A Greedy Algorithm for Stope Boundaries Optimization

V. Nikbin[1], M. Ataee-pour[1], K. Shahriar[1], and Y. Pourrahimian[2]
[1]Amirkabir University of Technology, Tehran, Iran
[2]Mining Optimization Laboratory (MOL)
University of Alberta, Edmonton, Canada

ABSTRACT

One of the main steps after selecting underground mining method is determining stope boundaries. Obviously, regarding its impacts on mining economy, it should be an optimal plan. However, due to the problem complexities, a comprehensive algorithm has not been reported yet for it. Most of the presented algorithms are heuristic for which the true optimality is not guaranteed. Also, other algorithms whose solutions are seemed to be optimal, either fail to run on 3D problems or their applications are limited to a specific method. This paper introduces a new greedy algorithm. Although this proposed algorithm may fail to provide a globally optimum solution, it is a polynomial time algorithm. After algorithm description, it has been compared to MVN algorithm. Results show that this algorithm may find a higher value solution compared to its alternatives.

1. Introduction

As open pit mines deepen their stripping ratio and consequently their mining costs are increased. In these situations, extracting the rest of deposit by one of the underground mining methods may be more economically. Also, for some of the deposits in which overburdens are large enough, the underground mining methods are our only options. Given one of these underground methods as the suitable method for extracting the ore deposit, it is necessary to determine the optimal workable layout. Clearly, this layout should be as optimal as possible. However, after four decades from the presentation of the first algorithm for optimizing underground mining limit, the growth rate of these algorithms had been slow within this passed period. The majority of presented algorithms are heuristic, as well as, their solutions are suboptimal. Moreover, the presented rigorous algorithms either are not 3D or they have many simplifications.

Open pit and underground optimization algorithms are usually implemented on 2D or 3D economic block models. Indeed, a large cube covers the whole of the ore deposit and part of waste rocks. Afterwards, using the parallel vertical and horizontal planes, the large cube is divided into smaller cubes which are usually regular. Each of these smaller cubes is called a block and owns a specific position and also dimensions. In the next step, using one of the estimation methods, block grade is estimated. By knowing the mining cost, commodity price, and recovery, block economic value (BEV) is calculated.

The main purpose of optimizing underground mining limit is a determination of an appropriate plan for choosing a combination of blocks in the economic models that maximizes the overall profit, subject to some constraints.

Some researchers have been looking at optimization of stope boundaries since four decades ago such as; Riddle (1977), Ovanic and Young (1995, 1999), Alford (1995), Ataee-pour (1997), Jalali and et al (2004), Topal and Sens (2010), Bai and et al (2013, 2014). However, most of the presented algorithms are heuristic, as well as, their solutions are suboptimal. Moreover, the presented rigorous algorithms, either is not 3D or they have many simplifications. Although these heuristic algorithms usually fail to provide an optimal solution they are useful techniques to solve our large-scale problem at a reasonable time.

This paper proposes a new greedy algorithm for stope boundaries optimization. It is a heuristic algorithm and may not find out the global optimum solution. However, it has a great potential for solving such large-sized problem.

2. Proposed Algorithm

Indices, parameters, sets, and variables which are used in this algorithm are defined as below:

Index

t stage number

Parameters

P economic block model

$p_{i,j,k}$ value obtained from extracting block $B_{i,j,k}$

I number of P blocks along i axis

J number of P blocks along j axis

K number of P blocks along k axis

d_i minimum stope size in terms of block along i axis

d_j minimum stope size in terms of block along j axis

d_k minimum stope size in terms of block along k axis

$B_{m,n,o}$ origin block of a probable stope

$B_{\tilde{m},\tilde{n},\tilde{o}}$ origin block of a definitive stope

Sets

M feasible locations of origin blocks along i axis

N feasible locations of origin blocks along j axis

O feasible locations of origin blocks along k axis

$S_{m,n,o}$ blocks which are located in a probable stope

$S_{\tilde{m},\tilde{n},\tilde{o}}$ blocks which are located in a definitive stope

Variables

$\varphi^t_{m,n,o}$ value added to overall stope value from extracting a probable stope at the t^{th} stage

$\varphi^t_{\tilde{m},\tilde{n},\tilde{o}}$ value added to overall stope value from extracting a definitive stope at the t^{th} stage

Λ^t optimal stope boundaries which is the union of all definitive stopes at the t^{th} stage

λ^t overall stope value at the t^{th} stage

Ω^t non-investigated stopes set

Θ^t pre-investigated stopes set

2.1. Probable and Definitive Stopes

A probable stope is a subset of P and denoted by $S_{m,n,o}$. This subset creates a cube that its dimensions along i, j and k axis are fixed and they are pre-known parameters. These dimensions are also named as minimum stope sizes. Each cube is determined by an origin block ($B_{m,n,o}$, yellow blocks in Fig. 1).

