STOPE BOUNDARY OPTIMISATION FOR UNDERGROUND MINES

Supervisor: Prof. Montaz Ali

Alex Alochukwu, Babatunde Sawyerr, John Atherfold, Krupa Prag, Micheal Olusanya, Patience Adamu, Peter Popoola, Sakirudeen Abdulsalaam, Vincent Langat

January 19, 2018

GROUP 3: MATHEMATICS IN INDUSTRY STUDY GROUP (MISG) 2018
The Stope Boundary Optimization Problem (SBOP) involves choosing a certain configuration of stopes which maximizes the Net Profit Value, (NPV), subject to the stope dimension constraints.
Our approach involves the following:

- The development of a mathematical model for the SBOP
- The exploration of various optimization algorithms for the SBOP
- The development of a hybrid algorithm for the SOP which contains components of Dynamic Programming (DP) and Particle Swarm Optimization (PSO).
## 2D Schematic of Stope Configuration Instance

<table>
<thead>
<tr>
<th></th>
<th>15</th>
<th>0</th>
<th>3</th>
<th>-8</th>
<th>7</th>
<th>32</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>74</td>
<td>1</td>
<td>19</td>
<td>27</td>
<td>-4</td>
<td></td>
</tr>
<tr>
<td>-22</td>
<td>10</td>
<td>90</td>
<td>2</td>
<td>-71</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>60</td>
<td>32</td>
<td>-40</td>
<td>3</td>
<td>-20</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>1</td>
<td>-13</td>
<td>-44</td>
<td>22</td>
<td>85</td>
<td></td>
</tr>
<tr>
<td>-22</td>
<td>33</td>
<td>64</td>
<td>2</td>
<td>-9</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>
Mathematical Modelling

For the sake of simplicity, we developed the mathematical formulation in 2D, and extend it to 3D. The following assumptions were made in creating the 2D mathematical model:

- Let the mining area be represented by a grid with dimension, \( n \times m \).
- The grid is made up of distinct blocks with predefined values.
- The stope dimension is fixed for 2D case, say \( \alpha \times \beta \).
- The decision variable is binary.
- To ensure that the stopes are on the same level for easy mining, we use the following strategy: If \( x_{ij} = 0 \), then move to \( x_{ij+1} \); If \( x_{ij} = 1 \), then move to \( x_{ij+\beta} \); Once the level has been exhausted, move to \( x_{i+\alpha j} \), and repeat the steps.
MATHEMATICAL FORMULATION: 2D

Maximize \[
\sum_{i=1}^{n-p} \sum_{j=1}^{m-q} V_{ij} x_{ij},
\] (1)

Subject to:

\[
\sum_{i}^{i+p} \sum_{j}^{j+q} x_{ij} \leq 1, \forall i \in \{1, \cdots n - p\}, \forall j \in \{1, \cdots m - q\} \quad (2)
\]

\[
x_{ij} - \sum_{j' = j+1}^{j+q} x_{ij'} = 1 \quad \forall i \in \{1, \cdots n - p\}, \forall j \in \{1, \cdots m - q\} \quad (3)
\]

\[
x_{ij} - \sum_{i' = i+1}^{i+p} x_{i'j} = 1 \quad \forall i \in \{1, \cdots n - p\}, \forall j \in \{1, \cdots m - q\} \quad (4)
\]
MATHEMATICAL FORMULATION: 2D

\[ x_{ij} - \sum_{i' = i+1}^{i+p} \sum_{j' = j+1}^{j+q} x_{i'j'} = 1 \quad \forall \ i \in \{1, \ldots n - p\}, \forall \ j \in \{1, \ldots m - q\} \] (5)

where \[ V_{ij} = \sum_{i}^{i+p} \sum_{j}^{j+q} u_{ij}, \quad p = \alpha - 1 \text{ and } q = \beta - 1. \quad x_{ij} \in \{0, 1\} \]
Next, we extend this model to the 3D SBOP, allowing for variable stope dimensions:

Maximize \( \sum_{i=1}^{n-p} \sum_{j=1}^{m-q} \sum_{k=1}^{s-r} V_{ijk} x_{ijk} \), \hspace{1cm} (6)

