Noname manuscript No.
(will be inserted by the editor)

A Framework for Adaptive Open-pit Mining Planning under Geological Uncertainty

Tomás Lagos · Margaret Armstrong · Tito Homem-de-Mello · Guido Lagos · Denis Sauré

March 6, 2020

Abstract Mine planning optimization aims at maximizing the profit obtained from extracting valuable ore. Beyond its theoretical complexity—the open-pit mining problem with capacity constraints reduces to a knapsack problem with precedence constraints, which is NP-hard—practical instances of the problem usually involve a large to very large number of decision variables, typically of the order of millions for large mines. Additionally, any comprehensive approach to mine planning ought to consider the underlying geostatistical uncertainty as only limited information obtained from drill hole samples of the mineral is initially available. In this regard, as blocks are extracted sequentially, information about the ore grades of blocks yet to be extracted changes based on the blocks that have already been mined. Thus, the problem lies in the class of multi-period large scale stochastic optimization problems with decision-dependent information uncertainty. Such problems are exceedingly hard to solve, so approximations are required. This paper presents an adaptive optimization scheme for multi-period production scheduling in open-pit mining under geological uncertainty that allows us to solve practical instances of the problem. Our approach is based on a rolling-horizon adaptive optimization framework that learns from new information that becomes available as

T. Lagos · D. Sauré
Department of Industrial Engineering, University of Chile, Chile

M. Armstrong
School of Business, Universidad Adolfo Ibañez, Chile

M. Armstrong
Center for Mathematical Modelling, University of Chile, Chile

T. Homem-de-Mello E-mail: tito.hmello@uai.cl
School of Business, Universidad Adolfo Ibañez, Chile

G. Lagos
Faculty of Engineering and Sciences, Universidad Adolfo Ibañez, Chile

G. Lagos
Department of Industrial Engineering, Universidad de Santiago de Chile, Chile

blocks are mined. By considering the evolution of geostatistical uncertainty, the proposed optimization framework produces an operational policy that reduces the risk of the production schedule. Our numerical tests with mines of moderate sizes show that our rolling horizon adaptive policy gives consistently better results than the non-adaptive stochastic program for a range of realistic problem instances.

Keywords Mine planning; geostatistics; stochastic optimization; adaptive algorithms; learning

Declarations

- *Funding*: This research has been supported by grant Programa de Investigación Asociativa (PIA) ACT1407, Chile. Guido Lagos also acknowledges the financial support of FONDECYT grant 3180767, Chile.
- *Conflicts of interest*: None.
- *Availability of data and material*: Please contact authors.
- *Code availability*: Custom code.

1 Introduction

Motivation. Open-pit mining planning optimization aims at maximizing the profit obtained from extracting valuable ore when a mineral deposit is mined from the surface. A mining project typically starts with a prospecting stage, during which mineral deposits are discovered, followed by an exploration phase, where drilling campaigns are carried out to sample and collect information on the ore content. This information is then used to create a geological model of the mineral content, using techniques such as *kriging*, see **?**. Once a geological model of the deposit is in place, a mine is designed, together with an exploitation plan, both of which are evaluated economically. If this evaluation, which usually includes estimating the net present value of the operation, results in a positive value, the mine is developed and exploitation operations are carried out.

In this context, while long-term extraction plans (dictating when and how the material is extracted) are rather inflexible, their periodic operational counterparts often adapt to new information available and operational settings. In particular, consider the case of information about the ore content of the deposit: as material is extracted, ore content (which up to that point was only estimated based on the geological model of the deposit) is observed, and it might differ significantly from its estimate; thus, based on such values, the extracted material is either processed (to obtain ore), dumped as waste, or transported to a stockpile that can be processed in the future. All this, considering the mine's extraction and processing capacities.

The problem is a particular case of resource-constrained project scheduling, which seeks to optimally schedule activities over time while satisfying resource and precedence constraints. In the case of open-pit mining, these activities are the extraction and processing of the resource. Formulations for solving resource-constrained project scheduling, which is NP-hard in general, have been proposed since the 1960s; see, e.g., **?** and references therein for an overview of such problem and its applications to mine planning. Early work focused on deterministic formulations of the problem, without geological uncertainty, which often led to tractable but rather unrealistic settings. More recently, various stochastic models, incorporating market and/or geological uncertainty, have been presented. While in all such models uncertainty is revealed all at once, in practice however, uncertainty revelation depends on the extraction decisions. This, the problem is an instance of a stochastic optimization with *decision-dependent information uncertainty*, following the terminology proposed by **?**. Such problems are in general hard to solve; in this particular case of mine planning, because of the complexity associated with modeling uncertainty, extant work often trades realism for tractability, and thus available more practical models only have been solved for unrealistic (small) instances of the mining problem.

Objective and Assumptions. In this paper we study an adaptive optimization scheme for multi-period production scheduling in open-pit mining under geological uncertainty. In essence, *adaptive* or *learning* schemes in optimization under uncertainty seek to incorporate new information on mineral content (or *ore grade*) in the optimization decisions as it becomes available; see, e.g., **?**. This idea is especially sound for open-pit mining, where planning operations rely heavily on geostatistical estimates of the mineral content, which are usually based on a few costly drilling campaigns of the mining deposit. Consequently, our objective is to develop a practical framework for producing adaptive operation mining plans, that uses the grades of blocks that have already been mined to update the geostatistical model of grades of unmined blocks, and re-optimizes the production schedule so as to maximize the net present value (NPV) of the operation. While other dynamic models have been proposed in the literature [**???**], to the best of our knowledge we present the first *adaptive* optimization approach for the open-pit mining problem that takes into account the updates of the geostatistical models.

For the sake of context, we tackle the following simplified open-pit mining problem. We consider the usual open-pit setting, where the mine is discretized into a fixed collection of uniformly-sized blocks, and in order to mine a block we have to first mine all the blocks in a cone above it. The deposit is to be exploited over a sequence of time periods; at each period we have to decide which blocks to *extract*, and dependent on their grade, which of those to *process* and which to send to the *waste dump* without being processed. At each time period there is also a finite capacity for extraction and processing. A key feature of the operation is that the actual mineral content in each block is uncertain prior to their extraction. This makes mine planning a *high-risk high-stakes* task, because the size of the problem is very large and blocks can only be mined once. Several approaches have been proposed to tackle geostatistical uncertainty, most notably stochastic programming ones; see, e.g., **??????**.

Extant literature in mining planning that incorporates geological uncertainty [**??????**] typically does so by working with a (large) set of sample grades, which are generated so as to reproduce the statistical properties of the sampled grades (mean, variance, spatial covariance). Although different methods exist for generating these simulations, the one based on a multivariate Gaussian structure is most suitable for updating (for example, in the oil industry, the good properties of simulations based on multivariate Gaussian models are used for *history matching*, that is, for updating reservoir models as production figures become known; see **?** and **?**). The standard geostatistical model used in mining is a *Gaussian field*, which block discretization reduces to a multivariate normal (MVN) or a component-wise transformation of it [**?**] We adopt a multivariate Gaussian model for representing grade uncertainty, and use its properties to efficiently update grade distributions and generate sample grades.

Main contributions. Our first and principal contribution is a *practical* adaptive optimization framework that simultaneously adapts or learns from the latest grade realizations of the deposit, and can solve practical instances of the problem, as it does not introduce extra computational burden to existing non-adaptive approaches to the problem (e.g. two-stage stochastic programming). Indeed, our method takes a "base" non-adaptive scenarios-based approach to the problem, and introduces, at the end of each time period, an extra step consisting of observing the mineral content, updating uncertainty belief, and regenerating the ore grade block model. While these extra steps *per se* can be executed regardless of the base optimization method—so they can actually be performed in practice during the mining operation—a key element of our approach is the ability to incorporate the updated distributions to execute a new optimization plan at every time period. Such feature is turn requires us to implement the simulation of the geostatistical models within the optimization algorithms, unlike the approach typically found in the mine planning literature whereby a set of geostatistical scenarios is fixed in advance.

Our second contribution is to illustrate the particular challenges of studying adaptability in a prototypically "difficult" sequential decision-making problem of large size, but with a fairly "amicable" model of uncertainty, suitable to incorporating learning. Indeed, the problem we study is a prime example of a difficult and large problem. On the one hand, the open-pit mining problem with capacity constraints is known to be NP-hard (via a reduction to the knapsack problem). Moreover, the size of practical instances of the problem (typically driven by number of discretized blocks in the deposit) is usually large to very large (the number of blocks in the deposit is typically of the order of 100,000 in a small mine, and 10,000,000 in a large one). Despite its theoretical complexity, non-adaptive approaches to mine planning can be solved for practical instances of the problem via the use of specialized algorithms. On the other hand, while our model of grade uncertainty is most convenient for Bayesian belief updating (conditional on observed information, grade distribution remains MVN), the sheer number of blocks in practical settings makes it computationally intractable to straightforwardly use available closed-form formulas to compute conditional grade distributions. The proposed approach exploits in an ad-hoc manner the property of *second-order stationarity* of the Gaussian field model for the mineral grades, which is a common assumption in geostatistics and is often assumed in the Bayesian Learning and Bayesian Optimization literature; see, e.g., **?**. In this regard, our approach leverages available non-adaptive approaches to produce an (adaptive) rolling-horizon policy. We do so as a way to isolate the practical complexity of solving the (non-adaptive) mine planning problem from the challenges of the learning or adaptation step.

