

A Decomposition-Based Algorithm for the Scheduling of Open-Pit Networks over Multiple Time Periods

Michelle L. Blom, Adrian R. Pearce, Peter J. Stuckey

Department of Computing and Information Systems, University of Melbourne

We consider the multiple time period short-term production scheduling problem for a network of multiple open-pit mines and ports. Ore produced at each mine, in each period, is transported by rail to a set of ports and blended into products for shipping. Each port forms these blends to a specification, as stipulated in contracts with downstream customers. This problem belongs to a class of multiple producer/consumer scheduling problems in which producers are able to generate a range of products, a combination of which are required by consumers to meet specified demands. In practice, short-term schedules are formed independently at each mine, tasked with achieving a grade and quality target outlined in a medium-term plan. Due to uncertainty in the data available to a medium-term planner, and the dynamics of the mining environment, such targets may not be feasible in the short-term. We present, in this paper, an algorithm in which the grade and quality targets assigned to each mine are iteratively adapted, ensuring the satisfaction of blending constraints at each port, while generating schedules for each mine that maximise resource utilisation.

Key words: short-term open-pit mine production scheduling, hybrid optimisation, non-linear programming

1. Introduction

We consider the Multiple Time Period, Multiple Mine Planning Problem (MTP-MMPP) of scheduling the production of multiple open-pit mines, across a horizon of multiple time periods, to supply several ports with ore that can be blended to form products of a desired composition. This problem belongs to a class of multiple producer/consumer scheduling problems in which producers (mines) are able to generate a range of products (ore of varying grade and quality), and consumers (ports) require a combination (a blend) of these products to meet deterministic (known *a priori*) demands. We extend existing work by Blom et al. (2014) in which a decomposition-based algorithm for the single time period MMPP was developed. In this paper, we consider the significantly more complex multiple time period setting, in which multiple time period schedules must be generated for each mine that are both feasible to enact, and lead to correct blending at the ports. We incorporate additional constraints into our modelling of the MMPP, not present in the single time period model of Blom et al. (2014), to develop a higher fidelity representation of operational behaviour at each

mine. Moreover, we describe a general class of multiple producer/consumer problems to which our solving approach can be applied, instances of which appear in a wide range of domains.

The MTP-MMPP is significantly more complex than the single period MMPP solved by Blom et al. (2014). This is primarily due to the combinatorial complexity of selecting regions of material to extract across multiple time periods, and several mines – where many extraction schedules are possible – to optimise an objective that is dependent on the activity at each mine, in each period.

A solution to the short-term MTP-MMPP schedules the movement of material, from available sources of ore and waste at each mine, to appropriate destinations, and the transport of ore between each mine and port, in each time period of a scheduling horizon. In the short-term, for the case study we consider in this paper, this horizon is 13 weeks long, split into weekly periods. During each time period, at each mine, ore from a variety of sources is processed and blended in a stockyard, producing ore of a specific grade and quality. Ore is reclaimed from this stockyard onto trains, railed to a port, and blended with ore from other mines to form desired products. An optimal solution to the MTP-MMPP requires coordination across mines. The grade and quality of production at each mine, in each period, must support the formation of correctly blended products at each port.

In practice, short-term planning in a network of open-pit mines proceeds with the independent construction of block extraction schedules at each mine, by individual short-term planners. In this process, the short-term planner at each mine is guided by a five year, or medium-term, plan. This plan sets monthly grade and quality targets on mine production – assumed to be both achievable given the estimated composition of material in pit benches, and supportive of port blending constraints. In the short-term, such targets may not be achievable (in conjunction with full usage of processing plants) at one or more mine sites, during one or more time periods, jeopardising the production of blended products at each port. The collaborative adjustment of grade and quality targets assigned to a set of mines, in the generation of short-term plans, can ensure that each mine is assigned goals that can be met, while achieving maximal use of processing facilities.

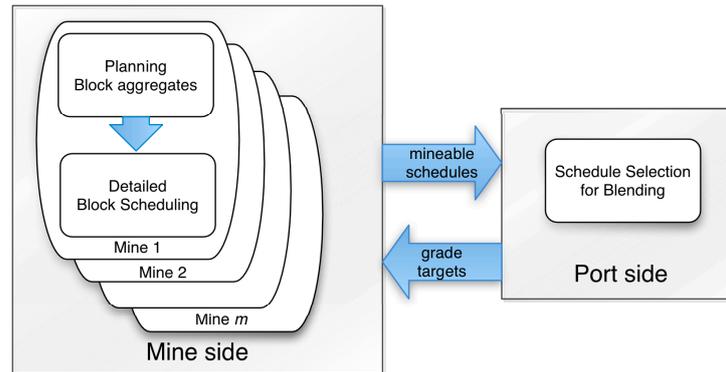
We present a non-linear mixed integer program (MINLP) modelling of the MTP-MMPP. This model is a bilinear program – involving the product of two continuous variables in its constraints. At the short-term horizon, a schedule for a mine site evaluates the average grade of processed material in each time period, and assumes that each train departing the mine in that period carries this grade. We can imagine that all material processed at the site feeds into linearly blended ‘virtual’ stockpiles, from which trains are loaded. This abstract view of mining operations is considered to be a reasonable approximation for the purposes of short-term scheduling. Bilinear constraints are required to ensure that the grade of material leaving each virtual stockpile is equivalent to the combined grade of the material entering it. The relaxation of these bilinear constraints, in terms of McCormick (1976) envelopes, results in solutions with significant deviations present between actual

port product composition and desired bounds (as shown in Section 6). Tighter piecewise-linear relaxations of these constraints were considered for the single period MMPP, and were unable to satisfactorily reduce the resulting deviations (Blom et al. 2014).

The MTP-MMPP, as modelled in this paper, is similar in structure to a pooling problem (Haverly 1978). Such problems model the blending of materials in a feed forward network of source nodes (sources of ore at each mine), intermediate blending pools (virtual stockpiles), and terminal nodes (products formed at each port), under the assumption of linear blending at pools and terminals. Optimisation of the network assigns a rate of flow along each arc, such that profit is maximised and correctly blended products formed at terminals (Misener and Floudas 2009). The principal source of non-linearity in the MTP-MMPP is the modelling of the composition of port products.

The assumption that ore produced by each mine is uniformly blended is reasonable in light of the way this material is stacked in the mine’s stockyard, and reclaimed onto trains. Each stockyard contains one or more chevron stacked stockpiles. Material is spread across these stockpiles in layers, each layer containing material of a similar composition. Each stockpile is reclaimed, once built to a desired size, in slices, extracting material from each layer in a roughly equal proportion. The blending efficiency of a stacking and reclaiming method is defined as the ratio of variance in the grade of material leaving the stockpile (σ_{out}^2) to that entering (σ_{in}^2) (Kumral 2006). Modern methods can achieve blending ratios of 1 : 10, reducing variance in grade by a factor of 10 (Müller 2010). For the mines and ports in the case study we consider in this paper, a 1 : 5 blending effect is achieved. In both planning and practice, running averages are used to represent stockpile composition (Everett 1996). Grade targets to be met by the blended products formed at the ports are placed on *average* grades. As ore moves through the supply chain in large quantities (trainloads are tens of kilotons, and each shipload ranging from tens to hundreds of kilotons), it is safe to assume that the average composition of reclaimed material does not differ appreciably from that of built stockpiles. We compute, in Section 6, the average variance in grade in the blends formed at each port in our case study network, across solutions generated over multiple runs of our decomposition-based algorithm.

At each mine site in our case study, ore that is mined, but not processed, in a time period, is placed on one of a number of stockpiles. Material on these stockpiles can be reclaimed, and processed, in a later period. Tracking the composition of stockpiles over multiple time periods is known to introduce non-linearities into the modelling of a mine (Bley et al. 2012). Additional bilinear constraints, of the form introduced by Bley et al. (2012), are included in our MINLP to model the changing composition of stockpiles at each mine. We show, in this paper, that stockpiles cannot be modelled as maintaining a constant grade across the scheduling horizon without introducing significant errors in the estimated (versus actual) composition of blended port products.

Figure 1 An overview of the decomposition-based algorithm for the MTP-MMPP.

Blom et al. (2014) introduce an algorithm for discovering high quality solutions to a MINLP model of the single period MMPP. In this algorithm, the open-pit production scheduling task is decomposed into multiple subproblems: a set of optimisation problems solved on behalf of a set of mines (generating a set of candidate production schedules for each mine); and an optimisation problem solved on behalf of a network of ports (selecting a candidate schedule to be enacted at each mine, and routing trains of ore between mines and ports). The solving of these subproblems is iterated, with the solution of each mine-side optimisation providing an input to the port-side problem, and the port-side optimisation informing the decisions made at each mine, in subsequent iterations. In this paper, we adapt this algorithm to find high quality solutions to the MTP-MMPP MINLP. The key differences between the single period algorithm of Blom et al. (2014), and the multiple time period algorithm of this paper, lies in the nature and implementation of the mine- and port-side optimisation problems, and the feedback mechanism that exists between these problems in each iteration. A diagram of the main components of the algorithm is shown in Figure 1.

The algorithm presented in this paper, to the best of our knowledge, is the first work to solve an integrated production scheduling problem, involving multiple open-pit mines and a horizon of multiple periods, where the grade and quality of ore to be produced by each mine is not known *a priori*, but determined as part of the optimisation. We show that the algorithm of Blom et al. (2014) can be successfully adapted to the more complex multiple time period setting. Most crucial in this transition is the existence of a sufficiently fast and reliable method to generate multiple time period schedules for each mine in our network. Each schedule must achieve acceptable levels of equipment utilisation in each period, while producing ore that is close to a desired grade and quality. The decision of what to mine in each period has a direct influence on what regions of ore will be accessible later in the horizon. These complexities do not arise in the single period problem of Blom et al. (2014). Full utilisation of available processing capacity is not enforced in this earlier work. Moreover, the need to ensure adequate availability of ore in future periods, or model the

changing state of stockpiles, is not considered. Given the absence of appropriate techniques in the literature, we develop and present a novel heuristic approach for constructing such schedules.

In the multiple time period setting, each mine-side problem is solved via the use of hierarchical planning methods (Bitran and Hax 1977, Hax and Meal 1973), namely: decomposition; aggregation (of units of extraction – known as ‘blocks’); and a rolling horizon solution strategy. The task of generating a multiple time period extraction schedule for a mine is decomposed into two subproblems: a high-level *planning task* in which blocks are grouped into aggregates, and the mining of these aggregates is scheduled; and a *detailed scheduling problem* in which we consider the mining of individual blocks. Both problems are solved with a rolling horizon heuristic. The latter problem is restricted to extracting blocks that form part of an aggregate mined in the solution to the high-level planning task. The result is an efficient heuristic for schedule generation at a mine. In the single time period setting of Blom et al. (2014), such schedules can be quickly found by solving a single mixed integer program (MIP). Given multiple time periods, and the presence of bilinear constraints, generating a schedule by solving a monolithic MINLP is prohibitively time consuming, motivating the need for a heuristic method. Our rolling horizon heuristic monitors the changing state of each stockpile at a mine site, without the need for non-linear constraints. It is important to emphasise, however, that a MIP model of this scheduling task, under the assumption of constant stockpile grades, is also prohibitively time consuming to solve. This is the result of combinatorial complexities that arise with the introduction of multiple inter-dependent time periods.

Each mine-side problem is designed to generate a set of (up to) N *mineable schedules* per mine, with the grade of the ore produced across these schedules clustered about a given grade and quality profile. The multiple-time period port-side subproblem is designed to accept a set of (up to) N mineable block extraction schedules from each mine-side optimisation. Formulated as a MIP, a solution characterises the flow of ore between each mine and port, in each time period, and *selects a schedule*, of the N schedules available for each mine, to be enacted. The objective in this blending problem is to form products at each port whose composition, in each time period, does not deviate from desired bounds on grade and quality. Ore available in the stockyards of each mine, in any given period t , includes that produced by the mine in t , and left-over material (ore produced but not railed to a port) of prior periods. The resulting MIP is more complex than the port-side MIP of Blom et al. (2014). The state of the stockyard of each mine, in each period, is no longer a known constant, but a variable dependent on the decisions of prior time periods. The port-side problem provides, as an output, *grade targets* forming the input to each mine-side optimisation in the next iteration. These targets are based on the grade and quality of ore produced by each mine, in each period, in the best solution found to the port-side MIP across all prior iterations. The schedules formed by each mine-side optimisation, in the next iteration, will produce ore whose grade is

clustered about these new targets. Each mine is, in this way, guided toward finding solutions to its optimisation problem that allow each port to form correctly blended products.

The MTP-MMPP is an instance of a multiple period, multiple producer/consumer scheduling problem in which a set of independently operating producers supply a set of consumers with products to meet deterministic demands. Each consumer combines products sourced from multiple producers to generate their own products for export to an external market. We present two examples of such problems in the domain of food production – the harvesting, transport, and processing of grapes in a wine production supply chain; and the preservation of fruit across a set of producers for supply to exporters, supermarkets, and other consumers. We describe how our decomposition-based approach to the scheduling of supply networks can be applied in these examples.

The key contributions of this paper are: the presentation of a MINLP model of the MTP-MMPP; a decomposition-based algorithm for solving the MTP-MMPP, significantly extending that of Blom et al. (2014); an efficient heuristic for the generation of 13 week schedules at individual mine sites (with stockpiles), forming a core component of our approach; the investigation of several methods for the generation of feedback between the port- and mine-side subproblems in our algorithm; an evaluation of our algorithm on a real, and currently operating, network of open-pit mines, using industry supplied data; and the identification of a general class of problems to which our approach can be applied. Our case study in this paper is a currently operating network of 8 mines and 2 ports, producing over 200 million tons of ore annually at the time of data collection.

The remainder of this paper is structured as follows. In Section 2, we highlight existing work related to the MTP-MMPP. We describe the MTP-MMPP domain in Section 3, presenting a MINLP representation of the problem in Section 4. We describe, and evaluate, our decomposition-based algorithm in Sections 5 and 6. In Section 7, we describe a class of multiple producer/consumer production planning problems to which the methodology we describe in this paper can be applied.

2. Related Work

Weintraub et al. (2008) propose the idea of aggregating MIP models, each designed to schedule a mine in a network of copper mines, for the purpose of making integrated decisions. Two types of aggregation are considered: the grouping of blocks into larger units of extraction; and the aggregation of columns in a MIP representation of the scheduling problem. Both types of aggregation are applied to a single-mine scheduling problem, and the extent to which problem size is reduced is analysed. These techniques were not, however, applied to a case-study involving multiple mines.

While there exists work in which the mine-to-port transport problem, in a network of multiple mines and ports, is optimised (Thomas et al. 2012, 2013, Singh et al. 2013), the composition and tons of ore produced at each mine, in each time period, is known *a priori*, in contrast to the

problem we tackle in this paper. Epstein et al. (2012) present an approach to integrate long-term production scheduling across multiple copper mines (both open-pit and underground) that share downstream processing plants. Each mine produces one or more commercial products, each with defined characteristics (copper grade and contaminant levels). Decision variables determine the tons of each product produced at each mine, and the flow of this material through a network of stockpiles and plants. The grade and quality of production at each mine in this network is known *a priori*, with the tons of each type of product formed determined as part of the optimisation. The problem can thus be modelled as a MIP, in contrast to the MTP-MMPP, which we model as a MINLP, with decision variables denoting the grade and quality of production at each mine.

Our method of generating single-mine extraction schedules, forming a key component of our algorithm, embodies the concepts of hierarchical production planning (Hax and Meal 1973, Bitran and Hax 1977). We first aggregate blocks into larger units of extraction, and schedule the mining of these aggregates over our planning horizon. The original problem is then solved, removing from consideration all blocks not part of an aggregate scheduled for extraction.

The use of aggregation – of data or time – is commonly used to reduce the complexity of an optimisation problem (Rogers et al. 1991). Newman and Kuchta (2007) aggregate time periods into phases in an approach for long-term production scheduling at an underground mine. The key decision variable is machine placement – when to place machines in, and consequently mine, specific regions in the orebody. The phases in which machine placements are started in a solution to this aggregated model, are used to restrict machine placements when solving the original problem. Tabesh and Askari-Nasab (2011) cluster blocks into larger units of mining, on the basis of material type, grade, and location, to reduce the complexity of long-term scheduling in a single open-pit mine. Ramazan (2007) aggregate blocks, in a long-term production scheduling problem, to form ‘fundamental trees’ – minimal collections of blocks that have a positive total economic value, and can be extracted without violating slope constraints – significantly reducing the number of integer variables required in MIP formulations of such problems. Boland et al. (2009) schedule the extraction of block aggregates, in the long-term scheduling of an open-pit mine, while allowing decisions regarding what material is sent to a processing plant to be made at a sub-aggregate level.

