Mixed integer linear programming formulations for open pit production scheduling

Hooman Askari-Nasab and Kwame Awuah-Offei[1]
Mining Optimization Laboratory (MOL)
University of Alberta, Edmonton, Canada

Abstract

We have proposed two mixed integer linear programming (MILP) formulations for large-scale long-term open pit production scheduling problem. We developed, implemented, and tested the proposed MILP theoretical frameworks for large-scale open pit production scheduling.

1. Introduction

Historical assessment of mineral project performances has demonstrated the sensitivity of projects' profitability to decisions based upon mine production schedules. The life-of-mine production schedule defines the complex strategy of displacement of ore, waste, overburden, and tailings over the mine life. The objectives of long-term production schedules are to determine the sequence of extraction and displacement of material in order to maximize the future cash flows of mining operations within the existing economic, technical, and environmental constraints. Long-term production schedules lead to definition of reserves and are the backbone of short-term planning and day to day mining operations. The long-term production schedules resolve mine and processing plant capacity and their expansion potential; the production schedule, also defines the management investment strategy. Deviations from optimal plans in mega mining projects will result in enormous financial losses, delayed reclamation, and resource sterilization.

In this study, we have proposed four mixed integer linear programming (MILP) formulations for large-scale long-term open pit production scheduling problem. We developed, implemented, and tested the proposed MILP theoretical frameworks for large-scale open pit production scheduling.

Current production scheduling methods in the literature are not just limited to, but can be divided into three main categories: heuristic methods, applications of artificial intelligence techniques, and operations research methods. Some of these algorithms are embedded into available commercial software packages.

[1] Assistant Professor, Department of Mining & Nuclear Engineering, University of Missouri-Rolla, USA,

One of the heuristic methods used in mine production scheduling was proposed by Gershon (1987). XPAC AutoScheduler (Runge Limited, 1996-2009), a commercial mine scheduling software is developed based on Gershon's (1987) proposed heuristic. Gershon's (1987) algorithm generates cones upward from each block to approximate the shape of a pit and to determine whether or not the block in question could be part of the schedule. A list of exposed blocks and a ranking of those blocks based on what makes it more desirable or less desirable to mine an exposed block at the present time is updated through the algorithm with an index called the positional weight. This weighted function is used to determine the removal sequence.

Another popular heuristic used in strategic mine planning software, such as Whittle (Gemcom Software International, 1998-2008) and NPV Scheduler (Datamine Corporate Limited, 2008) is based on the concept of parametric analysis introduced by Lerchs and Grossmann (1965) (LG). The LG algorithm provides an optimal solution to the final pit outline. There are unlimited numbers of strategies of reaching the final pit, which each has a different discounted cash-flow. The optimal production schedule is the strategy that would maximize the discounted cash-flow and meets all the physical and economical constraints. The parametric analysis generates a series of nested pits based on varying the price of the product (revenue factor) and finding an optimal pit layout using LG algorithm. These nested pits then are used as a guideline to identify clusters of high grade ore and to determine the production schedule. The main disadvantage of heuristic algorithms is that the solution may be far from optimal and in mega mining projects, this is equal to huge financial losses.

Various models based on a combination of artificial intelligence techniques have been developed (Denby and Schofield, 1994; Denby et al., 1996; Tolwinski and Underwood, 1996; Askari-Nasab, 2006; Askari-Nasab et al., 2008; Askari-Nasab and Awuah-Offei, 2009). Tolwinski and Underwood (1996) used a method which combines concepts from dynamic programming, stochastic optimisation, and artificial intelligence with heuristic rules to obtain ultimate pit limit and production planning concurrently. The method works by modelling the development of the mine as a sequence of pits where each pit differs from the previous pit by the removal of blocks. A probability distribution based on the frequency with which particular states occur is used to determine the state changes. Heuristic rules are incorporated to learn these characteristics of the sequence of pits which produce a good, or poor, result. Denby et al. (1996) employed genetic algorithms and simulated annealing by generation of random pit population and assessment of a fitness function to acquire the production schedule and final pit, concurrently. The advantages of their method are flexibility and solution for the ultimate pit limit and production schedule at the same time. The major drawback is that the results are not reproducible and there was no measure of the optimality of the solution.

In a series of publications the authors developed and tested the intelligent agent-based theoretical framework for open pit mine planning (Askari-Nasab et al., 2005; Askari-Nasab, 2006; Askari-Nasab et al., 2007; Askari-Nasab and Szymanski, 2007; Askari-Nasab et al., 2008) comprising algorithms based on reinforcement learning (Sutton and Barto, 1998) and stochastic simulation. This intelligent open pit simulator (IOPS) (Askari-Nasab, 2006) has a component that simulates practical mining push-backs over the mine life. An intelligent agent interacts with the push-back simulator to learn the optimal push-back schedule using reinforcement learning. The intelligent agent-based mine planning simulator, IOPS, was successfully used to determine the optimal push-back schedule of an open pit mine with a geological block model containing 883 200 blocks (Askari-Nasab and Awuah-Offei, 2009). A number of the artificial intelligence techniques, such as IOPS are based on frameworks that theoretically will converge to the optimal solution, given sufficient number of simulation iterations. The main disadvantage however, is that there is no quality measure to solutions provided comparing against the theoretical optimum.

A variety of operations research approaches including linear programming (LP) and mixed integer linear programming (MILP) have been applied to the mine production scheduling problem. The pioneer work of Johnson (1969) used an LP model, which led to the MIP formulations by Gershon (1983) for the production scheduling problem. Mixed integer linear programming mathematical optimization models have the capability to consider multiple ore processors and multiple elements during optimization. This flexibility of mathematical programming models result in production schedules generating significantly higher net present value (NPV) than those generated by the other traditional methods. Every orebody is different, but for a typical open pit long-term scheduling problem, the number of blocks is in the order of a couple of hundred thousands to millions and the number of scheduling periods are twenty and more for a life-of-mine yearly schedule. Evidently, the number of integer and linear decision variables, and the number of constraints formulating a problem of this size would easily exceed the capacity of current state of hardware and commercial mathematical optimization solvers.

Various models based on mixed integer linear programming mathematical optimisation have been used to solve the long-term open-pit scheduling problem (Caccetta and Hill, 2003; Ramazan and Dimitrakopoulos, 2004; Dagdelen and Kawahata, 2007; Boland et al., 2009). The applications of MILP models result in production schedules generating near theoretical optimal net present values. In practice, formulating a real size mine production planning problem by including all the blocks as integer variables will simply exceed the capacity of the current commercial mathematical optimisation solvers. Various methods of aggregation have been used to reduce the number of integer variables that are required to formulate the mine planning problem with MILP techniques. Ramazan and Dimitrakopoulos (2004) illustrated a method to reduce the number of binary integer variables by setting waste blocks as continuous variables instead of integer variables. Ramazan and Dimitrakopoulos (2004) reported a case study on a small single level nickel laterite block model with 2030 blocks over three periods.

Ramazan et al. (2005; 2007) presented an aggregation method based on fundamental tree concepts to reduce the number of decision variables in the MILP formulation. The fundamental tree algorithm has been used in a case study with 38 457 blocks within the final pit limits, Whittle strategic mine planning software (Gemcom Software International, 1998-2008) has been used to break-down the overall problem into four push-backs. Subsequently, the blocks within the push-backs were aggregated into 5512 fundamental trees and scheduled over eight periods using the formulation presented in Ramazan and Dimitrakopoulos (2004). Information about the run-time of the MILP models are not presented in Ramazan (2007); also the break-down of the problem into four push-backs based on the nested pit approach and formulating them as a separate MILP would not generate a global optimum solution to the overall problem. On the other hand the size of the problem of around thirty thousand blocks over eight periods is more a mid-range planning problem rather than a long-term life of mine schedule.

Caccetta and Hill (2003) presented a formulation that used binary integer variables, they developed and implemented a personalized branch-and-cut (Horst and Hoang, 1996) method in C++ using CPLEX (Bixby, 1987-2009) to solve the relaxed LP sub-problems. Boland et al. (2009) have demonstrated an iterative disaggregation approach to using a finer spatial resolution for processing decisions to be made based on the small blocks, while allowing the order of extraction decisions be made at an aggregate level. Boland et al. (2009) reported notable improvements on the convergence time of their algorithm for a model with 96 821 blocks and 125 aggregates over 25 periods. However, combining 96 821 blocks into only 125 aggregates would reduce the freedom of decision variables and the schedule generated could not be considered as an optimal solution in comparison to the case that 96 821 blocks had complete freedom. Moreover, in Boland et al. (2009) there is no representation of the generated schedules in terms of annual ore and waste production, average grade of ore processed, and cross sections and plan views of the schedules to assess the practicality of the solutions from mining operational point of view. Boland et al. (2009) also did not represent enough information on their method of aggregation, they assumed that an aggregation method similar to Ramazan (2007) would be used.

MineMax (Minemax Pty Ltd, 1998-2009) is a commercially available strategic mine scheduling software, which uses an MILP formulation solved by ILOG CPLEX (Bixby, 1987-2009) solver. Given that, MineMax is a commercial software we couldn't find detailed information about the approach and formulation, but our understanding from the evaluation of the demo tutorial version of MineMax is that as a general strategy it is suggested to initially breakdown the final pit into nested pit shells based on parametric analysis concepts represented by Lerchs and Grossmann (1965). The pit shells define a pit to pit precedence constrained by the minimum and maximum number of benches by which the mining of one specified pit shell is to lag behind the previous one. The other option to define rules for precedence of extraction is either by proportions mined on each bench or by block precedence based on the overall pit slopes. Next each pit shell is formulated as a separate MILP model which can contribute to the overall quantity of mining and processing targets within the grade and precedence constraints; this approach result in MILP formulations for each pit shell with smaller size which will converge faster, but it could not be considered a global optimization of the problem since the pit shells are defined by the parametric analysis first. Another optimization strategy is using sliding windows which are sub-problems that are tackled on a period by period basis.

Blasor (Stone et al., 2007) and Prober (Whittle, 2007) are other proprietary software which tackle the strategic mine production scheduling by an MILP. Current MILP formulations used for open pit production scheduling fail because of: (1) inability to solve large-scale real-size mining problems as global optimization problem, and (2) inability to quantify the geological uncertainty inherent within the problem and, as a result, the associated risk with the mine plans. To overcome these problems the first step is to develop, implement, and test a theoretical framework that is capable of handling real-size mining problems in a global optimization framework. The development of an MILP model that can handle a deterministic large-scale mine production schedule will fulfill this objective.

We have critically reviewed the MILP formulations of the open pit production scheduling problem. We have implemented, and tested two of these MILP formulations. The shortcomings and deficiencies of the formulations were documented. We have proposed two MILP formulations for the long-term open pit production scheduling problem to overcome the shortcomings of the reviewed methods. We have divided the major decision variables into two categories, continuous variables representing the portion of a block that is going to be extracted in each period and binary integer variables controlling the order of extraction of blocks or the precedence of mining-cuts through a dependency directed graph using depth-first-search algorithm. The depth-first-search algorithm component, added a very important practical mining feature to the model. This model allows variable pit slopes to be integrated in the MILP formulation. As a practical constraint our MILP formulation also ensures that the fractional extraction of blocks are not going to be split over more than three periods. We have implemented the optimization formulation in TOMLAB/CPLEX (Holmström, 1989-2009) environment. An iron ore mine intermediate scheduling case study over twelve periods was carried out to verify and validate the models and to illustrate the effectiveness of the generated schedules from mining point of view. The results proved that the models are accurate and efficient and showed enhanced CPU time performance comparing to the reviewed models.

