Open-Cut Mine Planning via Closed-Loop Receding-Horizon Optimal Control

Cristian R. Rojas, Graham C. Goodwin, María M. Seron, and Meimei Zhang

The University *of* Newcastle, Australia 2308.

Summary. To obtain maximal return from a mining operation it is important that the sequence of mining steps be carefully planned. In this chapter we show how this problem can be converted into a closed-loop receding-horizon optimal control problem. Of particular interest is the formulation of the associated optimisation problem in the face of uncertainty, e.g., future ore prices. We show how one can formulate "open-loop", "reactive" and "closed-loop" policies to deal with price uncertainty. A 'toy' example is presented to give insight into the problem. Also, a realistic mine-planning exercise is briefly described to highlight discrepancies between theory and practice.

Key words: Closed-Loop Control, Mine Planning, Reactive Control, Receding Horizon, Scenario Generation, Stochastic Programming

1 Background

In this chapter we will be concerned with optimal open-cut mine planning. This a quintessential example of a complex optimisation problem in the face of uncertainty.

Considerable benefits can be obtained by careful planning of mining operations. Hence, there has been a substantial research effort devoted to the topic of optimal mine planning, see for example [1–3, 6–13, 32].

The essence of this problem is as follows: One has preliminary available data on the location of an ore-body in a particular geological volume. Given the data, one would like to know 'where' and 'when' to dig so as to optimise the 'net present value'. A typical mining operation can span 15 to 20 years or more and hence there is a temporal aspect to questions. Also, the optimisation needs to respect a host of constraints, e.g., mining capacity in each year and slope constraints on the mine walls to avoid collapses. Finally, there are a number of uncertainties, e.g., the location and extent of the ore body and, of course, one does not know the future price that the ore will have.

When formulating solutions to optimisation problems of the type described above, it is helpful to distinguish between 'open-loop', 'reactive' and 'closed-loop' solutions. An open-loop solution is the easiest to understand and solve. Here one simply finds a single ‘control’ sequence over the optimisation horizon. Here, and in the sequel, we use the term ‘control’ to describe those variables that can be selected (at each time step) by the optimiser. Thus, in the case of mining, we could define $u_{jk}(t)$ as the action of mining (or not) at location (x_j, y_k) at time t. Then, $\{u_{jk}(t)\}$ becomes the ‘control variable’.

‘Reactive’ strategies are closely related to open-loop strategies. The key difference is that, whereas in an open-loop strategy one applies the entire open-loop control sequence blindly, in a reactive strategy one applies only the current control. Then, when new information (i.e., the current ore price) arrives at the next time step, then one does another open-loop plan beginning from the current ‘state’. Again one implements only the first control step and so on.

‘Closed-loop’ strategies are quite different in the sense that, in these strategies, one takes account of the fact that *in the future* additional information will become available. Thus closed-loop policies represent a mapping between the ‘information state’ and the control. It is well known that for a very special class of problems (e.g., linear-quadratic optimal control problems) separation holds, i.e., the current control is not affected by the fact that, in the future, additional information will become available. This is a remarkable result. Unfortunately, separation does not hold in general. Thus, for general temporal closed-loop optimisation problems, such as the mine-planning case, one needs, in principle, to carry out a function optimisation, where the function maps the ‘information state’ to the control.

The obvious problem with closed-loop optimisation, as described above, is the need for function optimisation. Thus, in principle, one needs a different ‘control’ for every possible outcome of the uncertain variables. This is a formidable task. Indeed, even in the open-loop case, typical optimisation problems can be overwhelmingly complex. For example, in the mine-planning problem one typically has thousands of variables per time step. Also, if one has, say, 15 steps, then one can easily have 15^{1000} variables to consider. This number is way beyond comprehension and should be compared with 10^{80} which is the number of atoms in the known universe. This is the well known ‘curse of dimensionality’ in temporal optimisation.

Clearly one needs to make various simplifications to the problem if one is to have any hope of solving it in realistic time. Fortunately, by various ‘aggregation strategies’ one can often reduce the problem to an acceptable size. For example, in typical mining problems, various strategies based on spatial quantisation are used to simplify the problem. These strategies have resulted in open-loop optimal mine planning becoming a feasible idea and, indeed, it is frequently employed in practice.

One relatively recent idea, proposed by the current authors [17], is to also utilise *non-uniform* time quantisation to simplify the calculations needed in open-loop planning. The basic idea here is that, although the future cannot be ignored, it often has a diminishing impact on the best control to use *now*. Based on this observation, one might conjecture that one will not compromise performance too much if one quantises time more coarsely in the distant future. One then solves for the non-uniformly quantised ‘control’ sequence and applies (or stores) the first step. One then advances time by one step and repeats the procedure using a different set of (non-uniformly) quantised times. This ‘rolling-horizon’ approach allows one to build a strategy with a small uniform time step by combining a sequence of moving non-uniformly quantised problems. This idea has been adopted by industrial colleagues and used in realistic mine planning problems. Indeed, this has led to computational savings of the order of $10 : 1$. Consequently, one might believe that there is now scope to try more daring strategies, e.g., those based on some form of closed-loop planning.

The latter strategies have the potential to make the planning more 'flexible' so that the mine is better able to respond to future price variations.

An immediate stumbling block is that, as discussed above, one needs to essentially map every possible realisation of the uncertain variables into a different control sequence. Consequently, one is obviously limited to a small number of possible realisations. Indeed, for the mining problem, it seems that only a handful (say 10 to 100) realisations could be contemplated.

Actually, the above problem has been long recognised in the planning literature, and there exists a substantial volume of work on how one can choose a *small*, *representative* set of realisations of uncertain variables. This is often called the problem of 'scenario generation' and has been extensively discussed in the applied optimisation literature including finance, management and statistics [5, 24, 25, 31, 34]. The problem is actually equivalent to approximating a given continuous probability distribution by a discrete distribution having finite support of given cardinality. In turn, this is a special type of quantisation problem and it thus has links to contemporary literature in signal processing.

2 Experiment Description

2.1 Formulation of Mine-planning Problem

To set the work in the current chapter in context, we will next give an outline of the mine-planning problem. (Of course, real mine planning problems involve a host of other practical issues, e.g., processing capacity, stockpiling, etc., not covered here.) The basic idea of open cut mining can be visualised in Figure 1, which shows the ultimate pit of a typical mine, that is, the opening left in the ground after mining operations have been completed. (Actually this particular mine will be used as the basis of the discussion of discrepancies between theory and experiments in Section 4.) For simplicity of exposition, we represent the potential mine by the 'box' shown in Figure 2, where the 'surface' is divided into $(J + 1) \times (K + 1)$ rectangles.

