

# Underground Stope Optimization

John Manchuk, Oy Leuangthong, Chad Neufeld and C.V. Deutsch

Department of Civil & Environmental Engineering  
University of Alberta

## Abstract

*Current trends in mining show that today's mines are more economically marginal compared to past mineral ventures. With technological advances, mining companies are pursuing deeper deposits that were once economically infeasible, technologically challenging and hazardous to access. Advances in continuous mining equipment in the past few decades reduce the risk to human life while increasing productivity; better equipment and safety practices make access a manageable part of the mineral venture. Rather, it is the economics of the venture, and not the technological limitations that will constrain the depths to which a deposit can be exploited (Hartman, 1987).*

*In the underground mining context, stope optimization becomes a key component of mine planning and design. Proper integration of stope optimization into a mine's planning will result in stopes yielding the highest possible return based on the given exploration data. This note focuses on implementing stope optimization in a manner that will result in the best choice for a stope. There is also an interest in the area of stope optimization and sequencing; however, only insight into this area will be discussed here.*

## Introduction

There are two aims of developing stope optimization software: to tweak stopes already designed and; to provide a feasible starting point for stope design. A program able to optimize and sequence multiple stopes is the desired outcome; however, a process to optimize a single stope will be developed initially.

To effectively optimize a stope, it must be parameterized properly. These parameters can include the initial stopes vertex coordinates as well as limitations determined from stability analysis or the type of mining being employed. Inputting these parameters into an optimization program will ideally result in a stope adhering to limitations and providing maximum return.

## The 'Stope Optimize' Program

A program to optimize a single stope was developed as an initial step towards optimizing and sequencing multiple stopes. The objective in developing this program was to acquire the optimal stope, while meeting specified criteria, as efficiently as possible. The faster a single stope can be optimized, the faster a set of stopes will be.

Currently, the stope optimization program works for stopes defined by 8 vertices in three dimensions. To optimize a stope, a grade model is needed along with initial stope vertex locations. As shown in Figure 1, various limitations can be imposed on the stope as well.

## Parameters for STOPE\_OPTIMIZE

\*\*\*\*\*

### START OF PARAMETERS:

Stope_simulations.out	-File containing stope realizations
initial_stope.dat	-File containing initial stope paramters
1 100 0	-Column for variable, # of realizations, realization choice (0=all)
1 2 3	-Columns for x,y,z stope coordinates
final_stopes.out	-Output file for optimized stope outlines
1 1	-Create a summary file, create a ranking file (1=yes)
stope_summary.out	-File for summary output
profit_distributions.out	-File for profit distributions for p10,50,90 stopes
stope_opt_dbg.dbg	-File for debugging output
0	-Debugging level 0=none,1=normal,2=high
15 0.5 1.0	-x grid definition, nx, xmn, xsiz
30 0.5 1.0	-y grid definition, ny, ymn, ysiz
35 0.5 1.0	-z grid definition, nz, zmn, zsiz
3	-Factor for optimization (cellsize/factor=iteration distance)
5 2.75	-Cutoff grade, specific gravity of rock mass
14.0 0.85 60.0	-Metal price \$/gram, recovery factor, operating cost \$/tonne
3.0	-Minimum allowable mining width in meters
50.0	-Minimum allowable dip in degrees
86668	-Random number seed

**Figure 1:** Parameter file for the initial stope optimization program. This version will optimize a single stope for multiple realizations and produce various statistics for comparison.

If the realization choice option is set to zero, then a final stope will be generated for every realization, otherwise a final stope will be generated for the specified realization. The summary file will include the iteration, tonnes of ore in the stope, recovered ore after processing, average grade, and profit. If a ranking file is to be created, the file name will be the same as the summary file name with 'rank' added to it. This file contains the optimized stope profit for all realizations. The x, y, and z grid definition define the grade model the stope lies within. The factor of optimization defines the distance a node is moved in a particular direction. A factor of 3 indicates a distance equal to one third the cell size in whatever direction the movement is in. If limitations such as minimum mining width and dip are specified, then these parameters will not be exceeded during optimization.

If the initial stope is to be optimized over all realizations of the grade model, an output file containing optimal stope outlines for each realization will be generated along with an expected optimal stope. The expected optimal stope is determined using the expected profit calculated from all realizations of the grade model. Optimizing the initial stope using the  $E\{\text{Profit}\}$  model yields the expected optimal stope.

