Clustering and Multi-Destination Production Scheduling

Mohammad Tabesh and Hooman Askari-Nasab
Mining Optimization Laboratory
University of Alberta, Edmonton, Canada

Abstract

Mine production planning is the process of deciding on the order of extraction of blocks from the block model. This can be done in various ways, but a known and reliable approach is through mathematical programming. The goal of this paper is to propose a mathematical model which can handle multiple destinations with different capacities, profits, mining and haulage costs, and processing costs. In order to make the problem tractable, two clustering techniques are proposed and compared based on their accuracy and resource consumption. Three procedures for removing unnecessary variable are also proposed, which together eliminate around 70% of variables in a test dataset.

1. Introduction[1]

Mining is a complex industry with large capital investment requirements. Critical decisions are made at different stages in the exploration, design, extraction and reclamation of mines; these deal with the movement of millions of tonnes of materials and cash flows in the order of billions of dollars. Therefore, operations research has been broadly used to assist with decision making processes in the mining industry.

Open pit mine planning is commonly defined as the process of deciding on the sequence of extracting blocks from a mine in order to gain the highest net present value (NPV) subject to a set of constraints. This process is usually undertaken based on two main approaches. The first approach is in two stages: 1) decide which blocks to extract (finding final pit limits); and 2) find the appropriate sequence of extraction (scheduling). In the second approach both problems are solved simultaneously. Although the second approach seems to result in a better solution with respect to maximization of the objective function, Caccetta and Hill (2003) have proven that the optimum pit found using the simultaneous approach will fall inside the pit outline found using pit optimization techniques. Therefore, the first approach is taken for this paper.

Open pit production scheduling is a known problem which has attracted many researchers. This problem has been studied in various levels of detail and various time horizons. Production scheduling models are usually categorized into long-term, mid-term, and short-term time horizons. One of the main issues with various mathematical models proposed for open pit production scheduling in the literature is the size of the problem. A typical midsize open pit mine block model consists of hundreds of thousands of blocks at least. The mathematical formulation of the production scheduling of such a mine is beyond the ability of today's commercial software and

[1] Some parts of introduction and literature review are direct excerpts from (Tabesh and Askari-Nasab, 2011)

hardware to solve the optimization problem in a reasonable time or to find a solution to the problem at all. Attempts have been made to overcome this curse of dimensionality, but one of the main obstacles of using exact optimization methods in open pit mine production scheduling is still the intractability of such formulations.

The first objective of this paper is to develop a mathematical formulation for multi-destination production scheduling of a mine. The proposed formulation is a mixed-integer linear programming model which uses continuous variables for determining the portion of block to each destination and binary variables for controlling the order of extraction of blocks and slope constraints. This means that for every block in the model there are T binary and $T \times D$ continuous variables, where T and D represent the number of planning periods and destinations respectively. For a real size block model this can lead to a mathematical formulation with tens of millions of variables – a situation which is not tractable using currently available hardware and software.

Clustering algorithms are then proposed in order to make the problem tractable and also to have more practical production plans from the mining point of view. The clustering algorithms aggregate blocks into selective mining units based on rock types, ore grades and distances between blocks. The aggregated blocks are referred to as mining-cuts, which are consequently used in the aforementioned mathematical formulation. Two known clustering algorithms are developed and compared based on the variability of grades and rock types inside cuts, the NPV that results when the algorithm is used with the multi-destination model, and the shape of the created clusters. The first algorithm is a hierarchical clustering technique proposed in Tabesh and Askari-Nasab (2011) which defines a similarity index between blocks and aggregates blocks together until it reaches a predefined number of clusters. The next clustering algorithm is an implementation of the k-means algorithm based on kernel functions. The term *k-means*, first used in MacQueen (1967), also covers a large series of heuristics proposed in subsequent literature, all of which partition data points by selecting mean points for clusters and assigning each data point to the nearest mean.

The next part of this paper presents some efforts made to reduce the number of variables and consequently the size of the problem. In many mathematical formulations, paying attention to the structure of the problem can lead to pre-determining some variable values and removing them from the model. Three variable reductions are proposed in this paper which together can remove 73% of variables on a test problem.

The rest of the paper is organized as follows. A review on the long-term open pit mine production planning (LTOPP) literature is followed by a review of the relevant clustering techniques and implementations in the next section. The third section explains the mathematical model and the clustering algorithm proposed. The algorithm is then implemented on a small subset of a real dataset and the results are presented in section four. The paper is concluded in the fifth section.

2. Review of literature

2.1. Long-term open pit mine production planning

Various models for LTOPP have been proposed in the literature. The traditional approach was to define production levels (in tonnages) of ore and waste (Gershon, 1983). Other models of interest consist of zero-one decision variables indicating whether a block is going to be extracted in a period, subject to some constraints on mining and processing capacities, grade blending, and slope constraints. Usually the objective function aims at maximizing the NPV in order to take the time value of money into account. For a complete review of the mathematical programming models used in mine production scheduling, see Osanloo et al. (2008) and Newman et al. (Newman et al., 2010). In this literature survey we will focus on block aggregation and clustering techniques.

The first attempts to cluster blocks and reduce the size of problem were made by (Busnach et al. (1985) and Klingman and Phillips (1988); the authors of these studies decided to aggregate all the blocks on the same bench without considering any of the block properties. Gershon and Murphy (1989) then tried to form layers of material labeled as ore or waste. Samanta et al. (2005) used the same approach but solved the problem using a genetic algorithm. Another interesting approach for grouping blocks and reducing the number of binary variables is taken in Ramazan (2001). The author of this thesis creates a tree based on the blocks in the model and their extraction sequence constraints. Ramazan (2001) then extracts the fundamental trees and solves the problem using these trees. The same author has refined the approach in Ramazan (2007). Although this can lead to optimal solution of the problem, finding the fundamental trees and solving the problem using them is still NP-Hard. Zhang (2006) uses a genetic algorithm to create mining cuts and solve the problem in cut level.