The next section of the paper covers the assumptions, problem definition, and the notations of variables. Section 3 presents four mixed integer linear programming formulations of the problem, while Section 4 presents the numerical modelling. The next section represents the numerical experiments used for verification and validation of the models. Finally, Section 6 presents the conclusions and future work followed by the list of references.

2. Assumptions, problem definition, and notation

We assume that the orebody is represented by a geological block model, which is a three-dimensional array of rectangular or cubical blocks used to model orebodies and other sub-surface structures. Numerical data are used to represent a single attribute of the orebody such as: rock types, densities, grades, elevations, or economic data. The size of the block that is needed for outlining the orebody depends on the shape and size of the ore body, mining bench height, mining method, and mining equipment. Geostatistics provides accurate and reliable estimations or simulation of block attributes at locations where no measurements are available. The most common estimation method used is Kriging (Krige, 1951). However, Kriging results do not capture uncertainty and may be systematically biased. Uncertainty always exists in presence of sparse geological data. Conditional simulation algorithms such as Sequential Gaussian Simulation (Isaaks, 1990) are geostatistical methods used to assess geological uncertainty. The generated realizations are equally probable and represent plausible geological outcomes. We will tackle the deterministic problem in this study so Kriging is the estimation method of choice.

We assume that each block is subdivided into smaller regions identified as parcels. A parcel is part of a block for which the rock-type, tonnage and element content are estimated or simulated by Geostatistical methods. A block may contain zero or more parcels. The total tonnage of the parcels may be the same as the tonnage of the block, or it may be less. If it is less, the difference is called undefined waste, which is waste of unknown rock-type. If a block has no parcels, the total tonnage of the block is undefined waste. Neither the position of a parcel within a block, nor its shape, are defined (Gemcom Software International, 1998-2008). The spatial location of each block is defined by the coordinates of its center; attributes such as rock-type, quantity of valuable and contaminant elements, and block tonnage are estimated and grade of each element of interest is calculated based on the quantity of the attributes and the ore tonnage in that block.

We consider that the geological block model is going to be extracted using open pit mining techniques. We assume that a classical optimum design of final pit limits is carried out; either based on graph theory (Lerchs and Grossmann, 1965; Zhao and Kim, 1992) or network flow algorithm (Johnson and Barnes, 1988; Yegulalp and Arias, 1992). This pit outline represents reserves that would maximize the profit. We have illustrated in Askari-Nasab and Awuah-Offei (2009) that a final pit outline obtained directly by using an optimal long-term scheduling algorithm will result in a pit outline that is a subset of the conventional final pit outline generated by the Lerchs and Grossmann's algorithm (1965). Consequently, we will follow the classical process of open pit long-term scheduling of first, finding the final pit limits and then generating a production schedule within the final pit outline.

The basic problem, in its simplest form, is finding a sequence in which ore and waste blocks should be removed from the predefined open pit outline and their respective destinations, over the life of mine, so that the net present value of the operation is maximized. The production schedule is subject to a variety of physical, technical and economic constraints. The constraints enforce the mining extraction sequence, overall pit slopes, mining, milling, and refining capacities, blending material to meet head grade requirements, minimum mining width, and the number of active mining benches in each production period. The problem presented here involves scheduling of N different ore and waste blocks within the final pit outline over T different periods of extraction.

Blocks within the same level or mining bench are grouped into clusters based on their attributes, spatial location, rock type, and grade distribution. We refer to these clusters of blocks as mining-cuts. Similar to blocks, each mining-cut has coordinates representing the center of the cut and its spatial location. Fig. 1 illustrates a schematic plan view of a mining bench. Blocks are aggregated into mining-cuts.

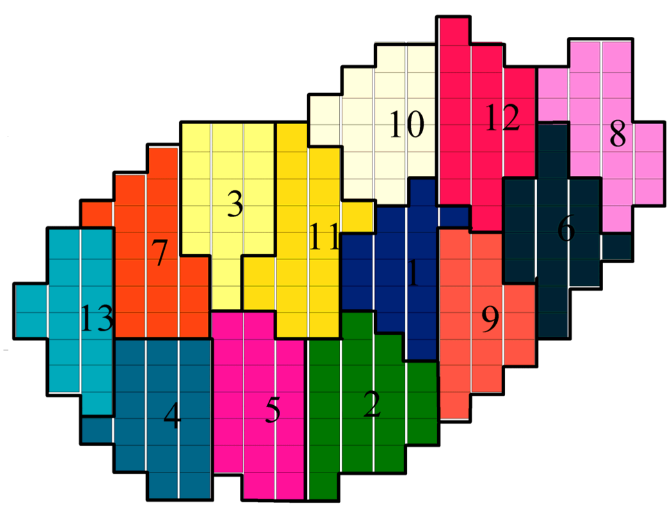

Fig. 1. Schematic plan view of aggregated blocks into mining-cuts on a mining bench.

This grade aggregation methodology will summarize ore data, but will maintain a relevant separation of lithology. Instead of the common approach of estimating grades for the element of interest in blocks, the total amount of the element in the block would be modeled. This tactic will allow to combine the smaller blocks into mining-cuts without sacrificing the accuracy of the estimated values and to model a more realistic equipment movement strategy. The mining-cut clustering algorithm developed uses fuzzy logic clustering (Kaufman and Rousseeuw, 1990) and is not within the scope of this paper and we will disseminate the clustering approach in another publication.

2.1. Notation

We will present four different MILP formulations for the open pit production scheduling problem. The notation of decision variables, parameters, sets, and constraints are as follows:

2.1.1 Sets

$\mathcal{N} = \{1,...,N\}$	set of all the blocks in the model.
$\mathcal{K} = \{1,...,K\}$	set of all the mining-cuts in the model.
$\mathcal{A} = \{1,..., A\}$	set of all directed arcs in the blocks' precedence directed graph denoted by $G_b(\mathcal{N}, \mathcal{A})$.
$\mathcal{B} = \{1,..., B\}$	set of all edges in the mining-cuts precedence directed graph denoted by $G_c(\mathcal{K}, \mathcal{B})$.
$D(J)$	for each block, n, there is a set $D(J) \subset \mathcal{N}$ which includes all the blocks that must be extracted prior to mining block n to ensure that block n is exposed for mining with safe pit slopes, where J is the total number of blocks in set $D(J)$.
$C(L)$	for each block, n, there is a set $C(L) \subset D(J)$ defining the immediate predecessor blocks that must be extracted prior to extraction of block n, where L is the total number of blocks in set $C(L)$.
$H(S)$	for each mining-cut c_k, there is a set $H(S) \subset \mathcal{K}$ defining the immediate predecessor cuts that must be extracted prior to extracting mining-cut k, where S is the total number of cuts in set $H(S)$.

2.1.2 Indices

A general parameter f can take four indices in the format of $f_{k,n}^{e,t}$. Where:

$t \in \{1,....,T\}$	index for scheduling periods.
$k \in \{1,..., K\}$	index for mining-cuts.
$n \in \{1,..., N\}$	index for blocks.
$e \in \{1,..., E\}$	index for elements of interest in each block.

2.1.3 Parameters

d_n^t	d_n^t is the discounted profit generated by extracting block n in period t.
v_n^t	the discounted revenue generated by selling the final product within block n in period t minus the extra discounted cost of mining all the material in block n as ore and processing it.
q_n^t	the discounted cost of mining all the material in block n as waste.
c_k	mining-cut k.
g_n^e, g_k^e	average grade of element e in ore portion of block n and average grade of element e in ore portion of mining-cut k.
$gu^{e,t}$	upper bound on acceptable average head grade of element e in period t.
$gl^{e,t}$	lower bound on acceptable average head grade of element e in period t.
o_n, o_k	ore tonnage in block n and ore tonnage in mining-cut k.
w_n, w_k	waste tonnage in block n and waste tonnage in mining-cut k.
pu^t	upper bound on processing capacity of ore in period t (tonnes).
pl^t	lower bound on processing capacity of ore in period t (tonnes).
mu^t	upper bound on mining capacity in period t (tonnes).
ml^t	lower bound on mining capacity in period t (tonnes).
$r^{e,t}$	processing recovery, is the proportion of element e recovered in time period t.
$p^{e,t}$	price in present value terms obtainable per unit of product (element e).
$cs^{e,t}$	selling cost in present value terms per unit of product (element e).
$cp^{e,t}$	extra cost in present value terms per tonne of ore for mining and processing.
cm^t	cost in present value terms of mining a tonne of waste in period t.

2.1.4 Decision Variables

<u>Model 01</u>

$z_n^t \in \{0,1\}$	binary integer variable, equal to 1 if block n is to be mined in period t, otherwise 0.

<u>Model 02</u>

$u_n^t \in [0,1]$	continuous variable, representing the portion of block n to be extracted and processed in period t or mined and treated as waste in period t.

$a_n^t \in \{0,1\}$ binary integer variable controlling the precedence of extraction of blocks. a_n^t is equal to one if extraction of block n has started by or in period t, otherwise it is zero.

Model 03

$x_n^t \in [0,1]$ continuous variable, representing the portion of block n to be extracted as ore and processed in period t.

$y_k^t \in [0,1]$ continuous variable, representing the portion of mining-cut c_k to be mined in period t, fraction of y characterizes both ore and waste included in the mining-cut.

$b_k^t \in \{0,1\}$ binary integer variable controlling the precedence of extraction of mining- cuts. b_k^t is equal to one if extraction of mining-cut c_k has started by or in period t, otherwise it is zero.

Model 04

$s_k^t \in [0,1]$ continuous variable, representing the portion of mining-cut c_k to be extracted as ore and processed in period t.

Decision variables y_k^t and b_k^t with the same definition as in Model 03 are used in Model 04 as well.

2.2. Economic block value modeling

The objective functions of the MILP formulations are to maximize the net present value of the mining operation. Hence, we need to define a clear concept of economic block value based on ore parcels which could be mined selectively. The profit from mining a block depends on the value of the block and the costs incurred in mining and processing. The cost of mining a block is a function of its spatial location, which characterizes how deep the block is located relative to the surface and how far it is relative to its final dump. The spatial factor can be applied as a mining cost adjustment factor for each block according to its location to the surface. The discounted profit from block n is equal to the discounted revenue generated by selling the final product contained in block n minus all the discounted costs involved in extracting block n, this is presented by Eqs. (1) and (2).

$$\text{discounted profit} = \text{discounted revenue} - \text{discounted costs} \tag{1}$$

$$d_n^t = \underbrace{[\sum_{e=1}^{E} o_n \times g_n^e \times r^{e,t} \times (p^{e,t} - cs^{e,t})}_{\text{discounted revenues}} - \underbrace{\sum_{e=1}^{E} o_n \times cp^{e,t}] - [(o_n + w_n) \times cm^t]}_{\text{discounted costs}} \tag{2}$$

For simplification purposes we denote:

$$v_n^t = [\sum_{e=1}^{E} o_n \times g_n^e \times r^{e,t} \times (p^{e,t} - cs^{e,t}) - \sum_{e=1}^{E} o_n \times cp^{e,t}] \tag{3}$$

$$q_n^t = (o_n + w_n) \times cm^t \tag{4}$$

3. Mixed integer linear programming models for open pit production scheduling

We present four different formulations for the open pit production scheduling problem, with the objective function to maximize the NPV of the mining operation. It is intuitively apparent that higher NPV's could be achieved by block models with small block sizes and high resolution of the orebody model. The block sizes for production scheduling must be chosen similar to the selective mining size. If the size of the block is not properly defined, the generated schedule will be simulating the mining operation with a selectivity that could not be achieved in practice. We have developed, implemented, tested, and compared four MILP formulations based on different precedence of extraction graphs and extraction and processing selectivity levels. The four MILP formulations are: (i) Model 01- this formulation is similar to Ramazan and Dimitrakopoulos (2004), it only consists of binary integer decision variables, z_n^t, and generates a schedule at block level resolution. We have extended the model by integrating a variable pit slope component into the implementation; (ii) Model 02 – we have proposed this model based on the concepts presented in Caccetta and Hill (2003). The schedule is generated based on a strict temporal sequence of blocks. Caccetta and Hill (2003) formulation only uses binary integer decision variables, which makes the size of branch and cut tree intractable for a large scale problem. The proposed formulation uses continuous variables, u_n^t to model extraction and processing at block level and binary integer decision variables, a_n^t, are used to control precedence of extraction; (iii) Model 03 – this formulation is developed based on the concepts of Boland et al. (2009), processing is controlled at block level with continuous decision variables, x_n^t; where y_k^t, controls the extraction at mining-cut level. Also, the precedence of extraction of blocks is controlled at the mining-cut level by means of binary integer variables b_k^t. The continuous decision variables (x_n^t and y_k^t) lead to fractional block extraction, the proposed model provides control over the maximum number of fractions that each block would take; (iv) Model 04 – this formulation is based on a combination of concepts presented in Models 02 and 03. Extraction, processing, and order of block extraction are controlled at mining-cut level.

