



Long-term production scheduling of open pit mines using particle swarm and bat algorithms under grade uncertainty

by A. Khan

Synopsis

Long-term production scheduling for open pit mines is a large-scale, complex optimization problem involving large data-sets, multiple hard and soft constraints, and uncertainty in the input parameters. Uncertainty in the input parameters may be caused by different geological, economic, or technical factors. The uncertainty caused by geological factors, which is commonly termed geological or grade uncertainty, is considered to be most important source of uncertainty among these factors. It is caused by the fact that the block grade values are estimated using very sparse drill-hole sample data and the actual grade values will only be known once a block is drilled and blasted. Geostatistical conditional simulation techniques provide a framework to quantify this geological/grade uncertainty by generating multiple, equally probable simulated realizations of the orebody. Different stochastic programming models have been proposed in recent years to integrate this grade uncertainty into the optimization process, but solving these models for actual-sized open pit mines is usually extremely difficult and computationally expensive. In this paper, two different computationally efficient population-based metaheuristic techniques based on particle swarm optimization (PSO) and the bat algorithm are used to solve one particular stochastic variant of the open pit mine scheduling problem, *i.e.* the two-stage stochastic programming model with recourse for determining the long-term production schedule of an open pit mine under the condition of grade uncertainty.

Keywords

open pit, production scheduling, grade uncertainty, stochastic programming, metaheuristic technique.

Introduction

Long-term production scheduling is an important and integral part of the planning process for any open pit mine. It aims to define an extraction sequence of the mineralized material from the ground that produces the maximum possible discounted profit, *i.e.* NPV, while satisfying a set of physical and operational constraints. In the conventional approaches, the planning process usually starts with the construction of a geological block model that divides the orebody and the surrounding rock into three-dimensional arrays of regular, usually identical-sized, blocks. A set of attributes such as grade, specific gravity *etc.* is then assigned to each of these blocks, estimated using some form of spatial interpolation technique *e.g.* kriging, inverse distance method *etc.* and the drill-hole sample data. The blocks are then divided into two groups – waste and ore blocks. The blocks for which the prospective profit exceeds the

processing cost are categorized as ore, while the rest are the waste blocks. An economic value is then assigned to each of these individual blocks by taking into account the group to which they belong, their respective estimated grade or metal content, the recovery, and the economic parameters such as metal price, mining cost, and processing costs. The ultimate pit limits (UPLs) are then determined using this economic block model, followed by the more complex production scheduling problem to define the most profitable annual extraction sequence of the blocks lying inside the ULP while satisfying different physical and operational constraints.

A major drawback of this approach is its assumption that all the input parameters are known with certainty, while on the contrary a certain degree of uncertainty is almost always associated with these parameters, and ignoring this may result into unrealistic and erroneous scheduling decisions. The uncertainty in the input parameters may be caused by different economic and technical factors such as metal prices, currency exchange rates, mining and processing costs, and block grades. The grade-related uncertainty is considered to be the most important source of uncertainty for the scheduling process for open pit mines and is the main focus of study in this paper. The grade uncertainty stems from the fact that the grade values of the individual blocks are estimated using very limited drill-hole sample data, and usually a significant and variable level of uncertainty is associated with each of these estimated values. Geostatistical conditional simulation techniques provide a framework to generate multiple, equally probable realizations of the orebody's

* Department of Mining Engineering, Karakoram International University, Gilgit, Pakistan.

© The Southern African Institute of Mining and Metallurgy, 2018. ISSN 2225-6253. Paper received Sept. 2014; revised paper received Mar. 2018.



Long-term production scheduling of open pit mines using particle swarm and bat algorithms

attributes to quantify both the local variability and the potential uncertainty (Deutsch and Journel, 1997; Goovaerts, 1997). The availability of these techniques provides the opportunity to integrate the potential uncertainty into the scheduling process in order to generate a better production schedule in terms of achievable maximum NPV while minimizing the risk of deviations from production target. Different stochastic programming models for integrating grade uncertainty into the scheduling process have been proposed in the literature to produce better production schedules (Ramazan and Dimitrakopoulos, 2004; Golamnejad *et al.*, 2006; Menabde *et al.*, 2007). However, depending on the size of the individual blocks and of the ore deposit, the UPL may contain thousands to millions of blocks that may have to be scheduled over a time horizon typically ranging from 5 to 30 years while considering different physical and operational constraints and the potential uncertainty in the input data. This entails scheduling a large combinatorial optimization problem that may be extremely difficult and computationally expensive to solve using conventional techniques such as the branch and bound algorithm. In recent years a new class of computationally less expensive algorithms (metaheuristic techniques) for solving the mine design and production scheduling problem has attracted the attention of researchers, such as genetic algorithms (Denby and Schofield, 1994, 1996), simulated annealing (Kumral and Dowd, 2002, 2005), and ant colony optimization (Sattarvand and Niemann-Delius, 2009, 2013). These techniques do not guarantee the optimality of the final solution that they produce, but they can produce better quality solutions with relatively less computational cost.