2.2. Clustering

Clustering is defined as the process of grouping similar entities together in a way that maximum intra cluster similarity and inter cluster dissimilarity is achieved. This can be modeled and solved as a mathematical programming problem, but the difficulty lies in the amount of resources and time required to solve the problem. The clustering problem has been proven to be NP-Hard (Gonzalez, 1982). Therefore, a wide range of non-exact algorithms has been developed in the literature. These algorithms are usually performed by defining a measure of similarity or dissimilarity between the objects. These techniques can be categorized into two major groups: hierarchical and partitional clustering. As its name implies, hierarchical clustering is performed by creating a hierarchy of clusters. On the other hand, partitional clustering is performed by partitioning data objects into a number of groups. Hierarchical clustering is known to result in better clusters than partitional algorithms, but by taking more CPU time (Feng et al., 2010). A famous example of partitional algorithms is the k-means, which attempts to find cluster means and assign data points to the closest mean. K-means is a partitioning technique which tries to find a good partitioning scheme by iteratively modifying the partitions. This has also proven to be an NP-Hard problem (Mahajan et al., 2010), but it can still be used to find good partitions on the data. Some recent modified versions of k-means clustering can be found in Chung and Lin (2006), Bagirov (2008) and Zalik (2008). This algorithm has also been combined with other methods to form hybrid faster or more accurate techniques (Chang et al., 2009), (Niknam and Amiri, 2010) and (Niknam et al., 2010). A more complete review of the algorithm and its extensions can be found in Jain (2010). One extension to k-means clustering which is relevant to this project is the kernel k-means which is developed to be able to partition data points which are not linearly separable by mapping them into a kernel space (Dhillon et al., 2004).

3. Mathematical formulation

The mixed integer linear programming model for multi destination mine planning is explained in this section. The model consists of continuous variables for extracting material and sending it to various destinations as well as binary decision variables for controlling the sequence of extractions and slope stabilities. Decision variable $x_k^{d,t} \in [0,1]$ is used in determining the portion of block/cut k in period t and sending the material to destination d . Therefore, $x_k^{d,t}$ has to be bounded by the rock tonnage in block/cut k that can be sent to the corresponding destination. This is done through defining the set R_d and constraints (6) and (7). Binary decision variables are also used in the model in order to control the order of extraction of blocks/cuts. b_k^t is set to one, if extraction of block/cut k is allowed in period t; i.e., all of the predecessor blocks/cuts are completely removed. This is guaranteed through constraint (8). Parameter L in this constraint is the number of members in set $C_k(L)$ which holds all the predecessor blocks/cuts of the block/cut k .When b_k^t gets a value of one, it remains at one until the end of the mine life. On the other hand, mining is prevented through constraint (10) as long as b_k^t is equal to zero.

The objective function of the model is the net present value (NPV) of extracting material from the mine and sending them to the corresponding destinations. The block value earned from sending one tonne of rock from each block/cut is calculated and discounted to the present value and is called $v_k^{d,t}$. This parameter holds all the profits and costs such as mining, haulage, dumping and processing of one ton of rock from the corresponding block/cut. Since haulage and dumping costs are considered, this value can also be different for different waste destinations.

Constraint set (2) is responsible for controlling the minimum and maximum mining capacity in each year. There is also a set of constraints for controlling the capacity of each processing destination in each period, as in constraint (3). Waste dump capacities are controlled through constraint (4). The limitation on waste dumps is usually the available lease area or the area prepared for dumping material. Therefore, having these as cumulative capacity constraints is more reasonable than having a yearly upper and lower bound. Average head grades of elements and deleterious material sent to each processing destination is also controlled by constraint (5). Finally, constraint (11) makes sure that all the material in the pit is extracted and sent to an appropriate destination.

The rest of parameters and sets used in the mathematical formulation are defined as follows.

3.1. Sets

D_P	Set of indices of the processing destinations
D_W	Set of indices of the waste dumps
$C_k(L)$	For each mining-cut k, there is a set $C_k(L)$ defining the immediate predecessor mining-cuts that must be extracted prior to extracting mining-cut k, where L is an integer number presenting the total number of cuts in the set $C_k(L)$.
R_d	For each destination d, R_d holds the rock types that are allowed to be sent to this destination if the there is enough capacity.

3.2. Indices

$k \in \{1,2,...,K\}$	Index for the mining cuts
$t \in \{1,2,...,T\}$	Index for the planning periods
$d \in \{1,2,...,D\}$	Index for the destinations
$e \in \{1,2,...,E\}$	Index for the elements of interest

3.3. Parameters

$v_k^{d,t}$	The discounted value (Net Present Value) of extracting one tonne of cut k in period t and sending it to destination d. If $d \in D_P$ the value is expected to be positive; otherwise, it is expected to be negative.
t_k	The total tonnage of material in mining-cut k
t_k^r	The tonnage of rock type r in mining-cut k
$\underline{m}^t$	The lower bound of mining capacity in period t
$\overline{m}^t$	The upper bound of mining capacity in period t

$\underline{p}^{d,t}$	The lower bound of processing capacity of destination $d \in D_p$ in period t
$\overline{p}^{d,t}$	The upper bound of processing capacity of destination $d \in D_p$ in period t
$\underline{w}^{d,t}$	The lower bound on waste dump cumulative capacity for destination $d \in D_W$ from the first period to period t
$\overline{w}^{d,t}$	The upper bound on waste dump cumulative capacity for destination $d \in D_W$ from the first period to period t t
g_k^e	The grade of element e in mining-cut k
$\underline{g}^{e,d,t}$	The lower bound of the average head grade of element e for destination $d \in D_p$ in period t
$\overline{g}^{e,d,t}$	The upper bound of the average head grade of element e for destination $d \in D_p$ in period t

3.4. Decision Variables

$x_k^{d,t} \in [0,1]$	Continuous variable, representing the portion of mining-cut k extracted and sent to processing destination $d \in D_p$, in period t
$b_k^t \in \{0,1\}$	Zero-one variable for controlling the extraction precedence. It is equal to one if it is allowed to extract mining-cut k in period t. If this variable gets the value of one it will remain at one until the end of the mining period.