Our third contribution is presenting an operational policy that reduces (relative to non-adaptive approaches) the risk associated with mine planning. Indeed, our work is motivated by the high risk inherent in the design of production plans in open-pit mining. In addition to ore-grade uncertainty, mine planning is subject to other sources of uncertainty, the prime example of which is commodity price uncertainty. One could argue that price uncertainty can be somewhat mitigated by the use of financial derivatives (e.g. options), although recent work [**?**] shows the benefit of incorporating price uncertainty into the model. Still, because planning horizons usually vary between 5 and

20 years, and mining deposits are very large, planning methods that incorporate geostatistical uncertainty while being computationally manageable might reduce operational risk significantly. To sum up, this makes the production planning problem a high-risk high-stakes operation. In this regard, our proposed framework introduces flexibility to operational planning by means of introducing two additional steps, one of *belief-updating* —or *learning*–, and the other of *re-optimization.* In a broader perspective, our work contributes to a body of literature in engineering that has studied the power of being adaptable, flexible, or both, in decision making under uncertainty; see e.g. **????**.

Finally, our last contribution is to show, via numerical experiments, that the proposed rolling-horizon policy is likely to capture most of the benefits associated with adapting production plans to new information on the grades of exploited blocks. In addition, we develop an alternative approximate dynamic programming (ADP) approach to solve the problem. In theory, Dynamic Programming policies might perform arbitrarily better than the proposed policies, and ADP policies aim at replicating such a performance improvement (we demonstrate that this is indeed the case by means of a “toy” example). In practical setting, however, we show that the ADP approach does not perform better than the proposed policies. This is explained by the facts that: i) the performance of our policy leave little room for improvement; and ii) additional approximations introduced by the ADP approach, necessary to maintain computational tractability, impact its performance negatively. Nonetheless, our analysis of the ADP approach should inform future research in adaptive planning in mining.

Review of non-adaptive approaches to mine planning. Production planning in open pit mining has been studied for several decades. Early approaches consisted in Linear Programming relaxations and heuristics; see, e.g., **??**. Rapid developments in the fields of Integer Programming and Mixed Integer and Linear Programming made it possible to devise algorithms that produce optimal or close to optimal production schedules; see, e.g., **???** and the recent work of **?**. The impact of uncertainty in commodity prices and ore-grades on the production scheduling problem only started to be considered in the 2000’s. The initial approaches consisted of heuristics, e.g. **?**, but developments in the theory of stochastic programming and robust optimization in the last two decades, see e.g. **??**, have allowed the development of approaches with theoretical properties, see e.g. **??**. In particular, the last two approaches consider uncertainty in the mineral content and use *wait and see* decision schemes, where some of the decisions are taken only after the actual grades are revealed. In this regard, to the best of our knowledge, ours is the first approach where observed grades are used to *learn* or *update the belief* of grades in unexploited parts of the mine.

As far as adaptive optimization or optimization with learning is concerned, a fairly comprehensive reference is **?**, extending mostly from Bayesian approaches for the problem of *ranking and selection.* More recently, **?** deepens in the direction of Bayesian optimization with Gaussian process regression;

see, e.g., Section 3.1 of that paper for a discussion covering the covariance functions we use in our work.

2 Background: ore grade modeling, simulation and update

In this section we give a brief overview on how mineral contents, or ore grades, are usually modeled and simulated in mining. We start in Section 2.1 with a typical model for mineral deposits, and then in Sections 2.2 and 2.3 we show how ore grades are estimated and simulated, respectively. We refer the reader to **?** for a comprehensive reference on ore-body modeling and geostatistics.

2.1 A model for mineral deposits

Drill-holes. When a mine is being planned, *samples* (understood as *probes* or *tests*) of the actual mineral content (ore grade) in the mine are obtained from diamond drill-holes. These holes are "cylinders" drilled from the surface through the rock on a regular grid (say $50m \times 50m$); in that way cylinders of rock $5cm$ in diameter, called *cores*, are extracted. They are divided into $10m$ lengths which are split in half along the length; one half is crushed and sent to a laboratory for analysis and the other half is retained for future reference. Each $10m$ long cylinder is called a *sample* in mining vernacular, and is thought of as an "atomized" observation of a point located, by convention, midway down the section of the core; the idea is that the volume of the cylinder is very small compared to that of the deposit.

Block model. For mine planning and scheduling, the orebody is divided into blocks typically of size $10m \times 10m \times 10m$. In order to extract a block, the blocks in a cone above it must already have been removed. To mine the blocks, blast holes $20cm$ in diameter typically on a $5m \times 5m$ grid are drilled right through the block; they are packed with explosives which are detonated to break the block into small enough pieces to be taken either to the processing plant or to the waste dump. As the rock comes out of the blast-holes, a representative sample is taken and analysed. This grade is considered as the "true" grade of the block.

Geostatistical model of the orebody. Let $\mathcal{V} \subseteq \mathbb{R}^3$ be a 3D volume containing the orebody, and for $x \in \mathcal{V}$ let $Z(x)$ denote the random ore grade at point x. A usual modeling assumption in geostatistics (e.g. for copper deposits), is that ore grades are *second-order stationary*, that is, the first two moments of $Z(\cdot)$ (mean, variance and spatial covariance) are invariant under translation.[1]

[1] This is reasonable for most deposits but not all. For example, diamond pipes have radial symmetry and are richer in the center and poorer on the outside. Similarly the impermeable dome on top of most oil reservoirs is curved so the depth to its surface cannot be treated as second order stationary because the mean depth varies.

That is, for all $x, y \in \mathcal{V}$ one has that

$$\begin{aligned} \mathbb{E}\left[Z(x)\right] &=: M \\ \mathbb{E}\left[\left(Z(x)-M\right)\left(Z(y)-M\right)\right] &=: C(x-y). \end{aligned} \tag{1}$$

A second-order stationary process is then characterized by the mean value M, and a *spatial covariance* function $C(\cdot)$. From above, for a given any set of points $x_1, \ldots, x_n$, the covariance matrix $\mathbf{C}_{n\times n}$ corresponding to the random variables $Z(x_1), \ldots, Z(x_n)$ is given by

$$\mathbf{C}_{ij} \;=\; \mathrm{Cov}(Z(x_i), Z(x_j)) \;=\; C(x_j - x_i) \;=\; C(x_i - x_j).$$

Thus, spatial covariance functions C must be *positive definite*[2]. The most commonly used spatial covariances in 3D are the exponential model, the spherical model and the *pure nugget* effect. The corresponding covariance functions are

$$C(h) = \begin{cases} C_0 \exp(-|h|/a) & \text{(exponential)} \\ C_0(1-(3/2)|h|/a + (1/2)(|h|/a)^3)\mathbf{1}\{|h|<a\} & \text{(spherical)} \\ C_0\mathbf{1}\{h=0\} & \text{(pure nugget)}, \end{cases}$$

where $\mathbf{1}\{\cdot\}$ denotes the indicator function, and C_0 is a constant, called the *sill* (a scale factor). It can be shown (see, e.g., **?**) that these functions are indeed positive definite. Note also that, for these models, $\mathrm{Var}\left[Z(x)\right] = C_0$.

In the optimization methods in Section 3, we represent the mine by a set of blocks B. In a slight abuse of notation, we let $Z(x_b)$ denote the (random) ore grade associated with block $b \in B$, where x_b denote a representative point inside block b. Let $\mathcal{V}(b) \subseteq \mathcal{V}$ denote points included in block b; we compute $Z(x_b)$ as the average of the grades at all the points in b, i.e.

$$Z(x_b) \;:=\; \frac{\int_{\mathcal{V}(b)} Z(s)\,\mathrm{d}s}{\mathrm{Volume}(\mathcal{V}(b))}.$$

Note that when ore grades are second order stationary, the mean grade of any block equals M, but its spatial covariance is smaller than that given by $C(\cdot)$. Next, we explain how to estimate and sample the ore grade $Z(x_b)$ of a block $b \in B$ using only a finite number of previous measurements.

2.2 Geostatistical estimation: kriging

Gaussian process regression, or *Kriging*, is a standard method in geostatistics for estimating grades (at points or of blocks); see, e.g. **?**. In such a method, suppose that we have sampled grades at n points, $x_1 \ldots x_n$, thus we observe $z(x_1) \ldots z(x_n)$, the realizations of the grades $Z(x_1)$, …, $Z(x_n)$; we would like to use this information to compute an estimate $\widehat{Z}(x)$ of the grade $Z(x)$ at

[2] A function $f : \mathbb{R} \mapsto \mathbb{R}$ is *positive definite* if f is even (i.e. $f(x) = f(-x)$) and, for any $x_1, \ldots, x_n \in \mathbb{R}$, the matrix $A_{n\times n}$ defined as $A_{ij} = f(x_i - x_j)$ is positive semidefinite.

another point x. The Kriging methods uses a linear combination of the sample grades, i.e.

$$\widehat{Z}(x) := \sum_{i=1}^{n} \lambda_i \, z(x_i).$$

The (kriging) weights $\lambda_1 \dots \lambda_n$ above depend on the location x relative to the locations of the samples $x_1 \dots x_n$ and are chosen so as to obtain a minimum variance unbiased estimator. Two cases have to be considered depending on whether the true mean M is known or not.

