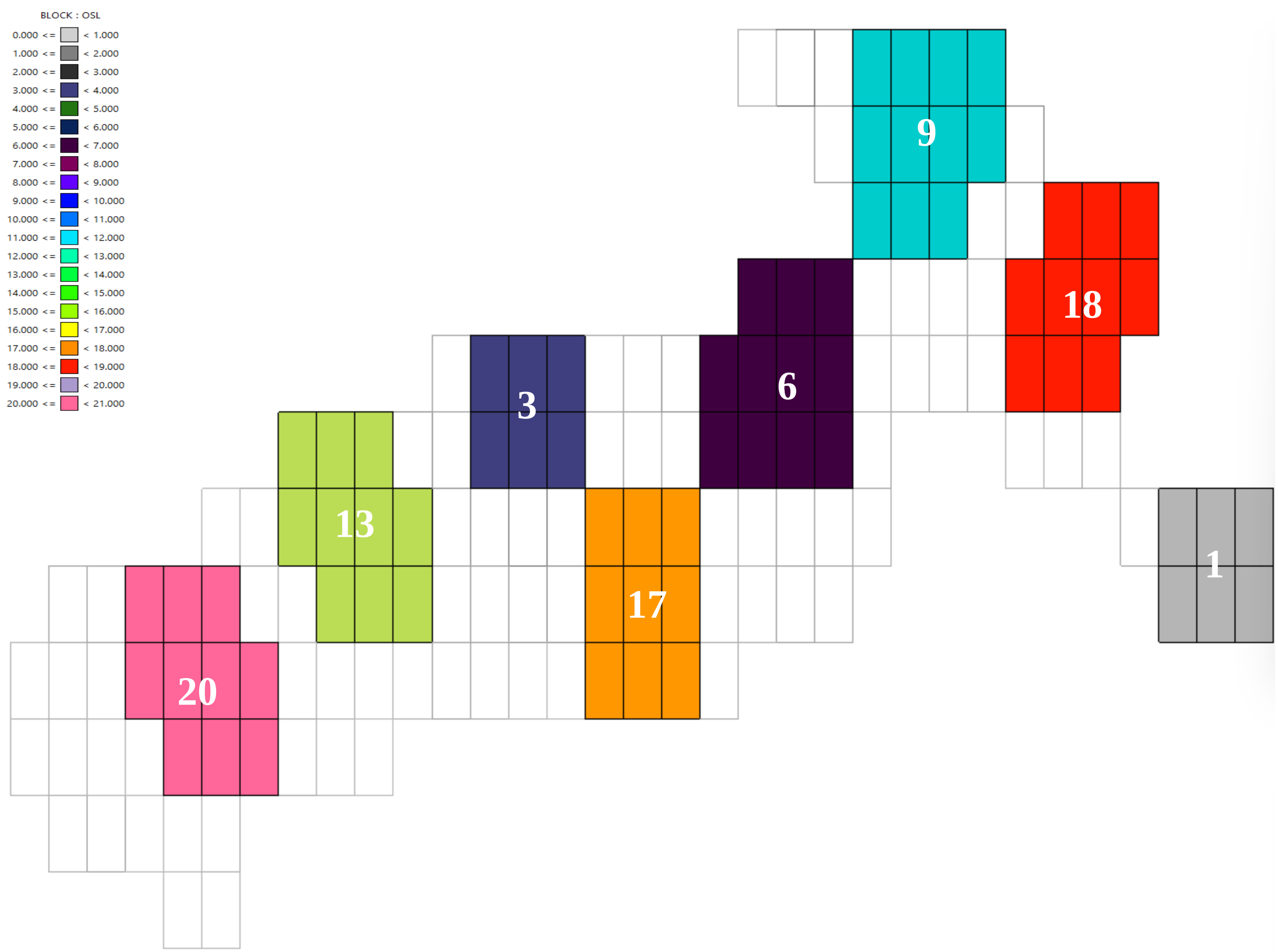

Figure 3.11 Optimal Stope Layout for Basecase Scenario

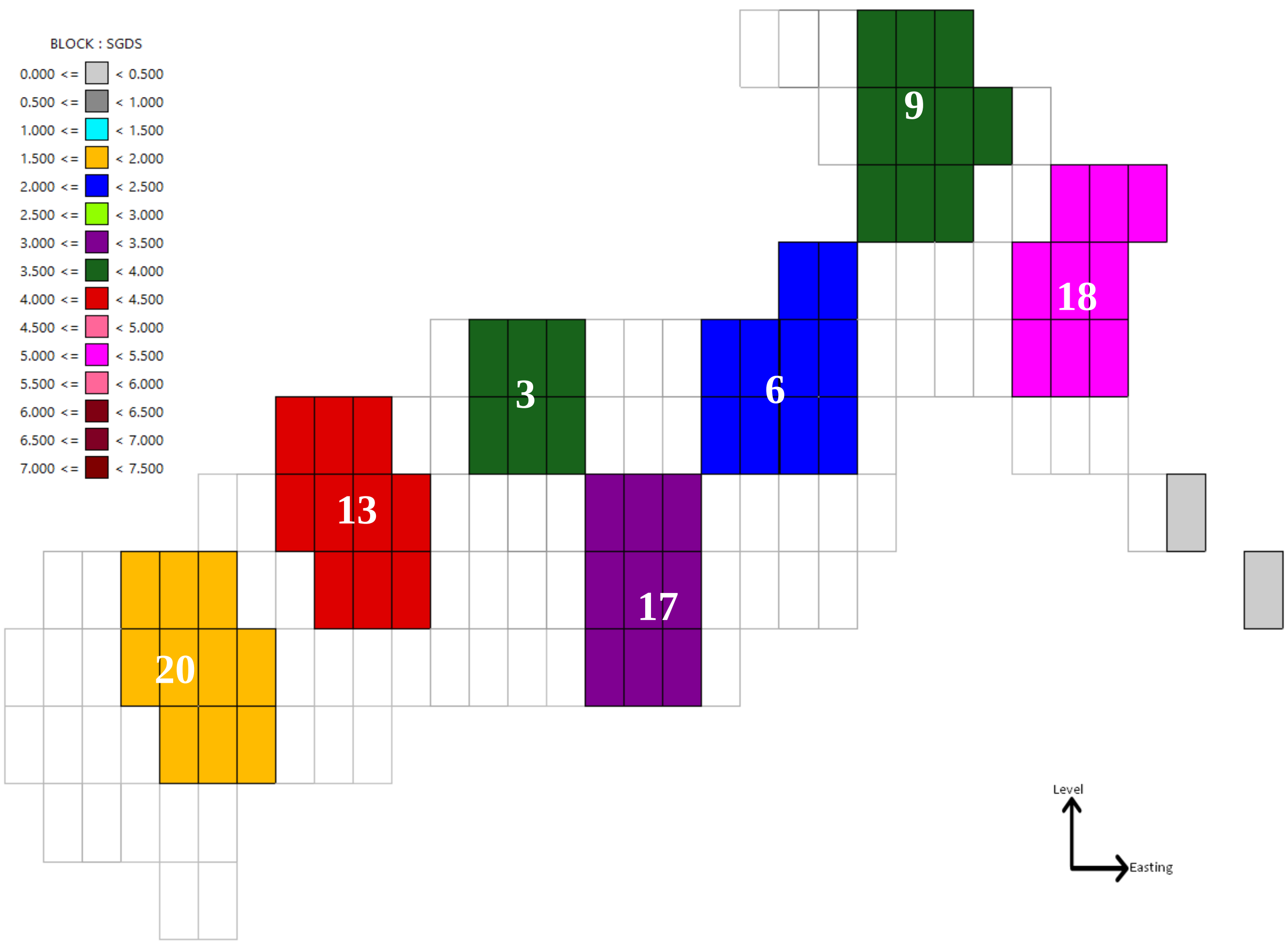

Figure 3.12 Optimal Stope Layout for Basecase Scenario (Showing Stope Grades)

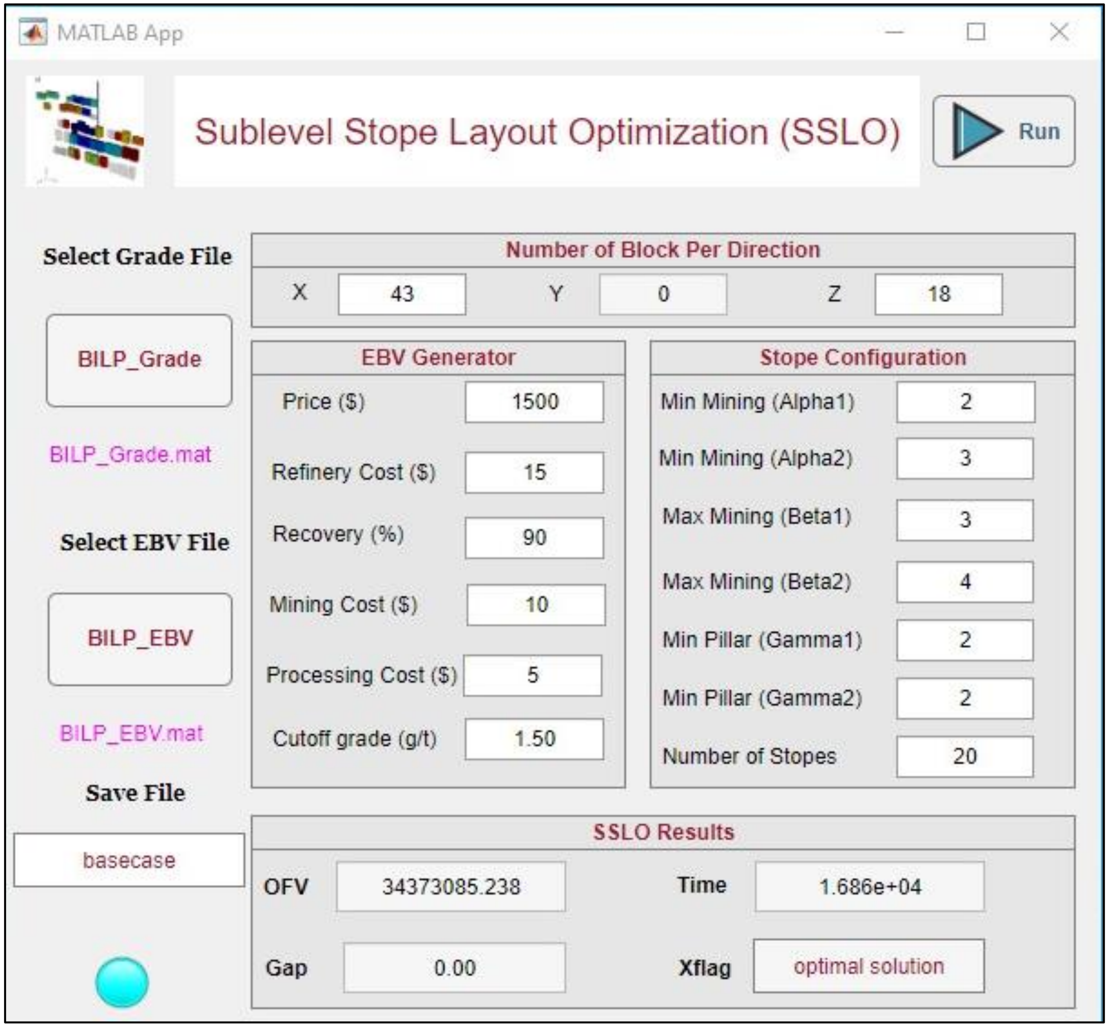

Figure 3.13 The SSLO.mlapp App Showing Basecase Results

3.8. SUMMARY

The model presented here proposes a method for optimizing the economic value of a sublevel stope layout based on a binary integer linear programming formulation. A case study of an underground gold mine has been used to successfully verify the BILP model. A MATLAB Application, code name SSLO.mlapp, has successfully been created through this research. The App together with the BILP model have been utilized to facilitate solving the SLO problem. The App has been successfully validated for the basecase of the case study. Based on results of the basecase, the following conclusions can be made:

- The results from the basecase study highlights the possibility to model shapes in LP-based techniques for the stope layout problem. Unlike most LP-based stope layout approaches, the proposed model accounts for efficient shape constraints in the geometric constraints.
- The model finds the optimal stope layout that maximizes the undiscounted profit for the deposit within a gap tolerance of 0.00%.
- The model allows the generation of variable stope length and height as well as incorporating geotechnical pillar requirements between the selected stopes. Thus, the model permits mining operation to follow irregular mineral deposit peripheries to minimize dilution.
- The BILP model has some limitations as seen from the optimal layout in Figure 3.11. Pillars are respected around the stopes however pillar widths are not maintained in the diagonal direction because the pillar constraints are defined along the vertical and horizontal directions.

- Lastly the use of binary variables makes the problem difficult to solve because it creates a combinatorial explosion of possible solutions as the number of variables increases, leading to long solution times.

4. EVALUATING THE MODELS SENSITIVITY TO INPUT PARAMETERS AND OPTIMIZATION PROBLEM SIZE

4.1. OVERVIEW

This section of the thesis describes computational experiments carried out to assess the performance and sensitivity of the proposed BILP model's solution to key input parameters. The key parameters of the model such as the stope dimensions, stope cutoff grade, the number of predefined stopes, the pillar dimensions as well as the size of the optimization problem were varied to evaluate their impact on the solution time, the objective function value, the optimality gap, and the final stope layout.

This study looks at the instance of changing the scale of the optimization problem since an increase in problem size leads to an explosion in variables and, from a practical standpoint, strategic mine engineers are unlikely to adopt an algorithm that takes more than a few minutes to converge to a solution. The work also examined the impact of different configurations of the stope and pillar dimensions because one of the strengths of this model is that it allows engineers to generate any rectangular mining dimension that mimics the deposit's peripheries.

Stope cutoff grade is a key factor that determines the quality and quantity of material that can be sent from the UG to the processing plant. Therefore, evaluating and understanding the impact of this parameter will help engineers to select an optimal cutoff grade that meets the processing plant requirement and overall project profitability. The number of stopes is set by the engineer as an input to the algorithm. This parameter determines the stopes that are formed in the final layout as such the author studies the effect of changing the number of stopes on solution time and overall profitability of the layout formed.

The subsections below present the computational experiments for the different scenarios run to evaluate the performance and sensitivity of the model. For each experiment, the author generated economic block values and block grades in the same way as for the base case in Section 3.7.2. The author used the same MATLAB code used in the base case study in Section 3.7.1 to solve all the different experimental runs. The full input data used for these experiments are available online in this GitHub repository (https://github.com/TheoMensah/BILP_SSLO).

4.2. EFFECT OF STOPE CUTOFF GRADE

Due to the highly selective nature of underground mining, all material hauled from the stopes generated must be at or above the predetermined cutoff grade[1]. This is to ensure maximum recovery of the deposit.

4.2.1. Input Data for Stope Cutoff Grade Evaluation. A low cutoff grade results in more ore tonnage, and overall higher metal output, but at the expense of additional capital cost while a high cutoff grade, denotes a short life of mine and lower overall metal output, which in most instance cannot be sufficient to justify the capital cost of establishing a mine [139]. Thus, the cutoff grade selected as an input for this model must but optimal. Techniques like MIP [140], [141] can be used to find the optimal cutoff grade for use in the BILP model.

[1] Cutoff grade is the grade value below which blocks in the deposit are uneconomical to mine.

Table 4.1 and Table 4.2 show the input data used to configure the BILP model for changing the cutoff grade. Three (3) scenarios were run for this experiment to investigate the effect of a lower and higher cutoff grade value, than the base case value. For this analysis, the stope configurations for each scenario for mining and geotechnical dimensions are selected to be the same as shown in Table 4.1 while Table 4.2 summarizes different cutoff grade values used for this experiment.

Table 4.1 BILP Input Data – Experiment 1

BILP Configuration	**Scenario**
Minimum Mining Height α_1	3
Minimum Mining Width α_2	3
Maximum Mining Height β_1	4
Maximum Mining Width β_2	4
Minimum Pillar Length γ_1	2
Minimum Pillar Length γ_2	2
Number Of Stopes k	20

Table 4.2 Experiment 1 – Cutoff Grades

BILP Configuration	**Scenario 1**	**Scenario 2**	**Scenario 3**
Cutoff Grade G_{off}	1.5g/t	2.5g/t	3.5g/t

4.2.2. Results and Discussion. Figure 4.1 – Figure 4.3 show the results of this experiment while Table 4.3 summarizes the results. The results show that the objective function value decreases as the cutoff grade increases from \$33M to \$28M. This results also show a higher cutoff grade causes the algorithm to converge faster to a solution. Scenario 3 recorded the fastest solution time of 0.11 hrs. All scenarios achieved an optimal solution[2].

From the results the lower the cutoff grade selected, the lower the average overall layout grade achieved however this leads to more blocks being selected and more stopes being formed in the optimal layout and consequently maximizing the economic value. The results also suggest that increasing cutoff grade does not improve the overall profitability of the designed layout although it leads to higher grade stopes.

The solutions show that five (5) stopes were formed in the optimal layout for Scenario 1 while four (4) stopes were formed in the optimal layout for Scenarios 2 and 3, respectively (Figure 4.1 – Figure 4.3). Each stope generated in the optimal layout had an average grade above the cutoff grade. The various geological domains of the blockmodel used have been described earlier in Section 3.7.2. Based on this understanding, Figure 4.1 – Figure 4.3 show the algorithm produced stopes aimed at the blocks in the central and upper right zones in each Scenario indicating the BILP models' power to target, select and combine blocks that maximize the value of a deposit.