In this paper an extension of the framework previously presented by authors (Khan and Niemann-Delius, 2014) for long-term production scheduling for open pit mines is presented. The extension was made to integrate grade uncertainty in the optimization process. Two different metaheuristic techniques based on the particle swarm algorithm and the bat algorithm are proposed to solve a particular variant of the abovementioned production scheduling problem, *i.e.* two-stage stochastic programming model with recourse (Birge and Louveaux, 2011).

The paper is organized as follows. A two-stage stochastic linear programming model with recourse of the production scheduling problem of the open pit mines is presented, followed by a discussion of the standard PSO and bat algorithm. A heuristic procedure for producing the initial population of feasible random solutions is then presented, with the proposed procedure for applying the PSO and bat algorithm to the open pit production scheduling problem. A case study is also presented.

Problem formulation

The two-stage stochastic (mixed integer) model with recourse formulation of the open pit mine scheduling problem, similar to the one proposed in Ramazan and Dimitrakopoulos (2004) can be represented as follows:

$$\text{Max} \sum_{t=1}^T \left[\sum_{i=1}^N E\{(NPV)_i^t\} x_i^t - \frac{1}{s} \sum_{s=1}^S (p_t^{o-} y_{ts}^{o-} + p_t^{o+} y_{ts}^{o+} + p_t^{m-} y_{ts}^{m-} + p_t^{m+} y_{ts}^{m+}) \right] \quad [1]$$

The objective function is composed of two parts. The first part is for the maximization of the expected NPV of the production schedule, and second part represents the minimization of the expected recourse costs whenever the stochastic constraints (Equations [6]–[9]) are violated.

Subject to:

Reserve constraints: A block cannot be mined more than once

$$\sum_{i=1}^T x_{it} \leq 1 \quad \forall i = 1, 2, \dots, N \quad [2]$$

Slope constraints: Each block can only be mined if its predecessors are already mined in or before period t .

$$x_{it} - \sum_{j=1}^t x_{jt} \leq 0 \quad [3]$$

$\forall i = 1, 2, \dots, N$; $t = 1, 2, \dots, T$; where $j \in$ (set of predecessors blocks of block i).

Mining capacity: The total material mined during each period should be within the predefined upper and lower limits.

$$\sum_{i=1}^T w_i * x_{it} \geq \underline{W}_t \quad t = 1, 2, \dots, T \quad [4]$$

$$\sum_{i=1}^T w_i * x_{it} \leq \overline{W}_t \quad t = 1, 2, \dots, T \quad [5]$$

Stochastic constraints: The upper and lower processing capacity constraints (Equations [6] and [7]) and the upper and lower metal production constraints Equations [8] and [9] represents the scenario-dependent stochastic constraints. These constraints are modelled as soft constraints where the violation is penalized in the objective function.

$$\sum_{i=1}^T o_{is} * x_{it} + y_{ts}^{o-} \geq \underline{O}_t \quad t = 1, 2, \dots, T, \quad s = 1, 2, \dots, S \quad [6]$$

$$\sum_{i=1}^T o_{is} * x_{it} - y_{ts}^{o+} \leq \overline{O}_t \quad t = 1, 2, \dots, T, \quad s = 1, 2, \dots, S \quad [7]$$

$$\sum_{i=1}^T m_{is} * x_{it} + y_{ts}^{m-} \geq \underline{M}_t \quad t = 1, 2, \dots, T, \quad s = 1, 2, \dots, S \quad [8]$$

$$\sum_{i=1}^T m_{is} * x_{it} - y_{ts}^{m+} \leq \overline{M}_t \quad t = 1, 2, \dots, T, \quad s = 1, 2, \dots, S \quad [9]$$

$$x_{it} \in (0, 1) \\ y_{ts}^{o-}, y_{ts}^{o+}, y_{ts}^{m-}, y_{ts}^{m+} \geq 0$$

where

T : Total number of periods

t : Time period index, $t = 1, 2, \dots, T$

N : Total number of blocks

i : Block index, $i = 1, 2, \dots, N$

S : Total number of simulations of the orebody

s : Simulation index $s = 1, 2, \dots, S$

$E\{(NPV)_i^t\}$ is the undiscounted expected economic value of block i and is calculated as follows:

$$E\{(NPV)_i^t\} = \sum_{s=1}^S BV_{is}^0 / S \quad [10]$$

where BV_{is}^0 is the undiscounted economic value of block i according to simulation s .

