Les Cahiers du GERAD ISSN: 0711–2440

Mining Supply Chain Optimization under Geological Uncertainty

R. Goodfellow
R. Dimitrakopoulos

G–2013–54

September 2013

Mining Supply Chain Optimization under Geological Uncertainty

Ryan Goodfellow

Roussos Dimitrakopoulos*

COSMO – Stochastic Mine Planning Laboratory
Department of Mining and Materials Engineering
McGill University
Montreal (Quebec) Canada, H3A 2A7

ryan.goodfellow@mail.mcgill.ca

roussos.dimitrakopoulos@mcgill.ca

* *and GERAD*

September 2013

Les Cahiers du GERAD

G–2013–54

Abstract: Mining operations can be modelled as a supply chain from the sources of raw material (mines) through various processing streams to the final saleable products sold to customers. Existing research in mine design optimization has predominantly focused on the extraction sequence of materials and neglects the impact that selecting the optimal processing paths has on the economic viability of a mining operation. Additionally, the vast majority of existing models assume that the value of the extracted material must be calculated for each block individually, and neglects the opportunity to blend material together to prolong the life of the resource and generate higher cash flows. One major contributing factor for not modelling such complexities is the difficulty in modelling and optimizing with non-linear relationships that occur in the supply chain's processing streams.

This paper addresses the issue of selecting the optimal block destinations and processing streams throughout a mining supply chain for complex blending operations. This non-linear optimization formulation is solved using a particle swarm optimization algorithm. The k-means clustering technique is used to aggregate the processing decision variables and substantially reduce the size of the problem, leading to more efficient solution times with minimal loss in the quality of the resultant solution. By clustering the decision variables and applying the same decisions over a set of geological simulations, the proposed method also provides complex destination policies for material extracted from a mine under geological uncertainty, somewhat akin to a more advanced cut-off policy. This method is tested at Vale's Onça Puma nickel laterite deposit, located in Pará, Brazil.

Key Words: Supply chain optimization; extractive blending; clustering; process streams; geological uncertainty.

Acknowledgments: The work in this paper was funded from NSERC CDR Grant 335696 and BHP Billiton, NSERC Discovery grant 239019 and the members of the COSMO Stochastic Mine Planning Laboratory: AngloGold Ashanti, Barrick, BHP Billiton, De Beers, Newmont, and Vale. The authors would like to thank Jean-Yves Cloutier and the Mining Engineering Department at the Onça Puma mine for their help, advice and insight related to the case study.

1 Introduction

As the world's mineral resources are continuously being depleted, mining companies are seeking to profitably extract and process increasingly complex material. Through tremendous technological and operational improvements, mining companies can often profitably produce products from material that was once considered waste. This is often a result of blending material from various sources together to obtain a more homogenous and predictable material that meets a set of specifications prior to processing. As the options for sending materials through different processing streams increases, a supply chain is formed, where one needs to make optimal decisions on where to send material extracted from the ground through to final products that are sold on the market (Sandeman et al., 2010).

Conventional mine optimization methods are often myopic in their abilities to harness the power of blending material and processing streams within a supply chain. More often than not, these methods require the assumption that each block (a discretized unit of material) in an orebody has an economic value, and are independent of other blocks (Lerchs and Grossmann, 1965; Gershon, 1983; Caccetta and Hill, 2003; Ramazan and Dimitrakopoulos, 2004; Gleixner, 2008; Boland et al., 2009; Bienstock and Zuckerberg, 2010). To alleviate this issue, sets of constraints are placed in the mathematical formulations to provide limits on the incoming feed to the various destinations. These constraints can sometimes be misleading, in particular for grade blending constraints in stochastic integer programming formulations (Ramazan and Dimitrakopoulos, 2012), where the penalty costs associated with the constraint require the assumption of constant tonnage to ensure that the constraint is linear. Additionally, these conventional optimization models often assume a "one-and-done" approach to the processing streams, where the blocks are sent to a single destination, which is then transformed into a saleable product. Naturally, these assumptions can be over-simplified and limiting for complex mining operations. In more complex scenarios, it is possible to include stockpiles (Caccetta and Hill, 2003; Ramazan and Dimitrakopoulos, 2012), however these often require restrictive assumptions in order to have a linear model of the system.

The concept of blending optimization for mineral resource supply chains has been a topic of research over the past few decades, particularly in the coal and cement industries. Sarker and Gunn (1997) propose a successive linear programming approach tailored to solve non-linear coal-blending formulations, where portions of the non-linear program's sub-problems are fixed and solved successively using linear programming until convergence is obtained. Asad (2010) proposes a linear programming-based optimization model for short-range production in the cement industry. There have also been some applications in the metal mining industry, albeit specifics are rarely discussed for intellectual property reasons. Hoerger et al. (1999[1]) and Hoerger et al. (1999[2]) give an overview of their mixed integer linear programming (MILP) formulation for scheduling, stockpiling and processing for Newmont's large mining complex in Nevada. Wharton (2007) and Whittle (2007) show in several case studies the added value in blending and improving the performance of the supply chain, including the ability to identify bottlenecks. Stone et al. (2007) discuss an application of BHP-Billiton's in-house software (Blasor) to their Yandi deposit, an iron ore blending operation in Western Australia. Chanda (2007) formulates the blending and supply chain as a linear programming network where the objective is to minimize the costs of material flow through the network, and tests the method on a hypothetical supply chain with six mines, five concentrators, three smelters and two refineries. This method is, however, limited in ability to model intricacies in the supply chain.

More recently, Sandeman et al. (2010) present two blending and supply chain case studies solved using discrete event simulation (DES) and a linear optimization model. In the DES model, each physical item is considered as a discrete entity that has unique characteristics (Sandeman et al., 2010). While specifics are not discussed, the methodology is certainly one of the most advanced works with mining applications to date because it is able to efficiently analyze what-if scenarios for a seemingly infinite number of possibilities (e.g. stochastic processing times and delays) at each step in the supply chain.

Other researchers attempt to optimize non-linear blending problems through the use of metaheuristics. In particular, particle swarm optimization (PSO) (Kennedy and Eberhart, 1995) appears to be an attractive method because of its ability to solve both discrete, continuous and mixed (discrete with continuous) non-linear optimization problems, which permits a high degree of flexibility for modelling complex supply chains.

Li et al. (2011) apply a hybridized particle swarm method with genetic algorithms to a cement blending operation, where the genetic mutations are used to ensure population diversity as the solution converges. Yanbin (2008) apply a hybridized simulated annealing (Kirkpatrick et al., 1983; Geman and Geman, 1984) and particle swarm optimization algorithm to power coal blending. Both approaches show adequate results, but are limited in their applicability to model advanced cases with complex supply chains, i.e. beyond the first stockpile or plant.

The purpose of this paper is to introduce a generalized mine supply chain optimization methodology, which is optimized using a particle swarm optimization algorithm. First, a property-based modelling approach is discussed. Given the large size of the problem, a method using the k-means clustering algorithm (Lloyd, 1982) to aggregate initial block destination decisions is discussed. This concept is then extended to mining supply chain optimization under geological uncertainty, where the clustering is performed over a set of simulations, leading to more robust long-term decision making. A particle swarm optimization algorithm for optimizing the proposed mining supply chain models is then discussed. The method is then applied to Vale's Onça Puma mine, a nickel laterite deposit in Pará State, Brazil. Finally, conclusions and future work are presented.

2 Optimization of Mining Supply Chains with Blending

For the purpose of this approach, it is assumed that the production schedule for the mine(s) is static. The optimization method does not attempt to generate a new production schedule for each of the mines; this is a topic for future research. The goal is to develop a generalized optimization framework that can be applied to many different types of mining complexes and can be solved efficiently. In order to generalize this framework, an approach to modelling mining supply chains based on materials (and their respective properties) is developed. A generalized mixed-integer non-linear programming formulation can then be developed and tailored to solve a specific problem. Given the fact that evaluating the mining supply chain is by far the most computationally costly function during the optimization process, an aggregation of the initial block destinations is proposed using the k-means clustering algorithm; the aggregation is based on the geological properties of the model, and the optimization framework can be modified to solve the clustered destination problem. Since clustering can be performed over a set of geological simulations, rather than each simulation uniquely, it is possible to jointly optimize all simulations, which are tied together via the initial clustering; this is somewhat akin to a two-stage stochastic programming problem where the first-stage decision variables choose the destinations for the clusters and the recourse variables optimize the various processing streams based on the first-stage decisions made. Given that these types of problems are incredibly difficult to solve for large-scale problems, an efficient framework using particle swarm optimization with heuristics (Kennedy and Eberhart, 1995) is proposed.

2.1 Problem modelling methodology

2.1.1 Background and definitions

Consider a set of locations in the mining supply chain, L, which can be split into two subsets: $L_m \subset L$, which represents the set of mines, and $L_d \subset L$, which represents the set of destinations. Note that $L_m \cup L_d = L$ and $L_m \cap L_d = \emptyset$. Let $l_m \in L_m$ represent a mine from the set of mines, and $l_d \in L_d$ represent a destination from the set of destinations. Let $l \in L$ represent any location in the mining supply chain. Each mine $l_m \in L_m$ can be discretized into blocks (or aggregates thereof), which will be referred to as B_{l_m}, where each block in the set can be denoted as $b_k^{l_m} \in B_{l_m}$ where $k \in \{1, \ldots, N_{l_m}\}$ is used to index each block and N_{l_m} is the total number of blocks in B_{l_m}.

The life of the entire supply chain can be represented as $P = \{1, \ldots, P^{\max}\}$, where each $p \in P$ represents a discrete unit of time (e.g. a production period), and $P^{\max}$ is the index of the maximum operating period for the supply chain. Each period that location $l \in L$ operates is defined as $p_l \in P_l \subseteq P$, where the operating periods for location l, p_l, may or may not be contiguous. It is assumed that each block $b_k^{l_m}$ must have an assigned extraction period, denoted by $p_k^{l_m} \in P_{l_m}$, and this period is not chosen dynamically during the

optimization (i.e. fixed block extraction periods). Integrating production scheduling with the mining supply chain is a topic of future research.

Any destination $l \in L$ has a set of input materials, I_l, a set of transitional materials, T_l, and a set of output materials, O_l, whereby $i_l \in I_l$, $t_l \in T_l$ and $o_l \in O_l$ are used to denote a material from their respective sets. The mines can only have output materials, and do not receive materials for processing, i.e. for any mine $l_m \in L_m$, $I_{l_m} \cup T_{l_m} = \emptyset$. For simplicity, let the agglomerated set of materials for l be represented as $M_l = I_l \cup T_l \cup O_l$. Let O denote the unique set of all output materials, i.e. $O = \bigcup_{l \in L} O_l$, where each output material can be denoted by $o \in O$. It is not necessary that any $o_l \in O_l$ is unique when comparing to any material from another destination, $o_{l'} \in O_{l'}$, $l' \in L \backslash l$. It is noted that $\bigcap_{l \in L} O_l$ may or may not be a null set. For cases where $\bigcap_{l \in L} O_l \neq \emptyset$, this indicates that there are multiple locations that produce the same output material. For example, two mines may produce output material that would be categorized as oxides, or several refineries may produce gold bars.

Consider a directed graph $G(L, O_{lL'})$, where the nodes L represent the various locations, and $O_{lL'}$ are the directed edges that represent output material from location $l \in L$ and can be used as input material to a set of candidate destinations, $L' \subseteq L_d \backslash l$, i.e. output material $(o_l \in O_l) = (i_{l'} \in I_{l'}) \, \forall l' \in L'$. In the case where all cycles in $G(L, O_{lL'})$ can be removed, $G(L, O_{lL'})$ can form a directed acyclic graph using a topological sorting algorithm (Cormen et al., 2009). The sorted graph $G^{TS}(L, O_{lL'})$ now represents the flow of materials through the mining supply chain, and can be used as an ordering for computing updates when material flow is active throughout the supply chain. The case where $G(L, O_{lL'})$ contains cycles between the vertices is not considered in this methodology.

