Open-Pit Mine Production Scheduling: Improvements to MineLib Library Problems

E. Jélvez, N. Morales and P. Nancel-Penard

1 Introduction

In open-pit mines, mineral is reached by digging material from the ground and then either processing or depositing it on stockpiles for later processing while waste material is deposited on dumps. To define which part of the mine should be extracted at each period of the lifetime of the mine, the terrain is modeled as a three-dimensional array of regular blocks and the planning horizon is discretized into time periods. For each block, estimations on the ore content, density, and other relevant attributes are constructed by using geostatistical methods [1].

The set of all blocks and their attributes form the so-called *block model*. Hence for each block, it is possible to specify: an extraction period, and a destination for processing, defining a *block scheduling*. The final value of a mine is, therefore, determined by the set of attributes and the block scheduling. The feasibility of a block scheduling for the open-pit method depends on accessibility and extraction constraints.

The extraction process must ensure the stability of the walls, which is expressed in terms of slope angles that must be satisfied at each moment (slope precedence constraints) as it follows the sequential extraction of blocks. In addition, there are certain limitations that are inherent to the process, for example, the amount of material to be transported and processed during each period is subject to lower and upper bounds given by transportation and plant capacity, respectively, which are usually expressed either in maximum tonnage or time available for transporting or processing. There exist other optional constraints (named general side constraints) that should be applied, including: (i) blending constraints, because the efficiency, feasibility (or even for regulatory reasons) of the plant process depends on the attributes of the

E. Jélvez · N. Morales · P. Nancel-Penard (✉)
Advanced Mining Technology Center & Delphos Mine Planning Laboratory & Department of Mining Engineering, Universidad de Chile, Santiago, Chile
e-mail: pierre.nancel@amtc.uchile.cl

E. Widzyk-Capehart et al. (eds.), *Proceedings of the 27th International Symposium on Mine Planning and Equipment Selection - MPES 2018*,
https://doi.org/10.1007/978-3-319-99220-4_18

combination of blocks that are processed at a given period and (ii) maximum vertical advance, among others.

Depending on the number of considerations included in the production scheduling model, Espinoza et al. summarized three specific problems [2]. First, the simplest problem in open-pit mine production planning is called Ultimate Pit (UPIT) Limit Problem and includes the selection of a subset of blocks that contains the maximum undiscounted value under slope precedence constraints. The time is not considered in this problem. Second, a generalized extension of the ultimate pit limit problem is the Constrained Pit Limit Problem (CPIT). This model incorporates temporal dimension, scheduling blocks for extraction over a fixed number of periods, maximizing discounted value under both slope precedence and capacity constraints, in which block destinations are fixed in advance. The Precedence Constrained Production Scheduling Problem (PCPSP) extends the last one mainly by considering multiple possible destinations for the blocks (therefore the model decides which one is the optimal choice) and respecting general side constraints, such as blending (where the quality of processed material is controlled).

Currently, a number of mine planning software developers are implementing the pseudoflow algorithm for UPIT (see [3, 4]) to compute both ultimate pit limit and nested pits, which were demonstrated to be more efficient than Lerchs and Grossmann algorithm [5].

While several open-pit block scheduling instances were published in MineLib [2], which presented good feasible solutions for CPIT and PCPSP by using the TopoSort algorithm presented in [6], other authors have proposed new methods and reported best-known solution when applied to CPIT instances of MineLib. Lamghari et al. [7] proposed a method to improve an initial feasible solution based on a local search algorithm called Variable Neighborhood Descent. Jélvez et al. presented an aggregation/disaggregation heuristic to generate good feasible solutions [8]. Liu and Kozan developed two new graph-based algorithms based on network flow graph and conjunctive graph theory, classified as topological ordering-based methods as well [9]. Samavati et al. outperform the TopoSort heuristic strengthening the LP relaxation of CPIT and generating better expected extraction times [10]. A similar approach was developed by [11]. Table 1 shows the best-known solutions for CPIT instances on MineLib expressed in terms of optimality gap.

PCPSP was first studied by Bienstock and Zuckerberg [12], who proposed a method based on Lagrangian relaxation to solve the linear relaxation of the PCPSP and reported a substantial computation time improvement with regards to the standard LP solvers. Espinoza et al. [2] also applied the TopoSort heuristic to PCPSP instances, but they did not generate a feasible solution on W23 (the only instance including blending constraints). Kenny et al. [13] reported improved solutions for some PCPSP instances by using a Greedy Randomized Adaptive Search Procedure, however the improvements do not include a feasible solution for W23 instance.

