Optimizing pushback design considering minimum mining width for open pit strategic planning

Pierre Nancel-Penard[a] and Nelson Morales[a]

[a]DELPHOS Mine Planning Laboratory, Department of Mining Engineering and Advanced Mining Technology Center (AMTC), University of Chile. Av. Tupper 2069, 8370448 Santiago, Chile.

ARTICLE HISTORY
Compiled March 28, 2021

Abstract
Open pit mines contain ore that is to be extracted from the surface downwards to obtain a profit. A minimum mining width is necessary for mining equipment to extract the material from the surface. Considering that current industry practice lacks a standard methodology for optimizing these designs, it may lead to inconsistent, subjective, and suboptimal results. Automatic design of practicable portions of the mines called pushbacks is a desirable tool for the mining industry. This article proposes a new approach based on an integer linear programming model (ILP) that determines which blocks should be extracted to maximize profit while respecting geospatial and design constraints. Design constraints require that the minimum width be respected at the bottom of the mine and between successive pushbacks. The results obtained in minutes by the application of this ILP to three cases are studied and they show the applicability of this programming model.

KEYWORDS
Minimum mining width; pushback design; Mathematical modelling; Integer linear programming

1. Introduction

In this article, we address the problem of open pit mine design, which consists of extracting valuable materials by digging from the surface downwards. This extraction method is very relevant to the mining industry and the Chilean economy.

To address this problem, mine planners rely on a discretization of the ore deposit into a 3-D array of blocks. This set of blocks and their attributes, such as ore grades, tonnages, recoveries, and others, form a database named the *block model*. This block model, together with the geotechnical constraints related to the pit wall slopes and the long-term economic parameters (such as costs and commodity prices), are the basic inputs for open pit strategic mine planning. From these data, mine planners determine a design for the mine operation and a mining schedule that indicates which blocks should be extracted, when this should happen, and what to do with the extracted blocks. These decisions must be made such that the value of the mine be maximized, and therefore, planners rely on models and algorithms to optimize profit.

The first step to design and plan an open pit mine is to compute the *ultimate* or *final pit*,

CONTACT Pierre Nancel-Penard. Email: pierre.nancel@amtc.cl ORCID: 0000-0001-7311-0073

which determines the limits of what is economically mineable from a given deposit. It identifies which blocks should be mined and which ones should be left in the ground (Dagdelen 2001). One particular by-product of the process described above is the definition of a specific set of blocks, namely a pushback, which corresponds to a manageable portion of the deposit inside the ultimate pit limit that may be mined, processed, and refined in a number of years/periods Asad and Topal (2011).

The ultimate pit problem can be formalized as follows. Let $\mathcal{B}$ be the set of blocks and b, b' denote elements of this set. We are given, for each $b \in \mathcal{B}$ an economic value v_b, and a set of *precedence arcs* $\mathcal{P} \subset \mathcal{B} \times \mathcal{B}$ such that $(b, b') \in \mathcal{P}$ means that the stability of the pit walls requires that to be able to extract block b it is also necessary to extract b'. The problem then consists in finding a subset $S \subset \mathcal{B}$ such that it has the maximum undiscounted economic value $\sum_{b \in S} v_b$ and respects the safety of walls, i.e. $\forall (b, b') \in \mathcal{P},\ b \in S \Rightarrow b' \in S$.

In order to split the ultimate pit into smaller manageable volumes, the ultimate pit problem can be used to generate not one but several nested pushbacks. This is done by using the fact that the economic value of each block is a function of the price p, (i.e. $v_b = \phi(p, b)$) and replacing the price by λp, where $\lambda \in]0, 1]$ is called the *revenue factor*. In this way, each value of λ induces a difference instance of the ultimate pit problem, where the economic value of each block is $v_{b,\lambda} = \phi(\lambda p, b)$ (in particular, the ultimate pit corresponds to the case $\lambda = 1$). Nested pushbacks are therefore generated by using the property that if S_λ is the optimal solution for λ, when $\lambda \leq \lambda' \Rightarrow S_\lambda \subset S_{\lambda'}$ and computing the solution for many values of λ.

The ultimate pit problem can be solved efficiently either by the Lerchs and Grossman (1965) algorithm or through its conversion to a max-flow min-cut problem (Picard 1976; Hochbaum and Chen 2000). However, it can also be formulated as an integer linear program. We recall its formulation here, but considering its parametrization in terms of the revenue factor λ, as it is convenient to present its extension later in the article.

Let x_b be a binary variable so that $x_b = 1$ if and only if block $b \in \mathcal{B}$ is extracted. Then to solve the ultimate pit problem with revenue factor λ is equivalent to solve

$$
\begin{array}{lll}
UP(\lambda) & \max & \displaystyle\sum_{b \in \mathcal{B}} v_{b,\lambda} x_b \\
 & & x_b \leq x_{b'} \qquad \forall (b, b') \in \mathcal{P} \\
 & & x_b \in \{0, 1\} \qquad \forall b \in \mathcal{B}
\end{array}
$$

Consider Figure 1, which shows a small example of the ultimate pit problem's potential output with colours representing different excavation levels or *benches* (darker grey colours are for deeper ones). Consider the Komatsu P&H 4100XPC shovel, which has approximate dimensions of 15x15 meters and a cut radius of 24m (Komatsu 2018). It is clear that this shovel does not fit at the deepest level, and it could not move freely at the bench just above due to the one block's isthmus illustrated in the figure. Using this output as a guide for the design, a mine planner would have to decide which part has to be eroded or extended such that the shovel can operate. For this equipment, 30 meters of minimum width or *minimum bottom width* are generally considered for its movements at the bottom of the mine. Therefore, a mine planner would exclude the lowest level from the design, and add more space in the other levels, probably increasing the total waste to be removed, which may produce a negative impact on profit.

In addition to calculating the ultimate pit, mine planners generate many nested pits by varying $\lambda \in]0, 1]$ and then select some specific ones. The chosen pits, which we refer to as *pushbacks* are then used as a basis for the mine design, which is not done with block support

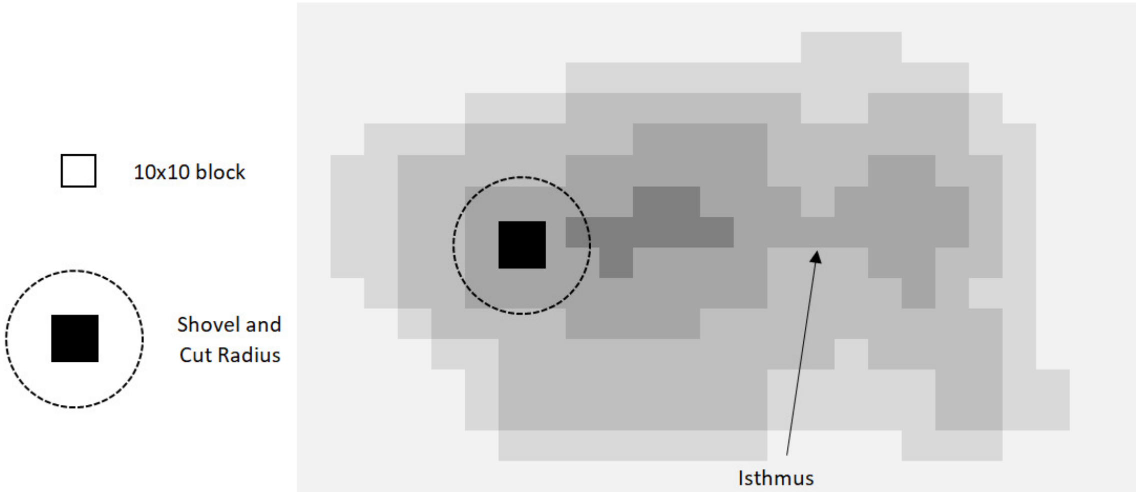


Figure 1. A conceptual 2-D example illustrating relative sizes of a shovel within a pushback profile. (Plan view, darker colours represent deeper mine levels.)

but considering actual mining volumes and other elements, such as access ramps and the use of specialized CAD software. Because the same base model ($UP(\lambda)$) is used to compute all pushbacks, it is clear that the problem of minimum bottom width may arise in each of them. Moreover, there is the issue of leaving enough mining width or *bench width* between successive pushbacks.