[2] Optimal solution means a solution with an optimality gap of 0.00%

Table 4.3 Experiment 1- Results of Changing Cutoff Grade

Parameter	Units	Optimization Results		
		Scenario 1	Scenario 2	Scenario 3
Objective Function Value	($)	33,727,574.0	30,959,129.3	28,468,130.7
Solution Time	(hrs)	1.03	0.67	0.11
Gap Tolerance	(%)	0.00	0.00	0.00
Number of Stopes	(#)	5	4	4
Ore Blocks Mined	(#)	70	57	46
Waste Blocks Mined	(#)	5	3	2
Number of mined blocks	(#)	75	60	48
Number of pillar blocks	(#)	65	60	55
Minimum Stope Grade	(g/t)	2.03	2.99	3.53
Maximum Stope Grade	(g/t)	4.03	4.03	4.03
Average Layout Grade	(g/t)	3.08	3.41	3.79
Total Layout Tonnage	(tonnes)	1,113,750	891,000	712,800

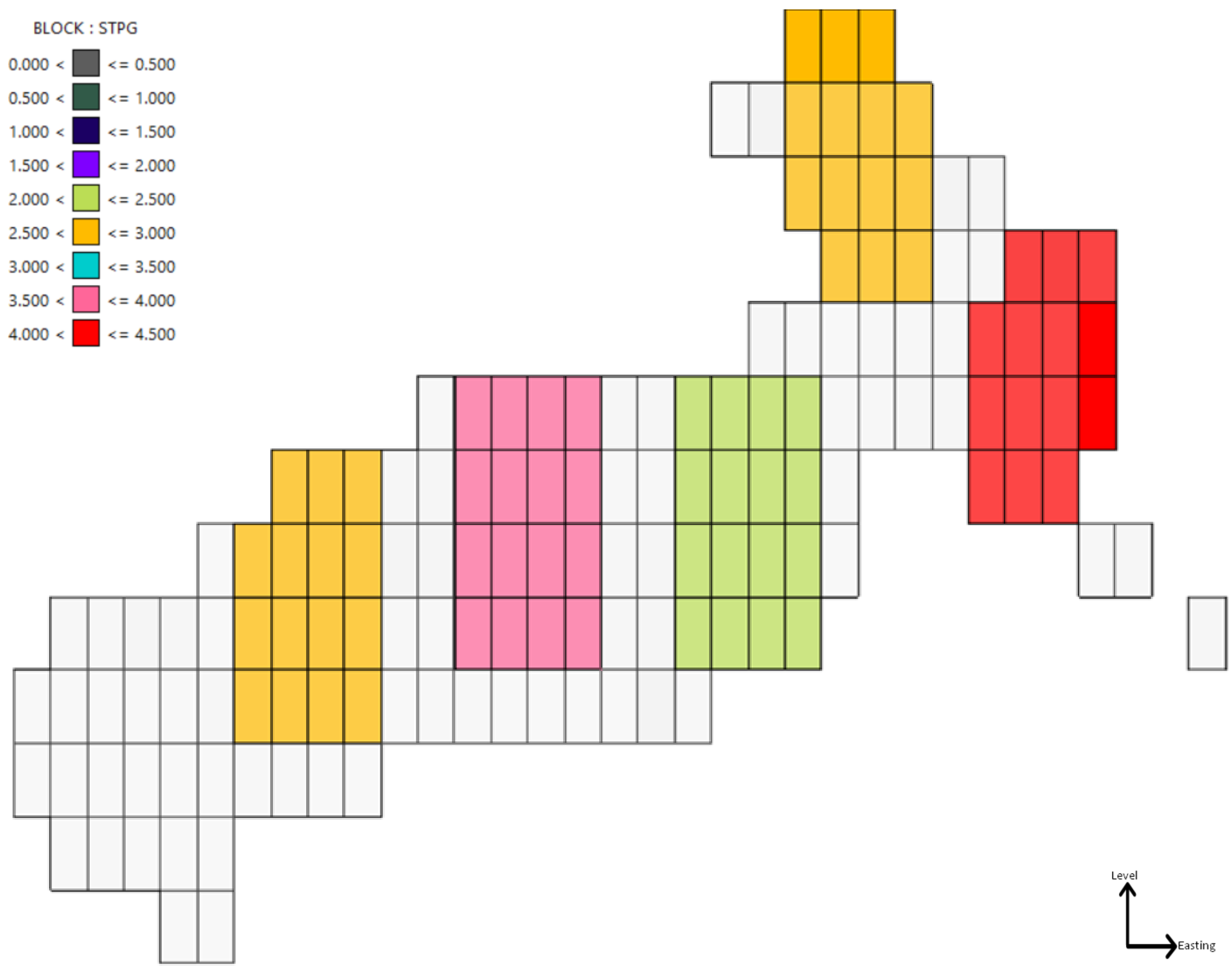

Figure 4.1 Experiment 1 – Optimal Layout Scenario 1

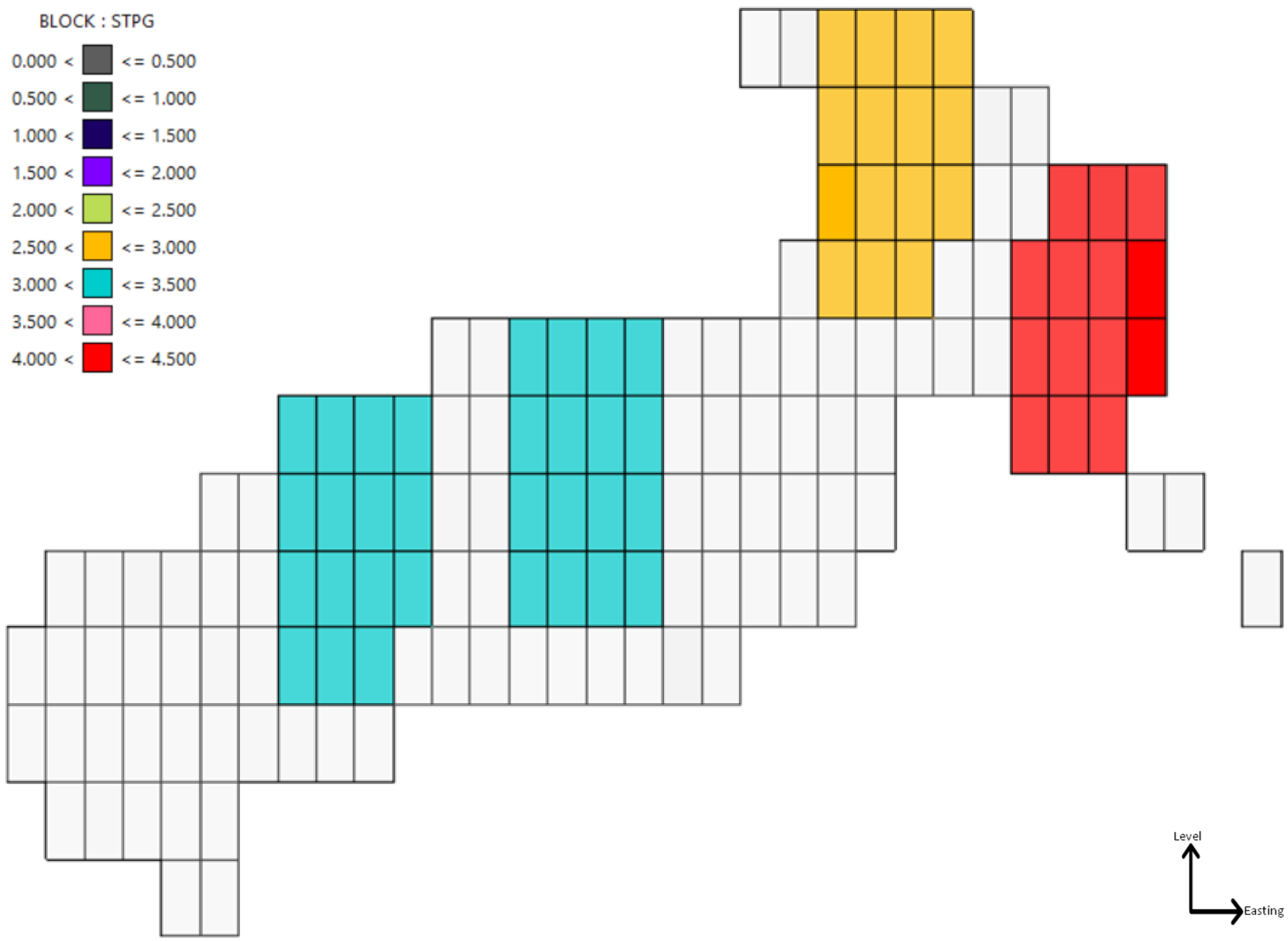

Figure 4.2 Experiment 1 – Optimal Layout Scenario 2

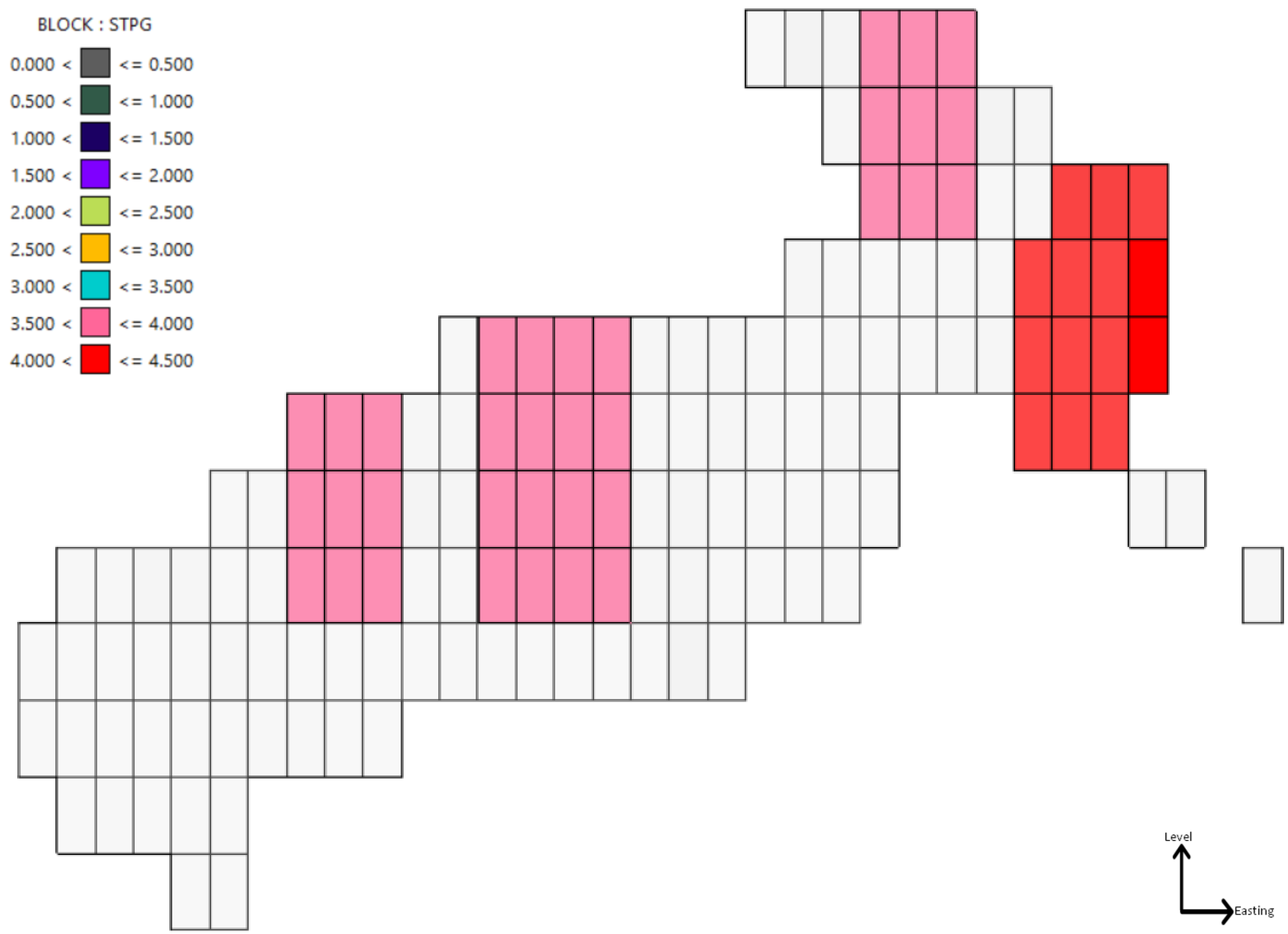

Figure 4.3 Experiment 1 – Optimal Layout Scenario 3

From Figure 4.1, the optimal layout for Scenario 1 (which specified a stope cutoff grade of 1.5 g/t) had three (3) stopes generated within the low-grade domain of $2\text{g/t} < \text{g}_{ij} \leq 3\text{g/t}$, one (1) stope within the medium-grade domain of $3\text{g/t} < \text{g}_{ij} \leq 4\text{g/t}$ and one (1) stope in the high-grade domain $4\text{g/t} < \text{g}_{ij} \leq 10\text{g/t}$.

All stopes formed had at least 14 blocks selected and a maximum of 16 blocks satisfying the mining requirements configured for this scenario. Since the cutoff grade is low, the algorithm takes a longer time to optimize the numerous possible block combinations thus leading to a longer solution time to find the optimal solution.

The optimal layout for Scenario 2 (which specified a stope cutoff grade of 2.5 g/t), shown in Figure 4.2, had one (1) stope generated within the low-grade domain of $2\text{g/t} < \text{g}_{ij} \leq 3\text{g/t}$, two (2) stopes within the medium-grade domain of $3\text{g/t} < \text{g}_{ij} \leq 4\text{g/t}$ and one (1) stope in the high-grade domain $4\text{g/t} < \text{g}_{ij} \leq 10\text{g/t}$. This layout was generated because of the higher stope cutoff grade. All stopes formed had at least 15 blocks selected and a maximum of 16 blocks satisfying the mining requirements configured for this scenario.

Figure 4.3 shows the optimal layout for the highest stope cutoff grade ($G_{off} = 3.5\text{g/t}$) of the three scenarios. The optimal layout had three (3) stopes generated within the medium-grade domain of $3\text{g/t} < \text{g}_{ij} \leq 4\text{g/t}$ and one (1) stope in the high-grade domain $4\text{g/t} < \text{g}_{ij} \leq 10\text{g/t}$. No stopes were generated in the low-grade domain. By further elevating the cutoff grade in this scenario, the algorithm went in to select the few blocks with very high grades in the blockmodel that when combined will achieve the elevated cutoff grade

($G_{off} = 3.5\text{g/t}$) as seen in Figure 4.3. Elevating the cutoff grade also resulted in the fastest solution time since few blocks could meet this requirement. Thus, two (2) stopes formed had 9 blocks selected and the other stopes had a maximum of 16 blocks satisfying the mining requirements configured for this scenario.

4.3. EFFECT OF STOPE DIMENSIONS (ALLOWABLE MINING DIMENSIONS)

Each designed stope must meet an allowable minimum and maximum mining dimension based on the geomechanical properties of the deposit and equipment sizes for practical extraction of material from the stopes. The experiment was designed to investigate the impact of frame sizes while keeping pillar sizes constant.

4.3.1. Input Data for Stope Dimension Evaluation. To ensure that stopes formed follow the deposit peripheries and minimize dilution while ensuring stability and operability, mine engineers using stope optimization algorithms should be able to control the generated stope shapes. This is one strength of the BILP model proposed in this work as it introduces efficient shape constraints as described in Section 3.6.2.1 that allows for any rectangular dimension. The shape constraints in this model, allows the engineer to control the minimum and maximum blocks for the stope in the Z-X plane or Z-Y plane.

This section investigates the sensitivity of the model to changes in stope dimensions. The author designed the experiment to investigate the impact of frame sizes while keeping pillar sizes constant. The reader should note that, practically, larger frame[3] sizes are likely to go with larger pillar sizes (more information in Section 2.2.2.3).

---

[3] Frame is the rectangular dimension of the stope generated in the layout.

By maintaining uniform pillar dimensions, the focus was solely on understanding how different frame configurations influence the outcomes. This approach allowed for a more precise evaluation of the frame size's individual contribution to the results, without the confounding effect of varying pillar sizes.

The analysis run four (4) scenarios for this experiment. The first scenario runs a smaller stope frame ($2\times3|2\times3$) for α_i and β_i, respectively. In Scenario 2 the author investigates a square stope frame ($3\times3|3\times3$) for α_i and β_i, respectively. The third scenario runs an adjustment to the maximum allowable mining (β_i) dimension of the stope frame in Scenario 2 increasing it to large rectangular ($3\times3|4\times4$) frame to investigate the change in increasing the β_i for the frame and the last Scenario 4, evaluates the effect of configuring an even larger stope frame ($3\times5|4\times6$). Table 4.4 shows the common input configuration for the BILP model while Table 4.5 shows different stope frame dimensions used for this evaluation.

4.3.2. Results and Discussion. Table 4.6 summarizes the results of this experiment and Figure 4.4 – Figure 4.7 shows the optimal layouts for these scenarios. From the results, changing the allowable mining dimensions impacts the optimal solution significantly. The stope frame configuration the engineer selects affects the objective function value, the solution time, the number of stopes formed, and general stope layout.

From the results, the objective function value improves with an increase in the stope dimensions although the overall layout grade of material in the stopes decreases with this increase in stope dimensions. The reason for the higher objective function values observed with increasing the frame size is the ability to select a greater number of oreblocks within the optimal layout. This selection of additional oreblocks maximizes the economic value of the mining operation.

Table 4.4 BILP Input Data – Experiment 2

BILP Configuration	**Scenario**
Minimum Pillar Length γ_1	2
Minimum Pillar Length γ_2	2
Number Of Stopes k	20
Cutoff Grade G_{off}	2g/t

Table 4.5 Experiment 2 – Input Stope Dimensions

BILP Configuration	**Scenario 1**	**Scenario 2**	**Scenario 3**	**Scenario 4**
Minimum Mining Height α_1	2	3	3	3
Minimum Mining Width α_2	3	3	3	5
Maximum Mining Height β_1	2	3	4	4
Maximum Mining Width β_2	3	3	4	6

Table 4.6 Experiment 2- Results of Changing Stope Dimensions

Parameter	Units	Optimization Results			
		Scenario 1	Scenario 2	Scenario 3	Scenario 4
Objective Function Value	($)	25,581,135.8	27,455,739.1	33,727,578.4	34,740,592.9
Solution Time	(hrs)	4.50	3.37	1.78	1.03
Gap Tolerance	(%)	0.00	0.00	0.00	0.00
Number Of Stopes	(#)	7	5	5	4
Ore Blocks Mined	(#)	38	43	69	69
Waste Blocks Mined	(#)	4	2	6	12
Number Of Mined Blocks	(#)	42	45	75	81
Number Of Pillar Blocks	(#)	55	45	35	25
Min Stope Grade	(g/t)	2.01	2.49	2.03	2.63
Max Stope Grade	(g/t)	7.21	5.52	4.03	3.31
Average Layout Grade	(g/t)	3.85	3.86	3.08	2.98
Total Layout Tonnage	(tonnes)	623,700	668,250	1,113,750	1,202,850