For pooling problems involving time, such as the scheduling of crude oil refineries (Shah 1996, Wenkai et al. 2002, Mäs and Pinto 2003, Reddy et al. 2004, Bengtsson et al. 2013) and the planning or design of oilfield infrastructure (Iyer and Grossmann 1998, van den Heever and Grossmann 2000, Carvalho and Pinto 2006), solution techniques are commonly hierarchical or decomposition-based in nature. Shah (1996) consider a crude oil scheduling problem in which oil arriving on ships is allocated into port tanks, piped into storage tanks at a refinery, and allocated to crude distillation units (CDUs) for processing. This problem is decomposed into two parts, solved sequentially: an

upstream problem responsible for the allocation, over time, of crude oil arriving on ships to port tanks, and the sequence in which port tanks are discharged to feed a pipeline to the refinery; and a downstream problem in which the refinery tanks being supplied by the pipeline, and the tanks being discharged to specific CDUs, in each time period, is determined. We take a similar approach in our decomposition of the mining supply chain into a (downstream) mine-port transportation and blending problem, and a series of (upstream) scheduling problems at each mine.

Sundar and Acharya (1995) present a two-stage approach for short-term scheduling at an open-pit mine, first determining the set of blocks to be blasted over the planning horizon, and secondly scheduling the extraction of material in some (or all) of these blocks, in each period. Our method of schedule generation, and the approach of Sundar and Acharya (1995), both select a subset of blocks to be excluded from any schedule formed. We aim to maximise the number of blocks in this set, to reduce the complexity of the scheduling problem, while Sundar and Acharya (1995) aim to minimise the size of this set, maximising the number of blocks blasted in the blasting schedule.

We make use of a rolling horizon technique to generate extraction schedules for individual mines. In this approach, the short-term horizon is discretised into two periods of length 1 and $T - 1$ where T denotes the length of the horizon. A two-period scheduling problem is solved, in which the grade of stockpiles is assumed to remain constant, while their volume is permitted to vary, and the activity of period 1 fixed. The grade of each stockpile at the mine, at the end of period 1, is calculated. The remainder of the horizon (periods 2 to T) is re-discretised, and this process repeated (using the updated grades for each stockpile), until all T periods are scheduled. The use of a rolling horizon allows us to monitor the changing composition of stockpiles over time, while avoiding the need for non-linear constraints. Goodwin et al. (2006) solve a long-term mine planning problem using a similar approach, termed receding horizon control. Time is discretised into periods of non-uniform size, with the quantisation becoming increasingly coarse toward the end of the horizon. The scheduling problem is solved, and the activities of the first period fixed. The remaining periods are re-discretised, and the scheduling problem solved on the reduced horizon. This process continues, and a schedule – whose time periods are uniformly discretised – is generated.

Cullenbine et al. (2011) describe a sliding time window heuristic (STWH) for solving a long term, multiple period, open-pit block sequencing problem, in which a series of integer programs (IPs) are generated and solved. In the first IP, the full set of problem constraints are enforced in the first τ time periods, and a Lagrangian relaxation of the model in the remainder. The solution to this IP is used to fix the variables in the first time period, after which the window of τ periods is moved forward by one, and a second IP enforcing all constraints between periods 2 and $1 + \tau$, is solved. The heuristic repeats this process, and terminates once the last period is scheduled. Lambert and Newman (2013) use the STWH to find an initial feasible solution to a constrained ultimate pit and

block sequencing problem. Where Cullenbine et al. (2011) use Lagrangian relaxation to reduce the complexity of the generated IPs, our approach aggregates time periods, while enforcing all problem constraints in those periods. Our heuristic, and the STWH, do not guarantee that a schedule will be found. However, we demonstrate in Section 6 that the frequency with which our heuristic fails to form a solution, for mines in our case study network, is extremely small. Moreover, we show that a less aggressive aggregation of time periods, forming subproblems of more than two periods, does not result in an improvement to the quality of solutions found to the MTP-MMPP, and incurs a significant computational penalty. Similarly, the increase in solve time resulting from setting $\tau > 1$ in the work of Cullenbine et al. (2011) was found to outweigh any improvement in solution quality.

Beyond the domain of mine planning, the use of aggregation, disaggregation, and rolling horizon-like techniques, is prevalent in the production scheduling literature (see Rodrigues et al. (1996), Bassett et al. (1996), Dimitriadis et al. (1997), Elkamel et al. (1997), Iyer and Grossmann (1998), van den Heever and Grossmann (2000), Màs and Pinto (2003), Reddy et al. (2004), Méndez et al. (2006), Janak et al. (2006), and Maravelias and Sung (2009), for examples). We refer the reader to Osanloo et al. (2008), Newman et al. (2010), Espinoza et al. (2012), and Lambert et al. (2014), for thorough reviews on the use of optimisation in open-pit mining, a description of common problems arising in the scheduling and operation of open-pit mines, unaddressed challenges for future research, and techniques for expediting the solving of block sequencing formulations.

3. Modelling the Multiple Mine Network

Let \mathcal{M} denote a set of mines, connected by rail to a set of ports, Π . We consider the open-pit mining of ores that are sold in two granularities – lump and fines – distinguished by their particle size. At each mine $m \in \mathcal{M}$, in each period t , $t \in \{1, 2, \dots, T\}$, ore and waste is extracted, by dig units (eg. loaders, shovels, and excavators), from geological regions (known as ‘blocks’), processed into lump (6 to 31 mm) and fines (< 6 mm) granularities, and loaded onto trains to be railed to a port $\pi \in \Pi$. Ore arriving at each port is blended onto stockpiles, from which it is loaded onto ships for delivery to customers. Appendix A outlines the meaning of notation used throughout this paper.

Each mine m contains a set of grade, \mathcal{B}_m^g , and blast blocks, \mathcal{B}_m^b . A short-term plan selects a number of blocks, from either set, to be extracted, and the destination of this material (stockpiles or processing plants), in each time period. Grade blocks are regions of similar composition, and form the broken stock of a mine – ore and waste that has been primed for extraction (blasted). Each blast block is a region of material that has not yet been blasted, and divided into grade blocks. For each block $b \in \mathcal{B}_m = \mathcal{B}_m^g \cup \mathcal{B}_m^b$, $\mathcal{A}_{m,b}^\wedge$ denotes the set of blocks that lie directly above b , *all* of which must be mined before b can be accessed. $\mathcal{A}_{m,b}^\vee$ denotes the set of blocks adjacent to $b \in \mathcal{B}_m$, in the same bench, only *one* of which must be mined before b can be accessed.

Each block $b \in \mathcal{B}_m$, at $t = 1$, contains one or more types of material – $T_{hi}^{t=1}(b)$ tons of high grade, $T_{lo}^{t=1}(b)$ tons of low grade, and $T_w^{t=1}(b)$ tons of waste – distributed throughout the block. For each grade block $b \in \mathcal{B}_m^g$, $T_k^{t=1}(b)$ is non-zero for only one value of k . The composition of the high, low grade, and waste material in $b \in \mathcal{B}_m$ is defined in terms of the percentage of a number of elements, \mathcal{Q} , denoting metal grade and impurities, in its lump and fines components ($G_{b,l,q}^{m,k}$ for $q \in \mathcal{Q}$, $l \in \mathcal{L}$, and $k = hi, lo$, and w). The split of material k in a block $b \in \mathcal{B}_m$, denoted $S_{b,l}^{m,k}$, defines the percentage of material k in b that will split (upon processing) into granularity $l \in \mathcal{L}$.

The waste in a block is hauled, by truck, to a waste dump ($\delta \in \Delta_m$). High grade ore is hauled to a dry processing plant (κ), or a high grade stockpile ($\theta \in \Theta_m$). Low grade ore is hauled to a low grade stockpile ($\lambda \in \Lambda_m$), or a wet processing plant (ω), if one exists at m . Material on high and low grade stockpiles is fed, if needed, to the dry and wet processing plants, respectively. The tons of material on each stockpile $s \in \Theta_m \cup \Lambda_m \cup \Delta_m$ at $t = 1$ is denoted $T^{t=1}(s)$. The split of ore in a stockpile $s \in \Theta_m \cup \Lambda_m$ is denoted $S_{s,l}^{m,t=1}$, and its composition by $G_{s,l,q}^{m,t=1}$ for $q \in \mathcal{Q}$ and $l \in \mathcal{L}$.

A wet processing plant upgrades low grade ore, producing a stream of (rejected) tailings and a concentrate, as described by Blom et al. (2014). The tons of valuable metal (and other attributes) in this concentrate is proportional to the tons of input feed (as per a recovery factor $R_{l,q}^{m,\omega}$ for $q \in \mathcal{Q}$). The tons of concentrate produced is proportional to the tons of input feed (as per a yield factor $Y_l^{m,\omega}$). This concentrate is blended with lump and fines ore produced by the dry plant.

Capacities exist on the: extraction of material at each mine, C_e^m tons per period, on the basis of available dig units; tons of material hauled by truck, C_τ^m tons per period; tons of ore processed by the dry and wet plants, C_κ^m and C_ω^m tons per period; the tons of material permitted on each stockpile $s \in \Theta_m \cup \Lambda_m \cup \Delta_m$, C_s^m tons; and the tons of ore handled at each port π , per period, C_π . The capacity of each plant must be fully utilised to maximise production across the network. To maximise productivity, dig and trucking resources must also be fully utilised. In practice, full utilisation of both types of resource is not possible, as the dig and truck fleets will differ in capacity.

Ore produced by each mine is transported in T_R ton trainloads to a port $\pi \in \Pi$. Ore arriving at each port π is blended to form a set $N_{\pi,l}$ of products of each granularity $l \in \mathcal{L}$. Each product $n \in N_{\pi,l}$ is associated, in each time period t , with bounds on its grade and quality, expressed in terms of a lower ($L_{n,q}^{\pi,l,t}$) and upper ($U_{n,q}^{\pi,l,t}$) bound on the percentage of each $q \in \mathcal{Q}$.

The set of ore and waste sources at mine m is denoted $\mathcal{S}_m = \mathcal{B}_m \cup \Theta_m \cup \Lambda_m$. The set of destinations to which such sources can be transported is denoted $\mathcal{D}_m = \{\kappa, \omega\} \cup \Delta_m \cup \Theta_m \cup \Lambda_m$. Variables $x_{s,d}^{m,t}$ for $s \in \mathcal{S}_m$ and $d \in \mathcal{D}_m$ denote the tons of each source s extracted and hauled to each of its possible destinations d , at m , in period t . Variable $r_{m,\pi,t}^{l,n,t'}$ denotes the floating point number of trainloads of granularity $l \in \mathcal{L}$, produced by mine m in period t' , and transported by rail from m to port π , in period t , to be blended into product $n \in N_{\pi,l}$. The port system need not rail the entirety of a

mine’s production to a port in any given time period. We assume that any remaining ore in the stockyard of a mine at the end of a time period t , must be transported to a port in period $t + 1$.

Our modelling of the MTP-MMPP differs from the single period problem of Blom et al. (2014) in that: scheduling decisions are indexed by time; blocks contain multiple types of material, each of which must be sent to different destinations; wet and dry plants are constrained to be fully utilised in each period; high and low grade stockpile composition is tracked over time (via the addition of non-linear constraints); and the integrality of trainloads is relaxed. The transport of ore between mines and ports is not modelled in great detail, with emphasis placed on mapping ore produced at each mine to the products shipped from each port. Relaxing trainload integrality allows us to model the distribution of trainloads of ore to multiple products at a port, and the transport of partial trainloads of ‘left over’ ore, produced at a mine in prior periods, but not railed to a port. While, in practice, ‘left over’ material is blended with new production and transported in full trainloads, this approximation avoids the need for additional non-linear constraints modelling the changing composition of stockyard stockpiles. A higher fidelity modelling of the transport network between mines and ports, and the operational processes of each port, is planned as future work.

The addition of time periods, in conjunction with the mining precedences that exist between blocks, significantly increases the complexity of the MMPP. Each mining decision has an influence on what can be accomplished in remaining time periods, and whether an adequate supply of processable ore, available for extraction, will exist to meet the future needs of each plant.

4. A MINLP Model of the MTP-MMPP

Given a network of mines \mathcal{M} , ports Π , and parameters (of Appendix A), the MTP-MMPP is defined as finding an instantiation of variables: $x_{s,d}^{m,t}$, for each $m \in \mathcal{M}$, $t \in \{1, 2, \dots, T\}$, $s \in \mathcal{S}_m$, and $d \in \mathcal{D}_m$; and $r_{m,\pi,t}^{l,n,t'}$ for each $m \in \mathcal{M}$, $\pi \in \Pi$, $t, t' \in \{1, 2, \dots, T\}$, $l \in \mathcal{L}$, and $n \in N_{\pi,l}$. A solution to the MTP-MMPP defines the flow of ore and waste across each mine, and the transport of ore between each mine and port, in each of the T time periods in our planning horizon.

Our objective in the MTP-MMPP is to minimise the total deviation present between the composition of products formed at each port, over the given horizon, and desired upper and lower bounds, while maximising the productivity of each mine. We define productivity in terms of the ‘desirable utilisation’ of dig and trucking resources. Dig and trucking capacity not used in the extraction of ore to supply processing plants, should be used for other purposes – for example, the removal of waste to expose ore for future extraction. The transfer of ore to a stockpile, and then from that stockpile to a plant – a process known as ‘double handling’ – is to be avoided, unless required to ensure that processing plants are able to be run continuously at full capacity. This may occur if a mine needs to enter a phase of increased waste removal, reducing the portion of resource capacities

devoted to mining processable ore. In this instance, ore from stockpiles, located near processing plants, can be used to offset the reduction in processable ore being supplied from each pit.

4.1. The Objective

Let variable $\overline{rv}_{\pi,n,t}^{m,l,q,t'}$ denote the tons of attribute $q \in \mathcal{Q}$, from granularity l produced at mine m in period t' , blended into product $n \in N_{\pi,l}$ at port π in period t ; β_1 and β_2 constants such that $\beta_1 \gg \beta_2$; \vec{x} and \vec{r} the set of all $x_{s,d}^{m,t}$ and $r_{\pi,n,t}^{m,l,t'}$ variables, respectively, where $m \in \mathcal{M}$, $s \in \mathcal{S}_m$, $d \in \mathcal{D}_m$, $\pi \in \Pi$, $l \in \mathcal{L}$, $n \in N_{\pi,l}$, and $t, t' \in \{1, 2, \dots, T\}$; $\eta(\vec{x}, \vec{r}, t)$ a measure of the extent to which port products deviate from desired bounds, in period t (Equation 2); and $\rho_m(\vec{x}, t)$ the productivity of mine m in period t (Equation 3). The objective of the MINLP, $Z_{MTP-MMPP}$ in Equation (1), is to minimise deviation that exists between the composition of port products and desired bounds, and to maximise the productivity achieved at each mine, across all time periods.

$$Z_{MTP-MMPP} = \min \sum_{t=1}^T \left[\beta_1 \eta(\vec{x}, \vec{r}, t) - \beta_2 \sum_{m \in \mathcal{M}} \rho_m(\vec{x}, t) \right] \quad (1)$$

In the following: Δ_q^+ denotes a significant change in the percentage of attribute $q \in \mathcal{Q}$ in a body of ore;¹ Φ_ω^m a binary parameter whose value is 1 if mine m has facilities to process low grade ore (and 0, otherwise); $f(t) = \max(1, t - 1)$; and $g(t) = \min(t + 1, T)$.

$$\begin{aligned} \eta(\vec{x}, \vec{r}, t) = & \sum_{\pi \in \Pi} \sum_{l \in \mathcal{L}} \sum_{n \in N_{\pi,l}} \sum_{q \in \mathcal{Q}} \frac{1}{\Delta_q^+} \max \left[0, \sum_{m \in \mathcal{M}} \sum_{t'=f(t)}^t \overline{rv}_{\pi,n,t}^{m,l,q,t'} - U_{n,q}^{\pi,l,t} \sum_{m \in \mathcal{M}} \sum_{t'=f(t)}^t T_R r_{\pi,n,t}^{m,l,t'} \right] \\ & + \sum_{\pi \in \Pi} \sum_{l \in \mathcal{L}} \sum_{n \in N_{\pi,l}} \sum_{q \in \mathcal{Q}} \frac{1}{\Delta_q^+} \max \left[0, L_{n,q}^{\pi,l,t} \sum_{m \in \mathcal{M}} \sum_{t'=f(t)}^t T_R r_{\pi,n,t}^{m,l,t'} - \sum_{m \in \mathcal{M}} \sum_{t'=f(t)}^t \overline{rv}_{\pi,n,t}^{m,l,q,t'} \right] \end{aligned} \quad (2)$$

Equation (3), below, calculates the productivity of a mine in terms of the tons of waste hauled to a dump, and the occurrence of undesirable stockpiling. Hauling low grade ore to a stockpile is considered undesirable only if the mine has facilities for its upgrade.

$$\rho_m(\vec{x}, t) = \sum_{s \in \mathcal{S}_m} \left[\sum_{\delta \in \Delta_m} x_{s,\delta}^{m,t} - \sum_{\theta \in \Theta_m} x_{s,\theta}^{m,t} + (1 - 2\Phi_\omega^m) \sum_{\lambda \in \Lambda_m} x_{s,\lambda}^{m,t} \right] \quad (3)$$

¹ The value of Δ_q^+ may be 0.1%, for example, if q denotes metal percentage, or on the order of 0.001% for an impurity.