3.5. Objective Function

$$\max \sum_{t=1}^{T} \sum_{d=1}^{D} \sum_{k=1}^{K} v_k^{d,t} \times t_k \times x_k^{d,t} \tag{1}$$

3.6. Constraints

$$\underline{m}^t \le \sum_{k=1}^{K} \left(t_k \times \sum_{d=1}^{D} x_k^{d,t} \right) \le \overline{m}^t \qquad \forall t \tag{2}$$

$$\underline{p}^{d,t} \le \sum_{k=1}^{K} t_k \times x_k^{d,t} \le \overline{p}^{d,t} \qquad \forall d \in D_P, t \tag{3}$$

$$\underline{w}^{d,t} \le \sum_{t'=1}^{t} \sum_{k=1}^{K} t_k \times x_k^{d,t'} \le \overline{w}^{d,t} \qquad \forall d \in D_W, t \tag{4}$$

$$\underline{g}^{e,d,t} \le \frac{\sum_{k=1}^{K} t_k \times g_k^e \times x_k^{d,t}}{\sum_{k=1}^{K} t_k \times x_k^{d,t}} \le \overline{g}^{e,d,t} \qquad \forall e, d \in D_P, t \tag{5}$$

$$\sum_{t=1}^{T} x_k^{d,t} \le \frac{\sum_{r \in R_d} t_k^r}{t_k} \qquad \forall k, d \in D_P \tag{6}$$

$$\sum_{t=1}^{T} x_k^{d,t} + \sum_{t=1}^{T} x_k^{d',t} \le \frac{\sum_{r \in R_D \cup R_{D'}} t_k^r}{t_k} \qquad \forall k,d,d' \tag{7}$$

$$b_k^t \le \frac{1}{L} \sum_{d=1}^{D} \sum_{l \in C_k} x_k^{d,t} \qquad \forall k,t \tag{8}$$

$$b_k^t \le b_k^{t+1} \qquad \forall k,t \tag{9}$$

$$\sum_{d=1}^{D} x_k^{d,t} \le b_k^t \qquad \forall k,t \tag{10}$$

$$\sum_{t=1}^{T} \sum_{d=1}^{D} x_k^{d,t} = 1 \qquad \forall k,t \tag{11}$$

4. Clustering

The first proposed clustering algorithm in this paper is based on a hierarchical approach presented in Tabesh and Askari-Nasab (2011). In this approach, there is a need for a similarity index to be defined in order to define similarities between objects. It is defined based on grade difference, distance, rock type and the beneath bench clustering scheme. Then the clusters are formed in a hierarchical manner, as the name implies. Each block is considered as one cluster at the beginning of the algorithm. In each step, the two most similar clusters are merged together and form a new cluster. Then the similarities between the newly created cluster and all others are updated based on one of these techniques: single-link, complete-link or average-link. The single-link approach takes the similarity between the two most similar objects from each cluster as the similarity between the two clusters. In contrast, the complete-link approach looks for the two most dissimilar objects. The average-link method, as its name implies, averages all of the pair similarities. In this paper, the update is done based on the complete-link approach. The procedure continues until a predefined number of clusters are achieved. The main contributions by Tabesh and Askari-Nasab (2011) are the definition of the similarity index and calibration of parameter weights, forcing the algorithm to respect maximum cluster size and to prevent non-adjacent clusters from merging together. For more details readers are referred to Tabesh and Askari-Nasab (2011).

The next clustering technique used is an implementation of the k-means algorithm based on gradient descent search. In this approach, K initial cluster centers are randomly selected at each replication. Then the objects are assigned to the nearest center. Afterwards, based on the gradient descent search technique, the centers are manipulated in such a way that the summation of distances between the objects and the means is locally minimized. Another replication is then started with a new random set of means and the process continues for a limited number of replications.

The first step for this algorithm is to form the feature matrix, which holds all the important properties of all objects. To be consistent with the hierarchical clustering technique, the same parameters are used with the same weighting approach. Then the matrix has to be kernelized in order to get better results. When objects are not linearly separable in their initial space, kernel functions are used to map data points from the initial space to the kernelized space and do the clustering in there. Then the same map is used in returning to the initial space with all the objects labeled as belonging to various clusters. Having tested various kernel functions and parameters, a polynomial kernel function with $d = 1$ is used in this implementation. Afterwards, K initial cluster centers are randomly selected in the kernelized space and objects are assigned to the closest mean. Then the objective function, which is a summation of Euclidean distances between all objects and cluster means, is calculated. Cluster means are then manipulated in an iterative manner based on gradient descent until a local minimum is found. This is stored as a solution to the clustering problem and a new replication starts with another random definition of cluster means. Finally, all of the replications are compared, and the one with the lowest objective function is selected as the solution to the clustering problem on that bench.

5. Variable reduction techniques

5.1. Clustering

As mentioned earlier, clustering brings two advantages to the problem. The first is that it reduces the number of variables in the model, since only one variable is defined for each group of blocks referred to as mining-cuts. The second advantage of clustering lies in having bigger units for planning, which makes the resulted long-term production plan more practical from the mining point of view.

5.2. Removing unnecessary variables

Since the model is built to deal with multi-destination production planning, a variable is used for sending a portion of each cut to some (but not all) destinations. As mentioned above, clustering seeks to have cuts with homogenous rock types; i.e., most of the blocks grouped together have the same rock type. Therefore, it can be determined based on R_D (the possible rock type destination combinations) which cuts cannot be sent to which destinations. The corresponding variables can be removed without making any change to the optimality of the final solution. The only exception is for a waste destination called general waste dump which accepts all types of material. For example, ore rock types can be sent to this destination if the processing capacity is insufficient to let them into processing destinations.

5.3. Predecessor cone

Another technique used is borrowed from Bley et al. (2010), who remove extraction decision variables for certain blocks based on their availability. To do so, the whole tonnage of rock in the predecessor cone of the block is calculated. This is the cone which has to be extracted before one can get access to the block based on slope constraints. The cone tonnage is then compared against the cumulative mining capacity from the first period up to period t. If the cone tonnage exceeds this cumulative mining capacity, the binary and consequently continuous variable for extracting that cut in that period is removed from the problem, because one can already be sure the variables cannot have values other than zero. The idea can be expanded to processing capacities too, i.e. by comparing the ore tonnage in the cone to the cumulative processing capacity. However, this makes the decision a bit tricky since there are multiple destinations and rocks that can be sent to different destinations if it results in higher objective function value. Therefore, this approach is not used in this paper.

5.4. Successor cone

Since there is an assumption that all the material has to be extracted from the mine, the same idea can be applied to the cone below each block. This is the set of blocks which can only be extracted after the block is completely extracted. In this case, if the total tonnage in the cone below the block exceeds the cumulative mining capacity from period t to the end of mine life, the block has to be totally extracted prior to period t. The two-cones concept is also used in a meta-heuristic approach based on simulated annealing in Kumral and Dowd (2005). The predecessor and successor cones are shown in Fig 1 with green dashed and blue dotted filling patterns respectively.