2.2 State-space Model of the Mine

So that we can utilise control theory insights, we next cast the mine-planning problem in the form of a control problem. Accordingly, we define the mine state as the set of pit depths at the locations of the surface. We represent the evolution of this state via a linear, discrete-time dynamic model where mining action is the control input. Specifically, we denote by $x_{jk}(t)$ the mine depth at location jk at time t. Similarly, we denote by $u_{jk}(t)$ the action to mine (or not) at time t in the location jk, $j \in$

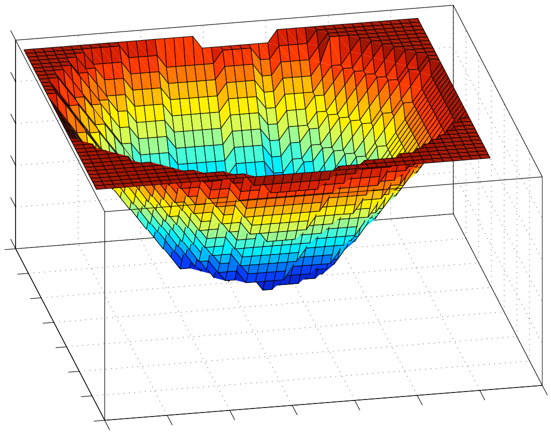


Fig. 1. Illustration of a typical mine ultimate pit.

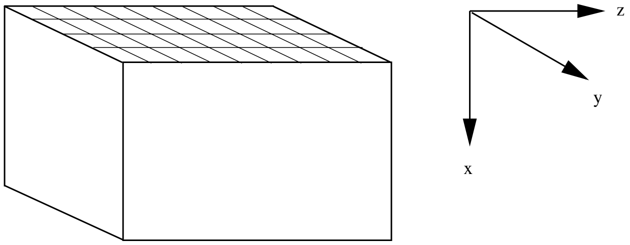


Fig. 2. Simplified representation of the mine.

$\{0, \ldots, J\}$, $k \in \{0, \ldots, K\}$. We thus think of $u_{jk}(t)$ as an $(J+1) \times (K+1)$ input vector. A state model for the system can then be written as

$$\begin{aligned} x_{jk}(t+1) &= x_{jk}(t) + b_1 u_{jk}(t); \quad t \in \mathbb{N}_0 \\ x_{jk}(0) &= 0; \quad j \in \{0, \ldots, J\}, \quad k \in \{0, \ldots, K\} \end{aligned} \tag{1}$$

where b_1 is a constant that reflects the effect of one unit of mining action.

2.3 Constraints

Mining operations are subject to a number of constraints including constraints on the order that material can be mined, slope constraints, mining capacity, etc., see for example [1–3, 6–13, 30, 32]. In the state-space formulation presented in Section 2.2, these constraints can be incorporated in a natural way.

For example, note that $u_{jk}(t)$ can take either the value 1 or 0 indicating the action of mining or not at location jk at time t. Thus, $u_{jk}(t)$ is nonnegative and the model (1) readily ensures that the mine depth cannot decrease at any location. Also, slope constraints on the mining depth can be directly incorporated by means of state constraints of the form

$$|x_{\ell n}(t) - x_{jk}(t)| \leq b_2; \qquad t \in \mathbb{N}_0, \quad |\ell - j| = 1, \quad |n - k| = 1 \tag{2}$$

The mining capacity constraint can be easily handled by imposing an input constraint such that only a certain number of $u_{jk}(t)$ can be nonzero at any t. Other constraints, such as processing plant constraints, can also be modelled by introducing functions to model ore content. Finally, the state-space formulation presented here can be extended to more complex situations, such as multiple processing plants with variable capacities, multiple material stockpiles, variable material price, etc.

2.4 The Cost Function Representing Net Present Value

The value of the body of ore at different locations is typically obtained by preliminary drilling work. Using this information one can construct a value function $V_{jk}(x_{jk})$ which represents the value assigned to the material in location jk at depth x_{jk}. We also introduce a time discounting function d_t to yield net present value and assume that the price of ore at time t is c_t. The cost function to use for mine planning, representing the net present value achieved by a given mining strategy over a planning horizon T, then takes the form

$$J := \sum_{t=1}^{T} \sum_{j=1}^{J} \sum_{k=1}^{K} d_t c_t \, V_{jk}(x_{jk}(t)) \, u_{jk}(t-1) \tag{3}$$

Note that we multiply by $u_{jk}(t-1)$ in (3) since the value in the ore is only liberated when it is mined.

2.5 Introduction of Uncertainty

The formulation described so far assumes that all quantities are known into the future. To illustrate the impact of uncertainty on the mine-planning problem, we assume that the only uncertain variable is c_t, the price of the ore at time t.

The simplest possible model for the ore price is as a deterministic sequence $\{c_t\}$. For example, if we take copper as an illustration, then the price has been falling in real terms (i.e., when expressed in 2006 dollars). However, this falling trend has also been accompanied by random fluctuations and shocks. Thus there is a strongly uncertain component in the planning description.

We will illustrate the nature of this uncertainty by examining copper price.

We will consider data normalised by the Consumer Price Index (CPI) conversion factor in the USA, that is, in terms of a fixed buying power for a unit of currency (USD). Typical data is shown in Figure 3 (USA copper data from 1967 to 2002) normalised by the CPI.

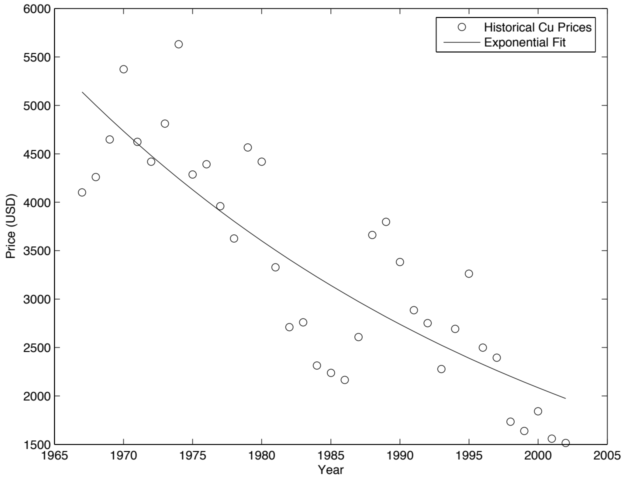


Fig. 3. USA Copper price (normalised by the CPI) and its exponential trend.

We see from Figure 3 that there has been a general trend downwards (exponential curve in the figure) with some added stochastic behaviour including some key 'shocks' at some points in time. It will be convenient for data-fitting purposes to first remove the deterministic trend by fitting an exponential and then subtracting it. The detrended data is shown in Figure 4.