### Stope Optimization Procedure

A single stope defined by 8 vertices is optimized by the ‘stope optimize’ program using the following procedure:

1. Using the first grade model realization, determine which nodes are within the initial stope and calculate its profit.
2. Pick one of the stope vertices and randomly choose a direction to move it in. If the strike of the deposit is in the  $x$  direction, movement will either be in the  $\pm y$  direction. Stope vertices are chosen in the order they were read in from the input file.
3. Determine which nodes of the grade model are within the updated stope and calculate profit.
4. If the new profit is greater than the initial profit, keep this change, otherwise discard it. Move on to the next vertex (Step 2). Continue this process until no change in any vertex results in a positive change in profit.
5. Move on to the next realization (Step 1).

An efficient method of determining if grid blocks are within a stope in three dimensional space was needed and to solve this, an algorithm used in finite elemental analysis was implemented [1, 2]. A set of eight shape functions (one function per stope vertex) are known for an 8-noded brick and are used to transform coordinates within or on a cube with side length 2 and centered at 0,0,0 to the coordinates of the corresponding point within or on the original 8-noded shape, this being the stope.

$$N_i = \frac{1}{8}(1 + r \cdot r_i)(1 + s \cdot s_i)(1 + t \cdot t_i) \quad 1$$

where  $N_i$  is the shape function for vertex  $i$ ,  $r$ ,  $s$ , and  $t$  are the coordinates of a point with a cube of side length 2 and centered at 0,0,0, and  $r_i$ ,  $s_i$ , and  $t_i$  are the coordinates of vertex  $i$  of the cube. To determine the  $x$ ,  $y$ , and  $z$  coordinates of the corresponding point, the following equation is used:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} N_1 & 0 & 0 & N_2 & 0 & 0 & N_8 & 0 & 0 \\ 0 & N_1 & 0 & 0 & N_2 & 0 & \dots & 0 & N_8 & 0 \\ 0 & 0 & N_1 & 0 & 0 & N_2 & 0 & 0 & 0 & N_8 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ x_2 \\ y_2 \\ z_2 \\ x_8 \\ y_8 \\ z_8 \end{bmatrix} \quad 2$$

where x, y, and z are the coordinates of a point within the stope corresponding to the r, s, and t coordinates of a point within the cube, N<sub>1</sub> to N<sub>8</sub> are the shape functions for the 8 vertexes, and x<sub>1</sub> to x<sub>8</sub>, y<sub>1</sub> to y<sub>8</sub>, and z<sub>1</sub> to z<sub>8</sub> are the coordinates of the stope vertexes.

To describe how the trimming process of determining if grid blocks are in or out of the stope works, a stope striking in the x direction will be referred to. Since the stope strikes in the x direction, movement of the stope vertices will take place only in the y direction. For the trimming process to work, the grid must be the same size as the stope in the fixed dimensions, those being x and z since the stope will not change size in these direction. If the grade model is larger than the input stope, only the required grid blocks will be used during optimization. The cube's faces in r,s,t space that correspond to the xz faces of the stope are discretized into the same number of nodes as the grade model's xz planes. Transforming these values using Equation 2 yields the x,y,z coordinates on the xz stope faces. Since the nodes along the stope face are indexed exactly the same as the grade model's blocks, it is automatically known which set of y coordinates on the stope will be used for comparison. As the grade model is cycled through, blocks can be flagged as in or out and a profit can be calculated.

