Les Cahiers du GERAD ISSN: 0711–2440

Updating short-term material flow optimization in a mining complex with new information

C. Paduraru,
R. Dimitrakopoulos

G–2016–101

November 2016

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

Updating short-term material flow optimization in a mining complex with new information

Cosmin Paduraru [a]

Roussos Dimitrakopoulos [a,b]

[a] *COSMO – Stochastic Mine Planning Laboratory, Department of Mining and Materials Engineering, McGill University, Montréal (Québec) Canada, H3A 0E8*

[b] *GERAD Montréal (Québec), Canada, H3T 2A7*

cosmin.paduraru@mail.mcgill.ca
roussos.dimitrakopoulos@mcgill.ca

November 2016

Les Cahiers du GERAD

G–2016–101

Abstract: With the advent of inexpensive sensors and digital storage, increasing amounts of data about a mining complex can be collected. This can include camera imaging, mill sensors, blasthole analysis, GPS devices etc. This paper shows how the reduction in uncertainty resulting from this data can be incorporated into stochastic short-term decision-making. This is done through the use of adaptive decision-making policies, which encode recipes for responding to new information as it comes along. Focusing on short-term planning, the paper describes how adaptive policies for allocating the extracted material can be computed in conjunction with optimizing the schedule. The resulting plan can be applied across different short-term time scales, marking an important step towards simultaneously optimizing different time scales. An implementation of the proposed method for a copper-gold deposit shows that it can improve over simple heuristic approaches.

1 Introduction

An increasing array of devices for measuring material flow in a mining complex has become available in recent years. Near-infrared reflectance spectroscopy placed on conveyor belts have been shown to provide accurate measurements of mineral concentration and other properties (Goetz et al., 2009). Camera images can also be used for inferring various mineral properties using machine learning methods (Horrocks et al., 2015; Chatterjee, 2013). Fleet location data is increasingly collected thanks to the widespread use of GPS devices.

All of this data, together with more traditional information such as blasthole and in-fill drilling analysis, can greatly reduce the uncertainty regarding the properties of the extracted and processed material. In practice, decisions made by mine operators do adapt to this uncertainty reduction - for instance, the mill will not be fed material that is revealed to be unsuitable, even if the long-term plan might require it. However, mine planning methods are not currently equipped for including this type of adaptation into their models.

This paper proposes a mechanism for optimizing material destination policies that respond to progressively-revealed information about the extracted material, and shows how to integrate the resulting adaptive policies with short-term production planning. The optimization step can increase value by determining the best way to adapt to new information; the integration with scheduling can lead to a more realistic assessment of mineral value, and therefore to a better schedule.

The key concept for computing the adaptive policies described herein is that of *state*, which is a numerical summary of the system's properties. For destination policies, this can include information about the block being allocated, as well as information about the status of the different processing destinations. An adaptive policy can then be defined as a mechanism that outputs a decision (in our case, a destination decision) for every state that the system (in our case, the mining complex) might be in.

Since this work deals with responding to uncertainty reduction, it is naturally set within a stochastic mine planning context. There has been significant progress in recent years in the optimization of mining complexes under uncertainty (Hoerger et al., 1999; Whittle, 2007, 2010; Stone et al., 2007; Dimitrakopoulos, 2011; Peevers and Whittle, 2013; Goodfellow, 2014; Montiel, 2014). However, few works study how the mining complex should adapt to new information and to the resulting decrease in uncertainty, and most of the ones that do only consider long-term (strategic) planning. For instance, Dimitrakopoulos and Sabour (2007) and Del Castillo and Dimitrakopoulos (2014) consider a real option valuation framework for analyzing the impact of using stochastic programming (MSP) in order to plan for how a long-term schedule should be updated as new information is obtained. The same methodology is used by Armstrong et al. (2012) in order to develop contingency plans for incidents that lead to production interruptions, and by Pimentel et al. (2013) for examining the option of adding new supply stations and ports to a regional supply chain. Boland et al. (2008) use multi-stage stochastic programming for designing a production schedule that provides contingencies for different future realizations of uncertainty. Paduraru and Dimitrakopoulos (2014) use an approximate dynamic programming approach for computing adaptive destination policies in a long-term planning context.

In contrast to the works above, the work presented in this paper focuses on short-term time scales. The adaptive destination policies introduced herein respond to new information obtained at a high temporal resolution (roughly corresponding to the time it takes to extract a block of material). This ensures consistency between different time scales, allowing for mutually consistent plans for different short-term time scales (e.g. months, weeks, days) to be produced by a simple process of temporal aggregation.

The closest existing work is perhaps that of Benndorf et al. (2014), who propose a general framework for integrated updates of the model and the plan based on new information. This includes a proposal to perform simulation-based optimisation for updating the short-term production plan in response to the new information. However, this has not yet been implemented, their case study focusing solely on the model update component.

This paper presents a complete algorithm for computing adaptive state-dependent policies that update short-term decisions given new information. The algorithm is implemented for computing state-dependent destination policies for a copper-gold case study, with the results illustrating the benefits of using the optimized policies.

2 Computing adaptive short-term policies

This section shows how to compute state-dependent short-term policies that encode how the mining complex should adapt to new information. Throughout the section, destination policies will be used as an illustrative example (a destination policy is a mechanism for deciding which destination each block will be sent to). However, many of the concepts are general and can be applied for computing other types of policies as well.

2.1 Short-term destination policies

As mentioned in the introduction, most previous work on adapting to new information in a mining complex has focused on long-term planning aspects. While undoubtedly important, the long-term view can nevertheless miss important details that have a significant impact on the final value obtained.

As an example, consider the problem of computing destination policies for a mining complex composed of a mine and two processing destinations, a mill and a heap leach. Assume that a long-term production schedule is available, so that it is known what blocks will be extracted in any given year. A destination policy for this example must decide whether to send each ore block to the mill or to the heap leach.

Figure 1 illustrates a typical way of evaluating a destination policy within the long-term view. For each year, there are two main steps to perform: compute the destinations for each block scheduled during that year, and compute the cumulative effect of sending the blocks to their respective destinations. Reducing the evaluation to these two steps simplifies the problem, but it may miss details that are ultimately important. For instance, it fails to identify situations where there is enough suitable material for meeting mill demand within the whole year, but due to physical constraints most of that material can only be extracted in the second part of the year. This can lead to the mill not operating at capacity during the first part of the year, and to subsequent loss of production that is not accounted for in the long-term view. This issue could be addressed by modelling a feed pile for the mill (from which material processed by the mill at a constant rate), and keeping track of the amount of ore in the feed pile as time progresses during the year, rather than only looking at total capacity for the year. A similar issue may arise when dealing with non-linear recovery curves. The amount of recovered metal computed at the end of the year under the long-term view will be based on the average grade for that year, which may not be equal to the total recovered amount throughout the year.