When the mean M is unknown, the procedure is called *ordinary kriging* and the kriging weights satisfy the so-called *kriging system*:

$$\begin{bmatrix} C(x_1-x_1) & C(x_1-x_2) & \dots & C(x_1-x_n) & 1 \\ C(x_2-x_1) & C(x_2-x_2) & \dots & C(x_2-x_n) & 1 \\ \dots & \dots & & \dots & 1 \\ C(x_n,x_1) & C(x_n-x_2) & \dots & C(x_n-x_n) & 1 \\ 1 & 1 & \dots & 1 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \\ \mu \end{bmatrix} = \begin{bmatrix} C(x_1-x) \\ C(x_2-x) \\ \vdots \\ C(x_n-x) \\ 1 \end{bmatrix} \tag{2}$$

where λ_i is i-th kriging weight, and μ is a Lagrange multiplier. Note that the value of each λ_i depends on x, so we could write (as we shall do later) $\lambda_i(x)$. The row and column of "1s" correspond to the Lagrange multiplier μ which filters out the unknown mean M.

On the other hand, when the mean M is known (as it is when simulating, see Section 2.3), the procedure is called *simple kriging.* The simple kriging estimator $\widehat{Z}(x)$ is given now by

$$\widehat{Z}(x) := M + \sum_{i=1}^{n} \lambda_i (z(x_i) - M),$$

where the kriging weights λ_i solve the kriging system (2), after setting $\mu = 0$ (in which case is known as the *simple* kriging system).

When a very large number of samples is available, it may be difficult to invert the covariance matrix on the left-hand side of (2). In that case, only the data within a moving neighbourhood near the point or block are used. See **?** for more information.

2.3 Geostatistical Scenario Sampling

2.3.1 Unconditional Scenario Sampling

Non-adaptive optimization methods, which our proposed policy uses, typically require sampling grades for all unexploited blocks (see Section 3). Said samples must (asymptotically) reproduce conditional (on available data) means and spatial covariance, so as to ensure the asymptotic optimality of such methods.

As mentioned in Section 1, traditional sampling methods can not be used in practical instances of the problem, because of the large number of blocks in said instances. This, despite the availability of closed-form expressions for the blocks' unconditional and conditional grade joint distributions. Next, we use key properties of the Gaussian distribution[3] to produce sample grades efficiently.

For the sake of exposition, we assume that the ore grade process is a Gaussian field, so that for any finite set of points, $x_1 \dots x_n$, the joint distribution of their grades $Z(x_1) \dots Z(x_n)$ is MVN. We also assume that the marginal distribution for each point is the standard normal $N(0, 1)$, and that the spatial covariance function $C(\cdot)$ is known.

In practice, ore grades are not normally distributed (e.g. ore grades are non-negative). Thus, assuming that the marginal distribution of $Z(x)$ is available (i.,e. has been fit into a model using experimental data), we apply a transformation, so that distribution of such transformation is indeed standard normal.[4] We compute the experimental spatial covariance of the grades (or, in practice, of its transformation) and fit a model to it (for instance, the exponential, or spherical models described earlier). With the function $C(\cdot)$ at hand, we can compute the covariance matrix for the grades of any finite set of points $x_1 \dots x_n$ via equation (1).

With the parameters of the multivariate normal distribution at hand, we can then sample the random vector $\mathbf{Z} = \{Z(x_1) \dots Z(x_n)\}$, for any finite set of points $\{x_1 \dots x_n\}$ (called an unconditional sample). This can be done by an LU decomposition (Cholesky) of the covariance matrix if it is small enough; that is, if n is not too large. Otherwise, we use the turning bands method [**?**], which involves generating suitably chosen 1D processes (which depend on the covariance model) along a set of 15 or more bands (lines) equally spread in space. The projection from each band to each target point is computed and these are then summed. We refer the reader to **???** for details. The advantage of this method are that it reduces 3D problems into one-dimensional problems, and also is capable of sampling grades from models with a very large number of points.

2.3.2 Conditional scenario sampling

The next step is to produce sample grades conditional on the available data. To set the stage, consider a finite set of points $\{x_1 \dots x_n\}$, and let $D \subset \{1 \dots n\}$ denote the subset of indices of points for which we have information, either from drill holes or extracted blocks. (That is, grade realizations $z(x_i), i \in D$ have been observed.)

Let $I \subseteq D^c$ denote the (index) set of points we would like to sample (ore) from, conditional on (the information from points in) D, and let $\widehat{Z}(x_i|D)$

[3] Other simulation methods have been developed, e.g., sequential simulations, see **?**, **?**, **?**, and multi-point simulations, see **?**, **?**.

[4] This is, instead of working directly with grade $Z(x)$, we define the function $f(z) = \Phi^{-1}(F(z))$ (which is well-defined provided that F, the distribution function of the grade, is increasing), so that $P(f(Z) \leq z) = P((Z) \leq F^{-1}\Phi(z),) = \Phi(z)$, so $Y(x) \equiv f(Z(x))$ is normally distributed. (Here, Φ denotes the standard normal distribution.)

denote the estimate of $Z(x_i)$, conditional on D, $i \in I$. Suppose that we have access to an unconditional sample $\{z'(x_1) \dots z'(x_n)\}$ of $\{Z(x_1) \dots Z(x_n)\}$ (see prior section). Then, the conditional estimate $\widehat{Z}(x_i|D)$, $i \in I$, is given by

$$\widehat{Z}(x_i|D) = z'(x_i) + \sum_{j \in D} \lambda_j(x_i)(z(x_j) - z'(x_j)). \tag{3}$$

where $\{\lambda_j(x_i), j \in D\}$ denote the solution to the kriging system (2), solved at point x_i, using the input $\{x_j, j \in D\}$). Note that the conditional estimate is such that difference between the estimated grade and its unconditional sample equals the weighted (using the kriging weights) sum of the same differences across the points in D. Moreover, note that the last term in (3) corresponds to the difference in the krige estimates when using inputs $\{z(x_j), j \in D\}$ and $\{z'(x_j), j \in D\}$. See **?** for details. In summary, given ore data for points in D (drill holes and extracted blocks), and a set of points I for which we want to produce grade estimates, we

1. Generate an unconditional sample $(z'_i, i \in I \cup D)$ of (transformed) grades using the turning bands method.
2. Compute kriging weights by solving (2), for each target point $i \in I$, using $\{x_j,\ j \in D\}$ as input (because the transformed grades have known mean, we set $\mu = 0$).
3. Given the observed (transformed) grades $\{z(x_j),\ j \in D\}$ compute the conditional estimates using equation (3). (Alternatively compute the difference between kriging estimates using unconditional sampled and observed input)
4. Un-transform the conditional kriging estimates to obtain conditional grade estimates at the target points.

3 Optimization models

In this section we propose an adaptive production scheduling policy for the mine planning problem under geological uncertainty. To make the model clear, we give a gradual exposition where we first present a simplified bare-bones version of the multi-stage mining problem with no stochasticity or adaptivity. Then in Section 3.1 we present a non-adaptive version of the problem when the ore grades are uncertain; and finally, in Section 3.2, we present the proposed adaptive policy. Our methodology is tailored to the specific geostatistical models frequently used to represent the geological uncertainty of the mineral contents in mines; in particular, we use the model for geological uncertainty discussed in Section 2 to estimate, simulate and update the ore grades in our optimization policy.

Simplified model with no stochasticity or adaptivity. Consider the following simplified version of the mine planning problem, when grades are deterministic. The objective is to maximize the economic value from exploiting the mine over a (finite) set T of time periods. The ore-body is discretized into a finite set of n of blocks, which are clustered into a partition $\mathcal{B} := \{B_1 \dots B_m\}$ (here, m denotes the number of clusters); that is, a cluster is a subset of blocks, and clusters are such that each block belongs to a cluster, and no two clusters share blocks.

At each time period $t \in T$, all blocks in a cluster $B_i \in \mathcal{B}$ are extracted, or none is. We let binary variable $\theta_{i,t}$ denote such an extraction decision, i.e. $\theta_{i,t} = 1$ if all blocks in cluster B_i are extracted, and $\theta_{i,t} = 0$ if none is. (In the sequel, we refer to a cluster by its index). If cluster i is extracted during period $t \in T$, then each block $b \in B_i$ can be processed, either partially or completely. Let continuous variable $y_{b,t} \in [0, 1]$ denote the portion of block b that is processed in period t.

A cluster can be extracted in at most one time period, and extraction decisions are subject to a series of precedence rules, summarized in the set P, such that $(i, j) \in P$ if in order to extract cluster i in period t, then one must extract cluster j prior or during period t.

At each time period there is a finite total extraction capacity K^{ex}; extracting cluster i on any given period requires k_i^{ex} units of capacity. Similarly, each period has a total processing capacity of K^{pr}; processing whole block b requires k_b^{pr} units of such capacity (we assume that required capacity scales linearly with the processing decision). Lastly, we assume that there is a cost $w_{i,t}$ associated with extracting cluster i at time t, and that processing block b at time t generates economic value equals to $v_{b,t}$ (scales linearly).[5] Note that such an economic value depends on the ore grades, which in this model are deterministic.