3.1. Model 01 – extraction and processing at block level – only binary decision variables

Objective function:

$$\max \sum_{t=1}^{T} \sum_{n=1}^{N} d_n^t \times z_n^t \tag{5}$$

Subject to:

$$gl^{t,e} \le \sum_{n=1}^{N} g_n^e \times o_n \times z_n^t \Bigg/ \sum_{n=1}^{N} o_n \times z_n^t \le gu^{t,e} \qquad \forall t \in \{1,...,T\}, \quad e \in \{1,...,E\} \tag{6}$$

$$pl^t \le \sum_{n=1}^{N} o_n \times z_n^t \le pu^t \qquad \forall t \in \{1,...,T\} \qquad (7)$$

$$ml^t \le \sum_{n=1}^{N} (o_n + w_n) \times z_n^t \le mu^t \qquad \forall t \in \{1,...,T\} \qquad (8)$$

$$J \times z_n^t - \sum_{j=1}^{J} \sum_{i=1}^{t} z_j^i \le 0 \qquad \forall n \in \{1,...,N\}, \quad t \in \{1,...,T\}, \quad j \in D(J) \quad (9)$$

$$\sum_{t=1}^{T} z_n^t = 1 \qquad \forall n \in \{1,...,N\} \qquad (10)$$

Where Eq. (6) is grade blending constraints; these inequalities ensure that the head grade of the elements of interest and contaminants are within the desired range in each period. There are two equations (upper bound and lower bound) per element per scheduling period in Eq. (6). Eq. (7) is processing capacity constraints; these inequalities ensure that the total ore processed in each period is within the acceptable range of processing plant capacity. There are two equations (upper bound and lower) per period per ore type. Eq. (8) is mining constraints; these inequalities ensure that the total tonnage of material mined (ore, waste, overburden, and undefined waste) in each period is within the acceptable range of mining equipment capacity in that period. There are two equations (upper bound and lower bound) per period. Eq. (9) controls the precedence relationship of block extraction and pit slopes. For each block there is a set $D(J)_n \subset \mathcal{N}$,which includes all the blocks that must be extracted prior to mining block n to ensure that block n is exposed for mining with safe pit slopes, where J is the total number of blocks in set $D(J)_n$. The set $D(J)_1$ for the block labeled as 1 is hatched in Fig. 2. J is the total number of blocks in the set; it is equal to 28 in this case. Fig. 2 also demonstrates how variable slopes could be modeled by means of constructing the set $D(J)$; the block labeled as 5 is added to set $D(J)_1$ to construct a model with flatter slope on the right hand side. There is one equation per block per period for Eq. (9). Depending on the spatial location of the block in the block model the set of $D(J)$ would have different number of blocks. The slope constraints presented by Eq. (9) is the main reason of increase in the number of constraints and the complexity of Model 01 formulation. Finally, Eq. (10) defines reserve constraints; we assume that a final pit limit is superimposed on the block model and we are going to schedule the extraction of all the blocks within the final pit limit or push-back. Eq. (10) ensures that all the blocks within the final pit are going to be extracted once.

3.2. Model 02 - extraction and processing at block level – binary and continuous variables

In this model the mining and processing are at block level resolution; the schedule is controlled by continuous variables so fractional extraction of blocks may occur. The order of block extraction is controlled by binary integer variables at block level.

Objective function:

$$\max \sum_{t=1}^{T} \sum_{n=1}^{N} d_n^t \times u_n^t \qquad (11)$$

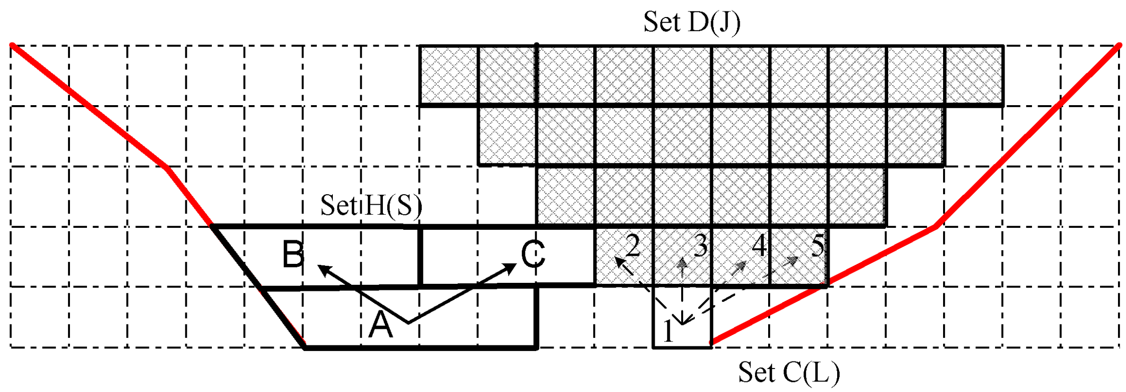

Fig. 2. Precedence of block extraction in the proposed MILP models.

Subject to:

$$gl^{t,e} \le \sum_{n=1}^{N} g_n^e \times o_n \times u_n^t \Big/ \sum_{n=1}^{N} o_n \times u_n^t \le gu^{t,e} \qquad \forall t \in \{1,\ldots,T\}, \quad e \in \{1,\ldots,E\} \tag{12}$$

$$pl^t \le \sum_{n=1}^{N} o_n \times u_n^t \le pu^t \qquad \forall t \in \{1,\ldots,T\} \tag{13}$$

$$ml^t \le \sum_{n=1}^{N} (o_n + w_n) \times u_n^t \le mu^t \qquad \forall t \in \{1,\ldots,T\} \tag{14}$$

$$a_n^t - \sum_{i=1}^{t} u_l^i \le 0 \qquad \forall n \in \{1,\ldots,N\}, \quad t \in \{1,\ldots,T\}, \quad l \in C(L) \tag{15}$$

$$\sum_{i=1}^{t} u_n^i - a_n^t \le 0 \qquad \forall n \in \{1,\ldots,N\}, \quad t \in \{1,\ldots,T\} \tag{16}$$

$$a_n^t - a_n^{t+1} \le 0 \qquad \forall n \in \{1,\ldots,N\}, \quad t \in \{1,\ldots,T-1\} \tag{17}$$

$$\sum_{t=1}^{T} u_n^t = 1 \qquad \forall n \in \{1,\ldots,N\} \tag{18}$$

Eq. (12) controls the grade blending constraints. Eqs. (13) and (14) are processing and mining capacity constraints. Eqs. (15) to (17) control the relationship of block extraction precedence by binary integer variables at block level. Model 02 only requires the set of immediate predecessors' blocks on top of each block to model the order of block extraction relationship. This is presented by set $C(L)$ in Eq. (15). Fig. 2 illustrates the set $C(L)_1$ for the block labeled as 1, the set includes $C(L)_1 = \{2,3,4,5\}$. Where L the number of blocks is equal to 4; compare this to formulation in Model 01 where J was 28. Eq. (18) ensures that the fractions of blocks that are extracted over the scheduling periods are going to sum up to one, which means all the block within the final pit outline are going to be scheduled.

3.3. Model 03 – extraction at mining-cut level and processing at block level – binary and continuous variables

In this model, processing is at block level and extraction is at mining-cut level. The amount of ore processed is controlled by the continuous variable x_n^t, this allows fractional extraction of blocks in different periods. The order of block extraction which is controlled by the directed graph $G_b(\mathcal{N},\mathcal{A})$ is transferred into the control of the precedence of mining-cuts by means of directed graph $G_c(\mathcal{K},\mathcal{B})$. The precedence of mining-cuts relationship is modeled via the binary integer variable b_k^t. This is superior to Models 01 and 02, in that, the amount of ore processed and amount of material mined are controlled by two separate variables.

Objective function:

$$\max \sum_{t=1}^{T}\sum_{k=1}^{K}\left(\sum_{n\in c_k} v_n^t \times x_n^t - (\sum_{n\in c_k} q_n^t)\times y_k^t\right) \tag{19}$$

Subject to:

$$gl^{t,e} \le \sum_{n=1}^{N} g_n^e \times o_n \times x_n^t \Big/ \sum_{n=1}^{N} o_n \times x_n^t \le gu^{t,e} \qquad \forall t\in\{1,...,T\},\quad e\in\{1,...,E\} \tag{20}$$

$$pl^t \le \sum_{n=1}^{N} o_n \times x_n^t \le pu^t \qquad \forall t\in\{1,...,T\} \tag{21}$$

$$ml^t \le \sum_{k=1}^{K}\left(\sum_{n\in c_k}(o_n + w_n)\right)\times y_k^t \le mu^t \qquad \forall t\in\{1,...,T\} \tag{22}$$

$$x_n^t \le y_{k,n}^t \qquad \forall n\in\{1,...,N\},\quad n\in c_k,\quad t\in\{1,...,T\} \tag{23}$$

$$b_k^t - \sum_{i=1}^{t} y_s^i \le 0 \qquad \forall k\in\{1,...,K\},\quad t\in\{1,...,T\},\quad s\in H(S) \tag{24}$$

$$\sum_{i=1}^{t} y_k^i - b_k^t \le 0 \qquad \forall k\in\{1,...,K\},\quad t\in\{1,...,T\} \tag{25}$$

$$b_k^t - b_k^{t+1} \le 0 \qquad \forall k\in\{1,...,K\},\quad t\in\{1,...,T-1\} \tag{26}$$

Eqs. (20) to (22) control the grade blending, processing capacity, and mining capacity. In Model 02 grade blending, processing capacity, and mining capacity are modeled by only one decision variable, u_n^t. In Model 03 the extraction and processing of ore is controlled by continuous decision variable x_n^t at block level, where mining is modeled using a continuous variable y_k^t at mining-cut level. This method enables us to have a high resolution solution for selection of ore and processing. The total amount of material mined and the order of extraction is then modeled at mining-cut level y_k^t. This approach reduces the number of binary integer variables in the model drastically. Fig. 2 illustrates set $H(S)=\{B,C\}$, the predecessor mining-cuts that must be extracted prior to extraction of mining-cut A. Eq. (23) represents inequalities that ensure the amount of ore of any block which is processed in any given period is less than or equal to the amount of rock extracted from the mining-cut that the block belongs to in the considered time period. For each mining-cut, k, Eqs. (24) to (26) check the set of immediate predecessor cuts that must be extracted prior to extracting mining-cut, k.