Subject to:

\[ \sum_{i}^{i+p} \sum_{j}^{j+q} \sum_{k}^{k+r} x_{ijk} \leq 1, \forall i \in \{1, \cdots n-p\}, \forall j \in \{1, \cdots m-q\}, \forall k \in \{1, \cdots s-r\} \]

\[ x_{ijk} - \sum_{j'=j+1}^{j+q} x_{ij'k} = 1 \hspace{1cm} \forall i \in \{1, \cdots n-p\}, \forall j \in \{1, \cdots m-q\}, \forall k \in \{1, \cdots s-r\} \]
MATHEMATICAL FORMULATION: 3D

\[ X_{ijk} - \sum_{i'=i+1}^{i+p} X_{i'jk} = 1 \quad \forall \ i \in \{1, \ldots n-p\}, \forall \ j \in \{1, \ldots m-q\}, \forall \ k \in \{1, \ldots s-r\} \]

\[ X_{ijk} - \sum_{k'=k+1}^{k+r} X_{ijk'} = 1 \quad \forall \ i \in \{1, \ldots n-p\}, \forall \ j \in \{1, \ldots m-q\}, \forall \ k \in \{1, \ldots s-r\} \]

\[ X_{ijk} - \sum_{i'=i+1}^{i+p} \sum_{j'=j+1}^{j+q} \sum_{k'=k+1}^{k+s} X_{i'j'k'} = 1 \quad \forall \ i \in \{1, \ldots n-p\}, \forall \ j \in \{1, \ldots m-q\}, \forall \ k \in \{1, \ldots s-r\} \]

\[ \forall \ k \in \{1, \ldots s-r\} \]
MATHEMATICAL FORMULATION: 3D

\[ X_{ijk} - \sum_{i'=i+1}^{i+p} \sum_{j'=j+l}^{m-1} \sum_{k'=k+1}^{s-1} X_{i'j'k'} = 1 \quad \forall \ i \in \{1, \ldots n-p\}, \forall \ j \in \{1, \ldots m-q\}, \]

\[ \forall \ k \in \{1, \ldots s-r\} \]

where \( V_{ijk} = \sum_{i}^{i+p} \sum_{j}^{j+q} \sum_{k}^{k+r} u_{ijk} \), \quad \text{(7)}

\( p = \alpha - a \), \( q = \beta - b \) and \( r = \gamma - 1 \) \quad \text{(8)}

\( x_{ijk} \in \{0, 1\} \) \quad \text{(9)}

\( a = 1, \ldots, n \), \( b = 1, \ldots, m \). \quad \text{(10)}
We have been able to successfully implement and test the following heuristics for the SBOP:

- Maximum value algorithm
- Multi-Start Algorithm
- DP-inspired Heuristic
- Particle Swarm Optimisation
This heuristic is inspired by the Dynamic Programming idea of sub-dividing a large problem into smaller ones, solving them, and then combining the solutions to get a solution to the large version of the problem. The algorithm is as follows:

- **Input**: Array containing the value of each block, stope size, mining site dimensions. Let \( K \) be the stope size, \( fd \), the width of the site, and \( fl \) the length of the site.
- **STEP 1**: Create an array \((fd - k + 1 \times fl - k + 1)\) that stores all the possible stopes.
- **STEP 2**: Get the value of each possible stope.
- **STEP 3**: Get all the possible configurations with their values, skipping stopes with negative values.
- **STEP 4**: Pick the stope layout with the highest revenue.
STEP 1: Calculate the $V_{ij}$ by summing the $u_{ij}$ for each possible stope.

STEP 2: Demarcate the whole orebody into levels of size equal to the height of the stope.

STEP 3: In each level determine the possible number of stopes that can be extracted based on their values (A stope with a maximum value chosen first, followed by a stope with the second maximum value, and so on, ensuring that there is no stope overlap.