Since $S_{m,n,o}$ satisfies minimum stope size constraints, it has a potential to be chosen as part of the final layout. However, based on the problem's objective function (it can be maximization of profit, NPV, etc) the algorithm might reject it. Therefore, it is called a probable stope. When a probable stope is decieded to be included in the optimal solution, its name changes to definitive stope. The relationship between a probable stope and its blocks is presented in Eq (1).

$$S_{m,n,o} = \{B_{i,j,k} \mid i \in \{m,\ldots,m+d_i-1\}, j \in \{n,\ldots,n+d_j-1\}, k \in \{o,\ldots,o+d_k-1\}\} \quad (1)$$

Also, feasible locations of origin blocks along i, j and k axis are defined by M, N and O sets (Eqs. (2) and (4)).

$$M = \{1,2,\ldots,I-d_i+1\} \quad (2)$$

$$N = \{1,2,\ldots,J-d_j+1\} \quad (3)$$

$$O = \{1,2,\ldots,K-d_k+1\} \quad (4)$$

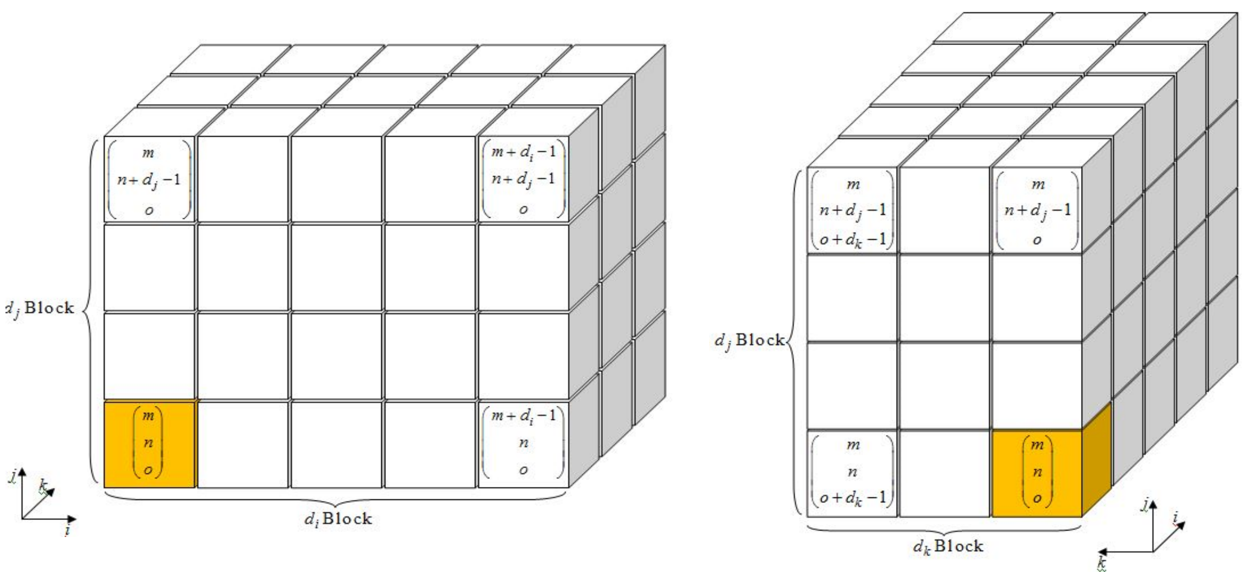

Fig. 1. A 3D view of a probable stope

2.2. Greedy Algorithm

Based on the type of objective function, e.g. maximization of profit, a greedy algorithm at each stage selects the best solution without considering its previous and future decisions. Obviously, these approaches may lead to a local and near-optimal solution. Although they do not guarantee a global solution, they are very fast and useful algorithms to solve any large-scale problem at a polynomial time.

In this paper, it is tried to propose a new greedy algorithm for optimization of stope boundaries considering minimum stope sizes. The main core of this algorithm is to find the most valuable probable stope between non-investigated stopes set by Eq. (5). This stope is called definitive stope and it is added to the optimal stope boundaries set (Λ^t) if its value is positive (Eq. (6)). Since the algorithm checks any stopes only once, any definitive stope is removed from non-investigated stope sets (Ω^t), Eq. (7), and is added to pre-investigated stope sets (Θ^t), Eq. (8).

$$\varphi^t_{\tilde{m},\tilde{n},\tilde{o}} = \max\{\varphi^t_{m,n,o} \,|\, B_{m,n,o} \in \Omega^{t-1}\} \qquad (5)$$