All the above makes the selection of pushbacks from the mine pits a complex task that involves simultaneous consideration of different elements, such as the geometry of each pushback, but also the distance that must exist between the contour of subsequent pushbacks to provide space for access and operation of large pieces of equipment, the tonnage of materials they contain, and ultimately that they can support an efficient production schedule, i.e., an extraction order that maximizes the net present value.

Unfortunately, this process, despite of the availability of software that can assist with the computation and selection of the pushbacks, is time-consuming and highly dependent on the user. It follows that there is no guarantee that the set of pushbacks selected without considering such constraints will be near the optimal selection.

To show that (UP) may not provide a good approximation to the problem with constraints at the bottom, let us consider the artificial 2-D example in Figure 2, where safety wall constraints correspond to a slope angle of 45^o. In this example, we have coloured six sets of blocks (orange, red, green, grey, blue, and yellow), labelled A, B, C, D, E, and F, respectively. Sets A and B consists of one block each. For this example, we assume that all blocks (except those in A and B) have a negative economic value. Moreover, we assume that the total economic values of C, D, E, F are $v(C) = -100, v(D) = -50, v(E) = -50$ and $v(F) = -v(A)$, respectively. Let (UP_5) be the ultimate pit problem that considers a minimum bottom width of five blocks. We observe that:

- if $v(A) > 150$, then the optimal solution to (UP) is $P_1 = A \cup B \cup C \cup D$ and has an economic value $v_1 = v(A) + v(B) - 150$;
- if $v(B) > 100$, then the optimal solution to $(UP5)$ is the set $P_2 = B \cup D \cup E$, with economic value $v_2 = v(B) - 100$.

It follows that if $v(A) > 150$ and $v(B) > 100$, the difference between the economic values of

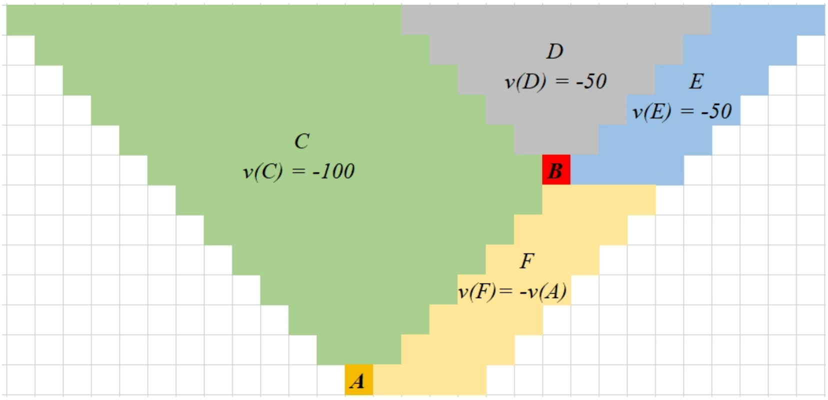


Figure 2. A conceptual 2-D example of the solutions for the ultimate pit problem with and without minimum bottom width. Values of blocks A, B are positive. Depending on the values of blocks A and B, the optimal solution of the ultimate pit and the one considering a minimum bottom width of five blocks can be very different in value and shape.

the optimal solutions is $v_1 - v_2 = v(A) + v(C) - v(E) = v(A) - 50$, which can be large if $v(A)$ is large. Thus, even though the optimal value of (UP) is an upper bound of (UP_5), it may be far from being tight. For example, for $v(A) = 1000$ and $v(B) = 150$, we have $v_1 = 1000$ and $v_2 = 50$ that corresponds to a 95% decrease in economic value. We also notice that $P_2 \not\subseteq P_1$, $P_1 \not\subseteq P_2$, $P1$ is composed of 169 blocks and $P2$ of 60 blocks that corresponds to a 64.5% decrease in the number of blocks.

The main contribution of this work is that we address the issue of modelling the problem of computing a pushback that complies with slope angles and also with a shape that provides enough space for the operation and movement of large pieces of equipment at the bottom of the mine. We achieve this by extending the ultimate pit problem to consider additional constraints related to the bottom of the pit and requiring the design to respect two geometric properties,

- *Continuity* imposes that disconnected parts of the design correspond to different pushbacks. Otherwise, it would result in more haulage access points and increased operational costs due to relocation.
- *Smoothness* is related to the shape of the bottom of the pit and looks to have enough space for operating and moving equipment within the bottom of the pit. The minimum mining width is ensured by preventing the extraction of lonely blocks at the bottom and, instead of that, to force extraction of minimum areas. The space for moving equipment is ensured by not allowing the creation of isthmuses, i.e., narrow passages between operational areas.

An example of the smoothness can be seen in Figure 3, which illustrates the case where the minimum width requirement is of squares of 3×3 blocks. Figure 3(a) shows an unfeasible case because of the one block protuberances that cannot be covered by the squares, Figure 3(b) shows the union of two squares (i.e., without protuberances) that is feasible in terms of minimum bottom width, but not for moving equipment because of the narrow intersection. Finally Figure 3(c) shows a feasible pit bottom, which consists of the case (b) plus two greyed blocks that provide enough space for moving between zones.

Notice that the constraints we incorporate in the problem affect the size and shape of the bottom of the pit, but because of slope precedences, their effects propagate and affect the

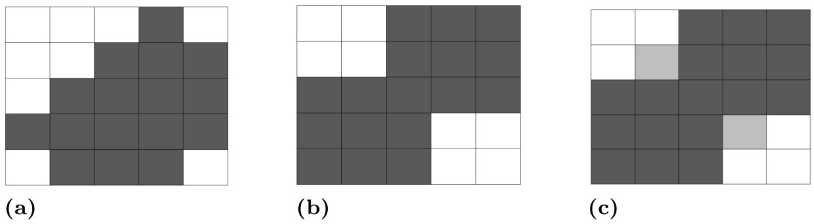


Figure 3. Plan-view of examples of bottom designs for a minimum bottom width of three blocks

whole pushback's shape and size. However, as in the case of the ultimate pit, the mathematical model that we present computes only one pushback at a time. Therefore, it does not address the problem of the minimum width between consecutive pushbacks. In this sense, our model can mimic the current approach to generate multiple pushbacks by changing the revenue factor and then selecting pushbacks with a minimum width between consecutive ones.

The article is organized as follows. In Section 2, we provide a summary of the related works. Section 3 contains all the details concerning the modelling, notation and problem statement. Three different cases are studied in Section 4, showing how the model and techniques can be used to generate nested pits with the geometric properties defined above. Finally, Section 5 contains some concluding remarks and perspectives.

2. Related work

The problem of strategic mine planning must deal with two dimensions: the design of the mine and the optimization of production scheduling. The former is related to the definition of operational volumes that support a proper extraction over time. The latter corresponds precisely to scheduling the extraction and processing of portions of these volumes over time in such a way that the maximum net present value (NPV) is attained.

In order to approach the problem of strategic mine design, the industry has relied on a method consisting of the following steps: (i) compute the ultimate and nested pits (ii) select some pits (now pushbacks) based on several criteria and preliminary production plans, (iii) use the pushbacks as a guide to design the mine, and (iv) optimize production scheduling using updated cost values and designed volumes.

The proposed model aims to improve the results of step (i), by providing better candidates for pushback selection for step (ii) and easing the design at step (iii). Because of this, this literature review focuses not only on mine design aspects but also covers some background on production scheduling.

2.1. *Mine Design*

As reported before, the fundamental problem in mine design consists of determining the pushbacks, for which planners rely on the final pit or ultimate pit problem and its parametrization to determine nested pits. The (UP) problem (formalized in the previous section) was introduced together with an algorithm for its solution by Lerchs and Grossman (1965). Picard (1976) demonstrated that the final pit problem is equivalent to the maximum closure problem, which in turn can be formulated as a minimum cut problem. Hochbaum proposed an efficient polynomial algorithm based on a maximum pseudo-flow approach that solves large minimum cut problems fast (Hochbaum and Chen 2000; Hochbaum 2008; Hochbaum and Orlin 2012).