3.4. Model 04 – extraction and processing at mining-cut level – binary and continuous variables

In this model, mining and processing are both at mining-cut level. The blocks are aggregated prior to schedule optimization and the ore processing and mining are controlled by two continuous variables.

Objective function:

$$\max \sum_{t=1}^{T}\sum_{k=1}^{K}(v_k^t \times s_k^t - q_k^t \times y_k^t) \tag{27}$$

$$gl^{t,e} \le \sum_{k=1}^{K} g_k^e \times o_k \times s_k^t \Big/ \sum_{k=1}^{K} o_k \times s_k^t \le gu^{t,e} \qquad \forall t \in \{1,...,T\}, \quad e \in \{1,...,E\} \tag{28}$$

$$pl^t \le \sum_{k=1}^{K} o_k \times s_k^t \le pu^t \qquad \forall t \in \{1,...,T\}, \quad e \in \{1,...,E\} \tag{29}$$

$$ml^t \le \sum_{k=1}^{K}(o_k + w_k)\times y_k^t \le mu^t \qquad \forall t \in \{1,...,T\} \tag{30}$$

$$s_k^t \le y_k^t \qquad \forall k \in \{1,...,K\}, \quad t \in \{1,...,T\} \tag{31}$$

Equations (24) to (26)

Eqs. (28) to (30) control the grade blending, processing capacity, and mining capacity constraints at mining-cut level with fractional extraction from mining-cuts. The extraction from mining-cuts is assumed to be uniform among all the blocks which belong to that mining-cut. Eq. (31) ensures that the amount of ore extracted and processed from any mining-cut in any given period is less than or equal to the amount of rock extracted from that mining-cut. Eqs. (24) to (26) are similar to those demonstrated in Model 03 for precedence of block extraction.

4. Numerical modeling

In most linear optimization problems, the variables of the objective function are continuous in the mathematical sense, with no gaps between real values. To solve such linear programming problems, ILOG CPLEX implements optimizers based on the simplex algorithms (Winston, 1995) (both primal and dual simplex) as well as primal-dual logarithmic barrier algorithms.

Branch and cut is a method of combinatorial optimization for solving integer linear programs. The method is a hybrid of branch and bound and cutting plane methods (Horst and Hoang, 1996). Refer to Wolsey (1998) for a detailed explanation of the branch and cut algorithm, including cutting planes. In recent years there has been significant improvements in mathematical programming optimizers such as ILOG CPLEX (Bixby, 1987-2009). This optimizer uses branch and cut techniques to solve MILP models and it makes the latest theory in optimization of large-scale industrial problems available commercially. In this study we used TOMLAB/CPLEX version 11.2 (Holmström, 1989-2009) as the MILP solver. TOMLAB/CPLEX efficiently integrates the solver package CPLEX (ILOG Inc, 2007) with MATLAB environment (MathWorks Inc., 2007). An important termination criterion that the user can set explicitly in CPLEX is the MILP gap tolerance. We have used the relative MILP gap tolerance, which indicates to CPLEX to stop when an integer feasible solution has been proved to be within the gap tolerance of optimality.

4.1. Size and complexity

One of the major obstacles in using the MILP formulations for mine production scheduling is the sheer size of the problem. The number of blocks, N, in the model is usually between tens of thousands to millions which will lead to a formulation with an objective function with many variables. Moreover, the main physical constraint in open pit mining is the block extraction precedence modeled by binary integer variables. This set of constraints also controls the overall pit slope in different regions. The numbers of blocks that must be extracted prior to mining each block are numerous and will result in formulations with many constraints. Therefore, we are dealing with an MILP formulation with many variables and many constraints.

The most common difficulty with MILPs is the size of the branch and cut tree. The tree becomes so large that insufficient memory remains to solve an LP sub-problem. The number of binary integer variables in the formulations determines the size of the branch and cut tree. As a general strategy in our formulations we aimed at reducing the number of binary integer variables, we also focused on developing formulations that will mainly use continuous optimization techniques rather than discrete optimization. Table 1 shows the number of decision variables and the number of binary integer variables required for the proposed MILP formulations as a function of number of blocks, N, number of mining-cuts, K, and number of scheduling periods, T. The goal has been to reduce the number of binary integer variables in the models by introducing mining-cuts as the means of controlling the precedence of extraction of blocks rather than having one binary integer variable per block.

Table 1 – number of decision variables in the MILP formulations

MODEL	Number of decision variables	Number of integer variables
Model 01	$N \times T$	$N \times T$
Model 02	$2 \times N \times T$	$N \times T$
Model 03	$(2 \times K + N) \times T$	$K \times T$
Model 04	$3 \times K \times T$	$K \times T$

5. Results and discussions

We have developed, implemented, and tested the proposed MILP models presented in section 3 in TOMLAB/CPLEX environment (Holmström, 1989-2009). We compare the performances of the proposed models based on net present value generated, practical mining production constraints, smoothness of the generated schedules, size of the mathematical formulations, the number of integer variables required in formulation, and computational time required for convergence.

All the developed formulations are verified by numerical experiments on a synthetic data set containing 120 blocks and a real mining push-back including 2 598 blocks with seven mining benches. We have also validated Model 03 and Model 04 with an iron ore life-of-mine schedule with 26 000 blocks over a 20 year scheduling horizon. We tested our models on a Quad Core Dell Precision T7400 computer at 3.00 GHz, with 3.25 GB of RAM. Since we aimed at a comparative analysis of the proposed models a relative tolerance of 2% on the gap between the best integer objective and the feasible integer solution was chosen.

Table 2a and b show the numerical results of the tests of MILP models with the data set containing 120 blocks over four periods of extraction. To reach a feasible solution in different models we have used a mining capacity upper bound of 64 to 66 million tonnes per period, whereas the processing capacity varies between 4.9 to 5.3 million tonnes per period. We were forced to set different upper and lower bounds for different models, because the tighter boundaries on some models resulted in infeasible solutions. As expected Model 02 generated the highest NPV; in this formulation processing and mining are both at block level with continuous variables so fractional extraction of blocks are allowed. The variables in Model 02 have the highest resolution among all the other models. Model 01 had the longest runtime with 42.53 seconds as expected; this is due to the numerous numbers of constraints that are generated by Eq. (9). Also, the model is a pure MIP with no continuous variables, the formulation searched 4 432 branch and bound nodes to reach an optimized solution within a 2.25% gap. Ramazan et al. (2005) proposed to reduce the number of binary integer variables required by Model 01 by defining only the ore blocks as binary variables; this approach will reduce the number of binary integer variables but since the formulation assigns blocks to periods of extraction, rather than determining a strict temporal sequence of blocks, still the size of the problem and the runtime even for a small number of blocks is not within a reasonable timeframe and simply this formulation is not a practical tool. In Models 03 and 04 the concept of mining-cuts are introduced and the number of binary integer variables and the computational time required is reduced drastically compared to Model 01 and 02 (>72% and >99.6%, respectively). On the other hand, the NPV of Model 03 and 04 are reduced by 0.72% and 1.11% when compared to Model 02, as in Model 03 and Model 04 the variables are aggregated and have less freedom in the MILP formulations.

Table 2a – Inputs and numerical results for the synthetic data set containing 120 blocks

MODEL	Blocks – Cuts ($N-K$)	Periods (T)	mu^t / ml^t (MT)	pu^t / pl^t (MT)	NPV ($M)
01	120 - 0	4	64 / 0	5.3 / 0	387.88
02	120 - 0	4	64 / 0	4.9 / 0	391.34
03	120 - 21	4	66 / 0	5.0 / 0	388.53
04	120 - 21	4	66 / 0	5.0 / 0	387.00

Table 2b – Inputs and numerical results for the synthetic data set containing 120 blocks

MODEL	Root Node Gap %	CPU TIEM (S)	Coefficient Matrix $A(rows \times col)$	No. decision variables	No. Integer variables	No. of B & B nodes visited
01	2.25	42.53	372 × 480	480	480	4432
02	2.37	0.61	3004 × 960	960	480	38
03	1.29	0.17	920 × 648	648	84	0
04	1.40	0.15	530 × 252	252	84	0

Table 3a and Table 3b show the numerical results of the MILP models for an iron ore push-back data set containing 2 598 blocks over twelve scheduling periods. Fig. 3 and 4 illustrate cross sections of the final pit limits including the orebody and rock type model. The push-back studied includes seven benches of the final pit demonstrated in Fig. 3 and 4 from elevation 1500m to 1590m. The blocks represent a volume of rock equal to 50m×25m×15m. The model contains 155 million tonnes of material with 84 million tonnes of iron ore with an average grade of 73% magnetic weight recovery (MWT%). Sulfur and phosphor are present as deleterious elements and their grades need to be controlled within an acceptable range in the processing plant feed.

Table 3a - Inputs and numerical results for the data set containing 2598 blocks

MODEL	Blocks -Cuts ($N-K$)	Periods (T)	mu^t / ml^t (MT)	pu^t / pl^t (MT)	$gu^{e,t} / gl^{e,t}$ (MWT%)	$gu^{e,t} / gl^{e,t}$ (S% & P%)	NPV ($M)
01	2598 - 0	12	13 / 0	7.15 / 0	No bounds	0 / 1.8 & 0.14	-
02	2598 - 0	12	13 / 0	7.15 / 0	No bounds	0 / 1.8 & 0.14	3011.65
03	2598 - 148	12	13.2 / 0	7.15 / 0	65 / 80	0 / 1.8 & 0.14	2947.68
04a	2598 - 148	12	13.5 / 0	7.15 / 7.0	65 / 80	0 / 1.8 & 0.14	2947.09
04b	2598 - 239	12	13.5 / 0	7.15 / 7.0	65 / 80	0 / 1.8 & 0.14	2985.02
04c	2598 - 436	12	13.5 / 0	7.15 / 7.0	65 / 80	0 / 1.8 & 0.14	2991.55

Table 3b – Inputs and numerical results for the data set containing 2598 blocks

MODEL	Root Node GAP %	CPU TIME (S)	Coefficient Matrix $A(rows \times cols)$	No. Of nonzero elements in A	No. Integer variables	No. of B & B nodes visited
01	No Integer	Solution	-	5 668 788	31 176	-
02	0.05	14 748.32	280 944 × 62 352	2 091 882	31 176	340
03	1.08	1 168.60	43 202 × 34 728	226 774	1 776	70
04a	0.52	26.73	11 232 × 5 325	75 640	1 776	50
04b	0.35	34.37	17 580 × 8 604	115 898	2 868	40
04c	0.58	34.94	30 264 × 15 696	198 082	5 232	30

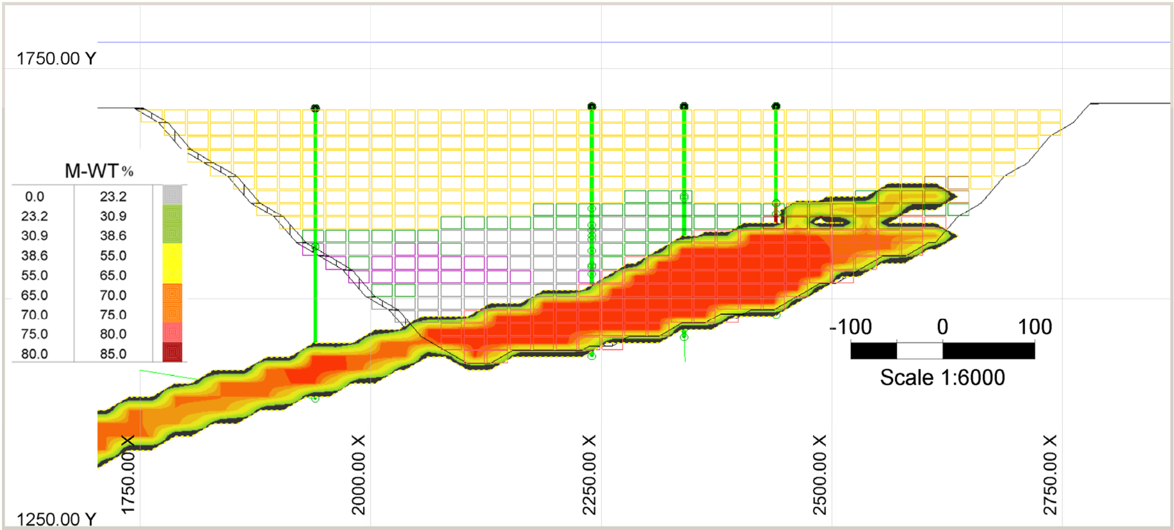

Fig. 3. Cross section 98400 N of the final pit including the orebody and rock type model looking west (meters).