Considering the above, a simplified version of the mine planning problem is given by

$$
\begin{aligned}
\max \quad & \sum_{t\in T} \left[\sum_{b\le n} v_{b,t}\, y_{b,t} - \sum_{i\le m} w_{i,t}\, \theta_{i,t} \right] && \\
\text{s.t.} \quad & \sum_{t\in T} \theta_{i,t} \le 1 && \forall\, i \le m && (4)\\
& \theta_{i,t} \le \sum_{s\le t} \theta_{j,s} && \forall\, (i,j) \in P, \quad t \in T && (5)\\
& \sum_{i\le m} \theta_{i,t}\, k_i^{ex} \le K^{ex} && \forall\, t \in T && (6)\\
& y_{b,t} \le \theta_{i,t} && \forall\, i \le m, \quad b \in B_i, \quad t \in T && \\
& \sum_{b\le n} y_{b,t}\, k_b^{pr} \le K^{pr} && \forall\, t \in T && \\
& \theta_{i,t} \in \{0,1\} && \forall\, i \le m, \quad t \in T && (7)
\end{aligned}
$$

[5] Economic value from processing might include discount factors, operational costs, commodity prices, mineral ore grades, etc. per cluster or block. For clarity of exposition we leave them as general parameters that represent the value of the operation.

$$0 \leq y_{b,t} \leq 1 \qquad\qquad \forall\, b \leq n, \quad t \in T.$$

3.1 Non-adaptive stochastic programming problem

We consider now a setting where ore grades are stochastic. For that, we assume that a model for ore grade uncertainty is available. That is, we assume that the joint distribution of the ore grades, at the block level, is available. In formulating the problem, we build upon the simplified optimization problem shown above, and consider the *two-stage stochastic programming* formulation in **?**. Such a formulation is non-adaptive, as it does not take into account any information update based on new data that may become available as the mine is exploited. Because such a formulation is scenario-based, we assume that there exists an efficient procedure to generate samples from the joint ore grade distribution. As detailed in Section 2, such a procedure is indeed available for Gaussian-based uncertainty models, which are quite standard in geostatistics.

Ir order to formulate the problem, we make two assumptions. First, we assume that a block's ore grade does not affect its extraction value (although, we allow it to affect the value of processing the block). Second, we assume that, once a block is extracted, its true ore grade is immediately observed, and that such data can inform the processing decision (i.e., how much of the block to process).

From above, one can formulate a *two-stage* stochastic program, where in the first stage extraction decisions are made for all periods, and processing decisions are relegated to the second stage. Thus, in this formulation, extraction decisions are made *before* ore grades are observed, and processing decisions are made *after* the uncertainty is realized, as grade uncertainty is resolved once blocks are extracted. In terms of the formulation in the previous section, we now have that the economic value associated with processing block b in period t is a random variable, which we denote by $V_{b,t}$.

Defining $\boldsymbol{\theta} := \{\theta_{i,t} : i \leq m,\, t \in T\}$ and $\mathbf{V} := \{V_{b,t} : b \in B, t \in T\}$, we can formulate the (non-adaptive) stochastic program (nA-SP) proposed in **?**,

$$\begin{aligned} \text{(nA-SP)} \quad \max \quad & \mathbb{E}\left[Q(\boldsymbol{\theta}, \mathbf{V})\right] - \sum_{t \in T} \sum_{i \leq m} w_{i,t}\, \theta_{i,t} \\ \text{s.t.} \quad & (4),\ (5),\ (6) \text{ and } (7) \end{aligned}$$

where

$$\begin{aligned} Q(\boldsymbol{\theta}, \mathbf{V}) := \ \max \ & \sum_{t \in T} \sum_{b \in B} V_{b,t}\, y_{b,t} \\ \text{s.t.} \ & y_{b,t} \leq \theta_{i,t} && \forall\, b \in B_i, \quad i \leq m, \quad t \in T \\ & \sum_{b \leq n} y_{b,t}\, k_b^{pr} \leq K^{pr} && \forall\, t \in T \\ & 0 \leq y_{b,t} \leq 1 && \forall\, b \leq n, \quad t \in T. \end{aligned}$$

Note that the term $\mathbb{E}\left[Q(\boldsymbol{\theta}, \mathbf{V})\right]$ depends in a non-trivial fashion on the distribution of $\mathbf{V}$. For that reason, the standard approach to solving the formulation above is to use a Sample Average Approximation (SAA) (see, e.g., **?** and **?** for overviews of such an approach). Suppose that we have a set $\{v^s : s \leq S\}$ of S independent samples (also called *scenarios*) from $\mathbf{V}$. Such samples might be obtained, for example, by using the unconditional simulation procedure described in Section 2.3.1 to sample ore grades, and use those to compute economic values. The SAA approach consists on approximating the $\mathbb{E}\left[Q(\boldsymbol{\theta}, \mathbf{V})\right]$ by its sample mean, i.e.

$$\mathbb{E}\left[Q(\boldsymbol{\theta}, \mathbf{V})\right] \approx \frac{1}{S} \sum_{s=1}^{S} Q(\boldsymbol{\theta}, v^s).$$

Note that the number of variables and constraints in the resulting formulation depend on the number of samples considered, and thus, might result in prohibitively long solution times using standard commercial solvers. Such a formulation, however, might be solved by specialized algorithms, for example, Benders-type decomposition algorithms that separates the extraction decisions (first-stage decisions) from the processing decisions for each scenario s (second-stage decisions). See also **?** for a decomposition algorithm based on the Bienstock-Zuckerberg algorithm.

3.2 Adaptive rolling-horizon policy

In this section we present our adaptive policy for the production optimization in open-pit mining. Our starting point is the non-adaptive stochastic program nA-SP in the previous section. Using such a formulation, we propose a rolling-horizon policy that, at each time period, updates the ore-grade distribution, and generates new scenarios, based on the latest ore-grade observations, and then solves an instance of nA-SP to set extraction decisions for the current period.

At a general level, the key ingredients of the procedure are: an efficient method for updating ore-grade distributions of non-extracted blocks, conditional on ore-grades realizations for extracted blocks; and an efficient method to solve the non-adaptive version of the problem. In particular, regarding the first of these ingredients, we consider a Gaussian-based model for ore-grade uncertainty, and thus, we can use the procedure discussed at the end of Section 2.3.2 to generate conditional ore-grade samples. Regarding the second ingredient, we formulate the mine planning problem so that its non-adaptive formulation corresponds to the two-stage stochastic program nA-SP, for which there exists efficient (specialized) algorithms.

The adaptive policy we propose can be easily described as follows.

Algorithm 1: A-RH policy

```
Generate a set of $S$ ore-grade scenarios, conditioned on drill-holes data,
  using the procedure in Section 2.3.1.
for $t = 1 \in T$ do
    1. Formulate and solve nA-SP with the extraction and processing decisions
       for periods $s < t$ fixed.
       Result: Set extraction decisions for period $t$ according to the solution to
       nA-SP. In particular, extract clusters for which $\theta_{i,t} = 1$.
    2. Observe ore grades for blocks in clusters extracted in period $t$, i.e.,
       observe $v_{b,t}$ for $b \in B_i$ for all $i$ such that $\theta_{i,t} = 1$.
    3. Set processing decisions for period $t$ according to the solution to
       $$\max \left\{ \sum_{i:\theta_{i,t}=1} \sum_{b \in B_i} v_{b,t}\, y_b : \sum_{i:\theta_{i,t}=1} \sum_{b \in B_i} y_b\, k_b^{pr} \le K^{pr},\ 0 \le y_b \le 1\ \forall\ b \le n \right\}.$$
    4. Update the set of $S$ ore-grade scenarios, conditioned on drill-holes data
       and observed ore grades from blocks in clusters extracted so far, using the
       procedure in Section 2.3.2.
end
```

Note that the optimization problem in step 3 above adapts processing decisions to the ore grades observed for extracted blocks. This is necessary because the solution to nA-SP might not include said observed grades in any of the scenarios considered in said formulation. In this regard, the maximization in step 3 is a sub-problem embedded in nA-SP. It is important to note that, because of the continuous nature of its decision variables, it can be solved efficiently by a greedy algorithm.

The resulting policy adapts extraction and processing decisions to the latest observations of the ore-grades, by using the (non-adaptive) stochastic program (nA-SP) in a rolling horizon manner. Consequently, we call this the *adaptive rolling-horizon* policy and denote it by (A-RH).

Remark 1 As mentioned before, the complexity of solving nA-SP depends on the size of the instance, and the number of scenarios considered. Thus, in practical instance, solution times for practical instances of the problem are typically measured in hours. In addition, the complexity of sampling conditional ore-grades depend on the number of non-extracted blocks as well on that of extracted blocks, and drill holes (in addition to various other tuning parameters). Thus, as our numerical experiments show, running times for generating conditional samples might also go as high as an hour, for practical instances of the problem. In this regard, it is important to note that the typical length of a time period in mine planning is a calendar year (with time horizons going as high as 25 years).

4 A one-step look-ahead policy

A major drawback of the solution prescribed by nA-SP is that extraction decisions are made before the beginning of the planning horizon, and are not revisited as more information (on ore grades) is gathered. It stands to reason that policy performance can be improved by adapting decisions to such information, which is precisely the motivation behind the rolling-horizon policy discussed in Section 3.2. It is important, though, to note that such a rolling horizon scheme, while adapting to upcoming information, does not anticipate the fact that extraction and processing decisions will be revisited in the future, as more information is collected.