Long-term production scheduling of open pit mines using particle swarm and bat algorithms

$E\{(NPV)_i^t\}$ is the expected discounted economic value of block i if mined in period t and is calculated as follows:

$$E\{(NPV)_i^t\} = E\{(NPV)_i^0\} / (1 + dis1)^t \quad [11]$$

x_{it} are the first-stage scenario-independent binary decision variables, which take the value unity if block i is mined in period t , and zero otherwise.

p_t^{o-}, p_t^{o+} are the discounted unit costs for shortage or surplus ore produced in period t respectively, and are calculated as:

$$p_t^{o-} = \frac{p_0^{o-}}{(1 + dis2)^t}, \quad p_t^{o+} = \frac{p_0^{o+}}{(1 + dis2)^t} \quad [12]$$

p_t^{m-}, p_t^{m+} are the discounted unit costs for shortage or surplus metal produced in period t respectively and are calculated as:

$$p_t^{m-} = \frac{p_0^{m-}}{(1 + dis2)^t}, \quad p_t^{m+} = \frac{p_0^{m+}}{(1 + dis2)^t} \quad [13]$$

y_{is}^{o-}, y_{is}^{o+} are the second-stage scenario-dependent continuous variables representing the shortage or surplus amount of ore produced in period t if scenario s occurs, respectively.

y_{is}^{m-}, y_{is}^{m+} are the second-stage scenario-dependent continuous variables representing the shortage or surplus amount of metal produced in period t if scenario s occurs, respectively. The second-stage (recourse) decision variables are dependent on the outcome of the orebody realizations and of the first-stage decision variables *i.e.* x_{it} .

- w_i : Tonnage of block i
- o_{is} : Ore content of block i according to simulation s
- m_{is} : Metal content of block i under scenario s
- \underline{W}_t and \bar{W}_t : Lower and upper limits of the available mining capacity in period t , respectively
- \underline{Q}_t and \bar{Q}_t : Lower and upper limits of the available processing capacity in period t , respectively
- \underline{M}_t and \bar{M}_t : Lower and upper limits of the required metal production in period t , respectively.

Particle swarm and bat algorithms

Two different population-based metaheuristic optimization techniques are briefly presented. Both techniques use a population of solutions to explore the search space in order to find better solutions.

Particle swarm algorithm

Particle swarm optimization (PSO) is a population-based optimization technique first proposed by Kennedy and Eberhart in 1995 (Eberhart and Kennedy, 1995; Kennedy and Eberhart, 1995). The PSO algorithm mimics the swarming behavior of individuals living together in groups, *e.g.* fish schools, bird flocks, animal herds *etc.* The PSO algorithm performs the search process by using a population (swarm) of individuals (particles). Each individual (particle) is a potential solution to the optimization problem. A random starting position and random velocity are assigned to each particle of the swarm. The velocity and position of each particle is then updated during the iterative process using Equations [14] and [15] respectively to find the optimum or

near-optimum solution.

$$\vec{v}_{i,t} = \omega \vec{v}_{i,t-1} + c_1 \vec{r}_1 \cdot (\vec{p}_{i,t-1} - \vec{x}_{i,t-1}) + c_2 \vec{r}_2 \cdot (\vec{g}_{t-1} - \vec{x}_{i,t-1}) \quad [14]$$

$$\vec{x}_{i,t} = \vec{x}_{i,t-1} + \vec{v}_{i,t} \quad [15]$$

where

- $\vec{x}_{i,t}, \vec{x}_{i,t-1}$ represent the current and previous positions of particle i
- $\vec{v}_{i,t}, \vec{v}_{i,t-1}$ represent the current and previous velocities of particle i
- $\vec{p}_{i,t}$ is the personal best position experienced by particle i until iteration t
- $\vec{g}_{i,t}$ is the global best position experienced by the population of particles until iteration t
- ω is the inertia weight used to control the influence/contribution of the particle's previous velocity to the current velocity, with its value fixed to 0.7298
- \vec{r}_1, \vec{r}_1 represent uniform random real numbers in the range of (0, 1)
- c_1, c_2 are called the acceleration coefficients and are used to control the influence of cognitive and social terms on the particle's velocity, with the value commonly fixed to 1.49445 (Clerc and Kennedy, 2002).

Algorithm 1: Pseudo code of the PSO algorithm

- Initialize a population (swarm) of particles with random initial positions $\vec{x}_{i,0}$ and random velocities $\vec{v}_{i,0}$ in the search space.
- Initialize each particle's personal best position $\vec{p}_{i,0}$ to $\vec{x}_{i,0}$ its initial position
- Calculate the fitness value of each particle at its initial position $\vec{x}_{i,0}$ and determine the initial global best position \vec{g}_0
- **While** ($t <$ maximum number of generations)
- **For** all particles do
- Update the particle's velocity and position using Equations [14] and [15] respectively.
- Calculate the fitness value of each particle at its current position $\vec{x}_{i,t}$
- **If** fitness ($\vec{x}_{i,t}$) is better than the fitness ($\vec{p}_{i,t-1}$) then
- $\vec{p}_{i,t} = \vec{x}_{i,t}$
- **end if**
- **If** fitness ($\vec{p}_{i,t}$) is better than the fitness (\vec{g}_{t-1}) then
- $\vec{g}_t = \vec{p}_{i,t}$
- **end if**
- **end for**
- If the stopping criteria are met: **end while**

Bat algorithm

The bat algorithm (BA) is a stochastic population-based optimization technique first proposed by Yang (2010). It is based on the echolocation ability of bats. The micro-bats use their echolocation capabilities to find prey and avoid obstacles, even in complete darkness. The micro-bats emits short-duration (typically in the range of 8 to 10 ms) loud