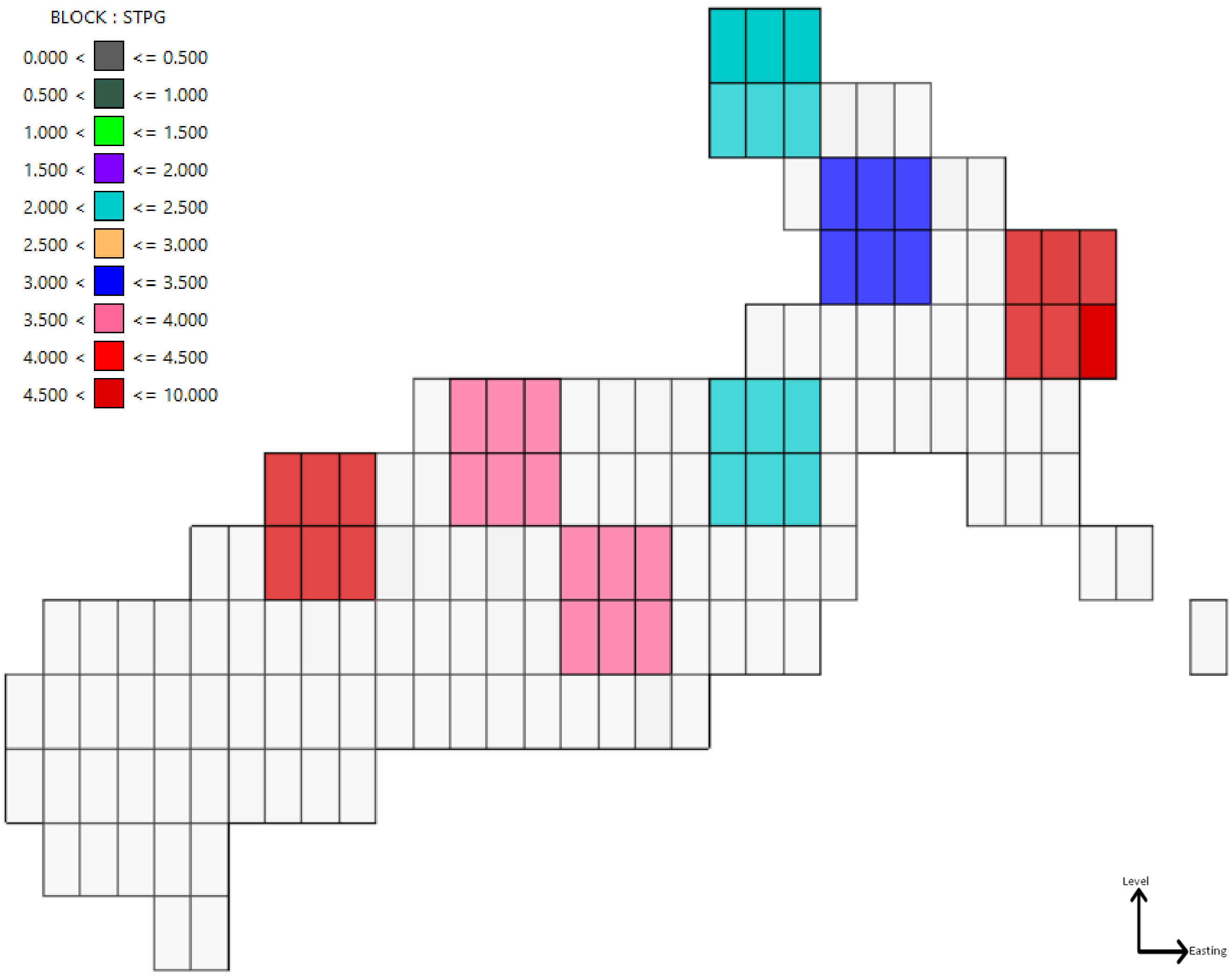

Figure 4.4 Experiment 2 Optimal Layout Scenario 1

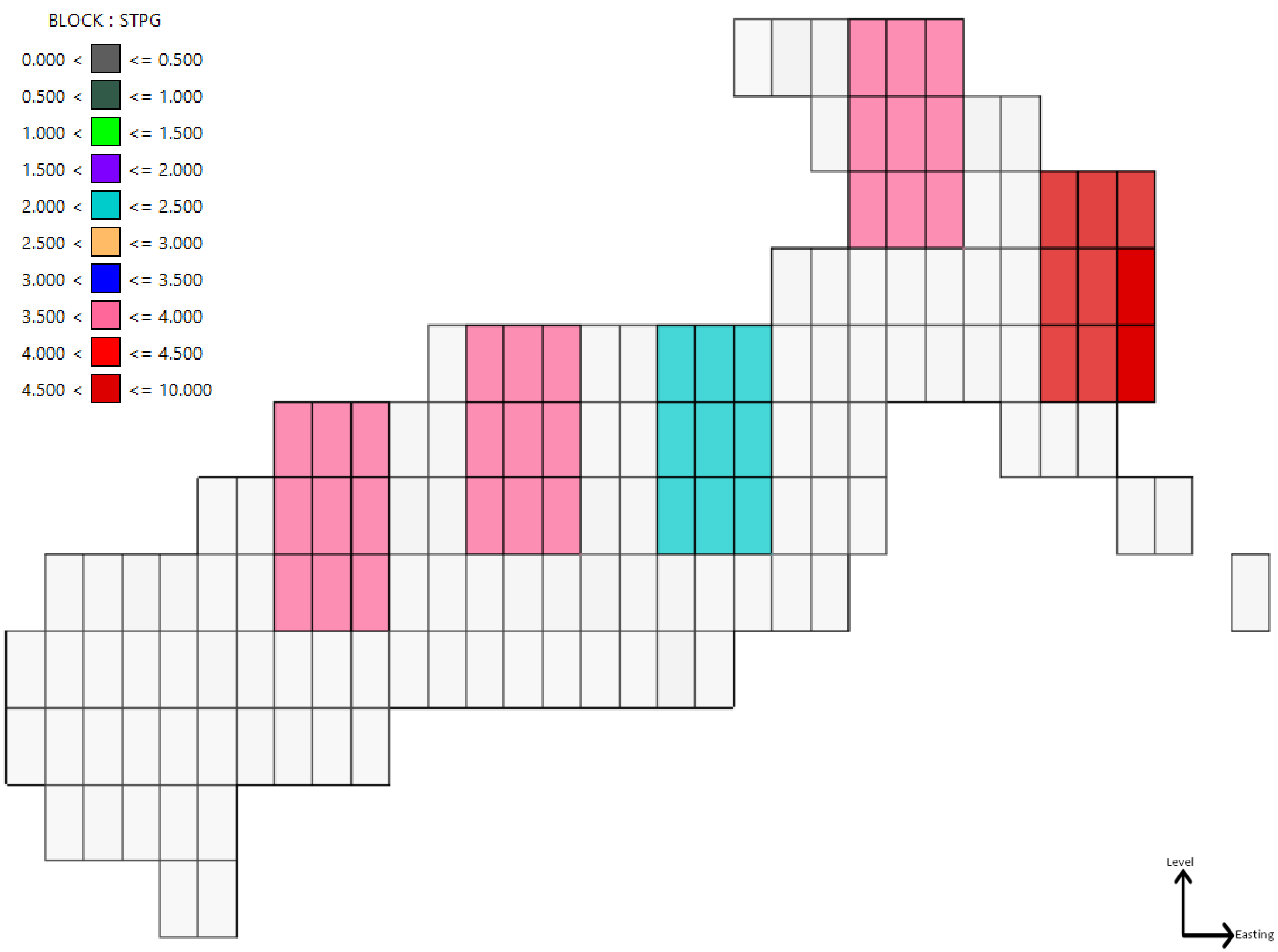

Figure 4.5 Experiment 2 Optimal Layout Scenario 2

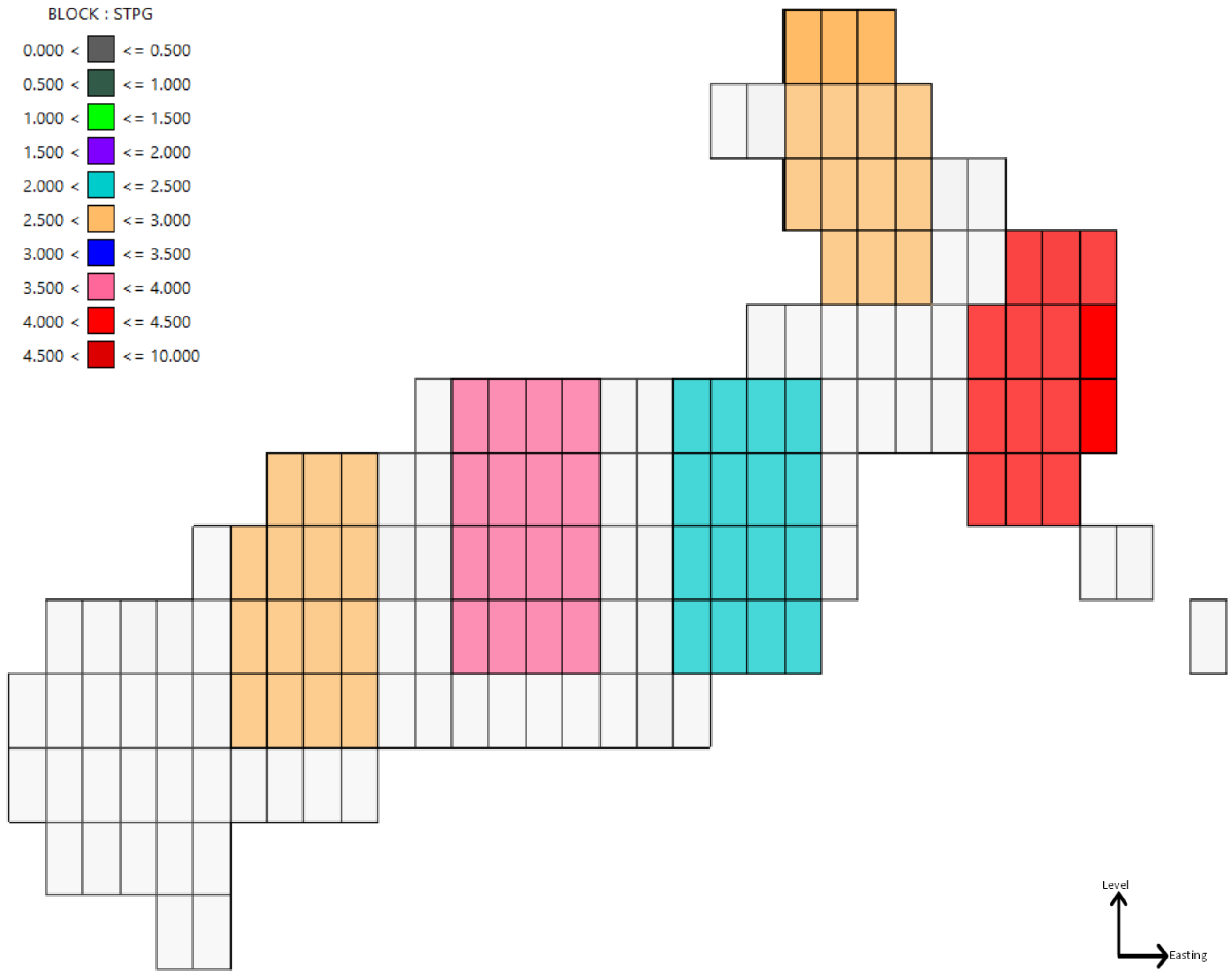

Figure 4.6 Experiment 2 Optimal Layout Scenario 3

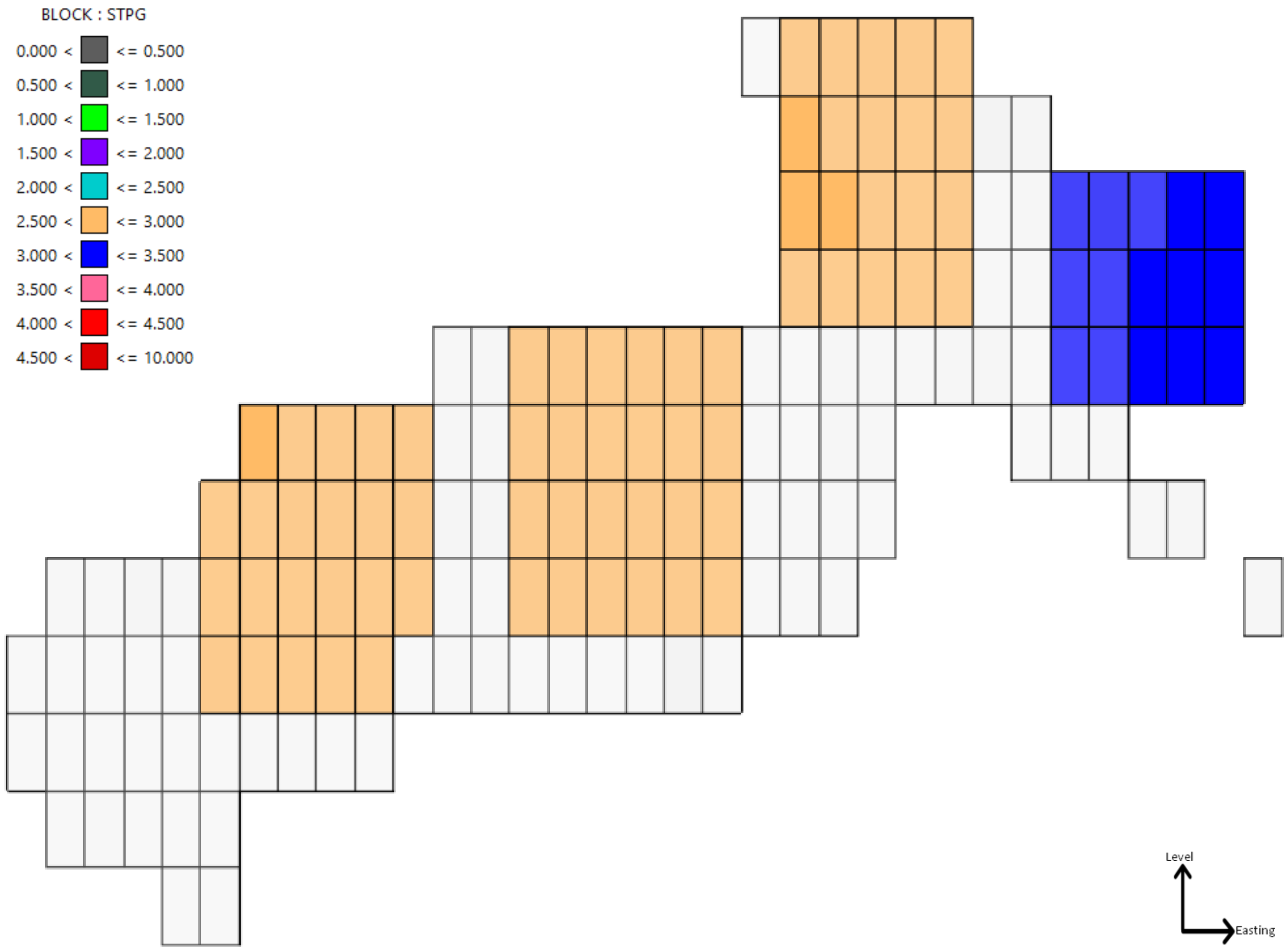

Figure 4.7 Experiment 2 Optimal Layout Scenario 4

It is essential to acknowledge that, in practice, larger frame sizes are often accompanied by larger pillar sizes to ensure stability in the layout and during stoping operations. Consequently, the results of this experiment may not be "optimal" in a practical sense where pillar sizes vary with stope dimensions. However, it is worth noting that, in this experiment, pillar sizes were deliberately kept constant to isolate and assess the sole impact of frame sizes. This approach allowed for a detailed examination of how varying frame sizes alone influence the outcomes, independent of changes in pillar dimensions.

Table 4.6 shows that if a smaller frame configuration is selected, the algorithm takes a longer time to converge to an optimal solution. This extended time is primarily attributed to the increased selectivity achieved with a smaller frame. Therefore, with smaller frame sizes, the options to evaluate increase and the number of constraints also increase, leading to longer computational times.

In each scenario, at least four stopes were created, but the number fell as the stope size increased. All the stopes generated targeted blocks in the deposit's upper right and central zones. The optimal layouts shown in Figure 4.4 – Figure 4.7 produced variable stope heights and lengths that matched the input data in Table 4.5. To respect the shape constraints a few waste blocks are selected as part of the optimal layout. These blocks must be carefully selected as they introduce internal dilution to the generated stopes and reduce the overall value. Thus, one strength of the BILP model is the ability to optimally select waste blocks as part of the optimal stope generated but ensure that the stope cutoff grade and the shape constraints are respected by the solution. The number of waste blocks selected in the final layout also increases with an increase in the stope frame dimensions. The mining dimensions are site specific and varies from mine to mine thus a good geological and geomechanical database of the deposit is recommended when determining the stope dimension to select for optimization.

In Scenario 1, seven (7) stopes were generated in the optimal layout, each conforming to the frame $(2\times3|2\times3)$ configured for the scenario. This solution included 42 blocks; 38 ore blocks from a possible 144 oreblocks and 4 waste blocks[4] to complete the layout. This translated into an economic value of $26 M with total mineralized material of 623 Kt at 3.85 g/t. The small size of this stope frame means more stopes will be formed; however, due to the variability of grades in the deposit and the requirement of pillars, not all blocks can be included in the optimal layout of stopes. Thus, to ensure the economic value is maximized, the algorithm targets blocks with medium to high grades (Figure 4.4).

The pillar constraints are only respected in the X and Z-directions (not along the diagonal directions). Some of these stopes in our solution will be unsafe to mine because they are next to each other diagonally. Stopes 3 – 17 – 6 and stopes 9 – 6, in Figure 4.4, illustrate this issue. Section 3.8 discusses this limitation of the proposed BILP model. One possible solution to this problem is to post-process the solution with heuristics to avoid these situations. Another possible solution will be to model sill pillars into the layout. This will separate the stopes into levels and can potentially eliminate the direct diagonal interaction of some of the stopes generated in the layout.

Figure 4.5 illustrates the optimal layout of stopes for Scenario 2 which has $(3\times3|3\times3)$ stope frames. From the layout in Figure 4.5, given the stope cutoff grade

---

[4] Waste blocks are included as internal dilutions and to ensure stopes respect the shape constraints.

($G_{off} = 2\text{g/t}$) and the square frame, five (5) stopes were formed each with 9 blocks (min/max of 3 blocks in the Z-direction and 3 blocks in the X-direction) conforming to the square frame configuration.

Figure 4.6 illustrates the optimal layout for Scenario 3, which used a ($3\times3|4\times4$) stope frame. As can be seen in Figure 4.6, all the stopes conformed to the minimum mining requirement of 3 blocks in the X direction and 3 blocks in the Z direction.

Two stopes formed with the maximum dimensions demonstrating the strength of the BILP model to generate variable and efficient stope shapes in the optimal layout. From Table 4.6, Scenario 3 recorded an economic value of $34 M by targeting and selecting 75 blocks in total consisting of 69 ore blocks from a possible 144 and six (6) waste blocks to complete the layout. This translates into total mineralized material of 1.1 Mt with average grade of 3.08 g/t in 1.78 hrs. As stated earlier in Section 4.2.2 the lower the grade the more metal and thus more value. Similarly, since there are more blocks to mine, the algorithm is smart enough to go after lower grades that meet the cutoff grade to maximize the value.

Figure 4.7 shows the optimal solution for Scenario 4, which uses a larger ($3\times5|4\times6$) stope frame. Similar results were achieved for this scenario relative to Scenario 3. All the stopes in the optimal layout conformed to the minimum mining requirement of 3 blocks in the Z direction and 5 blocks in the X direction. Also, all the stopes formed had variable stope lengths and all respected the allowable mining dimensions. The stopes generated had 81 blocks in total consisting of 69 ore blocks from a possible 144 and 12 waste blocks to complete the layout. This translates into total mineralized material of 1.2 Mt with average grade of 2.98 g/t in 1.03 hrs.

The effect of adding more waste blocks is offset by the benefit of having more oreblocks in the stopes since the algorithm does not have to leave ore behind in pillars. This makes the stope grade positive. Also, the decision time to include a block or not is reduced since the frame allows consideration of more blocks to be included in a stope.

4.4. EFFECT OF STOPE PILLAR DIMENSIONS (GEOTECHNICAL REQUIREMENT)

In naturally supported underground mining operations, ground control is essential to ensure excavation stability, worker, and equipment safety. To achieve this goal, geomechanical engineers design pillars based on rock mass quality, stress distribution, potential ground instability area as well as host and surrounding rock geological characteristics (more information in Section 2.2.2.3).

4.4.1. Input Data for Pillar Dimensions Evaluation. These pillar dimensions (in number of block units for this algorithm) are then included in the algorithm to generate stopes in the final layout that are safe, operable, and profitable. The author investigates the effect of changing the pillar dimensions and evaluates the impact on the solution. The reader should note that advanced computational models and algorithms are currently employed to determine the most effective arrangement of pillars within the stope layout. Table 4.7 shows the different pillar lengths used in this experiment while Table 4.8 shows the BILP model configuration for the three (3) scenarios in this experiment.

Table 4.7 Experiment 3 – Pillar Dimensions

BILP Configuration	**Scenario 1**	**Scenario 2**	**Scenario 3**
Minimum Pillar Length γ_1	2	3	2
Minimum Pillar Length γ_2	2	3	4

Table 4.8 BILP Input Data – Experiment 3

BILP Configuration	**Scenario**
Minimum Mining Height α_1	3
Minimum Mining Width α_2	3
Maximum Mining Height β_1	3
Maximum Mining Width β_2	3
Number Of Stopes k	15
Cutoff Grade G_{off}	2 g/t

4.4.2. Results and Discussion. Table 4.9 summarizes the results while Figure 4.8 – Figure 4.10 show the optimal stope layout for the scenarios. The results show that the objective function value drops from $27M to a low $23M as the pillar size increases in this experiment. The findings from the study reveal that in the optimal stope layout, larger pillar sizes result in a greater number of blocks being designated as pillars, contributing to enhanced structural stability.