In order to find the optimal policy, which anticipates the effect adapting future decisions to upcoming information, one can formulate the mine planning problem as a dynamic programming (DP) formulation. A conceptual scheme of such a formulation is the following: at time t extraction and processing decisions come from solving

$$\begin{array}{l}\max\ \mathbb{E}\left[\substack{\text{ext. \& proc.}\\ \text{cost, year } t}\right]\\ \text{s.t. }\left(\substack{\text{prec. \& cap.}\\ \text{constraints, year } t}\right)\end{array} + \mathbb{E}\left[\begin{array}{l}\max\ \mathbb{E}\left[\substack{\text{ext. \& proc.}\\ \text{cost, year } t+1}\right]\\ \text{s.t. }\left(\substack{\text{prec. \& cap.}\\ \text{constraints, year } t+1}\right)\end{array} + \mathbb{E}\left[\begin{array}{l}\max\ \mathbb{E}\left[\substack{\text{ext. \& proc.}\\ \text{cost, year } t+2}\right]\\ \text{s.t. }\left(\substack{\text{prec. \& cap.}\\ \text{constraints, year } t+2}\right)\end{array} + \ldots \mathbb{E}[\cdots]\right]\right].$$

In the DP formulation above, expectations are taken with respect to the conditional distribution of ore grades, where the conditioning is with respect to the information made available from the blocks that have already been extracted. This is a case of a sequential decision-making problem under uncertainty with *decision-dependent information uncertainty*, following the terminology proposed by **?**. Models of this type are notoriously hard to solve — typically, one needs to resort to integer variables to model the dependence on decisions [**??**]. In what follows, we propose a policy based on approximate dynamic programming (ADP) approach. In particular, we consider a *one-step look-ahead* policy, which is a standard approach in the ADP literature; see for example **?**. The main idea of such a policy is the following: at each period t, we make extraction and processing decisions considering that on period $t+1$ such decisions might be revisited and adapted (for the last time) to the information collected during period t. In terms of the scheme illustrated above, the one step look ahead policy is given by

$$\begin{array}{l}\max\ \mathbb{E}\left[\substack{\text{ext. \& proc.}\\ \text{cost, year } t}\right]\\ \text{s.t. }\left(\substack{\text{prec. \& cap.}\\ \text{constraints, year } t}\right)\end{array} + \mathbb{E}\left[\begin{array}{l}\text{value of nA-SP}\\ \text{starting at}\\ \text{period } t+1\end{array}\right].$$

Note that, the expectation in the second term in the formulation above is taking with respect to the ore grades of blocks extracted in period t. Thus, decisions made for period t factor in the fact that decisions made in period $t+1$ depend on the information revealed during period t. Next, we describe this one-step look-ahead policy formally.

4.1 An Approximate Dynamic Programming Formulation

Consider the mine planning problem starting at period $t \in T$, and let D_t denote the set of clusters already extracted prior (and including) to time t. Similarly, let $\mathcal{V}_t$ denote the set of blocks/economic-value pairs extracted/observed prior to period t, so that $(b, v) \in \mathcal{V}_t$ implies that block b was extracted prior to period t, and its observed economic value was v.

We can write the NPV of the operations (starting in period t) as the sum of: (i) the extraction cost in period t, (ii) the expected revenue from processing blocks in period t; and (iii) the expected profit of subsequent periods conditioned to the information gained. Thus, we get the following dynamic programming formulation

$$J_t(D_t, \mathcal{V}_t) = \max_{I \in \mathcal{I}(D_t)} \left\{ \mathbb{E}\left[F_t(I, \mathbf{V})|\mathcal{V}_t\right] - \sum_{i \in I} w_{i,t} + \mathbb{E}\left[J_{t+1}(D_t \cup I, \mathcal{V}_{t+1})|\mathcal{V}_t\right] \right\}, \tag{8}$$

where we recall $\mathbf{V}$ denotes the random vector of economic values (across blocks and periods). In the above, $\mathcal{I}(D)$ denotes the collection of subsets of clusters that can be extracted in a period if clusters in D have already been extracted in the past, i.e,

$$\mathcal{I}(D) := \left\{ I \subseteq D^c : \sum_{i \in I} k_i^{ex} \leq K^{ex}, \bigcup_{i \in I} \{j : (i,j) \in P\} \subseteq I \cup D \right\},$$

and $F_t(I, v)$ denotes the optimal profit from processing blocks from the clusters in I when block economic values' are given by v, i.e.

$$F_t(I, v) := \max \left\{ \sum_{i \in I} \sum_{b \in B_i} v_{b,t}\, y_b : \sum_{i \in I} \sum_{b \in B_i} y_b\, k_b^{pr} \leq K^{pr},\ 0 \leq y_b \leq 1 \right\}.$$

Note that this is indeed the optimization problem solved in step 3 of Algorithm 1.

Formulating (and solving) (8) is very hard in general: because of its recursive nature, backward induction is typically required, and the numerical complexity is tied to the size of the state space, which in this case is uncountable. For these reasons, we combine a SAA approach to approximating expectations, and a one-step look-ahead approach to approximate the continuation value (starting on period $t+1$) to obtain an approximate solution. This is, we solve

$$\widetilde{J}_t(D_t, \mathcal{V}_t) = \max_{I \in \mathcal{I}(D_t)} \left\{ \frac{1}{S} \sum_{s \leq S} F_t(I, v^s) - \sum_{i \in I} w_{i,t} + \frac{1}{S} \sum_{s \leq S} \text{nA-SP}(D_t \cup I, v^s, t+1) \right\}, \tag{9}$$

where $\{v^s : s \leq S\}$ is a set of S independent samples from $\mathbf{V}$, conditional on the ore grades of blocks in clusters in D_t, and nA-SP(D, v, t) denotes the solution to nA-SP, starting in period t when clusters in D have already been extracted, and ore grades are given by v. That is,

$$
\begin{aligned}
\text{nA-SP}(D, v, t) = \max\ & \frac{1}{S} \sum_{s \le S} Q(\boldsymbol{\theta}, v^s, t) - \sum_{u \ge t} \sum_{i \le m} w_{i,u}\, \theta_{i,u} \\
\text{s.t.}\ & \sum_{u \ge t} \theta_{i,u} \le 1 \qquad \forall\, i \le m \\
& \theta_{i,u} \le \sum_{u' \le u} \theta_{j,u'} \qquad \forall\, (i,j) \in P, \quad u \ge t \\
& \sum_{i \le m} \theta_{i,u}\, k_i^{ex} \le K^{ex} \qquad \forall\, u \ge t \\
& \theta_{i,u} \in \{0,1\}, \quad \forall\, i \le m, u \ge t, \quad \theta_{i,t-1} = 1, \quad \forall i \in D,
\end{aligned}
$$

where

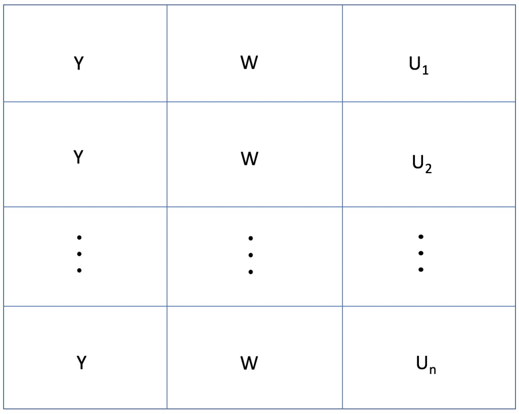
$$
\begin{aligned}
Q(\boldsymbol{\theta}, v, t) :=\ \max\ & \sum_{u \ge t} \sum_{b \in B} v_{b,u}\, y_{b,u} \\
\text{s.t.}\ & y_{b,u} \le \theta_{i,u} \qquad \forall\, b \in B_i, \quad i \le m, \quad u \ge t \\
& \sum_{b \le n} y_{b,u}\, k_b^{pr} \le K^{pr} \qquad \forall\, u \ge t \\
& 0 \le y_{b,u} \le 1 \qquad \forall\, b \le n, \quad u \ge t.
\end{aligned}
$$

and, as before, $\{v^s : s \le S\}$ is a set of S independent samples from $\mathbf{V}$, conditional on the ore grades of blocks in clusters in D.

A key observation about the formulation above is that the set of S samples used in formulating $Q(\boldsymbol{\theta}, v, t)$ depends on the set D (equivalently, on the extraction decision in period t). For this reason, it is not possible solve nA-SP$(\cdot)$ directly as stated above using regular integer programming techniques. We sidestep this difficulty by solving nA-SP$(\cdot)$ via complete enumeration. In this regard, note that, precedence and capacity constraints considerably limit the possible extraction decisions in practical settings, especially early in the planning horizon. Thus, for a fixed extraction decision $\boldsymbol{\theta}_t$, it is possible to generate S ore grade samples, conditional on the information available from all extracted clusters. While the procedure is computationally intensive, our numerical results show reasonable solution times are achievable. The effort is, of course, justified as long as the ADP approach is able to improve upon the performance of the A-RH policy. We explore this issue next.

4.2 Illustration of the theoretical policy performance

To illustrate the theoretical advantage of an ADP approach, we consider a small stylized instance of the mine planning problem. With this example, we show that the ADP-based policy might perform arbitrarily better than the A-RH policy. The mine associated to such an instance is illustrated in Figure 1. The diagram represents a mine with three columns of n blocks each. The symbols inside the blocks correspond to random variables representing the Gaussian (transformed) ore grades. As can be seen from the figure, all blocks in the first column are identical (with an ore grade given by Y) and all blocks in the second column are identical, with an ore grade W. Suppose now that $W = 2\mu - Y$, where $\mu = \mathbb{E}[Y]$. This implies that $\mathbb{E}[W] = \mu = \mathbb{E}[Y]$, $\mathrm{Var}\,[W] = \mathrm{Var}\,[Y] = \sigma^2$ and $\mathrm{Corr}(Y, W) = -1$. Suppose in addition that $U_1, \ldots, U_n$ are i.i.d. with mean $\mathbb{E}[U_i] = \eta > \mu$, and that each U_n is independent of Y and W. There are n periods, and the capacity for extraction and processing is one block per period. Finally, to complete the description of the model, suppose that (i) all blocks have the same extraction cost w, (ii) each cluster contains exactly one block, and (iii) the value of each block b at time t is equal to its (transformed) ore grade.