Lerchs and Grossman (1965) also showed that nested pits could be produced by applying their algorithm by varying the block revenues, therefore providing the basis to address the whole problem using a trial and error approach to satisfy resource constraints, among other requirements. This method is commonly used in commercial software (Whittle 2018). A complete reference to the generation of pushbacks from the nested pits can be found in Meagher, Dimitrakopoulos, and Avis (2014), which focus on the inconsistent sizes between successive pushbacks that may occur, namely the gap problem. Many studies about pushback generation deal with NPV optimization under resource constraints without considering enough geometric requirements (Ramazan and Dagdelen 1998; Consuegra and Dimitrakopoulos 2010; Goodfellow and Dimitrakopoulos 2013; Meagher, Dimitrakopoulos, and Vidal 2014; Asad, Dimitrakopoulos, and van Eldert 2014). The selected pushbacks are then used as guides to plan the phases, including the ramps and minimum mining widths required for the material extraction.

A problem closely related to ours is addressed by Bai et al. (2018). They proposed an approach based on an iterative application of geometric operators to generate feasible pushbacks that respect the minimum bench and bottom widths, smoothness, and continuity. Their approach works sequentially for each pushback in two stages: first, it generates a current pushback through a parametrization of the price and the search of volumes that comply with capacity constraints in such a way that the current pushback contains the previous one. Second, it applies *geometric operators* from the bottom to the top of the mine that looks to repair the shape of the obtained volumes. For this second stage,

- They model the smoothness by a parameter named NS and they require that the pit bottom's shape and its complement (as sets of blocks) contain at least NS consecutive blocks in the X and Y-axis directions. This is different from our approach because, for example, for $NS = 3$, the case Figure 3(b) is feasible in their modelling, even though there is a narrow area between the two squares.
- They address the problem of minimum width by introducing a *wide* area, where they impose the minimum distance between pushbacks, and a *sub-wide* area, where this constraint is relaxed because it could be extracted using smaller equipment but at a higher cost. They define a parameter AW that is the maximum distance between a block in the sub-wide area and the adjacent wide-area. This parameter permits to control the size of this sub-wide area.

The article considers four tools, which are used to construct geometrical operators: (i) dilating, which adds blocks to a pushback to respect the minimum width; (ii) eroding, which removed blocks; (iii) opening, which first erodes and then dilates; and (iv) closing, which does the opposite of opening. The geometric operators are mainly,

- An adaptive opening operator to obtain shapes of the sub-wide area that are triangles of base length AW. For this, they apply an opening considering the minimum width TW and operates AW times the dilating tool lowering the TW parameter iteratively.
- A smoothness operator that uses both opening and closing tools to eliminate cavities and protuberances.
- A continuity operator that computes connected components and keeps the largest one.

Bai et al. (2018) apply their algorithm to two case studies and report a decrease of 0.5% and 7% in the NPV between the initial pushbacks they used and the practicable ones.

Another related work by Tabesh, Mieth, and Askari Nasab (2014) develops a multi-step pushback design algorithm based on mathematical programming and clustering to generate pushbacks composed of mining polygons. They apply a greedy heuristic approach and local search that assigns blocks to pushbacks in such a way that tonnages in each pushback do not exceed maximum values. They use a hierarchical clustering algorithm that generates a lot of small shapes. A shape refinement procedure is provided as a post-treatment to obtain more practicable forms that are considered for the scheduling.

From a practical point of view, software proposes different tools to address the problem of minimum mining width. For example, Geovia Whittle has a mining width module that applies distance templates to respect mining width requirements in the X and Y-axis directions on the bottom and benches of pit shells (Whittle 2018). Juarez et al. (2014) reported a tool in DeepMine software that uses an approximate dynamic programming algorithm that chooses the pushbacks corresponding to the best NPV from a generated tree of alternative practicable pushbacks. However, the details of this methodology are not publicly available.

Finally, it is worth mentioning two complementary works related to optimized mine design. Nancel-Penard et al. (2019) introduce a methodology based on a linear programming model that automatically generate a pushback design at block level that facilitates the space of ramps with the resulting envelope having the maximum undiscounted value. Parra et al. (2017) study the impact of pushback selection in terms of geomechanical constraints and stability of the pit walls.

2.2. *Production Scheduling*

Most studies in this area fall under the application of *direct block scheduling* proposed by Johnson (1968, 1969), which is an alternative to the nested pits methodology that relies on mathematical programming to schedule the extraction of blocks over a set of time-periods. The objective is to maximize the NPV of the set of extracted blocks. The modelling, in this case, considers: precedences due to slope constraints, capacity and blending constraints at each period, and the possibility of sending blocks to different processing facilities. These constraints can also be combined to address extraction limited to predefined pushbacks. Espinoza et al. (2013) introduced a set of publicly available instances for direct block scheduling allowing researchers to compare results from different heuristic approaches (Lamghari, Dimitrakopoulos, and Ferland 2014; Liu and Kozan 2016; Samavati et al. 2017; Jelvez et al. 2016; Jélvez et al. 2020).

Previous references focus on block scheduling, considering only precedence constraints to determine the obtained geometry. However, the work from Cullenbine, Wood, and Newman (2011) stands out because it incorporates some simple geometric constraints. Indeed, this article introduces an extension of the direct block scheduling that aims to have some space at the bottom, for which it adds an extra constraint that requires at least two adjacent blocks to be extracted at the bottom of the pit. The obtained model is difficult to be solved, and; therefore, the authors also proposed a rolling horizon heuristic to find good solutions.

Indeed, it is possible to use precedence constraints to convey the solutions to respect the extracting order between the pushbacks calculated using nested pits. An alternative to direct block scheduling is bench-phase scheduling, where sets of blocks composed by the intersection of phases with benches are scheduled in such a way that slope constraints are respected.

BHP Billiton software tool called Blasor implements a MIP for bench-phase scheduling through a clustering algorithm (Stone et al. 2007). The clustering is done with a two-level aggregation. Similar blocks are aggregated into bins, and these are joined into specific sets called "AGG" such that the boundaries of aggregated blocks respect the maximum slope constraints. A recent study by Letelier et al. (2020) exposes direct block scheduling and bench-phase models for which they present preprocessing, cutting plane techniques, heuristic approaches and a customized branch-and-bound that permit to obtain better bounds and feasible solutions with low relative gaps.

Because using direct block scheduling to calculate NPV can be time-consuming, in this article, we preferred to follow a simpler approach described in Bai et al. (2018), which generates a feasible block schedule following simple rules and applying a discount rate on each block. We provide the details in Section 4.1.2.

3. Integer linear programming model for the ultimate pit with minimum bottom width

As reported before, the integer linear programming model (ILP) presented here is an extension of the final pit model presented in Section 1, which aims to generate a pit geometry that complies with the minimum bottom width. Because of this, as in the case of the ultimate pit, let $\mathcal{B}$ be the set of blocks, v_b be the profit associated to block $b \in \mathcal{B}$ and $\mathcal{P}$ be the set of slope precedences.

The extension that we consider takes into account geometric properties treated in detail in next section. It also considers tonnage targets, which may be useful to control the gap problem. For this, we introduce a target W, and a tolerance parameter δ, and require that the pit's total tonnage has to be between the values $(1-\delta)W$ and $(1+\delta)W$.

For practical reasons, it is convenient to introduce a set $\mathcal{O} \subset \mathcal{B}$ on which the smoothness constraints are imposed. This set is defined as leaving out blocks that are too close to the topography, where smoothness constraints are irrelevant or may not apply.

3.1. *Modeling the shape of the pit bottom*

We consider three aspects for modelling the smoothness: minimum space around each block, space for movement of equipment, and absence of cavities. We explain these concepts that involve blocks at the same level and how they are modelled in the ILP.

3.1.1. Minimum space around each block

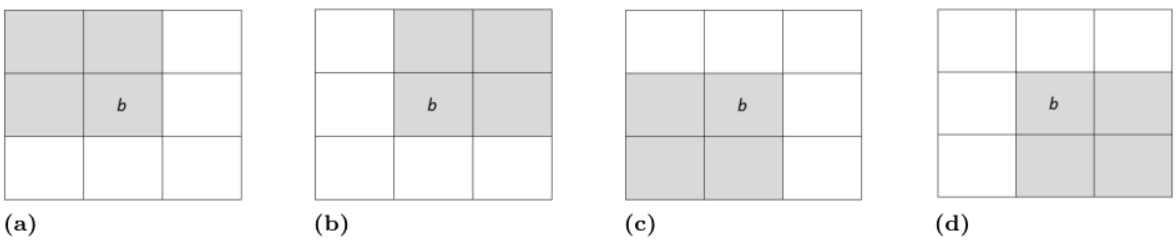


Figure 4. (a) (b) (c) (d) Plan-view of the square areas S (in grey) relative to a block b for a minimum bottom width of 2 blocks.