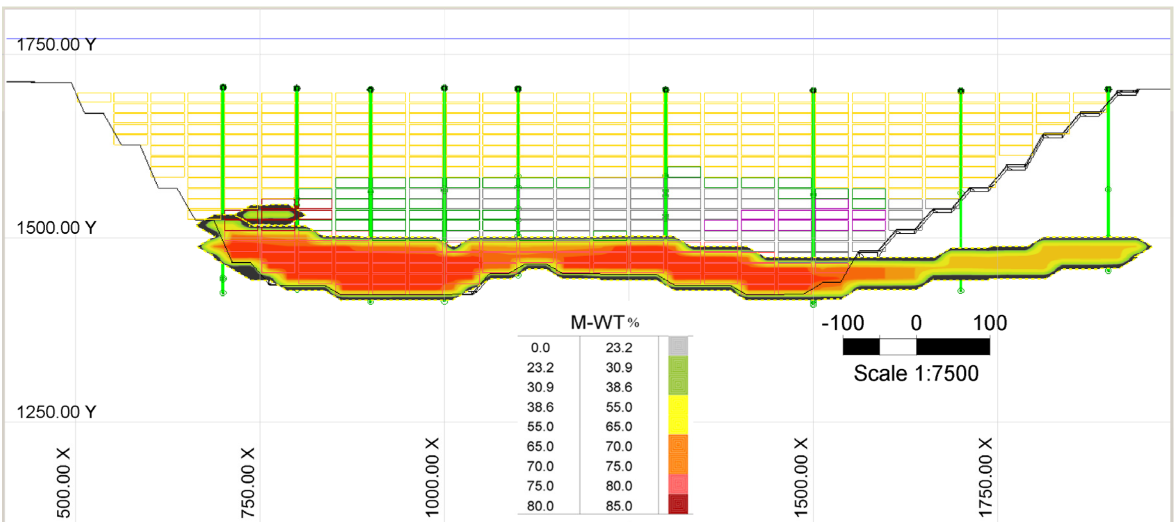

Fig. 4. Cross section 600640 E of the final pit including the orebody and the rock type model looking north (meters).

The maximum allowable average grade for sulfur is 1.8% and for phosphor is 0.14% per period. It is also desirable to keep an average head grade between 65% and 80% of magnetic weight recovery (MWT). Our goal was to generate a schedule with a uniform feed within a range of 7 to 7.15 million tonnes of ore per period. Furthermore, we intended to keep a steady (1.89 to 1.93) stripping ratio over the scheduling horizon, so we chose a maximum mining equipment capacity of 13.5 million tonnes per period. This would ensure that the mining equipment capacity required is not going to fluctuate over time.

To schedule 2598 blocks over 12 scheduling periods, Model 01 generated a huge coefficient matrix with more than 5.6 million nonzero elements and an integer solution for the formulation did not exist (Table 3b). Model 02 generated a coefficient matrix with more than 2 million nonzero elements and ran for over four hours. As expected, Model 02 generated the highest NPV of $3011.65 million of all the models, with a 0.05% gap tolerance. Fig. 5a to 5c show the plan view and cross sections of the generated schedule by Model 02. A closer examination of Fig. 5a to 5c reveals a tight schedule from a practical mining point of view. Fig. 5b shows that in the second period mining occurs on six active benches. This requires considerable equipment movement. Implementation of such a schedule is a challenge in the field, if limited numbers of mining shovels are available.

In Model 03 we have used clustering techniques to aggregate the blocks into 148 mining-cuts, clustering has reduced the number of integer variables to 1 776 from 31 176 in Model 02. Although in Model 03 the processing is at block level, the NPV has dropped to $2947.68 million, a 2.1% reduction (Table 3b). This is because mining occurs at the mining-cut level and has reduced flexibility when compared to Model 02. The run time has been reduced to half an hour from four hours.

Figs. 6a to 6c illustrate the plan view and cross sections of the schedule generated by Model 03. This schedule is more practical from a mining point of view since there are only three active benches in the second period compared to the six active benches of Model 02.

We have investigated the effect of number of mining-cuts on the quality of solutions in terms of NPV and the run-time on three different cases with Model 04. We clustered the 2598 blocks into 148, 239, and 436 mining-cuts using the clustering algorithm. We refer to these as Model 04a, 04b, and 04c, respectively. A model with fewer mining-cuts will have fewer number of integer variables in the MILP formulation and result in reduced run-time, as shown in Table 3b.

Alternatively, models with more mining-cuts imply more freedom for the decision variables and higher NPV would be expected. The NPV for Model 04a with 148 mining-cuts is $2947.09 million, while the NPV for the Model 04c with 436 cuts has increased to $2991.55 million. One of the very important improvements with the Model 04 formulations is the drastic drop in runtime. Compare the four hour runtime for Model 02 to the less than 35 seconds for Model 04. This is a very significant improvement; one must take into account that we are examining a very small model with only 2598 blocks in this study. The increase in the number of blocks would very quickly make Model 02 intractable and the model is not going to converge for larger real size models. Scrutinizing Fig. 7a to 7c shows that the number of active benches for different periods of extraction has decreased to two or three benches. The reduction of number of active benches is because of clustering blocks into mining-cuts. A smaller number of mining-cuts would generate clusters with more blocks. On the other hand, we are constructing the mining-cuts on a bench by bench basis; which tends to generate schedules that expand the mine outlines more horizontally rather than vertically. More research is required to develop a framework that will optimize the number of blocks included in each mining-cut in terms of making a balance in aiming for the highest NPV possible, while the maximum number of active benches included in the schedule would be practical from a mining point of view.

One of the features that make MILP formulations a robust platform for mine planning is not only the NPV maximization but also the control that the mine planner would have on upper and lower bounds of ore and waste production targets.

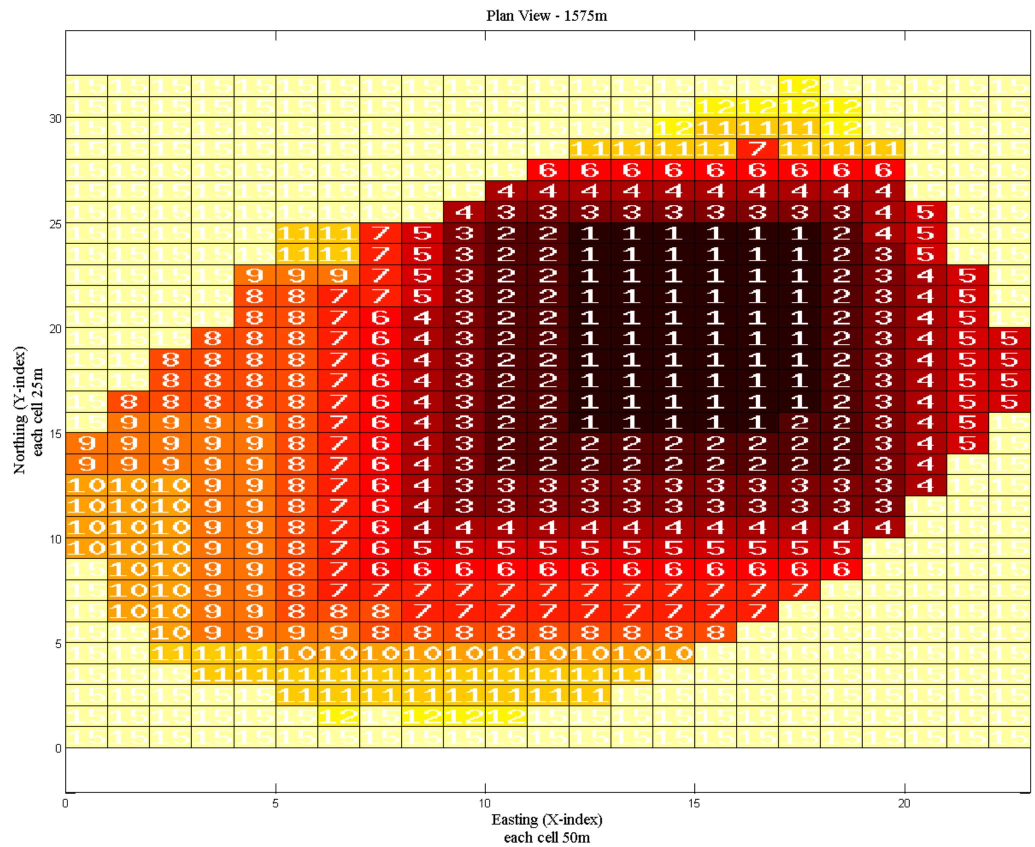

Fig. 5a. Model 02, plan view of bench 1575m.

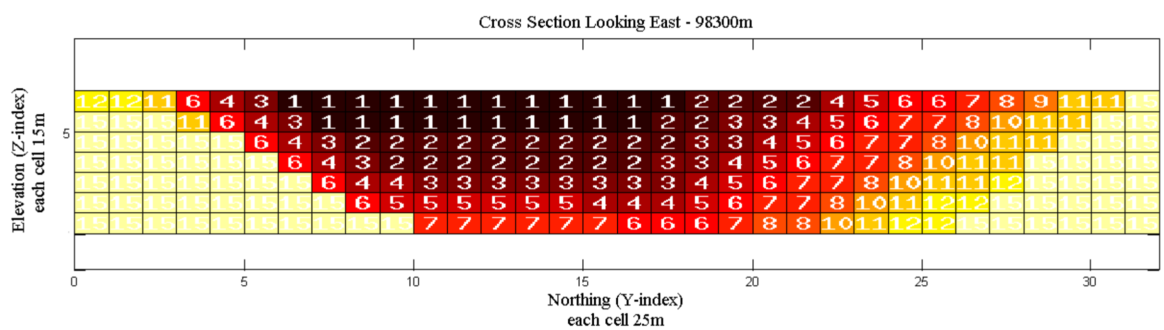

Fig. 5b. Model 02, cross section 98300m looking east.

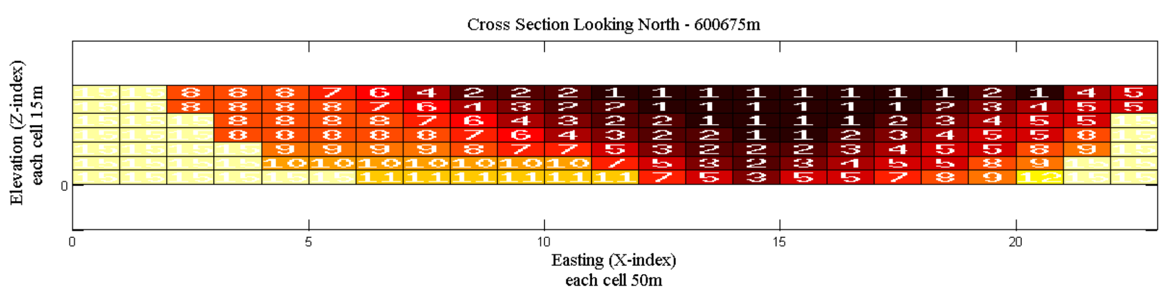

Fig. 5c. Model 02, cross section 600675m looking north.