4.2. Constraints

For brevity, we present the subset of constraints in the MTP-MMPP that either do not appear in the single period MINLP of Blom et al. (2014), or differ significantly in the multiple period setting. The full set of constraints in our MINLP model of the MTP-MMPP is presented in Appendix B.

Let $\tau_l^{m,t}$ denote the tons of granularity l produced at m in t ; $v_{l,q}^{m,t}$ the (fractional) percentage of $q \in \mathcal{Q}$ in the ore of granularity l produced at m in t ; $y_{m,t}^{\sigma,b}$ a binary variable with value 1 if and only if block b is, at least, partially mined by the end of period t ; $y_{m,t}^{\tau,b}$ a binary variable with value 1 if and only if b is completely mined by the end of t ; $a_{m,l}^{t,s,d}$ the tons of granularity l in the material on stockpile $s \in \Theta_m \cup \Lambda_m$ sent to destination $d \in \{\kappa, \omega\}$ in t ; $a_{m,l,q}^{t,s,d}$ the tons of attribute q in the ore of granularity l sent to destination d , from stockpile s , in t ; $o_s^{m,t}$ the tons of material on stockpile s at the start of t ; $o_{s,l}^{m,t}$ the tons of granularity l in the material on stockpile s at the start of t ; and $o_{s,l,q}^{m,t}$ the tons of attribute q in the ore of granularity l on stockpile s at the start of t . The values of all variables, excepting binaries $y_{m,t}^{\tau,b}$ and $y_{m,t}^{\sigma,b}$, are restricted to non-negative reals.

Constraint (4) ensures that, in each period, the processing of ore at each plant is equal, within a tolerance ϵ , to C_d^m for $d \in \{\kappa, \omega\}$. Constraint (5) ensures that all ore produced by each mine m in each period t is transported to a port by the end of period $t + 1$.

$$C_d^m - \epsilon \leq \sum_{s \in \mathcal{S}_m} x_{s,d}^{m,t} \leq C_d^m + \epsilon \quad \forall m \in \mathcal{M}, d \in \{\kappa, \omega\}, t \in \{1, 2, \dots, T\} \quad (4)$$

$$\sum_{t=t'}^{g(t')} \sum_{\pi \in \Pi} \sum_{n \in N_{\pi,t}} T_R r_{\pi,n,t}^{m,l,t'} = \tau_l^{m,t'} \quad \forall m, l, t' \in \{1, 2, \dots, T\} \quad (5)$$

For $k \in \{hi, lo, w\}$, Constraint (6) sets the value of binaries $y_{m,t}^{\sigma,b}$ (1 if the mining of $b \in \mathcal{B}_m$ has been scheduled by, or during, t) and $y_{m,t}^{\tau,b}$ (1 if b is to be entirely extracted by, or during, t).

$$y_{m,t}^{\tau,b} \sum_k T_k^{t=1}(b) \leq \sum_{t'=1}^t \sum_{d \in \mathcal{D}_m} x_{b,d}^{m,t'} \leq y_{m,t}^{\sigma,b} \sum_k T_k^{t=1}(b) \quad \forall m \in \mathcal{M}, b \in \mathcal{B}_m, t \in \{1, 2, \dots, T\}, \quad (6)$$

Constraint (7) ensures that no blast blocks are mined prior to a specific time period, TB_m , at each mine $m \in \mathcal{M}$. Prior to TB_m , only grade blocks can be extracted at m . Constraints (8)–(11) ensure that stockpile capacities are respected and that no more than $o_s^{m,t}$ tons (the tons of ore on stockpile s at the start of t) can be extracted from any stockpile $s \in \Theta_m \cup \Lambda_m$ in any period t .

$$y_{m,t}^{\sigma,b} = 0 \quad \forall m \in \mathcal{M}, b \in \mathcal{B}_m^b, t < TB_m, \quad (7)$$

$$o_s^{m,t} = T^{t=1}(s) \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, t = 1, \quad (8)$$

$$o_s^{m,t} = o_s^{m,t-1} - \sum_{d \in \{\kappa, \omega\}} x_{s,d}^{m,t-1} + \sum_{b \in \mathcal{B}_m} x_{b,s}^{m,t-1} \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, t \in \{2, 3, \dots, T\}, \quad (9)$$

$$o_s^{m,t} - \sum_{d \in \{\kappa, \omega\}} x_{s,d}^{m,t} + \sum_{b \in \mathcal{B}_m} x_{b,s}^{m,t} \leq C_s^m \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, t \in \{2, 3, \dots, T\}, \quad (10)$$

$$o_s^{m,t} - \sum_{d \in \{\kappa, \omega\}} x_{s,d}^{m,t} \geq 0 \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, t \in \{2, 3, \dots, T\}, \quad (11)$$

Constraints (12)–(15) define the tons of each granularity $l \in \mathcal{L}$ and attribute $q \in \mathcal{Q}$ residing on each stockpile $s \in \Theta_m \cup \Lambda_m$ at the start of period t .

$$o_{s,l}^{m,t} = S_{s,l}^{m,t=1} T^{t=1}(s) \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, l \in \mathcal{L}, t = 1, \quad (12)$$

$$o_{s,l}^{m,t} = o_{s,l}^{m,t-1} - \sum_{d \in \{\kappa, \omega\}} a_{s,l}^{m,t-1,d} + \sum_{b \in \mathcal{B}_m} S_{b,l}^m x_{b,s}^{m,t-1} \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, l \in \mathcal{L}, \quad (13)$$

$$t \in \{2, 3, \dots, T\},$$

$$o_{s,l,q}^{m,t} = S_{s,l}^{m,t=1} G_{s,l,q}^{m,t=1} T^{t=1}(s) \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, l \in \mathcal{L}, \quad (14)$$

$$q \in \mathcal{Q}, t = 1,$$

$$o_{s,l,q}^{m,t} = o_{s,l,q}^{m,t-1} - \sum_{d \in \{\kappa, \omega\}} a_{s,l,q}^{m,t-1,d} + \sum_{b \in \mathcal{B}_m} S_{b,l,q}^m G_{b,l,q}^m x_{b,s}^{m,t-1} \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, l \in \mathcal{L}, q \in \mathcal{Q}, \quad (15)$$

$$t \in \{2, 3, \dots, T\},$$

Constraints (16)–(17) ensure that the composition of material leaving each stockpile $s \in \Theta_m \cup \Lambda_m$ for a processing plant $d \in \{\kappa, \omega\}$ in each period t is equal to that of the stockpile at the start of t . These bilinear constraints, introduced in the work of Bley et al. (2012), add $\sum_m T |\mathcal{L}| |\Theta_m \cup \Lambda_m|$ and $\sum_m T |\mathcal{L}| |\mathcal{Q}| |\Theta_m \cup \Lambda_m|$ bilinear terms to the model, respectively. Each stockpile can supply either the dry (κ) or wet (ω) processing plants, but not both. Constraints restricting the movement of material along valid source to destination pathways are provided in Appendix B.

$$a_{s,l}^{m,t,d} o_s^{m,t} = x_{s,d}^{m,t} o_{s,l}^{m,t} \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, d \in \{\kappa, \omega\}, l \in \mathcal{L}, t \in \{1, 2, \dots, T\} \quad (16)$$

$$a_{s,l,q}^{m,t,d} o_s^{m,t} = a_{s,l}^{m,t,d} o_{s,l,q}^{m,t} \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, d \in \{\kappa, \omega\}, l \in \mathcal{L}, q \in \mathcal{Q}, t \in \{1, 2, \dots, T\} \quad (17)$$

Variables $\tau_l^{m,t}$ and $v_{l,q}^{m,t}$ are linked by Constraint 18, where $v_{l,q}^{m,t}$ denotes the tons of $q \in \mathcal{Q}$ in the ore of granularity $l \in \mathcal{L}$ produced by mine m in t . The equations defining variables $\tau_l^{m,t}$, $v_{l,q}^{m,t}$, and the quantity $\nu_{l,q}^{m,t}$, are provided in Appendix B. Constraint (19) defines variable $\overline{rv}_{\pi,n,t}^{m,l,q,t'}$, used in

Equation (2). Constraints (18)–(19) introduce $T|\mathcal{M}||\mathcal{L}||\mathcal{Q}|$ and $\sum_{\pi \in \Pi} \sum_{l \in \mathcal{L}} (2T - 1)|\mathcal{Q}||N_{\pi,l}|$ bilinear terms to the model, respectively.

$$v_{l,q}^{m,t} \tau_l^{m,t} = \nu_{l,q}^{m,t} \quad \forall m \in \mathcal{M}, t \in \{1, 2, \dots, T\}, l \in \mathcal{L}, q \in \mathcal{Q}, \quad (18)$$

$$T_R r_{\pi,n,t}^{m,l,t'} v_{l,q}^{m,t'} = \bar{r} v_{\pi,n,t}^{m,l,q,t'} \quad \forall m \in \mathcal{M}, t \in \{1, 2, \dots, T\}, t' \in \{f(t), t\}, l \in \mathcal{L}, q \in \mathcal{Q}, n \in N_{\pi,l}. \quad (19)$$

5. A Decomposition-Based Algorithm

Blom et al. (2014) present an iterative, decomposition-based algorithm for the single time period MMPP. The steps of this algorithm, shown in Algorithm 1, remain the same in the case of multiple time periods. The mine- and port-side subproblems solved in each iteration, however, are significantly more complex in the multiple time period setting. The key elements of this algorithm are the decomposition of the MTP-MMPP into mine- and port-side subproblems, denoted \mathcal{O}_m and \mathcal{O}_{Π} , respectively; the solving of each \mathcal{O}_m and \mathcal{O}_{Π} in succession to produce a solution to the MTP-MMPP (Steps 7–8); the recording of the best found solution, denoted \vec{s}_{best} , after each iteration (Step 9); and the passing of feedback, by \mathcal{O}_{Π} , to each \mathcal{O}_m , at the end of each iteration (Step 10).

Each \mathcal{O}_m subproblem is responsible for generating a set of (up to) N block extraction schedules for mine $m \in \mathcal{M}$ (Step 7). A grade and quality target, denoted $\vec{\phi}_m$, stating the desired percentage of each attribute $q \in \mathcal{Q}$ in each granularity $l \in \mathcal{L}$ to be produced by m , in each time period, is provided as input. A set of (up to) N schedules, whose production lies in the vicinity of this target, is formed. Recall that each schedule defines the movement of material across mine m – from each source $s \in \mathcal{S}_m$ (blocks and stockpiles) to each destination $d \in \mathcal{D}_m$ (stockpiles and plants) – in each time period t , instantiating variables $x_{s,d}^{m,t}$ for all $t \in \{1, 2, \dots, T\}$.

Diversity in the grade of ore produced, across the N schedules formed by \mathcal{O}_m , is controlled by a vector of standard deviations $\vec{\sigma}_m$, containing one standard deviation for each combination of $q \in \mathcal{Q}$, $l \in \mathcal{L}$, and $t \in \{1, 2, \dots, T\}$, forming a second input to each \mathcal{O}_m . Smaller standard deviations lead to a set of schedules, across which, the grade of produced ore is more tightly clustered about the assigned target, $\vec{\phi}_m$. Section 5.1 describes, in more detail, how each \mathcal{O}_m is formulated and solved.

The \mathcal{O}_{Π} subproblem is given a set of (up to) N schedules, from each \mathcal{O}_m , and must select one schedule in each set to be enacted (Step 8). The goal of \mathcal{O}_{Π} is to form correctly blended products at each port, from the ore produced at each mine. If \mathcal{O}_{Π} cannot find such a selection of schedules, one for each mine, for which port products are correctly formed, the selection that allows it to minimise deviation between the grade and quality of port products and desired bounds is made.

Algorithm 1 A decomposition-based algorithm for the MTP-MMPP, where: Δ_q^+ and Δ_q^- denote significant and insignificant changes in $q \in \mathcal{Q}$ percentage, respectively; Ξ_m a medium-term grade and quality target assigned to mine $m \in \mathcal{M}$; and MAX_i a cap on the number of iterations of the algorithm executed.

- 1: $\vec{s}_{best} \leftarrow \emptyset$
 - 2: $\vec{\sigma}^+ \leftarrow \{\sigma_{l,q}^+ = \Delta_q^+ | l \in \mathcal{L}, q \in \mathcal{Q}\}$
 - 3: $\vec{\sigma}^- \leftarrow \{\sigma_{l,q}^- = \Delta_q^- | l \in \mathcal{L}, q \in \mathcal{Q}\}$
 - 4: $i \leftarrow 1$
 - 5: Initialise expected mine targets and standard deviation sets: $\vec{\phi}_m^i \leftarrow \Xi_m$ and $\vec{\sigma}_m^i \leftarrow \vec{\sigma}^+$, for all $m \in \mathcal{M}$.
 - 6: **repeat**
 - 7: Solve each \mathcal{O}_m to find (up to) N schedules for mine m , Ω_m^i , producing ore whose composition is clustered about $\vec{\phi}_m^i$ with a spread determined by the standard deviations in $\vec{\sigma}_m^i$.
 - 8: Solve \mathcal{O}_Π given sets $\Omega_m^i \cup \{\vec{s}_{best,m}\}$ from each $m \in \mathcal{M}$, where $\vec{s}_{best,m} \in \vec{s}_{best}$ is the schedule to be enacted by m in the best solution found thus far. Select a schedule to be enacted at each mine, and a routing of ore between mines and ports, forming a solution \vec{s}_i to the MTP-MMPP.
 - 9: Update best solution \vec{s}_{best} if and only if $Z_{MTP-MMPP}(\vec{s}_i) < Z_{MTP-MMPP}(\vec{s}_{best})$.
 - 10: Generate feedback to each \mathcal{O}_m by adapting $\vec{\phi}_m^i$ and $\vec{\sigma}_m^i$ to form $\vec{\phi}_m^{i+1}$ and $\vec{\sigma}_m^{i+1}$.
 - 11: $i \leftarrow i + 1$
 - 12: **until** [$Z_{MTP-MMPP}(\vec{s}_i) \geq Z_{MTP-MMPP}(\vec{s}_{best}) \wedge \nexists m \in \mathcal{M}. \vec{\sigma}_m^i \neq \vec{\sigma}^-$] $\vee i > MAX_i$
 - 13: **return** \vec{s}_{best}
-

In doing so, the remaining variables in our MINLP are instantiated, forming a solution to the MTP-MMPP. Section 5.3 describes, in more detail, how \mathcal{O}_Π is formulated and solved.

Our decomposition-based algorithm repeats the solving of each \mathcal{O}_m and \mathcal{O}_Π , in sequence, generating a series of monotonically improving solutions to the MTP-MMPP. The quality of solution \vec{s}_i , found by \mathcal{O}_Π in iteration i , is given by its objective value, $Z_{MTP-MMPP}(\vec{s}_i)$, as per Equation (1). A record of the best solution found by \mathcal{O}_Π , \vec{s}_{best} , is maintained over the course of the algorithm. A new solution, \vec{s}_i , replaces \vec{s}_{best} if and only if $Z_{MTP-MMPP}(\vec{s}_i) < Z_{MTP-MMPP}(\vec{s}_{best})$ (Step 9). In each iteration $i > 1$, \mathcal{O}_Π is able to select the schedule chosen for mine m in \vec{s}_{best} , denoted $\vec{s}_{best,m}$, in place of those newly generated, ensuring that $Z_{MTP-MMPP}(\vec{s}_i) \leq Z_{MTP-MMPP}(\vec{s}_{best})$, for all $i > 1$.

Our algorithm encourages each \mathcal{O}_m to construct schedules that will allow \mathcal{O}_Π to form correctly blended products at each port. This is accomplished via the use of feedback, passed from \mathcal{O}_Π to each \mathcal{O}_m , at the end of each iteration (Step 10). The composition of production at each mine m , in \vec{s}_{best} , forms a new grade target, given to \mathcal{O}_m as input in the next iteration.

The set of standard deviations given to each \mathcal{O}_m , in each iteration, serves a dual purpose. \mathcal{O}_Π will increase and decrease these deviations over the course of the algorithm to encourage more, or less diversity in the composition of produced ore, across the schedule sets generated by each \mathcal{O}_m . When these standard deviations reach a minimum size, termination of the algorithm is triggered, at which point the best found solution, \vec{s}_{best} is returned (Steps 12-13). This termination mechanism,

and the feedback that takes place between \mathcal{O}_Π and each \mathcal{O}_m , influencing production across the schedules formed by \mathcal{O}_m , in successive iterations, are described in more detail in Section 5.4.

The algorithm presented in this section is an anytime approach, forming a solution to the MTP-MMPP in each iteration. A cap on the number of iterations performed, MAX_i , can be specified to keep run-time within an acceptable range. In practice, we have found the algorithm to terminate within 200 iterations in each of the experiments of Section 6.