Table 4.9 Experiment 3- Results of Changing Pillar Dimensions

Parameter	Units	Optimization Results		
		Scenario 1	Scenario 2	Scenario 3
Objective Function Value	($)	27,455,739.1	24,472,902.5	23,467,732.5
Solution Time	(hrs)	2.26	1.70	2.36
Gap Tolerance	(%)	0.00	0.00	0.00
Number Of Stopes	(#)	5	5	5
Ore Blocks Mined	(#)	43	39	37
Waste Blocks Mined	(#)	2	6	8
Number Of Mined Blocks	(#)	45	45	45
Number Of Pillar Blocks	(#)	60	75	90
Minimum Stope Grade	(g/t)	2.49	2.15	2.36
Maximum Stope Grade	(g/t)	5.52	5.52	5.52
Average Layout Grade	(g/t)	3.86	3.53	3.42
Total Layout Tonnage	(tonnes)	668,250	668,250	668,250

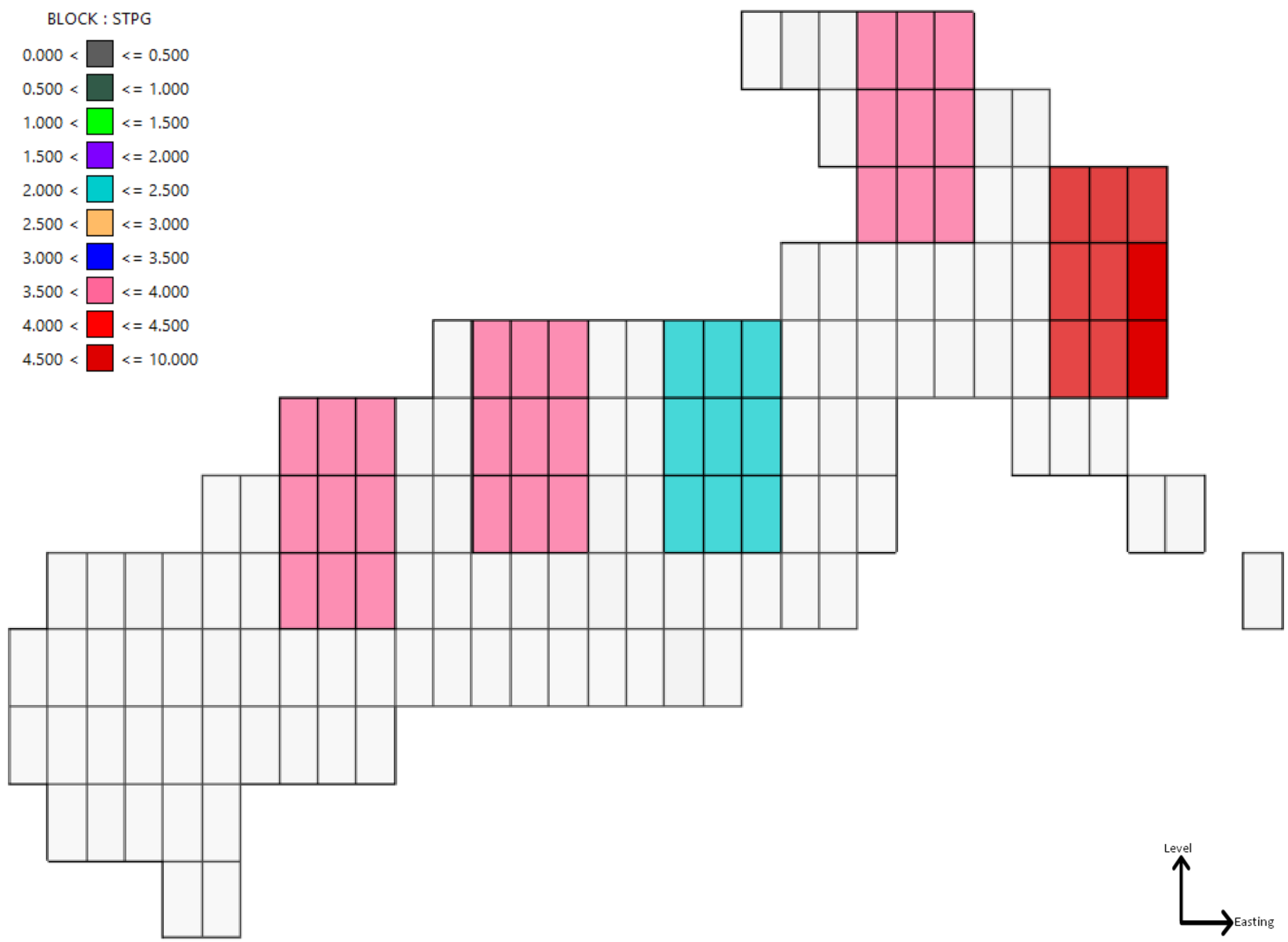

Figure 4.8 Experiment 3 Scenario 1 Optimal Layout

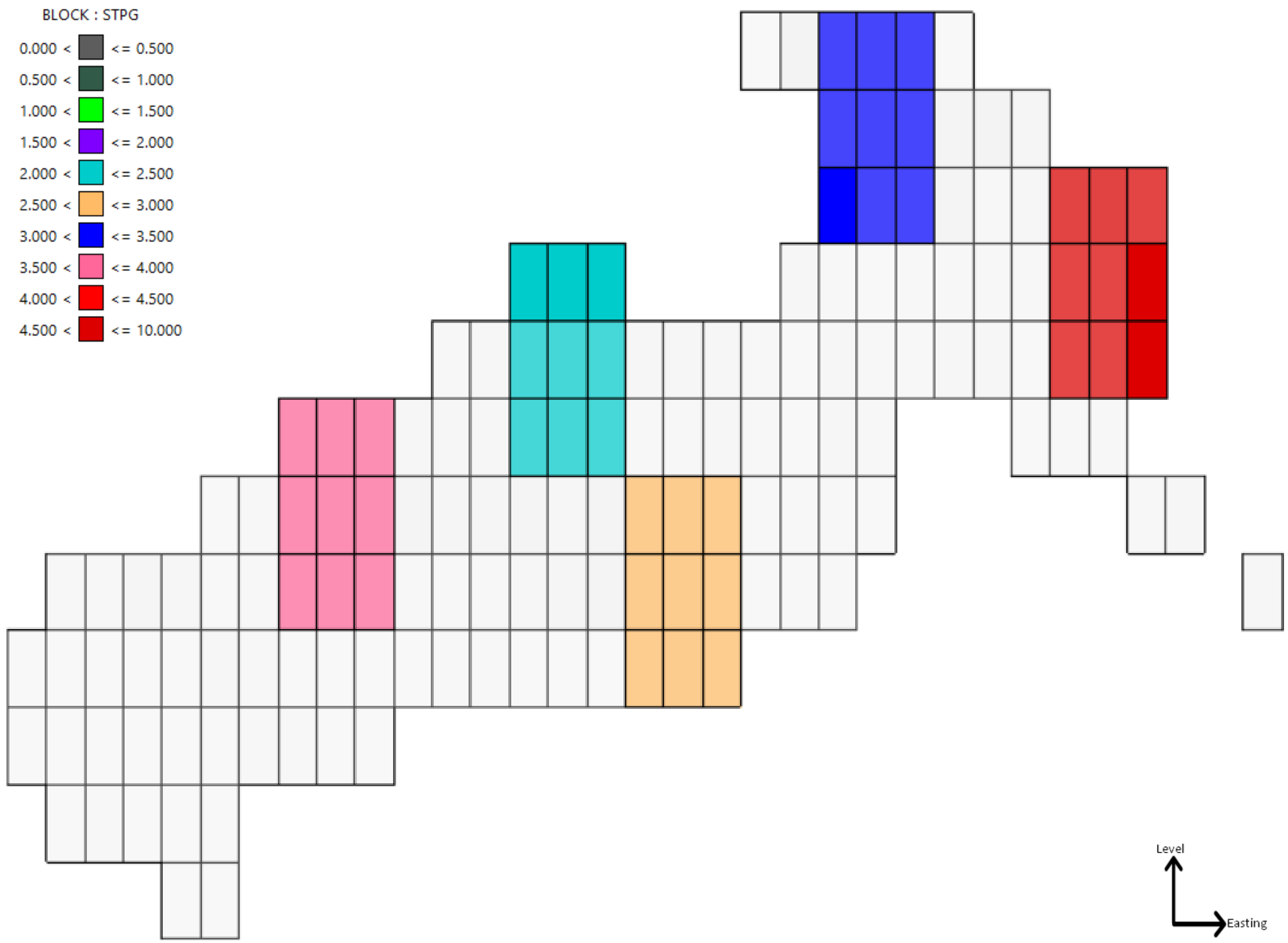

Figure 4.9 Experiment 3 Scenario 2 Optimal Layout

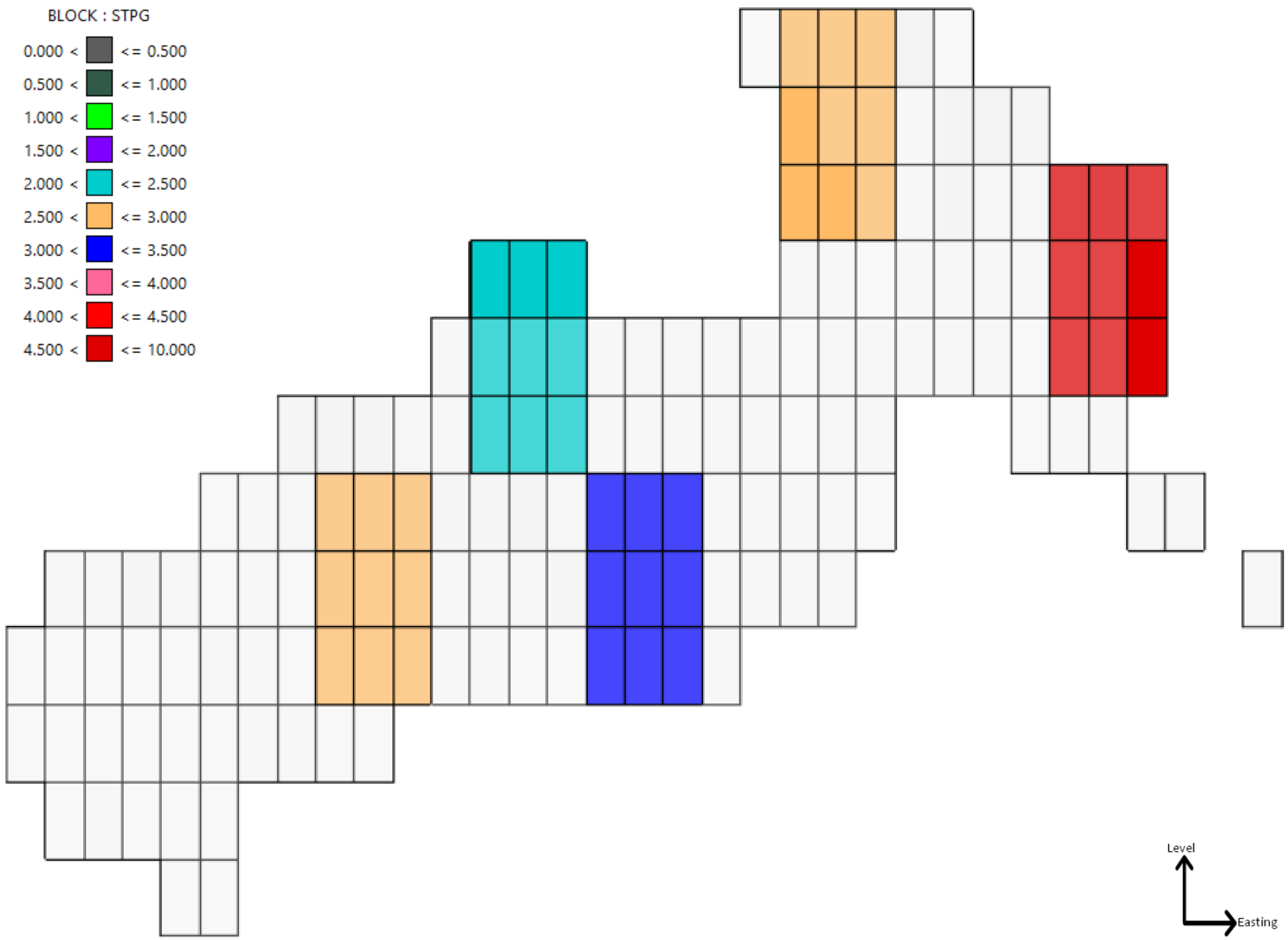

Figure 4.10 Experiment 3 Scenario 3 Optimal Layout

Conversely, selecting smaller pillars leads to a more compact layout, effectively minimizing the number of blocks left behind as pillars and optimizing resource extraction. There is no consistent trend in the solution times. All scenarios achieved the same overall block count (45 blocks) due to the square stope frame configuration however the smaller the pillar size the more ore blocks are mined and the larger the pillar size the more waste blocks are selected in the stopes. This is further supported by the finding that when pillar sizes grow, the optimal stope layout's average grade similarly falls. All these lead to the lower objective function value as the pillar size increases in the problem.

Each scenario had pillars in units of blocks forming around the stopes in the vertical and horizontal directions in the final layout. The pillars in the X-direction are more visible because of the orientation of the deposit. The spatial arrangement of the stopes in the layouts move from compact to sparse as pillar size increases. Figure 4.8 shows the result of Scenario 1 with all the five (5) stopes formed respected the pillar dimensions constraint with two units of blocks (pillars) separating each adjacent stope in the layout. The stopes generated also looked compact in terms of spatial distance to each other because of selecting a smaller dimension for the pillars between the stopes.

Figure 4.9 shows the optimal layout for Scenario 2, which contains five (5) stopes in the optimal layout each spatially separated by 3 units of blocks in both Z and X directions. The stopes generated are more spread out spatially in the final layout due to an increase in the minimum pillar requirement. The algorithm selects blocks farthest from the deposit's key central zone thus generating two (2) low-grade stopes, two (2) medium-grade stopes and one (1) high-grade stope. Two stopes (stopes 1 and 3) lie diagonally adjacent in the middle zone, highlighting the previously discussed limitation of this BILP model.

Figure 4.10 shows the optimal layout of Scenario 3, which contains five (5) stopes that respect the pillar dimension constraint. A total of 90 blocks were left as pillar blocks, which resulted in the stopes formed in the final layout being widely spaced apart. This explains the reason why it takes a long time to converge to a solution since the algorithm needs to find stopes that meet, the cutoff grade, the shape requirement as well as respect the required minimum pillar dimensions.

4.5. EFFECT OF NUMBER OF STOPES SELECTED

The number of stopes *K* is an important parameter that controls the number of variables and constraints created in the optimization problem. The number of stopes is chosen a priori as an input for this algorithm.

4.5.1. Input Data for Number of Stopes Selected Evaluation. It is important that the engineer avoids selecting a value of *K* that is lower than the number of "optimal" stopes for the specific problem. Otherwise, the problem will converge to a suboptimal solution. On the other hand, because *K* is directly related to the number of decision variables and constraints, too large a value of *K* will unnecessarily increase the computational time. The author investigated the effect of changing the number of stopes chosen for the optimization. Table 4.10 shows the variable stope numbers selected for each scenario in this experiment while Table 4.11 shows the BILP configuration for the experiment.

Table 4.10 Experiment 4 – Number of Stopes

BILP Configuration	**Scenario 1**	**Scenario 2**	**Scenario 3**	**Scenario 4**
Number Of Stopes k	4	10	20	30

Table 4.11 BILP Input Data – Experiment 4

BILP Configuration	**Scenario**
Minimum Mining Height α_1	3
Minimum Mining Width α_2	3
Maximum Mining Height β_1	3
Maximum Mining Width β_2	3
Minimum Pillar Length γ_1	2
Minimum Pillar Length γ_2	2
Cutoff Grade G_{off}	2 g/t

4.5.2. Results and Discussion. Table 4.12 shows a summary of the results while Figure 4.11 – Figure 4.13 show the optimal layouts of the scenarios. The result for Scenario 2 in this experiment is the same as Scenario 2 of Experiment 2. The results in Table 4.12 shows the solution time increases as the number of stopes increases, as one would expect. This is because of the explosion of variables and constraints from the higher number of stopes. When the number of stopes was lower than the optimal number of stopes (Scenario 1), the objective function value was lower than the *optimal* objective function value. The objective function value for the other scenarios (where the specified number of stopes is higher than the optimal number of stopes) is the same for all scenarios. This is what one would expect for this problem.

The limitation of the proposed BILP approach illustrated by this result is that the engineer seeking to optimize his/her stope layout needs to select the maximum number of stopes a priori that can result in an optimal solution. Otherwise, the model can yield suboptimal results as shown in this experiment. To address this the author recommends selecting a large enough number of stopes (e.g., by estimating the maximum number of stopes that will fit the domain if there was to be a stope in every region possible). The downside of this approach is that it will lead to the algorithm taking a long time to converge to an optimal solution.