Each material $m \in M_l$ for location $l \in L$ has a set of properties, denoted herein as R_{m_l}. The individual properties can be denoted by $r_m \in R_{m_l}$. Using the sorted graph $G^{TS}(L, O_{lL'})$, consider a destination $l \in L$ and the set of destinations $L' \subseteq L \backslash l$ where $(i_l \in I_l) = (o_{l'} \in O_{l'}) \, \forall l' \in L'$, i.e. the destinations that produce material i_l as an output. The input property $r_m \in R_{m_l} \forall m \in I_l$ has a value $v^p_{r_m}$ for period p, where $v^p_{r_m}$ is assumed to be a linear addition of the incoming output properties, i.e.:

$$v^p_{r_m} = \sum_{l' \in L'} \sum_{m_{l'} \in O_{l'}} x\left(r_{m_{l'}}\right) v^{p'}_{r_{m_{l'}}} \tag{1}$$

where $x\left(r_{m_{l'}}\right) = 1$ if the property r^p_m has been specified to add property $r_{m_{l'}}$ or $x\left(r_{m_{l'}}\right) = 0$ otherwise. Some examples of properties that are linearly additive and can be used as input properties are total tonnages, metal tonnes, throughput hours, etc.; metal grade (%) is not a linearly additive property, hence is not acceptable. While initially this may appear confusing, the fundamental concept is simple: a property for an input material is a linear combination of a set of property values for the various output materials from the material sources. One example might be a mill adding the incoming tonnages from three different mines; two mines may feed only sulfides, and the other mine may feed a small amount of oxides and sulfides combined. Using two properties, it is possible to count the incoming tonnages for sulfides (from three mines) separately from oxides (from one mine), and treat them differently in the processing streams.

Any intermediate or output material properties, $r_m \in R_{m_l} \forall m \in T_l \cup O_l$ and $l \in L$, are functions of previously calculated properties from destination l, i.e. $r_m = f(R')$ where R' is the set of all properties for all $m \in M_l$. One common example might be a calculation of the grade, which is a non-linear function where the input metal tonnage property is divided by the total input tonnage property.

Given that the properties $r_m \in R_{m_l}$ have an ordering, it is possible to construct a directed graph $G(L, O_{lL'})$ for each destination $l \in L_d$, where the vertices, R_l, are all properties that must be evaluated, i.e. $R_l = \bigcup_{m \in M_l} R_{m_l}$, and the directed arcs, $A_{r_1 r_2}$, are used to describe the fact that some property $r_1 \in R_l$ must be evaluated prior to evaluating property $r_2 \in R_l \backslash r_1$. The property graph, $G_l(R_l, A_{r_1 r_2})$, is assumed to not contain cycles, and can therefore be sorted using a topological sort algorithm to yield $G^{TS}_l(R_l, A_{r_1 r_2})$. This graph can now provide the order of evaluation for each destination $l \in L_d$ for when new incoming material to the destination is added, through to the properties of the output materials.

2.1.2 Example

To solidify this concept, consider a simple copper mine where the mined blocks go to a concentrator then refinery; this ordering from mine to refinery defines the sorted material graph, $G^{TS}(L, O_{lL'})$. The concentrator's incoming properties might be the incoming copper metal (tonnes) and total tonnage. An intermediate (non-linear) property for copper grade can be generated, which is simply the tonnage of copper divided by the total tonnes processed. This intermediate copper grade can then be used to evaluate a non-linear recovery function, which can subsequently be applied to the copper tonnage to yield an output property, the recovered copper tonnage. Another transformed output property may be, for example, the total concentrate tonnage. These two properties can then be sent to the refinery, where the same sorts of transformations are applied until a final, refined, copper metal is produced and sold.

2.1.3 Algorithmic evaluation of supply chain

With the two graphs modelled, $G^{TS}(L, O_{lL'})$ and $G_l^{TS}(R_l, A_{r_1r_2}) \forall l \in L_d$, it is possible to now define the flow of materials from the mine through the supply chain into its final product form. Two sets of decision variables are used to define this flow. First, let the set $D\left(b_k^{l_m}\right)$ denote the set of candidate destinations for a block $b_k^{l_m}$, and $N\left(D\left(b_k^{l_m}\right)\right) = \sum_{d \in D(m_l)} 1$ define the number of candidate destinations for the block. The set $D\left(b_k^{l_m}\right)$ must take into account the extraction period of the block, $p_k^{l_m} \in P_{l_m}$ hence the availability of the candidate destinations in that period, i.e. for each destination $d \in D\left(b_k^{l_m}\right) \subseteq L_d$, $p_k^{l_m} \cap P_d \neq \emptyset$. Let $z_d\left(b_k^{l_m}\right) \in \{0,1\} \forall d \in D\left(b_k^{l_m}\right)$ represent the binary decision variable of whether or not to send block $b_k^{l_m}$ to destination d, where $z_d\left(b_k^{l_m}\right) = 1$ if $b_k^{l_m}$ is sent to d or 0 otherwise. Note that $\sum_{d \in D\left(b_k^{l_m}\right)} z_d\left(b_k^{l_m}\right) = 1$ to ensure that the block is only (and entirely) sent to a single destination.

Next, define $D(m_l)$ as the set of candidate destinations for output material $m_l \in O_l$ at location $l \in L_d$. Let $y_d^t(m_l) \in [0,1]$ define the proportion of output material m_l sent to destination $d \in D(m_l)$ in period t. At this point, it is necessary to define whether or not the location $l \in L_d$ is able to carry-over material to the next period, or it must all be sent out in a single period. One example of a destination that may carry-over material from one period to the next is a long-term stockpile that feeds a concentrator slowly over time. If the destination l is permitted to carry over material, it is assumed that $\sum_{d \in D(m_l)} y_d^t(m_l) \leq 1 \; \forall m_l \in O_l$, otherwise it must be guaranteed that all of the material leaves the process, i.e. $\sum_{d \in D(m_l)} y_d^t(m_l) = 1 \; \forall m_l \in O_l$.

For any given solution of variables, $z_d\left(b_k^{l_m}\right)$ and $y_d^t(m_l)$, it is now possible to evaluate the mining supply chain as follows:

1. For each block $b_k^{l_m}$, send (update) the linearly additive properties to its destination, d, where $z_d\left(b_k^{l_m}\right) = 1$. This is an incremental update per-block to Eq. (1).
2. For each destination d that received material in the previous step, evaluate the subsequent properties in the sorted property graph, $G_d^{TS}(R_d, A_{r_1r_2})$.
3. For each destination d evaluated in the previous step and each period $t \in T_d$ (from lowest to highest), send $y_{d'}^t(m_d)$ of output material $m_d \in O_d$ to the subsequent destination $d' \in D(m_d)$ in period t', where $D(m_d)$ is the set of destinations that accept material m_d as input, i.e. $m_d \in I_{d'}$. Since all outgoing properties $r_{m_l}^p \; \forall m_d \in O_d, p \in P_d$ associated with m_d are assumed to be linearly additive, the amount of this property supplied to d' can be multiplied by the proportion $y_d^t(m_d)$. It is noted that t' can be used as a lag time to control the delay between processes (e.g. long-term stockpiles).
4. For each destination d evaluated in the previous step that permits carrying over material and each period t, the remaining material must be carried over to the subsequent production period subsequent period, t'. i.e.:

$$v_{r_{m_l}}^{t'} += \left(1 - \sum\nolimits_{d' \in D(m_d)} y_{d'}^t(m_d)\right) v_{r_{m_d}}^t(m_d) \; \forall m_d \in O_d, t \in P_d, t' \in P_d, t < t'$$

5. If all destination vertices $d \in G^{TS}(L, O_{lL'})$ have been evaluated or there does not exist a vertex $d \in G^{TS}(L, O_{lL'})$ where material needs to be sent to a subsequent destination $d' \in D(m_d)$, quit. Otherwise, set $d = d'$ (where d' is defined in Step 3), and go to Step 2.

2.2 Supply chain optimization with particle swarm optimization

2.2.1 Optimization formulation

With the previously outlined modelling methodology, it is possible to perform optimization over the mining supply chain. While the proposed formulation is complex to describe mathematically, the fundamental concept behind the formulation is simple and flexible. Given that each process can have input, intermediate and output properties, the objective is to minimize some weighted addition of the property values along with minimizing the weighted deviations from the set of constraints (if desired). This might be as simple as maximizing the net present value of the metal produced from a process plant (negative weight in the objective function) minus penalties for deviating from set production capacities (positive weight in the objective function). Consider the following formulation:

Objective:

$$\min \sum_{l \in L_d} \sum_{t \in P_l} \sum_{m \in M_l} \sum_{r_m \in R_m} \left(c^t_{r_m} \cdot \left(v^t_{r_m}\right)^{p(v_{r_m})} + \overline{d^t_{r_m}} \cdot \left(\overline{a^t_{r_m}}\right)^{p(\overline{a_{r_m}})} + \underline{d^t_{r_m}} \cdot \left(\underline{a^t_{r_m}}\right)^{p\left(\underline{a_{r_m}}\right)} \right) \tag{2}$$

Subject to:

$$v^t_{r_m} - \overline{a^t_{r_m}} \leq \overline{P^t_{r_m}} \qquad \forall l \in L_d, m \in M_l, r_m \in R_m, t \in T_l \tag{3}$$

$$v^t_{r_m} + \underline{a^t_{r_m}} \geq P^t_{r_m} \qquad \forall l \in L_d, m \in M_l, r_m \in R_m, t \in T_l \tag{4}$$

$$\overline{a^t_{r_m}}, \underline{a^t_{r_m}} \geq 0 \qquad \forall l \in L_d, m \in M_l, r_m \in R_m, t \in T_l \tag{5}$$

where:

- $v^t_{r_m}$ is the value of property r_m in period t, where $r_m \in R_{m_l}$, $m \in M_l$ and $l \in L_d$.
- $a^t_{r_m}$ is positive deviation value to ensure that constraints (3) and (4) are satisfied.
- $c^t_{r_m} = \frac{j(r_m)}{(1+d(r_m))^t}$ is a cost associated with property r_m in period t. $j(r_m)$ is a constant value (e.g. 1 or -1) and $d(r_m)$ is a discount rate expressed as a decimal. One common application of the value $c^t_{r_m}$ is when maximizing the net present value, where $j(r_m) = -1$ (recall that this is a minimization problem) and $d(r_m)$ would be the discount rate associated with the time value of money.
- $\overline{d^t_{r_m}} = \frac{j\left(\overline{dev^t_{r_m}}\right)}{\left(1+d\left(\overline{dev^t_{r_m}}\right)\right)^t} \geq 0$ and $\underline{d^t_{r_m}} = \frac{j\left(\underline{dev^t_{r_m}}\right)}{\left(1+d\left(\underline{dev^t_{r_m}}\right)\right)^t} \geq 0$ are penalty costs associated with the deviation variables, $\overline{a^t_{r_m}}$ and $\underline{a^t_{r_m}}$. $j\left(\overline{dev^t_{r_m}}\right)$ and $j\left(\underline{dev^t_{r_m}}\right)$ are constant penalty costs that are used to weight the deviations (some constraints are more important to adhere to than others). $d\left(\overline{dev^t_{r_m}}\right)$ and $d\left(\underline{dev^t_{r_m}}\right)$are discount rates that can be used if necessary to relax the requirement on constraint deviations over time (e.g. geological risk discounting).
- $p(v_{r_m})$, $p(\overline{a_{r_m}})$ and $p\left(\underline{a_{r_m}}\right)$ are exponents associated with the property value v_{r_m} or deviation variables, $\overline{a_{r_m}}$ and $\underline{a_{r_m}}$, respectively, and can be used to heavily penalize high deviations and only slightly for lower deviations.

Note that any property r_m mentioned in the objective function is actually a linear or non-linear function of the decision variables $z_d\left(b^{l_m}_k\right)$ and $y^t_d(m_l)$, and that it is necessary to re-evaluate equations (2)–(5) each time the decision variables have been updated. For penalty costs $\overline{d^t_{r_m}}$ and $\underline{d^t_{r_m}}$ greater than zero, the deviations

from the constraints can be calculated using equations (6) and (7) given below, and the constraints can be removed entirely from the formulation.

$$\overline{a^t_{r_m}} = \max\left\{0, v^t_{r_m} - \overline{P^t_{r_m}}\right\} \qquad \forall l \in L_d, m \in M_l, r_m \in R_m, t \in T_l \tag{6}$$

$$\underline{a^t_{r_m}} = \max\left\{0, \underline{P^t_{r_m}} - v^t_{r_m}\right\} \qquad \forall l \in L_d, m \in M_l, r_m \in R_m, t \in T_l \tag{7}$$

2.2.2 Solution encoding scheme

Consider a solution vector, Sol, of size $N + M$, where $N = \sum_{l_m \in L_m} N_{l_m}$, $M = \sum_{l_d \in L_d} \sum_{t \in T_{l_d}} \sum_{m_{l_d} \in O_{l_d}} \sum_{d' \in D(m_{l_d})} 1$, N_{l_m} is the number of blocks in mine $l_m \in L_m$, $l_d \in L_d$ is a destination in the set of destinations L_d, T_{l_d} is the set of active periods for destination l_d, O_{l_d} is the set of output materials for l_d and $D(m_{l_d})$ is the set of destinations that accepts material m_{l_d}. In other words, the first N elements of Sol are used to represent the block destination solution $z_d\left(b^{l_m}_k\right)$, and the remaining elements are $y^t_{d'}(m_{l_d})$, the proportions of output material m_{l_d} going to destination d' from destination l_d in period t.