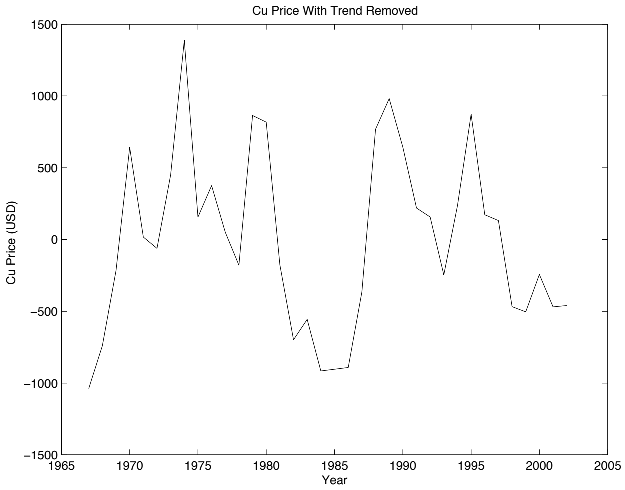


Fig. 4. Copper price after removing exponential trend.

It would be relatively easy to give a stochastic description for the data given in Figure 4. Indeed, there is a number of standard models used in mine-planning applications. These include mean-reverting models, etc. (see, for example, [22]). Note, however, that these models contain real valued random variables. These models are useful for generating Monte Carlo simulations but are unsuitable for use in *closed-loop* mine planning due to the cardinality of the uncertainty description. Indeed, as we have already remarked, the mine-planning problem is barely computationally feasible even if all variables are considered deterministic and known. Hence, attempting function optimisation with real valued random variables for ore price would be absolutely out of the question. Thus, this represents an interesting theory/practice gap. Consequently, in practice we are forced to model the uncertain variables by a random variable having a discrete distribution. Indeed, the cardinality of the support of the distribution must be very small to make the problem computationally feasible. Thus we must carefully choose a 'representative' set of scenarios to describe the uncertain variables for the purpose of closed-loop planning. Further information on scenario generation is obtained in [15, 18–20, 23, 26–28]. Note that these methods have an additional complexity beyond that normally encountered in system identification [16], since the cardinality of the possible realisations must be restricted. Assuming that the scenarios are $c_t(s)$ for $s = 1, \ldots, S$, where t denotes the time (in years), then the cost function for the s-th scenario becomes

$$J_s := \sum_{t=1}^{T} \sum_{j=1}^{J} \sum_{k=1}^{K} d_t c_t(s)\, V_{jk}(x_{jk}(t))\, u_{jk}(t-1) \tag{4}$$

In the sequel, we will describe methods for optimising (the expected value of) cost functions of the general form of (4). We first take a short digression to outline, in general terms, the key concepts underlying 'open-loop', 'reactive' and 'closed-loop' planning.

2.6 Open-loop and Reactive Planning

In 'open-loop' planning one solves an optimisation problem at the outset based on the expected value of future ore price. Specifically, this policy minimises (subject to the relevant constraints) the cost function

$$J^{OL} := \sum_{s=1}^{S} p_s J_s \tag{5}$$

where J_s, defined in (4), is the cost associated with each scenario and p_s denotes the probability of the s-th scenario. The resulting optimal mining strategy is a sequence of mining actions over the planning horizon T. Note that this policy does not use the information that future knowledge about price will be available, that is, that the current ore price will actually become known at each time step.

In 'reactive' planning one applies only the first mining action resulting from the open-loop policy described above. At the next step one solves another open-loop problem over a planning horizon of length $T - 1$ and 'reacts' to the current value of ore price by using this value for the first time step. The procedure is repeated at each time step in a 'rolling-horizon' fashion. Hence, although this policy uses the current value of ore price at each step, the fact that future information about ore price will be available is not taken into account.

2.7 Closed-loop Planning

'Closed-loop' planning (sometimes called 'with recourse' in the stochastic programming literature) takes advantage of the fact that, in the future, the price will be known. Such policies involve a function optimisation that links the information state to the control action.

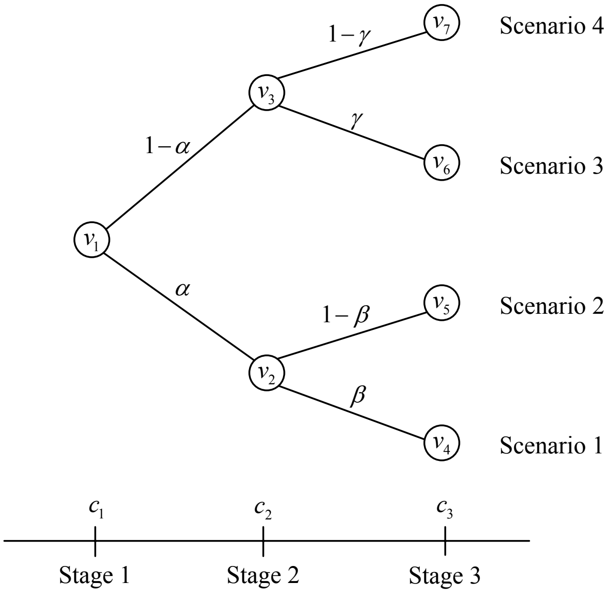


Fig. 5. An example of a scenario tree structure for closed-loop mine planning.

A simple way of capturing the closed-loop planning idea is to allocate a separate control strategy to each scenario. This idea is illustrated by Figure 5, where four different price scenarios are considered. In this figure, price at stage 1 can only take the value $c_1 = v_1$; at stage 2, price can take the value $c_2 = v_2$, with probability α, or $c_2 = v_3$, with probability $1 - \alpha$; at stage 3, if price at stage 2 was v_2, then price can take either the value $c_3 = v_4$, with *conditional* probability β, or $c_3 = v_5$, with *conditional* probability $1 - \beta$; and similarly for the values v_6 and v_7. The price scenarios are then defined by each of the four branches of the scenario tree (for example, scenario 1 corresponds to $c_1 = v_1$, $c_2 = v_2$ and $c_3 = v_4$). In principle, we think of using 4 separate mines, each corresponding to a separate price scenario. Note that the scenarios have a tree like structure, i.e., $c_1(1) = c_1(2) = c_1(3) = c_1(4)$, $c_2(1) = c_2(2)$ and $c_2(3) = c_2(4)$. To bring all of the separated problems together, we note that we should not use future price information until we actually reach that time. This can be captured by ensuring that the mining strategies are nested (i.e., they share common components).

We now introduce four corresponding mining optimisation problems with inputs $u_{jk}(t, s)$, for $s \in \{1, 2, 3, 4\}$, where

- $u_{jk}(t, s) = 1$ means we mine at time t at the location jk under price scenario s.
- $u_{jk}(t, s) = 0$ means we do not mine at time t at the location jk under price scenario s.