Most of the real instances of the CPIT and PCPSP in the mining industry are difficult to solve with block models containing large number of blocks for a time horizon that can be as long as several decades. This paper focuses on PCPSP, a new full-binary formulation and an improvement to the current best-known results for

Table 1 Current best-known solution on 11 CPIT instances available in MineLib library

CPIT instance	Source	GAP (%)
Newman	[10]	1.26
Zuck_small	[11]	0.71
KD	[11]	0.14
Zuck_medium	[11]	5.24
P4HD	[9]	0.08
Marvin	[11]	0.64
W23	[9]	0.19
Zuck_large	[8]	0.24
SM2	[7]	0.04
McLaughlin_lim	[10, 11]	0.06
McLaughlin	[8, 10, 11]	0.06

PCPSP instances on MineLib, which was made by means of a heuristic based on both a sliding time window and a linear relaxation to preselect a small subset of blocks to be scheduled within each time window. The approach includes blending constraints in its solution.

2 Mathematical Modeling

In this section, the main notation and the mathematical formulation of the optimization model used in this work are introduced. The formulation is referred to as the open-pit block scheduling problem (OPBSP). The only difference from the PCPSP is that the blocks cannot be split and sent to different destinations, hence this problem is fully binary and not mixed. However, the solutions are feasible for PCPSP as well.

2.1 Notation

Let us consider a set of blocks B. The elements of B (the blocks) are denoted by letters b, b' unless otherwise stated. The set of periods is denoted by T, hence the production is scheduled in periods $t = 1, \ldots, |T|$. There exists a set of destinations D (each destination is coded by a number, so the possible destinations for a block are $d = 1, \ldots, D$).

The net benefit perceived if a block $b \in B$ is sent to destination $d \in D$ at time period t is given by v_{bdt}. The block values will be denoted by $V(B, D, T) = (v_{\text{bdt}})_{b \in B, d \in D, t \in T}$ or simply V if there is no ambiguity. We consider two sets of attributes, namely A and $\tilde{A}$: A refers to the block attributes that participate in capacity constraints, like tonnage while $\tilde{A}$ relates to the attributes that are averaged (blending constraints), such as grades or pollutant contents. The value of attribute $a \in A$ (or $\tilde{a} \in \tilde{A}$) in block b is denoted by g_{ba} (or $h_{b\tilde{a}}$), where $g_{ba} \geq 0$ (and similarly $h_{b\tilde{a}} \geq 0$) when the attributes denote tonnage and concentrations. Some constraints are applied on a subset of destinations $\delta \subseteq D$, for example, for processing. For each $a \in A$ and $\delta \subseteq D$, a minimum capacity (thus a demand) $L_{a\delta t} \in \mathbb{R}$ and a maximum capacity $U_{a\delta t} \in \mathbb{R} \cup \{+\infty\}$ are imposed. Similarly, for each $\tilde{a} \in \tilde{A}$, minimum $\tilde{L}_{a\delta t} \in \mathbb{R}$ and maximum $\widetilde{U}_{\tilde{a}\delta t} \in \mathbb{R} \cup \{+\infty\}$ average values are allowed.

An attribute ton_b representing the tonnage of block b is used as a weight for computing averages. Due to stability requirements, slope constraints are given by one or several slope angles that define the maximum slopes that are possible in pit walls. The standard way to model these slope constraints is using precedencies as follows: for any given block b, there exists a set of other blocks (called predecessors) that must be mined before in order to gain access to block b. A very general way to encode this is by defining a set of arcs $P \subset B \times B$, where $(b, b') \in P$ means that block b' (predecessor of block b) has to be extracted in the previous or the same period that block b (successor of block b').

2.2 An Alternative Formulation for PCPSP Model: OPBSP

This subsection introduces a new formulation for PCPSP. The decision variables are related to the decision of whether to mine or not a given block, when to do so, and what destination is chosen for that block. The objective function is to maximize net present value. The constraints considered are: structural (related to the nature of the variables), precedence, capacity and general side (blending). For each block $b \in B$, destination $d \in D$ and period $t \in T$, the variable is defined in (1):

$$x_{\mathrm{bdt}} = \begin{cases} 1 \text{ if block } b \text{ is to a destination } d' < d \text{ at period } t, \\ \quad \text{or sent to any detination at some period } t' < t. \\ 0 \text{ otherwise} \end{cases} \tag{1}$$