Where:

$$\varphi^t_{m,n,o} = \sum_{(i,j,k)\in(S_{m,n,o}-\Lambda^{t-1})} p_{i,j,k}$$

$$\Lambda^t = \Lambda^{t-1} \bigcup S_{\tilde{m},\tilde{n},\tilde{o}} \qquad (6)$$

$$\Omega^t = \Omega^{t-1} - B_{\tilde{m},\tilde{n},\tilde{o}} \qquad (7)$$

$$\Theta^t = \Theta^{t-1} \bigcup B_{\tilde{m},\tilde{n},\tilde{o}} \qquad (8)$$

At the beginning of algorithm, pre-investigated stope set (Θ^0) and also the stope boundaries set (Λ^0) are empty, because the algorithm has not found any definitive stope yet. However, the initial non-investigate stope set (Ω^0) consists of all feasible origin blocks (Eq. (9)).

$$\Omega^0 = \{B_{i,j,k} \,|\, i \in M, j \in N, k \in O\} \qquad (9)$$

Also, the overal stope value at each stage (λ^t) is determined by Eq. (10).

$$\lambda^t = \lambda^{t-1} + \varphi^t_{\tilde{m},\tilde{n},\tilde{o}} \qquad (10)$$

The pseudo code of this algorithm has been illustrated in Fig. 2.

3. Numerical Example

Let us suppose that there is a section of the economic block model with six rows and ten columns (Fig. 3), and minimum stope sizes along i and j axis to be equal to three and one blocks, respectively. Based on the following calculations the greedy algorithm determines a layout with a value of 63 (Fig. 4). However, the MVN algorithm determines a layout with a value of 59 when its search direction is from the right side to left side (Fig. 5).

algorithm *greedy algorithm for stope boundaries optimization*;

begin

1) $t := 1$;

2) $\Omega^0 := \{B_{i,j,k} \mid i \in M, j \in N, k \in O\}$;

3) $\Theta^0 := \varnothing$;

4) $\Lambda^0 := \varnothing$;

5) $\lambda^0 := 0$;

6) find a stope with the maximum value among the non-investigated stopes:

$\varphi^t_{\tilde{m},\tilde{n},\tilde{o}} = \max\{\varphi^t_{m,n,o} \mid B_{m,n,o} \in \Omega^{t-1}\}$ and $B_{\tilde{m},\tilde{n},\tilde{o}} \in \Omega^{t-1}$

7) remove this stope from the non-investigated stopes set (Ω^t) and add it to pre-investigated stopes set (Θ^t):

$\Omega^t = \Omega^{t-1} - B_{\tilde{m},\tilde{n},\tilde{o}}$

$\Theta^t = \Theta^{t-1} \bigcup B_{\tilde{m},\tilde{n},\tilde{o}}$

While $\varphi^t_{\tilde{m},\tilde{n},\tilde{o}}$ is positive **do**

begin

a) update stope boundaries (Λ^t): $\Lambda^t = \Lambda^{t-1} \bigcup S_{\tilde{m},\tilde{n},\tilde{o}}$

b) update overall stope value (λ^t): $\lambda^t = \lambda^{t-1} + \varphi^t_{\tilde{m},\tilde{n},\tilde{o}}$

c) $t := t + 1$;

d) go to step 6

end;

end;

Fig.2. Pseudo code of the greedy algorithm

i \ j	1	2	3	4	5	6	7	8	9	10	11	12
1	2	1	-1	0	3	2	-2	4	1	2	-2	-1
2	5	-1	-1	2	3	-2	1	0	1	3	-1	-1
3	3	0	4	1	-2	-1	0	-1	2	1	-1	2
4	-1	0	-2	1	2	0	4	-1	1	2	2	-1
5	6	-1	-3	1	0	-3	5	3	1	-4	2	0

Fig. 3. The economic block model example

$$\boldsymbol{t=1:}(\tilde{m},\tilde{n})=(5,7),\varphi^{1}_{(5,7)}=9,\Theta^{1}=\{B_{5,7}\},\Lambda^{1}=\{S_{5,7}\},\lambda^{1}=9$$

$$\boldsymbol{t=2:}(\tilde{m},\tilde{n})=(1,8),\varphi^{2}_{(1,8)}=7,\Theta^{2}=\{B_{5,7},B_{1,8}\},\Lambda^{2}=\{S_{5,7},S_{1,8}\},\lambda^{2}=16$$

$$\boldsymbol{t=3:}(\tilde{m},\tilde{n})=(3,1),\Phi^{3}_{(3,1)}=7,\Theta^{3}=\{B_{5,7},B_{1,8},B_{3,1}\},\Lambda^{3}=\{S_{5,7},S_{1,8},S_{3,1}\},\lambda^{3}=23$$

.