Table 4.12 Experiment 4- Results of Changing Number of Stopes

Parameter	Units	Optimization Results			
		Scenario 1	Scenario 2	Scenario 3	Scenario 4
Objective Function Value	($)	24,828,692	27,455,739	27,455,739	27,455,740
Solution Time	(hrs)	0.07	0.66	3.37	9.05
Gap Tolerance	(%)	0.00	0.00	0.00	0.00
Number Of Stopes	(#)	4	5	5	5
Ore Blocks Mined	(#)	34	43	43	43
Waste Blocks Mined	(#)	2	2	2	2
Number Of Mined Blocks	(#)	36	45	45	45
Number Of Pillar Blocks	(#)	48	60	60	60
Minimum Stope Grade	(g/t)	3.63	2.49	2.49	2.49
Maximum Stope Grade	(g/t)	5.52	5.52	5.52	5.52
Average Layout Grade	(g/t)	4.25	3.86	3.86	3.86
Total Layout Tonnage	(tonnes)	534,600	668,250	668,250	668,250

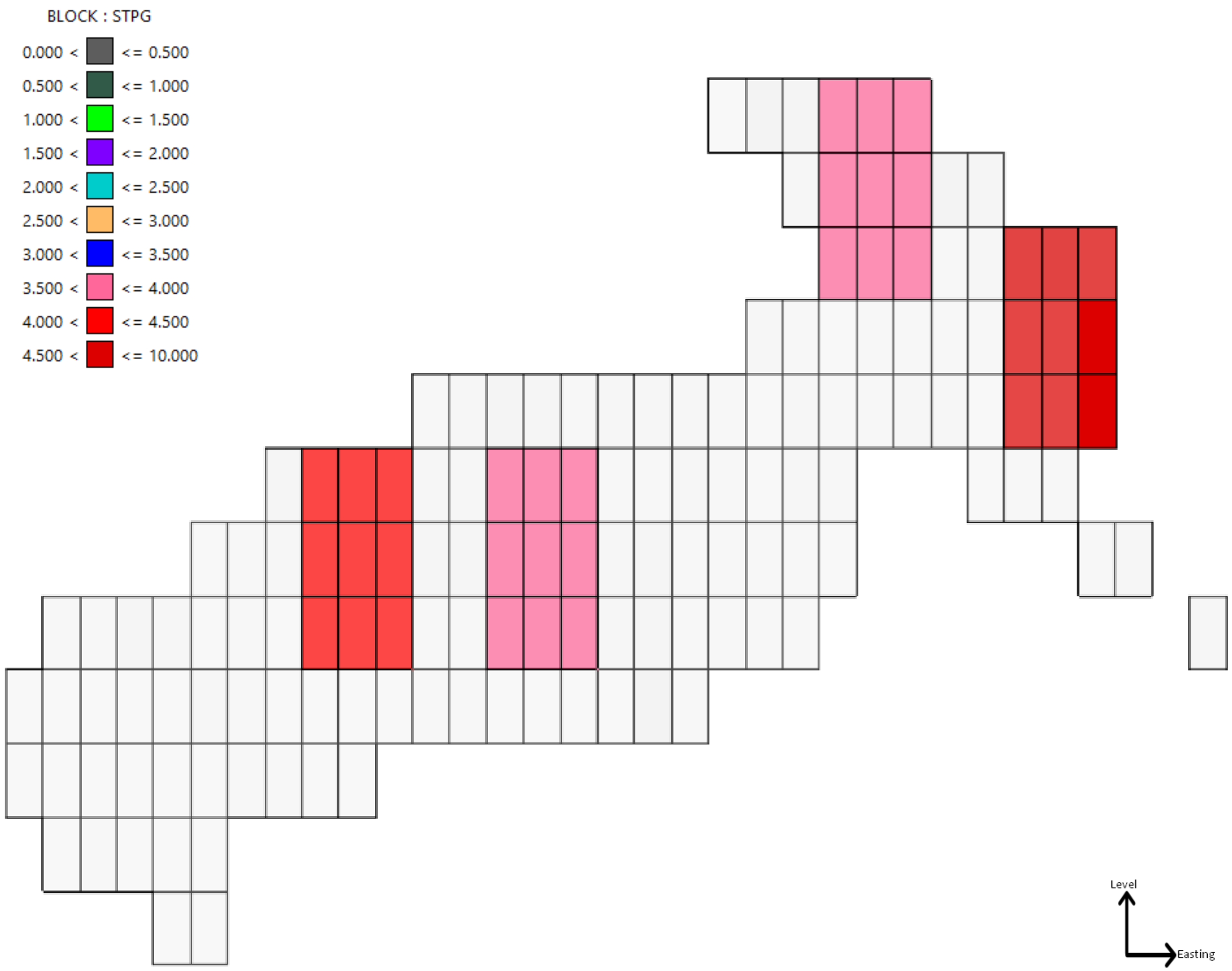

Figure 4.11 Experiment 4 Scenario 1 Optimal Layout

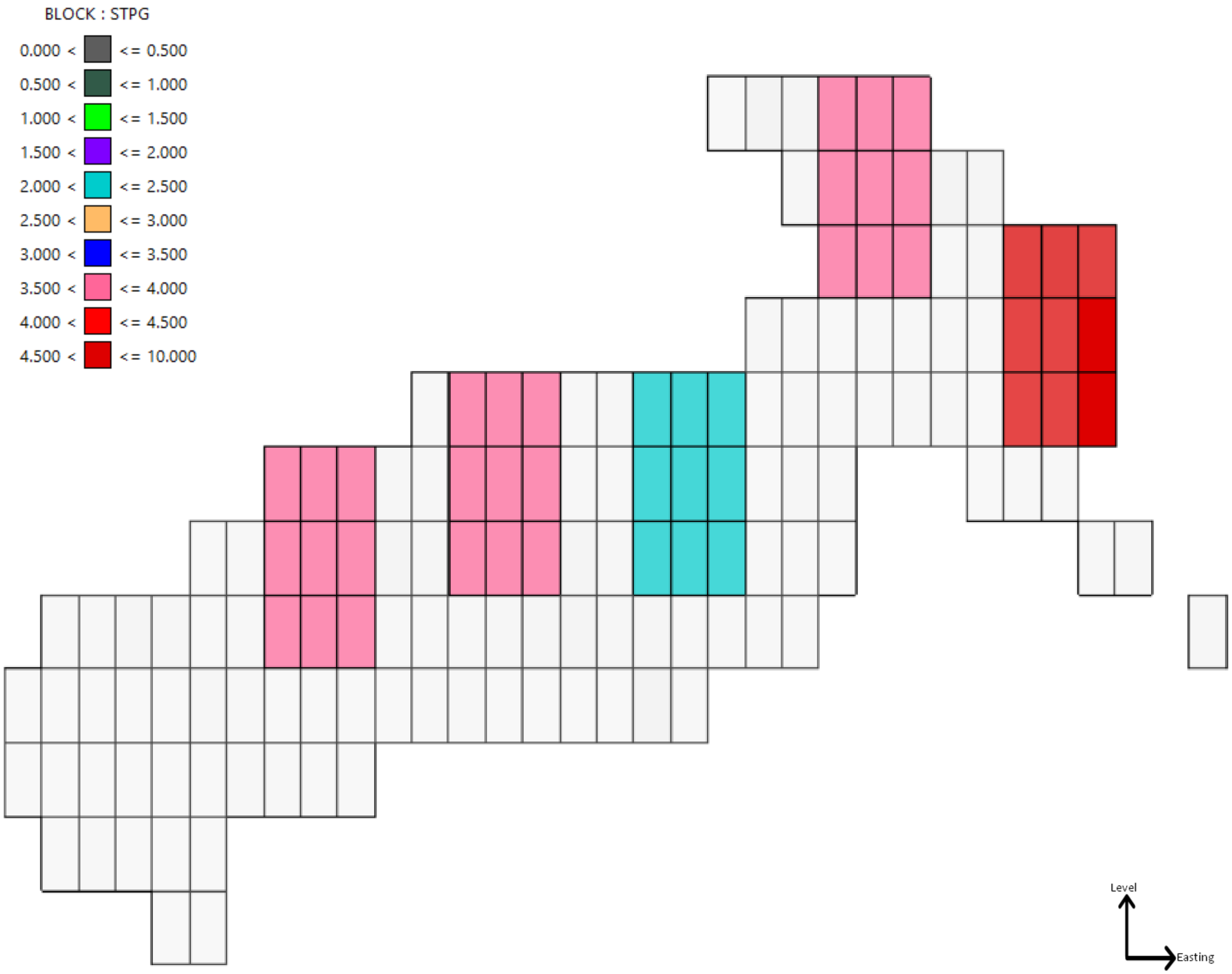

Figure 4.12 Experiment 4 Scenario 2 Optimal Layout

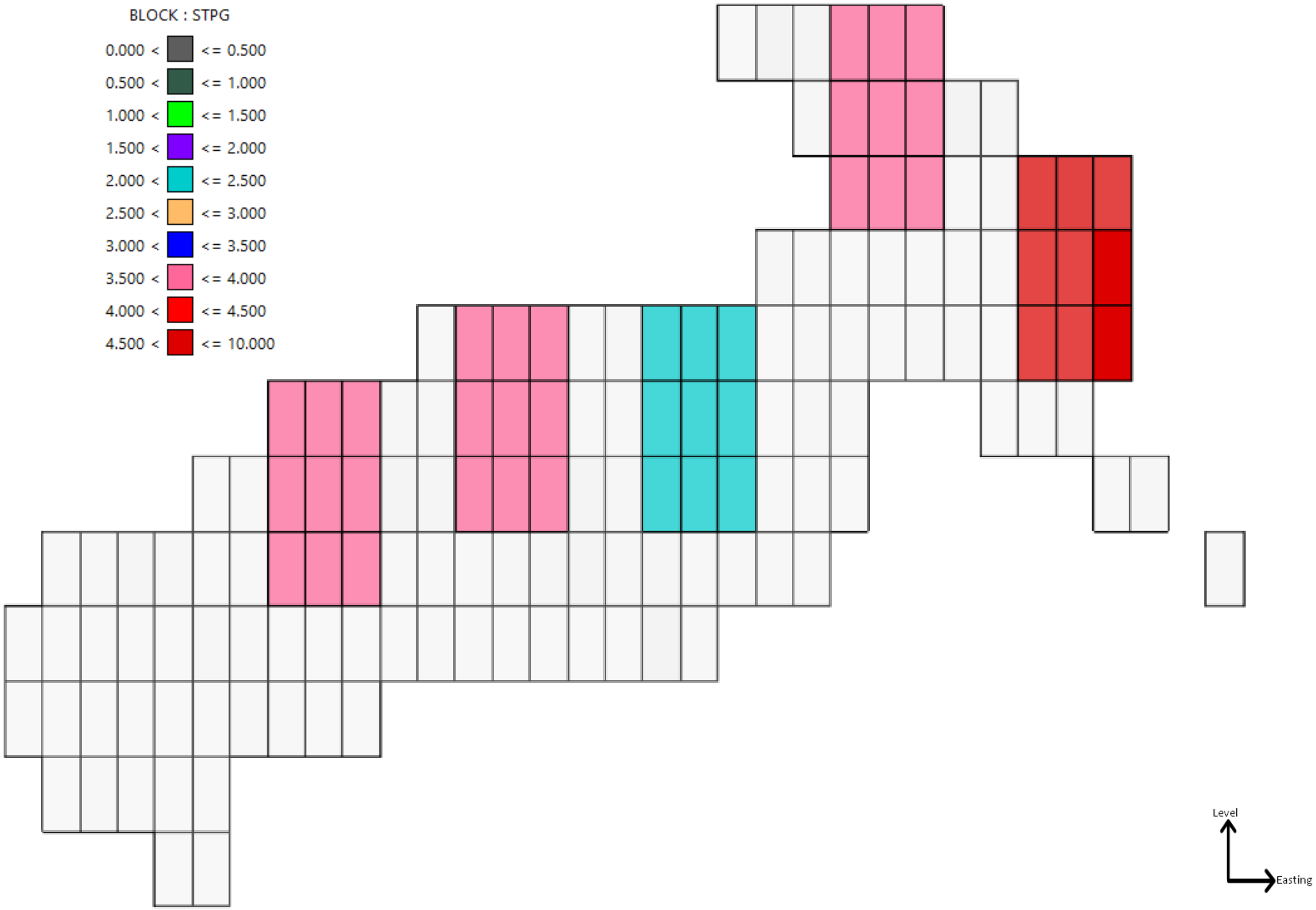

Figure 4.13 Experiment 4 Scenario 3 Optimal Layout

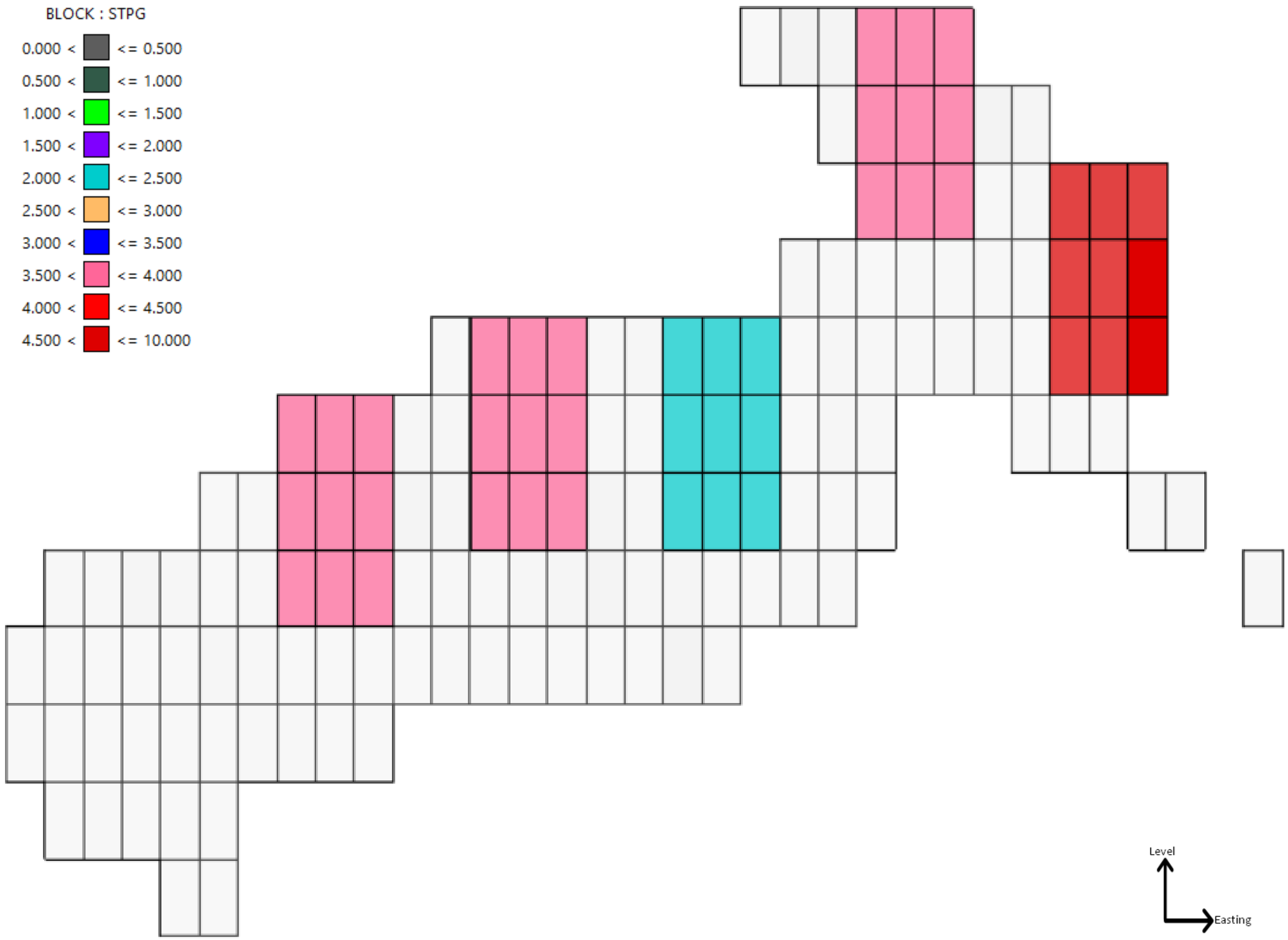

Figure 4.14 Experiment 4 Scenario 4 Optimal Layout

4.6. EFFECT OF OPTIMIZATION PROBLEM SIZE

As was previously mentioned in Section 3.8, binary integer programming models can evolve into a combinatorial explosion of variables and constraints. The author assesses the impact of applying the BILP model to solve different sized stope layout optimization problems.

4.6.1. Input Data for Optimization Problem Size Evaluation. To conduct this experiment the author reblocked the sample data set used for the basecase study into a two blockmodels with smaller block dimensions consequently generating more blocks in those blockmodels (1,000 blocks and 1,300 blocks). The author then generated economic values for the blockmodels for this experiment using the same procedure that was employed for the basecase scenario in Section 3.7.2.

To ensure consistency in the results, the scenarios in this experiment were solved using the same MATLAB code used in the prior analysis. The complete input data for this experiment can be accessed on GitHub (https://github.com/TheoMensah/BILP_SSLO.git). Table 4.13 shows the summary statistics of the block models. Table 4.14 shows the common input data for this experiment while Table 4.15 shows the number of blocks for each of the three models used in this experiment.

4.6.2. Results and Discussion. Table 4.16 shows a summary of the results of this experiment and Figure 4.16 –Figure 4.18 show the layouts. The result from this study suggests that the size of the optimization problem does have an impact on the solution obtained and this is evident in the solutions times achieved in each scenario (Figure 4.15). All formulations converged to an optimal solution with gap tolerance of 0%.

Table 4.13 Summary Statistics of Block Models

Blockmodel Attribute	Reblocked Model 1	Reblocked Model 2	Reblocked Model 3
Metal	Au	Au	Au
Number of blocks (#)	774	1,000	1,300
Blocks Au > 0	144	191	246
Total mineralized material (t)	2,138,400	2,138,400	2,138,400
Maximum Au value (g/t)	14.91	9.72	9.97
Minimum Au value (g/t)	0.15	0.11	0.29
Average Au value (g/t)	2.63	2.55	2.59
Standard deviation (%)	2.09	1.66	1.67
Variance ($\%^2$)	4.35	2.74	2.80
Block density (kg/m3)	2.2	2.2	2.2
Block size (#)	15 ×15 ×30	13 ×15 ×27	10 ×15 ×27
Block tonnage (t)	6,750	5,225	4,050
Depth from surface (m)	560 – 1,100	560 – 1,100	560 – 1,100

Table 4.14 BILP Input Data – Experiment 5

BILP Configuration	Scenario
Minimum Mining Height α_1	4
Minimum Mining Width α_2	4
Maximum Mining Height β_1	4
Maximum Mining Width β_2	4
Minimum Pillar Length γ_1	3
Minimum Pillar Length γ_2	3
Number Of Stopes k	20
Cutoff Grade G_{off}	3g/t

Table 4.15 Experiment 5 –Optimization Problem Size

BILP Configuration	Scenario 1	Scenario 2	Scenario 3
Number of Blocks (i, j)	774	1000	1300

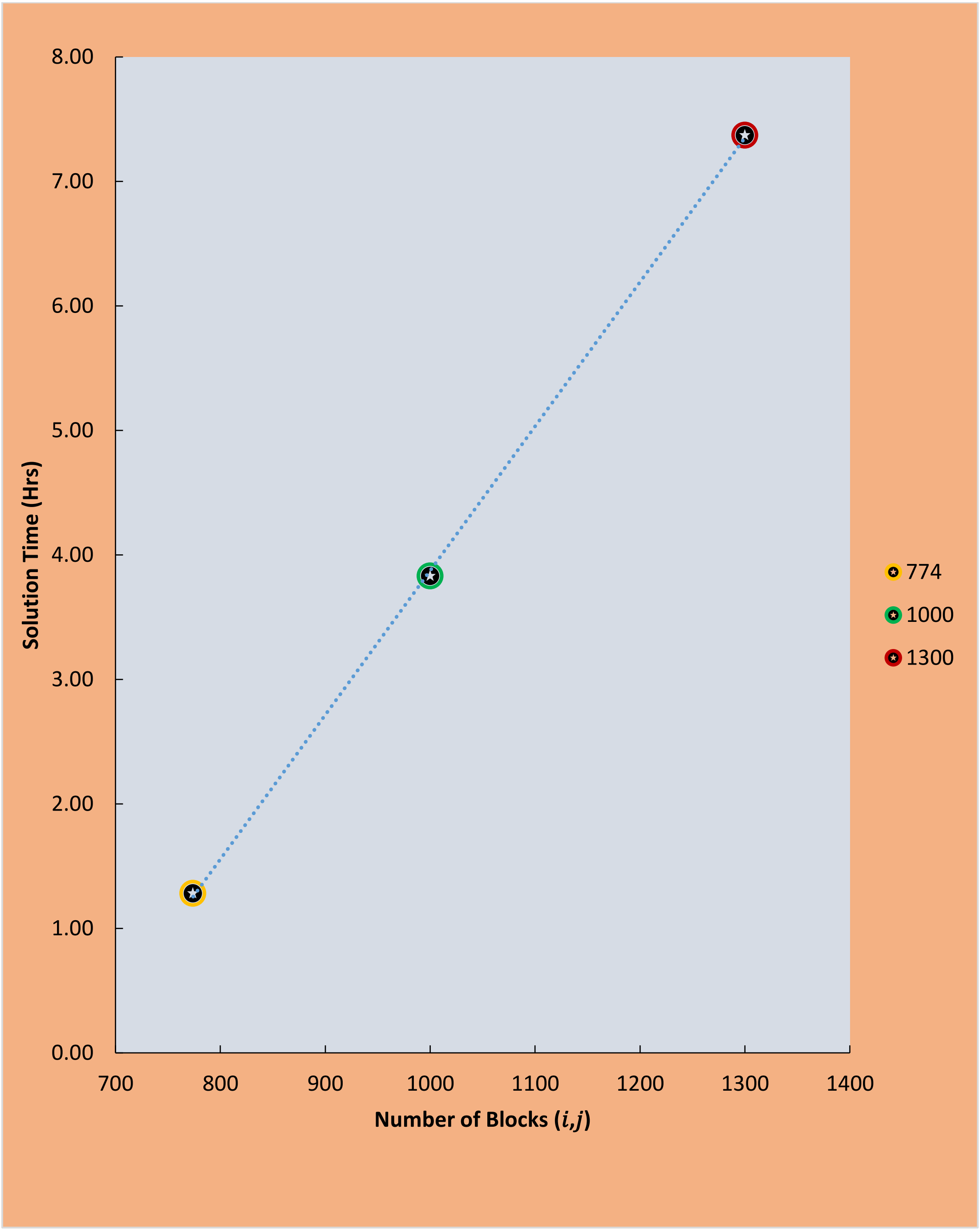

Figure 4.15 Solution Time versus Optimization Problem Size (Number of Blocks)