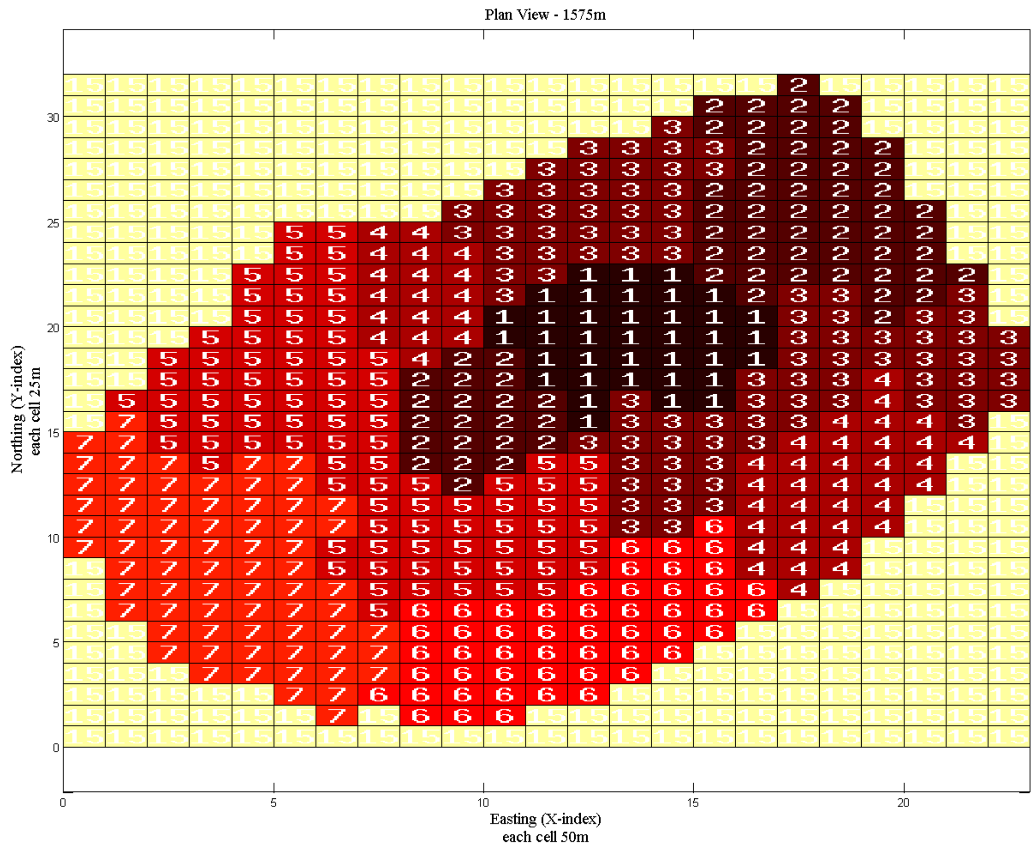

Fig. 6a. Model 03, plan view of bench 1575m.

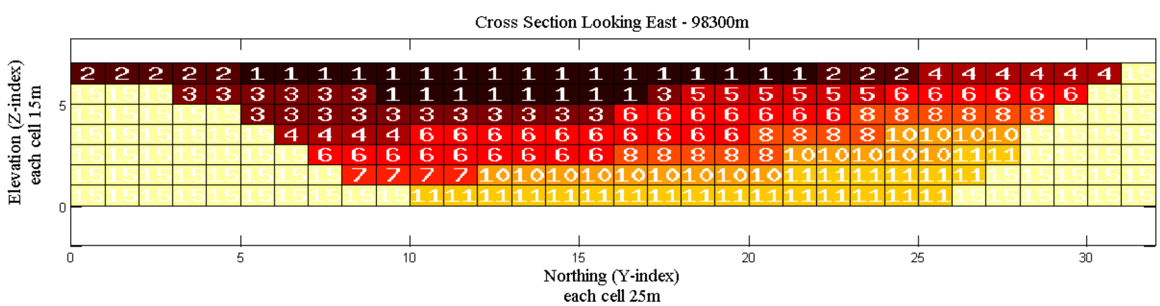

Fig. 6b. Model 03, cross section 98300m looking east.

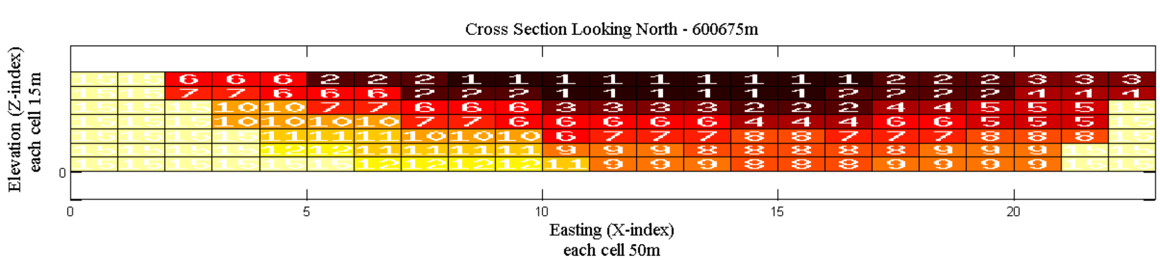

Fig. 6c. Model 03, cross section 600675 looking north.

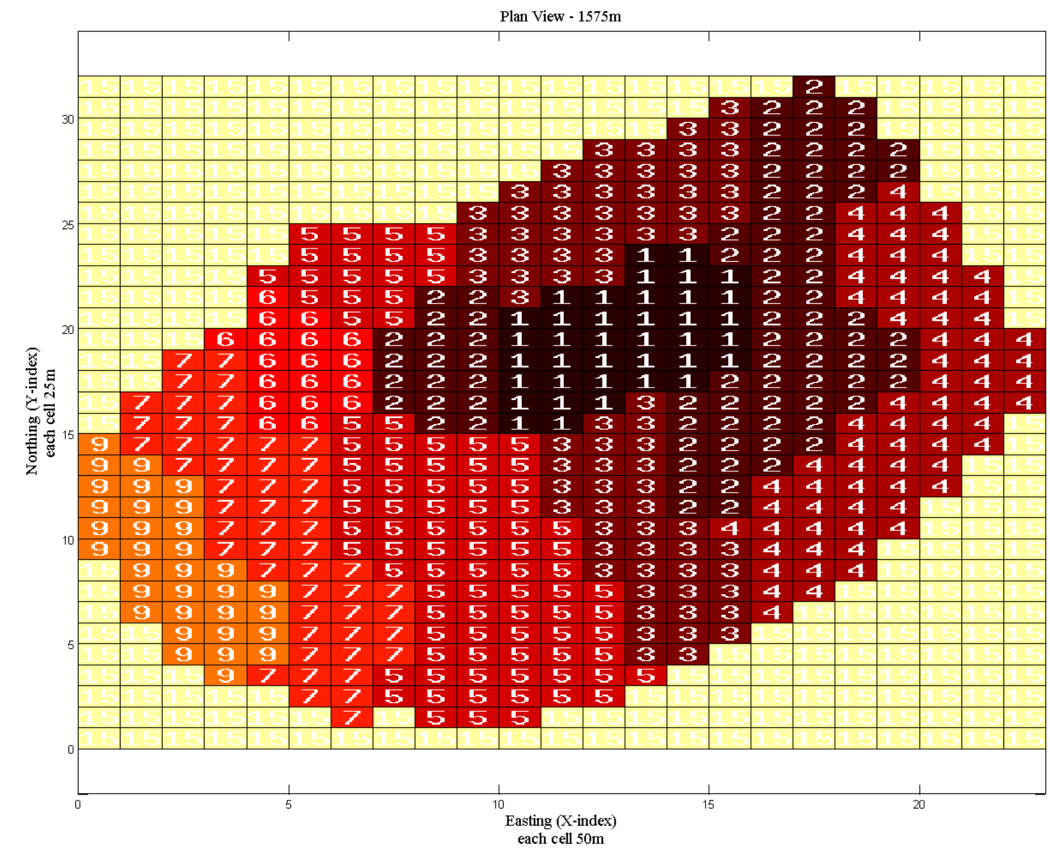

Fig. 7a. Model 04a, plan view of bench 1575m.

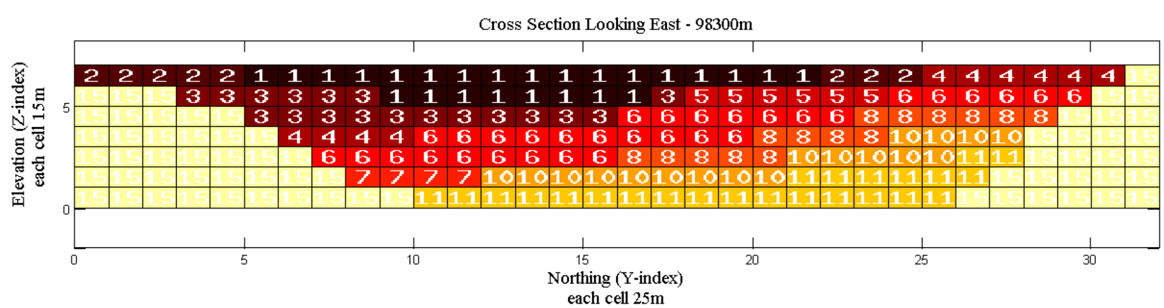

Fig. 7b. Model 04a, cross section 98300m looking east.

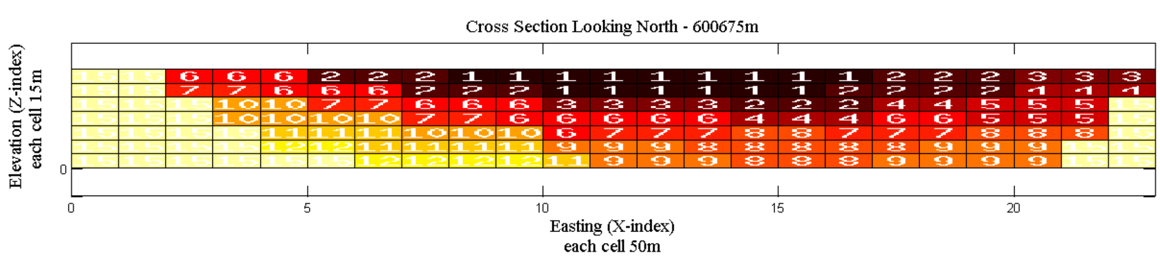

Fig. 7c. Model 04a, cross section 600675m looking north.