STEP 4: Determine the the Net Present Value (NPV) by summing up the $V_{ij}$ for all the possible stopes that can be extracted.
## BLOCK VALUES AT EACH CELL

\[
\begin{pmatrix}
-20 & -20 & 2 & 7 & 9 & -1 & 4 & -5 & 7 \\
9 & 6 & 3 & 2 & 3 & -13 & 23 & 56 & -21 \\
1 & -2 & 4 & 5 & 7 & -40 & 0 & -11 & 51 \\
3 & 0 & 5 & 4 & -60 & 30 & 14 & 1 & 31 \\
-4 & -7 & 6 & 3 & -10 & 4 & 12 & 14 & 34 \\
3 & 0 & 7 & 2 & -23 & 2 & 4 & 22 & -15 \\
-4 & -7 & 8 & 1 & 12 & 12 & 3 & 1 & 11 \\
5 & 2 & 6 & 2 & 3 & 0 & 1 & -50 & -20 \\
11 & 8 & 0 & -1 & 17 & -42 & 11 & 0 & 17 \\
17 & 0 & 3 & 4 & 23 & -2 & 2 & 65 & 23 \\
-3 & -6 & 2 & -60 & 18 & 11 & 1 & 54 & 12 \\
8 & 5 & 6 & 3 & 1 & 0 & 3 & 3 & 2 \\
10 & 7 & -7 & -2 & 18 & 4 & 14 & 9 & -57 \\
-8 & -11 & 0 & -50 & 12 & 18 & -45 & 21 & 7
\end{pmatrix}
\]
### STOPE VALUES AT EACH CELL

$$
\begin{pmatrix}
-25 & -9 & 14 & 21 & -2 & 13 & 78 & 37 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2 & 7 & 18 & -44 & -63 & 4 & 4 & 72 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-8 & 6 & 18 & -28 & -27 & 22 & 52 & 55 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-4 & 9 & 17 & 18 & 27 & 16 & -45 & -58 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
36 & 11 & 6 & 43 & -4 & -31 & 78 & 105 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
4 & 7 & -49 & -38 & 30 & 15 & 61 & 71 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-2 & -11 & -59 & -22 & 52 & -9 & -1 & -20 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 
\end{pmatrix}
$$
MULTI-START ALGORITHM

For $k$ iterations

Randomly select a block

Blocks used?

Yes

No

Perform Local Search

Improves Global Value?

Reject

Accept

Update Global Value
### Max Search Multi-Start

<table>
<thead>
<tr>
<th></th>
<th>Case A</th>
<th>Case B</th>
<th>Case C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global Searches</td>
<td>14</td>
<td>2 * 14</td>
<td>14^2</td>
</tr>
<tr>
<td>Local Searches</td>
<td>9</td>
<td>2*9</td>
<td>9^2</td>
</tr>
<tr>
<td>Average Stopes</td>
<td>11.2</td>
<td>11.6</td>
<td>11.6</td>
</tr>
<tr>
<td>Average Global Value</td>
<td>471.4</td>
<td>493.8</td>
<td>484.0</td>
</tr>
<tr>
<td>Stope Fraction</td>
<td>0.36</td>
<td>0.37</td>
<td>0.37</td>
</tr>
</tbody>
</table>

### Random Search Multi-Start

<table>
<thead>
<tr>
<th></th>
<th>Case A</th>
<th>Case B</th>
<th>Case C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global Searches</td>
<td>14</td>
<td>2 * 14</td>
<td>14^2</td>
</tr>
<tr>
<td>Local Searches</td>
<td>9</td>
<td>2*9</td>
<td>9^2</td>
</tr>
<tr>
<td>Average Stopes</td>
<td>2.6</td>
<td>2.6</td>
<td>2.4</td>
</tr>
<tr>
<td>Average Global Value</td>
<td>112.8</td>
<td>78.8</td>
<td>74.6</td>
</tr>
<tr>
<td>Stope Fraction</td>
<td>0.08</td>
<td>0.08</td>
<td>0.07</td>
</tr>
</tbody>
</table>
• Is a greedy modification beneficial?

**Figure**: MS Evaluation using Max Search and Random Search
Figure: MS Evaluation using Max Search and Random Search
· Modification of Pure Random MS algorithm improves solutions.
· Number of search iterations influences the outcome.
· Possible improvement implementations:
  · Parallel MS
  · Alternate local search techniques
Simulates the motion of flocking birds
- *Population* - Initial positions of each particle in the swarm
- $P^i_k$ - Personal best position of the $i^{th}$ particle after the $k^{th}$ time step
- $g_k$ - Global best position of all particles at $k^{th}$ time step.