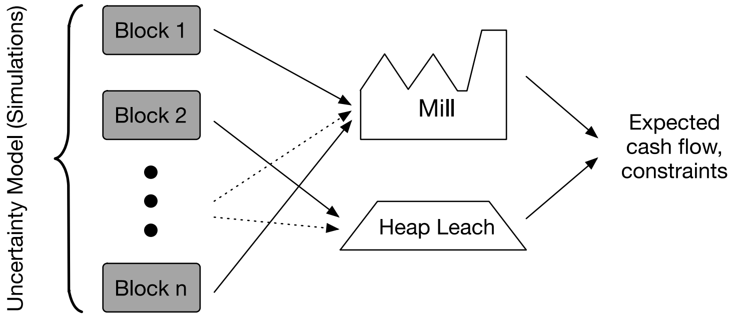
Uncertainty Model (Simulations)
Block 1
Block 2
Block n
Mill
Heap Leach
Expected cash flow, constraints

Figure 1: The typical way to evaluate destination decisions only considers what happens at the end of the year, not what happens throughout the year.

The short-term view is also better suited for answering the question of how to respond to new information obtained during the operation of the mining complex. New information is obtained at different time scales, but most of these are shorter than the yearly time scale used by long-term planning. Therefore, this information should be combined with the type of detailed short-term modelling discussed in the previous paragraph. This is illustrated by Figure 2, which shows how the available information changes after each block allocation. This new information can be used by a destination policy that, rather than only selecting destinations based on the initial uncertainty, updates its decisions as more information is obtained. The decisions could be made based on data collected about the block itself (e.g. material type), as well as information about the status of the different destinations at the point in time when the block is allocated. Notice how more detailed short-term modelling (such as using the feed pile rather than the mill as a destination) goes hand in hand with incorporating new information.

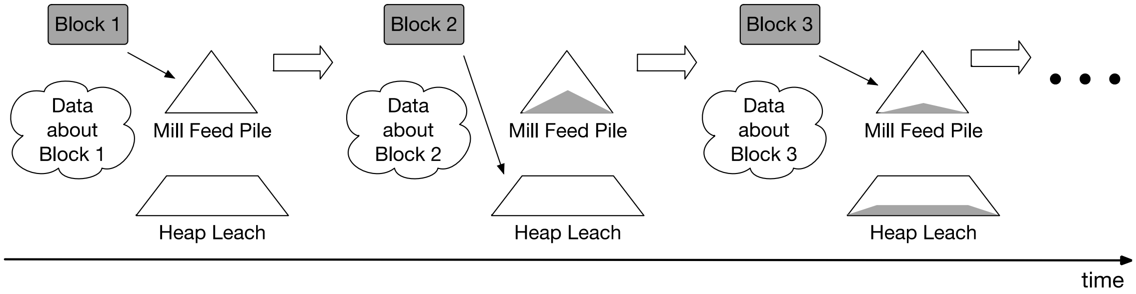

Figure 2: A short-term view of information flow for the block destination problem. At the time of each allocation, the destination policy can use information about the block that is being allocated, as well as about the material at the two destinations.

2.2 State-dependent short-term policies

This section introduces *state-dependent policies* that can explicitly encode how to respond to new information. State-dependent policies take their name from the concept of system state. The state of a system at some point in time can be informally described as a numerical representation containing all the necessary information about that system that can be extracted from data obtained up to that point in time. For a more in-depth and theoretical discussion of the concept of state, see Chapter 5 of Powell (2007).

Once again, the destination policy example from the previous section will be used for illustration and for introducing the concepts. In order to mathematically formalize destination policies, this paper assumes that extracted material can be described using a set of additive properties, similarly to Goodfellow (2014). Examples of such properties are total tonnage and metal tonnage; non-linear properties such as metal grade can be computed based on the additive properties if needed.

The details for the state-based destination policy example are as follows. Each time point *t* corresponds to sending a new block to its destination, and there are a total of *T* blocks in the period of interest. Given an order in which the blocks are sent for processing, they can be labelled $block_1, \ldots block_t, \ldots block_T$ corresponding to this order. The data obtained up until *t* can be used to compute the material properties of $block_1, \ldots block_t$ (note that uncertainty still remains regarding the properties of the blocks that need to be allocated from *t* onwards). The question of how the properties for the first *t* blocks are computed based on particular types of data is beyond the scope of this paper and will be left for future work.

The state at time *t* (denoted by s_t) needs to summarize the information available up to time *t* in a way that is suitable for decision making. The properties of $block_t$ (total tonnage $TT(block_t)$ and metal tonnage $MT(block_t)$ for this example) are certainly important for the allocation decision at time *t*, so they will be included in s_t. The properties of blocks $block_1, \ldots block_{t-1}$ are important insofar as they affect the status of the processing destinations. For the mill, the tonnage of the material in the feed pile at time *t* (denoted $TT(FP_t)$) is important because the mill has to stop processing if the feed pile is empty, and also in order to make sure that sending $block_t$ to the mill does not result in exceeding feed pile capacity. The heap leach tonnage (denoted $T(HL_t)$) is also important in order to make sure capacity is not exceeded. In addition, the metal tonnage at both destinations is important for computing the head grade and therefore the recovery at time *t*. Therefore, the state at time *t* is the vector

$$s_t = [TT(block_t), MT(block_t), TT(FP_t), MT(FP_t), TT(HL_t), MT(HL_t)].$$

Note that the total tonnage and the metal tonnage for both destinations can be computed based on the total tonnage and metal tonnage of the blocks allocated before time *t*.

A state-dependent policy is any mechanism for making decisions based on the state. State-based policies are typically represented as functions from the state space to the decision space. For the example used here, the decision space is composed of two possible decisions for each block: sending it to the mill (d_M) or sending it to the heap leach (d_{HL}). Therefore, a state-dependent destination policy for this example is a function dp such that, for any state s_t, $dp(s_t) \in \{d_M, d_{HL}\}$. More generally, if there are D destinations denoted $d_1, \dots d_D$, the destination policy would have to be such that $dp(s_t) \in \{d_1, \dots d_D\}$ for any state s_t.

2.3 Optimizing state-dependent policies

The first step to optimizing a state-dependent policy requires expressing the objective function and constraints in terms of that policy. For traditional optimization problems each potential solution is a finite-dimensional decision vector, and the objective function can be written as a mathematical function of the variables in the decision vector. For state-dependent policies, the decision vector may be infinitely dimensional if the number of states is not finite. The reason for this is that a different decision must be specified for each potential state value. This problem can be addressed by ensuring that the policy (rather than the decision vector) can be expressed using a finite number of parameters.