.

.

$$\boldsymbol{t=17:}(\tilde{m},\tilde{n})=(5,2),\Phi^{17}_{(5,2)}=1,$$
$$\Theta^{17}=\{B_{5,7},B_{1,8},B_{3,1},B_{4,5},B_{1,4},B_{4,9},B_{2,3},B_{2,8},B_{2,1},B_{1,1},B_{3,9},B_{5,1},B_{3,10},B_{2,7},B_{3,2},B_{4,4},B_{5,2}\},$$
$$\Lambda^{17}=\{S_{5,7},S_{1,8},S_{3,1},S_{4,5},S_{1,4},S_{4,9},S_{2,3},S_{2,8},S_{2,1},S_{1,1},S_{3,9},S_{5,1},S_{3,10},S_{2,7},S_{3,2},S_{4,4},S_{5,2}\},$$
$$\lambda^{17}=63$$

$$\boldsymbol{t=18:}(\tilde{m},\tilde{n})=(1,2),\Phi^{18}_{(1,2)}=0,\text{it is not a positive number}\rightarrow\text{End}$$

i \ j	1	2	3	4	5	6	7	8	9	10	11	12
1	2	1	-1	0	3	2	-2	4	1	2	-2	-1
2	5	-1	-1	2	3	-2	1	0	1	3	-1	-1
3	3	0	4	1	-2	-1	0	-1	2	1	-1	2
4	-1	0	-2	1	2	0	4	-1	1	2	2	-1
5	6	-1	-3	1	0	-3	5	3	1	-4	2	0

Fig. 4. The greedy algorithm output with value of 63

i \ j	1	2	3	4	5	6	7	8	9	10	11	12
1	2	1	-1	0	3	2	-2	4	1	2	-2	-1
2	5	-1	-1	2	3	-2	1	0	1	3	-1	-1
3	3	0	4	1	-2	-1	0	-1	2	1	-1	2
4	-1	0	-2	1	2	0	4	-1	1	2	2	-1
5	6	-1	-3	1	0	-3	5	3	1	-4	2	0

Fig. 5. MVN output with value of 59

4. Conclusion

Undoubtedly, underground mining limit optimization is one of the main mining problems which play an important role in the industrial mining economy. However, after four decades since the first algorithm has been presented for this problem, there is no complete algorithm yet. Most of them are heuristic and their solutions are not guaranteed. Also, due to many simplifications in designing rigorous algorithms, they are not appropriate to solve real case problems. In this research, a new greedy algorithm was introduced. Although it may provide a locally optimum solution, it is able to find the solution at a reasonable time. Actually, less processing time for running this algorithm is very important due to our hardware limitations.

5. References

[1] Alford, C. (1995). Optimisation in Underground Mine Design. in *25th International Symposium on the Application of Computers and Operations Research in the Mineral Industry*: The Australasian Institute of Mining and Metallurgy, pp. 213-218.

[2] Ataee-pour, M. (1997). *A New Heuristic Algorithm to Optimise Stope Boundaries*. in Proceedings of APCOM Symposium on Computer Applications in the Mineral Industry, Moscow,

[3] Bai, X., Marcotte, D., and Simon, R. (2013). Underground stope optimization with network flow method. *Computers and Geoscience*, *52* 361-371.

[4] Bai, X., Marcotte, D., and Simon, R. (2014). A Heuristic Sublevel Stope Optimizer With Multiple Raises. *Southern African Institute of Mining and Metallurgy*, *114* 427-434.

[5] Jalali, S. E., Ataee-pour, M., and Shahriar, K. (2004). A 2D Dynamic Programming Algorithm to Optimize Stope Boundary. in *Mine Planning and Equipment Selection*, pp. 45-52.

[6] Ovanic, J. and Young, D. (1995). Economic Optimisation of Stope Geometry Using Separable Programming with Special Branch and Bound Techniques. in *Third Canadian Conference on Computer Applications in the Mineral Industry*. Quebec, pp. 129-135.

[7] Ovanic, J. and Young, D. (1999). Economic Optimisation of Open Stope Geometry. in *28th International APCOM Symposium*. Colorado: Colorado school of Mines, pp. 855-862.

[8] Riddle, J. (1977). A Dynamic Programming Solution of a Block-Caving Mine Layout. in *14th International Symposium on the Application of Computers and Operations Research in the Mineral Industry*. Colorado: Society for Mining, Metallurgy and Exploration, pp. 767-780.

[9] Topal, E. and Sens, J. (2010). A new algorithm for stope boundary optimization. *Coal Science & Engineering*, *16* (2), 113-119.