Correspondingly, the state models satisfy, for $s = 1, 2, 3, 4$,

$$\begin{aligned} x_{jk}(t+1, s) &= x_{jk}(t, s) + b_1 u_{jk}(t, s); \qquad t \in \mathbb{N}_0 \\ x_{jk}(0, s) &= 0; \qquad j \in \{0, \dots, J\}, k \in \{0, \dots, K\} \end{aligned} \tag{6}$$

As outlined above, it is important that we do not use information about price until it becomes available. This is captured by adding constraints that ensure that the mining strategies are equal at each node of the price scenario tree. For the example above, these constraints have the form:

$$\begin{aligned} &u_{jk}(0, 1) = u_{jk}(0, 2) = u_{jk}(0, 3) = u_{jk}(0, 4) \\ &u_{jk}(1, 1) = u_{jk}(1, 2) \\ &u_{jk}(1, 3) = u_{jk}(1, 4); \qquad j \in \{0, \dots, J\}, k \in \{0, \dots, K\} \end{aligned} \tag{7}$$

3 Simulation Results

Real mine-planning problems are exceedingly complex and can take weeks to solve on high-speed computers. Also, they are notoriously non-convex. Hence, it is difficult to gain insights by examining real problems since one readily becomes lost in details. Accordingly, we will take a simplified ‘toy’ example to gain insight into the structure of the various planning strategies. We thus choose a (very) simplified mining example as depicted in Figure 6.

A vertical ore bar of length $x_{\max} - x_{\min}$ is assumed to be located at a depth $x_{\min}$ in the soil. Our goal is to extract this ore in an optimal fashion subject to constraints. In particular, we assume that we can only extract a certain maximum amount of material per year, and that wall slope constraints must be satisfied. We also assume that all extracted ore is immediately sold at the current ore price, which varies in a ‘random’ way. Our goal is to maximise the net present value of the total return. Without loss of generality, we include the discount factor in the ore price.

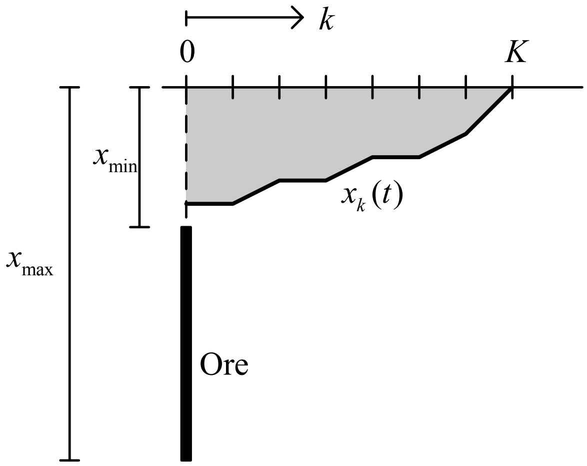


Fig. 6. A simplified mine-planning problem.

We consider a two-dimensional version of the problem, where we are looking at a cross section of the soil. Using time and space quantisation, let $x_k(t)$ be the depth of the soil at a distance $k \in \{0, \dots, K\}$ to the right of the bar, at a time $t \in \{0, \dots, T\}$, where $T \in \mathbb{N}$ is the time-horizon length and K is the maximum distance of the bar to be considered. Notice that, due to the symmetry of the problem, we need only consider positive values of k, since the solution for $k < 0$ is necessarily the mirror reflection of the solution for $k > 0$.

The cost function to be maximised is

$$J := E\left\{\sum_{t=1}^{T} c_t \mu(x_0(t-1), x_0(t))\right\} \tag{8}$$

where μ is a (positive Borel) measure of the ore distribution and $\{c_t\}_{t \in \{1,\dots,T\}}$ is the ore price (considered as a stochastic process, to be specified later).

To simplify the problem further, we make the assumption that the bar has infinite length and that its top lies at the soil surface, i.e., that

$$\begin{aligned} x_{\min} &= 0 \\ x_{\max} &= \infty \end{aligned} \tag{9}$$

The cost function (8) can then be written as

$$J = E\left\{\sum_{t=1}^{T} c_t [x_0(t) - x_0(t-1)]\right\} \tag{10}$$

The constraints for this problem are:

1. *Initial condition*

$$x_k(0) = 0; \qquad \text{for all } k \in \{0, \dots, K\} \tag{11}$$

2. *Maximum amount of material to be extracted per year*

$$\frac{x_0(t) - x_0(t-1) + x_K(t) - x_K(t-1)}{2} + \sum_{k=1}^{K-1} [x_k(t) - x_k(t-1)] \leq \frac{C}{2}; \qquad \text{for all } t \in \{1, \dots, T\} \tag{12}$$

   where $C \in \mathbb{R}^+$. The form of this constraint comes from a trapezoidal approximation to the integral of the extracted material.

3. *Nonnegativity constraints*

$$x_k(t) \geq x_k(t-1); \qquad \text{for all } k \in \{0, \dots, K\},\ t \in \{1, \dots, T\} \tag{13}$$

4. *Slope constraints*

$$|x_{k-1}(t) - x_k(t)| \leq 1; \qquad \text{for all } k \in \{1, \dots, K\},\ t \in \{1, \dots, T\} \tag{14}$$

Notice that both the cost function and the constraints are linear in the decision variables. Thus, the problem can be described as a stochastic linear program and, as we will see, it can be solved by rewriting it as a deterministic linear program (thus avoiding having to state and solve Bellman's equation explicitly [4, 29]).

In the following, we will restrict attention to a simple three-stage price model as shown in Figure 5.

According to this scenario tree, we have that

$$\begin{aligned}
P\{c_2 = v_2\} &= \alpha \\
P\{c_2 = v_3\} &= 1 - \alpha \\
P\{c_3 = v_4 | c_2 = v_2\} &= \beta \\
P\{c_3 = v_5 | c_2 = v_2\} &= 1 - \beta \\
P\{c_3 = v_6 | c_2 = v_3\} &= \gamma \\
P\{c_3 = v_7 | c_2 = v_3\} &= 1 - \gamma
\end{aligned} \tag{15}$$

We next transform the above stochastic problem into a deterministic one by introducing a new state sequence $x'_k(t, s)$, where k denotes spatial position, t denotes time (or stage) and s denotes the scenario number. Thus, we have

$$x_k(1) = x'_k(1,1) = x'_k(1,2) = x'_k(1,3) = x'_k(1,4); \qquad k \in \{0, \dots, K\} \tag{16}$$

$$x_k(2) = \begin{cases} x'_k(2,1) = x'_k(2,2), & c_2 = v_2 \\ x'_k(2,3) = x'_k(2,4), & c_2 = v_3 \end{cases} \qquad k \in \{0,\dots,K\} \tag{17}$$

$$x_k(3) = \begin{cases} x'_k(3,1), & c_3 = v_4 \\ x'_k(3,2), & c_3 = v_5 \\ x'_k(3,3), & c_3 = v_6 \\ x'_k(3,4), & c_3 = v_7 \end{cases} \qquad k \in \{0,\dots,K\} \tag{18}$$