Let us use the two-destination example once again for illustration. A very simple type of destination policy is a cut-off grade policy. On the two-destination example, a cut-off grade policy for deciding whether an ore block should be processed by the mill or the heap leach can be written as

$$dp_g(s_t) = \begin{cases} d_M, if \ \dfrac{MT(block_t)}{TT(block_t)} \geq g \\ d_{HL}, if \ \dfrac{MT(block_t)}{TT(block_t)} < g \end{cases}$$

where g is a parameter denoting the cut-off grade. Assume that the mill and the heap leach have constant recovery factors of r_M and r_{HL}, respectively, that the processing costs per tonne are $c_M$and c_{HL} for the two destinations, that the price per tonne of concentrate is p, and that the objective is to maximize revenue minus processing cost. In addition, in order to represent uncertainty, denote each scenario (where each scenario corresponds to one realization of the uncertainty model) by sc, the number of scenarios by $|SC|$ and the total tonnage and metal tonnage for $block_t$ under scenario sc by $TT(block_t, sc)$ and $MT(block_t, sc)$ respectively. Then the optimization problem amounts to maximizing over g the expected sum of mill profit (first term) and heap leach profit (second term):

$$\frac{1}{|SC|}\sum_{sc=1}^{|SC|}\left[\sum_t \mathbf{1}_{\frac{MT(block_t,sc)}{TT(block_t,sc)}\geq g}(block_t)\big(r_M * MT(block_t, sc) * p - c_M * TT(block_t, sc)\big)\right.$$
$$\left. + \sum_t \mathbf{1}_{\frac{MT(block_t,sc)}{TT(block_t,sc)}< g}(block_t)\big(r_{HL} * MT(block_t, sc) * p - c_{HL} * TT(block_t, sc)\big)\right]$$

where for any variable x and predicate A_x the symbol $\mathbf{1}$ denotes the indicator function

$$\mathbf{1}_{A_x}(x) = \begin{cases} 1, & if \ A_x is \ TRUE \\ 0, & if \ A_x is \ FALSE \end{cases} .$$

Note that there is no discounting in the objective function above because the problem considered is short-term. Any constraints could also be written as a function of the policy, similarly to the objective function.

The cut-off grade policy used for this example is extremely simple, and ignores many components of the state vector that may have an impact on the final value. Nevertheless, it illustrates the point about representing a finite set of policy parameters rather than a decision vector: instead of explicitly representing a decision for each possible state (or block grade), there is a single parameter to optimize over (the cut-off grade g). However, this approach also results in additional issues that need to be considered. The first one is finding a suitable parametric representation for the policy, ideally such that a near-optimal policy can be found within the parameter space used. An example of a more complex state-dependent destination policy will be used for the case study in Section 3. The second issue is that, as the space of parameterized policies become more complex, the optimization problem becomes more difficult as it has to deal with a larger number of variables and complex, highly non-linear terms.

Despite these difficulties, or perhaps because of them, a large number of methods have been proposed for optimizing state-dependent policies under uncertainty. A good place to start understanding existing approaches is the literature review / tutorial titled “Clearing the Jungle of Stochastic Optimization” (Powell, 2014). In it, the author clarifies the terminology used across different disciplines that study these types of problems, identifies four main types of state-dependent policies, and reviews existing approaches for representing and optimizing these policies.

Due to the wealth and breadth of existing approaches, this paper will not attempt to provide a review of existing methods. They all amount to some sort of search in the space of policy parameters, which can be performed through a variety of methods ranging from many variations of gradient-based optimization to heuristic search and approximate dynamic programming. Instead, the current paper focuses on a conceptual understanding of state-dependent policies and their potential applications to the optimization of mining complexes, and uses a combination of grid search and the simplex method (Nelder and Mead, 1964) for optimizing policy parameters.

Destination policies are only one of several interacting components of a mining complex. Figure 3 contains a conceptual diagram describing the interaction of production scheduling and state-dependent destination policies. It can be seen from the diagram that both overall system behaviour (i.e. what material gets processed and where) as well as the value of the objective function (which is a function of the revenue and the costs) depend on both the scheduler and the destination policy. Therefore, the optimization of state-dependent policies should ideally be integrated with production scheduling.

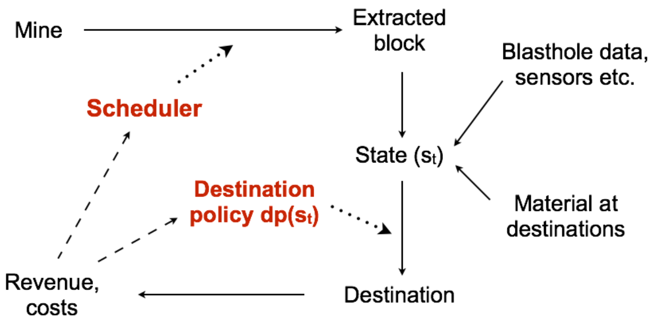

Figure 3: Integrating destination policies and production scheduling. The solid lines represent material and information flow, the dashed lines represent information that informs the optimizer, while the dotted lines represent decision-making.

Similar considerations arise for the remaining components of the mining complex (processing options, transportation, further material flows etc.). As discussed in the Introduction, this type of global optimization has been the focus of an increasing body of recent work, although it is typically not addressed from an adaptive perspective. This paper proposes the simple iterative approach outlined below in order to integrate production scheduling and adaptive destination policies:

1. Generate n random feasible schedules $sched_1, \dots sched_n$
2. Find the destination policy dp^* that maximizes over dp

$$\frac{1}{n}\sum_{i=1}^{n}\frac{1}{|SC|}\sum_{sc=1}^{|SC|} Obj(dp, sched_i, sc) \tag{1}$$

   where $Obj(dp, sched_i, sc)$ is the value of the objective function (cash flows, NPV etc.) for destination policy dp, schedule $sched_i$, and scenario sc
3. Find a schedule $sched^*$ that maximizes over $sched$

$$\frac{1}{|SC|}\sum_{sc=1}^{|SC|} Obj(dp^*, sched, sc) \quad (2)$$

4. Generate n random perturbations $sched_1, \dots sched_n$ of $sched^*$ and go to Step 2
5. Stop according to an appropriate criterion (e.g. small changes in objective function or maximum number of iterations) and output $sched^*$ and dp^*

Generating randomized perturbations of $sched^*$ in Step 4 aims to ensure that the final destination policy dp^* is robust with small changes to the schedule. This is important for situations where the schedule cannot be implemented exactly as planned due to unforeseen circumstances and practical constraints.