To keep the notation simple, auxiliary variables Δx_{bdt} representing the exact notion of a block b sent to destination d at period t are defined:

$$\Delta x_{\mathrm{bdt}} = \begin{cases} x_{\mathrm{bdt}} & d = t = 1 \\ x_{\mathrm{bdt}} - x_{bD(t-1)} & d = 1, t > 1 \\ x_{\mathrm{bdt}} - x_{b(d-1)t} & d > 1 \end{cases} \tag{2}$$

For a block model B, precedence arcs P, set of destinations D, set of time periods T, block values $V = V(B, D, T)$, sets of capacity $C = C(B, A, D, T)$ and blending $\tilde{C} = \tilde{C}(B, A, D, T)$ constraints, an open-pit block scheduling problem $\mathbf{OPBSP}\left(B, P, D, V, T, C, \tilde{C}\right)$ is defined as

$$\text{Max} \sum_{b \in B} \sum_{d \in D} \sum_{t \in T} v_{\text{bdt}} \Delta x_{\text{bdt}} \tag{3}$$

subject to

$$x_{\text{bDt}} \leq x_{\text{b}'\text{Dt}} \quad \forall \left(b, b'\right) \in P, t \in T \tag{4}$$

$$\Delta x_{\text{bdt}} \geq 0 \quad \forall b \in B, d \in D, t \in T \tag{5}$$

$$\sum_{b \in B} \sum_{d \in \delta} g_{ba} \Delta x_{\text{bdt}} \leq U_{a\delta t} \quad \forall a \in A, \delta \subset D, t \in T \tag{6}$$

$$\sum_{b \in B} \sum_{d \in \delta} g_{ba} \Delta x_{\text{bdt}} \geq L_{a\delta t} \quad \forall a \in A, \delta \subset D, t \in T \tag{7}$$

$$\frac{\sum_{b \in B} \sum_{d \in \delta} h_{b\tilde{a}} ton_b \Delta x_{\text{bdt}}}{\sum_{b \in B} \sum_{d \in \delta} ton_b \Delta x_{\text{bdt}}} \leq \tilde{U}_{\tilde{a}\delta t} \quad \forall \tilde{a} \in \tilde{A}, \delta \subset D, t \in T \tag{8}$$

$$\frac{\sum_{b \in B} \sum_{d \in \delta} h_{b\tilde{a}} ton_b \Delta x_{\text{bdt}}}{\sum_{b \in B} \sum_{d \in \delta} ton_b \Delta x_{\text{bdt}}} \geq \tilde{L}_{\tilde{a}\delta t} \quad \forall \tilde{a} \in \tilde{A}, \delta \subset D, t \in T \tag{9}$$

$$x_{\text{bdt}} \in \{0, 1\} \quad \forall b \in B, d \in D, t \in T \tag{10}$$

Equation (3) presents the objective function, which is the discounted benefit from the extracted blocks over time horizon $|T|$. Equation (4) corresponds to the precedence constraints given by the slope angle and Eq. (5) means that the definition of the variables is satisfied. Moreover, Eqs. (6) and (7) limit the maximum and minimum resource consumption in each period, respectively. Equations (8) and (9) represent the blending constraints, and Eq. (10) establishes that all variables assume binary values.

The main difference between PCPSP (as presented in [2, 12]) and OPBSP relates to the fact that in OPBSP blocks cannot be split and thus a given extracted block is sent to only one destination. However, OPBSP solutions are feasible for PCPSP as well.

3 An Incremental Heuristic Based on Expected Time

Expected Time Incremental Heuristic (ETInc) is the proposed algorithm to approximate the solution of OPBSP and consists in a combination of an incremental heuristic that works on a subset of blocks and periods by using a sliding time window (as in [8]) plus expected extraction times computed from the linear relaxation of the problem as introduced in [6]. A more detailed version of this heuristic may be found in [14].

3.1 Incremental Heuristic Based on Sliding Time Window

The heuristic iteratively constructs a schedule for each period by solving the OPBSP for a time window $\tilde{T} = \{t, \ldots, \min(t + w - 1, |T|)\}$ limited to $w \leq |T|$ periods, starting from period $t = 1$ of the planning horizon, where w is an integer parameter used to determine the maximum length of the time window. Each time the OPBSP subproblem is solved, the variables x_{bdt} are fixed for the first w' periods of the incumbent time window, where w' is a parameter to be determined, the time window is moved forward by w' periods, and the OPBSP subproblem is solved for the new time window. The procedure stops when the last OPBSP subproblem corresponding to the period $t = |T|$ has been solved.

3.2 Block Preselection Using Expected Extraction Times

To solve each OPBSP subproblem, the heuristic preselects a subset $\tilde{B}$ of blocks based on a modified definition of the expected extraction time introduced by [6] according to the following procedure.