Long-term production scheduling of open pit mines using particle swarm and bat algorithms

sound pulses with constant frequency in the region of 25 kHz to 150 kHz and listen for the echo that bounces back from the surrounding objects to find the food or avoid the obstacles. They usually emit 10 to 20 such sound pulses per second and can increase the pulse emission rate to about 200 pulses per second as they get close to their prey. To transform these unique properties of bats into an optimization algorithm, Yang idealized the following rules (Yang, 2010; Yilmaz, 2014) :

- ▶ All bats use their echolocation capabilities to find out their distance from a certain object and can differentiate between food/prey and background barrier in some way.
- ▶ Bats fly randomly with velocity v_i at position x_i with a frequency f_{min} , and can vary the wavelength λ and loudness A_0 of their emitted sound pulses to find the food. Depending on their distance from the prey, they can adjust the rate and wavelength or frequency of their emitted pulse.
- ▶ The loudness varies from a large positive value A_0 to a minimum constant value A_{min} .

Each bat's current position is a potential solution to the optimization problem. The bats use the following rules to update their positions:

$$f_i = f_{min} + (f_{max} - f_{min})\beta \quad [16]$$

$$v_i^{t+1} = v_i^t + (x_i^t - x_*) f_i \quad [17]$$

$$x_i^{t+1} = x_i^t + v_i^t \quad [18]$$

where f_i represents the current frequency of particle i , and f_{min} , f_{max} are the minimum and maximum allowable frequencies respectively. Initially a random frequency drawn from the uniform distribution in the interval $[f_{min}, f_{max}]$ is assigned to each bat. $\beta \in [0,1]$ is a random number drawn from the uniform distribution in the mentioned interval, x_* is the current global best position/solution, x_i^{t+1} and x_i^t are the current and previous positions of particle i , v_i^{t+1} and v_i^t are the current and previous velocities of particle i . For the local search part (steps 6 to 9), after selecting one solution among the current selected best solutions a new solution is generated using random walk.

$$x_{new} = x_{old} + \varepsilon \bar{A}^t \quad [19]$$

where $\varepsilon \in [-1,1]$ is a uniform random number and \bar{A}^t is the average loudness of all the bats in the current time step. The pulse emission rate and the loudness of the bats are updated as the iteration proceed using the following equations:

$$A_i^{t+1} = \alpha A_i^t \quad [20]$$

$$r_i^{t+1} = r_i^0 [1 - e^{(-\gamma t)}] \quad [21]$$

where A_i^{t+1} and A_i^t are the current and previous loudness of bat i , α and γ are constants, and r_i^{t+1} and r_i^0 are the current

and initial pulse rates. The values of A_i^t and r_i^0 can typically be $[1,2]$ and $[0,1]$ respectively. Yilmaz and Cengiz (2014) proposed a modified bat algorithm (MBA) to improve the poor exploration capabilities of the BA. They proposed to assign pulse emission rate r and loudness A to each dimension of the solution separately instead of assigning a single value to all the dimensions of a solution, and proposed the following procedure for updating the position along a certain dimension j of a solution:

$$x_{ij}^{t+1} = \begin{cases} x_{ij}^t + \varepsilon \bar{A}_j^t & \text{rand}_j > r_{ij} \\ x_{ij}^t & \text{Otherwise} \end{cases} \quad [22]$$

where \bar{A}_j^t represents average loudness of dimension j of all solutions at time t , and u indicates a solution selected among the best solutions. The loudness and pulse rate are updated as:

$$A_{ij}^{t+1} = \begin{cases} \alpha A_{ij}^t & \text{rand}_j > r_{ij} \\ A_{ij}^t & \text{Otherwise} \end{cases} \quad [23]$$

$$r_{ij}^{t+1} = \begin{cases} r_i^0 [1 - e^{(-\gamma t)}] & \text{rand}_j > r_{ij} \\ r_{ij}^t & \text{Otherwise} \end{cases} \quad [24]$$

In the MBA a candidate solution is included in the population if r and $< \bar{A}_i$, where \bar{A}_i is the average loudness of a solution i .

Algorithm 2: Pseudo code of the bat algorithm

- ▶ Initialize bat population with random initial positions $\bar{x}_{i,0}$ and random velocities $\bar{v}_{i,0}$ in the search space
- ▶ Initialize each bat's frequency f_i , pulse rate r_i , and the loudness A_i
- ▶ Calculate the fitness value of each particle at its initial position $\bar{x}_{i,0}$ and determine the initial global best position x_*
- ▶ **While** ($t <$ maximum number of generations)
- ▶ Generate a new solution by updating the frequency, velocity, and position using Equations [16], [17], and [18] respectively
- ▶ **If** ($\text{rand} > r_i$)
- ▶ Select a solution among the best solutions
- ▶ Generate a local solution around the selected best solution
- ▶ **end if**
- ▶ **If** (r and $< A_i$ and $f(x_i)$ is better than $f(x_*)$)
- ▶ Accept new solution
- ▶ Increase r_i , reduce A_i
- ▶ **end if**
- ▶ Rank the bats and find the current best x_*
- ▶ If the stopping criteria are met: **end while**