Y	W	U_1
Y	W	U_2
⋮	⋮	⋮
Y	W	U_n

Fig. 1 An artificial example of a mine with $3n$ blocks.

Consider first the solution prescribed by the nA-SP formulation in Section 3.1: one can see that its optimal solution is to extract the n blocks in the third column (one per period), which yields an optimal value equal to $\mathbb{E}[\sum_{t=1}^{n} U_t] - \sum_{t=1}^{n} w = n(\eta - w)$.

Consider now the A-RH policy: because extracting a block in the third column does not give any new information about the remaining blocks (due to independence), the solution prescribed is that prescribed by the nA-SP formulation, thus achieving the same performance.

Consider now the ADP model (9) applied to this example: in the first period, the decision is whether to extract the top block from the first, second, or third columns. Let us consider the three cases separately.

<u>Case 1</u>*: Extract a block from the third column.* In this case, the immediate contribution to the objective function is $U_1 - w$. Moreover, since no information

is acquired, the calculation for the approximating value function is identical to the two-stage case, so the objective function value corresponding to this solution is given by

$$f_1 = \mathbb{E}\left[U_1 - w\right] + (n-1)(\eta - w) = n(\eta - w).$$

Case 2: *Extract a block from the first column.* Suppose its grade is y. Then, the immediate contribution to the objective function is $y - w$. Because blocks in the first column are identical a.s., after observing the grade y of the extracted block, all the blocks below that one have mean y and variance zero. Similarly, all blocks in the second column have mean $\mu - (y - \mu) = 2\mu - y$ and variance zero. Finally, due to independence, the distribution of blocks in the third column is not affected. Therefore, if $y > \eta$ the two-stage problem inside the approximating value function will extract the remaining $n-1$ blocks from the first column, whereas if $2\mu - y > \eta$ it will extract the top $n-1$ blocks from the second column. It follows that in this case, the objective function associated with this solution is given by

$$f_2 \;=\; \mathbb{E}\left[Y - w + (n-1)\left(Y1_{\{Y>\eta\}} + (2\mu - Y)1_{\{2\mu - Y > \eta\}} + \eta 1_{\{\max(Y, 2\mu - Y) \leq \eta\}} - w\right)\right]$$

Define

$$\varDelta := \mathbb{E}\left[\left(Y1_{\{Y>\eta\}} + (2\mu - Y)1_{\{2\mu - Y > \eta\}} + \eta 1_{\{\max(Y, 2\mu - Y) \leq \eta\}}\right)\right] - \eta.$$

Because the sets $\{Y > \eta\}$ and $\{2\mu - Y > \eta\}$ have positive probability (Y is normally distributed), we have that $\varDelta > 0$ (in the strict sense). It follows that, for any $n > 1 + \frac{\eta - \mu}{\varDelta}$ we have that

$$f_2 = \mu - w + (n-1)(\eta + \varDelta - w) > n(\eta - w) = f_1.$$

Case 3: *Extract a block from the second column.* Following the same steps as in Case 2 (reversing the roles of Y and W), we have that $f_3 = f_2$, and thus for sufficiently large n we have $f_3 > f_1$.

From above, we see that the ADP model attains an objective value equal to $f_3 = f_2 > f_1$. Because f_1 equals the objective function value for both the nA-SP and A-RH approaches, for this example the ADP approach yields a better solution. Moreover, we have that the difference in performance can be made arbitrarily large by taking n large enough.

5 Computational Results

In this section we test (numerically) the performance of the policies described so far. For this, we use instances from mines of several sizes, ranging from artificially small (120 blocks) to realistically-sized mines of more than 100,000 blocks. The covariance structure considered consists of the weighted sum of a spherical structure (0.45), an exponential structure (0.45) and nugget effect

(0.1). Such a structure is used by **?** to fit experimental data from copper deposits in Chile.

For each instance, we test policy performance on a set of 100 ore grade samples, drawn from our geostatistical model for uncertainty. For each sample, we compute policy performance for the nA-SP, A-RH and ADP (when possible, more details below) approaches, assuming the "true" ore-grades are as in the sample. In addition, we compute the performance of a *perfect knowledge* (PK) policy, which has upfront knowledge of the true ore-grades (thus, there is no uncertainty) and therefore provides an (probably unattainable) upper bound on performance.

For each instance, and "true" ore grade sample, we proceed as follows. First, a sample for the drill holes grades is generated. Then, conditional on this realization, we generate sample grades for the representative points of all blocks (this produces the "true" scenario). Then, we simulate decisions made by each policy, period by period. We use the number of drill holes as a proxy for the amount of knowledge policies have a priori: we conduct sensitivity analysis on policy performance as a function of the number of drill holes in order to assess the role of uncertainty in the model.

In our experiments, we use a HP ProLiant SL230s Gen8 computer with 20 cores available (2 × Intel Xeon E5-2660 10 cores each). With five of these machines, we were able to parallelize all 100 simulations for each instance.

Instance set-up. We parameterize a mine by constants $h \geq 2$ and ℓ by considering a grid of drill hole grade samples represented by the locations specified in the set DH, defined as follows:

$$DH = \Big\{(x_1, x_2, x_3) = (20i, 5 + 20(j-1), 5 + 10(k-1)) \in \mathbb{Z}^3 \ : \\ i, j \in \{1, \ldots 2^{h-1}\},\ k \in \{1, \ldots, \ell\}\Big\}. \tag{10}$$

In the above, x_3 corresponds to the depth coordinate, and (x_1, x_2) are its locations on a plane parallel to the surface. With this, there are $2^{2(h-1)}$ drill holes. For some experiments, we did not use all the drill holes in DH; in such cases we took a subset of 2^{2r} points in DH $(r = 0, \ldots, h-1)$ in such a way that such points were equally spaced on the (x_1, x_2) plane.

Similarly, the set of blocks is represented by B, which is defined as follows:

$$B = \Big\{(x_1, x_2, x_3) = 5 + 10 \cdot (i-1, j-1, k-1) \in \mathbb{Z}^3 \ : \\ i, j \in \{1, \ldots 2^{h}\},\ k \in \{1, \ldots, \ell\}\Big\}. \tag{11}$$

The total number of blocks in B is $2^{2h}\ell$. In order to ensure the slope stability of the open-pit, the walls must be inclined at less than about 45^o (the precise angle depends on the properties of the rock.), which means that there are effectively fewer blocks at lower levels. In our experiments, we only considered blocks that were feasible for extraction.

We discuss now the structure of the precedence constraints. For that, let us index blocks so that if block b_1 is closer to the surface than block b_2, then $b_1 < b_2$. At the block level, a block cannot be extracted until the blocks above them have been removed. Thus, we first define precedence constraints at the block level as follows

$$P_{Block} = \Big\{(b_1, b_2) \,:\, |x_1^{b_1} - x_1^{b_2}| \leq 10,\, |x_2^{b_1} - x_2^{b_2}| \leq 10,\, x_3^{b_2} = x_3^{b_1} + 10, \\ 1 \leq b_1 < b_2 \leq |B|\Big\}.$$

With this constraints at hand, we define the set P by using a bench-phase design (see, e.g. **?**) considering precedence constraints for adjacent clusters in the North-South and West-East directions, and also in the top-down direction.

5.1 Numerical tests

In our tests, we experiment with different number of drill holes and blocks (by changing the values of h and ℓ), the number of time periods $|T|$ in the horizon, and the number of scenarios S used by the SAA approach. With regard to the extraction and processing capacities, we set the extraction capacity as $K^{ex} = \frac{1}{|T|+1} \sum_{b \in B} \text{weight}_b$, where weight_b denotes the weight of block b, and the processing capacity is set to be half of that amount: $K^{pr} = K^{ex}/2$. Table 1 gives the values of the input parameters we used to run our experiments, and Table 2 presents the results of these runs having a three hours limit to finish all the evaluations, for all policies, given an out-of-sample scenario (for the virtual true grades).