Table 4.16 Experiment 5- Results of Changing Optimization Problem Size

Parameter	Units	Optimization Results		
		Scenario 1	Scenario 2	Scenario 3
Objective Function Value	($)	24,627,142.8	24,990,276.6	25,358,764.9
Solution Time	(hrs)	1.28	3.83	7.37
Gap Tolerance	(%)	0.00	0.00	0.00
Number of Stopes formed	(#)	3	3	5
Ore Blocks Mined	(#)	46	46	78
Waste Blocks Mined	(#)	2	0	2
Number of mined blocks	(#)	48	48	80
Number of pillar blocks	(#)	72	95	130
Minimum Stope Grade	(g/t)	3.08	3.22	3.12
Maximum Stope Grade	(g/t)	3.92	3.55	3.70
Average Layout Grade	(g/t)	3.38	3.37	3.45
Total Layout Tonnage	(tonnes)	712,800	712,800	712,800

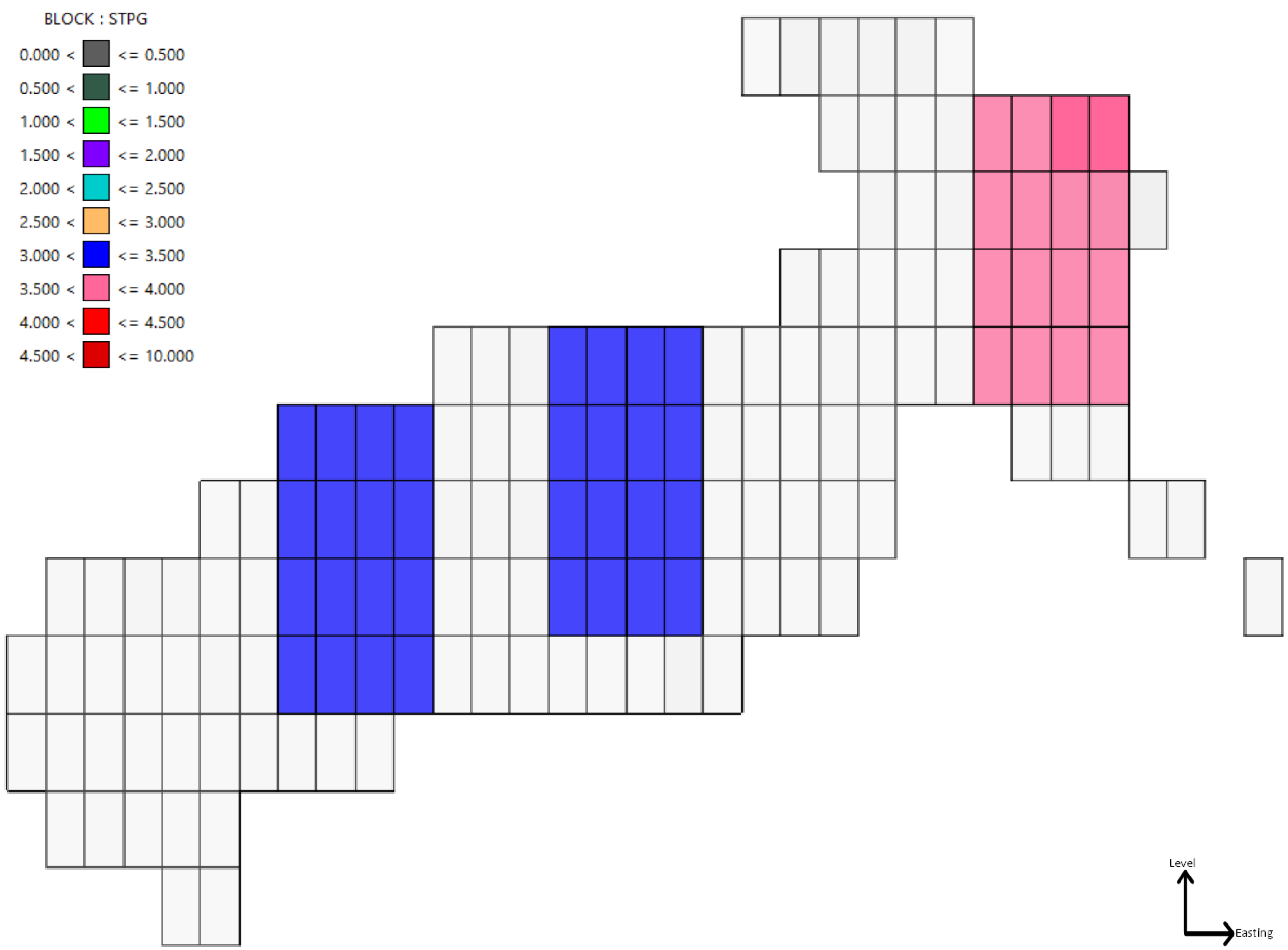

Figure 4.16 Experiment 5 Scenario 1 Optimal Layout

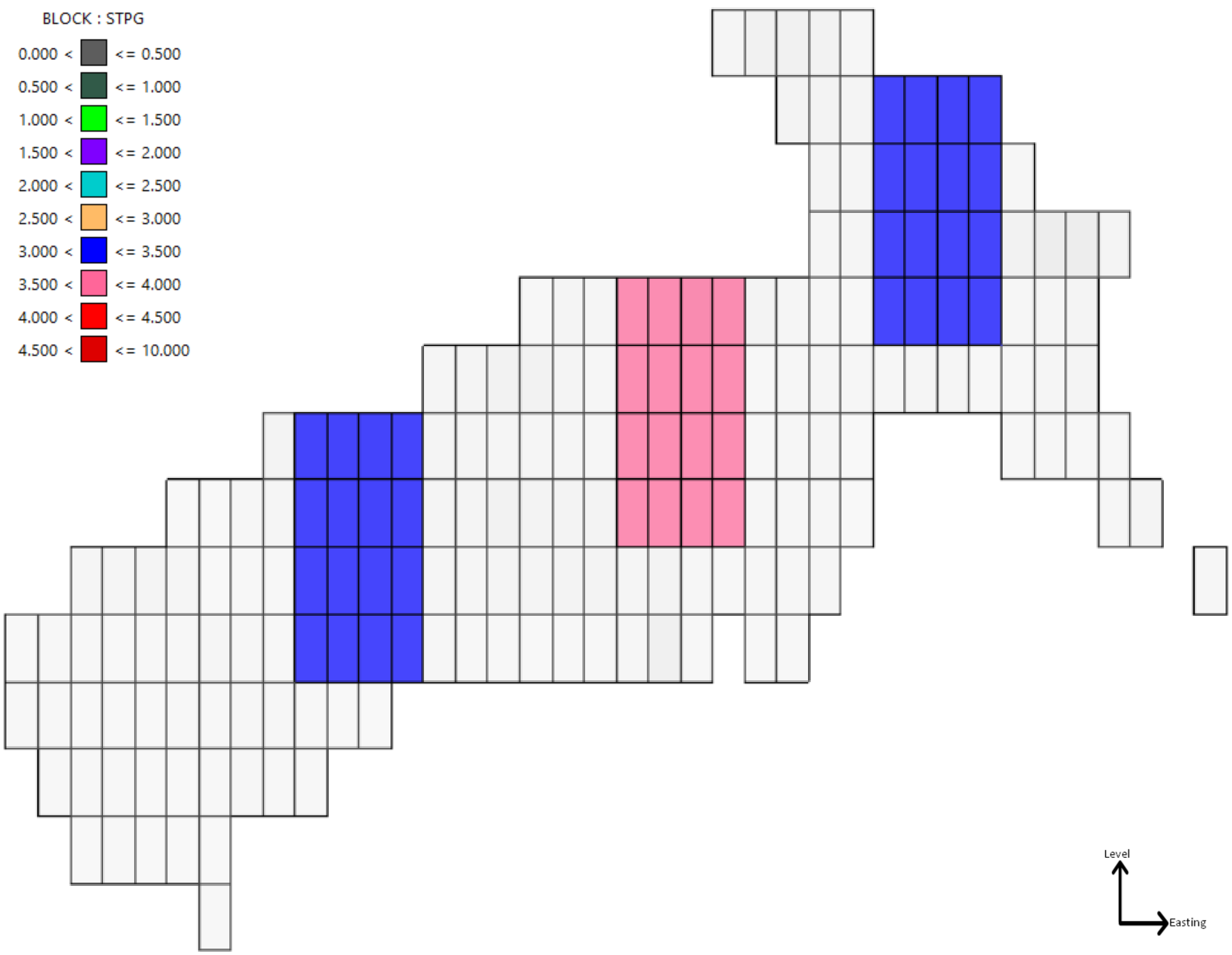

Figure 4.17 Experiment 5 Scenario 2 Optimal Layout

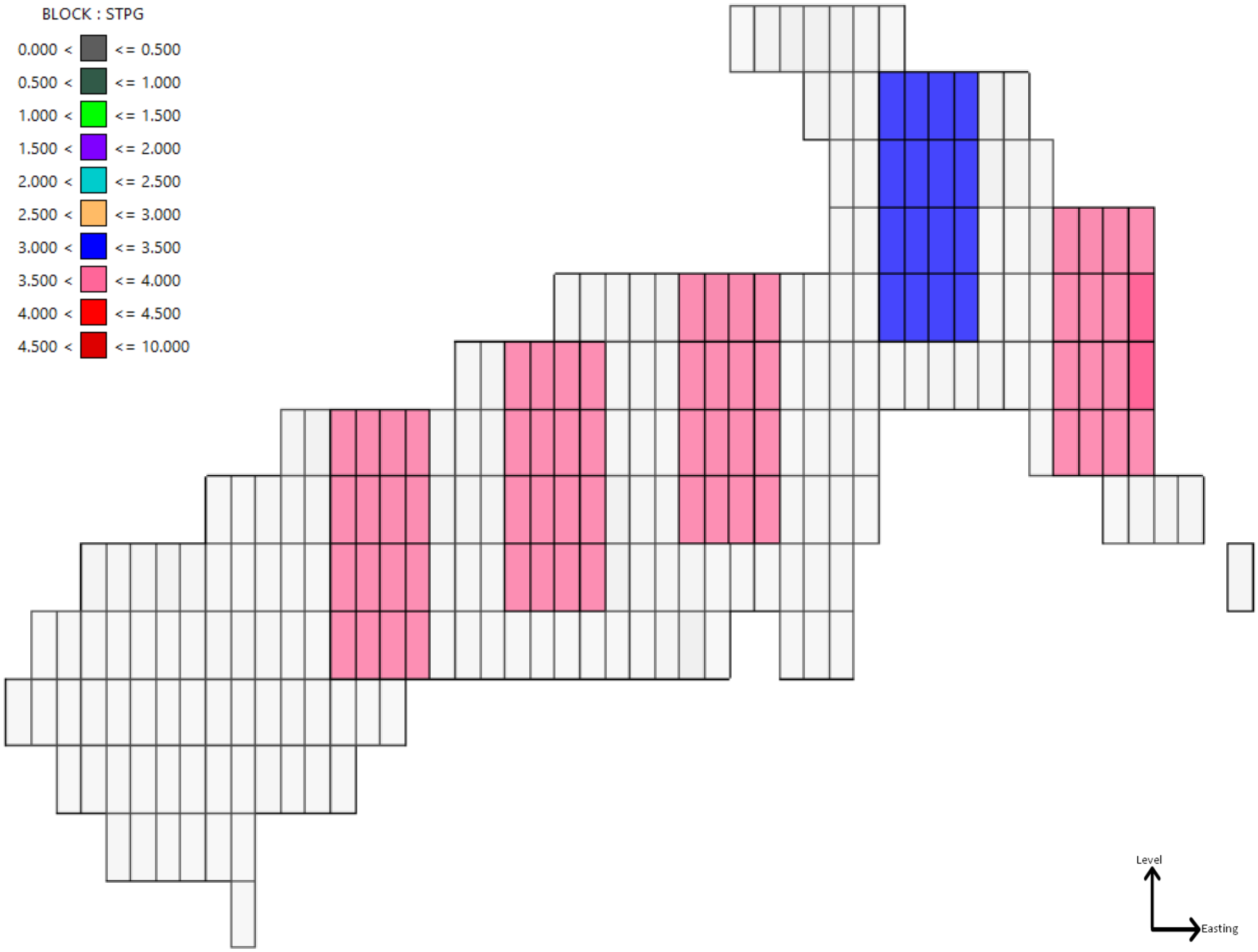

Figure 4.18 Experiment 5 Scenario 3 Optimal Layout

From the study, Figure 4.15 illustrates a scatter plot between optimization problem (number of blocks) and the solution time achieved. The plot demonstrates that there is a linear correlation between the optimization problem size and the time required to converge to an optimal solution. This relationship occurs because as the number of blocks increases, the decision variables and constraints also grow, resulting in a more complex problem necessitating greater computational time and resources for the algorithm to converge towards an optimal solution. The results in Table 4.16 indicates Scenario 1 (blocks $(i, j) =$ 774) had the fastest solution time of 1.28 hrs. Scenario 2 (blocks $(i, j) =$ 1,000) achieved this in 3.83 hrs while Scenario 3 (blocks $(i, j) =$ 1,300) achieved the optimal solution in the longest time of 7.37 hrs. While this experiment shows a linear relationship, it is not yet clear whether this is the case for a broad range of problem sizes. If indeed, the computational time grows only linearly, this will be an advantage of this model. Further work is required to examine this observation.

This findings from this experiment demonstrates yet another flaw in the BILP approach, namely the tendency for variable and constraint combinations to explode as optimization problem size increases requiring greater computational time and resources for the algorithm to converge to an optimal solution.

From Figure 4.16 – Figure 4.18 the author reblocked the basecase model into varying block sizes. To maintain consistency in this experiment, the size of the pillars and stopes dimensions were kept the same. However due to the varying block sizes the final designs will change. This approach was adopted to avoid introducing additional confounding factors, such as changes in the number of pillar and stope constraints per block that would occur with different stope sizes. The reblocking process redistributes the grade in the model as well as increase the block count for small sized blocks.

From the results, smaller block sizes allow the algorithm more selectivity. Thus, as can be seen in the optimal layout of Scenario 3 (Figure 4.18), the algorithm had the flexibility to include more blocks hence forming 5 stopes – 2 more than Scenarios 1 and 2. This is because the smaller sized blocks have a lower tonnage and to achieve the maximum value more higher grade stopes needs to be formed in order to achieve the maximum economic value. Though the same final layout tonnage is achieved in all scenarios, objective function value is slightly higher for the smaller block sized problem (Scenario 3). This is because grade variability is also increased in the smaller block sizes making the algorithm more selective in the blocks to include in a stope.

4.7. SUMMARY

This section presented a set of computational experiments to assess the proposed BILP model's sensitivity to the key input parameters of the stope layout optimization problem. The work in this chapter evaluated the effect of differences in cutoff grade, stope dimensions, pillar dimensions, (maximum) number of stopes selected by user, and size of the optimization problem (measured in terms of the number of blocks).

The following are the major findings from the experiments:

- The BILP model can find the optimal solution for many different types of problems. For all the scenarios evaluated, the solution was found within an optimality gap of 0.00%.

- The model is sensitive to changes in the cutoff grade. While a high cutoff grade will speed up the algorithm's solution time, this will generate optimal layouts with a lower objective function value.
- The model can generate stopes of any rectangular shape (stope frame) specified by the engineer and will mimic the deposits peripheries. The stope frames the engineer selects affects the objective function value, solution time, number of stopes formed, and general stope layout.
- The model is also sensitive to the specified pillar sizes. Specifying larger pillar sizes results in more spatially spread stopes in the layout which leads to optimal layouts with a lower objective function (economic) value. Specifying smaller pillars lead to more compact stopes in the optimal layout with higher objective function (economic) values.
- The model is highly sensitive to specified (maximum) number of stopes and the number of blocks (used as a proxy for optimization problem size) because both parameters affect the number of decision variables and constraints. The larger the specified (maximum) number of stopes and the higher the number of blocks the longer the solution time.
- Because the engineer needs to select the maximum number of stopes a priori and it has such a significant effect on solution times and the solution, this work proposes that engineers using this model estimate the maximum number of stopes possible in the geometry and use that estimate for the maximum number of stopes. This ensures there are enough stopes to yield the optimum solution but not more than necessary.

5. CONCLUSIONS, RECOMMENDATIONS & FUTURE WORK

5.1. OVERVIEW

Stope layout optimization is a critical aspect of underground mining operations. It involves determining the most effective arrangement of stopes within a mine to maximize resource extraction and operational efficiency. The primary objective is to design a layout that optimizes the economic value while considering various factors such as geotechnical constraints, stope grade, equipment limitations, and safety requirements.

Previous researchers have used meta-heuristic optimization methods, including swarm intelligence algorithms, genetic algorithms, and particle swarm optimization, to find optimal stope layouts. These approaches however do not guarantee optimality. One approach that researchers are utilizing now involves formulating the problem as a mathematical optimization model, typically using linear programming techniques to determine the optimal arrangement of stopes. This approach is well known to guarantee optimality and if well formulated, can be configured to solve complex problems.

Thus, the goal of this thesis work was to formulate the stope layout optimization problem (SLOP) as a binary integer linear programming problem that maximizes the value of the stopes mined subject to novel grade, geotechnical (minimum and maximum pillar sizes), and allowable mining (minimum and maximum stope width and height) constraints in two-dimensional space. To achieve this goal, the author:

1. Drew from Queyranne and Wolsey's [25], [26] formulations of tight constraints for bounded up/down times in production planning problems to formulate novel

and efficient geometric constraints along with geotechnical and grade constraints for the BILP stope layout optimization problem.

2. verified the novel BILP model with a sample gold mining data set to verify the model. The original geological model of the orebody was regularized to generate equal-sized blocks ideal for conversion into an economic model which served as the primary input for the 5-experimental 15-scenario runs to verify the BILP model as a model that applies efficient shape constraints in solving the SLOP in two-dimensional space.

5.2. CONCLUSIONS

The study concludes the following from the outcome of the basecase and experimental computations:

- With respect to the outcomes from the basecase:
    - ❖ The results from the basecase study show that it is possible to model shapes using LP-based techniques for the stope layout problem. Unlike most LP-based stope layout approaches, the proposed model accounts for efficient shape constraints in the geometric constraints.
    - ❖ The developed model can find the optimal stope layout that maximizes the undiscounted profit for the deposit within a gap tolerance of 0.00%.
    - ❖ The model allows the user to generate variable stope length and height as well as incorporate geotechnical pillar requirements between the generated stopes. Thus, the model permits mining operations to follow irregular mineral deposit peripheries to minimize dilution while ensuring a stable operating environment.