3 Case study

This section presents an implementation of short-term planning using state-dependent policies for a copper-gold mine. The case study is based on the one used by Goodfellow and Dimitrakopoulos (2013), but it is modified in order to include more details relevant to short-term planning.

The mine complex under consideration consists of a single mine from which the extracted blocks can be sent to one of six initial destinations. The material extracted from the mine is comprised of three main groups (sulphides, transition and oxides), each of which is divided into two sub-groups, as illustrated o the left side of Figure 4. There are six destinations to which the material can be sent: a sulphide mill, a sulphide heap leach, a sulphide dump leach, a transition heap leach, an oxide heap leach and an oxide waste dump. The sulphide mill produces both copper and gold, and the leaches produce either copper or gold, as shown in Figure 4. The destinations are selective about the types of material they accept, as illustrated in Figure 4.

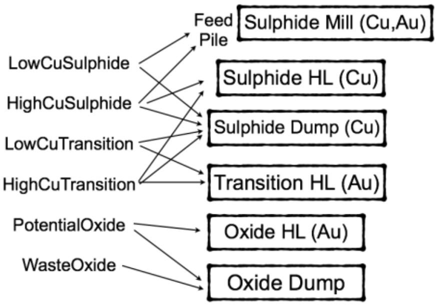

Figure 4: Graphical representation of the various material types and destinations. Each destination only accepts material types corresponding to the arrows pointing to it.

The sulphide mill is modelled to include a feed pile with a capacity of 500,000 tonnes. There is a ramp-up period of one month, during which no material is processed at the mill and any material sent to the mill accumulates on the feed pile. After the ramp-up period, the mill processes material from the feed pile at a fixed rate, unless there is no material left on the feed pile in which case it is stopped and restarted once material is deposited on the feed pile again. In order to provide a high level of detail, this process is simulated at a high temporal resolution. Thus, each time increment corresponds to the time between the extraction of two consecutive blocks. The amount that the mill processes at each of these time increments (assuming there is sufficient material on the feed pile) is equal to 2,066 tonnes. This results in a yearly mill processing capacity of 3 million tonnes. For the heap leaches (including the sulphide dump, which acts as a heap leach), material is piled on until the leach contains 1,000,000 tonnes of material, at which point leaching occurs.

The amount of concentrate produced by the different destinations is computed according to the grade-recovery curves shown in Figure 5. For the mill, the recovery at each point in time is computed based on the average grade in the feed pile at that point. The recovery for the heap leaches is computed based on the average grade of the material that has been piled on before leaching occurs. The processing costs (relative to a “base cost”) and metal prices used can be found in Table 1.

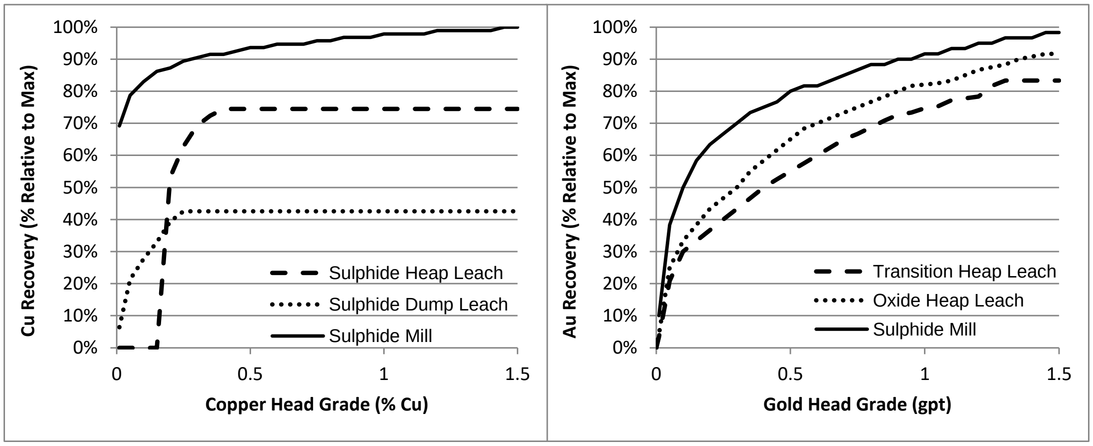

Figure 5: Grade-recovery curves for the various destinations. Recoveries expressed as percentage of maximum recovery for confidentiality purposes. Figure reproduced from Goodfellow and Dimitrakopoulos (2013).

Table 1: Processing costs and economic parameters. Note that all costs are expressed relative to a base cost (x) that is not disclosed.

Processing costs (relative to base cost x)	
Sulphide mill	\$11.30*x/tonne
Sulphide heap leach	\$2.98*x/tonne
Sulphide dump leach	\$1.87*x/tonne
Transition heap leach	\$2.15*x/tonne
Oxide heap leach	\$2.06*x/tonne
Economic parameters	
Copper price (including selling and G&A costs)	\$2.88 /lb Cu recovered
Gold price (including G&A costs)	\$1480 /oz Au recovered

The iterative process combining short-term production scheduling and state-dependent destination policy optimization described at the end of Section 2.3 was applied to this case study. In order to model geological uncertainty, 50 pre-computed equiprobable orebody simulations were used. Each of these simulations contains spatially correlated realizations for the block property values, namely the tonnage, the amount of gold and copper contained, and the material type. In order to avoid overfitting, the first 30 simulations were used for optimizing the schedule and the policy parameters, and the remaining 20 were used for computing risk profiles.