5.1. Optimisation at the Mines: The \mathcal{O}_m Problem

A solution to each \mathcal{O}_m , in iteration i , is a set of (up to) N extraction schedules, denoted Ω_m^i . Each schedule instantiates variables $x_{s,d}^{m,t}$, from the MINLP of Section 4, for all $t \in \{1, 2, \dots, T\}$, $s \in \mathcal{S}_m$, and $d \in \mathcal{D}_m$. A grade and quality target $\vec{\phi}_m^i = \{\phi_{l,q}^{m,t,i} | t \in \{1, 2, \dots, T\}, l \in \mathcal{L}, q \in \mathcal{Q}\}$ and standard deviation vector $\vec{\sigma}_m^i = \{\sigma_{l,q}^{m,t,i} | t \in \{1, 2, \dots, T\}, l \in \mathcal{L}, q \in \mathcal{Q}\}$ is given to \mathcal{O}_m as input. We have devised a heuristic to generate Ω_m^i , across which the percentage of $q \in \mathcal{Q}$ in the ore of granularity l produced by m in each t is clustered about $\phi_{l,q}^{m,t,i}$ with a spread determined by the standard deviation $\sigma_{l,q}^{m,t,i}$.

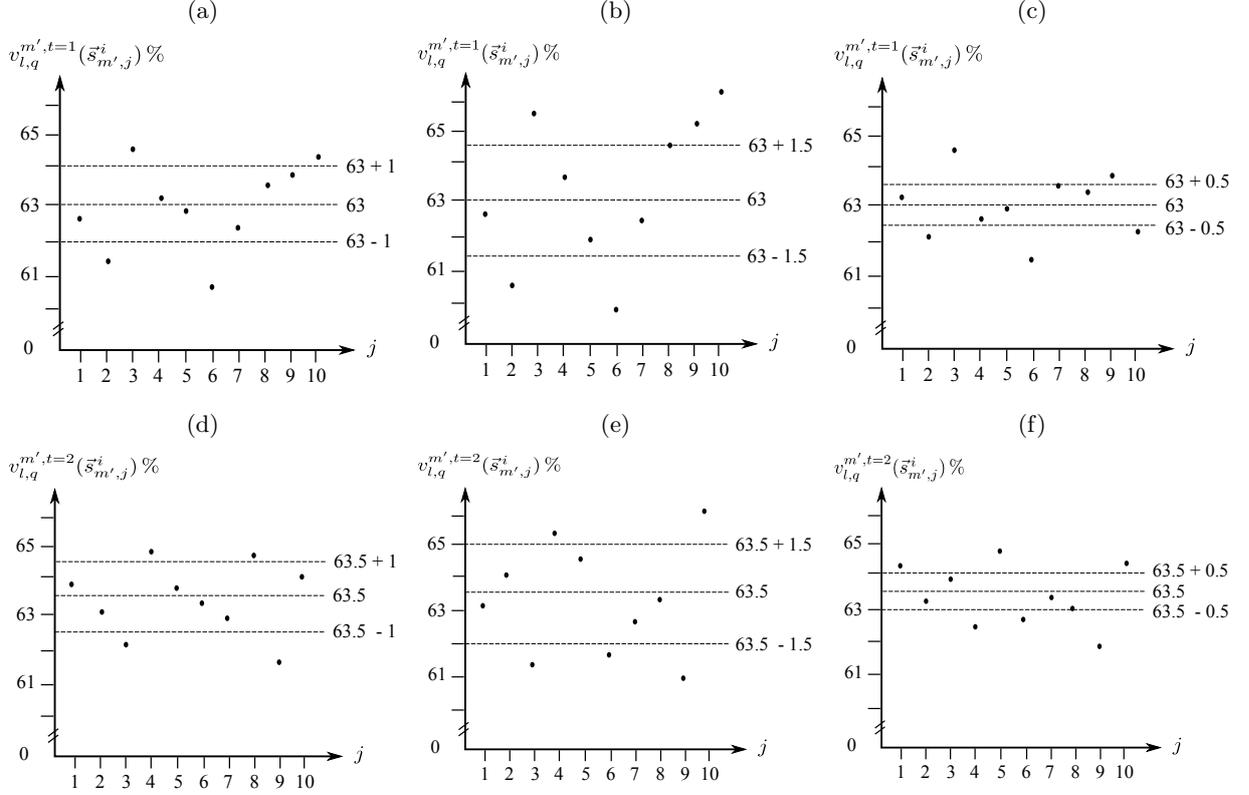
Let $\vec{s}_{m,j}^i \in \Omega_m^i$ denote the j^{th} schedule generated by \mathcal{O}_m , in iteration i , and $v_{l,q}^{m,t}(\vec{s}_{m,j}^i)$ the value of variable $v_{l,q}^{m,t}$ in schedule $\vec{s}_{m,j}^i$. Given a mine $m' \in \mathcal{M}$, a grade and quality target $\vec{\phi}_{m'}^{i=1}$, and a vector of standard deviations $\vec{\sigma}_{m'}^{i=1}$, Example 5.1 presents the first schedule generated by $\mathcal{O}_{m'}$ in iteration $i = 1$, $\vec{s}_{m',j=1}^{i=1}$. Examples 5.3–5.7 demonstrate how this schedule is formed.

Example 5.1 *Mine m' produces ore of a single granularity l , characterised by a single attribute q (metal grade), over three periods $t = 1, 2, 3$. For all t , $\phi_{l,q}^{m',t,i=1}$ is 62.5%, and $\sigma_{l,q}^{m',t,i=1}$ is 0.4. Mine m' contains grade blocks $\mathcal{B}_{m'}^g = \{b_0, b_1, b_2, b_3, b_4\}$, blast blocks $\mathcal{B}_{m'}^b = \{b_5, b_6, \dots, b_{14}\}$, and a waste dump δ . For each block $b \in \mathcal{B}_m \setminus \{b_7, b_8, b_9\}$, $T_b^{hi} = 10kt$. Moreover, $G_{b,l,q}^{m',hi} = 63\%$ for $b = b_0, b_5, b_6$; $G_{b,l,q}^{m',hi} = 62\%$ for $b = b_1, b_2, b_{10}$; $G_{b,l,q}^{m',hi} = 62.5\%$ for $b = b_3, b_{11}, b_{12}$; and $G_{b,l,q}^{m',hi} = 63.5\%$ for $b = b_4, b_{13}, b_{14}$. Blocks b_7, b_8 , and b_9 , each contain 20kt of waste, and no low or high grade ore. The capacity of the dry plant κ at m' is 20kt (per period). Mine m' can extract up to 20kt of material in $t = 1$, and 30kt in periods $t = 2, 3$. The first schedule generated by $\mathcal{O}_{m'}$ in iteration $i = 1$ is shown below. Schedule $\vec{s}_{m',j=1}^{i=1}$ produces ore of grade 62.88%, 62.4%, and 62.98%, in periods 1, 2, and 3, respectively.*

$$\vec{s}_{m',j=1}^{i=1} = \begin{cases} x_{b_0,\kappa}^{m',t=1} = 2kt & x_{b_0,\kappa}^{m',t=2} = 8kt & x_{b_2,\kappa}^{m',t=3} = 7kt \\ x_{b_2,\kappa}^{m',t=1} = 1kt & x_{b_1,\kappa}^{m',t=2} = 10kt & x_{b_4,\kappa}^{m',t=3} = 3kt \\ x_{b_3,\kappa}^{m',t=1} = 10kt & x_{b_2,\kappa}^{m',t=2} = 2kt & x_{b_{13},\kappa}^{m',t=3} = 10kt \\ x_{b_4,\kappa}^{m',t=1} = 7kt & x_{b_7,\delta}^{m',t=2} = 10kt & x_{b_8,\delta}^{m',t=3} = 10kt \end{cases}$$

For $i = 1$, $\vec{\phi}_m^{i=1}$ is initialised with an expected target, denoted Ξ_m , derived from a medium-term (five year) plan. This plan provides a target composition for the ore mine m should produce in each period t of the short-term horizon, to ensure correct blending at the ports. We initialise $\vec{\sigma}_m^{i=1}$ with large enough values to ensure significant diversity, across Ω_m^i , in the grade of produced ore.

Figure 2 A set of $N = 10$ schedules for mine m' , producing a single granularity l , characterised by one quality attribute q , over two periods, formed with varying $\bar{\phi}_{m'}^i$ and $\bar{\sigma}_{m'}^i$. For $\phi_{l,q}^{m',t=1,i} = 63\%$, and $\phi_{l,q}^{m',t=2,i} = 63.5\%$, (a) & (d) plot the value of variable $v_{l,q}^{m',t}$, across schedules, when $\sigma_{l,q}^{m',t,i} = 1$, for all t . Similarly, (b) & (e), and (c) & (f), depict the value of $v_{l,q}^{m',t}$, across schedules, when $\sigma_{l,q}^{m',t,i} = 1.5$, and $\sigma_{l,q}^{m',t,i} = 0.5$, respectively, for all t .



Example 5.2 Consider mine m' of Example 5.1, in periods $t = 1, 2$. Figure 2 shows the value of $v_{l,q}^{m',t}(\bar{s}_{m',j}^i)$, across 10 possible schedules, for differing targets $\bar{\phi}_{m'}^i$ and standard deviations $\bar{\sigma}_{m'}^i$. Larger values for each $\sigma_{l,q}^{m',t,i}$ encourages the creation of a schedule set, across which, the percentage of q in produced ore exhibits a greater range of values. Smaller values for each $\sigma_{l,q}^{m',t,i}$ results in schedules, across which, the percentage of q in produced ore is more tightly clustered about $\bar{\phi}_{m'}^i$.

To construct $\bar{s}_{m',j}^i$, for each $j = 1..N$, we first define desired upper and lower bounds on each $v_{l,q}^{m',t}(\bar{s}_{m',j}^i)$, denoted $[L_{l,q}^{m',t,j}, U_{l,q}^{m',t,j}]$, via Algorithm 2. A normally distributed random value $\Delta_{l,q}^{j,t}$, for each $j \in \{1..N\}$, $t \in \{1, 2, \dots, T\}$, $l \in \mathcal{L}$, and $q \in \mathcal{Q}$, is generated from a distribution with mean 0 and standard deviation $\sigma_{l,q}^{m',t,i}$ (Step 4). Shifting $\phi_{l,q}^{m',t,i}$ by $\Delta_{l,q}^{j,t}$, and subtracting (adding) a small quantity, Δ_q^- , to the result,² computes our lower $L_{l,q}^{m',t,j}$ (and upper $U_{l,q}^{m',t,j}$) bound (Steps 5–6). By forming a set of varying bounds on the grade and quality of production in each period t , we are able to generate multiple schedules that produce varying grades of ore in each period t .

² Δ_q^- denotes an insignificant change in the percentage of $q \in \mathcal{Q}$ in an orebody.

Algorithm 2 Generation of bounds on the blend of produced ore at mine $m \in \mathcal{M}$.

```

1: for  $j \leftarrow 1$  to  $N$  do
2:   for  $t \leftarrow 1$  to  $T$  do
3:     for each  $l \in \mathcal{L}$  and  $q \in \mathcal{Q}$  do
4:        $\Delta_{l,q}^{j,t} \leftarrow \text{RANDNORMAL}(0, \sigma_{l,q}^{m,t})$  where  $\sigma_{l,q}^{m,t} \in \vec{\sigma}_m$ 
5:        $L_{l,q}^{m,t,j} \leftarrow \phi_{l,q}^{m,t} + \Delta_{l,q}^{j,t} - \Delta_q^-$ 
6:        $U_{l,q}^{m,t,j} \leftarrow \phi_{l,q}^{m,t} + \Delta_{l,q}^{j,t} + \Delta_q^-$ 
7:     end for
8:   end for
9: end for

```

Example 5.3 (Example 5.1 cont.) Let $\Delta_q^- = 0.01\%$. Algorithm 2, for $j = 1$, computes $\Delta_{l,q}^{j,t=1} = 0.3$, $\Delta_{l,q}^{j,t=2} = -0.2$, and $\Delta_{l,q}^{j,t=3} = 0.5$. The following bounds on the percentage of q in ore produced by m' in periods $t = 1, 2, 3$ are formed. Recall that $\phi_{l,q}^{m',t,i=1} = 62.5\%$ and $\sigma_{l,q}^{m',t,i=1} = 0.4$ for all t .

$$\begin{array}{lll}
L_{l,q}^{m',t=1,j=1} = 62.7\% & L_{l,q}^{m',t=2,j=1} = 62.2\% & L_{l,q}^{m',t=3,j=1} = 62.9\% \\
U_{l,q}^{m',t=1,j=1} = 62.9\% & U_{l,q}^{m',t=2,j=1} = 62.4\% & U_{l,q}^{m',t=3,j=1} = 63.1\%
\end{array}$$

We apply a two-stage process to generate a schedule, $\vec{s}_{m,j}^i$, for which the deviation between each $v_{l,q}^{m,t}(\vec{s}_{m,j}^i)$ and bounds $[L_{l,q}^{m,t,j}, U_{l,q}^{m,t,j}]$ is minimised, as a first priority, and total productivity at mine m maximised, as a second. The productivity achieved at mine m in period t of $\vec{s}_{m,j}^i$, denoted $\rho_m(\vec{s}_{m,j}^i, t)$, is computed as per Equation 3. In the first stage of this process, we aggregate blocks in the grade and blast block models of each mine into larger units of extraction, given a maximum aggregate size of M_A blocks. The set of block aggregates at mine m is denoted \mathcal{B}_m^A , where $IN(a)$ denotes the subset of \mathcal{B}_m in aggregate $a \in \mathcal{B}_m^A$ ($IN(a) \subset \mathcal{B}_m$). The procedure used to create these block aggregates is, for brevity, omitted from this paper. A rolling horizon heuristic, described in Algorithm 3 and Section 5.2, generates a block extraction schedule – identifying which of these aggregated blocks are to be mined in each period t , and the destination of this mined material.

Let $\tilde{s}_{m,j}^i$ and $\overline{\mathcal{B}}_m^A$ respectively denote the schedule generated by our rolling horizon heuristic, given blocks \mathcal{B}_m^A , and the set of aggregates in \mathcal{B}_m^A that have been mined (partially or completely) in $\tilde{s}_{m,j}^i$. In the second stage of our method of mine schedule generation, we reapply the rolling horizon heuristic of Section 5.2 with respect to a new set of blocks, $\mathcal{B}_m^* = \mathcal{B}_m^g \cup \{b | b \in \mathcal{B}_m^b \wedge \exists a \in \overline{\mathcal{B}}_m^A. b \in IN(a)\}$ to generate schedule $\vec{s}_{m,j}^i$. The set \mathcal{B}_m^* denotes the union of the set of grade blocks \mathcal{B}_m^g , with the set of all blast blocks in \mathcal{B}_m^b that appear in a mined aggregate $a \in \overline{\mathcal{B}}_m^A$. Our first application of the rolling horizon heuristic seeks to reduce the complexity of the problem of generating $\vec{s}_{m,j}^i$, by reducing

the number of blocks that can be considered for mining. The result is an efficient method for the generation of extraction schedules, given a large number of available blocks.

We show in Section 6 that the use of an initial scheduling pass to reduce the number of blocks under consideration in the generation of schedules by each \mathcal{O}_m , substantially reduces the runtime of our algorithm, while preserving, on average, the quality of solutions found.

Example 5.4 (Example 5.3 cont.) *Let $M_A = 2$. We form the following aggregates of blocks in $\mathcal{B}_{m'}$: $a_0 = \{b_0, b_3\}$; $a_1 = \{b_1, b_2\}$; $a_2 = \{b_4\}$; $a_3 = \{b_5, b_6\}$; $a_4 = \{b_{10}\}$; $a_5 = \{b_{11}, b_{12}\}$; $a_6 = \{b_{13}, b_{14}\}$; and waste aggregates $a_7 = \{b_7, b_8\}$, and $a_8 = \{b_9\}$. The percentage of q in a_{0-6} is 62.75%, 62%, 63.5%, 63%, 62%, 62.5%, and 63.5%, respectively. Examples 5.5–5.7 show how our rolling horizon heuristic generates $\tilde{s}_{m',j=1}^{i=1}$, scheduling the mining of aggregates $\mathcal{B}_{m'}^A = \{a_0, \dots, a_8\}$ given the bounds on grade shown in Example 5.3. In $\tilde{s}_{m',j=1}^{i=1}$, aggregates a_{0-2} and a_{6-7} are mined. Thus, $\overline{\mathcal{B}_{m'}^A} = \{a_0, a_1, a_2, a_6, a_7\}$ and $\mathcal{B}_{m'}^* = \{b_0, b_1, b_2, b_3, b_4, b_7, b_8, b_{13}, b_{14}\}$. Reapplication of the rolling horizon heuristic, restricted to the mining of blocks in $\mathcal{B}_{m'}^*$, yields $\tilde{s}_{m',j=1}^{i=1}$, shown in Example 5.1.*

5.2. A Rolling Horizon Heuristic

Algorithm 3 presents a rolling horizon heuristic for the generation of an extraction schedule, \tilde{s}_m , for a single mine $m \in \mathcal{M}$. Given a set of sources \mathcal{S} , containing blocks $\mathcal{B} \subset \mathcal{S}$ and stockpiles $\Theta_m \cup \Lambda_m \subset \mathcal{S}$, a set of destinations \mathcal{D}_m , T time periods, and bounds on the composition of production, $[L_{l,q}^{m,t}, U_{l,q}^{m,t}]$ for all $l \in \mathcal{L}$, $q \in \mathcal{Q}$, and $t \in \{1, 2, \dots, T\}$, \tilde{s}_m is formed by solving $T - 1$ two-time-period MIPs, and a final one-time-period MIP. Starting at $t' = 1$, the horizon is decomposed into two periods, h_1 and h_2 , of size 1 and $T - t'$ (Steps 1–2). A two-time-period extraction schedule is formed (Steps 7–11), and the activities of h_1 become part of schedule \tilde{s}_m (Step 12). This process is repeated for $t' \in \{2, 3, \dots, T\}$ (Steps 15–16). The heuristic terminates, returning \tilde{s}_m , after period T is scheduled.