Each of the elements sol_j $\forall j \in \{1, \ldots, N\}$ are encoded based on the number of candidate destinations, where the values of sol_j range from 0 to $N\left(D\left(b^{l_m}_k\right)\right)$, the maximum number of candidate destinations for block $b^{l_m}_k$. A decoding scheme is applied to convert the value from $sol_j \in \left\{0, \ldots, N\left(D\left(b^{l_m}_k\right)\right)\right\}$ to the block's destination d where $z_d\left(b^{l_m}_k\right) = 1$. The specifics of this encoding and decoding scheme are trivial and therefore not discussed. This automatically ensures that there is only a single destination for each block, and that the block is only sent to the destination once. The remaining elements of sol_j $\forall j \in \{N+1, \ldots, M\}$ must strictly adhere to the constraint that $\sum_{d \in D(m_l)} y^t_d(m_l) \leq 1$ $\forall m_l \in O_l$ for destinations that carry over material from one period to the next, or $\sum_{d \in D(m_l)} y^t_d(m_l) = 1$ $\forall m_l \in O_l$ for processes that do not. The correctness of the solution can be ensured by adding all sol_j $\forall j \in \{N+1, \ldots, M\}$ related to destination $l \in L_d$ and output material $m_{l_d} \in O_{l_d}$ in period t. In the case where $\sum_{d \in D(m_l)} y^t_d(m_l) > 1$ $\forall m_l \in O_l$, the variables, sol_j $\forall j \in \{N+1, \ldots, M\}$, can be corrected by dividing by the total proportion of m_l sent from destination l in period t, $\sum_{d \in D(m_l)} y^t_d(m_l)$.

2.2.3 Aggregation of initial destination decisions using k-means clustering

For realistic mining supply chains, it is often true that the number of block destination decision variables, N, is substantially larger than the number of processing decision variables, M, where N can be in the millions to hundreds of millions. Given that the vast majority of the time spent is evaluating the flow of material from the mines to the initial destination in the supply chain (Section 2.1.3), it is of interest to reduce the amount of block destination decision variables. Some methods have been proposed in the literature to aggregate and disaggregate processing decision variables in the mine production scheduling problem (Gleixner, 2008, Boland et al., 2009). These methods rely on duality theory and linear formulations of the problem, and are therefore not necessarily the best option for mine supply chain optimization, which often requires non-linear functions and complex constraints on the blended material. One popular method for aggregating information is k-means clustering (Lloyd, 1982), whereby groups of information can be identified based on the similarity of their basic properties. Intuitively, aggregating processing decisions reduces the resolution on which decision-making can be performed, and therefore would yield lower value than a solution based solely on block destinations. It has been experimentally found that this, however, is not the case, and that with a larger number of clusters, optimization takes much longer without significant improvements in the value of the objective function (Eq. (2)).

Consider the subset of blocks $B^m_l \subseteq B_l$ for any mine $l \in L_m$ in the mining supply chain that share a common material type $m \in O_l$. All blocks can be aggregated into $n^{lm}_{clusters}$ clusters, where each cluster is assigned a multi-dimensional position (location) based on the average value of the properties for the blocks that are closest to the cluster. Consider the set of properties $R_m \subseteq R_{m_l}$, where R_{m_l} is the set of block properties for material $m \in O_l$ at mine $l \in L_m$ (e.g. grades, tonnages, or economic value). Each property

$r \in R_m$ is used to define a dimension of the multi-dimensional cluster position. Without going into substantial detail about the k-means clustering algorithm as it is widely used and accepted, it is sufficient to state that each $r \in R_m$ is standardized by the average of the values in the subset R_m to avoid magnitude relationships that are common in mining data. If this standardization is not performed, it is possible that some properties will take clustering precedence over others – one example might be tonnage, which can vary in value quite a bit, but when compared to the average, varies very little (small coefficient of variation), whereas a metal grade may not vary much in value, but when compared to the average, varies substantially (large coefficient of variation). Averaging the properties tends to give more precedence to variability in grades versus tonnages, which is often more desirable. For the proposed methodology, the Euclidean distance metric is used to compare the distance between the property locations of blocks and clusters.

Recall from Section 2.1.1 that output properties sent from destination $l \in L$ to destination $l' \in L$ must be linearly additive (e.g. tonnages, metal content, throughput-hours). When the concept of clustered or aggregated groups of blocks is considered, a significant proportion of the data can be pre-processed by creating a generalized set of properties for each cluster, $R^{ml}_{cluster} \subseteq R_{m_l}$, where the value each property $r^t_{cluster} \in R^{ml}_{cluster}$ is the sum of the respective property values from each of the blocks that are members of the cluster and are mined in period t.

It is now possible to remove the integer block destination variables $z_d\left(b^{l_m}_k\right)$ entirely when modelling the problem and replace the variables with integer cluster destination variables. Let $clus^{lm}_k$ denote the k^{th} cluster for material for mine $l \in L_m$ and material $m \in O_l$. Additionally, let $z^t_d\left(clus^{lm}_k\right)$ denote where to send $clus^{lm}_k$ in period t. Recall that material $m \in O_l$ is used to define the set of candidate destinations for material and all blocks within an aggregate or cluster are of the same material, hence the set of candidate destinations for cluster $clus^{lm}_k$ is $D\left(m_l\right)$.

The size of the mining supply chain Sol' solution vector now becomes $N' + M$, where $N' = \sum_{l \in L_m} \sum_{t \in T_l} \sum_{m \in O_l} n^{lm}_{clusters}$, and the definition of M remains unchanged. The first N' variables in Sol now represent the destination of cluster $clus^{lm}_k$ in period t, i.e. the destination where $z^t_d\left(clus^{lm}_k\right) = 1$. It is noted that similar to the block destinations, each position $sol_j \ \forall j \in \{1, \ldots, N'\}$ ranges between $\left[0, N\left(D\left(clus^{lm}_k\right)\right)\right]$, where $N\left(D\left(clus^{lm}_k\right)\right)$ is the number of candidate destinations for the cluster, i.e. $N\left(D\left(clus^{lm}_k\right)\right) = \sum_{d \in D(m_l)} 1$. The proposed PSO optimization algorithm to be proposed in Section 2.2.5 can then use Sol' in place of the original solution vector Sol without changing the mechanics of the optimization algorithm itself.

2.2.4 Mining supply chain optimization under geological uncertainty

Moving from block destination decision variables $z_d\left(b^{l_m}_k\right)$ to cluster destination decision variables $z^t_d\left(clus^{lm}_k\right)$ provides an opportunity to optimize the mining supply chain when considering geological uncertainty. Consider each mine $l \in L_m$ is now represented by a set of geological simulations S_l, where each simulation $s \in S_l$ may contain simulated properties R^s_l. Using the mathematical model outlined in equations (2) to (5), it would be necessary to perform the optimization for each $s \in S_l$ individually (or in the case of multiple mines, each combination of simulations across the various mines), which requires a new solution vector Sol' for each scenario. Clustering across a set of geological simulations S_l not only reduces the quantity of decision variables, but permits aggregating the initial cluster destination decision variables across the set of simulations, effectively uniting them into a two-stage non-linear mixed integer stochastic programming problem, where the first-stage (discrete) decision variables decide the optimal destinations of each cluster $clus^{lm}_k$, and the second-stage (continuous) recourse variables decide the optimal processing streams once the random variables (the cluster properties R^s_l) are revealed at the initial destination. For a review of two-stage stochastic programming problems, see Birge and Louveaux (1997).

A new solution vector, Sol'', requires $N' + M'$ decision variables, where N' remains unchanged from the previous section, and M' is now $M' = M \times \left(\prod_{l \in L_m} \left(\sum_{s \in S_l} 1\right)\right)$, which takes into account all combinations of mine simulations. The continuous processing stream decision variables are then renamed to $y^{ts}_{d'}\left(m_{l_d}\right)$, which is used to denote the proportion of output material $m_{l_d} \in O_{l_d}$ from destination $l_d \in L_d$ sent to destination $d' \in L_d \backslash l_d$ in period t and scenario s, where $s \in S = \{1, \ldots, M'\}$ is used to denote the possible multiple mine simulation combination.

Consider the set of destinations $L_{init} \subset L$ that can be fed by a mine, i.e. $\forall l_{init} \in L_{init}$ there exists some cluster where $z^t_{l_{init}}\left(clus^{lm}_k\right) = 1$. Let $L^t_{opp} \subseteq L \backslash L^t_{init}$ be the set of destinations further along the supply chain that are not fed directly by mines, where $L^t_{opp} \cup L^t_{init} = L$. The objective function in Eq. (2) can now be modified to represent a two-stage mixed integer non-linear stochastic programming problem, as shown next.

Objective:

$$\begin{aligned}
\min \; & \underbrace{\sum_{l\in L_{init}} \sum_{t\in P_l} \sum_{m\in M_l} \sum_{r_m\in R_m} E\left\{c^t_{r_m} \times \left(v^t_{r_m}\right)^{p(v_{r_m})}\right\}}_{\text{First stage decisions}} \\
& + \underbrace{\frac{1}{M'} \sum_{l\in L_{init}} \sum_{t\in P_l} \sum_{m\in M_l} \sum_{r_m\in R_m} \sum_{s\in S} \left(\overline{d^t_{r_m}} \times \left(\overline{a^{ts}_{r_m}}\right)^{p(\overline{a_{r_m}})} + \underline{d^t_{r_m}} \times \left(\underline{a^{ts}_{r_m}}\right)^{p\left(\underline{a_{r_m}}\right)} \right)}_{\text{Recourse 1}} \\
& + \underbrace{\frac{1}{M'} \sum_{l\in L_{opp}} \sum_{t\in P_l} \sum_{m\in M_l} \sum_{r_m\in R_m} \sum_{s\in S} \left(c^t_{r_m} \times \left(v^{ts}_{r_m}\right)^{p(v_{r_m})} + \overline{d^t_{r_m}} \times \left(\overline{a^{ts}_{r_m}}\right)^{p(\overline{a_{r_m}})} + \underline{d^t_{r_m}} \times \left(\underline{a^{ts}_{r_m}}\right)^{p\left(\underline{a_{r_m}}\right)} \right)}_{\text{Recourse 2}}
\end{aligned} \tag{8}$$

Subject to:

$$v^{ts}_{r_m} - \overline{a^{ts}_{r_m}} \leq \overline{P^t_{r_m}} \qquad \forall l \in L_d, m \in M_l, r_m \in R_m, t \in T_l, s \in S \tag{9}$$

$$v^{ts}_{r_m} + \underline{a^{ts}_{r_m}} \geq P^t_{r_m} \qquad \forall l \in L_d, m \in M_l, r_m \in R_m, t \in T_l, s \in S \tag{10}$$

$$\overline{a^{ts}_{r_m}}, \underline{a^{ts}_{r_m}} \geq 0 \qquad \forall l \in L_d, m \in M_l, r_m \in R_m, t \in T_l, s \in S \tag{11}$$

where:

- $v^{ts}_{r_m}$ is the value of property r_m in time t and simulation s.
- $E\left\{c^t_{r_m} \times \left(v^t_{r_m}\right)^{p(v_{r_m})}\right\} = \frac{c^t_{r_m}}{M'} \sum_{s\in S} \left(v^{ts}_{r_m}\right)^{p(v_{r_m})}$ defines the expected values of the properties r_m when sent to the initial destinations.
- $\overline{a^{ts}_{r_m}}$ and $a^{ts}_{r_m}$ are deviations from the targets $\overline{P^t_{r_m}}$ and $P^t_{r_m}$ for property r_m in period t and simulation s.
- $c^t_{r_m}$, $\overline{d^t_{r_m}}$, $d^t_{r_m}$, $p\left(v_{r_m}\right)$, $p\left(\overline{a_{r_m}}\right)$ and $p\left(\underline{a_{r_m}}\right)$ remain unchanged from their original definition in Section 2.2.1.