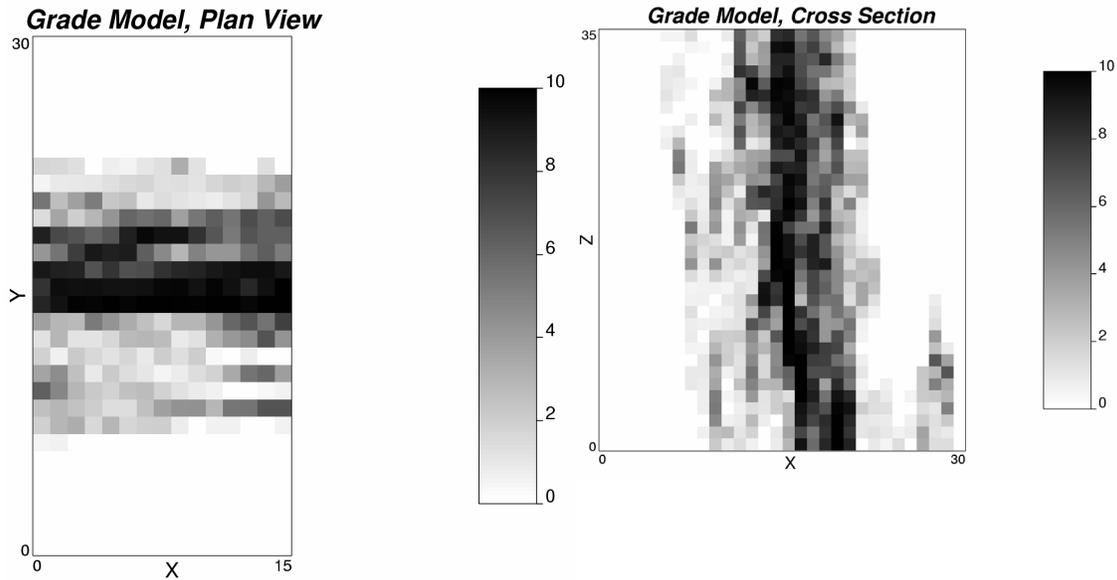
### Implementing the Optimization Program

To test the program, a grade model was generated to represent a vein type deposit, see Figure 2. The grade model had the following grid definition:

Direction	Number	Origin	Size
X	15	0.5	1.0
Y	30	0.5	1.0
Z	35	0.5	1.0

For an initial stope, vertex coordinates were chosen to approximately enclose the high grade zone, see Table 1.

Initially, the stope optimization program was run with one realization while varying the movement distance per iteration. This was done to find an optimal factor for optimization which is set to 3 in Figure 1. A graph was generated showing the progress of optimization and maximum profit achieved, see Figure 3. It was found that a factor for optimization of 3 provided an optimal profit with the least number of iterations.



**Figure 2:** Plan view and cross section of the grade model. This is for one of the 100 realizations generated.

Vertex	X	Y	Z	Vertex	X	Y	Z
1	0.0	10.0	35.0	5	15.0	10.0	35.0
2	0.0	20.0	35.0	6	15.0	20.0	35.0
3	0.0	15.0	0.0	7	15.0	15.0	0.0
4	0.0	25.0	0.0	8	15.0	25.0	0.0

**Table 1:** Vertex coordinates of the initial stope

Running the stope optimization program for all 100 realizations was done next to determine which stope should be taken as the best average. A number of stopes were taken to check this: the p10, p50, and p90 stopes (regarding profit) were taken; an average of all optimal stope vertex locations were taken, and; an optimal stope, found using an E-type profit model calculated from the grade model, was used. A profit was calculated for each selected stope and for each realization to produce profit distributions which are shown in Figure 4.

The stope optimized using the E-type profit grade model shows the highest mean profit value over the realizations, which is expected. It may have been expected that the p50 and E-type stopes should have had similar mean profit values and that the p90 mean profit would be higher than both; however, the maximum mean profit is acquired from the E-type stope. Choosing the E-type stope as optimal would be best.