Given time periods, h_1 and h_2 , a two-time-period extraction schedule is constructed as follows. In Step 7, the bounds $[L_{l,q}^{m,t}, U_{l,q}^{m,t}]$, for each $l \in \mathcal{L}$, $q \in \mathcal{Q}$, and $t \in \{t', \dots, T\}$, are averaged across the periods represented by h_1 and h_2 , to form new bounds $[L_{l,q}^{m,h}, U_{l,q}^{m,h}]$ for all combinations of $l \in \mathcal{L}$, $q \in \mathcal{Q}$, and $h \in \{h_1, h_2\}$. For $t' = 1$, h_1 denotes period 1, and h_2 periods $\{2, 3, \dots, T\}$. Let $\tilde{x}_{s,d}^{m,h}$ denote the tons of source $s \in \mathcal{S}$ hauled to destination $d \in \mathcal{D}_m$ in period h and $\tilde{\mathbf{x}}_m = \{\tilde{x}_{s,d}^{m,h} \mid s \in \mathcal{S}, d \in \mathcal{D}_m, h \in \{h_1, h_2\}\}$. A schedule for periods h_1 and h_2 is formed by solving two MIPs, $\mathcal{O}_{m,1}$ and $\mathcal{O}_{m,2}$, in succession, the objectives of which, denoted $Z_{\mathcal{O}_{m,1}}$ and $Z_{\mathcal{O}_{m,2}}$, are defined in Equations (20)–(21).

Example 5.5 (Example 5.4 cont.) *To generate $\tilde{s}_{m',j=1}^{i=1}$ in Example 5.4 we first assign period $t = 1$ to h_1 , and periods $t = 2, 3$ to h_2 . The bounds shown in Example 5.3 for periods $t = 2, 3$ are averaged to form bounds on the percentage of q in the ore produced at m' in h_1 and h_2 .*

Algorithm 3 Generation of schedule \tilde{s}_m for mine $m \in \mathcal{M}$, given a set of material sources \mathcal{S} (including a set of blocks $\mathcal{B} \subset \mathcal{S}$), material destinations \mathcal{D}_m , T time periods, and bounds $[L_{l,q}^{m,t}, U_{l,q}^{m,t}]$, on each $q \in \mathcal{Q}$ in granularity $l \in \mathcal{L}$ produced at m in each period $t \in \{1, 2, \dots, T\}$.

- 1: $h_1 \leftarrow 1$
 - 2: $h_2 \leftarrow \{2, 3, \dots, T\}$
 - 3: Initialise block tonnages $T_k^{t=1}(b)$ for all $k \in \{hi, lo, w\}$ and $b \in \mathcal{B}$.
 - 4: Initialise stockpile tonnages $T^{t=1}(s)$, ore splits $S_{s,l}^{t=1}$, and grades $G_{s,l,q}^{t=1}$ for all $s \in \Theta_m \cup \Lambda_m$, $l \in \mathcal{L}$, and $q \in \mathcal{Q}$.
 - 5: $\tilde{s}_m \leftarrow \emptyset$
 - 6: **while** $h_1 \leq T$ **do**
 - 7: Compute bounds on the percentage of $q \in \mathcal{Q}$ in granularity $l \in \mathcal{L}$ to be produced at m in periods h_1 and h_2 , denoted $[L_{l,q}^{m,h}, U_{l,q}^{m,h}]$, by averaging $[L_{l,q}^{m,t}, U_{l,q}^{m,t}]$ over $t = h_1$, and (if $h_1 < T$) $t = h_1 + 1$ to $t = T$.
 - 8: Solve a MIP, denoted $\mathcal{O}_{m,1}$, to schedule the extraction of blocks in \mathcal{B} across time periods h_1 and h_2 , while minimising deviation between the chemistry of produced ore and desired bounds.
 - 9: If a solution to $\mathcal{O}_{m,1}$ could not be found, terminate and **return** \emptyset .
 - 10: Minimally shift $L_{l,q}^{m,h}$ or $U_{l,q}^{m,h}$ for each $q \in \mathcal{Q}$, $l \in \mathcal{L}$, and $h \in \{h_1, h_2\}$ to cover the solution obtained to $\mathcal{O}_{m,1}$. Incorporate these bounds into $\mathcal{O}_{m,1}$, as hard constraints, forming a new MIP, $\mathcal{O}_{m,2}$.
 - 11: Alter the objective of $\mathcal{O}_{m,2}$ to maximise productivity at mine m , and solve $\mathcal{O}_{m,2}$ to schedule the extraction of blocks in \mathcal{B} across time periods h_1 and h_2 .
 - 12: Fix the values of variables $\tilde{x}_{s,d}^{m,t=h_1}$ in schedule \tilde{s}_m , for each $s \in \mathcal{S}$ and $d \in \mathcal{D}_m$, to those of corresponding variables $\tilde{x}_{s,d}^{m,h_1}$ in the solution obtained to $\mathcal{O}_{m,2}$, if found, and to $\mathcal{O}_{m,1}$, otherwise.
 - 13: Compute remaining block tonnages $T_k^{t=h_1+1}(b)$ for all $k \in \{hi, lo, w\}$ and $b \in \mathcal{B}$.
 - 14: Compute updated stockpile tonnages $T^{t=h_1+1}(s)$, ore splits $S_{s,l}^{t=h_1+1}$, and grades $G_{s,l,q}^{t=h_1+1}$ for all $s \in \Theta_m \cup \Lambda_m$, $l \in \mathcal{L}$, and $q \in \mathcal{Q}$.
 - 15: $h_1 \leftarrow h_1 + 1$
 - 16: $h_2 \leftarrow \{h_1 + 1, \dots, T\}$ if $h_1 < T$ and \emptyset otherwise.
 - 17: **end while**
 - 18: **return** \tilde{s}_m
-

$$\begin{array}{ll} L_{l,q}^{m',h_1} = 62.7\% & L_{l,q}^{m',h_2} = 62.55\% \\ U_{l,q}^{m',h_1} = 62.9\% & U_{l,q}^{m',h_2} = 62.75\% \end{array}$$

The objective of $\mathcal{O}_{m,1}$, denoted $Z_{\mathcal{O}_{m,1}}$, acts to minimise deviation between the composition of each granularity $l \in \mathcal{L}$ formed at mine m in each period $h \in \{h_1, h_2\}$ and desired bounds $[L_{l,q}^{m,h}, U_{l,q}^{m,h}]$, for all $q \in \mathcal{Q}$ (Step 8). A measure of the extent of this deviation is denoted $\eta_m(\tilde{\mathbf{x}}_m, h)$, and is computed as shown in Equation (22), where $\nu_{l,q}^m(\tilde{\mathbf{x}}_m, h)$ denotes the tons of attribute $q \in \mathcal{Q}$ in the ore of granularity l formed by m in period h (see Equation 25), and $\tau_l^m(\tilde{\mathbf{x}}_m, h)$ the tons of granularity l

formed by m in h (see Equation 24). Constraints (39)–(41), (44)–(51), (58)–(62), and (67), from the MINLP of Appendix B, form the constraints of $\mathcal{O}_{m,1}$, redefined for a single mine m in terms of variables $\tilde{\mathbf{x}}_m$. In both $\mathcal{O}_{m,1}$ and $\mathcal{O}_{m,2}$, the composition of each low and high grade stockpile s at m is assumed to have a constant split and grade, denoted $S_{s,l}^{t=h_1}$ and $G_{s,l,q}^{t=h_1}$, for granularity $l \in \mathcal{L}$ and attribute $q \in \mathcal{Q}$, across periods h_1 and h_2 . We place a time limit on the solving of $\mathcal{O}_{m,1}$ and $\mathcal{O}_{m,2}$, accepting the best solution found by the time this limit is reached. If $\mathcal{O}_{m,1}$ is found to be infeasible, or no solution is found within the prescribed time limit, the heuristic terminates in failure (Step 9). Across the experiments conducted in Section 6, our rolling horizon heuristic fails 3.7% (with a standard deviation of 1.4%) of the time, on average.

$$Z_{\mathcal{O}_{m,1}} = \min [\eta_m(\tilde{\mathbf{x}}_m, h_1) + \eta_m(\tilde{\mathbf{x}}_m, h_2)] \quad (20)$$

$$Z_{\mathcal{O}_{m,2}} = \max [\rho_m(\tilde{\mathbf{x}}_m, h_1) + \rho_m(\tilde{\mathbf{x}}_m, h_2)] \quad (21)$$

$$\begin{aligned} \eta_m(\tilde{\mathbf{x}}_m, h) = & \sum_{l \in \mathcal{L}} \sum_{q \in \mathcal{Q}} \frac{1}{\Delta_q^+} \max [0, \nu_{l,q}^m(\tilde{\mathbf{x}}_m, h) - U_{l,q}^{m,h} \tau_l^m(\tilde{\mathbf{x}}_m, h)] \\ & + \sum_{l \in \mathcal{L}} \sum_{q \in \mathcal{Q}} \frac{1}{\Delta_q^+} \max [0, L_{l,q}^{m,h} \tau_l^m(\tilde{\mathbf{x}}_m, h) - \nu_{l,q}^m(\tilde{\mathbf{x}}_m, h)] \end{aligned} \quad (22)$$

$$\rho_m(\tilde{\mathbf{x}}_m, t) = \sum_{s \in \mathcal{S}} \left[\sum_{\delta \in \Delta_m} \tilde{x}_{s,\delta}^{m,h} - \sum_{\theta \in \Theta_m} \tilde{x}_{s,\theta}^{m,h} + (1 - 2\Phi_\omega^m) \sum_{\lambda \in \Lambda_m} \tilde{x}_{s,\lambda}^{m,h} \right] \quad (23)$$

$$\begin{aligned} \tau_l^m(\tilde{\mathbf{x}}_m, h) = & \sum_{\lambda \in \Lambda_m} S_{\lambda,l}^{t=h_1} Y_l^{m,\omega} \tilde{x}_{\lambda,\omega}^{m,h} + \sum_{\theta \in \Theta_m} S_{\theta,l}^{t=h_1} \tilde{x}_{\theta,\kappa}^{m,h} \\ & + \sum_{b \in \mathcal{B}} S_{b,l}^{m,hi} \tilde{x}_{b,\kappa}^{m,h} + S_{b,l}^{m,lo} Y_l^{m,\omega} \tilde{x}_{b,\omega}^{m,h} \end{aligned} \quad \forall h \in \{h_1, h_2\}, l \in \mathcal{L}, \quad (24)$$

$$\begin{aligned} \nu_{l,q}^m(\tilde{\mathbf{x}}_m, h) = & \sum_{\lambda \in \Lambda_m} S_{\lambda,l}^{t=h_1} G_{\lambda,l,q}^{t=h_1} R_{l,q}^{m,\omega} \tilde{x}_{\lambda,\omega}^{m,h} \\ & + \sum_{\theta \in \Theta_m} S_{\theta,l}^{t=h_1} G_{\theta,l,q}^{t=h_1} \tilde{x}_{\theta,\kappa}^{m,h} \\ & + \sum_{b \in \mathcal{B}} S_{b,l}^{m,hi} G_{b,l,q}^{m,hi} \tilde{x}_{b,\kappa}^{m,h} + S_{b,l}^{m,lo} G_{b,l,q}^{m,lo} R_{l,q}^{m,\omega} \tilde{x}_{b,\omega}^{m,h} \end{aligned} \quad \forall h \in \{h_1, h_2\}, l \in \mathcal{L}, q \in \mathcal{Q}. \quad (25)$$

The objective of $\mathcal{O}_{m,2}$, denoted $Z_{\mathcal{O}_{m,2}}$, acts to maximise the productivity of mine m across periods h_1 and h_2 , denoted by $\rho_m(\tilde{\mathbf{x}}_m, h_1) + \rho_m(\tilde{\mathbf{x}}_m, h_2)$ in Equation (21), given hard constraints on the percentage of each attribute $q \in \mathcal{Q}$ in the ore of each granularity $l \in \mathcal{L}$ produced at m in each period (Steps 10–11). Let $\tilde{Q}_{l,q}^{m,h}$ denote the percentage of q in granularity l produced in period h by m in

the solution found to $\mathcal{O}_{m,1}$. The $\mathcal{O}_{m,2}$ MIP applies the constraints of $\mathcal{O}_{m,1}$, with the addition of Constraints (26)–(27), shown below (Step 10). With these additional constraints, $\mathcal{O}_{m,2}$ ensures that the solution found to $\mathcal{O}_{m,1}$ remains a solution of $\mathcal{O}_{m,2}$, and that deviation between the composition of production and desired bounds is limited to that present in the solution to $\mathcal{O}_{m,1}$.

$$\nu_{l,q}^m(\tilde{\mathbf{x}}_{\mathbf{m}}, h) \geq \min[\tilde{Q}_{l,q}^{m,h}, L_{l,q}^{m,h}] \tau_l^m(\tilde{\mathbf{x}}_{\mathbf{m}}, h) \quad \forall h \in \{h_1, h_2\}, l \in \mathcal{L}, q \in \mathcal{Q}, \quad (26)$$

$$\nu_{l,q}^m(\tilde{\mathbf{x}}_{\mathbf{m}}, h) \leq \max[\tilde{Q}_{l,q}^{m,h}, U_{l,q}^{m,h}] \tau_l^m(\tilde{\mathbf{x}}_{\mathbf{m}}, h) \quad \forall h \in \{h_1, h_2\}, l \in \mathcal{L}, q \in \mathcal{Q}. \quad (27)$$

Example 5.6 (Example 5.5 cont.) Consider the set of aggregates, $\mathcal{B}_{m'}^A$, in Example 5.4. Solving $\mathcal{O}_{m',1}$, with respect to the bounds of Example 5.5, leads to the following instantiation of $\tilde{\mathbf{x}}_{\mathbf{m}'}$.

$$\tilde{\mathbf{x}}_{\mathbf{m}'} = \{\tilde{x}_{a_0,\kappa}^{m',h_1} = 20kt, \tilde{x}_{a_1,\kappa}^{m',h_2} = 20kt, \tilde{x}_{a_2,\kappa}^{m',h_2} = 10kt, \tilde{x}_{a_6,\kappa}^{m',h_2} = 10kt\}$$

In this solution to $\mathcal{O}_{m',1}$, $\tilde{Q}_{l,q}^{m',h_1} = \tilde{Q}_{l,q}^{m',h_2} = 62.75\%$. Constraints (26)–(27) in $\mathcal{O}_{m',2}$ are instantiated as shown in Equations (28)–(29).

$$0.627 \tau_l^{m'}(\tilde{\mathbf{x}}_{\mathbf{m}'}, h_1) \leq \nu_{l,q}^{m'}(\tilde{\mathbf{x}}_{\mathbf{m}'}, h_1) \leq 0.629 \tau_l^{m'}(\tilde{\mathbf{x}}_{\mathbf{m}'}, h_1) \quad (28)$$

$$0.6255 \tau_l^{m'}(\tilde{\mathbf{x}}_{\mathbf{m}'}, h_2) \leq \nu_{l,q}^{m'}(\tilde{\mathbf{x}}_{\mathbf{m}'}, h_2) \leq 0.6275 \tau_l^{m'}(\tilde{\mathbf{x}}_{\mathbf{m}'}, h_2) \quad (29)$$

The solution to problem $\mathcal{O}_{m',2}$ extracts and processes the same high grade material in periods h_1 and h_2 as does that of $\mathcal{O}_{m',1}$. To maximise productivity, waste aggregate a_7 is extracted in h_2 .

$$\tilde{\mathbf{x}}_{\mathbf{m}'} = \{\tilde{x}_{a_0,\kappa}^{m',h_1} = 20kt, \tilde{x}_{a_1,\kappa}^{m',h_2} = 20kt, \tilde{x}_{a_2,\kappa}^{m',h_2} = 10kt, \tilde{x}_{a_6,\kappa}^{m',h_2} = 10kt, \tilde{x}_{a_7,\delta}^{m',h_2} = 20kt\}$$

For each $s \in \mathcal{S}$ and $d \in \mathcal{D}_m$, we fix $\tilde{x}_{s,d}^{m,t=1}$ to the value of $\tilde{x}_{s,d}^{m,h_1}$ in the solution found to $\mathcal{O}_{m,2}$ (Step 12). If no such solution is found in the time limit allocated to $\mathcal{O}_{m,2}$, each $\tilde{x}_{s,d}^{m,t=1}$ is fixed to the value of $\tilde{x}_{s,d}^{m,h_1}$ in the solution found to $\mathcal{O}_{m,1}$. We set $h_1 = 2$ and $h_2 = \{3, 4, \dots, T\}$ (Steps 15–16), and reconstruct $\mathcal{O}_{m,1}$ and $\mathcal{O}_{m,2}$. Each $T_k^{t=1}(b)$, for $k = \{hi, lo, w\}$ and $b \in \mathcal{B}$, appearing in the constraints of $\mathcal{O}_{m,1}$ and $\mathcal{O}_{m,2}$ is replaced with $T_k^{t=2}(b)$ (computed in Step 13). Constants $T^{t=1}(s)$, $S_{s,l}^{m,t=1}$, and $G_{s,l,q}^{m,t=1}$, for each stockpile $s \in \Theta_m \cup \Lambda_m$, granularity $l \in \mathcal{L}$, and attribute $q \in \mathcal{Q}$, are replaced with $T^{t=2}(s)$, $S_{s,l}^{t=2}$, and $G_{s,l,q}^{t=2}$ (computed in Step 14). The computation of updated block tonnages, and stockpile compositions, is straightforward, and thus explicit equations are omitted from this paper.