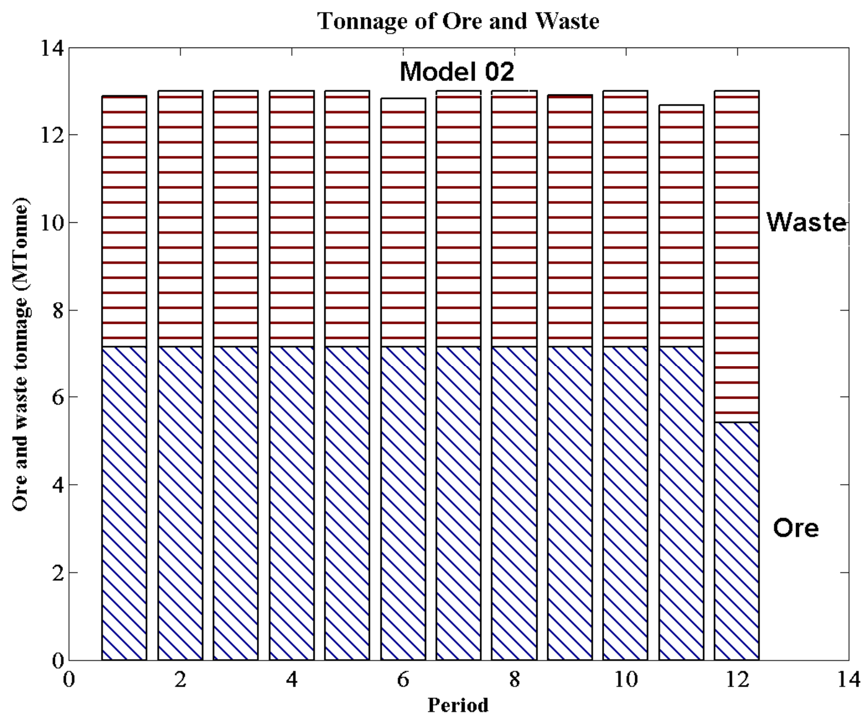

Fig. 8a. Model 02 tonnage of ore and waste per period.

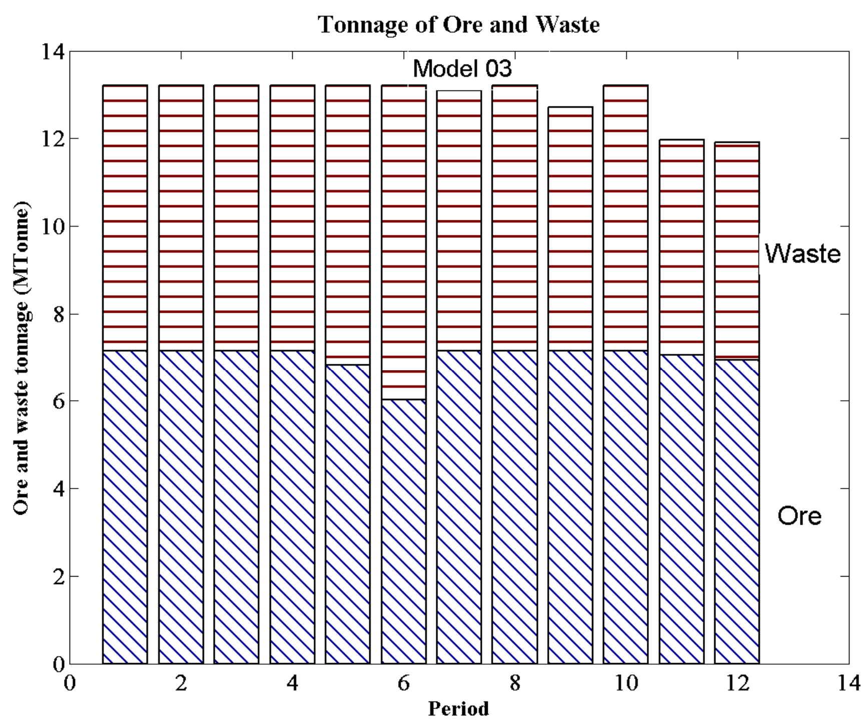

Fig. 8b. Model 03 tonnage of ore and waste per period.

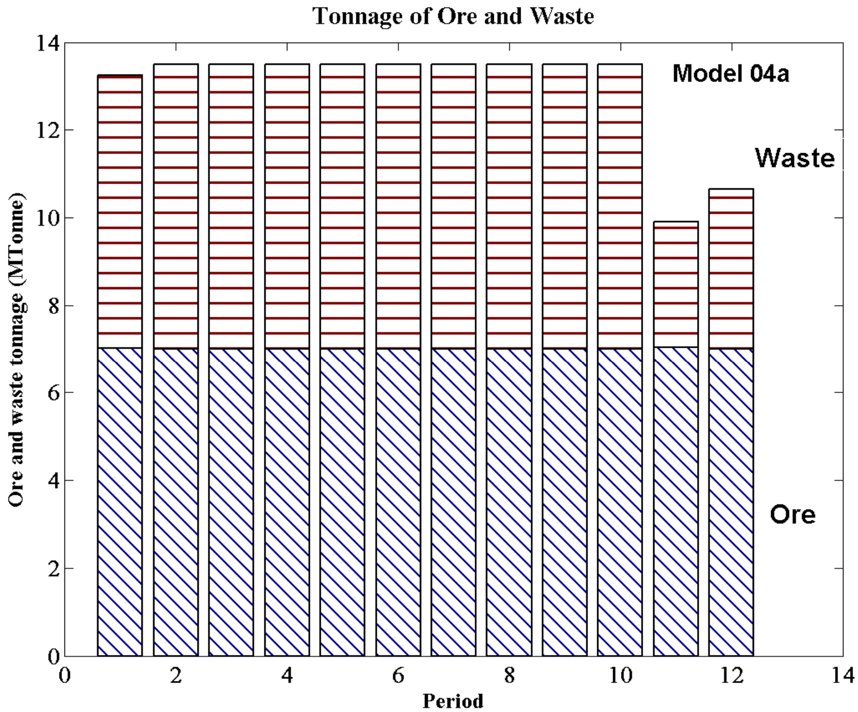

Fig. 8c. Model 04a tonnage of ore and waste per period.

Also, there is an inherent task of blending the run-of- mine materials before concentration. The objective is to mine in such a way that the resulting mix meets the quality and quantity specifications of the processing plant. The blending problem becomes more important as we get more into detailed planning in short/medium range plans.

Fig. 8a to 8c illustrate the yearly tonnage of ore processed, waste mined, and the total tonnage of material mined in each period of production. In Model 02 we set a maximum mining capacity of 13 million tonnes per period and a processing capacity of 7.15 million tonnes per period (Table 3a) with no lower bound on mining and processing capacities. Fig. 8a illustrates the results of Model 02, the generated schedule is smooth with very little fluctuations in the tonnage of feed and stripping ratio. Meanwhile, referring Fig. 9 to 11 for the blending results show that the average grade of MWT, sulphur, and phosphor are within the acceptable range defined in Table 3a (plots are for Models 02, 03, and 04a). A very interesting phenomenon that should be noticed is how in Model 02 (Fig. 9), the MILP high grades for iron ore in the early periods and then the MWT average grade starts to get lower every year. The high grading phenomenon in early periods is completely in accordance with the highest NPV generated by Model 02.

In Model 03 we set a maximum mining capacity of 13.2 million tonnes per period and a processing capacity of 7.15 million tonnes per period (Table 3a) with no lower bound on mining and processing capacities. We were forced to increase the mining upper bounds since the 13 million tonnes upper bound did not generate a feasible integer solution; this is because of introduction of mining-cuts into the model, which reduces the freedom of the variables. Assessment of Fig. 8b shows that the ore feed is not as smooth as Model 02 schedule. Grade blending constraints are all honored except for phosphor in the first period that exceeded to 0.145% rather than 0.14%.

In Model 04a we set a maximum mining capacity of 13.5 million tonnes per period and a processing capacity of 7.15 million tonnes per period (Table 3a) with a lower bound of 7 million tonnes on the processing capacity. All the constraints presented in Table 3a are honored and the ore production schedule is the smoothest among all with an iron ore grade of 65% to 80% average for the MWT.

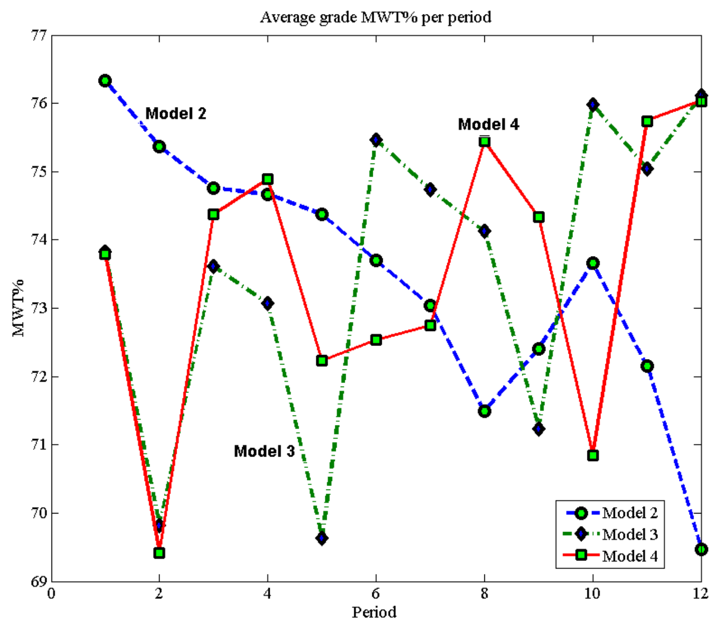

Fig. 9. Average iron ore (MWT%) grade per period.

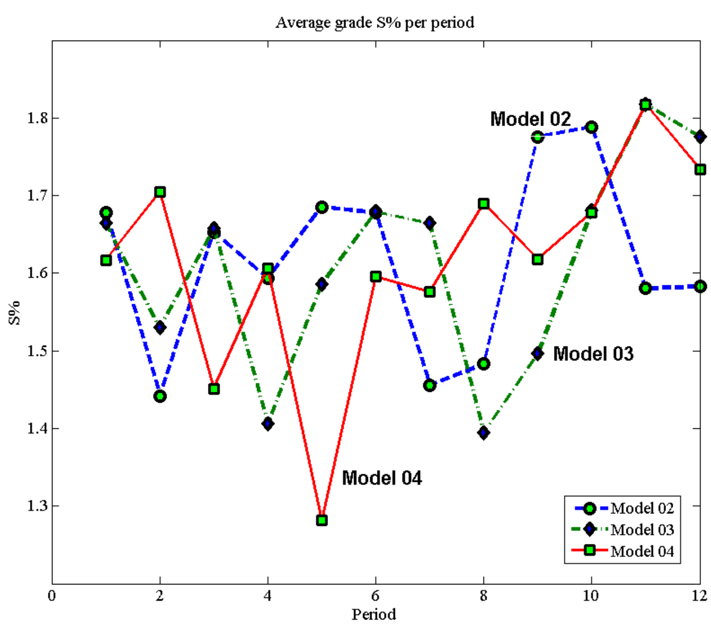

Fig. 10. Average sulphur grade per period.

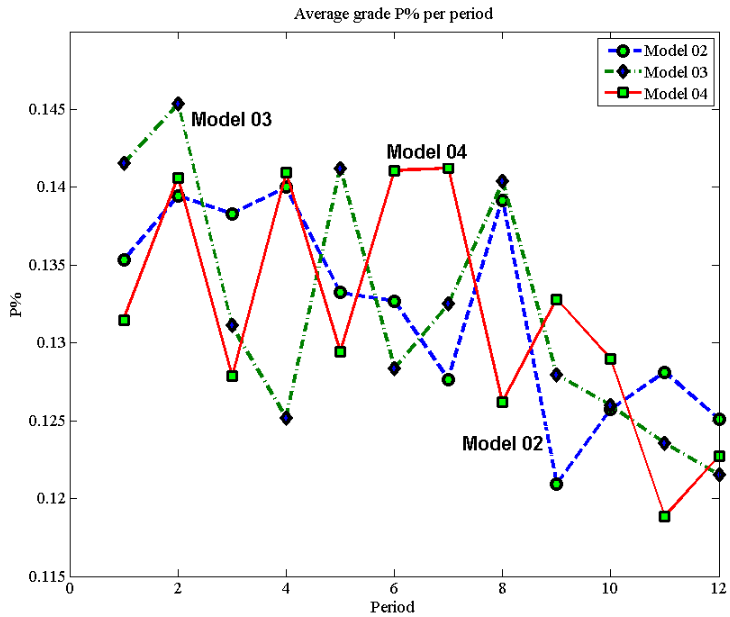

Fig. 11. Average phosphor grade per period.