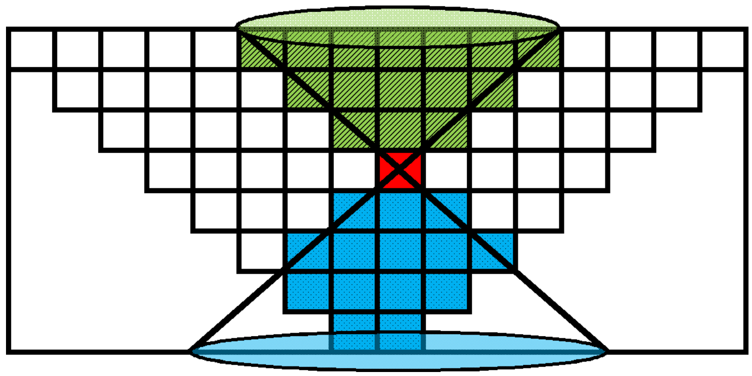

Fig 1.Predecessor and Successor Cones

6. Results

Three different experiments are undertaken to verify the ideas proposed in this paper. The first experiment is designed in order to compare the two clustering techniques based on the homogeneity of the created clusters as well as runtimes. This is done on an iron ore block model with more than 19,000 blocks. The goal of the second experiment is to study the effects of the variable reduction techniques. The same dataset is clustered with hierarchical clustering and variable reduction techniques are applied after forming mathematical programming matrices. In the third experiment, the clustering techniques are used to form mining-cuts as the input to the multi-destination mathematical model, and the resulting NPVs and required CPU times are compared for different techniques as well as different cluster sizes. Since running the multi-destination mathematical model is time- and resource-consuming, a small subset of the model with 2413 blocks from the bottom of the pit is selected and the optimal production plan for 11 periods is determined.

6.1. Hierarchical versus K-Means

The test dataset is an iron ore block model with 19,561 blocks which are distributed in 12 benches. Three elements are tracked in the model: iron, which is tracked as magnetic weight recovery (MWT), and sulfur and phosphor, which are tracked as deleterious material. The properties of the block model are summarized in Table 1. The most significant fact from this pre-processing is the high correlation between Sulfur and Phosphor grades and the MWT grade. Hence, Sulfur and Phosphor grades are removed from the feature vector.

Table 1.Block model summary

	MWT Grade	S Grade	P Grade
MIN	0	0	0
Max	0.83	0.0380	0.0039
Mean	0.11	0.0023	0.0002
STDev	0.27	0.0056	0.0005
Range	0.83	0.0380	0.0039
Correlation with MWT grade		0.96	0.97

After running both clustering algorithms on the dataset, the techniques are compared based on the following measures. As mentioned earlier, it is required to have mining-cuts homogenous in grade and rock types. Consequently, the resulting mining-cut scheme is evaluated using two separate measures for grade and rock type, which are numerical and categorical attributes respectively. It is common to evaluate numerical attributes by the use of squared error criterion (Hsu et al., 2007). This is the square distance of each data point from the cluster mean. Since there is only one numerical attribute having an effect on the quality of the results, this distance is the same as the standard deviation of that attribute. Therefore, the first criterion is the average coefficient of variation (CV) of block grades in each mining-cut, which is a normalized version of the standard deviation. Mining-cuts with lower grade variation are more effective in practice. The same thing applies to the rock type distribution. Since it is a categorical variable, neither standard deviation nor CV can be used. One measure used for categorical attributes is the categorical utility which is the probability of having two objects from same category in one cluster (Hsu et al., 2007). Another measure is the relative frequency of categories in clusters. This measure puts more weight on rare categories if considered as the objective of clustering (Huang, 1997). Based on the structure of the problem and existing criteria for categorical variables, a new index is defined as the percentage of blocks in a mining-cut belonging to the most dominant rock type in that mining-cut[2]. This is called the rock unity and is depicted in Table 2 along with other comparison measures. The hierarchical clustering technique takes more time to run but provides more accurate results, as can be seen in Table 2. The shape of the created clusters is also of importance. Therefore, the clustering schemes of one bench in the middle of the block model are provided for comparison in Fig 2 and Fig 3.

Table 2.Comparing clustering techniques

Technique	CPU Time (s)	Number of Clusters	Average Rock Unity (%)	Average MWT CV (%)	Average S CV (%)	Average P CV (%)
Hierarchical	119	402	94.6	0.3	0.008	0.0008
K-Means	67	399	67.4	0.4	1.8	1.2