Remember that the iterative optimization involves the optimization of a function of the form

$$\frac{1}{|SC|}\sum_{sc=1}^{|SC|} Obj(dp, sched, sc),$$

where $Obj(dp, sched, sc)$ is the value of the objective function for destination policy dp, schedule $sched$, and scenario sc. For the current case study, Obj corresponds to short-term cash flows and has the general form

$$\sum_{t=1}^{T}\Bigg(Rev_t(Mill) - ProcCost_t(Mill) - StoppingCost_t(Mill) - CapPen_t(FP) + \sum_{HL}\Big(\mathbf{1}_{TT(HT_t)>LT}\big(Rev_t(HL) - ProcCost_t(HL)\big)\Big)\Bigg) \tag{3}$$

In the equation above, t is the time increment (corresponding to the time it takes to extract one block), and T is the total number of time increments (blocks) in the period of interest (e.g. year). For the results shown below, the period of interest is the second year of production. Note that, because the problem considered is short-term, there is no discounting. The mill stopping cost $StoppingCost_t(Mill)$ is equal to $\$308{,}000 * x$ for the first time step the mill needs to be closed, and $\$61{,}500 * x$ for each subsequent time step, where x is the base cost mentioned above.[1] The inner sum reflects the fact that revenue for the heap leaches occurs when the unleached material exceeds the "leaching tonnage" LT (once leached, that material is subtracted from the total tonnage for that heap leach). In order to write out the revenue and processing cost for each destination, as well as the penalty for surpassing feed pile capacity ($CapPen_t(FP)$), the following quantities are defined:

- $TT(FP_t)$ is the amount on the feed pile at time t. Because the mill processes a fixed tonnage (2,066 tonnes) from the feed pile at each time step if sufficient material is available, $TT(FP_t)$ changes over time according to

$$TT(FP_{t+1}) = \begin{cases} TT(FP_t) + TT(block_t) - \min\left(2066, TT(FP_t)\right), if\ dp(s_t) = d_M \\ TT(FP_t) - \min\left(2066, TT(FP_t)\right), otherwise \end{cases}$$

- $CuT(FP_t)$ is the amount of copper on the feed pile at time t. The feed pile is assumed to be homogenized, therefore $CuT(FP_t)$ changes over time according to

$$CuT(FP_t) = \begin{cases} CuT(FP_t) + CuT(block_t) - \dfrac{\min\left(2066, TT(FP_t)\right)}{TT(FP_t)} * CuT(FP_t), if\ dp(s_t) = d_M \\ CuT(FP_t) - \dfrac{\min\left(2066, TT(FP_t)\right)}{TT(FP_t)} * CuT(FP_t), otherwise \end{cases}$$

- $AuT(FP_t)$ is the amount of gold on the feed pile at time t, and it changes over time similarly to $CuT(FP_t)$
- $in_t^{TT}(dest)$ is the total tonnage processed at destination $dest$, and is computed as follows:
  - for the mill, $in_t^{TT}(Mill) = \min\left(2066, TT(FP_t)\right)$
  - leaching occurs when enough material has been heaped since the last leaching; therefore, $in_t^{TT}(dest) = \sum_{LLt \le i < t} TT\ (block_i)$ for each heap leach, where LLt is the last time step when leaching occurred for $dest$
- $in_t^{Cu}(dest)$ is the total copper tonnage processed by destination $dest$ in period t, computed as follows:
  - $in_t^{Cu}(Mill) = \frac{\min\left(2066, TT(FP_t)\right)}{TT(FP_t)} * CuT(FP_t)$
  - $in_t^{Cu}(dest) = \sum_{LLt \le i < t} CuT\ (block_i)$ if $dest \neq Mill$
- $in_t^{Au}(dest)$ is the total gold tonnage processed by destination $dest$ in period t, and is computed similarly to $in_t^{Cu}(dest)$
- $out_t^{Cu}(dest) = in_t^{Cu}(dest) \times Recovery^{dest}(in_t^{Cu}(dest)\ , in_t^{TT}(dest))$ is the total copper produced at $dest$ (for destinations that produce only gold, $out_t^{Cu}(dest) = 0$)
- $out_t^{Au}(dest) = in_t^{Au}(dest) \times Recovery^{dest}(in_t^{Au}(dest), in_t^{TT}(dest))$ is the total gold produced at $dest$ (for destinations producing only copper, $out_t^{Au}(dest) = 0$)
- p_{Cu} and p_{Au} are the selling prices for copper and gold, respectively
- c_{proc}^{dest} and c_{sell}^{dest} are the processing and selling costs, respectively, for $dest$

[1] The reason why the stopping cost appears to be so large is that x is expressed as a cost per tonne, whereas the stopping cost is incurred for every time step, which corresponds to the time it takes to process a whole block. Blocks weigh around 14,000 tonnes on average.

The revenue and processing cost for each destination can now be written as:

$$Revenue_t(dest) = p_{Cu} \times \ out_t^{Cu}(dest) + \ p_{Au} \times \ out_t^{Au}(dest)$$
$$ProcCost_t(dest) = \left(c_{proc}^{dest} + c_{sell}^{dest}\right) \times \ in_t^{TT}(dest)$$

The penalty for surpassing feed pile capacity is

$$CapPen_t(FP) = \ 25 * \max(0, \mathrm{TT}(\mathrm{FP_t}) - 500{,}000)^{1.05}$$

The state s_t for this problem includes the material type, tonnage and metal content of $block_t$, the tonnage and metal content for the feed pile material at time t, as well as the total tonnage and metal content of the unleached material for each of the heap leaches at time t. Based on this, a state-dependent policy is constructed using a problem-specific parameterization. Namely, the policy assigns $block_t$ to the destination that optimizes the trade-off between sending $block_t$ to the destination that results in the largest improvement in the objective function and ensuring that the mill feed pile does not run out of material. In order to formally write the policy, denote by Obj_t the term of the objective function in Equation 3 corresponding to time step t, and by $TT(FP_t)$ the feed pile tonnage at time t. The policy then has the parametric form

$$dp(s_t) = \arg\max_{\mathrm{d}}\left[(Obj_{t+1} - Obj_t) + c * \left(\max\left(0, TT_{min} - \ TT(FP_t)\right)^p\right)\right] \tag{4}$$

where c, TT_{min} and p are the parameters to be optimized. The max is taken over the destinations d that $block_t$ can be sent to. Several things to note are:

- The first term represents the immediate increase in objective function resulting from sending $block_t$ to destination d. The second term is there to balance this immediate benefit against the risk of having to close the mill at a future time due to insufficient material on the feed pile.
- Obj_{t+1}, Obj_t and $TT(FP_t)$ can be readily computed using knowledge of s_t and d, the destination where $block_t$ is sent to.
- The inner optimization over d is computationally trivial to perform, amounting to comparing a small number of different values (as many as there are destinations that accept $block_t$'s material type).

3.1 Numerical results

This section presents results obtained by comparing different destination policies for the case study described above. The “optimized policy” is obtained by optimizing the parameters c, TT_{min} and p for the state-dependent destination policy dp in Equation 4. The resulting optimization problem is non-linear, non-smooth, and non-convex, exhibiting a large number of local optima. Therefore, and also because the small dimensionality of the policy parameter space allows it, a combination of grid-search and the simplex method (Nelder and Mead, 1964) was used for optimizing the policy parameters. The first 30 orebody simulations were used for performing the optimization, and the remaining 20 were used for computing risk profiles.