Example 5.7 (Example 5.6 cont.) We fix the mining and processing of aggregate a_0 in period $t = 1$ of schedule $\tilde{s}_{m',j=1}^{i=1}$, and set $T_{hi}^{t=2}(a_0) = 0$. Periods h_1 and h_2 are redefined ($h_1 = 2$ and $h_2 = \{3\}$) and problems $\mathcal{O}_{m',1}$ and $\mathcal{O}_{m',2}$ solved to complete schedule $\tilde{s}_{m',j=1}^{i=1}$.

$$\tilde{s}_{m',j=1}^{i=1} = \begin{cases} \tilde{x}_{a_0,\kappa}^{m,t=1} = 20kt & \tilde{x}_{a_1,\kappa}^{m,t=2} = 17kt & \tilde{x}_{a_1,\kappa}^{m,t=3} = 3kt \\ \tilde{x}_{a_2,\kappa}^{m,t=2} = 3kt & \tilde{x}_{a_2,\kappa}^{m,t=3} = 7kt & \\ \tilde{x}_{a_7,\delta}^{m,t=2} = 10kt & \tilde{x}_{a_6,\kappa}^{m,t=3} = 10kt & \tilde{x}_{a_7,\delta}^{m,t=3} = 10kt \end{cases}$$

5.3. The \mathcal{O}_Π Problem

In each iteration i , \mathcal{O}_Π is given a set of (up to) N block extraction schedules for each mine $m \in \mathcal{M}$, denoted Ω_m^i . \mathcal{O}_Π selects one schedule in the set $\Omega_m^i \cup \{\vec{s}_{best,m}\}$, to be enacted, where $\vec{s}_{best,m} \in \vec{s}_{best}$ denotes the schedule to be implemented at mine m in the best solution found by the algorithm, \vec{s}_{best} , across all iterations prior to i . The j^{th} schedule available for selection at mine m is denoted $\vec{s}_{m,j} \in \Omega_m^i \cup \{\vec{s}_{best,m}\}$. Ore railed from each mine m to each port π must originate from only one $\vec{s}_{m,j}$. Let $v_{l,q}^{m,t}(\vec{s}_{m,j})$ denote the percentage of attribute $q \in \mathcal{Q}$ in granularity $l \in \mathcal{L}$ produced by mine m in period t of schedule $\vec{s}_{m,j}$; $\rho_m(\vec{s}_{m,j}, t)$ the productivity of m in period t of schedule $\vec{s}_{m,j}$, as per Equation (3); P_j^m the total productivity achieved at m by $\vec{s}_{m,j}$ (the sum of $\rho_m(\vec{s}_{m,j}, t)$ over all t); $\tau_l^m(\vec{s}_{m,j}, t)$ the tons of $l \in \mathcal{L}$ produced by mine m in period t of $\vec{s}_{m,j}$; and $N_m = |\Omega_m^i \cup \{\vec{s}_{best,m}\}|$.

We now present a MIP formulation of the \mathcal{O}_Π subproblem. In contrast to the instantiation of \mathcal{O}_Π presented by Blom et al. (2014), a continuous variable $r_{\pi,n,t}^{m,l,j,t'}$ denotes the number of trainloads of granularity $l \in \mathcal{L}$, produced by mine m in period t' of schedule $\vec{s}_{m,j} \in \Omega_m^i \cup \{\vec{s}_{best,m}\}$, contributing to product $n \in N_{\pi,l}$ at port π in period t . Variable $s_{m,j}$ is assigned a value of 1 if and only if schedule $\vec{s}_{m,j}$ is chosen for implementation at mine m , and 0 otherwise. The objective of the port-side MIP is twofold. As a first priority, deviation between the composition of port products and desired bounds, denoted $[L_{n,q}^{\pi,l,t}, U_{n,q}^{\pi,l,t}]$ for product $n \in N_{\pi,l}$ of granularity $l \in \mathcal{L}$ at port π in period t , is minimised. The total productivity achieved across the network of mines is maximised, over the scheduling horizon, as a second priority. The objective of the port-side MIP, $Z_{\mathcal{O}_\Pi}$, is defined in Equation (30), where: β_1 and β_2 are constants such that $\beta_1 \gg \beta_2$; $f(t) = \max(1, t - 1)$ and $g(t) = \min(T, t + 1)$ for $t \in \{1, 2, \dots, T\}$; \vec{r}^j denotes the set of all $r_{\pi,n,t}^{m,l,j,t'}$ variables for $m \in \mathcal{M}$, $l \in \mathcal{L}$, $j \in \{1, N_m\}$, $\pi \in \Pi$, $n \in N_{\pi,l}$, $t \in \{0, 1, \dots, T\}$, and $t' \in \{f(t), t\}$; and $\eta(\vec{r}^j, t)$ the total deviation present between port products and desired bounds, in period t , as defined in Equation (31).

The domains of variables $r_{\pi,n,t}^{m,l,j,t'}$ and $s_{m,j}$ are respectively restricted to non-negative reals, and binary values.

$$Z_{\mathcal{O}_\pi} = \min \sum_{t=1}^T \left[\beta_1 \eta(\vec{r}^t, t) - \beta_2 \sum_{m \in \mathcal{M}} \sum_{j=1}^{N_m} s_{m,j} P_i^m \right] \quad (30)$$

$$\begin{aligned} \eta(\vec{r}^t, t) = & \sum_{\substack{\pi \in \Pi, l \in \mathcal{L} \\ n \in N_{\pi,l} \\ q \in \mathcal{Q}}} \frac{T_R}{\Delta^+} \max \left[0, \sum_{m \in \mathcal{M}} \sum_{j=1}^{N_m} \sum_{t'=f(t)}^t r_{\pi,n,t}^{m,l,j,t'} v_{l,q}^{m,t'}(\vec{s}_{m,j}) - U_{n,q}^{\pi,l,t} \sum_{m \in \mathcal{M}} \sum_{j=1}^{N_m} \sum_{t'=f(t)}^t r_{\pi,n,t}^{m,l,j,t'} \right] \\ & + \sum_{\substack{\pi \in \Pi, l \in \mathcal{L} \\ n \in N_{\pi,l} \\ q \in \mathcal{Q}}} \frac{T_R}{\Delta^+} \max \left[0, L_{n,q}^{\pi,l,t} \sum_{m \in \mathcal{M}} \sum_{j=1}^{N_m} \sum_{t'=f(t)}^t r_{\pi,n,t}^{m,l,j,t'} - \sum_{m \in \mathcal{M}} \sum_{j=1}^{N_m} \sum_{t'=f(t)}^t r_{\pi,n,t}^{m,l,j,t'} v_{l,q}^{m,t'}(\vec{s}_{m,j}) \right] \quad (31) \end{aligned}$$

Constraint (32) ensures that only one schedule is selected to be implemented at each mine m . Port capacities are enforced by Constraint (33). Constraint (34) ensures that all ore produced by each mine m , in each period t , is railed to a port by period $t + 1$. We assume that the stockyards at each mine are empty at the start of $t = 1$, and that they must be emptied by the end of $t = T$.

$$\sum_{j=1}^{N_m} s_{m,j} = 1 \quad \forall m \in \mathcal{M}, \quad (32)$$

$$\sum_{l \in \mathcal{L}} \sum_{n \in N_{\pi,l}} \sum_{m \in \mathcal{M}} \sum_{j=1}^{N_m} \sum_{t'=f(t)}^t T_R r_{\pi,n,t}^{m,l,j,t'} \leq C_\pi^t \quad \forall \pi \in \Pi, t \in \{1, 2, \dots, T\}, \quad (33)$$

$$\sum_{\pi \in \Pi} \sum_{n \in N_{\pi,l}} \sum_{t=t'}^{g(t')} T_R r_{\pi,n,t}^{m,l,j,t'} = s_{m,j} \tau_l^m(\vec{s}_{m,j}, t') \quad \forall m \in \mathcal{M}, l \in \mathcal{L}, t' \in \{1, 2, \dots, T\} \quad (34)$$

5.4. Port to Mine Feedback

Feedback passed between \mathcal{O}_Π , and each \mathcal{O}_m , drives our algorithm toward a solution to the MTP-MMPP that minimises $Z_{MTP-MMPP}$ (Equation 1). In each iteration i , \mathcal{O}_Π provides each \mathcal{O}_m with feedback in the form of a grade and quality target $\vec{\phi}_m^{i+1}$, and a vector of standard deviations $\vec{\sigma}_m^{i+1}$, to be used as its input in iteration $i + 1$. We investigate three methods for the generation of these new targets and standard deviation vectors, one of which (F3) extends the mechanism of Blom et al. (2014) to the multiple period setting. We evaluate each of these methods in Section 6.

5.4.1. Method 1 (F1) Equations (35)–(36) present the first of these methods. Equation (35) states that if \mathcal{O}_Π does not find a solution better than \vec{s}_{best} , in iteration i , the grade and quality target provided to each mine m does not change, $\vec{\phi}_m^{i+1} = \vec{\phi}_m^i$. Each standard deviation in $\vec{\sigma}_m^i$, however, is reduced by a predetermined factor γ , where $0 < \gamma < 1$. The intuition behind F1 is that there may be a target in the vicinity of $\vec{\phi}_m^i$ that, if produced by m , will allow \mathcal{O}_Π to improve upon \vec{s}_{best} . If such

schedules were not proposed in iteration i , it is possible that each \mathcal{O}_m has focused on achieving too large a spread in the composition of produced ore about $\vec{\phi}_m^i$. Reducing $\vec{\sigma}_m^i$ encourages each \mathcal{O}_m to generate schedules, across which, ore compositions are more tightly clustered about $\vec{\phi}_m^{i+1}$.

$$F1: Z_{MTP-MMPP}(\vec{s}_i) \geq Z_{MTP-MMPP}(\vec{s}_{best}) \rightarrow \\ \vec{\phi}_m^{i+1} = \vec{\phi}_m^i \wedge \vec{\sigma}_m^{i+1} = \max(\vec{\sigma}^-, \gamma \vec{\sigma}_m^i) \quad \forall m \in \mathcal{M} \quad (35)$$

Equation (36) states that if \mathcal{O}_Π finds a solution \vec{s}_i that is better than the current best, \vec{s}_{best} , in iteration i , the grade and quality target given to mine m in iteration $i+1$ is equal to the composition of the ore produced by m in \vec{s}_i , where $v_{l,q}^{m,t}(\vec{s}_i)$ denotes the percentage of $q \in \mathcal{Q}$ in granularity $l \in \mathcal{L}$ produced by m in period t of schedule \vec{s}_i .

$$F1: Z_{MTP-MMPP}(\vec{s}_i) < Z_{MTP-MMPP}(\vec{s}_{best}) \rightarrow \\ \sigma_{l,q}^{m,t,i+1} = \sigma_{l,q}^{m,t,i} \wedge \phi_{l,q}^{m,t,i+1} = v_{l,q}^{m,t}(\vec{s}_i) \quad \forall m \in \mathcal{M}, t \in \{1, 2, \dots, T\}, l \in \mathcal{L}, q \in \mathcal{Q} \quad (36)$$

5.4.2. Method 2 (F2) We generalise $F1$ to form a second feedback method, $F2$, in which standard deviations are reduced, by γ , only after N_f successive iterations have been performed in which \mathcal{O}_Π was unable to improve upon \vec{s}_{best} , where $N_f \geq 1$. If $N_f = 1$, $F2$ reduces to $F1$.

5.4.3. Method 3 (F3) Equations (37)–(38), together with Equation (35), form our third feedback method, $F3$. If \mathcal{O}_Π finds a solution, \vec{s}_i , that is better than \vec{s}_{best} , we consider each $\sigma_{l,q}^{m,t,i}$ in $\vec{\sigma}_m^i$ and increase it by a predetermined factor γ , $0 < \gamma < 1$, if $v_{l,q}^{m,t}(\vec{s}_i)$ is sufficiently distant from the given target $\phi_{l,q}^{m,t,i}$ ($|v_{l,q}^{m,t}(\vec{s}_i) - \phi_{l,q}^{m,t,i}| > \sigma_{l,q}^{m,t,i}$). The intuition is that prior reductions in the size of $\sigma_{l,q}^{m,t,i}$ may have been premature. Increasing $\sigma_{l,q}^{m,t,i}$, for iteration $i+1$, will encourage \mathcal{O}_m to form schedules, denoted $\vec{s}_{m,j}^{i+1}$ for $j = 0, 1, \dots, N$, across which there is more diversity in $v_{l,q}^{m,t}(\vec{s}_{m,j}^{i+1})$.

$$F3: Z_{MTP-MMPP}(\vec{s}_i) < Z_{MTP-MMPP}(\vec{s}_{best}) \wedge |v_{l,q}^{m,t}(\vec{s}_i) - \phi_{l,q}^{m,t,i}| \leq \sigma_{l,q}^{m,t,i} \rightarrow \\ \sigma_{l,q}^{m,t,i+1} = \sigma_{l,q}^{m,t,i} \quad \forall m \in \mathcal{M}, t \in \{1, 2, \dots, T\}, l \in \mathcal{L}, q \in \mathcal{Q} \quad (37)$$

$$F3: Z_{MTP-MMPP}(\vec{s}_i) < Z_{MTP-MMPP}(\vec{s}_{best}) \wedge |v_{l,q}^{m,t}(\vec{s}_i) - \phi_{l,q}^{m,t,i}| > \sigma_{l,q}^{m,t,i} \rightarrow \\ \sigma_{l,q}^{m,t,i+1} = \min(\sigma_{l,q}^+, \gamma^{-1} \sigma_{l,q}^{m,t,i}) \quad \forall m \in \mathcal{M}, t \in \{1, 2, \dots, T\}, l \in \mathcal{L}, q \in \mathcal{Q} \quad (38)$$

Table 1 MIP relaxation of the MTP-MMPP MINLP: Time taken (s) to first feasible solution (and it's duality gap), and to solve to optimality, for each test #. The maximum deviation in metal grade present (in a product formed at a port across the time horizon, from desired bounds) in the best found solution is stated. ‘-’ denotes a failure to find an integer feasible, or optimal, solution within a 12 hr time limit.

#	Time to 1 st (s)	Gap (%)	Time to opt (s)	Max. deviation (metal grade, %)	#	Time to 1 st (s)	Gap (%)	Time to opt (s)	Max. deviation (metal grade, %)
1	9539	0.61	23904	2.12	11	–	–	–	–
2	6672	0	6672	1.40	12	8846	0.26	–	1.76
3	5077	0.09	6807	1.27	13	8825	0.10	9929	1.62
4	3087	0	3087	1.29	14	13306	0	13306	1.49
5	–	–	–	–	15	13427	0	13427	1.51
6	2693	0.28	2751	1.67	16	11818	0.38	16364	1.79
7	4520	0.11	4899	1.79	17	29833	0.04	–	1.15
8	14198	0.07	27177	1.89	18	6415	0.35	6569	1.71
9	6261	0.29	9020	1.67	19	24917	0.42	34968	1.69
10	5181	0	5181	2.52	20	39761	0.39	–	1.47

6. Computational Results

To evaluate our decomposition-based algorithm, for the MTP-MMPP, we consider a currently operating system of 8 mines, connected to 2 ports, producing over 200 million tons of ore annually. Our planning horizon spans 13 weeks, divided into weekly time periods. We construct a set of test cases, each test characterising each mine in terms of: a set of grade blocks, sufficient to supply the mine’s processing plants for 2-3 weeks; a set of blast blocks, sufficient to supply these plants for at least 4 months; the composition, and tonnage of each block and stockpile; the mining precedences that exist between blocks; and all relevant capacities. Capacities at each mine and port, and bounds on the composition of port products, are constant across the set of test cases. Varied across test instances is the set, and number, of grade and blast blocks available for extraction at each mine. In each test, each port forms one product of each granularity ($|N_{\pi,l}| = 1$ for all $\pi \in \Pi$ and $l \in \mathcal{L}$).