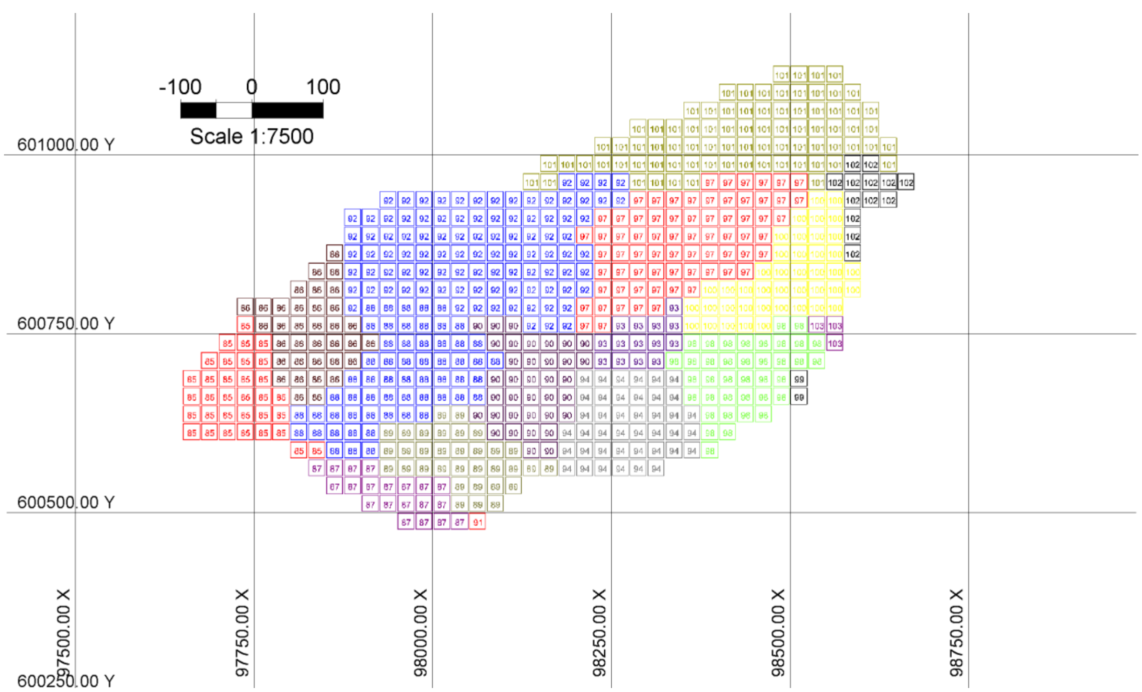

Fig 2.Hierarchical Clustering

2 This part is a direct excerpt from (Tabesh and Askari-Nasab, 2011)

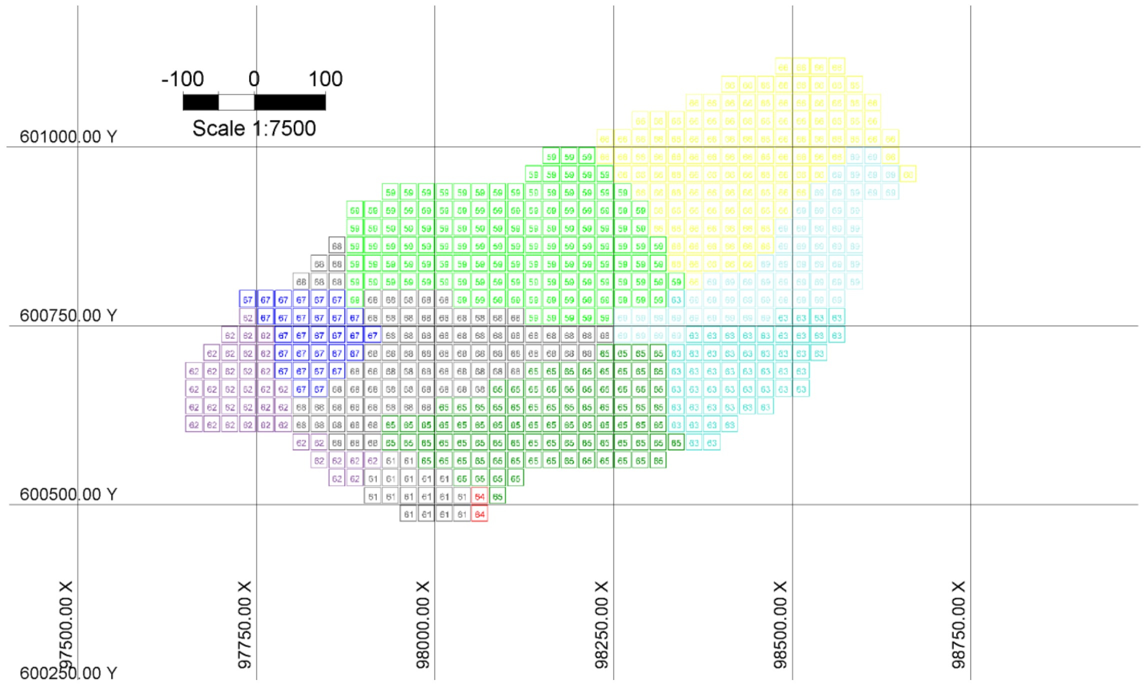

Fig 3.K-Means Clustering

6.2. Variable reduction

The same iron ore data set clustered with hierarchical clustering is used to test the effectiveness of the variable reduction techniques. A more accurate study on the effects of predecessor cone and some other techniques can be found in Bley et al. (2010). There are 19,561 blocks in the model and 402 mining-cuts. The number of binary and continuous variables for a model with 3 destinations over 21 periods is presented in Table 3.

Table 3.Effectiveness of variable reduction techniques

	Number of variables	**Number of continuous variables**	**Number of binary variables**
Original block model	1,643,124	1,232,343	410,781
Clustered model	33,768	25,326	8,442
Clustered model after variable reduction	9,147	4,863	4,284

6.3. Mathematical formulation

The multi-destination mathematical formulation is also tested on the same dataset. However, solving the MILP with available hardware and software is a resource- and time-consuming operation. Therefore, a small subset from the bottom of the pit with 2413 blocks on 6 benches is extracted and clustered with hierarchical clustering with various cluster sizes. The optimal schedule for each cluster size is obtained and compared in Table 4. The scheduling is to be done for 11 periods. The total rock tonnage in the dataset is 81 million tonnes, which is made up of 56 million tonnes of rock type 101 and 1.5 million tonnes of rock type 301, and the rest of which is waste. Lower and upper bounds of 5 and 8 million tonnes per year are considered for mining capacity. Two processing destinations are considered, the first of which is the main processing plant. The second destination is a backup process which has lower capacity and lower efficiency but can be used if the first process is working at full capacity. Rock type 101 is the main ore and can be processed in either of the destinations, unlike rock type 301, which can only be economically processed in the second destination. The processing capacities are considered to be 5 and 1 million tons per year, respectively, for the two processing destinations. For simplification, cumulative capacity constraints are not considered for the only waste dump in the model. The production schedule and the stripping ratio for the four settings are presented in Fig 4 to Fig 7. Blue bars represent the production schedule at the main processing plant, while green bars represent that of the second processing plant. Material sent to the waste dump is shown in brown. Generated schedules for a sample bench are also presented in Fig 8 to Fig 11. It can be seen that by having bigger cuts, the CPU time required for finding the optimal solution and the NPV drop. However, by increasing blocks per cut from 25 to 50, the CPU time is reduced by 99% while the NPV decreases by only 0.4%. On the other hand, the practicality of the generated schedule is one of the most important factors for judging on various clustering options. A higher number of drop-cuts, as can be seen in Fig 8, is a usual drawback of using smaller planning units. On the other hand, higher resolution clustering results in smoother schedule, as shown in Fig 4.

Table 4.Mathematical Modeling Summary

#	Average Blocks per Cut	Number of Cuts	Coefficient Matrix Size	Number of Binaries	CPU Time (s)	NPV (million dollars)
1	25	115	3345*2806	881	41,372	2501.8
2	50	62	1741*1312	407	42.26	2490.7
3	75	43	1195*816	245	6.64	2479.0
4	100	33	931*573	170	3.13	2461.6