Initial solutions

A 'greedy' heuristic procedure has been used to generate the initial population of feasible solutions. The process starts by generating a list of the free blocks that are available to be mined in the current period as their predecessors are either already scheduled to be mined in the current or prior period or due to their position in block model they do not have one. A block is selected at random from the list and assigned to the current period and the list is updated again. The process

Long-term production scheduling of open pit mines using particle swarm and bat algorithms

continues for the current period until the mining and processing capacities are satisfied in the average sense before moving on to the next period. Mining and processing capacity constraints are handled as hard and soft constraints respectively during this process, but more preference was given to satisfy the processing capacity constraint. While selecting an ore block more preference was given to the blocks with highest probability of being ore by considering the set of simulated values for that block and the predefined cut-off grade. The process stops when either the free blocks list is empty or the number of periods is finished for the current solution. To further diversify the generated solutions, the annual mining capacity per period is chosen at random from the range between upper and average mining capacities allowed in a period.

Adaptation of the PSO and bat algorithm to the open pit mine production scheduling problem

To apply the continuous variant of PSO and bat algorithms to the long-term production scheduling problem, the problem has been turned into optimum depth determination problem. After generating a population of initial feasible random solutions using the heuristic procedure discussed earlier, the mining depth along each column for each period is determined by using a so-called encoding scheme. These depths are then updated using the PSO or bat procedure during the iterative process. After each iteration, each solution is back-transformed to the discrete space, but due to the one-dimensional nature of the PSO and bat algorithms the resulting solutions may be infeasible in terms of the required slope angles. A normalization procedure is then used to turn the infeasible solution into a feasible one in terms of the required slope angles. A constant penalty method has been used to deal with the violation of the capacity constraints, where a constant penalty is added to the objective function for the per ton violation of the mining capacity constraints, to decrease the quality of the infeasible solutions by using Equation [25].

$$\begin{aligned} \text{Max} \sum_{t=1}^T \left[\sum_{i=1}^N E\{(NPV)_i^t\} x_i^t \right. \\ \left. - \frac{1}{s} \sum_{s=1}^s (p_t^{o-} y_{ts}^{o-} + p_t^{o+} y_{ts}^{o+} + p_t^{m-} y_{ts}^{m-} + p_t^{m+} y_{ts}^{m+}) \right] \\ - p_t^{M-} R_t^{M-} + p_t^{M+} R_t^{M+} \end{aligned} \quad [25]$$

where p_t^{M-} , p_t^{M+} are the discounted unit costs for shortage or surplus rock (ore plus waste) produced in period t respectively, and are calculated as:

$$p_t^{M-} = \frac{p_0^{M-}}{(1 + dis2)^t}, \quad p_t^{M+} = \frac{p_0^{M+}}{(1 + dis2)^t} \quad [26]$$

R_t^{M-} and R_t^{M+} represent the shortage or excess amount of rock (ore plus waste) produced in period t . The value of the penalty is problem-dependent and is obtained for each problem by trial and error. A detailed description of the procedure can be found in Khan and Niemann-Delius (2014). The pseudo codes of applying the PSO and bat algorithms to the open pit mine scheduling problem using this proposed procedure are given in algorithms 3 and 4.

Algorithm 3: Pseudo code of the PSO algorithm for the production scheduling problem for open pit mines

- **Input:** block model, economic and technical parameters
- Initialize a population (swarm) of particles with random initial positions $\vec{x}_{i,0}$ and random velocities $\vec{v}_{i,0}$ in the search space ($i = 1, 2, \dots, N$)
- Initialize each particle's personal best position $\vec{p}_{1,0}$ to its initial position $\vec{x}_{i,0}$
- Calculate the fitness value of each particle at its initial position $\vec{x}_{i,0}$ and determine the initial global best position \vec{g}_0
- **While** ($t <$ maximum number of generations)
- Encode each particle's current position ($\vec{x}_{i,t}$), its personal best position ($\vec{p}_{i,t}$), and the populations' global best position (\vec{g}_t)
- **for** all particles do
- Update the particle's velocity and position using the equation of the PSO algorithm
- Back-transform
- Normalize the solution
- Calculate the fitness value of the particle at its current position $\vec{x}_{i,t}$
- **If** fitness ($\vec{x}_{i,t}$) is better than the fitness ($\vec{p}_{i,t-1}$) then
- $p_{i,t} = \vec{x}_{i,t}$
- **end if**
- **If** fitness ($\vec{p}_{i,t}$) is better than the fitness (\vec{g}_{t-1}) then
- $\vec{g}_t = \vec{p}_{i,t}$
- **end if**
- **end for**
- If the stopping criterion is met, end while.