This property is modelled by asking that if a block is extracted, then it belongs at least to a square area of d^2 blocks that is also extracted at the same level. For this, given a block $b \in \mathcal{B}$, we denote as $\mathcal{S}^d(b)$ the set of squared areas of side d containing b. Figure 4 shows the squares area S of $\mathcal{S}^2(b)$. Notice that for blocks near the limits of the block model, some "squared area" is not complete because some hypothetic blocks fall out of the block model. Indeed, for implementation purposes, we do not consider these as elements of $\mathcal{S}^d(b)$.

3.1.2. Minimum space for movement between areas

This property is to prevent isthmuses between zones, like the one in Figure 3(b), which may prevent equipment movement between areas at the same level. To model this, we consider that if a specific *pattern* of extracted blocks appears in a bench, other blocks must be extracted. Figure 5 offers a plan-view of some examples of patterns and the blocks whose extraction they force: if the blocks in the blue pattern are extracted, then at least one red block must also be extracted.

For the patterns in Figure 5(a) that correspond to a distance $d = 3$ blocks, Figure 5(b) shows some feasible designs (blocks in dark grey) among others permitted by those patterns. We denote as $\mathcal{I}_d$ the set of patterns, $\mathcal{F}_i$ its i-th pattern (blue blocks) and $|\mathcal{F}_i|$ the size of i-th pattern. In Figure 5(a), the first three $|\mathcal{F}_i|$ patterns have a size of 18, the last pattern a size of 17.

Notice that a pattern is only a shape, i.e., it is not linked to actual blocks, and therefore it may appear several times in the pit. To anchor a given pattern to a specific set of blocks, let us define $\mathcal{F}_i(b) \subset \mathcal{B}, i \in \mathcal{I}_d$ as the blocks $b' \in \mathcal{B}$ in the i-th pattern (blue blocks) when block b is located at the South-West corner of the pattern.

As mentioned before, the extraction of blocks in $\mathcal{F}_i(b)$ forces at least one block in a second pattern (red blocks) to be extracted. The set of such blocks is denoted as $\mathcal{T}_i(b)$. It is worth noting that the constraints relative to the minimum space around each extracted red block will apply as well as the ones that fill cavities.

This new modelling is less restrictive than classic precedences to allow the mathematical model to find a precise design that eliminates the isthmuses, optimizes the overall value, and may keep some parts of the original pit's contour.

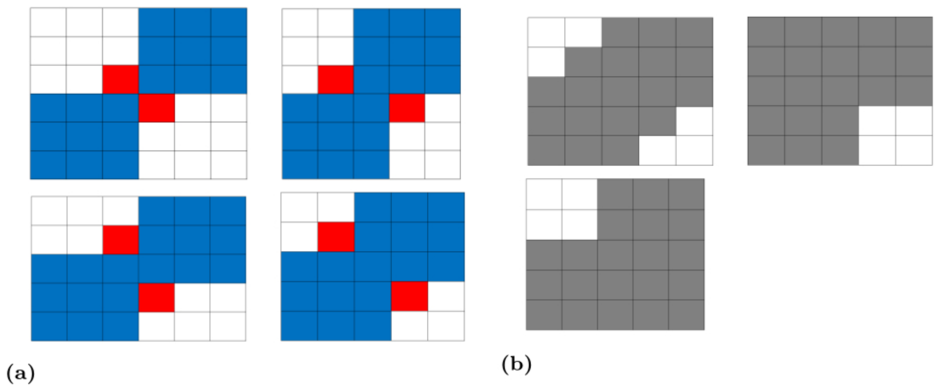


Figure 5. (a) Plan-view of patterns to ensure movement space at the pit bottoms; (b) plan-view of feasible designs

3.1.3. Cavities

The fact that each block in dark grey belongs to a square area is not sufficient to remove all the unwanted small cavities similar to those described in Bai et al. (2018). A cavity is composed of unextracted blocks in the X or Y-axis direction positioned between 2 extracted blocks (Figure 6(a) presents two cavities: one between blocks i and j and one between j and

k). To prevent this type of shape, we introduce a distance parameter d' that corresponds to the maximum size of unwanted cavities in the X or Y-axis direction. For this, let us denote as b_x, b_y, b_z the x, y and z coordinates of block b respectively, and let Δ be the size of the blocks (each side), then

$$\begin{aligned}
\mathcal{WE}_{d'} &= \left\{(b', b, b'') \in \mathcal{B}^3 \ : b'_y = b_y = b''_y, b'_z = b_z = b''_z, b''_x - b'_x \le d'\Delta, b'_x \le b_x \le b''_x\right\}, \\
\mathcal{NS}_{d'} &= \left\{(b', b, b'') \in \mathcal{B}^3 \ : b'_x = b_x = b''_x, b'_z = b_z = b''_z, b''_y - b'_y \le d'\Delta, b'_y \le b_y \le b''_y\right\}.
\end{aligned}$$

Figure 6(b) presents an example of $\mathcal{WE}_{d'}$ when $d' = 9$. Because blocks b' and b'' have the same y coordinate and at a distance at most d', then triplet (b', b, b'') is an element of $\mathcal{WE}_{d'}$.

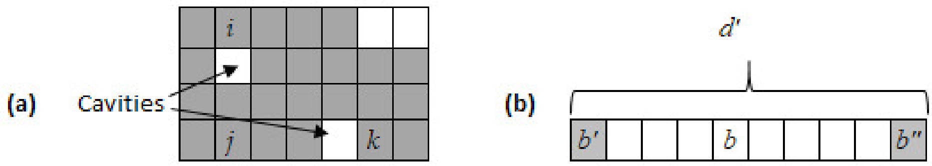


Figure 6. An example of cavities and convexity constraint. (a) Plant-view of blocks (extracted blocks are in grey) showing two cavities. (b) An element of $\mathcal{WE}_{d'}$ for $d' = 9$. Greyed blocks b', b'' represent blocks that, if extracted, force the extraction of all blocks b in between.

3.2. *Mathematical formulation*

As in the case of the ultimate pit, we consider binary variables x_b, where $x_b = 1$ if and only if block $b \in \mathcal{B}$ is extracted. We introduce the additional decision variables y^d_{bS}, which is equal to 1 if and only block b is extracted and all blocks in $S \in \mathcal{S}^d(b)$ are also extracted.

Table 1 summarizes the notation of the model for the *geometrically constrained ultimate pit problem* ($\mathcal{GCUP}(\lambda)$ for short), which is presented now.

$$\mathcal{GCUP}(\lambda) \quad \max \sum_{b \in \mathcal{B}} v_{b,\lambda}\ x_b \tag{1}$$

$$x_b \le x_{b'} \qquad \forall (b, b') \in \mathcal{P} \tag{2}$$

$$\sum_{S \in \mathcal{S}^d(b)} y^d_{bS} \ge x_b \qquad \forall b \in \mathcal{O} \tag{3}$$

$$y^d_{bS} \le x_{b'} \qquad \forall b \in \mathcal{O},\ \forall S \in \mathcal{S}^d(b), \forall b' \in S \tag{4}$$

$$x_{b'} + x_{b''} - x_b \le 1 \qquad \forall (b', b, b'') \in \mathcal{WE}_{d'} \cup \mathcal{NS}_{d'} \tag{5}$$

$$\sum_{b' \in \mathcal{F}_i(b)} x_{b'} - \sum_{b'' \in \mathcal{T}_i(b)} x_{b''} \le |\mathcal{F}_i| - 1 \qquad \forall b \in \mathcal{O}, i \in \mathcal{I}_d \tag{6}$$

$$(1 - \delta)W \le \sum_{b \in \mathcal{B}} w_b\ x_b \le (1 + \delta)W \tag{7}$$