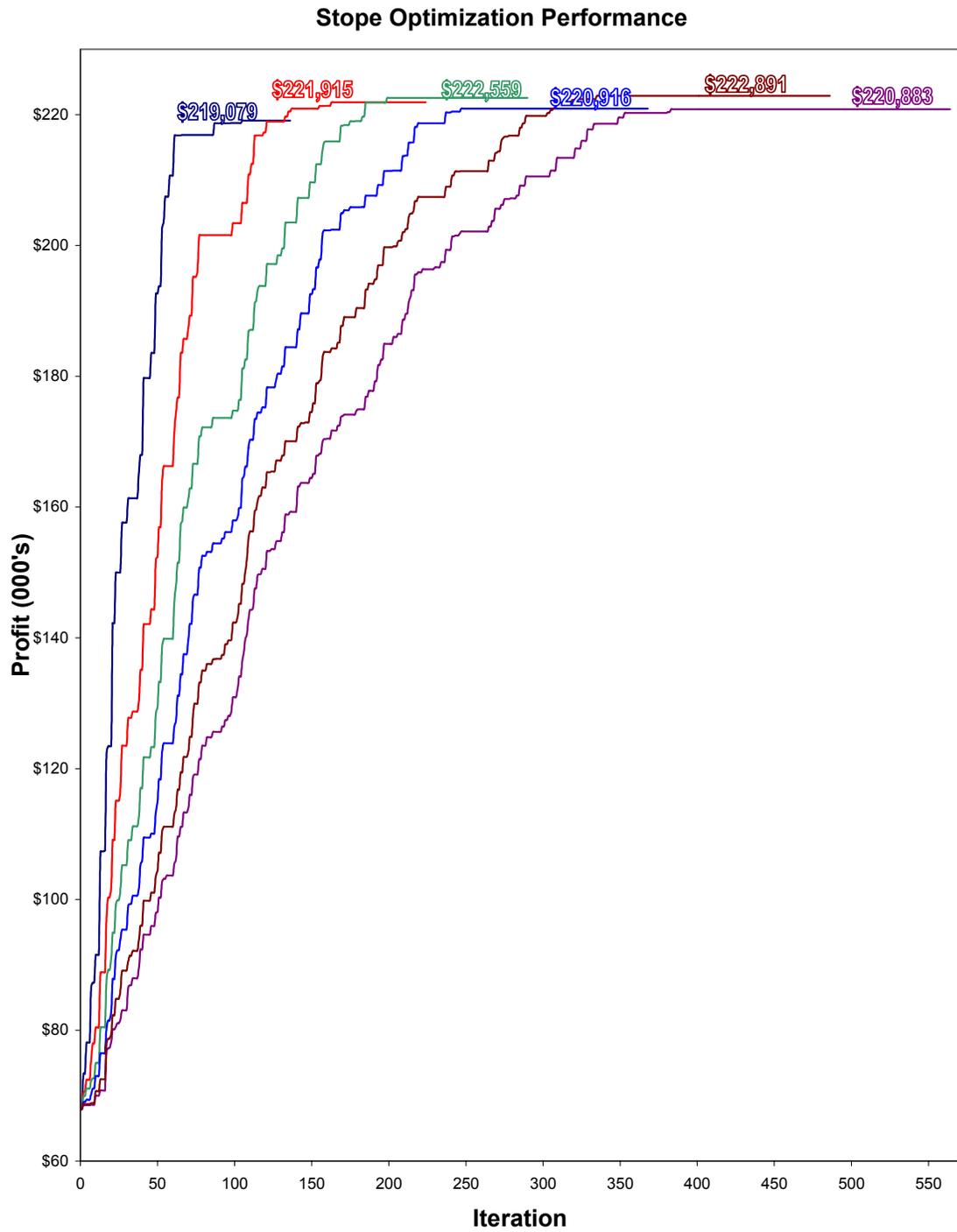
## **Conclusions and Future Work**

Stope optimization is a fairly straightforward process and is easily applied to individual stopes defined by 8 vertices. Parameters such as minimum allowable dip and mining width are easily incorporated. When choosing the most optimal stope to mine, it is best to calculate an E-type profit model from the grade model realizations and optimize the stope based on it. In the face of uncertainty, this stope would be most likely to produce the maximum return.