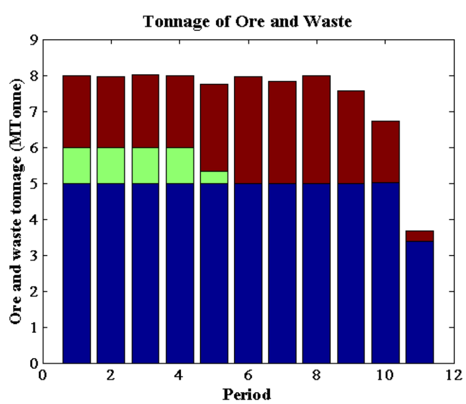

Fig 4.Production Schedule #1

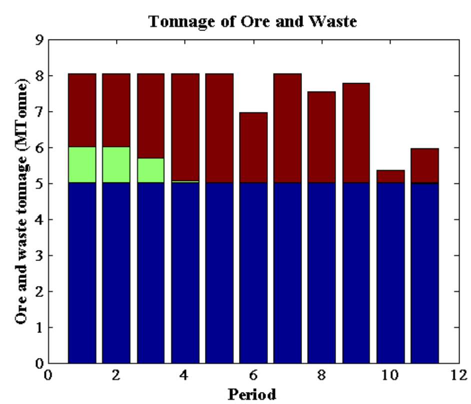

Fig 5.Production Schedule #2

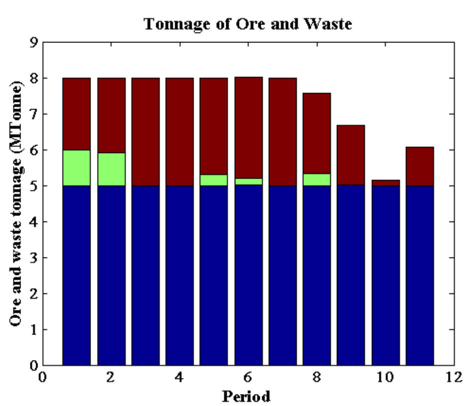

Fig 6.Production Schedule #3

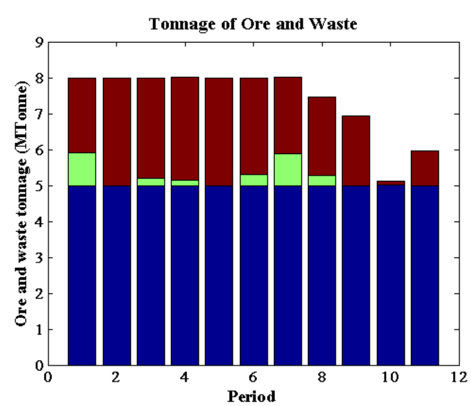

Fig 7.Production Schedule #4

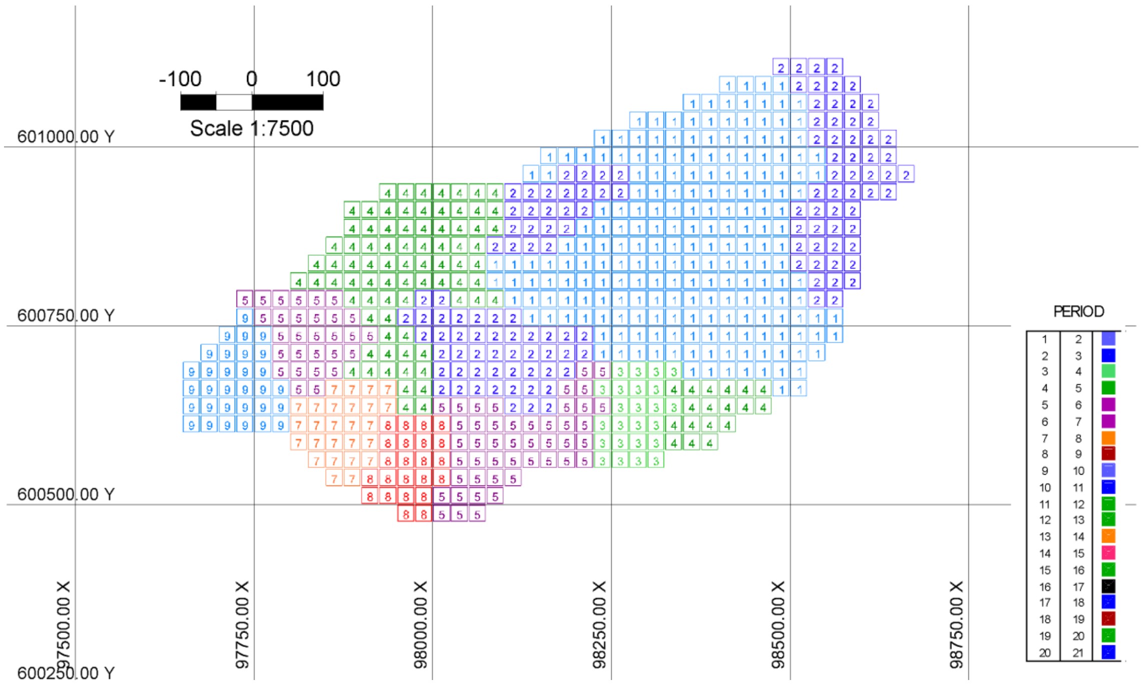

Fig 8.Sample Plan View of Schedule #1

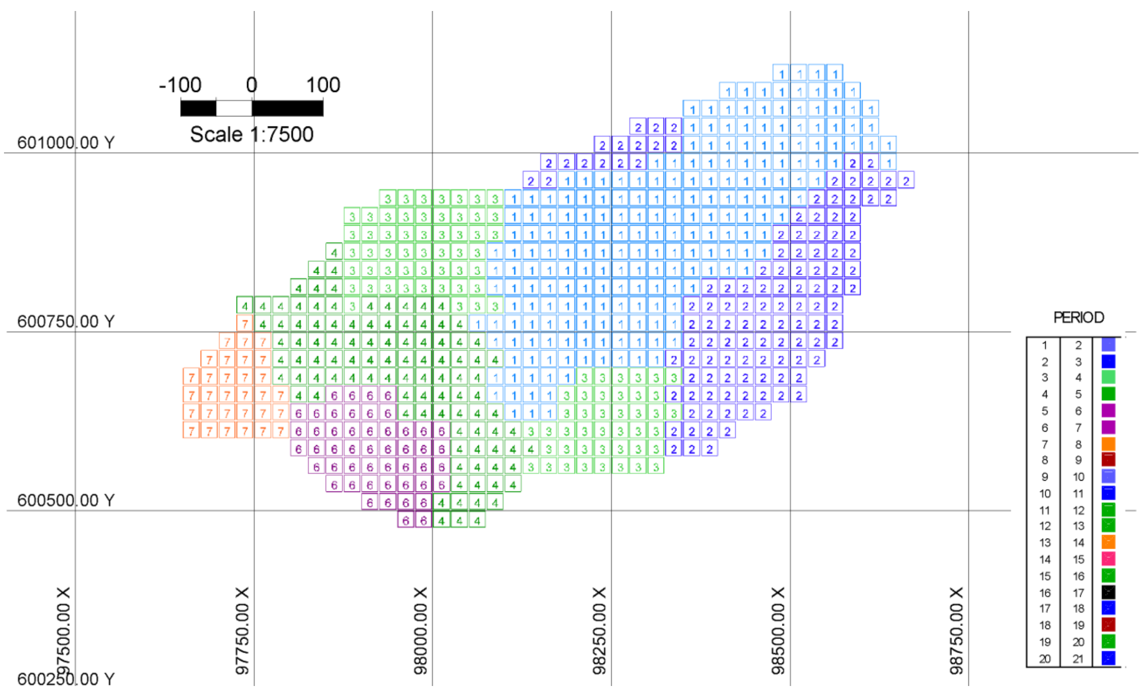

Fig 9.Sample Plan View of Schedule #2

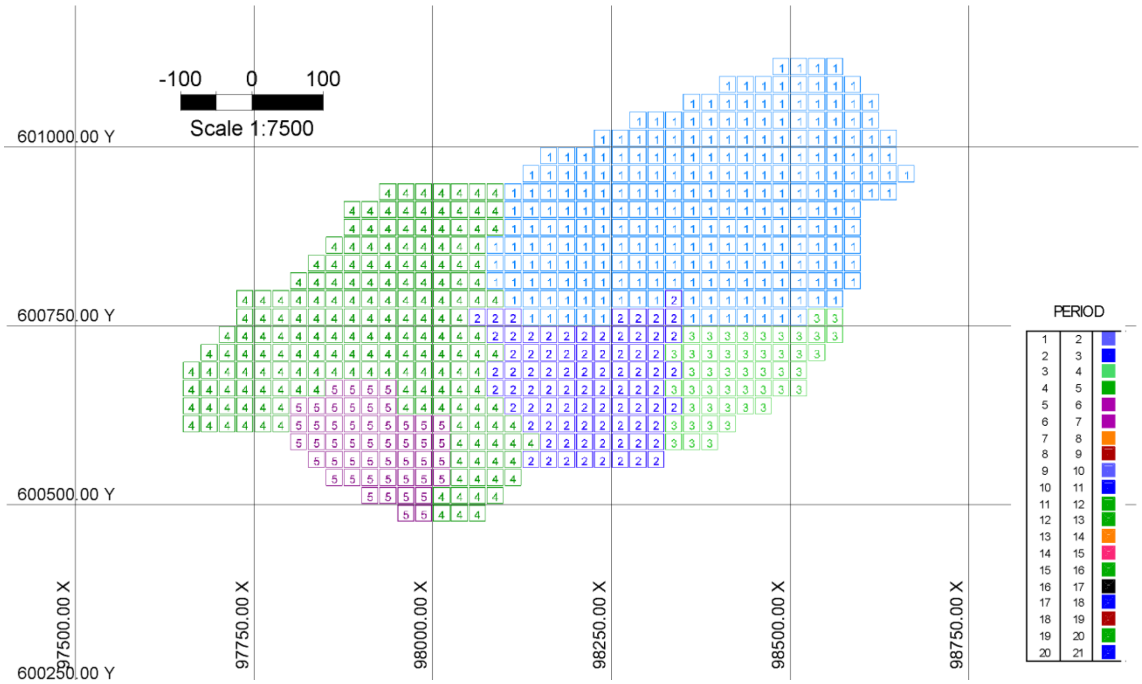

Fig 10.Sample Plan View of Schedule #3

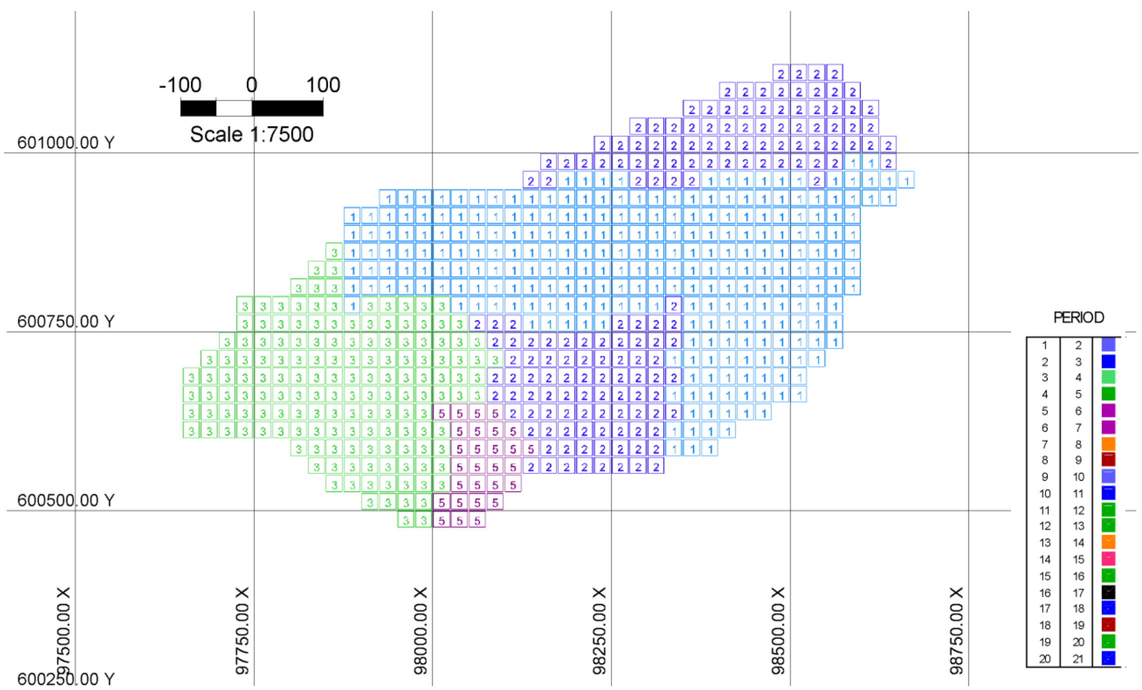

Fig 11.Sample Plan View of Schedule #4

7. Conclusions

The term "open pit long-term production planning" has received significant attention from researchers and practitioners. Various models have been introduced in literature, and solution procedures have been proposed to address the tremendous amount of computational resources required to solve the problem. One well-known approach is to cluster blocks into larger sets in order to reduce the size of the problem[3]. Two clustering algorithms are developed and compared in this paper. The hierarchical clustering technique results in more homogenous clusters than k-means, but it takes more processing time. Hierarchical clustering is then used to provide input to the proposed mathematical formulation for studying the effectiveness of the variable reduction techniques and the schedules generated using the multi-destination MILP. There is an obvious trend in the results suggesting that having smaller cuts leads to higher NPV but consumes more resources for divergence. However, the two factors do not change at the same rate, as a significant drop can be seen in CPU time while decreasing the number of cuts from 115 to 62, which is dramatically larger than the drop in NPV. The practicality of the generated schedules and the number of drop-cuts also change when the resolution of the model changes.