The number of blocks available for scheduling at each mine ranges from 102 to 437 across the test suite. The total number of blocks in the network ranges from 1967 to 2095. Test cases were formed using data provided by an industry partner, following the approach used by Blom et al. (2014) in the single time period setting. All experiments have been run on a machine with 12 Intel(R) Xeon(R) E5-2440 cpus and 64GB RAM, and afforded 12 hours of (wall) clock time to complete. All MIPs have been solved with CPLEX 12.6.

We first consider whether the MINLP of Section 4 can be solved directly, or if good solutions can be discovered by solving a MIP relaxation. For each test, we find a lower bound on the value of $Z_{MTP-MMPP}$ (Equation 1) by solving a MIP relaxation of the MINLP. This relaxation is formed by replacing each bilinear term in Constraints (16)–(17) and (18)–(19) with its McCormick (1976) envelope. We use this lower bound to evaluate the quality of solutions found by our decomposition-based algorithm. Table 1 states, for each test: the time required (in seconds) to find the first

Table 2 Best solution \vec{s}_{best} found by our algorithm for $N = 3, 5,$ and $7,$ $\gamma = 0.25,$ and $M_A = 4,$ in each test #, recording: elapsed time to solve (s); and the gap (%) between $Z_{MTP-MMPP}(\vec{s}_{best})$ and the best known lower bound. An average (μ) and standard deviation (σ) for all quantities, over 10 seeded runs, is recorded.

#	$N = 3, \gamma = 0.25, M_A = 4$				$N = 5, \gamma = 0.25, M_A = 4$				$N = 7, \gamma = 0.25, M_A = 4$			
	Time (s)		Gap to (%) MINLP _{lb}		Time (s)		Gap to (%) MINLP _{lb}		Time (s)		Gap to (%) MINLP _{lb}	
	μ_T	σ_T	μ_G	σ_G	μ_T	σ_T	μ_G	σ_G	μ_T	σ_T	μ_G	σ_G
1	5317	747	2.36	1.21	6261	1434	1.87	0.48	9273	1179	1.74	1.00
2	4035	962	11.65	29.48	6502	1263	1.12	0.49	8994	1247	0.88	0.21
3	5739	906	1.72	0.91	7089	1450	0.96	0.62	10843	1772	0.89	0.58
4	5572	981	1.54	0.52	6601	1108	1.12	0.6	9792	1380	1.03	0.54
5	4362	934	12.29	29.29	7912	1098	0.88	0.27	10030	1712	0.77	0.43
6	4948	945	1.48	0.74	6068	1320	1.08	0.54	9815	2201	0.4	0.27
7	4062	321	2.37	1.08	6350	1323	0.94	0.75	8736	2469	0.71	0.46
8	5309	1471	2.65	1.98	6858	1245	2.09	1.32	9103	1862	1.5	1.08
9	5029	1270	2.1	1.18	6132	956	0.99	0.61	10145	2067	0.57	0.23
10	4607	1541	3.18	1.43	6770	1285	1.35	0.62	11736	1806	0.77	0.29
11	4849	690	11.4	29.55	6792	1362	0.65	0.27	9715	1203	0.59	0.55
12	5105	994	13.11	29.07	6822	1156	1.89	1.98	11006	2826	1.27	0.76
13	4872	1056	2.77	1.31	7474	1368	1.2	0.81	9108	1208	1.32	0.48
14	4973	1253	22.18	38.94	8078	1588	1.64	0.76	9851	2076	0.96	0.58
15	4642	1584	22.61	38.72	7522	809	1.48	0.53	8952	2155	0.93	0.44
16	4538	653	1.44	0.26	7208	729	1.11	0.53	9222	2094	0.85	0.40
17	5366	1162	12.57	29.18	8637	1475	1.43	0.69	9205	2035	1.44	0.56
18	4165	558	2.36	1.12	7543	1820	1.22	0.83	9335	1054	1.28	0.77
19	5240	919	31.26	45.03	8203	1221	1.34	0.4	10734	1307	0.74	0.34
20	5135	1127	21.84	39.1	6893	574	0.93	0.57	9490	1749	0.77	0.58

feasible solution of the MIP relaxation (alongside the gap between its objective value and the best known lower bound); the time required to solve the MIP to optimality; and the maximum deviation present, in the best solution obtained, between the actual (metal) grade of port products formed across the scheduling horizon, and desired bounds. Constraints (16)–(17) track the composition of stockpiles at each mine, while Constraints (18)–(19) define the grade of ore produced by each mine, and transported to each port, in each time period. Their relaxation leads to discrepancies between the actual grade of products formed at each mine and port, and that inferred by the relaxed MIP. Due to the size of these discrepancies, it is not possible to solve the MIP relaxation with narrowed bounds on the composition of port products, and obtain solutions in which these products are correctly blended. Blom et al. (2014) consider piecewise-linear relaxations (Gounaris et al. 2009) of the bilinear constraints present in the single period MMPP to generate a MIP relaxation of greater fidelity. For each bilinear term, the domain of one variable is partitioned into several intervals, and its value constrained to lie within one of these intervals. Given a 12 hour timeout, a solution to the relaxed single period MMPP could not be found without significant deviations in port product compositions present.

Table 3 Best solution \bar{s}_{best} found by our algorithm for $N = 5$, $M_A = 4$, $\gamma = 0.25, 0.50, 0.75$. Columns are defined as in Table 2. An average (μ) and standard deviation (σ) for all quantities, over 10 seeded runs, is recorded.

#	$N = 5, \gamma = 0.25$				$N = 5, \gamma = 0.50$				$N = 5, \gamma = 0.75$			
	Time (s)		Gap to (%)		Time (s)		Gap to (%)		Time (s)		Gap to (%)	
	μ_T	σ_T	μ_G	σ_G	μ_T	σ_T	μ_G	σ_G	μ_T	σ_T	μ_G	σ_G
1	6261	1434	1.87	0.48	7208	1185	1.72	0.71	10633	1874	1.04	0.42
2	6502	1263	1.12	0.49	6923	1287	0.63	0.2	10877	1763	0.4	0.22
3	7089	1450	0.96	0.62	8355	1580	0.84	0.38	12465	1531	0.54	0.29
4	6601	1108	1.12	0.6	7802	1288	0.7	0.27	11021	1367	0.59	0.35
5	7912	1098	0.88	0.27	7887	1612	0.74	0.25	10929	1628	0.4	0.22
6	6068	1320	1.08	0.54	7194	1322	0.47	0.25	10656	1375	0.32	0.29
7	6350	1323	0.94	0.75	7222	915	0.63	0.52	10471	1095	0.5	0.26
8	6858	1245	2.09	1.32	8162	1933	1.01	0.51	12533	3146	0.8	0.38
9	6132	956	0.99	0.61	7548	1503	0.65	0.5	13753	1194	0.44	0.23
10	6770	1285	1.35	0.62	7697	799	1.02	0.56	13433	1318	0.77	0.31
11	6792	1362	0.65	0.27	7614	935	0.8	0.67	14366	1550	0.28	0.17
12	6822	1156	1.89	1.98	7876	1684	1.17	0.52	13039	1877	0.8	0.57
13	7474	1368	1.2	0.81	7837	1407	1.1	0.42	13097	2188	1.16	0.66
14	8078	1588	1.64	0.76	8309	1174	0.77	0.3	12907	1973	0.84	0.76
15	7522	809	1.48	0.53	7734	1253	1.25	0.6	12977	1302	0.75	0.71
16	7208	729	1.11	0.53	7550	369	0.67	0.35	13609	991	0.44	0.25
17	8637	1475	1.43	0.69	7921	1126	1.11	0.73	14782	2124	0.87	0.47
18	7543	1820	1.22	0.83	7251	983	0.99	0.23	12672	1247	0.84	0.42
19	8203	1221	1.34	0.4	8286	1093	1.36	0.47	13635	1337	0.88	0.47
20	6893	574	0.93	0.57	7075	995	0.62	0.36	11976	1485	0.56	0.29

Blom et al. (2014) demonstrate that their decomposition-based algorithm was able to find solutions to the single period MMPP that were as good, or better, than a range of alternative approaches, commonly applied to problems with a pooling component, in orders of magnitude less time. The best performing of these approaches was the ALT heuristic of Audet et al. (2004). ALT fixes the value of one variable in each bilinear term, producing a MIP that is solved to find values for the remaining bilinear variables. Fixing these alternate variables forms a second MIP that is solved to find new values for the first variable set. This process is repeated until a fixed point is reached. We have applied ALT to the MINLP of Section 4, partitioning and alternately fixing the its bilinear variables. Given a time limit of 12 hrs and 2000s afforded to each MIP solve, ALT could not find a solution, in any of our tests, in which port products were correctly blended.

We now consider the performance of our decomposition-based algorithm in varying settings. We first examine the impact of varying the N and γ parameters on the quality of solutions found by our algorithm (Tables 2–3). Recall that N denotes the number of schedules formed during the solving of each \mathcal{O}_m , γ the degree to which the standard deviations given to each \mathcal{O}_m as input are increased or decreased (a larger γ results in smaller changes) during feedback generation, and M_A the maximum size of block aggregates formed during the generation of schedules by each \mathcal{O}_m . We next demonstrate the need to monitor evolving stockpile states, by evaluating the difference between expected and

Table 4 Best solution \vec{s}_{best} found by our algorithm for $N = 5$, $\gamma = 0.25$, $M_A = 4$, and a single application of the rolling horizon heuristic of Section 5.2 (1-Stage) used by each \mathcal{O}_m to generate schedules. Results obtained using two applications of this heuristic (2-Stage, from Table 2) are listed for comparison. Columns are defined as in Table 2. An average (μ) and standard deviation (σ) for all quantities, over 10 seeded runs, is recorded.

#	1-Stage				2-Stage				#	1-Stage				2-Stage			
	Time (s)		Gap to (%)		Time (s)		Gap to (%)			Time (s)		Gap to (%)		Time (s)		Gap to (%)	
	μ_T	σ_T	μ_G	σ_G	μ_T	σ_T	μ_G	σ_G		μ_T	σ_T	μ_G	σ_G	μ_T	σ_T	μ_G	σ_G
1	9629	1991	1.93	1.23	6261	1434	1.87	0.48	11	7430	1252	1.08	0.4	6792	1362	0.65	0.27
2	10147	3300	1.52	0.97	6502	1263	1.12	0.49	12	8363	1647	2.23	1.3	6822	1156	1.89	1.98
3	8859	1142	1.43	0.78	7089	1450	0.96	0.62	13	9110	2084	1.91	1.33	7474	1368	1.2	0.81
4	9201	2157	1.07	0.46	6601	1108	1.12	0.6	14	7480	2033	0.92	0.47	8078	1588	1.64	0.76
5	8743	1975	1.15	0.93	7912	1098	0.88	0.27	15	8908	3599	2.28	1.09	7522	809	1.48	0.53
6	8480	1876	0.97	0.49	6068	1320	1.08	0.54	16	8034	1548	1.18	1.01	7208	729	1.11	0.53
7	7882	1776	1.54	1.04	6350	1323	0.94	0.75	17	7426	2196	2.25	1.49	8637	1475	1.43	0.69
8	9509	3231	1.95	1.01	6858	1245	2.09	1.32	18	8660	1470	1.52	0.62	7543	1820	1.22	0.83
9	9520	1892	1.21	0.55	6132	956	0.99	0.61	19	8293	1567	1.46	0.99	8203	1221	1.34	0.4
10	7507	935	0.9	0.62	6770	1285	1.35	0.62	20	8647	1969	0.8	0.37	6893	574	0.93	0.57

achieved port product composition under the assumption of constant stockpile grades. The merits of our two stage approach for the generation of single mine schedules (see Section 5.1), relative to a single application of the rolling horizon heuristic of Section 5.2, are highlighted in Table 4. We then examine the relative performance of the algorithm when instantiated with each of the feedback methods of Section 5.4. Feedback method $F3$ is used in all other experiments. We conclude by examining the impact of splitting the scheduling horizon into more than two time periods in the application of our rolling horizon heuristic. In the remainder of this section, each average value is reported with an associated standard deviation following it in brackets.

Table 2 records the results of our decomposition-based algorithm, averaged over 10 seeded runs, on each of our benchmark tests, with: $N = 3, 5$, and 7 ; $\gamma = 0.25$; $M_A = 4$; and feedback method $F3$ implemented. We record, for the best solution found by the algorithm, \vec{s}_{best} : the elapsed time to termination (s); and the gap (%) between $Z_{MTP-MMPP}(\vec{s}_{best})$ and its best known lower bound. Quantities have been averaged over 10 seeded runs, with the average (μ) and standard deviation (σ) recorded. Across all tests, our decomposition-based algorithm was able to find solutions in which plant capacities, at each mine, are fully utilised. For $N = 5$ and 7 , port products are formed to specification, in all solutions, in each period of the scheduling horizon. The bold entries in Table 2, for $N = 3$, indicate that one or more of the 10 runs of our algorithm on the associated test case did not result in correctly blended port products, in one or more periods. Across these instances, small deviations in one contaminant were present, and the duality gaps of solutions found, across all tests, ranged between 0.6% and 100%. Increasing N from 3 to 5 increases solve times by 2192s (847s) on average and solution quality by 7.88% (9.11%). For $N = 5$, duality gaps range between 0.2% and 5.6%. Increasing N from 5 to 7 increases solve times by 2668s (1071s) on average, with a 0.29% (0.27%) average improvement in quality, and duality gaps ranging between 0.1% and 3.5%. These results demonstrate that a value of N that is too small prevents the algorithm from sufficiently exploring the space of producible grades at each mine site before it terminates.

Reducing the gap between $Z_{MTP-MMPP}(\vec{s}_{best})$ and its lower bound by 1% can be achieved by mining on the order of 10kt more waste, at each mine, across the horizon, or hauling 10kt less material, at each mine, to stockpiles. The large optimality gaps present in some entries of Table 2, for $N = 3$, are the result of port product deviations, which are heavily penalised.

Table 3 records the results of our decomposition-based algorithm, averaged over 10 seeded runs, in each of our benchmark tests, with: $N = 5$; $M_A = 4$; $\gamma = 0.25, 0.50$, and 0.75 ; and feedback method *F3* implemented. Increasing γ from 0.25 to 0.50 increases solve times by 587s (589s) on average while increasing solution quality by only 0.35% (0.30%). For $\gamma = 0.50$, duality gaps range between 0% and 3%. Increasing γ from 0.50 to 0.75 increases solve times and solution quality by an average of 4819s (1158), and 0.25% (0.19%), respectively, with a range in duality gaps of 0.1% to 3%. A larger γ results in smaller changes to standard deviation parameters at each iteration, and as a result, more iterations are performed prior to termination of the algorithm. Smaller values of γ ($\gamma = 0.25$) are sufficient, however, for consistently generating solutions of reasonable quality.

The impact of varying N and γ on the quality of solutions found by our algorithm in the multiple time period setting is consistent with the results of the same experiment performed in the single period case (Blom et al. 2014). In both settings, increasing both N and γ improves, in general, the quality of solutions found by the algorithm, but at a cost of longer solve times.

In our modelling of the MTP-MMPP, we keep track of the composition of stockpiles at each mine, across the scheduling horizon. We have used a modified version of our algorithm (with $N = 5$, $\gamma = 0.50$, and $M_A = 4$), in which Step 14 of our rolling horizon heuristic (Algorithm 3) does not update the composition of stockpiles, to solve each of our tests. For each test, and each solution obtained from 10 differently seeded runs of our algorithm, we compute the actual state of each stockpile, at each mine, in each period, and the actual deviation that exists (if any) between the composition of port products formed in each time period (based on actual stockpile states) and desired bounds. We find that while the average deviations reported are small, across the set of relevant quality attributes, the maximum deviations experienced, over all obtained solutions, are significant. The average deviation in the metal grade of port products (from desired bounds), across all time periods and tests, is 0.01% (0.03%), with a maximum experienced deviation of 0.67%. The distance between the upper and lower bound on metal grade, in each port product, is 1%. These results demonstrate the importance of monitoring the state of stockpiles over time.

We have calculated, via experiment, the variance in the grade of ore produced by each mine and port, providing an indication of how grade varies between reclaimed slices of stockpiles. We consider the schedules produced for each mine, across our test suite, in the first iteration of our decomposition-based algorithm. We calculate the average variance in each quality attribute, across each schedule formed, in 10 seeded runs of the algorithm with $N = 5$, $M_A = 4$, and $\gamma = 0.25$.