Recall that the first-stage properties $v^t_{r_m}$ are (non)-linear functions of the first-stage binary decision variables, $z^t_{l_{init}}\left(clus^{lm}_k\right)$; the first set of recourse deviation variables penalize the random events at the first-stage destinations. The second set of recourse (second-stage) properties are (non)-linear functions of the continuous processing variables $y^{ts}_{d'}\left(m_{l_d}\right)$. If the optimization problem in Eq. (8)–(11) were to be solved for each geological scenario individually, where the scenarios denote the combinations of simulations from the various sources (mines), it is possible to average to obtain the wait-and-see (WS) solution (Birge and Louveaux, 1997). Coupled with a solution from the previous optimization problem, it is possible to obtain the expected value of perfect information (EVPI), the value one would pay to know the geology with certainty in order to optimize their mining supply chain.

In a simplistic case where the supply chain consists of a mill and a waste dump, the second set of recourse variables would drop out, and Eq. (8) could be written similar to the two-stage stochastic integer program (SIP) proposed by Ramazan and Dimitrakopoulos (2012), with the exception of the mining capacity constraints, which are assumed to be constant given the fixed production schedule and therefore do not influence the optimization of the supply chain. This formulation and model, of course, does not solve

production scheduling, but rather the optimal destination policies for material coming from the mines and the subsequent processing streams.

The proposed method for clustering and optimizing the mining supply chain under geological uncertainty also has many practical implications for operational mines. Recall that a solution to the problem outlined in Eq. (8) to (11) generates destination decisions for each cluster, where the clusters have been defined using all simulations for a mine and material. Consider the (almost certain) case where there is not a geological simulation that represents exactly what is in the ground. For any block extracted, it is possible to calculate its cluster membership by finding the cluster that is closest in terms of block properties. This block should then be sent to the destination that the cluster is sent to in the stochastic optimization problem solution. In practical terms, this provides an opportunity for very complex policies for operating mines, somewhat akin to cut-off grades, values or revenues. Since these initial cluster destination decisions have been made for potentially complex mining supply chains, these destination policies are more useful in that they address the core issue that cut-off policies cannot: blending operations do not consider the value of an individual block, a key assumption required for the application of cut-off grades or values.

2.2.5 Mining supply chain optimization using particle swarm optimization

Given that mixed integer non-linear programs can be extremely complicated to solve for problems of this magnitude using mathematical programming, an algorithm using particle swarm optimization (PSO) with local heuristics is used. PSO is a population-based metaheuristic that uses cognitive and social behaviour between simulated particles to achieve a high quality (but not necessarily optimal) solution in a reasonable amount of time (Kennedy and Eberhart, 1995). Unlike many other metaheuristics, the proposed solution scheme is suited for PSO because the method is capable of handling mixed integer non-linear problems with ease. The goal of PSO is to have all particles locate optima in a multi-dimensional space, where each particle is initially assigned random positions and velocities. At each position update, the objective function is sampled (i.e. the supply chain is evaluated), and the swarm's best-valued solutions are given more weight. Through exploration and exploitation of known high-quality solutions, the particle cluster converges together around an optimum or several optima (Brownlee, 2011).

Consider a set of N_p particles, where each particle is represented by its own distinct solution vector Sol, Sol' or Sol'' (depending on whether the problem is for block destinations, deterministic cluster destinations or stochastic cluster destinations, respectively). The current solution vector, or position vector, for a particle p will herein be denoted by x^p. Additionally, consider three vectors that are of the same size as the solution vector: vel^p, a particle velocity vector; $x^{p,best}$, which stores particle p's best solution found to date; and x^{global}, a global best solution vector that stores the positions of the best solution found throughout the algorithm. At iteration $t+1$, a particle's velocity is updated by using:

$$v^p(t+1) = c_1 v^p(t) + \left(c_2 \times rand() \times \left(x^{p,best} - x^p\right)\right) + \left(c_3 \times rand() \times \left(x^{global} - x^p\right)\right) \tag{12}$$

where $v^p(t+1)$ is the new velocity for the p^{th} particle, c_1, c_2 and c_3 are weighting coefficients for the particle's current inertia, its personal best solution and the global best solution, respectively, and $rand() \in [0,1]$ is a random uniform random number. The particle's velocity is therefore a combination of it's own inertia at the previous iteration, and an attraction towards it's best solution and the global best solution. The particle's position is then updated at iteration $t+1$ using:

$$x^p(t+1) = x^p(t) + v^p(t+1) \tag{13}$$

where if any element x_j^p of the particle exceeds its bounds (defined in Section 2.2.2), x_j^p will be set to the closest (minimum or maximum) bound.

After the velocity and position updates are performed for each particle, the solution is then evaluated by using the methodology for supply chain evaluation presented in Section 2.1.3. Let $f(x^p(t))$, $f\left(x^{p,best}\right)$ and $f\left(x^{global}\right)$ denote the current, best and global best objective function values, respectively, from Eq. (2) at iteration t. If $f(x^p(t)) \leq f\left(x^{p,best}\right)$, update the particle's best solution vector to $x^{p,best} = x^p(t)$. If $f(x^p(t)) \leq f\left(x^{global}\right)$, update the global best solution vector to $x^{global} = x^p(t)$.

2.2.6 Localized improvements using heuristics

Several types of heuristics can be employed to quickly improve the quality of a solution and ensure that the particle is not trapped in a local minimum. Rather than applying heuristics on each of the particles (which can be time consuming) consider the global best particle solution vector $x^{global}(t)$ found to date at iteration t of the PSO algorithm. Experimentally, it has been seen that it is often useful to employ one of the four heuristics:

1. When the particles have converged, freeze the block or cluster destinations, $z_d\left(b_k^{l_m}\right)$ or $z_d^t\left(clus_k^{lm}\right)$ and perform the next set of iterations solely changing the processing stream decisions, $y_{d'}^{ts}(m_{l_d})$.
2. Restart each of the particles from the global best solution, $x^{global}(t)$, with new, random velocity vectors.
3. Randomly select a block or cluster destination decision variable, $z_d\left(b_k^{l_m}\right)$ or $z_d^t\left(clus_k^{lm}\right)$ and change it to a different candidate destination. Accept the new solution if it yields an improvement in the objective function value, otherwise reject.
4. Randomly select a destination decision $y_{d'}^{ts}(m_{l_d})$ from the global best particle solution vector and randomly choose whether to multiply or divide the value by 2. The related processing decision variables need to be re-normalized to guarantee material flowing from the destination does not exceed the amount flowing in. Accept the new solution if it yields an improvement in the objective function value, otherwise reject.

3 Application at Vale's Onça Puma Nickel Laterite Mine

The proposed methodology is tested at Vale's Onça Puma nickel laterite deposit. While the model used for this case study does not demonstrate a full supply chain from pit to port, the blending streams are complex and can be used to highlight the features of the proposed method. For confidentiality purposes, most of the economic values and production capacities have been modified from what the mine uses in reality or are withheld.

3.1 Overview of the operation

The Onça Puma nickel laterite mining complex is located 20 km north of Ourilândia do Norte, in Pará State, Brazil, and primarily uses a side-hill mining method. The Onça and Puma deposits are situated approximately 16 km apart, with an overall concession of approximately 40 km^2. The primary chemical constituents of interest within the nickel laterite profile are nickel, cobalt, silica, magnesia, alumina and iron. A strict control must be placed on the silica (SiO_2) to magnesia (MgO) ratio and iron (Fe) content to ensure that the plant operates with minimal disruptions. A failure to deliver the appropriate plant feed blend can have significant economical and operational implications.

The lithologies of the ore deposits can be simplified into three distinct layers: limonite, which has been completely altered by chemical weathering and contains some nickel, low MgO and high Fe; saprolite, which is the primary source of nickel for the ferronickel pocess and contains of low Fe and high MgO; and bedrock, which consists of low nickel grades, high MgO and low Fe. For the purpose of this study, only the Onça deposit will be considered. Within the deposit, five material types are considered and are classified as: bedrock, limonite, saprolite waste (less than a predetermined nickel waste cut-off), low-grade saprolite (between the nickel waste cut-off grade and intermediate cut-off grade) and high-grade saprolite (above the intermediate nickel cut-off grade). While the mine uses a stockpile classification scheme based on the nickel grade, silica to magnesia ratio and material type, the previously outlined classification scheme based solely on nickel grade is used in order to let the optimizer a high degree of flexibility in decision-making.

Figure 1 shows a diagram for material flow at the mine used for this study. The alternative stockpile scheme has seven initial destinations: three low-grade stockpiles, three-high grade stockpiles and a waste dump. The low- and high-grade stockpiles are short-term stockpiles with a total capacity of $T_{\max}^{SP}$ dry tonnes, where the capacities are all equal in size but the tonnage is withheld. For this study, material is permitted to

carry-over between periods in these intermediate stockpiles. Each of the six intermediate stockpiles then feed two homogenization piles, each having a capacity of $T^H_{\max}$ dry tonnes per period. The homogenization piles alternate between feeding the pyrometallurgy processing plant and being filled by the intermediate stockpiles every 36 days (considered as one production period in this study); during the filling period, material cannot flow from the homogenization pile to the processing plant. The processing plant has a $T^H_{\max}$ dry tonne capacity, and has strict requirements on the blended feed from the homogenization piles. The silica-to-magnesia ratio is of primary importance and must lie between 1.5 and 1.8. Additionally, the plant has a requirement that the iron grade (Fe) lies between 12% and 16%. Because of these strict constraints on the blended feed to the processing plant, conventional mine planning optimizers are often inapplicable and generate unrealistic solutions due to the assumption that the value can be calculated based on the block's nickel grade; in the case of a nickel laterite mining operation, the nickel grade is of secondary importance to the blended plant feed specifications.

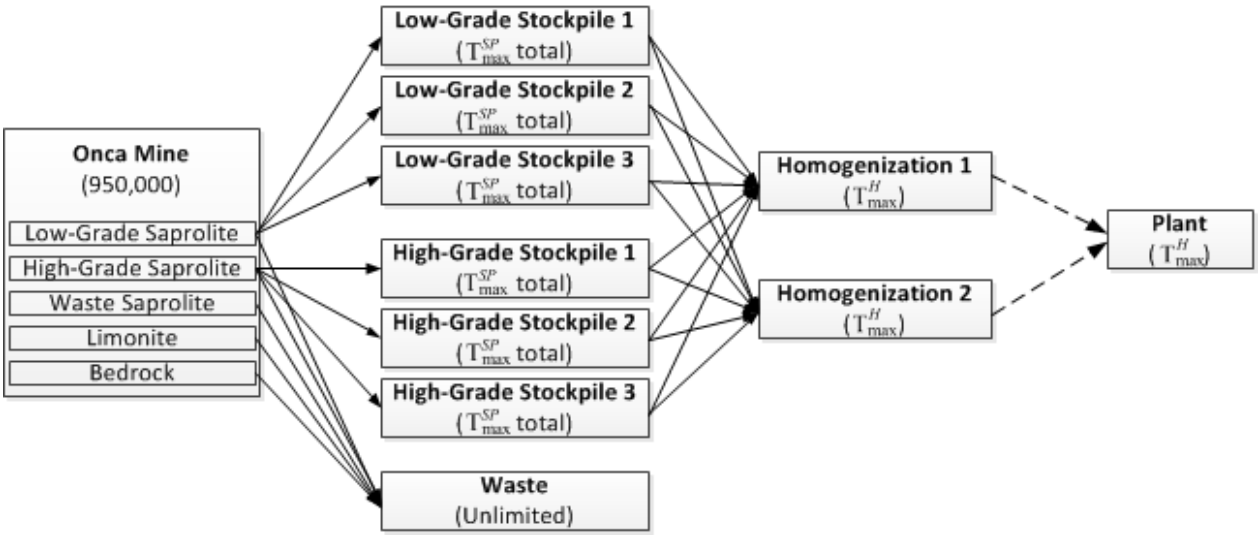

Figure 1: Schematic of material flow at the Onça Puma nickel laterite mining operation.

A set of 10 simulations of the lateritic profile was simulated using the single normal equation simulation (SNESIM) algorithm (Strebelle, 2002) for the Onça deposit; in particular, the limonite, saprolite and bedrock layers were simulated to quantify the high variability and volumetric uncertainty for the deposit. For each lithological simulation, the saprolite layer is retained for further joint simulation of nickel, magnesia, silica, iron and the dry tonnage factor using the direct block Min/Max Autocorrelation Factor simulation method (Boucher and Dimitrakopoulos, 2009; Goodfellow et al., 2012). The simulations show substantial variability from one simulation to the next, and, as will be seen in the proceeding sections, is quite challenging for the proposed method to find an adequate solution. In total, 20 simulations are used for the Onça deposit on 12.5 $\times$ 12.5 $\times$ 3 m^3 blocks in the x-, y- and z-directions, respectively. It is noted that the mine generally does short-term production scheduling on 6.25 $\times$ 6.25 $\times$ 3 m^3 block sizes, however the larger block sizes were kept for comparative purposes with the estimated (kriged) models supplied by the mine.