Algorithm 4: Pseudo code of the bat algorithm for the production scheduling problem for open pit mines

- **Input:** block model, economic and technical parameters
- Initialize bat population with random initial positions $\vec{x}_{i,0}$ and random velocities $\vec{v}_{i,0}$ in the search space ($i = 1, 2, \dots, N$).
- Initialize each bat's frequency f_i , pulse rate r_i , and the loudness A_i
- Calculate the fitness value of each particle at its initial position $\vec{x}_{i,0}$ and determine the initial global best position x_*
- **While** ($t <$ maximum number of generations)
- Encode each particle's current position ($\vec{x}_{i,t}$) and the population's global best position (x_*)
- Generate a new solution by updating the frequency, velocity and position using Equations [16], [17], and [18] respectively
- **If** ($\text{rand} > r_i$)
- Select a solution among the best solutions
- Generate a local solution around the selected best solution
- **end if**
- Back-transform
- Normalize
- Calculate the fitness value of each particle at its current position $\vec{x}_{i,t}$
- **If** ($\text{rand} < A_i$ and $f(x_i)$ is better than $f(x_*)$)
- Accept new solution
- Increase r_i , reduce A_i
- **end if**
- Ranks the bats and find the current best x_*
- If the stopping criterion is met: **end while**.

Long-term production scheduling of open pit mines using particle swarm and bat algorithms

Case study

To check the capabilities and the efficiency of the proposed procedure for generating a long-term production schedule under the condition of grade uncertainty, a set of 15 simulated realizations of a copper deposit was used. Examples of these realizations are shown in Figure 1. These realizations were generated using sequential Gaussian simulation technique (SGS) (Deutsch and Journel, 1997).

A predefined fixed cut-off grade strategy has been used to determine the destination of a particular block. While using this fixed cut-off grade strategy a block may be categorized as an ore block according to some simulations, and a waste block according to others (Figure 2). This fact has been taken into account while determining the expected economic value of the blocks.

The planning and scheduling process for open pit mines traditionally starts with the determination of the ultimate pit limits (UPLs), which define the extent to which it is economically feasible to mine. The UPLs were determined by solving the following mathematical formulation:

$$\text{Maximize } \sum_{i=0}^N V_i x_i \quad [27]$$

Subject to:

$$\begin{aligned} x_i &\leq x_j, i = 1, 2, \dots, \dots, N; \quad j \in P_i \\ x_i &\in [0,1] \end{aligned} \quad [28]$$

where V_i represents the expected economic value of block i , N represents the total number of blocks in the block model, x_i represents a binary variable corresponding to block i , and P_i represents the predecessor group of block i . The required slope angles were assumed to be 45° in all directions. In the next step to define the optimum extraction sequence of the blocks lying inside the predetermined UPLs, the technical and economic parameters mentioned in Table I were used. The length of each scheduling period was assumed to be one year. The upper and lower limits for mining, processing, and metal capacity were set to be within $\pm 20\%$ and $\pm 10\%$ of the average available quantity of rock, ore, and metal available within the predefined UPL for each period of the scheduling horizon respectively.

All the numerical experiments were completed on an AMD Phenom II X4 945 (3.00 GHz and 4 GB RAM) running under Windows 7. The two-stage stochastic programming formulation of the open pit production scheduling problem (as mentioned in the section ‘Problem formulation’) with recourse was solved using the commercial solver CPLEX,

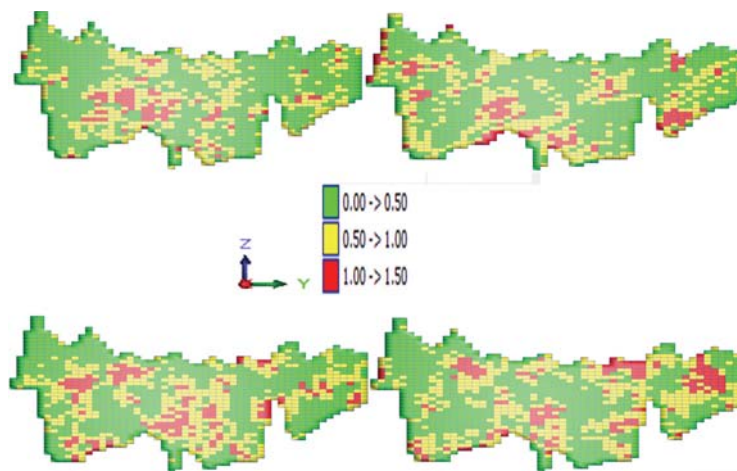


Figure 1—Cross-sectional view of four different simulated realizations of the copper deposit in the north-south direction