The results in Table 2 are quite striking. We see that, even for the large mines, nA-SP and A-RH policies lead to values that correspond to 95% or more of the upper bound given by the PK policy. Such a percentage increases even further with the number of drill holes. While it is indeed expected that the gap to the PK upper bound goes to zero with the number of drill holes (because uncertainty is reduced), it is still remarkable that even with the information from only one drill hole the gap is small. We also see that the A-RH policy produces even smaller gaps than those obtained with nA-SP. The table also show that, as expected, the problems become more complex as the number of blocks, the number of periods ($|T|$) and the size S of the in-sample increase. However, the results suggest that computation times grow less than linearly in terms of number of blocks and size of the in-sample; the most critical factor seems to be the number of time periods—which is not surprising, as it is known from the literature on multistage stochastic optimization that the complexity of the models grows quickly with the number of stages, in some cases even exponentially [**?**].

id.	h	ℓ	$\|T\|$	S	# Instances Solved to optimality.
Case 1	3	4	5	1	100
Case 2	3	4	10	1	100
Case 3	3	4	5	50	100
Case 4	3	4	10	50	100
Case 5	3	4	5	100	100
Case 6	3	4	10	100	100
Case 7	5	6	5	1	100
Case 8	5	6	5	50	100
Case 9	5	8	5	1	100
Case 10	5	8	5	25	98
Case 11	5	8	5	50	100
Case 12	6	14	5	1	92
Case 13	7	6	5	1	99
Case 14	7	8	5	1	61

Table 1 Parameters used to run numerical tests: h and ℓ determine the size of the mine, $|T|$ and S are the number of periods and scenarios respectively, and the last column has the number of instances solved out of 100.

id.	Number of blocks	*Extraction Capacity	Number of precedences	Number of Clusters	Av No Blocks per Cluster	**nA-SP–A-RH	
						Time–sec.	1−gap

1	120	20.0	32	20	6.0	6.85; 8.08;
						0.50; 1.23;
						0.49 1.23
							0.992; 0.993;
							0.975; 0.975;
							0.955 0.959

2	120	10.91	32	20	6.0	10.13; 18.7;
						4.6; 13.03;
						4.16 12.51
							0.987; 0.988;
							0.966; 0.97;
							0.925 0.955

3	120	20.0	32	20	6.0	8.05; 10.33;
						1.9; 3.71;
						2.2 4.1
							0.999; 1;
							0.989; 0.988;
							0.963 0.973

4	120	10.91	32	20	6.0	56.08; 145.02;
						107.1; 286.91;
						234.77 622.45
							0.995; 0.994;
							0.976; 0.98;
							0.949 0.966

5	120	20.0	32	20	6.0	8.13; 12.22;
						4.52; 8.28;
						4.68 8.68
							0.999; 1;
							0.987; 0.988;
							0.964 0.975

6	120	10.91	32	20	6.0	146.96; 387.05;
						318.83; 789.53;
						677.87 1622.12
							0.992; 0.993;
							0.984; 0.984;
							0.963 0.972

7	4444	740.67	104	48	92.58	49.46; 67.8;
						33.49; 52.7;
						43.29; 64.26;
						61.23; 86.88;
						111.49 138.5
							0.995; 0.995;
							0.993; 0.993;
							0.982; 0.986;
							0.963; 0.977;
							0.948 0.971

8	4444	740.67	104	48	92.58	288.81; 482.63;
						318.65; 534.51;
						369.9; 636.96;
						460.67; 742.29;
						605.41 936.9
							0.997; 0.997;
							0.996; 0.997;
							0.987; 0.989;
							0.964; 0.982;
							0.958 0.98

9	5168	861.33	144	64	80.75	101.88; 135.13;
						83.64; 120.13;
						71.3; 109.79;
						90.38; 121.15;
						70.31 107.68
							0.999; 0.999;
							0.996; 0.996;
							0.980; 0.984;
							0.954; 0.970;
							0.948 0.969

10	5168	861.33	144	64	80.75	222.66; 377.01;
						246.14; 411.4;
						328.09; 517.53;
						1018.12; 1229.98;
						1817.17 2032.93
							0.997; 0.997;
							0.996; 0.996;
							0.985; 0.991;
							0.961; 0.981;
							0.952 0.979

11	5168	861.33	144	64	80.75	374.95; 651.57;
						364.03; 672.53;
						429.49; 766.49;
						774.68; 1185.31;
						1900.44 2339.23
							0.997; 0.998;
							0.996; 0.997;
							0.985; 0.989;
							0.968; 0.98;
							0.951 0.979

12	37324	6220.67	264	112	333.25	410.85; 553.59;
						604.73; 801.7;
						782.65; 910.89;
						2247.25; 2457.88;
						12709.5; 13287.9;
						12003.3 12329.5
							0.998; 0.998;
							0.997; 0.997;
							0.992; 0.994;
							0.979; 0.985;
							0.964; 0.98;
							0.96 0.98

13	90844	15140.67	104	48	1892.58	372.14; 502.7;
						966.44; 1092.16;
						245.71; 366.28;
						457.11; 606.13;
						605.95; 888.7;
						1140.25; 1400.11;
						1206.48 1440.74
							0.999; 0.999;
							0.998; 0.999;
							0.997; 0.997;
							0.989; 0.991;
							0.98; 0.986;
							0.976; 0.984;
							0.974 0.983

14	117296	19549.33	144	64	1832.75	693.71; 857.89;
						503.13; 690.55;
						397.87; 638.12;
						1402.4; 1606.31;
						21117.4; 21510.5;
						30773.2; 32003.9;
						32360.4 33004.2
							0.999; 1.0;
							0.999; 0.999;
							0.996; 0.997;
							0.989; 0.991;
							0.981; 0.987;
							0.977; 0.984;
							0.973 0.984

* Extraction capacity is normalized in the weight (assumed equal), so it can be expressed in number of blocks.

** Rows in each cell under nA-SP and A-RH contain the values for different number of drill holes ($2^{2\cdot(h-1)}$, $2^{2\cdot(h-2)}$,..., 2^0).

Table 2 **Results from numerical tests:** Cases are mapped from Table 1. Going from left to right, columns 2 to 4 show the number of blocks for each instance, the extraction capacity normalized in number of blocks, and the average number of blocks across clusters, while the fifth and sixth columns present the performance in solution time for the optimization and average 1−gap with respect to the solution of the PK policy, for the nA-SP (right) and A-RH (left) policies.

Policy performance in the numerical experiments is presented in Figures 2 and 3 below as a function of the number of drill-holes, where it is compared with that for the PK policy (red line). As policies cannot perform as well as PK, they are below the red line, with those for the A-RH policy shown in green and the nA-SP policy in grey. In each case, the solid line shows the mean value with the confidence interval for the mean shown around it. As expected, as the number of drill holes increases, the average tends toward the PK value and the width of the confidence interval decreases. Most importantly we see that the A-RH policy (in green) consistently outperforms the nA-SP policy.

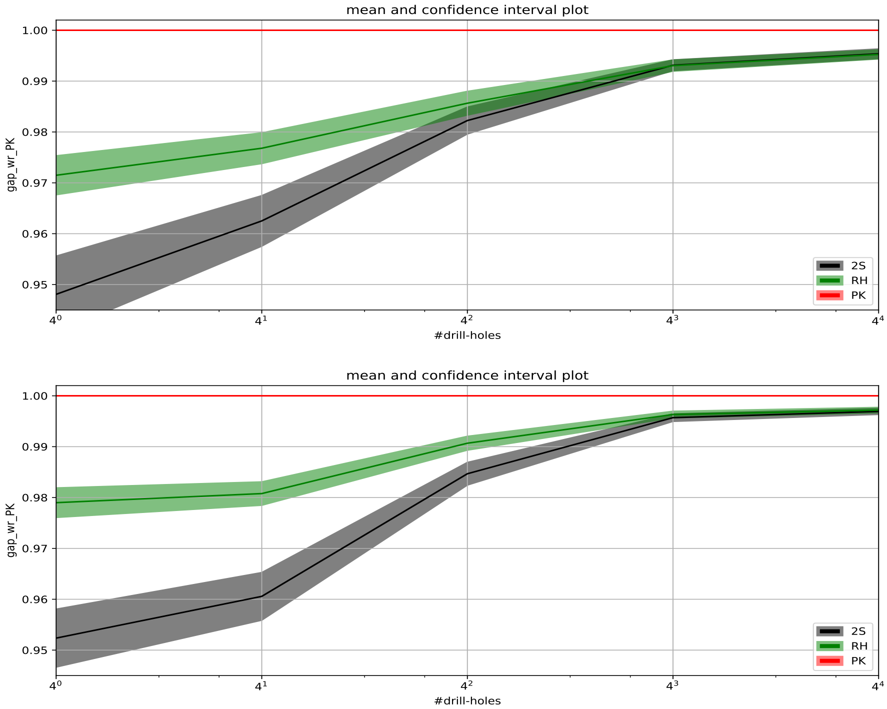


Fig. 2 Comparing the results for the A-RH (green), nA-SP policy (grey) and PK (red) policies. In each case the solid line shows the mean value with the confidence interval around it. Top graph is for case 7 with 4444 blocks, then case 10 with 5168 blocks.