The mine production schedule is based on 36-day periods (the time to fill a homogenization pile). The production schedule is generated from a portion of the mine's annual production schedule by sequentially mining each bench until the short-term total mining capacity is met (950,000 dry tonnes). It is noted that because this study does not consider the Puma deposit, the schedule has been condensed to represent continuous mining at Onça. The production schedule therefore spans over 44 consecutive periods and contains 69,877 blocks.

3.2 Proposed models

The Onça Puma mining complex (Figure 1) is modelled using the methodology proposed in Section 2.1. In specific, the block, nickel, silica, iron and magnesia tonnages are properties that are passed through the supply chain from the mine through the stockpiles and homogenization piles, and finally to the plant. It is assumed that the intermediate low- and high-grade stockpiles are able to carry over material from one period to the next for strategic purposes, with a lag time of one period. The homogenization piles are not permitted to carry over material between production periods, thus all material that goes to either of these piles is sent to the plant in the subsequent period. At no point does the algorithm consider the economic value of an individual block; the economic value of the processed metal, the silica-to-magnesia ratios and the iron grade are calculated at the plant for the blended material coming from the homogenization piles.

3.2.1 Deterministic model

The optimization model for the optimization of deterministic orebody models (not considering geological uncertainty) is as follows.

Objective:

$$\begin{aligned}\max \sum_{t=1}^{44} \Bigg(& \frac{v^t_{\text{MV}}}{(1.008)^t} - \sum_{s=1}^{6} \overline{d^t_{SC_s}} \left(\overline{a^t_{SC_s}}\right) - \overline{d^t_{MC}} \left(\overline{a^t_{MC}}\right) - \underline{d^t_{MC}} \cdot \left(\underline{a^t_{MC}}\right) \Bigg) \\ & - \sum_{t=1}^{44} \left(\overline{d^t_{MSiMg}} \left(\overline{a^t_{MSiMg}}\right) + \underline{d^t_{MSiMg}} \cdot \left(\underline{a^t_{MSiMg}}\right) + \overline{d^t_{MFe\%}} \left(\overline{a^t_{MFe\%}}\right) + \underline{d^t_{MFe\%}} \cdot \left(\underline{a^t_{MFe\%}}\right) \right)\end{aligned} \tag{14}$$

where:

- v^t_{MNi}, $v^t_{\text{MSiO}_2}$, v^t_{MMgO}, v^t_{MFe} and v^t_{MT} are the nickel, silica, magnesia, iron and total tonnages sent to the processing plant in period t. Recall from Section 2.1.3 that these values are calculated by algorithmically evaluating the supply chain by sending the blocks (or clusters) from the mine through the blending or processing paths.
- v^t_{MV} is the economic value of the nickel recovered from the blended feed of material at the processing plant in period t. Using a nickel price, p_{Ni}, a recovery, r_{Ni} and a processing cost, c_{proc}, all of which are not disclosed for confidentiality purposes, the economic value of the recovered nickel can be calculated as follows:
$$v^t_{\text{MV}} = v^t_{\text{MNi}} \times p_{Ni} \times r_{Ni} - v^t_{\text{MT}} \times c_{proc} \tag{15}$$
- $\overline{a^t_{MC}}$ and $\underline{a^t_{MC}}$ are deviations from the upper and lower plant capacity targets, and are calculated as follows:
$$\overline{a^t_{MC}} = \max\left\{0, v^t_{MT} - T^H_{\max}\right\} \tag{16}$$
$$\underline{a^t_{MC}} = \max\left\{0, T^H_{\max} - v^t_{MT}\right\} \tag{17}$$
- $\overline{a^t_{MSiMg}}$ and $\underline{a^t_{MSiMg}}$ are deviations from the upper and lower silica-to-magnesia ratio constraints, respectively, and are calculated using the following non-linear constraints for the blended feed at the plant:
$$\overline{a^t_{MSiMg}} = \max\left\{0, \frac{v^t_{MSiO_2}}{v^t_{MMgO}} - 1.8\right\} \tag{18}$$
$$\underline{a^t_{MSiMg}} = \max\left\{0, 1.5 - \frac{v^t_{MSiO_2}}{v^t_{MMgO}}\right\} \tag{19}$$
- $\overline{a^t_{MFe\%}}$ and $\underline{a^t_{MFe\%}}$ are deviations from the upper and lower iron grade constraints, and are calculated using the following non-linear constraints:

$$\overline{a^t_{MFe\%}} = \max\left\{0, \frac{v^t_{MFe}}{v^t_{MT}} - 16\right\} \tag{20}$$

$$\underline{a^t_{MFe\%}} = \max\left\{0, 12 - \frac{v^t_{MFe}}{v^t_{MT}}\right\} \tag{21}$$

- $v^t_{ST_s}$ is the value of the stockpile tonnage for stockpile s and period t.
- $\overline{a^t_{SC_s}}$ is the deviation from the total capacity for stockpile $s \in \{1, \ldots, 6\}$ in period t and is calculated as follows:
$$\overline{a^t_{SC_s}} = \max\left\{0, v^t_{ST_s} - T^{SP}_{\max}\right\} \tag{22}$$
- The penalty costs associated with deviations from the silica-to-magnesia ratio, $\overline{d^t_{MSiMg}}$ and $\underline{d^t_{MSiMg}}$, are both set to 30,000,000 to heavily penalize deviations that could disrupt the plant productivity.
- The penalty costs associated with stockpile capacity deviations, $\overline{d^t_{SC_s}}$, are set to 1 for the first two low-grade stockpiles, 1.125 for the third low-grade stockpile, 1.5 for the first two high-grade stockpiles and finally 1.625 for the third high-grade stockpile. It was found through experimentation that the PSO algorithm tends to put more material into the last stockpile and has difficulties moving backwards to lower stockpiles. The higher value helps the heuristics move material back to the other stockpiles from the final stockpile of the set.
- The penalty costs associated with plant capacity deviations, $\overline{d^t_{MC}}$ and $\underline{d^t_{MC}}$, are set to 2.
- The penalty costs associated with deviations from the upper and lower iron grade targets, $\overline{d^t_{MFe\%}}$ and $\underline{d^t_{MFe\%}}$, are set to 20,000,000 to heavily penalize deviations that could disrupt the plant productivity.

3.2.2 Stochastic model

The stochastic model for the Onça deposit is similar to the deterministic model, however it evaluates the effect that each simulation has on the supply chain and penalizes deviations for each of the geological simulations, effectively leading to more robust destination decisions. The model is as follows:

Objective:

$$\begin{aligned}\max \sum_{t=1}^{44}\left(\frac{E\left\{v^t_{\mathrm{MV}}\right\}}{(1.008)^t} - \sum_{g=1}^{20}\sum_{s=1}^{6}\overline{d^t_{SC_s}}\left(\overline{a^{t,g}_{SC_s}}\right) - \overline{d^t_{MC}}\left(\overline{a^{t,g}_{MC}}\right) - \underline{d^t_{MC}}\cdot\left(\underline{a^{t,g}_{MC}}\right)\right) \\ - \sum_{g=1}^{20}\sum_{t=1}^{44}\left(\overline{d^t_{MSiMg}}\left(\overline{a^{t,g}_{MSiMg}}\right) + \underline{d^t_{MSiMg}}\cdot\left(\underline{a^{t,g}_{MSiMg}}\right) + \overline{d^t_{MFe\%}}\left(\overline{a^{t,g}_{MFe\%}}\right) + \underline{d^t_{MFe\%}}\cdot\left(\underline{a^{t,g}_{MFe\%}}\right)\right)\end{aligned} \tag{23}$$

where:

1. $g \in \{1, \ldots, 20\}$ is used to denote the twenty geological simulations used to represent the Onça orebody.
2. The definition of all deviation variables and property values from the deterministic formulation have been modified by adding an additional superscript g to represent the value of the deviation or property value in simulation g.
3. $E\left\{v^t_{\mathrm{MV}}\right\}$ is used to denote the expected value of the recovered nickel metal of the blended material at the plant, i.e. $E\left\{v^t_{\mathrm{MV}}\right\} = \frac{1}{20}\sum_{g=1}^{20} v^{t,g}_{\mathrm{MV}}$ and $v^{t,g}_{\mathrm{MV}}$ is the value of the recovered nickel metal at the plant in time t and simulation g.
4. All penalty costs remain the same as the deterministic model.

3.3 Numerical Results

3.3.1 Deterministic optimization

The model and mathematical formulation presented in the previous section are tested on a deterministic (kriged) orebody model given by the mine. Given that mine supply chain optimization is a new area of

research, an analysis of the performance of the proposed methodology on a deterministic orebody model demonstrates the method's effectiveness under more simplistic and conventional circumstances.

Two cases using deterministic orebody models are tested: the first with block destinations, discussed in Section 2.1.3, and the second using clusters, discussed in Section 2.2.3. Theoretically, the solution generated by solving block destinations should outperform the clusters, given that the optimizer has more control over the decision-making, however, as will be demonstrated, this control comes with a substantial amount of additional computing time; it is often more practical to generate high-quality solutions quickly. Recall that in the case of block destinations, there are 69,877 block destination decisions (i.e. the number of blocks in the model); in the case of clusters, the number of decision variables is reduced to $n_{clusters} \times 44$, where 44 is the number of production periods considered. The block destination heuristic (method 3 in Section 2.2.6) is employed far less for the block destination problem because of the amount of time required to re-evaluate the problem; performing the heuristic on the block destination problem takes approximately 90 minutes to test 10% of the blocks, whereas the cluster destination takes approximately 5 minutes, and can therefore be done much more frequently.

Both problems are solved using 20 particles, and in the case of the clustering problem, 50 clusters each are used for low-grade and high-grade saprolite. This results in reducing the block destination problem from 69,788 decision variables to 4,532 variables (waste, limonite and saprolite waste are each assigned to a different cluster). Through trial and error, the particle's inertia is set to 0.6, the particle's best inertia is set to 0.25 and the global best inertia is set to 0.8 (Eq. 12). The block destination problem ran for 9 hours and 55 minutes and performed 750 iterations, whereas the cluster problem ran for 7 hours and 43 minutes and performed 1250 iterations.

Figure 2 shows they key indicators of the performance of the algorithm. It can be seen that the problem solved using clusters always meets the minimum silica-to-magnesia ratio (1.5) and never exceeds the upper limit of 1.8. The solution using blocks, however, only meets the minimum ratio requirement on a few occasions, meaning that the solution is infeasible and would likely shut down the plant if used in practice. The plant feed capacities (hence homogenization pile tonnages) are never exceeded in the clustering solution and on some occasions are not able to meet the targets. This solution is substantially better than the block solution, which often drastically exceeds the plant feed's capacity. In both cases, the minimum and maximum bounds on the plant feed's iron grade is always satisfied. Figure 2 also shows the cumulative net present value (NPV) of the solutions, expressed as a percentage of the final cumulative NPV for the deterministic block problem; this convention will be adopted herein to express economic values. While the NPV for the block destination solution may be quite a bit higher than the cluster solution, the fact that the solution far exceeds the silica-to-magnesia ratios implies that the solution is not realistic. Figure 3 shows a comparison between the deterministic block and cluster destination problems for the tonnages at each of the six stockpiles over time. In general, it can be seen that the low-grade stockpiles are often not used to capacity (with the exception of the third stockpile). The high-grade stockpiles, however, are much more used and the capacities are often exceeded. Given the under-utilization of the low-grade stockpiles, it is assumed that some of the peaks in stockpile tonnages can be smoothed out by temporarily placing the material in other stockpiles, which could be confirmed by re-optimizing the problem, but is not considered in this case study.

It should be once again mentioned that the problems should generate similar solutions, however, the amount of time required to solve the block destination problem is impractical for this case study. A large part of the reason why the block destination solution for the deterministic case is not obeying the key indicators shown in Figure 2 is due to the size of the solution vector. The particle swarm optimization algorithm has been documented to perform less effectively in cases with high-dimension solution vectors (Li et al., 2011; Yanbin, 2008). As a result of the higher dimension (more block decision variables), the greedy block destination heuristic used is less effective because it takes substantially longer to run. For reasons of practicality in terms of time and effectiveness, it can therefore be stated that solving the problem using cluster decisions rather than block decisions is much more appealing. Future work will focus on disaggregating, optimizing and re-aggregating the clusters to improve the quality of the solution and reduce some of the instabilities associated with the k-means clustering algorithm; however, as shown in Figure 2 and Figure 3, the clusters and optimizer do indeed perform extremely well for the deterministic case.