  - ❖ The proposed BILP model has some limitations as illustrated by the results of the base case experiment. Pillars are respected around the stopes however pillar widths are not maintained in the diagonal direction because the pillar limitations are defined along the vertical and horizontal directions. This will require future post-processing to ensure the stope layouts are safe.

- With respect to the computational experiments to evaluate the BILP model's sensitivity to key input variables and parameters:
  - ❖ The proposed BILP model is sensitive to the selected stope cutoff grade. The lower the cutoff grade selected the higher the objective function achieved and vice versa. Also, there is a trade-off between the solution time and the objective function value achieved. A high stope cutoff grade means solution converges faster but it does not improve the objective function value.
  - ❖ The model is also very sensitive to the required minimum stope dimensions. As stated above, the model can generate variable stope frames based on the input. Larger stope frames achieve higher objective function values (assuming pillar sizes stay the same) and converges faster since more blocks can be selected and fewer blocks are left behind as pillars. Smaller frames generate more stopes in the layout but requires more pillars thus leaving some blocks behind and achieving a lower objective function value.

- ❖ The model is sensitive to the specified pillar dimensions. The smaller the pillar dimensions selected the fewer the blocks left behind as pillars thus maximizing the economic value of the layout and vice versa. The solution time is not affected by the choice of pillar.
- ❖ The performance of the model is affected by the choice of maximum number of stopes. The maximum number of stopes is chosen a priori which means the engineer can select a larger number which will lead to a longer time for the algorithm to converge or a smaller number which will lead to suboptimal solutions. The author proposes that engineers using this model estimate the maximum number of stopes possible in the geometry and use that estimate for the maximum number of stopes. This ensures there are enough stopes to yield the optimum solution but not more than necessary.
- ❖ The BILP model proposed is highly sensitive to the size of the optimization problem. There appears to be a linear correlation between the problem size and the solution time. The larger the problem size the more exponentially the variables and constraints grow making the problem more complex and necessitating greater computational resources and time to solve such problems. However, because the experiments in this work are very limited, it is not clear if the trend is the same for a wide variety of problem sizes.

- ▪ The proposed model contributes to the research on underground stope layout optimization by demonstrating the possibility to formulate efficient shape constraints in a binary integer programming model to solve the stope layout optimization problem. The work evaluated systematically, the sensitivity and performance of the proposed BILP model with respect to stope layout optimization's key input parameters as well as optimization problem size. The work also demonstrated the adaptation of Queyranne and Wolsey's [25], [26] tighter constraints for production planning to model novel tighter and more efficient constraints for the stope layout optimization problem.

5.3. RECOMMENDATIONS FOR FUTURE WORK

To further improve and advance the proposed binary-integer linear programming model for optimizing the stope layout optimization problem, the following recommendations are made for future work:

- The model currently does not implement pillars diagonally which means stopes can be generated diagonally adjacent to each other. This can lead to stability issues during stoping. Future work should develop post-processing algorithms that can detect "diagonal" pillars that violate pillar constraints and use heuristics to adjust the layout to avoid these situations.
- The utilization of binary variables in this model leads to longer solution times, necessitating the implementation of strategies to reduce the computational cost associated with the model. One such strategy is to preprocess the problem using heuristic techniques prior to passing the problem to a standard BILP solver. Future work should develop such preprocessing algorithms to improve the solution times.

- Future work should account for mine access networks in the solution. Mine development layouts are essential for stope layout design and should be incorporated into the stope layout optimization problem.
- It will be essential to integrate the stope production scheduling problem with this model.
- Due to the high uncertainty that characterizes mining operations, it is recommended that future work should model a stochastic optimization model of this BILP model to address uncertainty in model parameters.
- Lastly, the proposed model is in two-dimensional space (2D). To make this model more practical with realistic outputs for application in real-life mining scenarios, the model should be extended to three-dimensional (3D) space. Future work should incorporate a third set of variables and constraints in the third dimension to extend the proposed model's framework to 3D space.

BIBLIOGRAPHY

[1] A. Copco, “Mining methods in underground mining,” in *Atlas Copco: Nacka, Sweden*, 2007.

[2] P. Darling, *SME Mining Engineering Handbook*, Third Edition., vol. 1. Englewood, Co: Society for Mining, Metallurgy and Exploration, 2011.

[3] S. Mijalkovski, D. Peltechki, Z. Despodov, D. Mirakovski, V. Adjiski, and N. Doneva, “METHODOLOGY FOR UNDERGROUND MINING METHOD SELECTION,” *Mining Science*, vol. 28, pp. 201–216, 2021, doi: 10.37190/msc202815.

[4] F. e F. Matheus, R. Dimitrakopoulos, and C. Pinto, “Stochastic stope design optimisation under grade uncertainty and dynamic development costs,” *Int J Min Reclam Environ*, vol. 36, no. 2, pp. 81–103, 2022, doi: https://doi.org/10.1080/17480930.2021.1968707.

[5] G. Samanta, T. Dey, and B. Samanta, “Stope boundary optimization for an underground copper deposit using mixed integer linear programming based algorithm,” *Journal of Mines, Metals and Fuels*, vol. 69, p. 73, Jan. 2021, doi: 10.18311/jmmf/2021/27781.

[6] Y. A. Sari and M. Kumral, “Sublevel stope layout planning through a greedy heuristic approach based on dynamic programming,” *Journal of the Operational Research Society*, vol. 72, no. 3, pp. 554–563, 2021, doi: 10.1080/01605682.2019.1700179.

[7] D. S. S. Sandanayake, E. Topal, and M. W. A. Asad, “Designing an optimal stope layout for underground mining based on a heuristic algorithm,” *Int J Min Sci Technol*, vol. 25, no. 5, pp. 767–772, 2015, doi: https://doi.org/10.1016/j.ijmst.2015.07.011.

[8] G. Erdogan, M. Cigla, E. Topal, and M. Yavuz, “Implementation and comparison of four stope boundary optimization algorithms in an existing underground mine,” *Int J Min Reclam Environ*, vol. 31, no. 6, pp. 389–403, Aug. 2017, doi: 10.1080/17480930.2017.1331083.

[9] G. Erdogan, M. Cigla, E. Topal, and M. Yavuz, "Implementation and comparison of four stope boundary optimization algorithms in an existing underground mine," *Int J Min Reclam Environ*, vol. 31, no. 6, pp. 389–403, Aug. 2017, doi: 10.1080/17480930.2017.1331083.

[10] B. O. Afum and E. Ben-Awuah, "A Review of Models and Algorithms for Surface-Underground Mining Options and Transitions Optimization: Some Lessons Learnt and the Way Forward," *Mining*, vol. 1, no. 1, pp. 112–134, May 2021, doi: 10.3390/mining1010008.

[11] K. Shahriar and K. Oraee, "A Study on the Optimization Algorithms for Determining Open-Pit and Underground Mining Limits," in *7th International Scientific Conference, SGEM 2007*, International Multidisciplinary Scientific GeoConferences (SGEM), 2007.

[12] M. Osanloo, J. Gholamnejad, and B. Karimi, "Long-term open pit mine production planning: A review of models and algorithms," *International Journal of Mining, Reclamation and Environment*, vol. 22, no. 1. pp. 3–35, Mar. 2008. doi: 10.1080/17480930601118947.

[13] E. J. A. Appianing, E. Ben-Awuah, and Y. Pourrahimian, "Life-of-mine optimization for integrated open stope development and production scheduling using a mixed-integer linear programming framework," *Mining Technology*, pp. 1–15, Mar. 2023, doi: 10.1080/25726668.2023.2182285.

[14] J. Little and E. Topal, "Strategies to assist in obtaining an optimal solution for an underground mine planning problem using Mixed Integer Programming," *Int J Min Miner Eng*, vol. 3, no. 2, pp. 152–172, 2011.

[15] J. Manchuk and C. Deutsch, "Optimizing stope designs and sequences in underground mines," *SME Transactions*, vol. 324, pp. 67–75, Jan. 2008.

[16] J. Little, E. Topal, and P. Knights, "Simultaneous optimisation of stope layouts and long term production schedules," *Transactions of the Institutions of Mining and Metallurgy, Section A: Mining Technology*, vol. 120, no. 3, pp. 129–136, 2011, doi: 10.1179/1743286311Y.0000000011.

[17] J. Hou, C. Xu, P. A. Dowd, and G. Li, "Integrated optimisation of stope boundary and access layout for underground mining operations," *Mining Technology*, vol. 128, no. 4, pp. 193–205, 2019, doi: 10.1080/25726668.2019.1603920.

[18] A. Chowdu, P. Nesbitt, A. Brickey, and A. M. Newman, “Operations Research in Underground Mine Planning: A Review,” *Interfaces (Providence)*, vol. 52, no. 2, pp. 109–132, 2022, doi: 10.1287/inte.2021.1087.

[19] E. J. A. Appianing and E. Ben-Awuah, “Underground Mining Stope Layout Optimization and Production Scheduling: A Review of Existing Solvers and Algorithms,” *MOL Report Nine*, pp. 271–304, 2018.

[20] A. S. Nhleko, T. Tholana, and P. N. Neingo, “A review of underground stope boundary optimization algorithms,” *Resources Policy*, vol. 56, pp. 59–69, Jun. 2018, doi: 10.1016/j.resourpol.2017.12.004.

[21] V. Nikbin, E. Mardaneh, M. W. A. Asad, and E. Topal, “Pattern search method for accelerating Stope boundary optimization problem in underground mining operations,” Taylor and Francis Ltd., 2022. doi: 10.1080/0305215X.2021.1932869.

[22] S. A. Abdulsalaam, “A FAST DYNAMIC PROGRAMMING ALGORITHM FOR STOPE BOUNDARY LAYOUT FOR UNDERGROUND MINE,” *Int J Adv Res (Indore)*, vol. 9, no. 09, pp. 86–90, Sep. 2021, doi: 10.21474/IJAR01/13378.

[23] S. E. Jalali and M. Ataee-pour, “A 2D dynamic programming algorithm to optimize stope boundaries,” in *Mine Planning and Equipment Selection 2004: Proceedings of the Thirteenth International Symposium on Mine Planning and Equipment Selection, Wroclaw, Poland, 1-3 September 2004*, 2004, p. 45.

[24] Z. Basiri and Y. Pourrahimian, “Application of mathematical programming for stope layout and production schedule optimization in sublevel stoping,” *Mining Optim. Laboratory*, vol. 8, no. 1, pp. 235–245, 2018.

[25] M. Queyranne and L. A. Wolsey, “Tight MIP formulations for bounded up/down times and interval-dependent start-ups,” *Math Program*, vol. 164, no. 1–2, pp. 129–155, Jul. 2017, doi: 10.1007/s10107-016-1079-2.

[26] M. Queyranne, “Extended Formulations for Shape Constraints in Mine Planning,” in *DEPLAMIN*, Santa Cruz Chile, 2018.

[27] J. Fuykschot, “Strategic Mine Planning Flexible mine planning to meet changes in the business environment,” in *Minex Conference 2009*, SRK Consulting, 2009, pp. 1–18.

[28] M. Emelly and V. Matamoros, “A Novel Stope Layout Optimization Based on Robust Genetic Algorithms,” Doctoral Thesis, McGill University (Canada), 2018.

[29] D. Suneth and S. Sandanayake, “Stope Boundary Optimization in Underground Mining Based on a Heuristic Approach,” Doctoral Thesis, Curtin University, 2014.

[30] A. M. Newman, E. Rubio, R. Caro, A. Weintraub, and K. Eurek, “A review of operations research in mine planning,” *Interfaces (Providence)*, vol. 40, no. 3, pp. 222–245, May 2010, doi: 10.1287/inte.1090.0492.

[31] J. De la Vergne, “Hard Rock Miners Handbook, edition 5,” *Ground Water*, vol. 15, pp. 35–45, 2008.

[32] C. Musingwini, “Presidential address: Optimization in underground mine planning-developments and opportunities,” *J South Afr Inst Min Metall*, vol. 116, no. 9, pp. 809–820, Sep. 2016, doi: 10.17159/2411-9717/2016/v116n9a1.

[33] M. Janiszewski, S. Pontow, and M. Rinne, “Industry Survey on the Current State of Stope Design Methods in the Underground Mining Sector,” *Energies (Basel)*, vol. 15, no. 1, Jan. 2022, doi: 10.3390/en15010240.

[34] “Review of:‘Techniques in Underground Mining: Selections from Underground Mining Methods Handbook’ (1998). Edited by Richard E. Gertsch and Richard L. Bullock. Society for Mining, Metallurgy, and Exploration Inc. P.O. Box 625002. Littleton, CO, U.S.A. 80162-5002.,” *International Journal of Surface Mining, Reclamation and Environment*, vol. 13, no. 1, pp. vii–vii, Jan. 1999, doi: 10.1080/09208119908944193.

[35] H. Hamrin, “Choosing an underground mining method,” in *Underground mining methods handbook*, W. A. Hustrulid, Ed., New York: AIME, 1982.

[36] P. Darling, “SME Underground Mining Handbook,” vol. 1. Society for Mining, Metallurgy, and Exploration (SME), 2023. [Online]. Available: https://app.knovel.com/hotlink/toc/id:kpSMEUMH01/sme-underground-mining/sme-underground-mining

[37] H. Hamrin, “Underground mining methods and applications,” in *Underground mining methods: engineering fundamentals and international case studies*, W. A. Hustrulid and R. L. Bullock, Eds., Littleton: Society for Mining Metallurgy and Exploration, 2001.

[38] H. Harraz, “Sublevel Stoping Underground Mining Methods,” 2014. https://www.researchgate.net/publication/301834296_Sublevel_Stoping_Underground_Mining_Methods (accessed Jun. 10, 2023).

[39] S. Xu, R. Liang, F. T. Suorineni, and Y. Li, “Evaluation of the use of sublevel open stoping in the mining of moderately dipping medium-thick orebodies,” *Int J Min Sci Technol*, vol. 31, no. 2, pp. 333–346, 2021.

[40] N. Morales Varela, D. Mancilla, R. Miranda, and J. Vallejos, “Optimal Drift and Level Design in Underground Mining Extracted by Sublevel Open Stoping Method,” *SSRN Electronic Journal*, 2021, doi: 10.2139/ssrn.3928373.

[41] G. Samanta, S. Sinha, and T. Dey, “A Case Study on Economic Impact Assessment of Developing Single Drill Level in Sublevel Stoping Method,” *Journal of The Institution of Engineers (India): Series D*, 2023, doi: 10.1007/s40033-023-00453-2.

[42] B. H. G. Brady and E. T. Brown, “Rock mechanics and mining engineering,” *Rock Mechanics for underground mining: Third edition*, pp. 1–16, 2006, doi: https://doi.org/10.1007/978-1-4020-2116-9.

[43] K. Kaklis, Z. Agioutantis, M. Masialeti, J. Yendaw, and T. B. Betsi, “Parametric Analysis of Rib Pillar Stability in a Longitudinal Sublevel Open Stoping Operation in an Underground Copper Mine in Southern Africa,” MDPI AG, Nov. 2021, p. 11. doi: 10.3390/materproc2021005011.

[44] Y. A. Sari and M. Kumral, “A planning approach for polymetallic mines using a sublevel stoping technique with pillars and ultimate stope limits,” *Engineering Optimization*, vol. 52, no. 6, pp. 932–944, Jun. 2020, doi: 10.1080/0305215X.2019.1624739.

[45] E. S. Villaescusa, *Geotechnical Design for Sublevel Open Stoping*, 1st ed. CRC Press, 2014. doi: 10.1201/b16702.

[46] A. S. Nhleko and C. Musingwini, “Optimisation of Three-Dimensional Stope Layouts Using a Dual Interchange Algorithm for Improved Value Creation,” *Minerals*, vol. 12, no. 5, May 2022, doi: 10.3390/min12050501.

[47] M. T. Bootsma, C. Alford, J. Benndorf, and M. W. N. Buxton, “Cut-off grade based sublevel stope mine optimisation: Introduction and evaluation of an optimisation approach and method for grade risk quantification,” in *Advances in Applied Strategic Mine Planning*, Springer International Publishing, 2018, pp. 537–557. doi: 10.1007/978-3-319-69320-0_31.

[48] J. J. Sens and E. Topal, “A new algorithm for Stope boundary optimization,” in *The AusIMM Newe Leaders’ Conference, Brisbane, QLD, Australia*, 2009, pp. 1–4.

[49] J. Hou, C. Xu, P. A. Dowd, and G. Li, “Integrated optimisation of stope boundary and access layout for underground mining operations,” *Mining Technology: Transactions of the Institute of Mining and Metallurgy*, vol. 128, no. 4, pp. 193–205, Oct. 2019, doi: 10.1080/25726668.2019.1603920.

[50] C. Alford, “Optimization in underground mine design,” in *25th International Application of Computers and Operations Research in the Mineral Industry symposium*, Brisbane, Australia: AusIMM, 1995, pp. 21–31.

[51] F. Sotoudeh, R. Kakaie, and M. Ataei, “Development of a computer program for underground mine stope optimisation using a heuristic algorithm,” Australian Centre for Geomechanics, 2017, pp. 689–700. doi: 10.36487/acg_rep/1710_58_soutoudeh.

[52] N. Castanho, “Mineable Stope Optimiser (MSO): A Mine Planning Tool More Important Than Ever,” *SRK Publications*, Sep. 2020.

[53] M. Ataee-pour, “A Critical Survey of the Existing Stope Layout Optimization Techniques,” *Journal of Mining Science*, vol. 41, no. 5, pp. 447–466, 2005, doi: 10.1007/s10913-006-0008-9.

[54] A. D. Mwangi, Z. Jianhua, H. Gang, R. M. Kasomo, and M. M. Innocent, “Ultimate Pit Limit Optimization Methods in Open Pit Mines:A Review,” *Journal of Mining Science*, vol. 56, no. 4, pp. 588–602, Jul. 2020, doi: 10.1134/S1062739120046885.

[55] J. Ovanic and D. S. Young, "Economic Optimization of Open Stope Geometry," in *APCOM -INTERNATIONAL SYMPOSIUM-,28th, Computer applications in the minerals industries*, Colorado School of Mines; , 1999, pp. 855–862. [Online]. Available: https://www.tib.eu/de/suchen/id/BLCP%3ACN035562679

[56] M. Ataee-Pour, "A heuristic algorithm to optimise stope boundaries," 2000. [Online]. Available: http://ro.uow.edu.au/theses/2923

[57] A. S. Nhleko, T. Tholana, and P. N. Neingo, "A review of underground stope boundary optimization algorithms," *Resources Policy*, vol. 56, pp. 59–69, 2018, doi: https://doi.org/10.1016/j.resourpol.2017.12.004.

[58] E. Topal and J. Sens, "A new algorithm for stope boundary optimization," *Journal of Coal Science and Engineering*, vol. 16, no. 2, pp. 113–119, Jun. 2010, doi: 10.1007/s12404-010-0201-y.

[59] D. S. S. Sandanayake, E. Topal, and M. W. A. Asad, "Designing an optimal stope layout for underground mining based on a heuristic algorithm," *Int J Min Sci Technol*, vol. 25, no. 5, pp. 767–772, Sep. 2015, doi: 10.1016/j.ijmst.2015.07.011.

[60] M. E. V. Matamoros and M. Kumral, "Heuristic stope layout optimisation accounting for variable stope dimensions and dilution management," *Int J Min Miner Eng*, vol. 8, no. 1, pp. 1–18, 2017.