$Population_{k+1} = Population + Vel_{k+1}$

$Vel_{k+1} = \omega \times vel_k + c_1 \times r_1 \times (P^i_k - Pop) + c_2 \times r_2 \times (g_k - Pop)$
PSO - Encoding Details

- Stope fraction = $\frac{\text{No. of stopes used}}{\text{Total number of allowed stopes}}$.
- Stope fraction was fixed - binary decision matrices were initiated.
- 1000 population members - 1000 matrices
- Each member of the population was a set of co-ordinates representing the ones in their respective matrix.
- $\text{Fitness} = \text{Config Value} - k \times (\text{Overlap Penalty}) - k \times (\text{Level Penalty})$
PSO - But does it work? - Yes

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-20</td>
<td>-20</td>
<td>2</td>
<td>7</td>
<td>9</td>
<td>-1</td>
<td>4</td>
<td>-5</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>6</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>-13</td>
<td>23</td>
<td>56</td>
<td>-21</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>-2</td>
<td>4</td>
<td>5</td>
<td>7</td>
<td>-40</td>
<td>0</td>
<td>-11</td>
<td>51</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>5</td>
<td>4</td>
<td>-60</td>
<td>30</td>
<td>14</td>
<td>1</td>
<td>31</td>
<td></td>
</tr>
<tr>
<td>-4</td>
<td>-7</td>
<td>6</td>
<td>3</td>
<td>-10</td>
<td>4</td>
<td>12</td>
<td>14</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>7</td>
<td>2</td>
<td>-23</td>
<td>2</td>
<td>4</td>
<td>22</td>
<td>-15</td>
<td></td>
</tr>
<tr>
<td>-4</td>
<td>-7</td>
<td>8</td>
<td>1</td>
<td>12</td>
<td>12</td>
<td>3</td>
<td>1</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>6</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>-50</td>
<td>-20</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>8</td>
<td>0</td>
<td>-1</td>
<td>17</td>
<td>-42</td>
<td>11</td>
<td>0</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>0</td>
<td>3</td>
<td>4</td>
<td>23</td>
<td>-2</td>
<td>2</td>
<td>65</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>-3</td>
<td>-6</td>
<td>2</td>
<td>-60</td>
<td>18</td>
<td>11</td>
<td>1</td>
<td>54</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>5</td>
<td>6</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>7</td>
<td>-7</td>
<td>-2</td>
<td>18</td>
<td>4</td>
<td>14</td>
<td>9</td>
<td>-57</td>
<td></td>
</tr>
<tr>
<td>-8</td>
<td>-11</td>
<td>0</td>
<td>-50</td>
<td>12</td>
<td>18</td>
<td>-45</td>
<td>21</td>
<td>7</td>
<td></td>
</tr>
</tbody>
</table>
• **Constant** stope size was considered (2 x 2)
• Various stope fractions were considered.
• For each stope fraction, five experiments were run, with 100 iterations per experiment.
• For each iteration, the maximum fitness value of that population is taken and stored.
• After each experiment, the mean maximum population values were considered, for every iteration.
• These results are presented for various stope fractions.
Different configuration values for different stope fractions.
Upon convergence, 0.4→worst, 0.48→best.
Is there a relationship between stope fraction and mine value? Consider mine 2...
Upon convergence, 0.84→worst, 0.76→best.
No clear relationship - maybe some intrinsic properties of each mine that dictate optimal stope fraction (BEV distribution?)
Looking at the data slightly differently...
- Variation of maximum values as a function of mass fraction
- Since these curves are completely uncorrelated, the BEV distribution per mine plays a large role in stope configuration design.
Hybrid Algorithm

The pseudo-code for the hybrid algorithm which we have developed is as follows:

- **Input:** Mining site dimensions, array containing mining data with BEVs for each block, stope dimensions.
- **STEP 1:** For \( i = 1 \) to \( \text{swarmSize} \)
  - \( dpSolution = \text{DPH()} \)  swarm.add(\( dpSolution \))
- **STEP 2:** \( \text{PSO(swarm)} \)
  - Output \( \text{PSO.gbest} \) as best solution
  - Output \( \text{swarm} \) as set of alternative solutions.
In conclusion, we have been able to successfully achieve the following:

- Develop Mathematical model for the 2D SBOP
- Develop Mathematical model for the 3D SBOP
- Develop and test DP-esque heuristic, Multi-Start, and PSO

What we haven’t been able to do (for shortness of time):

- Extend solution methods to 3D case
- Implement hybrid DP-esque and PSO algorithm