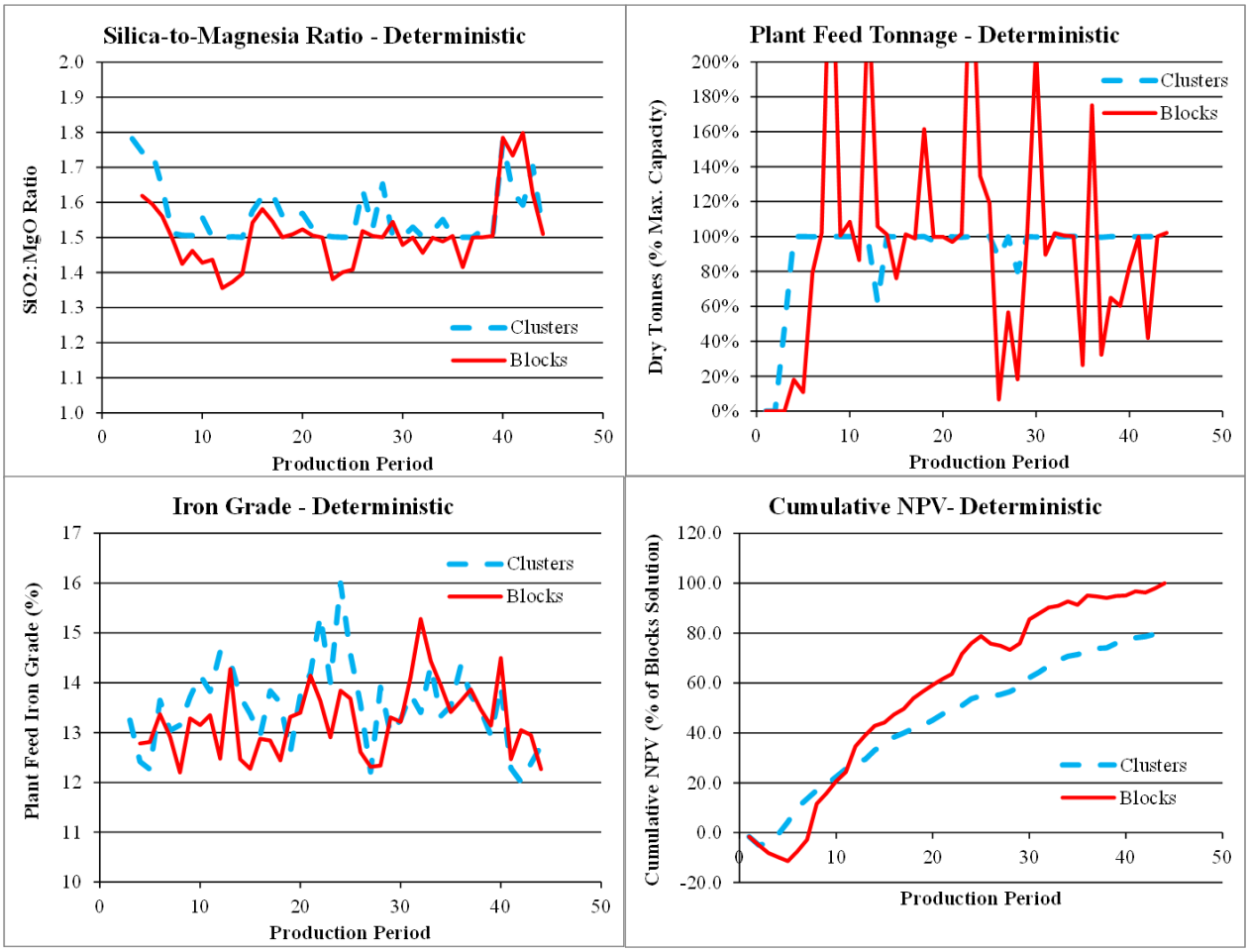

Figure 2: Graphs comparing the performance of the PSO algorithm when using block destinations or cluster destinations.

Figure 4 shows an analysis of the effect that the number of particles has on the average objective function value (Eq. 2); note that the objective function value itself is not disclosed, however the values are expressed relative to the first iteration with 5 particles. It can be seen that as more particles are used, the solutions tend to converge on an optimum in fewer iterations, however the final objective function values remain relatively unaffected, regardless of the number of particles used (within 4%). The solution times for the deterministic problem with clustering using 5, 10, 15 and 20 particles were 288, 302, 363 and 549 minutes, respectively. The tests for 5, 10 and 15 particles are close in time because the running time of the algorithm is dominated by the heuristics used. Due to the faster convergence (and despite the larger algorithm running time), all tests are performed using 20 particles, unless otherwise stated.

Figure 5 shows an analysis of the number of clusters used, and was generated using only 10 particles. Naturally, one would expect that as the resolution of the decisions decreases (i.e. aggregating more block information into a single cluster), the quality of the resultant solution would rapidly deteriorate. It can be seen while this is true, creating too many clusters (e.g. 100) is excessive and does not yield a substantially better solution; recall that as more clusters are added, the dimension of the optimization problem also increases. The higher dimensions also comes at a cost of longer solution times, which were 113, 120, 122, 140, 302 and 2158 minutes for 5, 10, 15, 25, 50 and 100 clusters for the low- and high-grade saprolite material, respectively. It is noted that the solution generally does converge faster as more clusters are used, but the difference in value between 25, 50 and 100 clusters is not substantial, and therefore does not indicate any added value for the associated computation time. Figure 5 also demonstrates some of the instabilities that can occur when choosing different starting points for the k-means clustering algorithm; in particular, it is noted

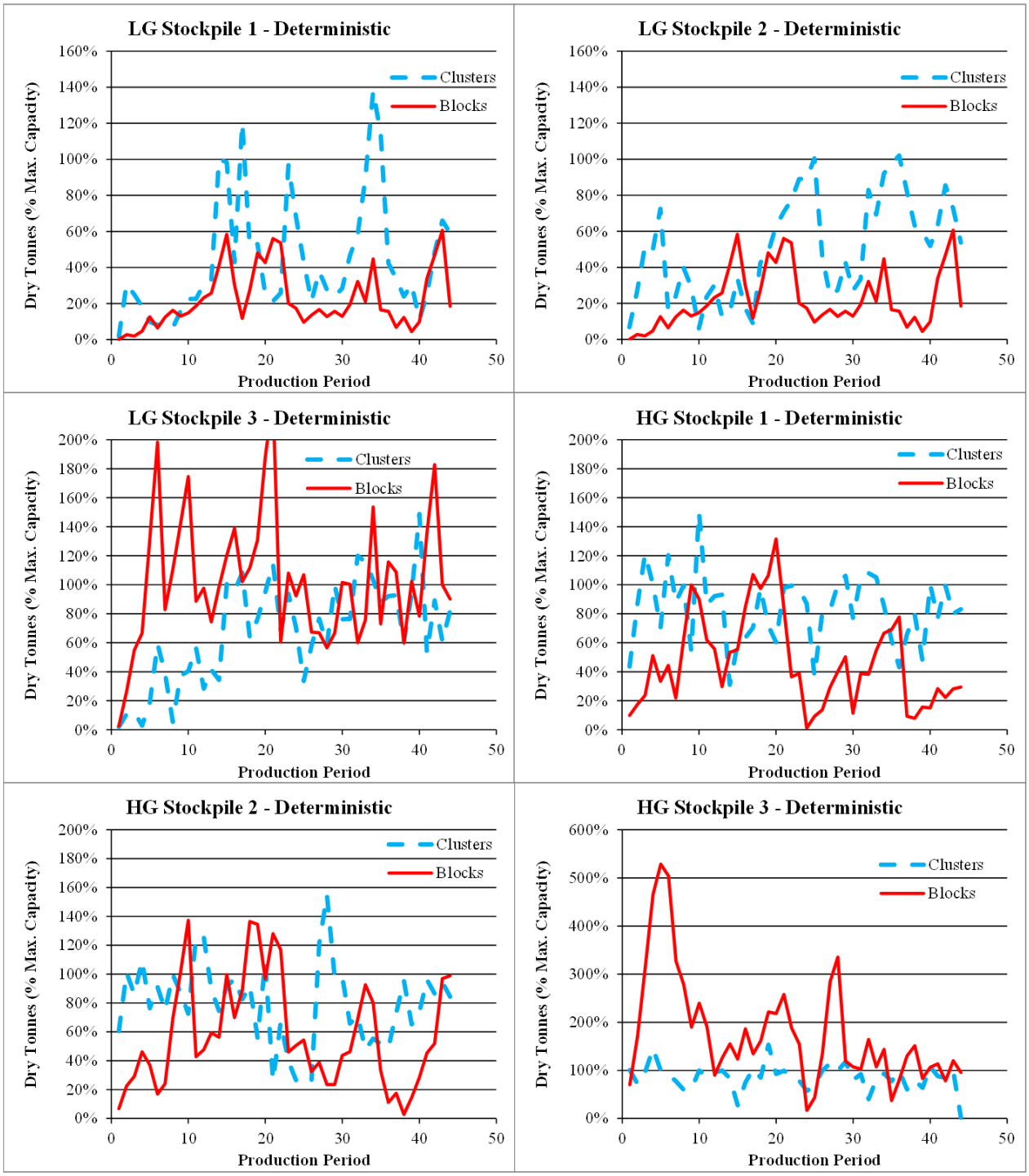

Figure 3: Comparison of tonnages going to the low-grade (LG) and high-grade (HG) stockpiles.

that the solution using 5 clusters is approximately 1.5% higher than the solution using 10 clusters. While these differences are essentially negligible, future work will investigate disaggregation and re-aggregation approaches during the algorithm's runtime to improve the stability of the clusters. For the purpose of this study, only 50 clusters for both the low- and high-grade saprolite will be used, as it balances solution quality and computation time.

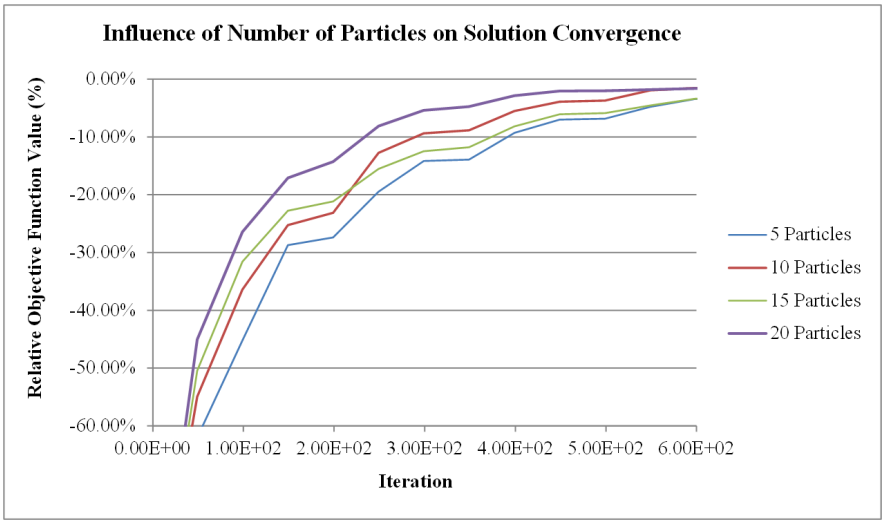

Figure 4: Analysis of the influence on number of particles on the objective function over the course of the algorithm.

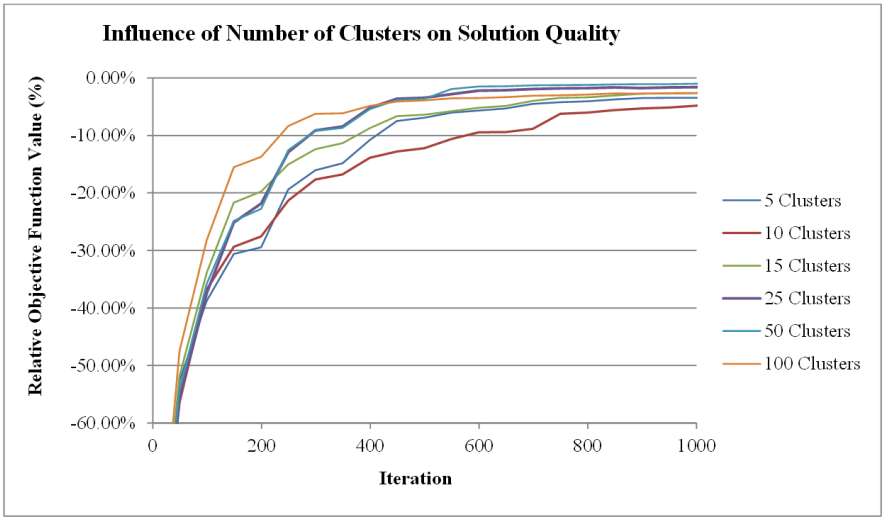

Figure 5: Analysis on the quality of the solution with respect to the number of clusters used.