Let $\bar{x}^*_{\text{bDt}}$ be the solution of the LP relaxation of the original OPBSP. The expected extraction time E_b of block b is defined as

$$E_b = \sum_{t \in T} t \cdot \Delta\bar{x}^*_{\text{bDt}} + (T + 1)\left(1 - \bar{x}^*_{\text{bDt}}\right) \tag{11}$$

A subset of blocks $\tilde{B}$ not yet extracted at period t is defined for which expected time E_b is smaller than $\min(t + w - 1, |T|) + s$, where $s > 0$ is a continuous parameter to be determined representing a tolerance or additional margin in the selection. In this procedure, the expected times are used as a block preselection tool to reduce the size of the subproblems, they are not used to generate a sequence of blocks as in the TopoSort heuristic proposed by [6].

3.3 Expected Time Incremental Heuristic—ETInc

ETInc algorithm depends on three parameters:

1. w, which is the length of the sliding time window,
2. w', which is the number of periods to be fixed in the current solution of OPBSP subproblem, where $w' \leq w$, and
3. s, representing the tolerance parameter to select a subblock model $\tilde{B}$ based on expected extraction times.

The main steps of the ETInc algorithm can be described as follows:

Table 2 List of PCPSP instances from MineLib. ETInc parameters used in the experiments

PCPSP instance	ETInc parameters		
	w	w'	s
Newman	1	1	1.5
Zuck_small	6	1	0.5
KD	2	1	0.5
Zuck_medium	2	1	5.5
Marvin	5	1	0.5
W23	1	1	0.5
Zuck_large	4	1	0.5
SM2	1	1	0.5
McLaughlin_lim	2	1	0.5
McLaughlin	2	1	0.5

1. Select a new time window $\tilde{T}$ according to Sect. 3.1.
2. Select a subblock model $\tilde{B}$ according to Sect. 3.2.
3. Construct an auxiliary instance of OPBSP (or OPBSP sub-problem) by using $\tilde{B}$ and $\tilde{T}$ and solve it.
4. Select blocks for extraction.
5. If not finished, go to step 1.

4 Numerical Experiments

The datasets for all instances can be found at [2]. The list of 10 PCPSP instances on which the algorithm was applied and the parameters used are presented in Table 2. The computational resources consisted of a core i5-3570, 3.4 GHZ PC with 16 GB of RAM, and GUROBI 6.5.2 was used as optimization software.

4.1 Results and Discussion

The results obtained from the numerical experiments and a comparison with the corresponding best-known results for PCPSP instances from MineLib are presented in this section.

Table 3 shows the value of the solutions for the linear relaxation obtained for each instance of PCPSP, as reported in [2], and OPBSP, which were obtained by implementing the Bienstock-Zuckerberg (BZ for short) algorithm. From the theoretical point of view, these values should be equal, however, there are very small differences, being the largest relative difference for W23 smaller than 4×10^{-6}. This

Table 3 List of LP upper bounds obtained for PCPSP and OPBSP

PCPSP instance	LP solution value		
	PCPSP	OPBSP	Abs. difference
Newman	24,486,549	24,486,549	0
Zuck_small	905,878,172	905,878,194	22
KD	410,891,003	410,891,003	0
Zuck_medium	750,519,109	750,519,188	79
Marvin	911,704,665	911,704,801	136
W23	387,693,394	387,691,933	1461
Zuck_large	57,938,790	57,938,804	14
SM2	1,652,394,327	1,652,394,357	30
McLaughlin_lim	1,324,829,727	1,324,829,835	108
McLaughlin	1,512,971,680	1,512,971,772	92

Table 4 Current best-known solution on 10 PCPSP instances available in MineLib library

Instance name	Source	Gap (%)	Best-known OPBSP objective	Gap (%)
Newman	[13]	1.58	24,176,861	**1.26**
Zuck_small	[13]	1.64	897,453,456	**0.93**
KD	[2]	0.98	409,715,160	**0.29**
Zuck_medium	[13]	**3.00**	701,157,160	6.58
Marvin	[13]	1.61	905,829,721	**0.64**
W23	–	100.00	368,005,675	**5.08**
Zuck_large	[2]	1.04	57,534,355	**0.70**
SM2	[2]	0.12	1,651,599,491	**0.05**
McLaughlin_lim	[2]	0.24	1,322,283,576	**0.19**
McLaughlin	[2]	0.19	1,510,373,891	**0.17**

is explained because different stopping criteria of the implementations of the BZ algorithm were used. Therefore, the differences between LP upper bounds are small enough not affect the optimality gap defined as

$$\text{Gap} = (\text{LP upper bound} - \text{best-known solution objective})/\text{LP upper bound} \quad (12)$$

Table 4 shows the current best-known feasible solutions for each instance as reported in [2, 13] in terms of optimality gap, the objective values of the feasible solutions obtained using ETInc and their respective optimality gaps. All feasible solutions (except Zuck_medium instance) implemented to improve on the already existing values and particularly that for the instance W23, ETInc was able to produce a solution with 5.1% optimality gap for OPBSP, therefore, improving on the current trivial null solution.