Then the cost of the deterministic equivalent program can be written as

$$\begin{aligned} J &= E\{c_1[x_0(1) - x_0(0)] + c_2[x_0(2) - x_0(1)] + c_3[x_0(3) - x_0(2)]\} \\ &= v_1 x'_0(1,1) + \\ &\quad + v_2 \alpha [x'_0(2,1) - x'_0(1,1)] + v_3(1-\alpha)[x'_0(2,3) - x'_0(1,1)] + \\ &\quad + \alpha\, (v_4 \beta [x'_0(3,1) - x'_0(2,1)] + v_5(1-\beta)[x'_0(3,2) - x'_0(2,1)]) + \\ &\quad + (1-\alpha)\, (v_6 \gamma [x'_0(3,3) - x'_0(2,3)] + v_7(1-\gamma)[x'_0(3,4) - x'_0(2,3)]) \end{aligned} \tag{19}$$

The constraints for this program are

$$\frac{x'_0(1,1) + x'_K(1,1)}{2} + \sum_{k=1}^{K-1} x'_k(1,1) \leq \frac{C}{2} \tag{20}$$

$$\frac{x'_0(2,s) + x'_K(2,s) - x'_0(1,1) - x'_K(1,1)}{2} + \sum_{k=1}^{K-1} [x'_k(2,s) - x'_k(1,1)] \leq \frac{C}{2}; \qquad s \in \{1,3\}$$

$$\frac{x'_0(3,s) + x'_K(3,s) - x'_0(2,s) - x'_K(2,s)}{2} + \sum_{k=1}^{K-1} [x'_k(3,s) - x'_k(2,s)] \leq \frac{C}{2}; \qquad s \in \{1,2,3,4\}$$

$$\begin{aligned} x'_k(1,1) \geq 0; &\quad k \in \{0,\dots,K\} \\ x'_k(t,s) - x'_k(t-1,s) \geq 0; &\quad k \in \{0,\dots,K\}, s \in \{1,2,3,4\}, t \in \{2,\dots,T\} \end{aligned} \tag{21}$$

$$|x_{k-1}(t,s) - x_k(t,s)| \leq 1; \qquad k \in \{1,\dots,K\}, s \in \{1,2,3,4\}, t \in \{1,\dots,T\} \tag{22}$$

The solution of this program gives the optimal (closed-loop) solution of the mine-planning problem.

Alternatively, if we are interested in the reactive solution to this problem, we can achieve this by adding extra constraints as follows:

$$\begin{array}{ll} x'_k(2,1) = x'_k(2,3); & k \in \{0,\ldots,K\} \\ x'_k(3,1) = x'_k(3,2) = x'_k(3,3) = x'_k(3,4); & k \in \{0,\ldots,K\} \end{array} \tag{23}$$

The inclusion of these constraints gives us a deterministic linear program whose solution provides $x'_k(1,1)$ for $k \in \{0,\ldots,K\}$, since they force a open-loop solution for stage 1. Similarly, we can repeat for stage 2 by considering two problems depending whether c_2 takes the value v_2 or v_3.

We will examine several sets of pricing data.

(i) Pricing data #1

Here we let $K = 40$, $C = 400$ and assume

$$\begin{array}{ll} v_1 = 0.5 & \alpha = 32/33 \\ v_2 = 0.1 & \beta = 17/24 \\ v_3 = 10 & \gamma = 17/24 \\ v_4 = 0.01 & \\ v_5 = 0.25 & \\ v_6 = 1 & \\ v_7 = 25 & \end{array} \tag{24}$$

Under these conditions, it turns out that the closed-loop and reactive policies coincide. The solution for the first two stages is shown in Figure 7, and the solution for stage 3 is shown in Figure 8.

It may seem strange that the reactive and closed-loop policies are identical for this example. However, this can be explained as follows:

We note that

$$\begin{aligned} c_1 = v_1 = 0.5 > 0.4 &= E\{c_2\} \\ E\{c_2\} = 0.4 > 0.32 &= E\{c_3\} \\ v_2 = 0.1 > 0.08 &= E\{c_3 | c_2 = v_2\} \\ v_3 = 10 > 8 &= E\{c_3 | c_2 = v_3\} \end{aligned} \tag{25}$$

Thus, at every stage the ore price ‘tends’ to decrease (in a mean sense), so common sense tells us that the best strategy is to extract as much ore as possible as soon as one can in order to maximise the return. Since this tendency is exhibited in both a conditional and unconditional sense (with respect to the information available at the present time), the optimality of this strategy does not depend on how much information we actually have at a specific stage, and thus the closed-loop and reactive policies must coincide.

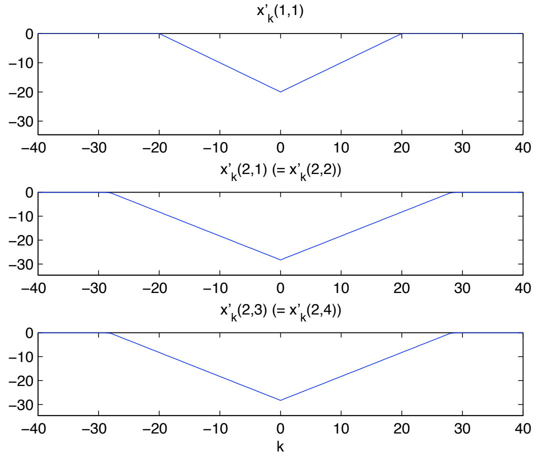


Fig. 7. First two stages of the solution to the three-stage mine planning problem, with parameters given by (24).

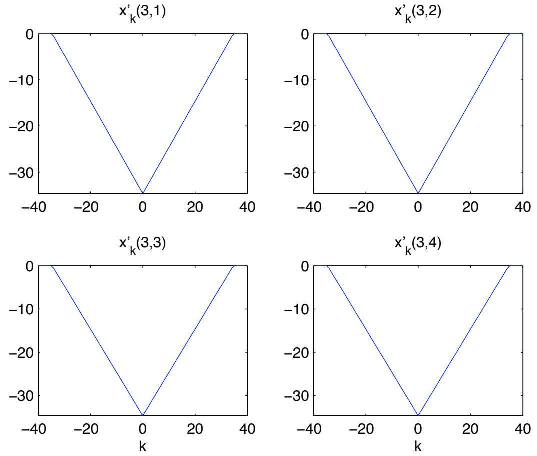


Fig. 8. Third stage of the solution to the three-stage mine planning problem, with parameters given by (24).