The first comparison is between the optimized policy and a “max-block-value policy” that sends each block to the destination $dest$ that maximizes the value that would be obtained if only that block was processed at $dest$. For linear recoveries, this would be equivalent to sending $block_t$ to the destination that maximizes

$$MetalTons(block_t) \times Recovery^{dest} \times price - TT(block_t) \times ProcCost^{dest}$$

a quantity also known as “economic block value”. For non-linear recovery curves, as is the case in the case study herein, $Recovery^{dest}$ is dependent on the grade and is replaced by R$ecovery^{dest}(MetalTons(block_t), TT(block_t))$.

In order for these policies to be applied it is necessary to know the order in which the blocks are sent for processing. A simple heuristic way for designing a block order is to assume that material will be extracted bench-by-bench starting from the top. The resulting block order will be called a "top-down schedule" (this is, of course, different from the long-term schedule since all the allocated blocks for this case study belong to the same year of production).

Results comparing the optimized policy to the max-value policy for the block order given by the top-down schedule can be seen in Figure 6 and Figure 7, which show the amount of material on the feed pile and the objective function value, respectively. Keep in mind that the mill incurs the stopping cost every time the feed pile amount is zero. Also note that the monthly values of the objective function are equal to the monthly cash flows, because neither policy exceeds the feed pile upper bound so the capacity penalty is not incurred. Finally, although the allocation process increments the time step for each block allocation, it is trivial to aggregate the resulting block-by-block behaviour into the weekly or monthly behaviour seen in Figure 6 and Figure 7.

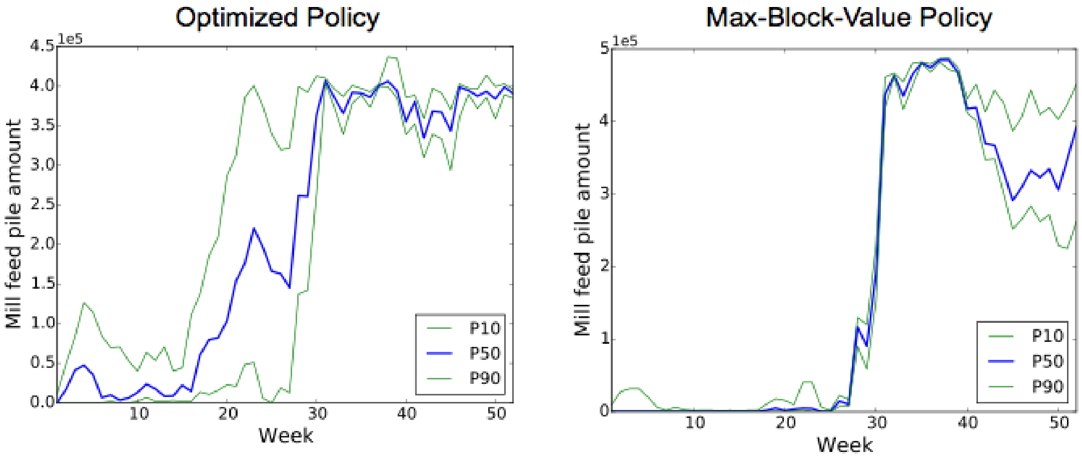

Figure 6: Risk profiles for the tonnage of material on the mill feed pile for the optimized policy and the max-block-value policy. Both policies use the block order provided by the heuristic top-down schedule.

Figure 6 show that the optimized destination policy is capable of utilizing mill capacity better than the max-block-value policy, resulting in a smaller duration for the empty feed pile and therefore a smaller stopping cost. The effect of this can be seen in Figure 7, which shows that the optimized policy is superior to the max-block-value policy in terms of cash flow.

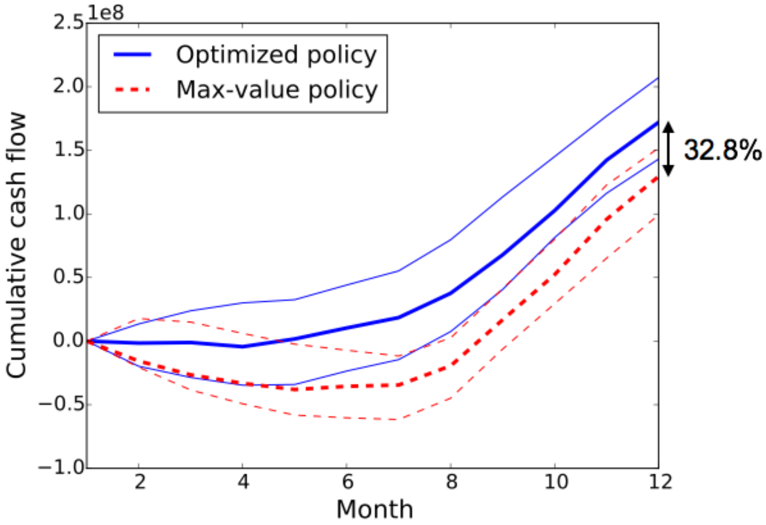

Figure 7: Cumulative cash flows for the optimized policy and the max-block-value policy using the block order provided by the top-down schedule.

While this first set of results shows that the optimized destination policy can help ensure more stable production and increase value, the second set of results answer the question of whether production stability and value can be increased even more by modifying the order in which the blocks are being sent for processing. This is done by comparing the top-down (bench-by-bench) schedule to a schedule that is optimized using simulated annealing, with the objective of maximizing the value resulting from applying the optimized destination policy. This optimization corresponds to Step 3 of the iterative process combining short-term production scheduling and state-dependent destination policy optimization described at the end of Section 2.3. The results are shown in Figure 8 (amount on feed pile) and Figure 9 (cumulative cash flow).