5 Conclusions

A new full-binary formulation for the Precedence Constrained Production Scheduling Problem (PCPSP) was presented. In this formulation (OPBSP), the blocks cannot be partitioned, therefore, only one processing destination must be chosen for each block.

An algorithm (ETInc) that aims to produce good feasible solutions for OPBSP, and therefore for the PCPSP, was used. ETInc is similar to other algorithms proposed in the literature as it uses the solution of the linear relaxation as a guide to generate integer feasible solutions by constructing a ranking of blocks for extraction and by resorting to solutions for auxiliary instances. ETInc was applied to a publicly available library of instances included in MineLib, which consists of 10 different cases of variable size, obtaining better results for the 9 out of 10 cases.

Further research is required for the library of problems. For example, even though both formulations OPBSP and PCPSP accept lower bounds for capacity constraints, the library does not have this type of constraint. In this sense, it is a challenge to expand the number of case studies or instances available in MineLib to evaluate new models and compare new algorithms.

Additional research areas should consider the inclusion of the uncertainty in market and geology as well as the improvements in the computational time and memory footprint.

Acknowledgements The authors acknowledge the support of CONICYT Basal Project FB0809—AMTC, Universidad de Chile.

References

1. Chiles, J., Delfiner, P.: Geostatistics: modeling spatial uncertainty, vol. 497. Wiley, Hoboken (2009)
2. Espinoza, D., Goycoolea, M., Moreno, E., Newman, A.: Minelib: a library of open pit mining problems. Ann. Oper. Res. **206**(1), 93–114 (2013)
3. Hochbaum, D.: The pseudoflow algorithm: a new algorithm for the maximum-flow problem. Oper. Res. **56**(4), 992–1009 (2008)
4. Hochbaum, D., Orlin, J.: Simplifications and speedups of the pseudoflow algorithm. Networks **61**(1), 40–57 (2013)
5. Lerchs, H., Grossmann, I.F.: Optimum design of open-pit mines. Trans. Can. Inst. Min. **68**(1), 17–24 (1965)
6. Chicoisne, R., Espinoza, D., Goycoolea, M., Moreno, E., Rubio, E.: A new algorithm for the open-pit mine production scheduling problem. Oper. Res. **60**(3), 517–528 (2012)
7. Lamghari, A., Dimitrakopoulos, R., Ferland, J.: A hybrid method based on linear programming and variable neighborhood descent for scheduling production in open-pit mines. J. Glob. Optim. **63**(3), 555–582 (2015)
8. Jélvez, E., Morales, N., Nancel-Penard, P., Peypouquet, J., Reyes, P.: Aggregation heuristic for the open-pit block scheduling problem. Eur. J. Oper. Res. **249**(3), 1169–1177 (2016)
9. Liu, S., Kozan, E.: New graph-based algorithms to efficiently solve large scale open pit mining optimisation problems. Expert Syst. Appl. **43**, 59–65 (2016)

10. Samavati, M., Essam, D., Nehring, M., Sarker, R.: A methodology for the large scale multi-period precedence-constrained knapsack problem: an application in the mining industry. Int. J. Prod. Econ. **193**, 12–20 (2017)
11. Samavati, M., Essam, D., Nehring, M., Sarker, R.: A new methodology for the open-pit mine production scheduling problem. Omega (2017b) In press
12. Bienstock, D., Zuckerberg, M.: Solving LP relaxations of large-scale precedence constrained problems. In: Proceedings of the International Conference on Integer Programming and Combinatorial Optimization, pp. 1–14. Springer, Heidelberg (2010)
13. Kenny, A., Li, X., Ernst, A., Thiruvady, D.: Towards solving large-scale precedence constrained production scheduling problems in mining. In: Proceedings of the Genetic and Evolutionary Computation Conference, pp. 1137–1144. ACM, New York (2017)
14. Jélvez, E., Morales, N., Nancel-Penard, P., Cornillier, F.: A New Hybrid Heuristic for the Precedence Constrained Production Scheduling Problem: A Mining Application. Submitted paper