8. References

[1] Bagirov, A. M. (2008). Modified global k-means algorithm for minimum sum-of-squares clustering problems. *Pattern Recognition*, *41* (10), 3192-3199.

[2] Bley, A., Boland, N., Fricke, C., and Froyland, G. (2010). A strengthened formulation and cutting planes for the open pit mine production scheduling problem. *Comput. Oper. Res.*, *37* (9), 1641-1647.

[3] Busnach, E., Mehrez, A., and Sinuany-Stern, Z. (1985). A Production Problem in Phosphate Mining. *Operational Research Society*, *36* (4), 285-288.

[4] Caccetta, L. and Hill, S. P. (2003). An Application of Branch and Cut to Open Pit Mine Scheduling. *J. of Global Optimization*, *27* (2-3), 349-365.

[5] Chang, D.-X., Zhang, X.-D., and Zheng, C.-W. (2009). A genetic algorithm with gene rearrangement for K-means clustering. *Pattern Recognition*, *42* (7), 1210-1222.

[6] Chung, K.-L. and Lin, K.-S. (2006). An efficient line symmetry-based K-means algorithm. *Pattern Recognition Letters*, *27* (7), 765-772.

[7] Dhillon, I., Guan, Y., and Kulis, B. (2004). *Kernel k-means: spectral clustering and normalized cuts*. in Proceedings of Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining SE - KDD '04, ACM, Seattle, WA, USA, pp. 551-556.

[8] Feng, L., Qiu, M.-H., Wang, Y.-X., Xiang, Q.-L., Yang, Y.-F., and Liu, K. (2010). A fast divisive clustering algorithm using an improved discrete particle swarm optimizer. *Pattern Recogn. Lett.*, *31* (11), 1216-1225.

[9] Gershon, M. E. (1983). Optimal mine production scheduling: evaluation of large scale mathematical programming approaches. *International Journal of Mining Engineering*, *1* 315-329.

[10] Gershon, M. E. and Murphy, F. H. (1989). Optimizing single hole mine cuts by dynamic programming. *European Journal of Operational Research*, *38* (1), 56-62.

[11] Gonzalez, T. F. (1982). On the computational complexity of clustering and related problems. in *System Modeling and Optimization*, Vol. 38, *Lecture Notes in Control and Information Sciences*, Springer Berlin / Heidelberg, pp. 174-182.

[3] This part is a direct excerpt from [(Tabesh and Askari-Nasab, 2011)

[12] Hsu, C.-C., Chen, C.-L., and Su, Y.-W. (2007). Hierarchical clustering of mixed data based on distance hierarchy. *Information Sciences*, *177* (20), 4474-4492.

[13] Huang, Z. (1997). *A Fast Clustering Algorithm to Cluster Very Large Categorical Data Sets in Data Mining*. in Proceedings of In Research Issues on Data Mining and Knowledge Discovery, pp. 1-8.

[14] Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern Recognition Letters*

Award winning papers from the 19th International Conference on Pattern Recognition (ICPR), 19th International Conference in Pattern Recognition (ICPR), *31* (8), 651-666.

[15] Klingman, D. and Phillips, N. (1988). Integer Programming for Optimal Phosphate-Mining Strategies. *The Journal of the Operational Research Society*, *39* (9), 805-810.

[16] Kumral, M. and Dowd, P. A. (2005). A Simulated Annealing Approach to Mine Production Scheduling. *The Journal of the Operational Research Society*, *56* (8), 922-930.

[17] MacQueen, J. B. (1967). *Some methods for classification and analysis of multivariate observations*. in Proceedings of Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, pp. 281–297.

[18] Mahajan, M., Nimbhorkar, P., and Varadarajan, K. (2010). The planar k-means problem is NP-hard. *Theoretical Computer Science*, *In Press, Corrected Proof*

[19] Newman, A. M., Rubio, E., Caro, R., Weintraub, A. s., and Eurek, K. (2010). A Review of Operations Research in Mine Planning. *Interfaces*, *40* (3), 222-245.

[20] Niknam, T. and Amiri, B. (2010). An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis. *Applied Soft Computing*, *10* (1), 183-197.

[21] Niknam, T., Taherian Fard, E., Pourjafarian, N., and Rousta, A. (2010). An efficient hybrid algorithm based on modified imperialist competitive algorithm and K-means for data clustering. *Engineering Applications of Artificial Intelligence*, *In Press, Corrected Proof*

[22] Osanloo, M., Gholamnejad, J., and Karimi, B. (2008). Long-term open pit mine production planning: a review of models and algorithms. *International Journal of Mining, Reclamation and Environment*, *22* (1), 3 - 35.

[23] Ramazan, S. (2001). Open pit mine scheduling based on fundamental tree algorithm. Ph.D. Thesis Thesis,

[24] Ramazan, S. (2007). The new Fundamental Tree Algorithm for production scheduling of open pit mines. *European Journal of Operational Research*, *177* (2), 1153-1166.

[25] Samanta, B., Bhattacherjee, A., and Ganguli, R. (2005). A genetic algorithms approach for grade control planning in a bauxite deposit. in *Application of Computers and Operations Research in the Mineral Industry*, null SV - null DO - doi:10.1201/9781439833407.ch44 ed, Taylor & Francis, pp. 337-342 SE.

[26] Tabesh, M. and Askari-Nasab, H. (2011). A Two Stage Clustering Algorithm for Block Aggregation in Open Pit Mines. *Transactions of the Institution of Mining and Metallurgy. Section A, Mining industry*,

[27] Zalik, K. R. (2008). An efficient k'-means clustering algorithm. *Pattern Recognition Letters*, *29* (9), 1385-1391.

[28] Zhang, M. (2006). *Combining genetic algorithms and topological sort to optimise open-pit mine plans*. in Proceedings of 15th International Symposium on Mine Planning and Equipment Selection (MPES), FIORDO S.r.l., Torino, Italy, pp. 1234-1239.

9. Appendix

MATLAB and CPLEX Code Documentation