$$x_b \in \{0, 1\} \qquad \forall b \in \mathcal{B} \tag{8}$$

$$y_b \in \{0, 1\} \qquad \forall b \in \mathcal{O} \tag{9}$$

Symbol	Description
$\mathcal{B}$	Set of blocks
$\mathcal{O}$	Set of blocks over which to apply smoothness constraints
$\mathcal{I}_d$	Set of indexes of smoothness pattern $\mathcal{F}_i$ with respect to the minimum bottom width d
$\mathcal{F}_i$,$\mathcal{T}_i$	Smoothness patterns
$\mathcal{S}^d(b)$	Set of squared areas of side d that contain block b
b, b', b''	Blocks, elements of $\mathcal{B}$
d	Minimum bottom width measured in quantity of block sides
d'	Maximum cavity size parameter measured in quantity of block sides to be filled by blocks in the X and Y-axis directions
$\mathcal{WE}_{d'}$	Set of triplets (b_1, b_2, b_3) such that the blocks are aligned in the X-axis and at the same level, distance between b_1, b_3 is d', and b_2 is between b_1 and b_3
$\mathcal{NS}_{d'}$	Set of triplets (b_1, b_2, b_3) such that the blocks are aligned in the X-axis and at the same level, distance between b_1, b_3 is d', and b_2 is between b_1 and b_3
$\mathcal{P}$	Set of slope precedence arcs
λ	Revenue factor, $\lambda = 1$ for the final pit compute
$v_{b,\lambda}$	Economic value perceived if block b in $\mathcal{B}$ is extracted
w_b	Tonnage of block b
W	Tonnage of the final pit without minimum mining width that corresponds to the revenue factor λ
δ	Tonnage tolerance

Table 1. Notation for the mathematical formulations.

The objective function (1) maximizes the overall undiscounted profit of all blocks in the designed pit. Constraints (2) is the classic slope precedence constraint of the final pit model. Constraints (3) state that all extracted blocks outside of the topography must belong at least to a $d \times d$ square of blocks. Constraints (4) state that all the blocks to complete the $d \times d$ square area S must be extracted to set the variable y^d_{bS} to the value 1. Constraints (5) state that the blocks between two extracted blocks b' and b'', that belong to an unwanted cavity in the X or Y-axis direction, have to be extracted. Constraints (6) state that if a smoothness pattern set $\mathcal{F}_i(b)$ of blocks are extracted, then at least one block of the set $\mathcal{T}_i(b)$ have to be extracted. Constraint (7) states that the overall extracted rock tonnage is inside an interval defined by a tolerance of $\delta\%$ around the rock tonnage of the final pit without minimum mining width. Constraints (8) and (9) denote the nature of the variables.

As the variables y^d_{bS} and the constraints (3)-(6) break the structure of the problem, the proposed problem cannot be solved with Lerchs and Grossmann or the pseudo-flow algorithm. We present here a preprocessing that is a heuristic approach. It consists of only considering an extended three-dimensional form of the ultimate pit in the X and Y-axis directions. This extension is done with blocks in such a way that the distance between them and blocks at the same bench in the ultimate pit is less than or equal to d.

4. Numerical experiments

The MineLink library developed at Delphos Mine Planning Laboratory at Universidad de Chile (Delphos 2019) was used to develop the code to implement this model. This library provides utilities to manage the block models, compute block values and generate precedence arcs. The code was written in Python, and it was developed to manage the data sets and implement the integer linear programming model $\mathcal{GCUP}(\lambda)$. For all computations, linear optimization of the model was solved using Gurobi 8.1 (Gurobi 2019) and the experiments run on a PC with 10 CPU Xeon E5 2660 v3 (2.60 GHz) with 128 GB of RAM under a Microsoft Windows 7 environment.

We applied the ILP to three block models. The first two models are available publicly in MineLib (Espinoza et al. 2013): KD, which is a copper mine and has 14,153 blocks of size $20m \times 20m \times 15m$, and Marvin, which is a fictitious gold and copper mine included in Whittle (Gemcom 2018) and consists of 53,271 blocks of size $30m \times 30m \times 30m$. The third model is called D3 and is a part of a copper mine that contains 52,850 blocks of size $25m \times 25m \times 15m$. We applied a minimum bottom width of three blocks for all block models, which corresponds to 60 meters for KD, 90 meters for Marvin, and 75 meters for D3 model. The maximum distance to fill cavities (parameter d') corresponded to 4 blocks in all models. The tonnage tolerance (parameter δ of the programming model) was 5% for all models. Marvin and KD block values are used from MineLib without changes. However, for D3, the economic parameters are shown in Table 2.

The final pit characteristics for the three models are given in Table 3. These characteristics are the undiscounted value in millions of US dollars, the extracted rock tonnage (Rock), the ore tonnage (Ore) in millions of tons, and the number of extracted blocks. These final pits were computed without considering a minimum mining width using an implementation of the pseudo-flow algorithm proposed by Hochbaum and Chen (2000).

Table 2. Parameters to generate the economic value of blocks for D3 model

Economic parameter	Value for D3
Metallurgical recovery Cu	0.88
Metal price Cu	2.5 \$/lb
Selling cost Cu	0.4 \$/lb
Mining cost	0.9 \$/ton
Processing cost	4.0 \$/ton

Table 3. Original final pit economic value, weight and number of blocks

Block model name	Final pit Value M\$	Rock Mton	Ore Mton	Blocks #
KD	652.2	190.6	95.8	12,154
Marvin	1,426.9	527.2	312.0	8,517
D3	7,491.3	397.4	256.5	16,726

4.1. ***Pushback selection and production scheduling***

In this section we describe how we generate the pushbacks and a production schedule to evaluate the NPV of the resulting plans.

4.1.1. Pushback generation

In order to generate pushbacks, we consider for each block model a series of values of revenue factors and select some $0 < \lambda_1 < \lambda_2 < \cdots < \lambda_L = 1$ as pushbacks. These selected values of λ are used for block valorization in the proposed model and their corresponding tonnages as targets. When generating a pushback, previous practicable ones are considered to be mined, which allows enforcing a minimum bench width at least equal to the bottom width and a minimum bottom width for each pushback. This approach also ensures that the results respect the nested property of $UP(\lambda)$.

4.1.2. NPV calculation

We used the same method as the one presented in Bai et al. (2018). It consists of defining a block sequence inside each pushback and then discount each block value depending on its sequence number in the overall scheduling. This block sequence is obtained by applying the following three precedence levels to the pushback designs: (1) the blocks in an earlier pushback precede those in later pushbacks; (2) in one pushback, the blocks on higher bench precede those on a lower bench; (3) on each bench, the blocks are extracted from west to east. The discount rate (5.6655×10^{-5}) is applied on each block. For the Marvin case, the discount of the last block of the final pit is $1/(1 + 5.6655 \times 10^{-5})^{8517} = 0.62$; this is the same rate value applied to the last block of the final pit from Bai et al. (2018) experiments.

4.2. *Numerical results*

Table 4 presents the results of the application of the proposed $\mathcal{GCUP}(1)$ linear programming model for the computation of the final pit that takes into account the minimum width and design requirements. The table presents the number of variables and constraints, the undiscounted value in million US dollars, the relative gap reported by Gurobi between the best continuous bound and the obtained integer solution, and the computation time in seconds are given for each experiment. The results are presented for two cases: the model applied to the whole block model and with the preprocessing described in Section 3, which eliminates blocks that are too far from the ultimate pit.

4.2.1. Ultimate pits and preprocessing

Table 4 shows that the preprocessing lowers the computation time from three times for KD to more than fourteen times for D3 without any loss on the objective value. As expected, the practicable final pit's undiscounted value is lower than the original one given in Table 3 (from 0.05% for Marvin to 0.31% for D3). A comparison between the extracted rock and ore tonnages show that those tonnages are lower for the practicable final pit than the original ones for KD (−0.27% and −0.36% respectively) and greater for Marvin (0, 15% and 0.11% resp.) and D3 (3.51% and 0.21% resp.). The most significant variation is observed for D3, which is consistent with the NPV results that show the same.