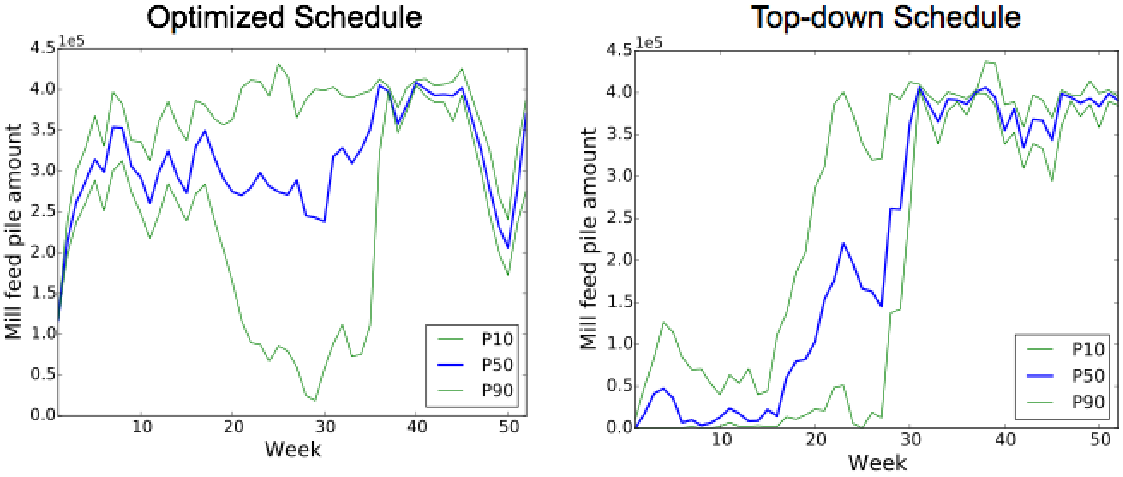

Figure 8: Risk profiles for the tonnage of material on the mill feed pile for the optimized schedule and the top-down schedule. The values for both graphs are obtained using the optimized destination policy.

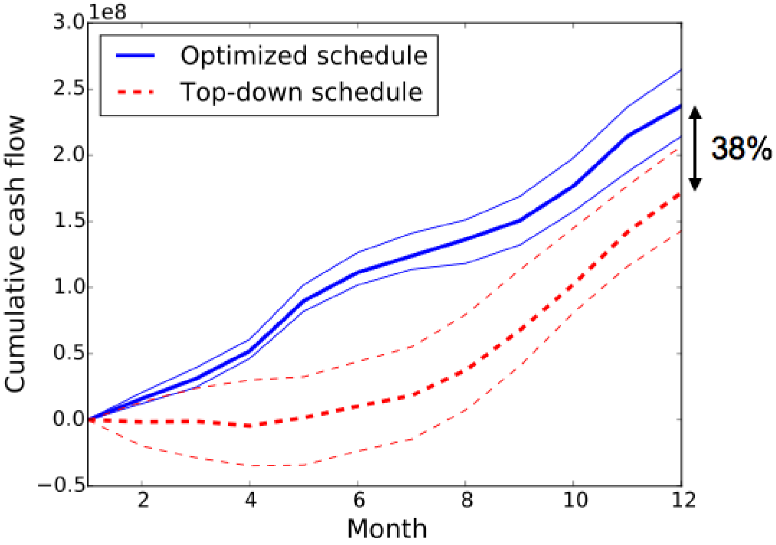

Figure 9: Cumulative cash flows for the optimized schedule and the top-down schedule.

The graphs show that there is a significant upside to optimizing the short-term schedule. For the part of deposit that this case study is looking at (blocks extracted in the second year), high-grade sulphide blocks tend to be located deeper in the ground. Therefore, the top-down schedule is unable to feed the mill for the first part of the year. The optimized schedule, on the other hand, balances extracting material that can go to the mill much better throughout the year, leading to increased value as seen in Figure 9. Experiments were also performed (not shown here) showing that for the optimized schedule the optimized destination policy still results in increased cumulative cash flows and better mill usage during the year with respect to the max-block-value policy.

This case study can also be used to illustrate the relationship between state-dependent destination policies and other types of destination decisions. It is fairly common for stochastic mine planning methods for mining complexes to adopt a two-stage approach, where the second stage optimizes processing decisions separately for each scenario. The block values computed based on the optimal per-scenario decisions are then fed back into the first-stage optimization, which produces a schedule that is robust over all scenarios (Goodfellow, 2014; Montiel, 2014). This is meant to model, much like state-dependent policies, the situation where new information is obtained before extracted material is sent for processing. However, it can lead to overly optimistic evaluations. For destination policies, for instance, the two-stage approach implicitly assumes that the properties of all blocks in a year (as given by a certain scenario) are being known when deciding where to send $block_t$. In contrast, a state-dependent policy would only assume that properties of blocks extracted up to and including time t are known when making an allocation decision for $block_t$. As such, the two-stage approach ends up assuming that more information is available than it will actually be the case (and than a state-dependent policy assumes). This is problematic because there the second-stage decisions will not actually be implemented in practice (since the scenario is never fully known in reality), and any evaluations should ideally be based on decisions that are actually implementable. Instead, the two-stage approach evaluates an idealized destination strategy where complete information is available for all the blocks extracted in a given year, leading to overly optimistic evaluations. A state-dependent policy, on the other hand, models the process of incorporating information as it becomes available.

In order to remediate this issue, a “nearest-neighbour” approach may be proposed. For destination decisions, this would involve establishing what scenario the observed reality is closest to (according to some metric) and sending each block to the optimal destination for that scenario. The idea of aggregating scenarios according to some metric is not new and has been proposed before for mine planning applications (e.g. Boland et al., 2008).

The feasibility of the nearest-neighbour approach was investigated using the case study herein. For this, a two-stage stochastic optimization was performed. The first stage variables determined the block extraction order, which was the same for all scenario. The second stage variables determined the destinations for each block, and a different set of decisions was optimized for each orebody simulation. Both stages were optimized using simulated annealing with the first 30 simulations, while the remaining 20 simulations (the “test simulations”) were used as representations of the unknown reality. The block extraction order for each test simulation was given by the first-stage variables from the two-stage optimization. The destination decisions were computed using the nearest-neighbour approach, with the distance between two simulations at time t being equal to the Euclidean distance between the vectors containing the normalized properties (tonnage and metal content) of bocks extracted up to and including time t. Two values for the expected cumulative cash flow were then computed:

- the first value was obtained by evaluating the plan for the 30 training simulations; the destination decisions for each of them were given by the second-stage decision variables, which were the result of an independent optimization for each simulation;
- the second value was obtained by evaluating the plan for the 20 test simulations; for each simulation, the destination decisions were computed using the nearest neighbour procedure as described above.

Similarly, two values were computed for the expected cumulative cash flows obtained by the combination of the state-dependent policy and optimized block order. The first value was computed using the 30 training simulations, which they were used for optimizing the policy parameters, and the second value was computed using the 20 test simulations. The results are shown below in Table 2.

Table 2: Cumulative cash flows computed using the training and test set of simulations for the state-dependent policy and the per-scenario two-stage approach.

	State-dependent	Per-scenario two-stage
Training set cumulative cash flow	$\$103{,}393{,}018 * x$	$\$112{,}623{,}861 * x$
Test set cumulative cash flow	$\$103{,}033{,}628 * x$	$\$94{,}944{,}505 * x$

The results in the table au-dessus show that the value for the plan that uses the state-dependent policy does not change much between the training and test set, suggesting that this is a robust approach. For the per-scenario two-stage plan, on the other hand, there is a large difference between the training set value and the test set value. This shows that evaluations reported using per-scenario decisions for the second-stage variables may not be robust, at least not unless there is a sound way of computing second-stage decisions that will actually get implemented in practice.