[61] V. Nikbin, M. Ataee-pour, K. Shahriar, and Y. Pourrahimian, "A Greedy Algorithm for Stope Boundaries Optimization," *Mining Optimization Laboratory (MOL)* , vol. 8, pp. 246–252, 2017.

[62] T. M. Mmola, A. S. Nhleko, and J. M. Atherfold, "Application of Particle Swarm Optimization Algorithm to Optimize Stope Layout for Underground Mines," in *Proceedings of the 27th International Symposium on Mine Planning and Equipment Selection - MPES 2018*, E. Widzyk-Capehart, A. Hekmat, and R. Singhal, Eds., Cham: Springer International Publishing, 2019, pp. 213–221.

[63] Y. A. Sari and M. Kumral, "Clustering-based iterative approach to stope layout optimization for sublevel stoping," *The Journal of the Southern African Institute of Mining and Metallurgy*, vol. 121, 2021, doi: 10.17159/2411.

[64] J. M. Riddle, “A Dynamic Programming Solution of A Block-Caving Mine Layout,” in *14th Application Of Computer Methods In The Mineral Industry*, 1977, pp. 767–780.

[65] E. J. A. Appianing, E. John, A. Appianing, and E. Ben-Awuah, “Underground Mining Stope Layout Optimization and Production Scheduling A Review of Existing Solvers and Algorithms,” 2018.

[66] N. Grieco and R. Dimitrakopoulos, “Managing grade risk in stope design optimisation: Probabilistic mathema-tical programming model and application in sublevel stoping,” *Min. Technology*, vol. 116, no. 2, pp. 49–57, 2007, doi: doi:10.1179/174328607X191038.

[67] X. Bai, D. Marcotte, and R. Simon, “Underground stope optimization with network flow method,” *Comput Geosci*, vol. 52, pp. 361–371, 2013, doi: 10.1016/j.cageo.2012.10.019.

[68] R. W. Eglese, “Simulated annealing: A tool for operational research,” *Eur J Oper Res*, vol. 46, no. 3, pp. 271–281, 1990, doi: https://doi.org/10.1016/0377-2217(90)90001-R.

[69] T. D. Mavridou and P. M. Pardalos, “Simulated Annealing and Genetic Algorithms for the Facility Layout Problem: A Survey,” Kluwer Academic Publishers, 1997.

[70] M. E. Villalba Matamoros and M. Kumral, “Underground mine planning: stope layout optimisation under grade uncertainty using genetic algorithms,” *Int J Min Reclam Environ*, vol. 33, no. 5, pp. 353–370, Jul. 2019, doi: 10.1080/17480930.2018.1486692.

[71] R. L. A. Verhoeff, M. W. N. Buxton, T. U. Delft, A. M. Msc, and R. Aachen, “Using Genetic Algorithms for Underground Stope Design Optimization in Mining A Stochastic Analysis,” 2017. [Online]. Available: http://repository.tudelft.nl/.

[72] J. Hou, G. Li, H. Wang, and N. Hu, “Genetic algorithm to simultaneously optimise stope sequencing and equipment dispatching in underground short-term mine planning under time uncertainty,” *Int J Min Reclam Environ*, vol. 34, no. 5, pp. 307–325, May 2020, doi: 10.1080/17480930.2019.1584952.

[73] G. Nelis, M. Gamache, D. Marcotte, and X. Bai, "Stope optimization with vertical convexity constraints," *Optimization and Engineering*, vol. 17, no. 4, pp. 813–832, Dec. 2016, doi: 10.1007/s11081-016-9321-6.

[74] X. Bai, D. Marcotte, and R. Simon, "A heuristic sublevel stope optimizer with multiple raises," *Journal of the Southern African Institute of Mining and Metallurgy,* , vol. 114, no. 6, pp. 472–434, 2014.

[75] J. Huchette and J. P. Vielma, "A geometric way to build strong mixed-integer programming formulations," *Operations Research Letters*, vol. 47, no. 6, 2019, doi: 10.1016/j.orl.2019.10.003.

[76] B. O. Afum, E. Ben-Awuah, and H. Askari-Nasab, "A mixed integer linear programming framework for optimising the extraction strategy of open pit – underground mining options and transitions," *Int J Min Reclam Environ*, vol. 34, no. 10, pp. 700–724, Nov. 2020, doi: 10.1080/17480930.2019.1701968.

[77] H. Askari-Nasab, Y. Pourrahimian, E. Ben-Awuah, and S. Kalantari, "Mixed integer linear programming formulations for open pit production scheduling," *Journal of Mining Science*, vol. 47, no. 3, p. 338, 2011.

[78] Y. Moysoglou, "Limitations of linear programming as a model of approximate computation," Ph. D. Disssertation, National and Kapodistrian University of Athens (EKPA), 2015.

[79] B. Alhijawi and A. Awajan, "Genetic algorithms: theory, genetic operators, solutions, and applications," *Evolutionary Intelligence*. Springer Science and Business Media Deutschland GmbH, 2023. doi: 10.1007/s12065-023-00822-6.

[80] T. M. Shami, A. A. El-Saleh, M. Alswaitti, Q. Al-Tashi, M. A. Summakieh, and S. Mirjalili, "Particle Swarm Optimization: A Comprehensive Survey," *IEEE Access*, vol. 10, pp. 10031–10061, 2022, doi: 10.1109/ACCESS.2022.3142859.

[81] S. Y. Amponsah, P. M. Takouda, and E. Ben-Awuah, "A Multiple Objective Genetic Algorithm Approach for Stochastic Open Pit Production Scheduling Optimisation," *Int J Min Reclam Environ*, pp. 1–28, Apr. 2023, doi: 10.1080/17480930.2023.2196918.

[82] O. Khare, S. Ahmed, and Y. Singh, "An Overview of Swarm Intelligence-Based Algorithms," in *Design and Applications of Nature Inspired Optimization: Contribution of Women Leaders in the Field*, D. Singh, V. Garg, and K. Deep, Eds., Cham: Springer International Publishing, 2022, pp. 1–18. doi: 10.1007/978-3-031-17929-7_1.

[83] Y. A. Sari and M. Kumral, "Dig-limits optimization through mixed-integer linear programming in open-pit mines," *Journal of the Operational Research Society*, vol. 69, no. 2, pp. 171–182, Feb. 2018, doi: 10.1057/s41274-017-0201-z.

[84] S. Kirkpatrick, C. D. Gelatt, and M. Vecchi, "Optimization by Simulated Annealing," *Science (1979)*, vol. 220, pp. 671–680, 1983.

[85] S. Kirkpatrick, C. D. Gelatt, and M. Vecchi, "Optimization by Simulated Annealing," *Science (1979)*, vol. 220, pp. 671–680, 1983.

[86] S. H. Zhan, J. Lin, Z. J. Zhang, and Y. W. Zhong, "List-Based Simulated Annealing Algorithm for Traveling Salesman Problem," *Comput Intell Neurosci*, vol. 2016, 2016, doi: 10.1155/2016/1712630.

[87] P. P. Tambe, "Selective Maintenance Optimization of a Multi-component System based on Simulated Annealing Algorithm," in *Procedia Computer Science*, Elsevier B.V., 2022, pp. 1412–1421. doi: 10.1016/j.procs.2022.01.342.

[88] A. G. Nikolaev and S. H. Jacobson, "Simulated Annealing," 2010, pp. 1–39. doi: 10.1007/978-1-4419-1665-5_1.

[89] K. Amine, "Multiobjective Simulated Annealing: Principles and Algorithm Variants," *Advances in Operations Research*, vol. 2019, 2019, doi: 10.1155/2019/8134674.

[90] İ. İLHAN, "An improved simulated annealing algorithm with crossover operator for capacitated vehicle routing problem," *Swarm Evol Comput*, vol. 64, p. 100911, Jun. 2021, doi: 10.1016/J.SWEVO.2021.100911.

[91] A. Y. Zomaya, "Natural and simulated annealing," *Comput Sci Eng*, vol. 3, no. 6, pp. 97–99, 2001, doi: 10.1109/MCISE.2001.963434.

[92] M. E. Villalba Matamoros and M. Kumral, “Underground mine planning: stope layout optimisation under grade uncertainty using genetic algorithms,” *Int J Min Reclam Environ*, vol. 33, no. 5, pp. 353–370, Jul. 2019, doi: 10.1080/17480930.2018.1486692.

[93] A. J. Fofanah, H. R. Bundu, and J. G. Kargbo, “generic heart diseases prediction and application of genetic algorithms in healthcare systems,” *Int J Health Sci (Qassim)*, 2022, doi: 10.53730/ijhs.v6ns3.9024.

[94] M. S. Umam, M. Mustafid, and S. Suryono, “A hybrid genetic algorithm and tabu search for minimizing makespan in flow shop scheduling problem,” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 9, 2022, doi: 10.1016/j.jksuci.2021.08.025.

[95] K. Drachal and M. Pawłowski, “A review of the applications of genetic algorithms to forecasting prices of commodities,” *Economies*, vol. 9, no. 1, 2021, doi: 10.3390/economies9010006.

[96] H. Zhi and S. Liu, “Face recognition based on genetic algorithm,” *J Vis Commun Image Represent*, vol. 58, 2019, doi: 10.1016/j.jvcir.2018.12.012.

[97] P. Yarsky, “Using a genetic algorithm to fit parameters of a COVID-19 SEIR model for US states,” *Math Comput Simul*, vol. 185, 2021, doi: 10.1016/j.matcom.2021.01.022.

[98] M. Mitchell, *An introduction to genetic algorithms.*, vol. 8. in Complex adaptive systems., vol. 8. Cambridge, MA, US: The MIT Press, 1996.

[99] T. Alam, S. Qamar, A. Dixit, and M. Benaida, “Genetic algorithm: Reviews, implementations and applications,” *International Journal of Engineering Pedagogy*, vol. 10, no. 6. 2021. doi: 10.3991/IJEP.V10I6.14567.

[100] “A STUDY ON GENETIC ALGORITHM AND ITS APPLICATIONS,” *International Research Journal of Modernization in Engineering Technology and Science*, 2023, doi: 10.56726/irjmets32980.

[101] K. F. Man, K. S. Tang, and S. Kwong, “Genetic algorithms: concepts and applications [in engineering design],” *IEEE Transactions on Industrial Electronics*, vol. 43, no. 5, pp. 519–534, 1996, doi: 10.1109/41.538609.

[102] R. Ingo, “Evolution strategy: Optimization of technical systems by means of biological evolution. Fromman-Holzboog,” *Stuttgart*, vol. 104, p. 15, 1973.

[103] H. Fuger, G. Stein, and U. Stilla, “Multi-population evolution strategies for structural image analysis,” in *Proceedings of the First IEEE Conference on Evolutionary Computation. IEEE World Congress on Computational Intelligence*, 1994, pp. 229–234 vol.1. doi: 10.1109/ICEC.1994.350011.

[104] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of ICNN’95 - International Conference on Neural Networks*, IEEE, pp. 1942–1948. doi: 10.1109/ICNN.1995.488968.

[105] Q. Bai, “Analysis of Particle Swarm Optimization Algorithm,” *Comput. Inf. Sci.*, vol. 3, pp. 180–184, 2010.

[106] H. Ahmed and J. Glasgow, “Swarm intelligence: concepts, models and applications,” *School Of Computing, Queens University Technical Report*, 2012.

[107] T. M. Mmola, A. S. Nhleko, and J. M. Atherfold, “Application of Particle Swarm Optimization Algorithm to Optimize Stope Layout for Underground Mines,” in *Proceedings of the 27th International Symposium on Mine Planning and Equipment Selection - MPES 2018,* Springer International Publishing, 2019, pp. 213–221. doi: 10.1007/978-3-319-99220-4_17.

[108] R. C. Eberhart and Y. Shi, “Guest Editorial Special Issue on Particle Swarm Optimization,” *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 201–203, 2004, doi: 10.1109/TEVC.2004.830335.

[109] E. S. Olivas, J. D. M. Guerrero, M. Martinez-Sober, J. R. Magdalena-Benedito, and L. Serrano, *Handbook of research on machine learning applications and trends: Algorithms, methods, and techniques: Algorithms, methods, and techniques*. IGI global, 2009.

[110] C. P. Lim and S. Dehuri, *Innovations in swarm intelligence*, 2010th ed., vol. 248. Springer Science & Business Media, 2009.

[111] K. E. Parsopoulos and M. N. Vrahatis, "Particle swarm optimization method for constrained optimization problems," *Intelligent technologies–theory and application: New trends in intelligent technologies* , vol. 76, no. 1, pp. 214–220, 2002.

[112] I. Rahman, P. Vasant, B. Singh, and M. Abdullah-Al-Wadud, "On the performance of accelerated particle swarm optimization for charging plug-in hybrid electric vehicles," *Alexandria Engineering Journal*, vol. 55, pp. 419–426, Mar. 2016, doi: 10.1016/j.aej.2015.11.002.

[113] V. R. Kulkarni and V. Desai, "ABC and PSO: A comparative analysis," in *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, IEEE, 2016, pp. 1–7.

[114] R. J. Vanderbei, "LINEAR PROGRAMMING: Foundations and Extensions," *Journal of Operations Research Society*, vol. 49, p. 94, 1998.

[115] H. P. Williams, *Model building in mathematical programming*, 5th ed. John Wiley & Sons, 2013. Accessed: Jun. 11, 2023. [Online]. Available: https://books.google.com/books?id=0tR8tcwYCM4C

[116] G. B. Dantzig and M. N. Thapa, *Linear programming: Theory and extensions*, 2003rd ed., vol. 2. Springer, 2003.

[117] E. K. Chong and H. Ż. Stanislaw, *AN INTRODUCTION TO OPTIMIZATION*, vol. 75. John Wiley & Sons, 2013.

[118] D. Luenberger and Y. Ye, *Linear and Nonlinear Programming*. 2021. doi: 10.1007/978-3-030-85450-8.

[119] A. Alochukwu *et al.*, "stope boundary optimisation for underground mines," *Group (New York)*, 2018.

[120] Gurobi Optimization LLC, "Gurobi Optimizer Reference Manual." 2023. [Online]. Available: https://www.gurobi.com

[121] E. L. Johnson, *Modeling and strong linear programs for mixed integer programming*. Springer, 1989.

[122] L. Wolsey, “Strong formulations for mixed integer programming: a survey,” *Math Program*, vol. 45, pp. 173–191, 1989.

[123] M. Asghari, A. M. Fathollahi-Fard, S. M. J. Mirzapour Al-e-hashem, and M. A. Dulebenets, “Transformation and Linearization Techniques in Optimization: A State-of-the-Art Survey,” *Mathematics*, vol. 10, no. 2, 2022, doi: 10.3390/math10020283.

[124] W. P. Adams and H. D. Sherali, “A tight linearization and an algorithm for zero-one quadratic programming problems,” *Manage Sci*, vol. 32, no. 10, pp. 1274–1290, 1986.

[125] X. He, A. Chen, W. A. Chaovalitwongse, and H. X. Liu, “An improved linearization technique for a class of quadratic 0-1 programming problems,” *Optim Lett*, vol. 6, no. 1, 2012, doi: 10.1007/s11590-010-0249-z.

[126] H. D. Sherali and A. Alameddine, “A new reformulation-linearization technique for bilinear programming problems,” *Journal of Global Optimization*, vol. 2, no. 4, 1992, doi: 10.1007/BF00122429.

[127] B. Rostami and F. Malucelli, “A revised reformulation-linearization technique for the quadratic assignment problem,” *Discrete Optimization*, vol. 14, 2014, doi: 10.1016/j.disopt.2014.08.003.

[128] A. Selvi, D. den Hertog, and W. Wiesemann, “A reformulation-linearization technique for optimization over simplices,” *Math Program*, vol. 197, no. 1, 2023, doi: 10.1007/s10107-021-01726-y.

[129] C. Dapogny, A. Faure, G. Michailidis, G. Allaire, A. Couvelas, and R. Estevez, “Geometric constraints for shape and topology optimization in architectural design,” *Comput Mech*, vol. 59, no. 6, 2017, doi: 10.1007/s00466-017-1383-6.

[130] M. Dagbert, “Review of the Book ‘Mineral Resource Estimation’ by Mario Rossi and Clayton Deutsch, 332p, Published by Springer in Early 2014 (ISBN 978-1-4020-5716-8),” *Math Geosci*, vol. 46, no. 8, pp. 1029–1034, 2014, doi: 10.1007/s11004-014-9547-9.

[131] J. Poniewierski, "BLOCK MODEL KNOWLEDGE FOR MINING ENGINEERS-AN INTRODUCTION." [Online]. Available: www.linkedin.com/in/julianponiewierski/

[132] M. Bustillo Revuelta, "Mineral Resource Evaluation," in *Mineral Resources: From Exploration to Sustainability Assessment*, M. Bustillo Revuelta, Ed., Cham: Springer International Publishing, 2018, pp. 223–309. doi: 10.1007/978-3-319-58760-8_4.

[133] R. L. Lowrie, "SME Mining Reference Handbook," *Society for Mining, Metallurgy, and Exploration (SME)*. Society for Mining, Metallurgy, and Exploration (SME), 2002. [Online]. Available: https://app.knovel.com/hotlink/toc/id:kpSMEMRH03/sme-mining-reference/sme-mining-reference

[134] V. Rudenno, *The mining valuation handbook mining and energy valuation for investors and management*. 4th ed. Milton, Qld. : John Wiley, 2012., 2012. [Online]. Available: https://search.library.wisc.edu/catalog/9911077337702121

[135] A. Esmaeili, J. K. Hamidi, and A. Mousavi, "Determination of sublevel stoping layout using a network flow algorithm and the MRMR classification system," *Resources Policy*, vol. 80, p. 103265, Jan. 2023, doi: 10.1016/J.RESOURPOL.2022.103265.

[136] L. Sipeki, A. M. Newman, and C. A. Yano, "Selecting support pillars in underground mines with ore veins," *IISE Trans*, vol. 52, no. 10, pp. 1173–1188, Oct. 2020, doi: 10.1080/24725854.2019.1699978.

[137] The MathWorks, "MATLAB version: 9.13.0 (R2022b)." The MathWorks Inc., Natick, Massachusetts, United States, 2022. [Online]. Available: https://www.mathworks.com

[138] MathWorks, "Problem-Based Optimization Workflow." [Online]. Available: https://www.mathworks.com/help/optim/ug/problem-based-workflow.html

[139] B. King and A. Newman, "Optimizing the cutoff grade for an operational underground mine," *Interfaces (Providence)*, vol. 48, no. 4, pp. 357–371, 2018.

[140] F. Sotoudeh, M. Nehring, M. Kizil, P. Knights, and A. Mousavi, "A novel cut-off grade method for increasing the sustainability of underground metalliferous mining operations," *Miner Eng*, vol. 172, p. 107168, 2021, doi: https://doi.org/10.1016/j.mineng.2021.107168.

[141] R. Martinelli, J. Collard, and M. Gamache, "Strategic planning of an underground mine with variable cut-off grades," *Optimization and Engineering*, vol. 21, no. 3, pp. 803–849, 2020, doi: 10.1007/s11081-019-09479-6.

VITA

Theophilus Mensah was born in Ghana, West Africa, and received his Bachelor of Science in Mining Engineering from the University of Mines and Technology "UMaT", Tarkwa, Ghana, in June 2016. He moved on to work with Perseus Mining Ghana Limited as a Mining Engineer until July 2021. In July 2021, he moved to the USA where he earned his Master of Science in Mining and Explosives Engineering from the Missouri University of Science and Technology "MST", Rolla in December 2023. Prior to completion, he accepted an offer to work for Freeport McMoran as a Mine Planning Engineer.