Table 4. Results of Branch and Bound applied on $\mathcal{GCUP}(1)$ for the practicable final pit computation

	without preprocessing					with preprocessing				
Instance	Var. #	Constr. #	Value M$	Gap %	Time [s]	Var. #	Const. #	Value M$	Gap %	Time [s]
KD	121,748	1,122,281	650.4	0.01	58	105,390	1,091,516	650.4	0.01	**19**
Marvin	499,752	4,910,844	1,426.2	0.01	406	109,010	840,100	1,426.2	0.01	**40**
D3	570,039	6,506,206	7,467.9	0.17	1,151	157,487	1,702,991	7,467.9	0.01	**78**

Figure 7 presents plan views of the bottom of the original and practicable final pit obtained by the proposed ILP with the preprocessing for each block model. Level 0 corresponds to the lowest level of each design. We observe the following for each block model,

- In the case of KD, the final pit is nearly a practicable pit; hence only a few protuberances are eliminated by the ILP at the lowest levels.
- For Marvin, four cavities in the original final pit are filled at levels 0 and one at level 1. Some blocks are added at the north of level 1 to complete a 3×3 square of blocks. At the south-east, a protuberance of 2 blocks from level 3 to level 14 in the Y-axis direction is dilated to 3 blocks in the practicable final pit to respect the minimum width requirements.
- As for D3, the final pit design is far from respecting the minimum bottom width, and therefore the outputs are very different. The practicable final pit for D3 shows larger floors at levels 2, 3, 13, and 14 that respect the minimum distance requirements and therefore is by far more operational than the original final pit design. It is worth noting that obtaining a practicable design with CAD software that optimizes the economic value for D3 is not an easy task to carry on.

As expected, the original 3D form is eroded somewhere and extended elsewhere to optimize the final pit's value respecting the design and slope precedence constraints of the proposed ILP. At some levels, the practicable contours are closed to the original ones and can be very different at other levels, as shown by the D3 case.

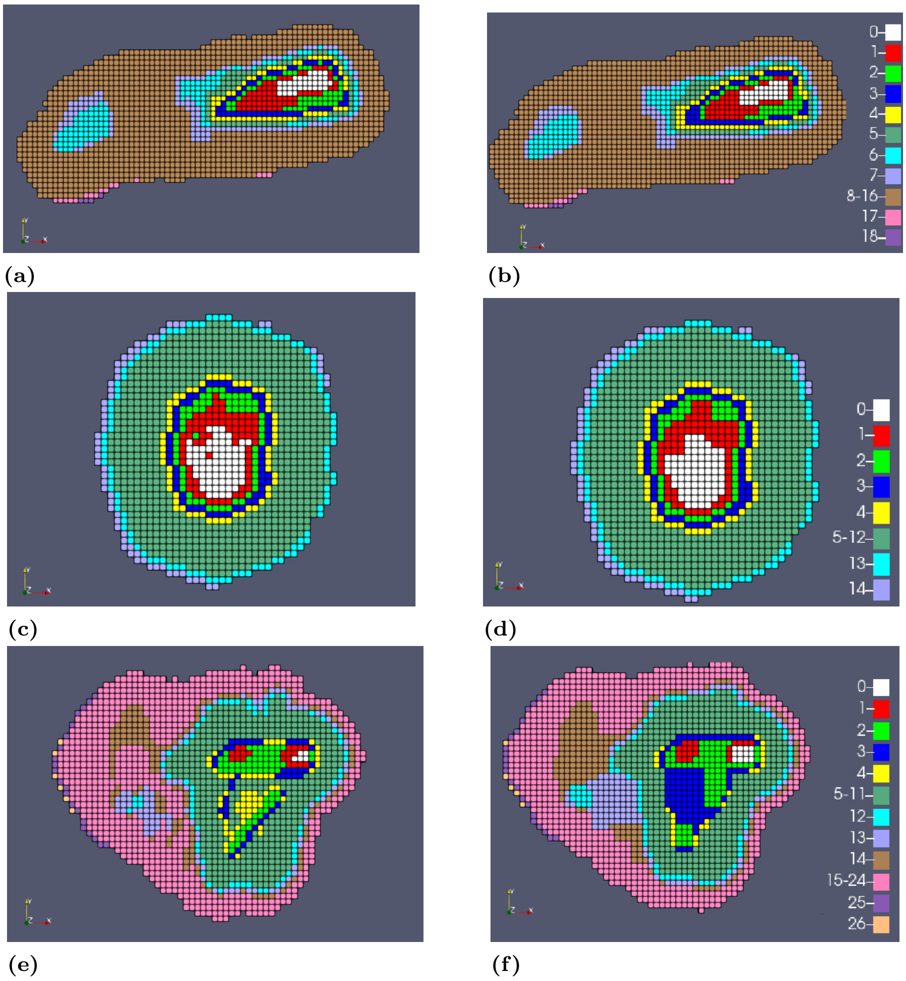


Figure 7. Bottom-view of (a) original final pit, (b) practicable final pit from KD model; (c) original final pit, (d) practicable final pit from Marvin model; (e) original final pit, (f) practicable final pit from D3 model

4.2.2. Pushback generation

Table 5 summarizes the results of the original pushbacks generated by the classical nested pits methodology using different revenue factors and those obtained by applying the proposed pushback generation methodology. For each experiment, the undiscounted value in million US dollars, the extracted rock tonnage (Rock), the extracted ore tonnage (Ore) in million of tons, and the number of extracted blocks are presented. We observe that for KD and D3, the proposed approach can achieve similar values in terms of tonnage and value, while generating better geometries for design. On the other hand, for Marvin, we see noticeable differences, and one pushback with a negative value, which is not desirable in practice. In terms of ore tonnages, for Marvin's case, the differences are important between the original and practicable pushbacks for the second (-23.84%) and the third ones (-12,06%). Therefore, it may be interesting to evaluate other revenue factors scenario to generate a smaller or different form for the first pushback that could permit other practicable forms for the second one. This approach also ensure that the results respect the nested property of $UP(\lambda)$.

Table 5. Results of original pushbacks and those obtained by the proposed pushback generation

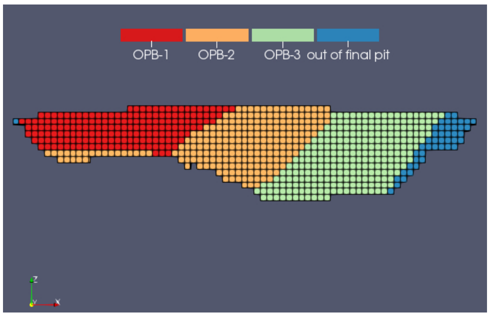
	Original pushbacks				Practicable pushbacks				
Inst. - λ	Value M$	Rock Mton	Ore Mton	Blocks #	Value M$	Rock Mton	Ore Mton	Blocks #	time [s]
KD - 0.15	388.5	68.9	53.6	4,384	387.1	72.4	55.0	4,613	404
KD - 0.28	164.5	51.7	25.9	3,396	139.8	43.5	20.8	2,767	39
KD - 1.00	99.2	70.0	16.3	4,474	123.5	74.1	19.6	4,743	28
KD Total	**652.2**	**190.6**	**95.8**	**12,154**	**650.4**	**190.0**	**95.4**	**12,123**	**476**
Marvin - 0.50	1,001.3	225.2	176.3	3,640	967.1	215.4	170.5	3,489	156
Marvin - 0.60	263.0	111.3	60.0	1,755	-90.6	105.1	9.4	2,029	15
Marvin - 0.75	131.9	107.8	46.3	1,757	203.4	142.7	68.6	2,114	6
Marvin - 1.00	30.7	82.9	29.4	1,365	346.3	64.8	63.8	899	1
Marvin Total	**1426.9**	**527.2**	**312.0**	**8,517**	**1,426.2**	**528.0**	**312.3**	**8,531**	**226**
D3 - 0.10	5,203.3	171.7	145.5	7,240	4,566.4	163.2	136.3	6,887	1,125
D3 - 0.14	1,464.4	88.2	57.4	3,710	953.8	84.6	40.9	3,556	30
D3 - 1.00	823.5	137.4	53.7	5,776	1,947.7	163.5	79.8	6,869	3
D3 Total	**7,491.3**	**397.4**	**256.5**	**16,726**	**7,467.9**	**411.3**	**257.1**	**17,312**	**1,158**