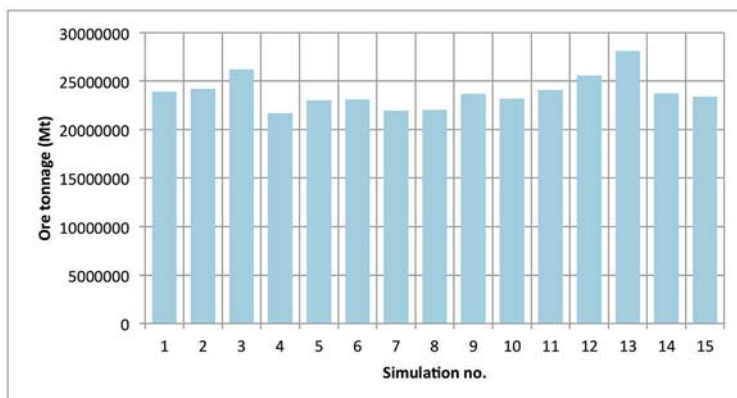


Figure 2—The ore content of the deposit according to different realizations of the orebody

Long-term production scheduling of open pit mines using particle swarm and bat algorithms

Table I

Technical and economic parameters used in the case study

Number of blocks	7107
Number of scenarios	15
Block dimensions (m)	25×25×12.5
Block tonnage (Mt)	6500
Metal price (\$/oz)	0.12
Mining cost (\$/Mt)	2
Processing cost (\$/Mt)	10
Refining or selling cost (\$/oz)	6.5E-3
Recovery	85%
Discount rate (dis1,dis2)	8%
No of periods (years)	5

Table II

General information about the solution found by CPLEX

Objective value (\$)	1.75818 E 008
CPU time (hours)	61.63
Optimality gap (%)	6

considering the blocks within the predefined UPL and the parameters mentioned in Table I. This solution will be used as a benchmark to assess the performance of the PSO and bat algorithms in terms of computational time and solution quality.

After conducting a series of experiments, a population of 50 solutions was found out to be working well for the problem under consideration. A set 50 solutions was generated using the greedy heuristic described in the section 'Initial solutions'. These same solutions were used during all the subsequent experiments. Due to the stochastic nature of the metaheuristic algorithms the problem was solved 50 times using different variants of the PSO and bat algorithms. The maximum number of iterations was used as the termination criterion and was fixed to 2000 after observing the convergence behaviour of both the algorithms. The average relative % gap (which will be used as a measure of quality of the solutions produced by the metaheuristic techniques under study) between best solutions generated by the different variants of the metaheuristic algorithms, *i.e.* Z_{approx} and the optimal solution found by CPLEX, *i.e.* Z_{CPLEX} are calculated using the following equation and are reported in Table II.

$$\% \text{ Gap} = \frac{Z_{CPLEX} - Z_{approx}}{Z_{approx}} * 100 \quad [29]$$

The values of the penalties used to report the violation of the capacity constraints, *i.e.* mining, processing, and metal capacity constraints in the objective function, were found to have a great effect on the performance of both the algorithms. When higher values were used, both the algorithms showed very few signs of improvement during the iterative process in comparison to the situations when smaller values were used. Therefore the following two sets of penalties have been used during this study.

- **Algorithmic penalties:** During the optimization process these penalties have been used for enabling the algorithms to efficiently explore the solution space and for generating the desired final solution with desired risk profile. These penalties are usually smaller in value than the corresponding actual penalties and are determined by trial and error.
- **Actual penalties:** These are the actual penalties that have been used to calculate the objective value of the final solution generated by the PSO or bat algorithm at the end of the iterative process.

For the problem under consideration, both variants of the PSO algorithm performed better than the variants of the bat algorithm in terms of the achieved average % gap as mentioned in Table III. The multi-start PSO algorithm showed the best performance in comparison to all its population-based competitors in the current study in terms of the achieved average % gap, showing the effect of diversification on the overall performance of the PSO algorithm. During the multi-start process the particle positions and velocities are re-initialized to their initial values after a fixed number of iteration as a diversification scheme. Both variants of the bat algorithm showed poor performance when the standard velocity equation, (Equation [17]) with maximum frequency value equal to unity was used to update the virtual bats' positions.

Therefore the following modified equation was used to update the velocity of a bat:

$$v_i^{t+1} = \omega v_i^t + (x_i^t - x_*) f_i \quad [30]$$

After conducting a series of experiments, the algorithm was found to be working well when the value of the maximum frequency was fixed to 0.6 and ω was chosen from the interval 0.4–0.6. The bat and MBA algorithms performed well when ω was fixed to 0.4, with a relatively small standard deviation showing the robustness of the technique in comparison to both variants of the PSO algorithm. In general both the metaheuristic techniques can produce better quality results with less computational cost in comparison to the exact optimization algorithm, thereby providing the opportunity of generating solutions with different risk profiles according to the needs of a certain open pit mining operation, which is quite difficult and expensive using an exact algorithm.