4 Conclusions and future work

This paper shows how state-dependent policies can be used for incorporating new information into short-term mine planning. The main mechanism for doing so consists of encoding solutions as parameterized functions of the state rather than encoding them as decision vectors. This allows the response to new information to be determined in advance for whatever the new information may be, and to formulate the search for the best response as a finite-dimensional optimization problem.

The case study in Section 3 illustrates the benefits of using state-dependent policies for determining how destination decisions should adapt to new information about the material that is allocated. Using state-dependent destination policies was shown to lead to better cash flows and more reliable mill usage compared to both a heuristic policy similar to maximizing economic block value, and to a policy that uses per-scenario decisions for the second stage of a two-stage stochastic optimization. Beyond value considerations, the type of short-term analysis illustrated by the graphs in Section 3 can point out where the risk lies in terms of both time (for instance, weeks 20 to 35 in Figure 8) and space (by identifying where the blocks extracted in that time frame are located within the deposit). It could also help mill operators decide in advance when is the best time to close the mill for maintenance.

The framework that this paper lays out for the inclusion of new information using state-dependent policy lays the groundwork for a variety of future improvements and extensions. One issue that has not been addressed is how to use specific data for updating the properties of the extracted material, as well as updating the remaining uncertainty. This can be approached using a variety of methods, from geostatistical methods such as conditional simulation with successive residuals (Jewbali and Dimitrakopoulos, 2011) to data assimilation methods such as ensemble Kalman filtering (Evensen, 1994) or particle filtering (Del Moral, 1997).

The case study in Section 3 used a fairly simple form for the parametric state-dependent policy. While this performed better than other approaches, it is worth investigating whether further gains can be made by using more complex policies and more advanced policy optimization methods.

Another direction for future research concerns more detailed modelling of aspects that are important for short-term planning. For the case study presented in this paper, this can involve accurately assessing the cost of stopping the mill based on energy, labour and opportunity costs. In general, the modelling of variables such as equipment availability, contractual requirements on the output, or seasonal variations in production in difficult climates may prove to be important for accurately assessing the value of a short-term plan.

References

Armstrong, M., Galli, A., & Razanatsimba, R. (2012). Using multistage stochastic optimisation to manage major production incidents. Mining Technology, 121(3), 125–131.

Benndorf, J., Buxton, M., & Shishvan, M. S. (2014). Sensor-based Real-time Resource Model Reconciliation for Improved Mine Production Control - A Conceptual Framework. In Orebody Modelling and Strategic Mine Planning, 223–233.

Boland, N., Dumitrescu, I., & Froyland, G. (2008). A Multistage Stochastic Programming Approach to Open Pit Mine Production Scheduling with Uncertain Geology. Optimization Online, 1–33.

Chatterjee, S. (2013). Vision-based rock-type classification of limestone using multi-class support vector machine. Applied Intelligence, 39, 14–27.

Del Castillo, F., & Dimitrakopoulos, R. (2014). Joint effect of commodity price and geological uncertainty over the life of mine and ultimate pit limit. Mining Technology, 123(4), 207–219.

Del Moral, P. (1997). Nonlinear filtering: Interacting particle resolution. Markov Processes and Related Fields, 2(4), 555–580.

Dimitrakopoulos, R. (2011). Strategic Mine Planning Under Uncertainty. Journal of Mining Science, 47(2), 138–150.

Dimitrakopoulos, R. G., & Sabour, S. a. A. (2007). Evaluating mine plans under uncertainty: Can the real options make a difference? Resources Policy, 32(3), 116–125.

Evensen, G. (1994). Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. Journal of Geophysical Research, 99(C5), 10143.

Goetz, A. F. H., Curtiss, B., & Shiley, D. A. (2009). Rapid gangue mineral concentration measurement over conveyors by NIR reflectance spectroscopy. Minerals Engineering, 22(5), 490–499.

Goodfellow, R. (2014). Unified Modelling and Simultaneous Optimization of Open Pit Mining Complexes with Supply Uncertainty. PhD thesis, McGill University.

Goodfellow, R., & Dimitrakopoulos, R. (2013). Global Asset Optimization of Open Pit Mining Complexes under Uncertainty.

Hoerger, S., Hoffman, L., & Seymour, F. (1999). Mine planning at Newmont's Nevada operations. Mining Engineering, 51(1), 26–30.

Horrocks, T., Wedge, D., Kovesi, P., Clarke, N., & Vann, J. (2015). Classification of Gold-Bearing Particles Using Visual Cues and Cost-Sensitive Machine Learning. Mathematical Geosciences, 47(5), 521–545.

Jewbali, A., & Dimitrakopoulos, R. (2011). Implementation of conditional simulation by successive residuals. Computers & Geosciences, 37(2), 129–142.

Montiel, L. (2014). On Globally Optimizing a Mining Complex under Supply Uncertainty: Integrating Components from Deposits to Transportation Systems. PhD thesis, McGill University.

Nelder, J. a., & Mead, R. (1964). A simplex method for function minimization. The Computer Journal, 7(4), 308–313.

Paduraru, C., & Dimitrakopoulos, R. (2014). Mineral Supply Chain Optimisation under Uncertainty Using Approximate Dynamic Programming. In Orebody Modelling and Strategic Mine Planning, 415–422.

Peevers, R., & Whittle, G. (2013). Enterprise Optimization for Mining Businesses. In SME Annual Meeting and Exhibit.

Pimentel, B. S., Mateus, G. R., & Almeida, F. a. (2013). Stochastic capacity planning and dynamic network design. International Journal of Production Economics, 145(1), 139–149.

Powell, W. (2014). Clearing the Jungle of Stochastic Optimization. Informs TutORials, 1–37.

Powell, W. B. (2007). Approximate Dynamic Programming: Solving the Curses of Dimensionality. Wiley-Interscience.

Stone, P., Froyland, G., Menabde, M., Law, B., Pasyar, R., & Monkhouse, P. (2007). Blasor — Blended Iron Ore Mine Planning Optimisation at Yandi. In Orebody Modelling and Strategic Mine Planning.

Whittle, G. (2007). Global Asset Optimisation. In Orebody modelling and strategic mine planning: Uncertainty and risk management models. AusIMM Spectrum Series 14, 331–336.

Whittle, G. (2010). Enterprise Optimisation. In Mine Planning and Equipment Selection (MPES) Conference, 105–117.