Figure 8 shows a section view of the pushbacks obtained with the proposed model and the standard nested pit approach, using different colours to represent the shape differences for consecutive values of the revenue factor λ. We observe the following for each block model,

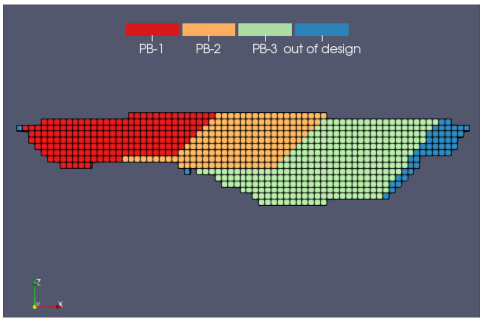
- In the case of KD, the original pushbacks are nearly operational ones and grow from West to East. But for example, in the South-West of the mine, there are parts of the second original pushback that do not respect the bottom width of three sides of block and do not respect the continuity property of the design. It can also be noted in Figure 8(b) that the second pushback respects all the design requirements.
- For Marvin, in Figure 8(c) all the original pushbacks but the first one do not respect the minimum bench width. In a practicable design that keeps the form of the first pushback, the other ones were obliged to grow from top to bottom to respect the design requirements, the slope precedences, and the extraction tonnages constraints. This explains the negative undiscounted value of the second pushback.
- Finally, in the case of D3, Figure 8(e) shows that all the original pushbacks do not respect the minimum bottom width of three sides of block. However, this issue is solved by the practicable design generated by our model (Figure 8(f)).

Figure 9 shows the NPV for the original nested pits and for the practicable pushbacks for each study case. We observe that for KD, the proposed approach achieves similar values in terms of NPV. For Marvin, the second pushback has a negative value, which is not desirable.

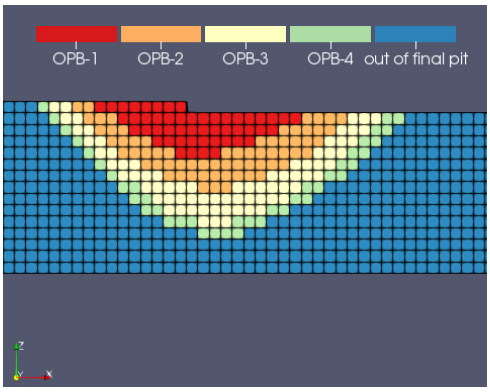
Finally, Table 6 summarizes the results presented in Table 5 and Figure 9 for each overall pit showing the variation on the undiscounted value, rock tonnage and NPV for each case. We can observe that the tonnage variation is under 5% around the original rock tonnage.



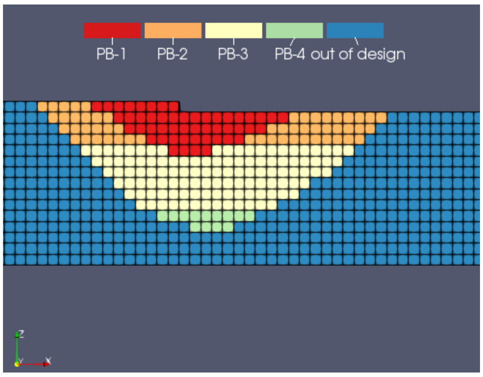
(a)



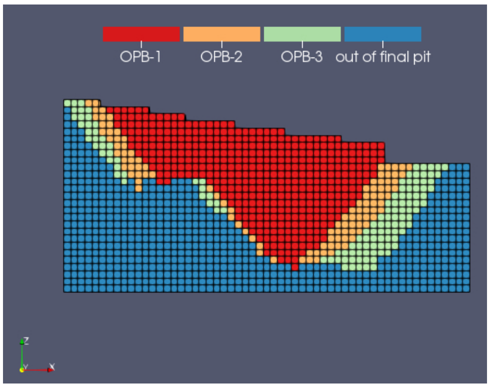
(b)



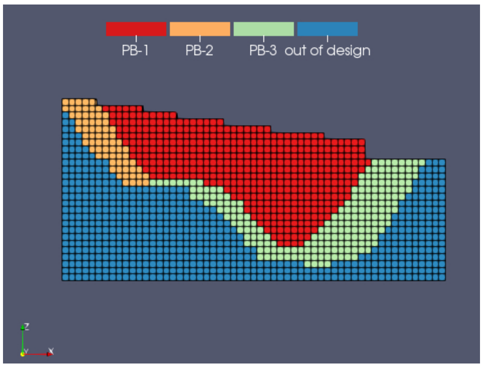
(c)



(d)

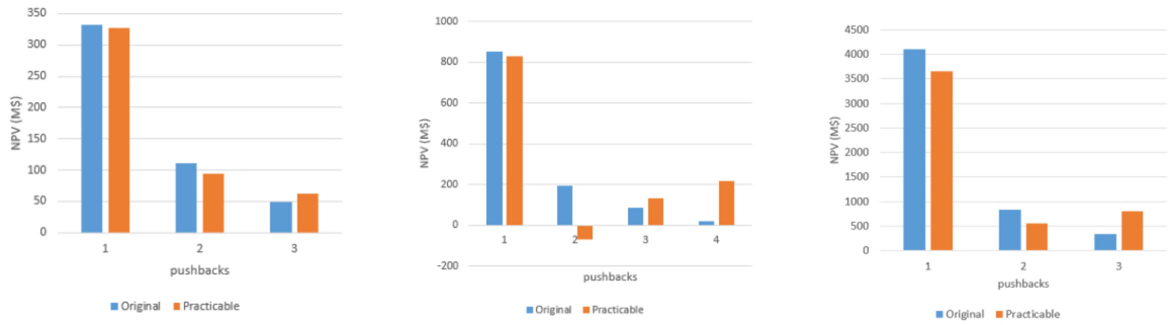


(e)



(f)

Figure 8. Section-view of (a) original pushbacks, (b) practicable pushbacks from KD model; (c) original pushbacks, (d) practicable pushbacks from Marvin model; (e) original pushbacks, (f) practicable pushbacks from D3 model. *OPB-i* means original pubshback number *i*. *PB-i* means practicable pubshback number *i*



(a) (b) (c)

Figure 9. Net contribution to NPV for each pushback in the original and practicable pushbacks. (a) KD model, (b) Marvin model, (c) D3 model

The variations between the undiscounted values are very small (from 0.05% to 0.31%). As expected, the net present values for practicable designs are lower than for the original pits (-1.54% to -5.34%). However, the differences are small. These results are compatible with the decrease of 0.5% to 7.3% in the NPV reported by Bai et al. (2018) in their experiments.

Table 6. Results of original pits and those obtained by the proposed pushback generation

	Original pit			Practicable pit			Variation		
Instance	Value M$	Rock Mton	NPV M$	Value M$	Rock Mton	NPV M$	Value %	Rock %	NPV %
KD	652.2	190.6	492.4	650.4	190.0	484.8	-0.28	-0.31	-1.54
Marvin	1426.9	527.2	1,156.5	1,426.2	528.0	1,116.0	-0.05	0.15	-3.51
D3	7,491.3	397.4	5,284.2	7,467.9	411.3	5,002.1	-0.31	3.50	-5.34

5. Conclusions

An integer linear programming model has been presented to generate ultimate pits considering minimum width at the mine's bottom. Large mining equipment need this minimum width to move and operate. A smoothness property was defined to ensure that there is enough space to move between different pit bottom zones, which is an improvement over previous attempts to model this problem.

The proposed programming model was applied to three block models (KD, Marvin, and D3), showing that the approach results are consistent and useful for applications. Of particular relevance is the case D3, where differences between the original and the practicable final pits are significant and therefore they show the interest of applying optimization to maximize the undiscounted value of a practicable bottom. This allows obtaining a range of 0.05% to 0.31% for the expected decrease in the original final pits' undiscounted value. It is worth noting that a proposed preprocessing lowers the computation time fourteen times for D3 case.