6. Conclusions and future work

The paper investigated the shortcomings of the current mixed integer linear programming (MILP) models used for open pit production scheduling, particularly the inability to solve large-scale real-size mining problems. In this study, we have developed, implemented, and tested MILP theoretical frameworks for large-scale open pit production scheduling. The developed models proved to be able to handle deterministic large-scale mine production problems.

To reduce the size of the open pit production scheduling problem we introduced the concept of mining-cuts into the MILP formulations. Blocks within the same level or mining bench are grouped into clusters based on their attributes, spatial location, rock type, and grade distribution. Four MILP formulations are presented: Model 01- only consists of binary integer decision variables and generates a schedule at block level resolution; Model 02 – the schedule is generated based on a strict temporal sequence of blocks. The formulation uses continuous variables to model extraction and processing at block level. Binary integer decision variables, are used to control precedence of extraction; Model 03 – processing is controlled at block level with continuous decision variables and the precedence of extraction of blocks is controlled at the mining-cut level by means of binary integer variables; Model 04 –Extraction, processing, and order of block extraction are controlled at mining-cut level. We have implemented the optimization formulations in TOMLAB/CPLEX (Holmström, 1989-2009) environment.

An iron ore mine intermediate scheduling case study over twelve periods was carried out to compare, verify and validate the models. The summary of the comparative analyses revealed that: (i) an MILP formulation at block level resolution (Model 02) is not suitable for long-term scheduling. Although, the block level formulation generates a higher NPV compared to the rest of the models, they would quickly go out of memory and it is almost impossible to generate a life-of-mine schedule for a real-size mine; (ii) block level resolution models (Model 02 and Model 03) are more appropriate for short-range scheduling where the number of blocks are in the order of thousands to ten thousand and the scheduling periods are in the order of ten to twelve periods. These formulations could be used to break-down the long-term yearly schedule into a monthly schedule; (iii) introduction of mining-cuts as the mining units (Models 03 and 04) and as a means to control the precedence of block extraction would drastically reduce the size and runtime of the MILP formulations and the number of binary integer variables required; (iv) MILP formulations based on processing and extraction at mining-cut level (Model 04) leads to practical mathematical tools addressing life-of-mine schedules of large-scale open pit operations which was impossible to solve with the previous formulations. These models provides control over all the mining, processing, and blending constraints, to the mine planner while *maximizing* the NPV; (vi) clustering algorithms and the number of mining-cuts affect the computational efficiency of developed MILP formulations and the generated optimal NPV.

Further focused research is underway to develop and test different clustering techniques that would generate an optimized clustering approach for mining-cuts. Also the next step is to extend the mixed integer linear programming frame work into stochastic mathematical programming domain to address the geological uncertainty issue.

7. References

[1] Askari-Nasab, H., (2006), "Intelligent 3D interactive open pit mine planning and optimization", PhD Thesis Thesis, © University of Alberta, Edmonton, Canada, Pages 167.

[2] Askari-Nasab, H. and Awuah-Offei, K., (2009), "Open pit optimization using discounted economic block value", *Transactions of the Institution of Mining and Metallurgy. Section A, Mining industry*, Vol. 118, 1, pp. 1-12.

[3] Askari-Nasab, H., Frimpong, S., and Awuah-Offei, K., (2005), "Intelligent optimal production scheduling estimator", *in Proceedings of 32nd Application of Computers and Operation Research in the Mineral Industry*, © Taylor & Francis Group, London, Tucson, Arizona, USA, pp. 279-285.

[4] Askari-Nasab, H., Frimpong, S., and Szymanksi, J., (2007), "Modeling open pit dynamics using discrete simulation", *International Journal of Mining, Reclamation and Environment*, Vol. 21, 1, pp. 35- 49.

[5] Askari-Nasab, H., Frimpong, S., and Szymanksi, J., (2008), "Investigating the continuous time open pit dynamics", *The Journal of the South African Institute of Mining and Metallurgy*, Vol. 108, 2, pp. 61-73.

[6] Askari-Nasab, H. and Szymanski, J., (2007), "Open pit production scheduling using reinforcement learning", *in Proceedings of 33rd International Symposium on*

Computer Application in the Minerals Industry (APCOM), © GECAMIN LTDA, Santiago, Chile, pp. 321-326.

[7] Bixby, R. E., (1987-2009), "ILOG CPLEX", ver. 11.0, Sunnyvale, CA , USA: ILOG, Inc.

[8] Boland, N., Dumitrescu, I., Froyland, G., and Gleixner, A. M., (2009), "LP-based disaggregation approaches to solving the open pit mining production scheduling problem with block processing selectivity", *Computers and Operations Research*, Vol. 36, 4, pp. 1064-89.

[9] Caccetta, L. and Hill, S. P., (2003), "An application of branch and cut to open pit mine scheduling", *Journal of Global Optimization*, Vol. 27, November, pp. 349 - 365.

[10] Dagdelen, K. and Kawahata, K., (2007), "Oppurtunities in Multi-Mine Planning through Large Scale Mixed Integer Linear Programming Optimization", *in Proceedings of 33rd International Symposium on Computer Application in the Minerals Industry (APCOM)*, © GECAMIN LTDA, Santiago, Chile, pp. 337-342.

[11] Datamine Corporate Limited, (2008), "NPV Scheduler", ver. 4, Beckenham, United Kingdom: Datamine Corporate Limited,.

[12] Denby, B. and Schofield, D., (1994), "Open-pit design and scheduling by use of genetic algorithms", *Transactions of the IMM Section A*, Vol. 103, January - April 1994,, pp. A21-A26.

[13] Denby, B., Schofield, D., and Hunter, G., (1996), "Genetic algorithms for open pit scheduling - extension into 3-dimensions", *in Proceedings of 5th International Symposium on Mine Planning and Equipment Selection*, © A.A.Balkema/Rotterdam/Brookfield, Sao Paulo, Brazil, pp. 177-186.

[14] Gemcom Software International, I., (1998-2008), "Whittle strategic mine planning software", ver. 4.2, Vancouver, B.C.: Gemcom Software International.

[15] Gershon, M., (1987), "Heuristic approaches for mine planning and production scheduling", *Geotechnical and Geological Engineering*, Vol. 5, 1, pp. 1-13.

[16] Gershon, M. E., (1983), "Mine scheduling optimization with mixed integer programming", *Mining Engineering*, Vol. 35, 4, pp. 351-354.

[17] Holmström, K., (1989-2009), "TOMLAB /CPLEX", ver. 11.2, Pullman, WA, USA: Tomlab Optimization.

[18] Horst, R. and Hoang, T., (1996), "Global optimization : deterministic approaches", © Springer, Berlin ; New York, 3rd ed, Pages xviii, 727 p.

[19] ILOG Inc, (2007), "ILOG CPLEX 11.0 User's Manual September", ver. 11.0: ILOG S.A. and ILOG, Inc.

[20] Isaaks, E. H., (1990), "The application of Monte Carlo methods to the analysis of spatially correlated data, Ph.D. thesis," PhD Thesis, © Stanford University, Stanford, CA, USA, Pages 226.

[21] Johnson, T. B., (1969), "Optimum open-pit mine production scheduling", *in Proceedings of Proceedings, 8th International Symposium on Computers and Operations Research*, © Salt Lake City, Utah, USA, pp. 539-562.

[22] Johnson, T. B. and Barnes, R. J., (1988), "Application of the maximal flow algorithm to ultimate pit design", in *Engineering design : better results through operations research methods*, Vol. 8, *Publications in operations research series*, R. R. Levary, Ed. New York, © North-Holland, pp. xv, 713.

[23] Kaufman, L. and Rousseeuw, P. J., (1990), "Finding groups in data : an introduction to cluster analysis", © Wiley, New York, Pages xiv, 342 p.

[24] Krige, D. G., (1951), "A statistical approach to some basic mine valuation and allied problems at the Witwatersrand", MSc Thesis, © University of Witwatersrand, South Africa,

[25] Lerchs, H. and Grossmann, I. F., (1965), "Optimum design of open-pit mines", *The Canadian Mining and Metallurgical Bulletin, Transactions*, Vol. LXVIII, pp. 17-24.

[26] MathWorks Inc., (2007), "MATLAB Software", ver. 7.4 (R2007a): MathWorks, Inc.

[27] Minemax Pty Ltd, (1998-2009), "MineMax Scheduler", ver. 4.2, West Perth, Western Australia: Minemax Pty Ltd.

[28] Ramazan, S., (2007), "Large-scale production scheduling with the fundamental tree algorithm - model, case study and comparisons", *in Proceedings of Orebody Modelling and Strategic Mine Planning*, © The Australian Institute of Mining and Metallurgy, Perth, Western Australia, pp. 121-127.

[29] Ramazan, S., Dagdelen, K., and Johnson, T. B., (2005), "Fundamental tree algorithm in optimising production scheduling for open pit mine design", *Mining Technology : IMM Transactions section A*, Vol. 114, 1, pp. 45-54.

[30] Ramazan, S. and Dimitrakopoulos, R., (2004), "Traditional and new MIP models for production scheduling with in-situ grade variability", *International Journal of Surface Mining, Reclamation & Environment*, Vol. 18, 2, pp. 85-98.

[31] Runge Limited, (1996-2009), "XPAC Autoscheduler", ver. 7.8: Runge Limited.

[32] Stone, P., Froyland, G., Menabde, M., Law, B., Pasyar, R., and Monkhouse, P. H. L., (2007), "Blasor - Blended Iron ore mine planning optimization at Yandi, Western Australia", *in Proceedings of Orebody Modelling and Strategic Mine Planning*, © The Australian Institute of Mining and Metallurgy, Perth, Western Australia,

[33] Sutton, R. S. and Barto, A. G., (1998), "Reinforcement learning, an Introduction", © The MIT Press, Cambridge, Massachusetts, Pages 432.

[34] Tolwinski, B. and Underwood, R., (1996), "A scheduling algorithm for open pit mines", *IMA Journal of Mathematics Applied in Business & Industry*, 7, pp. 247-270.

[35] Whittle, G., (2007), "Global asset optimization", *in Proceedings of Orebody Modelling and Strategic Mine Planning*, © The Australian Institute of Mining and Metallurgy, Perth, Western Australia, pp. 331-336.

[36] Winston, W. L., (1995), "Introduction to mathematical programming : applications and algorithms", © Duxbury Press, Belmont, Ca., 2nd ed, Pages xv, 818, 39 p.

[37] Wolsey, L. A., (1998), "Integer programming", © J. Wiley, New York, Pages xviii, 264 p.

[38] Yegulalp, T. M. and Arias, J. A., (1992), "A fast algorithm to solve the ultimate pit limit problem", *in Proceedings of 23rd APCOM Symposium*, © AIME, Littleton, Colorado, pp. 391-397.

[39] Zhao, Y. and Kim, Y. C., (1992), "A new optimum pit limit design algorithm", *in Proceedings of 23rd APCOM Symposium*, © SME, Littleton, Colorado, University of Arizona, pp. 423-434.

8. Appendix

MATLAB and TOMLAB/CPLEX code and documentation for Model 03.

The code is for processing at block level and mining at mining-cut level.