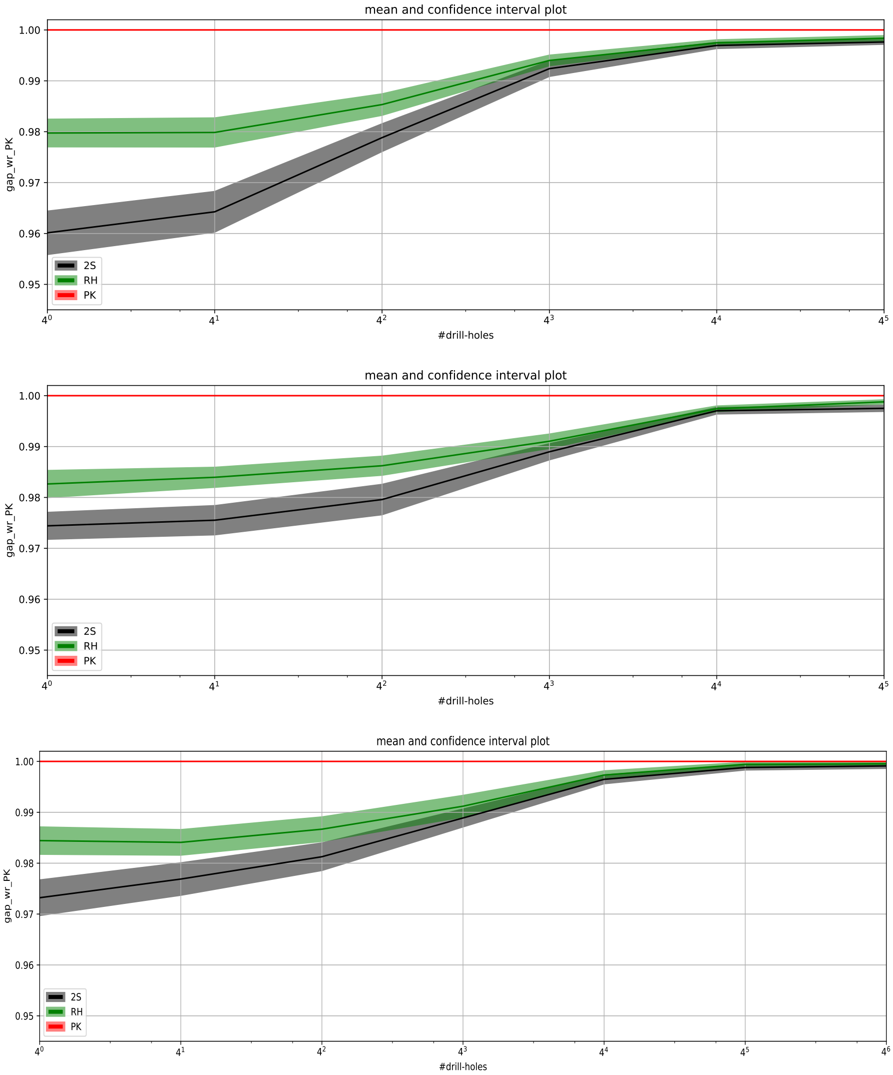


Fig. 3 Comparing policy performance for the A-RH (green), nA-SP (grey) and PK (red) policies. In each case the solid line shows the mean value with the confidence interval around it. Top graph is for case 12 with 37324 blocks, them case 13 with 90844 blocks, and finally case 14 with 117296 blocks.

Next, we explore the performance of the ADP policy of Section 4. Because of the higher computational cost associated with the ADP policy, we considered the following cases.

id.	h	ℓ	$\|T\|$	S	# Instances Solved to optimality.
Case 15	5	6	3	20	100
Case 16	2	2	3	20	100
Case 17	3	4	3	20	100

Table 3 Parameters used to run numerical tests for the ADP: h and ℓ determine the size of the mine, $|T|$ and S are the number of periods and scenarios respectively, and the last column has the number of instances solved out of 100.

The results are displayed in Table 4 and Figure 4. We see that the ADP approach unfortunately does not perform well in these problems, despite being theoretically better (recall the setting in Section 4.2), and more time being consumed on its computation, compared to the other approaches. We conjecture that the theoretical advantage of the ADP approach might arise in the presence of negative correlations, which rarely occur in practical settings, and certainly did not occur in our experiments.

id.	Number of blocks	*Extraction Capacity	Number of precedences	Number of Clusters	Av No Blocks per Cluster	**nA-SP–A-RH–ADP	
						Time–sec.	1−gap

15	4444	1111	104	48	92.58	1.58; 2.46; 784.36;
						2.07; 2.99; 789.15;
						5.27; 6.4; 732.83;
						8.64; 9.94; 683.24;
						9.26; 10.5; 672.19;
							1; 1; 0.99;
							1; 1; 0.91;
							0.98; 0.99; 0.88;
							0.94; 0.98; 0.87;
							0.93; 0.97; 0.87;

16	20	5.0	24	16	1.25	5.3; 5.48; 684.09;
						0.27; 0.48; 1040.69;
							0.97; 0.972; 0.966;
							0.951; 0.965; 0.964;

17	120	20.0	32	20	6.0	5.5; 5.71; 767.19;
						0.43; 0.66; 733.26;
						0.44; 0.69; 714;
							0.988; 0.99; 0.979;
							0.971; 0.976; 0.975;
							0.948; 0.969; 0.974;

* Extraction capacity is normalized in the weight (assumed equal), so it can be expressed in number of blocks.

** Rows in each cell under nA-SP, A-RH and ADP contain the values for different number of drill holes ($2^{2\cdot(h-1)}$, $2^{2\cdot(h-2)}$,..., 2^0).

Table 4 Results from numerical tests for the ADP: Cases are mapped from Table 3. Going from left to right, columns 2 to 4 show the number of blocks for each instance, the extraction capacity normalized in number of blocks, and the average number of blocks across clusters, while the fifth and sixth columns present the performance in solution time for the optimization and average 1−gap with respect to the solution of the perfect knowledge policy, for each policy (respectively nA-SP, A-RH, and ADP).

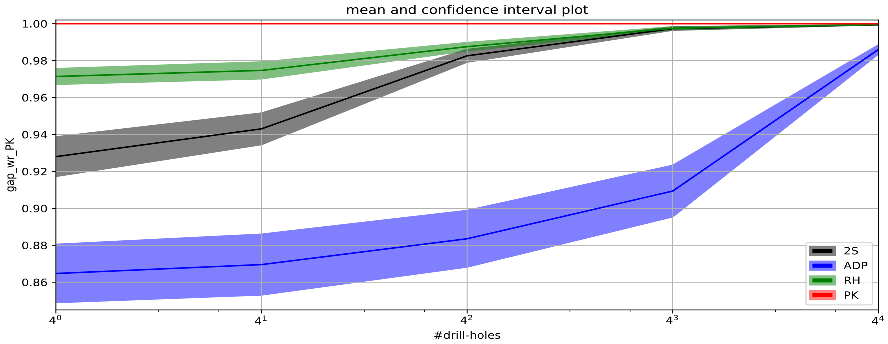


Fig. 4 Comparing the results for the A-RH (green), nA-SP (grey), ADP (blue) and PK (red) policies. In each case the solid line shows the mean value with the confidence interval around it. The graph is for case 15 with 4444 blocks.

6 Conclusions and discussion

We have studied adaptive optimization schemes for the open pit mining production scheduling problem; that is, the problem of deciding when and which blocks to exploit in an open-pit mine over a multi-period time horizon, in order to maximize the total profit of the mining operation. In addition to being theoretically hard, perhaps more importantly, the problem is quite difficult from a practical standpoint: it is a *high-risk high-stakes* problem as it considers long term planning subject to uncertainty, and practical instances are typically quite large.

As we have discussed, the mine planning problem under geostatistical uncertainty (i.e. when the mineral content of the blocks in the mine model are random) is very hard to tackle; in our work, we have studied production policies that sequentially adapt to the observed ore grade content of the mine. That is, we incorporate the fact that, as the deposit is mined over time, the actual ore grade content of mined blocks can be used to update and re-optimize the production schedule of the remaining parts of the mine still to be exploited.

Importantly, our work illustrates the particular challenges of studying adaptability in a prototypical “difficult” problem of large size but with a fairly “amicable” type of uncertainty and learning setting. Indeed, following a common approach in geostatistics we model ore grades as a Gaussian field, or a simple transformation of it. With this, the estimation and updating (or learning) of unobserved ore grades based on previous observations can be carried out using *kriging*, a standard technique from geostatistics.

The main adaptive policy we consider is the *adaptive rolling-horizon* policy (A-RH) (see Section 3.2), where essentially one iterates through the following at each time period: first simulate the ore grades of unextracted blocks based on all previous observations; then use a (non-adaptive) stochastic program to produce an scheduling decision for the remaining time horizon; then perform the exploitation decision for the current time period, and in particular observe the actual ore grades of the latest extracted blocks; then update the belief on the ore grades' distribution based on the latter information; and then re-iterate from the beginning for the upcoming time period. Our numerical experiments on a number of realistic instances show that the adaptive approach yields consistently better results than the non-adaptive approach—even though both approaches perform very well, reaching 95% or more of the value corresponding to the perfect-knowledge solution. Although our models do not include a number of features that are present in more realistic mine planning models (such as stockpiling, waste management, etc.), we believe our results demonstrate the potential of dynamically incorporating the updates of the underlying geological models into the optimization algorithms.

We have also studied a second adaptive one-step look-ahead ADP-based policy, which is standard in sequential decision-making problems under uncertainty, such as the multi-armed bandit framework and other optimization problems with learning, see **?**. Despite the theoretical soundness of the ADP approach, our computational experiments show that there is no clear gain in profit over the A-RH policy, at least in the cases we consider. Moreover, the computational burden is considerably higher, as one has to evaluate more exploiting decisions, simulate possible outcomes, and for each outcome one has to re-estimate and simulate further exploiting decisions. On the other hand, the toy example discussed in Section 4.2 suggests that, in theory, the ADP approach could perform arbitrarily better than the A-RH policy. Nonetheless, the example also suggests that the ADP policy is mostly valuable when there exist *negative* correlations among the random variables—a situation that is not observed in the case of ore grades where we see either positive or zero correlation among blocks. This may help explain why the ADP was outperformed by the other methods in our experiments. We believe that this evidence may be useful for future research in adaptive planning in mining, and furthermore for adaptive multi-stage stochastic programming.

Acknowledgments

The authors thank Xavier Emery (University of Chile) and Eduardo Moreno (Universidad Adolfo Ibañez) for their invaluable help regarding the geostatistics simulations and the optimization models, respectively. This research has been partially supported by the supercomputing infrastructure of the NLHPC (ECM-02) (University of Chile).