The second application of this programming model was presented to generate sequential practicable pushbacks in minutes via a modified Lerchs and Grossman (1965) methodology. We obtained a decrease of 1.54% to 5.34% in the NPV between the design with the original pushbacks and the practicable one. The parametrization process to determine the revenue factors used as inputs in this application could be done by computing a set of nested pits without considering a minimum mining width and then solving the mixed-integer linear programming model proposed by Jélvez, Morales, and Askari-Nasab (2020) for this set. Their model helps to select inside a defined set, the nested pits that balance the tonnage differences between a fixed number of successive pushbacks. This could permit to obtain revenue factors and tonnage targets of interest.

Some potential future works may consider using a spatial aggregation/disaggregation heuristic approach of the type presented in Jelvez et al. (2016) to lower the computation times of the presented ILP. A programming model able to generate production plans that consider the minimum width studied in this article is also a desirable tool.

Disclosure statement

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this article.

Funding

This work was partially funded by the CONICYT/PIA Project AFB180004.

References

Asad, Mohammad Waqar Ali, Roussos Dimitrakopoulos, and Jeroen van Eldert. 2014. "Stochastic production phase design for an open pit mining complex with multiple processing streams." *Engineering Optimization* 46 (8): 1139–1152. https://doi.org/10.1080/0305215X.2013.819094.

Asad, M.W.A, and E. Topal. 2011. "Net present value maximization model for optimum cut-off grade policy of open pit mining operations." *Journal of the Southern African Institute of Mining and Metallurgy* 111: 741 – 750.

Bai, Xiaoyu, Denis Marcotte, Michel Gamache, D. Gregory, and A. Lapworth. 2018. "Automatic generation of feasible mining pushbacks for open pit strategic planning." *The Journal of The Southern African Institute of Mining and Metallurgy* 118 (5): 515–530.

Consuegra, F. R. A., and R. Dimitrakopoulos. 2010. "Algorithmic approach to pushback design based on stochastic programming: method, application and comparisons." *Mining Technology* 119 (2): 88–101.

Cullenbine, C., R.K. Wood, and A. Newman. 2011. "A sliding time window heuristic for open pit mine block sequencing." *Optimization Letters* 5 (3): 365–377.

Dagdelen, K. 2001. "Open pit optimization-strategies for improving economics of mining projects through mine planning." In *17th International Mining Congress and Exhibition of Turkey*, 117–121.

Delphos. 2019. "MineLink M.P.L." http://delphos.dmi.uchile.cl. Accessed 20 December 2019.

Espinoza, D., M. Goycoolea, E. Moreno, and A. Newman. 2013. "MineLib: a library of open pit mining problems." *Annals of Operations Research* 206 (1): 93–114.

Gemcom. 2018. "Gemcom Whittle™ Strategic Mine Planning software." http://www.gemcomsoftware.com/products/whittle. Accessed 18 July 2018.

Goodfellow, R., and R. Dimitrakopoulos. 2013. "Algorithmic integration of geological uncertainty in pushback designs for complex multiprocess open pit mines." *Mining Technology* 122 (2): 67–77.

Gurobi. 2019. "Gurobi™ Gurobi Optimization." http://www.gurobi.com. Accessed 20 December 2019.

Hochbaum, D. S. 2008. "The Pseudoflow Algorithm: A New Algorithm for the Maximum-Flow Problem." *Operations Research* 56: 992–1009.

Hochbaum, D. S., and A. Chen. 2000. "Performance Analysis and Best Implementations of Old and New Algorithms for the Open-Pit Mining Problem." *Operations Research* 48: 894–914.

Hochbaum, D. S., and J. B. Orlin. 2012. "Simplifications and speedups of the pseudoflow algorithm." *Networks* 61: 40–57.

Jelvez, E., N Morales, P. Nancel-Penard, J. Peypouquet, and P. Reyes. 2016. "Aggregation heuristic for the open-pit block scheduling problem." *European Journal Of Operational Research* 249 (3): 1169–1177.

Jélvez, Enrique, Nelson Morales, and Hooman Askari-Nasab. 2020. "A new model for automated pushback selection." *Computers and Operations Research* 115: 104456. http://www.sciencedirect.com/science/article/pii/S0305054818301060.

Jélvez, Enrique, Nelson Morales, Pierre Nancel-Penard, and Fabien Cornillier. 2020. "A new hybrid heuristic algorithm for the Precedence Constrained Production Scheduling Problem: A mining application." *Omega* 94: 102046. http://www.sciencedirect.com/science/article/pii/S0305048318301750.

Johnson, T.B. 1968. "Optimum open-pit mine production scheduling." PhD diss., Operations Research Department, University of California, Berkeley.

Johnson, T.B. 1969. *A Decade Of Digital Computing In The Mineral Industry*, Chap. Optimum open-pit mine production scheduling, 539–562.

Juarez, Guillermo, Ricardo Dodds, Adriana Echeverría, Javier Ibanez Guzman, Matías Recabarren, Jacek RpDda, and E Vila-Echague. 2014. "Open Pit Strategic Mine Planning with Automatic Phase Generation." In *Proc. of the Orebody Modelling and Strategic Mine Planning Symposium 2014*, 147–154. Australian Institute of Mining and Metallurgy, Melbourne.

Komatsu. 2018. "P&H 4100XPC Electrical shovel general specification." https://mining.komatsu/docs/default-source/product-documents/surface/electric-rope-shovels/4100xpc-specification-sheet.pdf. Accessed: 2020-01-8.

Lamghari, Amina, Roussos Dimitrakopoulos, and Jacques A. Ferland. 2014. "A hybrid method based on linear programming and variable neighborhood descent for scheduling production in open-pit mines." *Journal of Global Optimization* 63 (3): 555–582.

Lerchs, H., and H.C. Grossman. 1965. "Optimal design of open-pit mines." *Transactions C.I.M.* 58: 47–54.

Letelier, Orlando, Daniel Espinoza, Marcos Goycoolea, Eduardo Moreno, and Gonzalo Muñoz. 2020. "Production Scheduling for Strategic Open Pit Mine Planning: A Mixed-Integer Programming Approach." *Operations Research* 68.

Liu, S.Q., and E. Kozan. 2016. "New graph-based algorithm to efficiently solve large scale open pit mining optimization problems." *Expert Systems With Applications* 43: 59–65.

Meagher, C., R. Dimitrakopoulos, and D. Avis. 2014. "Optimized open pit mine design, pushbacks and the gap problem—a review." *Journal of Mining Science* 50 (3): 508–526. https://doi.org/10.1134/S1062739114030132.

Meagher, C., R. Dimitrakopoulos, and V. Vidal. 2014. "A new approach to constrained open pit pushback design using dynamic cut-off grades." *Journal of Mining Science* 50 (4): 733–744.

Nancel-Penard, Pierre, Andrés Parra, Nelson Morales, Cristopher Díaz, and Eleonora Widzyk-Capehart. 2019. "Value-Optimal Design of Ramps in Open Pit Mining." *Archives of Mining Sciences* vol. 64 (No 2): 399–413.

Parra, Andrés, Nelson Morales, Javier Vallejos, and Phu Minh Vuong Nguyen. 2017. "Open pit mine planning considering geomechanical fundamentals." *International Journal of Mining, Reclamation and Environment* 1–18.

Picard, J.C. 1976. "Maximal Closure of a Graph and Applications to Combinatorial Problems." *Management Science* 22: 1268–1272.

Ramazan, S., and K. Dagdelen. 1998. "A new push back design algorithm in open pit mining." In *Proceedings of 17th MPES conference, Calgary, Canada*, 119–124.

Samavati, Mehran, Daryl Essam, Micah Nehring, and Ruhul Sarker. 2017. "A local branching heuristic for the open pit mine production scheduling problem." *European Journal of Operational Research*

257 (1): 261–271.

Stone, P., G. Froyland, M. Menabde, B. Law, R. Pasyar, and P.H.L. Monkhouse. 2007. “Blasor-blended iron ore mine planning optimisation at Yandi, Western Australia.” In *Proceedings of the International Symposium on Orebody Modelling and Strategic Mine Planning, Australasian Institute of Mining and Metallurgy, Perth*, 133–136.

Tabesh, Mohammad, Clemens Mieth, and Hooman Askari Nasab. 2014. “A multi-step approach to long-term open-pit production planning.” *International Journal of Mining and Mineral Engineering* 5: 273–298.