Notice that the coefficients of the cost function (19) are proportional to the differences in the *conditional* means. On the other hand, if we introduce the constraints (23) in the cost function (19), the coefficients of the resulting cost function are proportional to the differences in the *unconditional* means. This observation supports the conclusions of the previous paragraph.

(ii) Pricing data #2

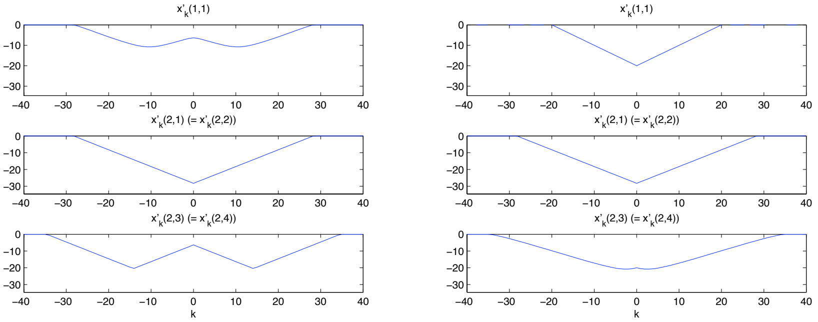
The above discussion suggests the form that alternative pricing data would need to take to ensure that reactive and closed-loop policies were different. For example, if we now let $K = 40$, $C = 100$ and choose

$$\begin{aligned} v_1 &= 0.41 & \alpha &= 32/33 \\ v_2 &= 0.1 & \beta &= 17/24 \\ v_3 &= 10.5 & \gamma &= 7/12 \\ v_4 &= 0.01 \\ v_5 &= 0.25 \\ v_6 &= 1 \\ v_7 &= 25 \end{aligned} \tag{26}$$

we now have that

$$\begin{aligned} c_1 = v_1 = 0.41 &> 0.40588 \approx E\{c_2\} \\ E\{c_2\} \approx 0.40588 &> 0.40118 \approx E\{c_3\} \\ v_2 = 0.1 &> 0.08 = E\{c_3 | c_2 = v_2\} \\ v_3 = 10.5 &< 11 = E\{c_3 | c_2 = v_3\} \end{aligned} \tag{27}$$

This means that the ore price exhibits an ‘unconditional tendency’ to decrease, but if $c_2 = v_3 = 10.5$, c_3 ‘conditionally’ tends to increase. Thus, the information available at stage 2 has the potential to be quite valuable when making a decision. The above line of reasoning suggests that, in this case, closed-loop and reactive policies might differ. This is indeed exactly what happens, as is shown in Figure 9.



(a) (b)

Fig. 9. Solution of the three-stage mine-planning problem for the parameters given in (26). (a) Closed-loop solution. (b) Reactive solution. The third stage of the solution given by both policies is the same as the one shown in Figure 8.

4 Discrepancy Between Theory and Experiments

4.1 Discussion of the Simple Example of Section 3

Whilst it may seem, at first sight, that closed-loop planning would offer substantial advantages, the differences between closed-loop and reactive planning appear to be relatively modest for the mine planning problem. Specifically, the costs obtained for the example of Section 3 when using the closed-loop and reactive policies are 14.37744 and 14.23304, respectively. The difference may seem small (1%), but in the mine industry even a 1% improvement can equate to tens of millions of dollars! This means that there may indeed be cases where it might be worthwhile to obtain the closed-loop solution instead of just computing the reactive one.

Perhaps of more interest in this example, is the qualitative difference between the strategies. Indeed, we see from Figure 9(b), that the reactive policy simply tries to extract as much ore as possible. On the other hand, the closed-loop policy (Figure 9(a)) makes preparations for a possible future price increase by removing some waste material so that one can be in a better position, at the next step, to exploit a potential price increase by then allowing more ore to be extracted.

4.2 Some General Observations

Although it is difficult in general to know in advance whether there will be differences between the three kinds of stochastic programming policies for a particular problem, we have seen that for the above mine-planning example it is relatively simple to arrive at some necessary conditions for the existence of discrepancies. In particular, there have to be discrepancies between the signs of the differences of the conditional means and the signs of the differences of the unconditional means.

Another interesting observation we can make from this example is that the number of possible optimal solutions is essentially finite, in the sense that to achieve the optimal cost, we only need consider a finite number of strategies. This follows from the fact that the mine-planning problem is equivalent to a deterministic *linear* program, which has a finite number of basic solutions [21, pg. 20].

Moreover, from the sensitivity theory of linear programming [33, chapter 7], we know that small changes in the coefficients of the cost function of a linear program do not change the optimal solution (except for a set of parameter values of Lebesgue measure zero), and the optimal cost is a continuous function of those coefficients.

Since the coefficients of the cost function are directly related to the scenario tree, the above line of reasoning implies that the solution of the mine planning problem is 'locally insensitive' to changes in the scenario tree (at least assuming that its structure remains unchanged), but it may suffer 'global bifurcations', since the solution may 'jump' to other entirely different solutions for other changes in the scenario tree. This behaviour is evident in the difference between the solutions to the problem described above under Pricing Data #1 and Pricing Data #2.

The previous remark is quite interesting, since it implies that it is unhelpful to try to find an 'exact' scenario tree to represent the uncertainty in the ore price. On the other hand, it can be very insightful to generate several scenario trees in order to study the robustness of the obtained solution to changes in the ore price structure. (This idea can be contrasted with the 'contamination technique' developed in [14], which is aimed at studying the local sensitivity, or *stability*, of the stochastic program.)

4.3 Application to a Real Mine-planning Problem

The work described in this chapter has been carried out in collaboration with personel at BHP Billiton Melbourne Technology Centre. They have tested the relative efficacy of the different policies on real problems. For simple ore geometries it turns out that reactive planning achieves near optimal performance. However, for more complex geometries, greater differences between open-loop, reactive and closed-loop planning become apparent.

To illustrate the potential discrepancies between the theory and practice, we consider a real mine planning exercise (actually based on the mine illustrated in Figure 1). Three planning methods were tested, namely,

- open-loop,
- reactive,
- perfect knowledge.

The latter policy cannot be carried out in practice since future prices are unknown. However, it does provide an upper bound on the achievable closed-loop performance.

Table 1 tabulates the average NPVs obtained over 6 tests using the expected forward price conditional on the current spot price (i.e., "present knowledge" or "open loop" approach), the reactive approach and perfect knowledge of future price. In each test, 25 equally likely price realisations were generated using Monte Carlo simulations based on a log-normal mean reverting price model.