Conclusion

An extension of the framework previously formulated by the authors (Khan and Niemann-Delius, 2014) has been presented. This extended framework can account for grade uncertainty in the production scheduling process for open pit mines. To check the efficiency of the proposed extended framework for finding an optimum or near-optimum solution, two different population-based metaheuristic techniques, the PSO and bat algorithm, have been used. Through numerical experiments it was learnt that both variants of the PSO algorithm performed better than the variants of the bat algorithm in terms average % gap achieved, but showed poor

Table III

Numerical results of different variants of the PSO and bat algorithm

	% gap		CPU time (minutes)
	Mean	Standard deviation	
Global PSO	0.533	0.160	8.14
Multi-start PSO	0.528	0.135	8.42
Bat algorithm			
$\omega=0.6$	0.817	0.126	8.35
$\omega=0.4$	0.790	0.094	8.37
Modified bat algorithm			
$\omega=0.6$	0.832	0.107	8.57
$\omega=0.4$	0.808	0.097	8.43

performance in terms of average % standard deviation for the problem under study. The numerical experiments further revealed that the proposed procedure can generate sufficiently good solutions of the problem at hand in a shorter period of time. This property of the proposed framework provides the opportunity to handle grade-related uncertainties in more efficient way, which is otherwise quite difficult with currently available commercial software on the current hardware.

References

BIRGE, J.R. and LOUVEAUX, F. 2011. Introduction to Stochastic Programming. Springer Series in Operations Research and Financial Engineering.

CLERC, M. and KENNEDY, J. 2002. The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1. pp. 58–73.

DENBY, B. and SCHOFIELD, D. 1994. Open pit design and scheduling by use of genetic algorithms. *Transactions of the Institution of Mining and Metallurgy Section A: Mining Industry*, vol. 103. pp. A21–A26.

DENBY, B. and SCHOFIELD, D. 1996. Genetic algorithms for open pit scheduling-extension into 3-dimensions. *Proceedings of the Fifth International Symposium on Mine Planning and Equipment Selection (MPES)*. Sao Paulo, Brazil. pp. 177–185.

DEUTSCH, C.V. and JOURNAL, A.G. 1997. GSLIB: Geostatistical Software Library and User's Guide. Oxford University Press.

EBERHART, R. and KENNEDY, J. 1995. A new optimizer using particle swarm theory. *Proceedings of the Sixth International Symposium on Micro Machine and Human Science (MHS '95)*, Nagoya, Japan. IEEE, NewY ork. pp. 39–43.

GOLAMNEJAD, J. OSANLOO, M., and KARIMI, B. 2006. A chance-constrained programming approach for open pit long-term production scheduling in stochastic environments. *Journal of The South African Institute of Mining and Metallurgy*, vol. 106. pp. 105–114.

GOOVAERTS, P. 1997. Geostatistics for Natural Resources Evaluation. Oxford University Press.

KENNEDY, J. and EBERHART, R. 1995. Particle swarm optimization. *Proceedings of*

the IEEE International Conference on Neural Networks, 1995. IEEE, New York.

KHAN, A. and NIEMANN-DELIUS, C. 2014. Production scheduling of open pit mines using particle swarm optimization algorithm. *Advances in Operations Research*, December 2014. doi: 10.1155/2014/208502

KUMRAL, M. and DOWD, P.A. 2002. Short-term mine production scheduling for industrial minerals using multi-objective simulated annealing. *Proceedings of the 30th International Symposium on Application of Computers and Operations Research in the Mineral Industry (APCOM 2015)*, Fairbanks, Alaska. pp. 731–742.

KUMRAL, M. and DOWD, P.A. 2005. A simulated annealing approach to mine production scheduling. *Journal of the Operational Research Society*, vol. 56. pp. 922–930.

MENABDE, M., FROYLAND, G. STONE, P., and YEATES, G.A. 2007. Mining schedule optimization for conditionally simulated orebodies. *Proceedings of Orebody Modelling and Strategic Mine Planning: Uncertainty and Risk Management Models. Spectrum Series 14*, 2nd edn. Australasian Institute of Mining and Metallurgy, Melbourne. pp. 379–384.

RAMAZAN, S. and DIMITRAKOPOULOS, R. 2004. Stochastic optimization of long-term production scheduling for open pit mines with a new integer programming formulation. *Proceedings of Orebody Modelling And Strategic Mine Planning: Uncertainty and Risk Management Models. Spectrum Series 14*, 2nd edn. Australasian Institute of Mining and Metallurgy, Melbourne. pp. 385–392.

YILMAZ, S. and CENGİZ, E.U.K.Y. 2014. Modified bat algorithm. *Elektronika ir Elektrotechnika*, vol. 20, no 2. pp. 71–78.

SATTARVAND, J. and NIEMANN-DELIUS, C. 2009. Long term open pit planning by ant colony optimization Institut für Bergbaukunde III, RWTH Aachen University.

SATTARVAND, J. and NIEMANN-DELIUS, C. 2013. A new metaheuristic algorithm for long-term open-pit production planning [Nowy meta-heurystyczny algorytm wspomagaj cy długoterminowe planowanie produkcji w kopalni odkrywkowej]. *Archives of Mining Sciences*, vol. 58, no. 1. pp. 107–118.

YANG, X.-S. 2010. A new metaheuristic bat-inspired algorithm. *Proceedings of Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*. *Studies in Computational Intelligence Series*, vol. 284. Springer, Berlin/Heidelberg. pp. 65–74. ◆