3.3.2 Risk Analysis of deterministic destination policies

As mentioned in Section 2.2.4, moving from block destination decisions to cluster destination decisions permits the optimization of the supply chain (in this case, blending streams) under geological uncertainty. By comparing the location of the clusters (metal content, tonnages, etc.) to each block in each simulation, it is possible to obtain the cluster membership of the block in each simulation. As a result, each cluster may have varying values for the properties (i.e. nickel, silica, magnesia, iron and total tonnages) across different simulations, but the decision-making can be done on a cluster-basis, hence tie decision making together across all simulations. Consider the previous deterministic case for the estimated model. Each cluster contains a set of "coordinates" for nickel, iron, silica, magnesia and tonnage and an assigned destination from optimization. For any given simulation and block, the cluster that closest matches the block's "coordinates" will be sent to the cluster's assigned destination. The clusters then become a more advanced policy for material destinations and can take into account much more information than cut-off grades or values and are more suitable for blending operations.

To highlight the risk in the cluster destination decisions from the deterministic case study from the previous section, the original cluster locations and destinations are kept. The PSO algorithm is used only to optimize the processing stream options across the various simulations, however the cluster destination decisions remain static for all simulations. Figure 6 shows a risk analysis for the key parameters for the cluster destinations obtained from solving the deterministic problem with kriged orebody model. It can be seen that when using the destination policies from the conventional model, there is an extremely low chance that the minimum silica-to-magnesia ratios will be met, which would likely stop the plant entirely. This is a result in the difference between the distributions between a kriged model and a simulated model; it is known that the distributions for simulations can be quite different than the estimated model; hence, the clusters locations for the estimated model do not appear to be adequate for the simulations. The plant tonnage also appears to be extremely variable, however this is a common occurrence when analyzing solutions with simulations for this deposit and is largely based on the local distribution of lithologies within the simulation; it is possible that these fluctuations can be reduced through stochastic production scheduling. For the most part, the iron grades obey the minimum and maximum constraints. The cumulative NPV for the simulations, however, are substantially different from the deterministic model using clusters in Figure 2 and show a net loss (note that the NPV has been scaled by the maximum cumulative NPV from the deterministic model using block decisions). Again, these substantial differences are attributed to the large differences in distributions between the simulations and the estimated model.

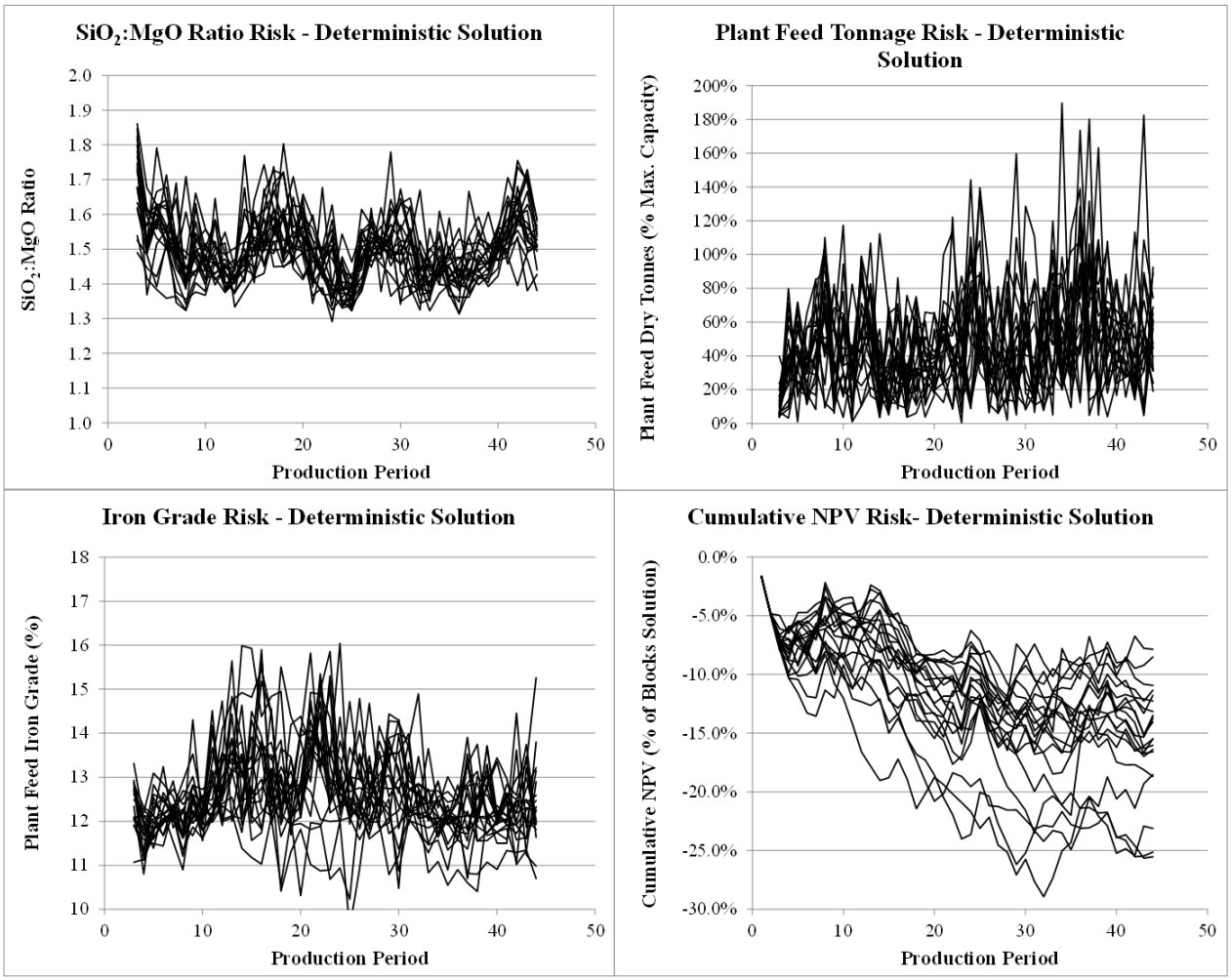

Figure 6: Risk analysis for the cluster decisions made for the deterministic orebody model.

3.3.3 Stochastic optimization

Similar to the deterministic model, each of the simulations can be optimized independently from the others to give an upper bound on the optimal solution using the deterministic model presented in the previous section; in stochastic optimization literature, this is referred to as the wait-and-see (WS) solution (Birge and Louveaux, 1997). This is similar in concept to mine planning where each simulation can be used to generate an independent production schedule to obtain the Expected Value of Perfect Information (EVPI), which can be useful to understand how much value there is in knowing the geology deposit perfectly. This is particularly important for the proposed methodology because, given that it involves metaheuristics and non-linear optimization, a theoretical bound on the solution is not easily obtained. Figure 7 shows each simulation's profiles when solved independently (including independent clusters). Unlike the results shown in Figure 6 (risk in the deterministic cluster solution), the results indicate that it is indeed possible to meet the limits silica-to-magnesia ratios, with only a few deviations. Additionally, it can be seen in Figure 7 that the plant feed capacities are rarely exceeded, despite the fact that there is substantial risk in being able to supply enough material; again, this is related to the variability of the lithologies of the simulations and is not related to the algorithm's abilities. The minimum and maximum bounds on the iron grade are also obeyed during the life of the schedule. Figure 7 also indicates that it is possible to obtain a positive net present value for the given production schedule if the geology is known perfectly. This highlights the fact that the clusters and optimal clustering destination decisions made for the deterministic model are inadequate

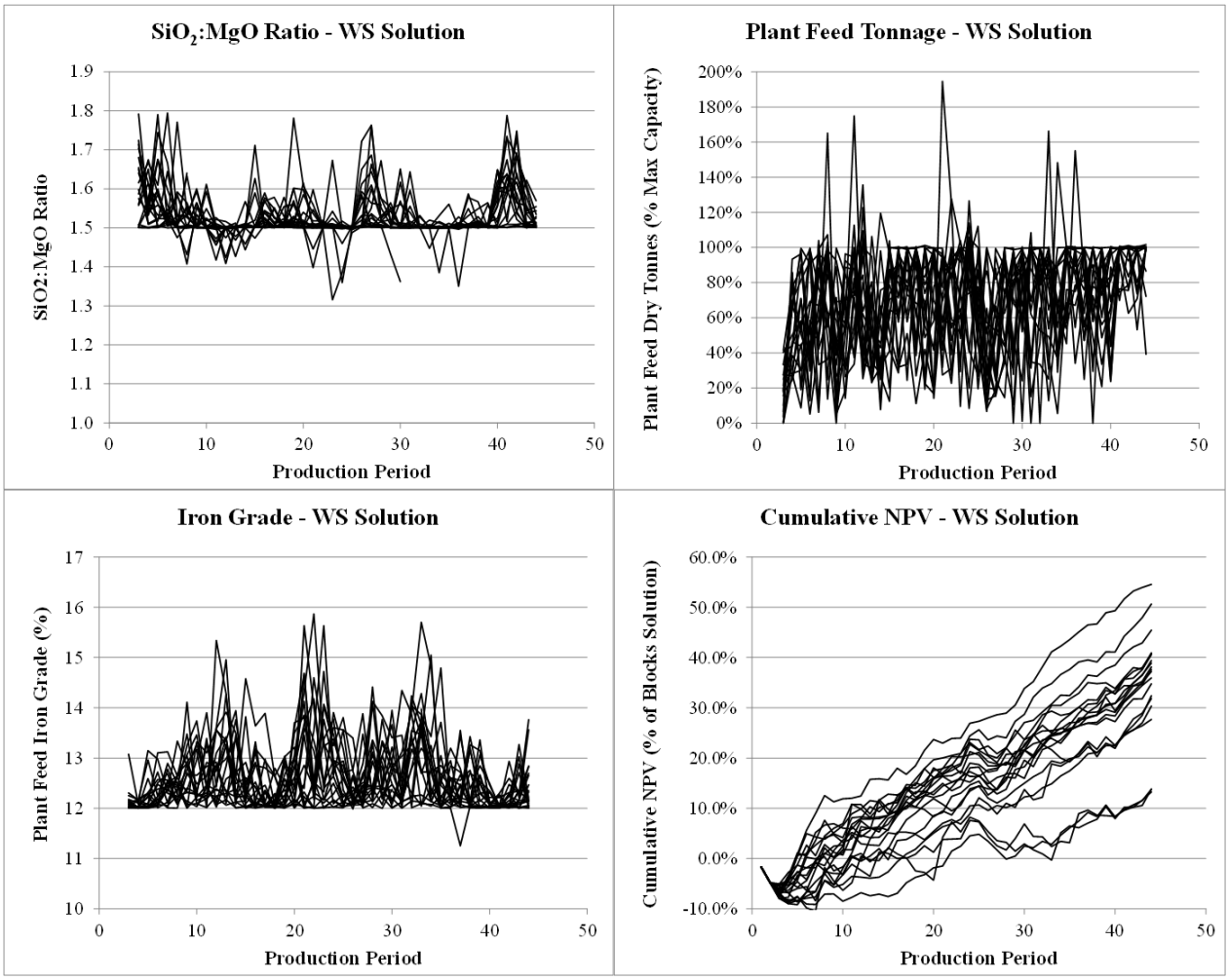

Figure 7: Risk analysis for the cluster decisions when each simulation's destinations are optimized individually (perfect knowledge).

when considering uncertainty, and that it may be possible to generate a more robust destination policy that considers geological uncertainty.

Using the stochastic optimization model outlined in Section 3.2.2, it is possible to link the geological simulations together in the optimization process through clusters and the cluster destination decision variables, which can in turn be used as a robust destination policy, somewhat akin to an advanced cut-off grade policy under geological uncertainty. The solution time for optimizing with this method is substantially longer for the stochastic case than the deterministic case; to solve the formulation with 50 clusters for the low- and high-grade material and 20 particles, it takes over 36 hours to perform 1250 iterations, where there is no substantial change in objective function value. The reason for this drastic increase is that the optimizer needs to evaluate a supply chain for each simulation and each particle for every iteration of the algorithm. While the algorithm is already parallelized on the computer's CPUs, it may be possible to drastically increase the performance by evaluating the supply chain on a GPU, which is a topic of future research. It is also possible to increase the performance by decreasing the number of particles or clusters, however these have been kept consistent for benchmarking purposes.