Test No.	p_0	$\bar{p}$	$\mathrm{NPV}_{Conditional}$	$\mathrm{NPV}_{Reactive}$	$\mathrm{NPV}_{Perfect}$
1	0.68	0.89	1622	1662	1685
2	0.89	0.68	1262	1287	1306
3	0.68	0.68	1180	1207	1231
4	0.25	0.45	303	358	370
5	0.35	0.35	185	246	256
6	0.45	0.25	164	197	206

Table 1. Average NPVs for different planning methods

The notation used in Table 1 is as follows:

- p_0 [$/lb]: the current spot price at time 0;
- $\bar{p}$ [$/lb]: the long term equilibrium commodity price;
- $\mathrm{NPV}_{Conditional}$ [million $]: the average NPV obtained using the expected forward price conditional on the current spot price;
- $\mathrm{NPV}_{Reactive}$ [million $]: the average NPV obtained using the reactive approach;

- $\mathrm{NPV}_{Perfect}$ [million \$]: the average NPV obtained assuming that the future price is perfectly known prior to producing a mine plan.

Figures 10 and 11 show some plots of the results of Test 4 (where $p_0 = 0.25$ and $\bar{p} = 0.45$). Figure 10 shows the NPVs obtained using the three planning methods for each price scenario. Figures 11 (a) and (b) show the histograms of the NPV for each price scenario using the "present knowledge" approach and the "reactive" approach, respectively.

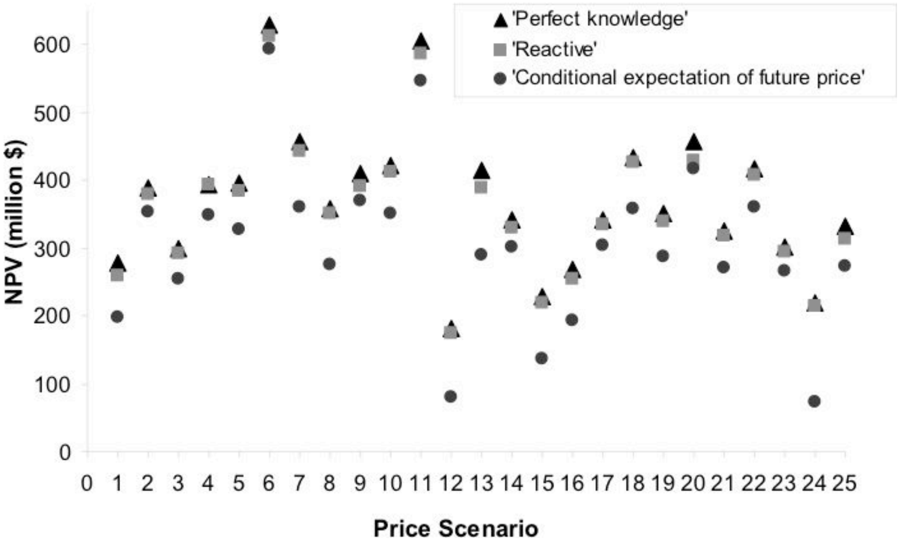


Fig. 10. NPV for each price scenario using three mining project evaluation methods

It can be seen from the above results that, for this particular mine, the reactive approach yields approximately 10% improvement over the open-loop approach. This strongly suggests that reactive planning offers considerable benefits over open-loop planning. On the other hand, perfect knowledge yields only a further 2.7% improvement. Thus, for this particular mine the gap between reactive planning and planning with perfect knowledge is quite small. Thus one might anticipate that the improvements resulting from closed-loop planning could be small. However, one should be careful to draw general conclusions from this simple example. Indeed, as discussed in Section 4.2, there can be situations where closed-loop planning can offer substantial qualitative and quantitative advantages over reactive planning.

4.4 Conclusions

This chapter has described the application of closed-loop stochastic optimal control to optimal mine planning. Key ideas that have emerged are:

1. This is a quintessential example of a complex optimisation problem.

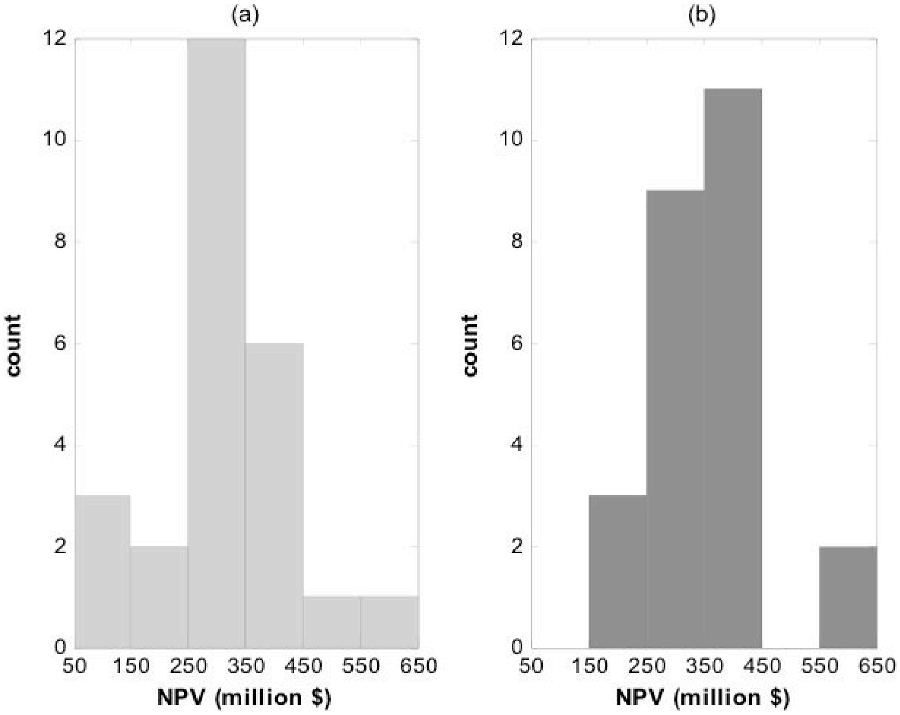


Fig. 11. Histograms of NPV for each price scenario using: (a) the method with expected price conditional on the current spot price and, (b) the reactive approach.

2. There is a substantial theory/practice gap associated with this problem and, indeed, to have the slightest hope of solving it, one needs to make major simplifications.
3. For certain ore geometries and problem parameters, there appears to be little practical gain to be achieved by using closed-loop planning rather than reactive planning.
4. For alternative ore geometries and/or different problem parameters, closed-loop planning can be beneficial.
5. Due to the enormous complexity of this problem, one needs to carefully choose a representative set of scenarios for the uncertain variables.
6. The cardinality of the scenarios must be quite small to be able to solve the closed-loop planning problem in practice.
7. Real mine-planning problems involve a host of complexities not touched on here.
8. The ideas presented here undoubtedly have relevance to other complex planning problems outside of the mining sphere.