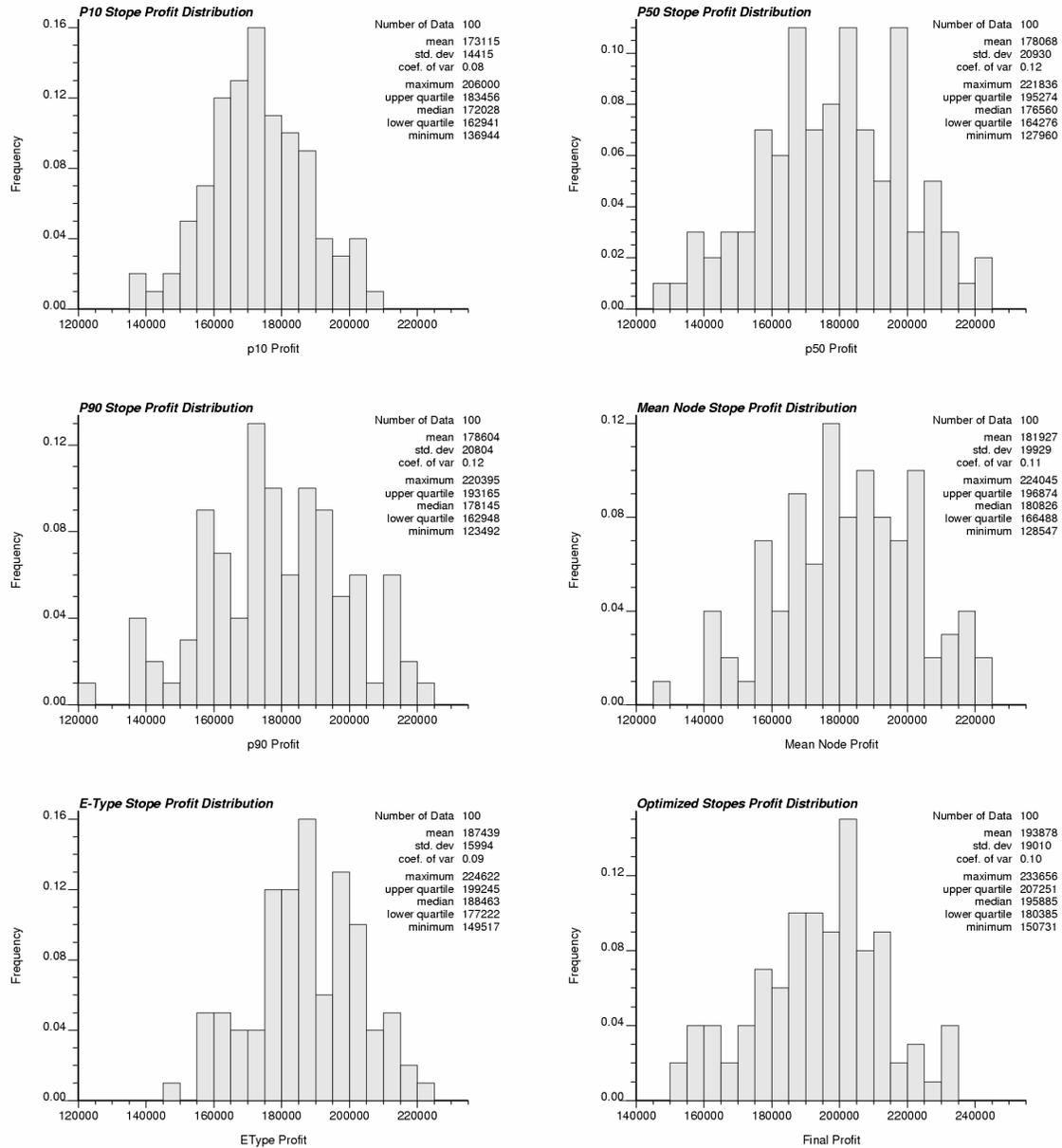
The next step is to develop a program to not only optimize a set of stopes, but to sequence them as well. Since time will be a factor in sequencing, more parameters will be introduced such as cost time curves, price time curves, and production time curves. Introducing these parameters along with others to develop a stope optimization-sequencing program is a sought after outcome. A possible workflow for the program is as follows: (1) generate the E-type profit model; (2) optimize each stope in the given panel using the profit model; (3) sequence the stopes based on such factors as stress, required production, and mining and backfilling time; (4) based on the sequence, which provides the time each stope will be mined, recalculate the E-type profit model using the cost-time/price-time curves; (5) re-optimize the stopes using the updated E-type profit model.

## **References:**

- [1] G.N. Pande, G. Beer, and J.R. Williams, *Numerical Methods in Rock Mechanics*, John Wiley & Sons Ltd., 1990
- [2] W. Wittke, *Rock Mechanics, Theory and Applications with case histories*, Springer-Verlag Berlin Heidelberg, 1990



**Figure 3:** Stope optimization performance as a function of the factor for optimization. Factors used range from 1 (furthest left line) to 6 (furthest right line) incrementing by 1. The shape of the lines is similar since the random number seed was kept constant.



**Figure 4:** Profit distributions for the selected stopes along with the profit distribution for all 100 optimized stopes.