Figure 8 shows the risk profiles for the stochastic destination solution. It can be seen that it is possible to make robust cluster destination decisions that meet both the plant feed's silica-to-magnesia ratio requirements and the plant's iron grade requirements with very minor deviations. When compared to the previous case in Figure 7, there are actually fewer deviations from the silica-to-magnesia ratio and iron grade. This is caused

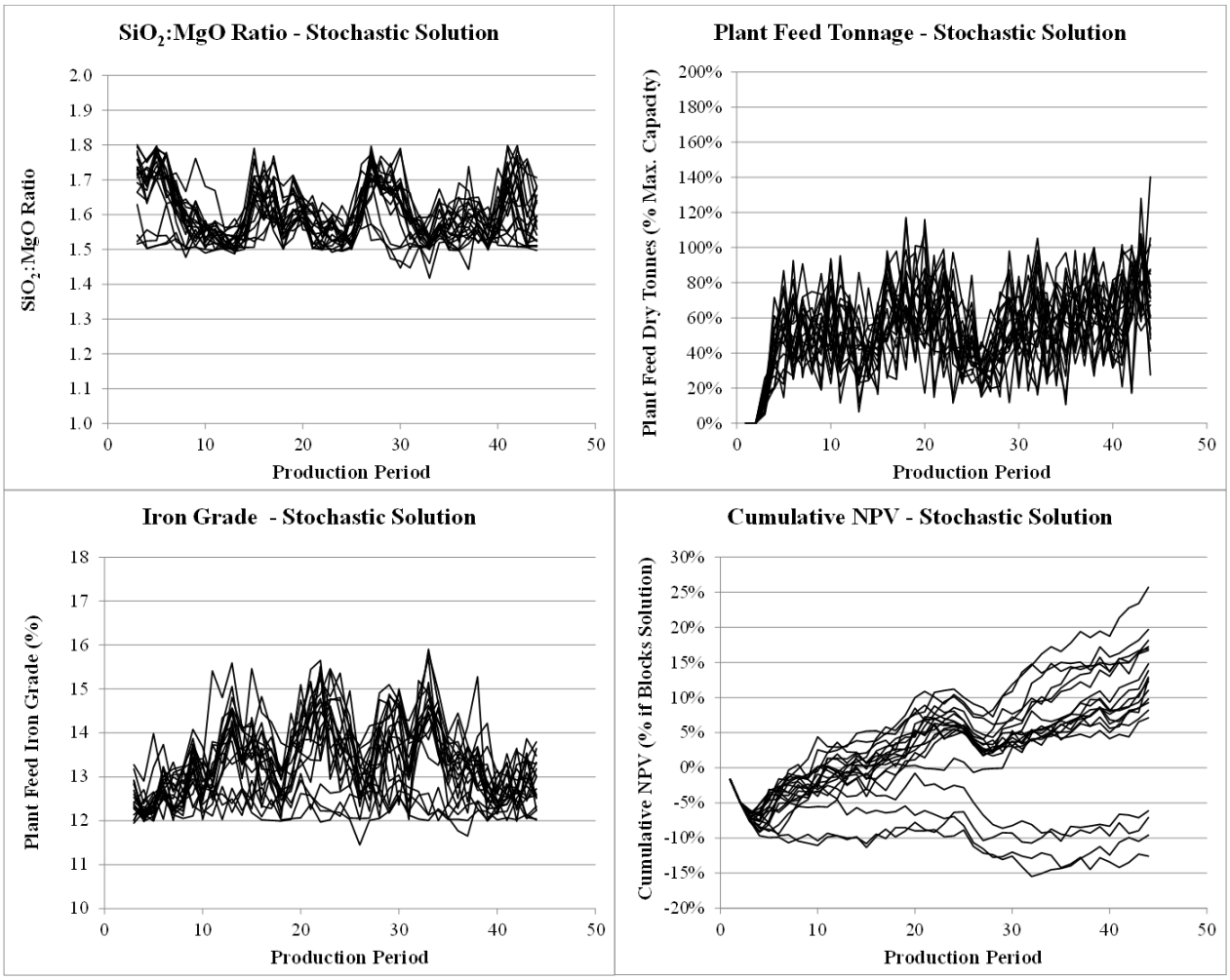

Figure 8: Risk analysis for the stochastic destination optimization solution where a common destination policy is applied to all simulations.

by the fact that the formulations are slightly different: the stochastic model considers the expected value of the blocks but continues to penalize each of the constraint deviations heavily whereas the previous case has more of a balance between net present value and deviations. The plant feed tonnages are often lower than the target of $T^H_{\max}$ tonnes, however the performance is not worse than the WS solution; the plant feed capacities are never exceeded, which indicates a high level of spatial variability in the saprolite layer between the simulations. The solution does generate a positive net present value for the majority of the simulations. The lower net present value for the stochastic solution when compared to the WS case is also normal given that the destination policies have been designed to be robust for the simulations, whereas the previous case will only generate solutions that are optimal for that specific simulation. While there appears to be a 20% chance that the schedule and destination policy will generate a negative net present value (loss), one must consider the high degree of variability between the simulations making it particularly challenging to generate a robust destination policy, and that these simulations also indicate a very low WS solution cumulative NPV in Figure 7.

4 Conclusions

This paper proposes a mining supply chain modelling methodology and is optimized using a particle swarm optimization algorithm with local heuristics. Given that the number of decision variables to direct material from a block to its initial destination can be orders of magnitude more than decision variables required to model the remainder of the supply chain, and that the bulk of the algorithm's time is spent evaluating these initial destination decisions, a method for aggregating blocks using the k-means clustering algorithm is proposed. These aggregations have been shown to substantially reduce the size of the problem, thus the computation time of the algorithm, with minimal loss in solution quality due to aggregation. It is possible to extend the concept of the aggregation into the context of supply chain optimization under geological uncertainty, whereby the aggregates are created based on the set of all simulations at once. This optimization problem can then be reformulated into a mixed integer nonlinear stochastic program, where the first stage decisions choose the optimal destinations for the clusters and the recourse decisions optimize the remainder of the supply chain. Using this method, the aggregates and optimal aggregate destinations can then be used as a destination policy that is robust under geological uncertainty.

The proposed method is tested at Vale's Onça Puma mine in Pará State, Brazil. Experimental results indicate that the proposed method works well for optimizing with complex blending operations with non-linear constraints. Additionally, the proposed method of stochastic optimization for the destination problem performs well and generates a single robust destination policy that adheres to the strict blending requirements at the processing plant. Future work will seek to improve cluster stability by disaggregating and re-aggregating the clusters as the algorithm progresses, test the proposed methodology on other mining supply chains and integrate production scheduling with the supply chain optimization under uncertainty in the context of stochastic global optimization.

References

Asad, M. W. A. (2010) Implementing a blending optimization model for short-range production planning of cement quarry operation. Journal of Mining Science, Vol. 46, No. 5, pp. 525–535.

Bienstock, D., Zuckerberg, M. (2010) Solving LP relaxations of large-scale precedence constrained problems. In proceedings, IPCO'10 Proceedings of the 14th International Conference on Integer Programming and Combinatorial Optimization, pp. 1–14.

Birge, J.R., Louveaux, F. (1997) Introduction to Stochastic Programming. Springer Series in Operations Research, Berlin.

Boland, N., Dumitrescu, I., Froyland, G., Gleixner, A. (2009) LP-based disaggregation approaches to solving the open pit mining production scheduling problem with block processing selectivity. Computers and Operations Research, Volume 36, Issue 4, pp. 1064–1089.

Boucher, A., Dimitrakopoulos, R. (2009) Block simulation of multiple correlated variables. Mathematical Geosciences, Vol. 41, No. 2, pp. 215–237.

Brownlee, J. (2011) Clever algorithms: nature-inspired programming recipes. Lulu Enterprises Inc., pp. 232–237.

Caccetta, L., Hill, S. P. (2003) An application of branch and cut to open pit mine production scheduling. Journal of Global Optimization, No. 26, pp. 349–365.

Chanda, E. K. (2007) Network linear programming optimization of an integrated mining and metallurgical complex. In proceedings, Orebody modelling and strategic mine planning: Uncertainty and risk management models, AusIMM Spectrum Series 14, 2nd Edition. pp. 149–155.

Cormen, T.H., Leiserson, C.E., Rivest, R.L. (2009) Introduction to Algorithms (3rd ed.). MIT Press, 1312 p.

Geman, S., Geman, D. (1984) Stochastic relaxation, Gibbs distribution and the Bayesian restoration of images. IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. PAMI-6, No. 6, pp. 721–741.

Gershon, M. (1983) Optimal mine production scheduling: evaluation of large scale mathematical programming approaches. Geotechnical and Geological Engineering, Vol. 1, No. 4, pp. 315–329.

Gleixner, A. M. (2008) Solving large-scale open pit mining production scheduling problems by integer programming. Master's thesis, Technische Universitt Berlin, Berlin, Germany, pp. 1–115.

Goodfellow, R., Albor Consuegra, F., Dimitrakopoulos, R., Lloyd, T. (2012) Quantifying multi-element and volumetric uncertainty, Coleman McCreedy deposit, Ontario, Canada. Computers & Geosciences, Vol. 42, pp. 71–78.

Hoerger, S., Bachmann, J., Criss, K., Shortridge, E. (1999[1]) Long term mine and process scheduling at Newmont's Nevada operations. In proceedings, 28th APCOM Symposium, pp. 739–748.

Hoerger, S., Hoffman, L., Seymour, F. (1999[2]) Mine planning at Newmont's Nevada Operations, Mining Engineering, Vol. 51, No. 10, pp. 26–30.

Kennedy, J., Eberhart, R. C. (1995) Particle swarm optimization. Proc. IEEE Int'l Conf. on Neural Networks IV, pp. 1942–1948.

Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P. (1983) Optimization by simulated annealing, Science, Vol. 220 (4598), pp. 671–680.

Lerchs, H., Grossmann, I. F. (1965) Optimum design of open-pit mines. Trans. CIM LXVII, pp. 47–54.

Li, J., Mao, J., Zhang, G. (2011) Multi-performance optimization of cement blending. In proceedings, 30th Chinese Control Conference, pp. 1877–1881.

Lloyd, S. P. (1982) Least squares quantization in PCM. IEEE Transactions on Information Theory, Vol. 28, No. 2, pp. 129–137.

Ramazan, S., Dimitrakopoulos, R. (2004) Traditional and new MIP models for production scheduling with in-situ grade variability. International Journal of Surface Mining, Reclamation and Environment, Vol. 18, No. 2, pp. 85–89.

Ramazan, S., Dimitrakopoulos, R. (2012) Production scheduling with uncertain supply: A new solution to the open pit mining problem. Optimization and Engineering, *in press*.

Sandeman, T., Fricke, C., Bodon, P., Stanford, C. (2010) Integrating optimization and simulation - a comparison of two case studies in mine planning. In proceedings, 2010 Winter Simulation Conference, pp. 1898–1910.

Sarker, R.A., Gunn, E. A. (1997) A simple SLP algorithm for solving a class of nonlinear programs. European Journal of Operational Research, Vol. 101, No. 1, pp. 140–154.

Stone, P., Froyland, G., Menabde, M., Law, B., Pasyar, R., Monkhouse, P. H. L. (2007) Blasor – blended iron ore mine planning optimization at Yandi, Western Australia. In proceedings, Orebody modelling and strategic mine planning: Uncertainty and risk management models, AusIMM Spectrum Series 14, 2nd Edition. pp. 133–136.

Strebelle, S. (2002) Conditional simulation of complex geological structures using multiple-point geostatistics. Mathematical Geology, Vol. 34, No. 1, pp. 1–21.

Wharton, C. (2007) The use of extracted blending optimization for improved profitability. In proceedings, Orebody modelling and strategic mine planning: Uncertainty and risk management models, AusIMM Spectrum Series 14, 2nd Edition. pp. 293–300.

Whittle, G. (2007) Global Asset Optimisation. In proceedings, Orebody modelling and strategic mine planning: Uncertainty and risk management models, AusIMM Spectrum Series 14, 2nd Edition. pp. 331-336.

Yanbin, C. (2008) Application of simulated annealing particle swarm optimization algorithm in power coal blending optimization. In proceedings, 4th International Conference on Wireless Communications, Networking and Mobile Computing, pp. 1–4.