Acknowledgements

The authors gratefully acknowledge the input of many others in writing this chapter. In particular, they wish to specifically mention Professor Richard H. Middleton and Mr. Brian F. Hennessy (University of Newcastle) and Dr. Peter M. Stone and Dr. Merab Menabde (BHP Billiton Melbourne Technology Centre).

References

[1] Appiah, P. B., Rosenman, M. A., and Sturgul, J. R. (1990). Introduction to Pareto optimal mine planning. *Journal of Mining and Geological Engineering* **8(4)**, 348–356.

[2] Appiah, P. B. and Sturgul, J. R. (1990). Pareto optimal stochastic mine planning. *International Journal of Surface Mining, Reclamation and Environment* **4(4)**, 181–195.

[3] Ataei, M. and Osanloo, M. (2003). Using a combination of genetic algorithm and the grid search method to determine optimum cutoff grades of multiple metal deposits. *International Journal of Surface Mining, Reclamation and Environment* **18(1)**, 60–78.

[4] Bellman, R. (1957). *Dynamic Programming*. Princeton University Press, Princeton, N. J.

[5] Beltratti, A., Consiglio, A., and Zenios, S. A. (1999). Scenario modeling for the management of international bond portfolios. *Annals of Operations Research* **85**(0), 227–247.

[6] Brazil, M., Lee, D. H., Van Leuven, M., Rubinstein, J. H., Thomas, D. A., and Wormald, N. C. (2003). Optimising declines in underground mines. *Mining Technology: Trans. of the Institution of Mining and Metallurgy, Section A* **112**, 164–170.

[7] Caccetta, L. and Hill, S. P. (2003). An application of branch and cut to open pit mine scheduling. *Journal of Global Optimization* **27**, 349–365.

[8] Clement, S. and Vagenas, N. (1994). Use of genetic algorithms in a mining problem. *International Journal of Surface Mining, Reclamation and Environment* **8**, 131–136.

[9] Darwen, P. J. (2001). Genetic algorithms and risk assessment to maximize NPV with robust open-pit scheduling. *Fourth Biennial Conference on Strategic Mine Planning*, pp. 29–34. Perth, Western Australia.

[10] Denby, B. and Schofield, D. (1994). Open-pit design and scheduling by use of genetic algorithms. *Transactions of the Inst. Min. Metall., Section A: Min. industry* **103**, A21–A26.

[11] Denby, B. and Schofield, D. (1995). Inclusion of risk assessment in open pit design and scheduling. *Transactions of the Inst. Min. Metall., Section A: Min. industry* **104**, A67–A71.

[12] Denby, B. and Schofield, D. (1995). The use of genetic algorithms in underground mine scheduling. *Proceedings of the 25th International Symposium Application of Computers Mineral Industries*, pp. 389–394. Brisbane, Australia.

[13] Denby, B., Schofield, D., and Surme, T. (1998). Genetic algorithms for flexible scheduling of open pit operations. *Proceedings of APCOM 27th International Symposium Mathematics in the Mineral Industries*. London, UK.

[14] Dupačová, J. (1996). Scenario-based stochastic programs: resistance with respect to sample. *Annals of Operations Research* **64**, 21–38.

[15] Dupačová, J., Consigli, G., and Wallace, S. W. (2000). Scenarios for multistage stochastic programs. *Annals of Operations Research* **100**, 25–53.

[16] Goodwin, G. C. and Payne, R. L. (1977). *Dynamic System Identification: Experiment Design and Data Analysis*. Academic Press, New York.

[17] Goodwin, G. C., *et al.* (2006). Receding horizon control applied to optimal mine planning. *Automatica* **42**(8), 1337–1342.

[18] Høyland, K., Kaut, M., and Wallace, S. W. (2003). A heuristic for moment-matching scenario generation. *Computational Optimization and Applications* **24**(2-3), 169–185.

[19] Høyland, K. and Wallace, S. W. (2001). Generating scenario trees for multistage decision problems. *Management Science* **47**(11), 295–307.

[20] Keefer, D. L. (1994). Certainty equivalents for three-point discrete-distribution approximations. *Management Science* **40**(6), 760–773.

[21] Luenberger, D. G. (1984). *Linear and Nonlinear Programming*. Addison-Wesley, Reading, Massachussetts, 2nd edition.

[22] Metcalf, G. and Hassett, K. (1995). Investment under alternative return assumptions: comparing random walks and mean reversions. *J. Econ. Dyn. Control* **19**, 1471–1488.

[23] Miller, A. C. and Rice, T. R. (1983). Discrete approximations of probability distributions. *Management Science* **29**(3), 352–362.

[24] Mulvey, J. M. and Vladimirou, H. (1992). Stochastic network programming for financial planning problems. *Management Science* **38**(11), 1642–1664.

[25] Pflug, G. (2001). Scenario tree generation for multiperiod financial optimization by optimal discretization. *Math. Programming, Ser. B* **89**, 251–257.

[26] Pflug, G. C. (1996). *Optimization of Stochastic Models: The Interface Between Simulation and Optimization*. Kluwer Academic Publishers, Boston.

[27] Pflug, G. C. (2001). Scenario tree generation for multiperiod financial optimization by optimal discretization. *Math. Program., Ser. B* **89**, 251–271.

[28] Rachev, S. T. (1991). *Probability Metrics and the Stability of Stochastic Models*. John Wiley and Sons, New York.

[29] Söderström, T. (1994). *Discrete-Time Stochastic Systems: Estimation and Control*. Prentice Hall, London.

[30] Stone, P., Froyland, G., Menabde, M., Law, B., Pasyar, R., and Monkhouse, P. (2004). Blasor–blended iron-ore mine planning optimisation at Yandi. *Orebody Modelling and Strategic Mine Planning", Proceedings of the International Symposium, AIMM*, pp. 285–288. Perth, Australia.

[31] Takriti, S., Birge, J., and Long, E. (1996). A stochastic model for the unit commitment problem. *IEEE Transactions on Power Systems* **11**(3), 1497–1508.

[32] Thomas, G. S. (1996). Optimization and scheduling of open pits via genetic algorithms and simulated annealing. *Proceedings of the 1st International Symposium on Balkema Publisher*, pp. 44–59. Rotterdam, The Netherlands.

[33] Vanderbei, R. J. (2004). *Linear Programming: Foundations and Extensions*. Springer, New York.

[34] Yu, L.-Y., Ji, X.-D., and Wang, S.-Y. (2003). Stochastic programming models in financial optimization: A survey. *AMO-Advanced Modeling and Optimization* **5**(1).

Index