Table 5 Best solution \vec{s}_{best} found by our algorithm for $N = 5$, $\gamma = 0.25$, $M_A = 2, 4$, and 8. Columns are defined as in Table 2. An average (μ) and standard deviation (σ) for all quantities, over 10 seeded runs, is recorded.

#	$M_A = 2$				$M_A = 4$				$M_A = 8$			
	Time (s)		Gap to (%)		Time (s)		Gap to (%)		Time (s)		Gap to (%)	
	μ_T	σ_T	μ_G	σ_G	μ_T	σ_T	μ_G	σ_G	μ_T	σ_T	μ_G	σ_G
1	10146	1856	1.88	0.87	6261	1434	1.87	0.48	6627	1210	1.81	1.38
2	9410	1538	0.68	0.29	6502	1263	1.12	0.49	5688	860	1.57	0.77
3	10471	1535	1.14	0.82	7089	1450	0.96	0.62	7370	984	0.96	0.3
4	10617	1629	0.79	0.48	6601	1108	1.12	0.6	6308	1626	1.8	1.31
5	9331	1190	1.19	0.65	7912	1098	0.88	0.27	6856	843	0.99	0.62
6	8989	1550	0.92	0.92	6068	1320	1.08	0.54	7005	863	1.29	0.59
7	10044	1241	0.81	0.3	6350	1323	0.94	0.75	5856	1434	3.23	2.4
8	10689	1718	1.4	0.76	6858	1245	2.09	1.32	7528	1736	1.41	0.32
9	9265	1093	0.88	0.19	6132	956	0.99	0.61	7426	1508	1.19	0.88
10	7765	2514	0.94	0.41	6770	1285	1.35	0.62	7436	1246	2.1	1.36
11	9646	1485	1.07	0.65	6792	1362	0.65	0.27	6429	665	1.41	0.53
12	11062	2295	2.26	2.55	6822	1156	1.89	1.98	6873	1159	1.4	0.72
13	10482	2025	1.02	0.38	7474	1368	1.2	0.81	6414	1123	2.58	1.07
14	9992	1238	1.33	0.56	8078	1588	1.64	0.76	7134	1692	2.62	1.45
15	9632	2011	1.67	0.76	7522	809	1.48	0.53	7188	1743	2.6	2.65
16	9283	1707	1.23	0.62	7208	729	1.11	0.53	7072	901	1.28	0.73
17	8897	2105	1.4	0.88	8637	1475	1.43	0.69	7031	1398	1.95	1.04
18	8620	1547	1.96	1.24	7543	1820	1.22	0.83	7023	915	1.11	0.32
19	7783	2317	0.98	0.69	8203	1221	1.34	0.4	7507	1417	1.33	0.9
20	9411	2153	1.05	0.77	6893	574	0.93	0.57	5975	715	1.12	0.55

The average variance in metal content, over the scheduling horizon, was 0.23 (0.12), indicating that samples of ore produced at each mine will be distributed about the average with a standard deviation of 0.48%. As ore arrives at each port, it is blended onto stockpiles, and reclaimed onto ships. Given a 1 : 5 blending effect arising from stacking and reclaiming at the ports, the variance in metal content of shipped ore is, on average, 0.25 (0.03). Samples of the shipped ore will be distributed about the average grade with a standard deviation of 0.50%. Such variability in blended products of this type lies within expectations (Everett et al. 2002, Minnitt and Pitard 2008).

To generate a single schedule, \vec{s}_m for a mine m , \mathcal{O}_m applies the two stage process described in Section 5.1. First, the set of available grade and blast blocks are aggregated to form a smaller number of larger units. The rolling horizon heuristic of Section 5.2 determines how these aggregates are to be mined over the course of the scheduling horizon. The original set of blocks is culled by removing all blocks that do not appear in an aggregate mined in the resulting schedule. A second application of this heuristic, in which only blocks in this restricted set, and the set of grade blocks, can be mined, generates our schedule, \vec{s}_m . The intuition is that this two-stage process is likely to be less time consuming than a single application of the rolling horizon heuristic to the unrestricted, original block set. Table 4 records the results of our algorithm with $N = 5$, $\gamma = 0.25$, $M_A = 4$, and each \mathcal{O}_m instructed to generate schedules with only one application of our rolling horizon heuristic

Table 6 Best solution \bar{s}_{best} found by our algorithm for $N = 5$, $\gamma = 0.25$, $M_A = 4$, and feedback method $F1$, $F2$ with $N_f = 2$, and $F3$ implemented. Columns are defined as in Table 2. An average (μ) and standard deviation (σ) for all quantities, over 10 seeded runs, is recorded.

#	F1				F2 with $N_f = 2$				F3			
	Time (s)		Gap to (%)		Time (s)		Gap to (%)		Time (s)		Gap to (%)	
	μ_T	σ_T	μ_G	σ_G	μ_T	σ_T	μ_G	σ_G	μ_T	σ_T	μ_G	σ_G
1	3438	592	21.82	39.13	11721	2362	1.37	0.72	6261	1434	1.87	0.48
2	3786	522	1.14	0.54	11354	1443	0.65	0.4	6502	1263	1.12	0.49
3	4702	709	1.04	0.33	12442	2905	1.01	0.72	7089	1450	0.96	0.62
4	4409	853	1.54	0.89	11013	1255	0.89	0.51	6601	1108	1.12	0.6
5	3370	501	21.13	39.48	12129	1996	1.03	0.91	7912	1098	0.88	0.27
6	3449	457	1.71	0.6	13038	2454	0.59	0.26	6068	1320	1.08	0.54
7	3652	606	1.64	0.96	12244	2299	0.72	0.51	6350	1323	0.94	0.75
8	4219	874	2.25	1.31	14611	2647	0.85	0.45	6858	1245	2.09	1.32
9	4195	665	1.08	0.32	12918	2371	0.6	0.27	6132	956	0.99	0.61
10	4150	720	1.48	0.77	13332	2464	0.85	0.42	6770	1285	1.35	0.62
11	3987	500	1.22	0.79	12862	2454	0.6	0.28	6792	1362	0.65	0.27
12	4288	570	11.76	29.44	12767	2250	1.58	0.98	6822	1156	1.89	1.98
13	3711	397	11.92	29.42	11773	1869	1.23	0.6	7474	1368	1.2	0.81
14	3955	580	2.15	1.08	11643	2216	0.73	0.38	8078	1588	1.64	0.76
15	3779	796	2.33	1.77	9018	1854	0.98	0.43	7522	809	1.48	0.53
16	4114	500	1.15	0.29	10518	1700	0.56	0.14	7208	729	1.11	0.53
17	4015	770	21.71	39.2	10860	1954	0.83	0.51	8637	1475	1.43	0.69
18	4158	879	1.6	1.17	12146	1178	0.63	0.25	7543	1820	1.22	0.83
19	4282	688	2.53	1.24	10686	1909	1.02	0.65	8203	1221	1.34	0.4
20	3673	685	1.89	1.15	9756	1875	0.64	0.33	6893	574	0.93	0.57

(1-Stage). In this setting, the duality gaps of solutions found, across all tests, range between 0.1% to 5.6%. For comparison, Table 4 replicates the results of Table 2 for $N = 5$, $\gamma = 0.25$, and $M_A = 4$, where two applications of the rolling horizon heuristic were completed (2-Stage), and duality gaps ranged between 0.2% and 5.6%. In the 1-Stage setting, our algorithm solves, on average, 1505s (1284s) slower, while producing solutions that are 0.2% (0.40%) further from lower bounds.

We examine the performance of our decomposition-based algorithm for different values of maximum aggregate size M_A . Table 5 reports the results of our algorithm given $N = 5$, $\gamma = 0.25$, and $M_A = 2, 4$, and 8. For $M_A = 2$, solution duality gaps range between 0.2% and 9.3%. Increasing M_A from 2 to 4 reduces solve times by 2491s (1298s), on average, with no significant change in average solution quality. The maximum duality gap observed, however, reduces to 5.6%. Increasing M_A from 4 to 8 reduces solve times by 248s (760s), with an average reduction in solution quality of 0.42% (0.68%). Duality gaps range between 0.2% and 9.9%. Reducing M_A allows our algorithm to be more selective in the blocks it discards from consideration in the first scheduling stage, but results in a greater number of blocks available for scheduling in the second, increasing solve times.

Table 6 reports the results of our algorithm when instantiated with feedback methods $F1$, $F2$ with $N_f = 2$, and $F3$, for $N = 5$, $\gamma = 0.25$, and $M_A = 4$. Each of these methods is described in

Table 7 Best solution \vec{s}_{best} found by our algorithm for $N = 5$, $\gamma = 0.25$, $M_A = 4$, $F3$ implemented, and the horizon split into 3 periods when the rolling horizon heuristic of Section 5.2 is applied. Columns are defined as in Table 2. An average (μ) and standard deviation (σ) for all quantities, over 10 seeded runs, is recorded.

#	2-period split				3-period split				2-period split				3-period split				
	Time (s)		Gap to (%)		Time (s)		Gap to (%)		Time (s)		Gap to (%)		Time (s)		Gap to (%)		
	μ_T	σ_T	μ_G	σ_G													
1	6261	1434	1.87	0.48	18539	2945	2.24	0.77	11	6792	1362	0.65	0.27	17572	2273	1.43	0.8
2	6502	1263	1.12	0.49	18592	3386	2.35	1.26	12	6822	1156	1.89	1.98	18411	2813	2.68	0.97
3	7089	1450	0.96	0.62	18914	3234	2.91	1.3	13	7474	1368	1.2	0.81	20308	2065	2.56	0.99
4	6601	1108	1.12	0.6	17921	3507	2.88	0.97	14	8078	1588	1.64	0.76	18966	4134	1.65	0.61
5	7912	1098	0.88	0.27	19880	3608	1.46	0.71	15	7522	809	1.48	0.53	20937	2317	2.56	1.49
6	6068	1320	1.08	0.54	20021	2720	1.05	0.67	16	7208	729	1.11	0.53	19270	2164	1.51	0.81
7	6350	1323	0.94	0.75	15923	2698	1.66	0.77	17	8637	1475	1.43	0.69	22140	4634	2.83	1.4
8	6858	1245	2.09	1.32	14156	2926	3.75	1.02	18	7543	1820	1.22	0.83	19740	4821	1.75	0.52
9	6132	956	0.99	0.61	17786	2626	1.31	0.84	19	8203	1221	1.34	0.4	18328	3082	2.74	0.96
10	6770	1285	1.35	0.62	17732	1984	2.07	1.44	20	6893	574	0.93	0.57	18618	3690	1.62	0.65

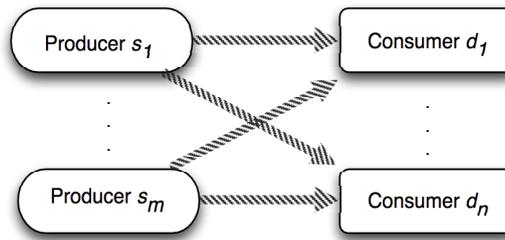
Section 5.4. Using $F1$, solve times are 7875s (1295s) and 3120s (768s) faster, on average, than when using $F2$ ($N_f = 2$) and $F3$, respectively. In five test cases, however, our algorithm (with $F1$ implemented) did not find solutions in which port products were correctly blended (these instances are highlighted in bold in Table 6). The duality gaps of solutions, across all tests, range from 0.1% to 100%. Using $F3$, in place of $F2$ ($N_f = 2$) results in solve times that are 4756s (1725s) faster, and solutions with duality gaps that are 0.78% (0.37%) higher, on average. Duality gaps range from 0% to 3.4% when $F2$ is implemented, and 0.2% to 5.6% for $F3$. While solution quality improves, in general, with method $F2$, it does so at the expense of longer solve times.

The rolling horizon heuristic that we use to generate mine-side schedules (Section 5.2) splits the horizon into two periods, solves a two-period problem to schedule the first period, and then repeats this process on the remaining periods in the horizon. Table 7 records the results of our algorithm when this heuristic splits the horizon into three periods. The results obtained using the original two-period split are also shown. In the 3-period setting, our algorithm is 11602s (1485s) slower, on average, finding solutions with duality gaps that are, on average, 0.89% (0.57%) higher – ranging from 0.3% to 5.9%. In the 2-period setting, duality gaps range between 0.2% and 5.6%. Recall that time limits are placed on the solving of all MIPs by the rolling horizon heuristic. While dividing the horizon into 3 periods, in place of 2, is a better approximation of the horizon, each 3-period MIP is more time-consuming to solve. The resulting solutions have higher duality gaps, with respect to the relevant objectives, than those found by solving 2-period MIPs.

7. Multi-Period, Multi-Producer/Consumer Production Planning

The MTP-MMPP is an instance of a general class of multiple period, multiple producer/consumer production planning problems. Consider a network of independently operating production sites, \mathcal{PS} , and multiple distribution centres, \mathcal{DC} , as shown in Figure 3. Each production site $s \in \mathcal{PS}$, in each time period $t \in \mathcal{T}$, is able to produce a set of products, \mathcal{P}_s , each characterised by a vector of numerical attributes, $\vec{a}_{p,s}^t$, $p \in \mathcal{P}_s$. These attributes may include the quantity of p produced in

Figure 3 A network of production sites, $s_i \in \mathcal{PS}$, and distribution centres, $d_j \in \mathcal{DC}$.



period t and/or its quality. Each distribution centre $d \in \mathcal{DC}$, in each period $t \in \mathcal{T}$, combines products formed across the set of sites to form its own products, \mathcal{P}_d , similarly characterised by a vector of attributes, $\vec{a}_{p,d}^t$, $p \in \mathcal{P}_d$, to be shipped to external markets. Each of these products, in each period, is associated with deterministic demands on the values of its associated attributes (for example, on the quantity and quality of production). In the open-pit supply network, each production site is a mine, each distribution centre is a port, lump and fines products are formed at each mine and port, and the attributes of mine and port products are metal percentage and impurity levels.

Such problems can be decomposed into $|\mathcal{PS} + 1|$ subproblems, one subproblem for each production site $s \in \mathcal{PS}$, denoted \mathcal{O}_s , and one distribution subproblem involving the set of production sites and distribution centres, denoted \mathcal{O}_D . Our decomposition-based algorithm, as defined in Section 5, can be readily applied to problems of this form. In the application of this algorithm, each \mathcal{O}_s is designed to accept, as input, a vector of target attributes, and an associated standard deviation for each attribute, for each of the products it is capable of producing. A solution to subproblem \mathcal{O}_s is a set of schedules for site s , each schedule producing a set of products, in each period, with varying attributes. The \mathcal{O}_D subproblem accepts these schedule sets, selects one schedule to be implemented at each site, and solves a distribution problem in which products are transported between sites and centres. The attributes of the products formed at each centre are dependent on the attributes of those produced at each site. A feedback mechanism, such as those described in Section 5.4, must be defined to give each \mathcal{O}_s a new vector of target attributes, and standard deviations, for each product $p \in \mathcal{P}_s$. The successive solving of each \mathcal{O}_s and \mathcal{O}_D , in conjunction with the standard deviations supplied to each \mathcal{O}_s reducing in size over the course of the algorithm, yields a sequence of monotonically improving solutions to the original multi-period, multi-site planning problem.

General production planning problems of this form can be found in a range of domains, including the natural resources sector, food production, and the chemical process industry. While our model of the MTP-MMPP exhibits the structure of a pooling problem, our approach is not restricted to solving problems that involve pooling. We require only that the problem be decomposable into several optimisation problems for the upstream components of the supply chain, and a single

optimisation problem representing the downstream component. The application of our approach to problems in this general class is planned as future work.

8. Concluding Remarks

In this paper we have presented a decomposition-based algorithm for a challenging multiple mine, multiple time period, open-pit production scheduling problem. Non-linear in nature, this problem can be formulated as a MINLP with on the order of 100,000 variables in realistic instances. In a simpler instantiation of this problem – the single time period MMPP – this decomposition-based algorithm was able to find solutions that were, in a majority of cases, higher in quality than those discovered by a range of alternative techniques (Blom et al. 2014). Where this was not the case, the algorithm was able to find good quality solutions for which the alternative methods required orders of magnitude more time to match. We have shown, in this paper, that this algorithm is able to scale to the significantly more complex multiple time period setting, generating solutions with optimality gaps within 6% of known lower bounds on the objective of the MINLP problem representation, in less than 2 hours, on average. The best performing alternative approach considered by Blom et al. (2014) in the solving of the single period MMPP could not scale to the multiple period setting.

The extension of the algorithm to the multiple time period setting required the development of a new heuristic for the generation of multiple time period extraction schedules for each mine and a new MIP model to represent the port-side optimisation problem. The modelling of time significantly increases the complexity of the MMPP. The composition of stockpiles at each mine must be tracked, over time, to avoid errors in the evaluation of port product grades. Each decision made at each mine has an influence on the material that will be available for mining in the future, and in consequence the ability to satisfy constraints on plant use. These decisions determine the composition of ore produced by each site, which in turn influences the decisions of other sites to ensure the correct blending of products at each port – resulting in a problem with high combinatorial complexity.

The decomposition-based algorithm presented in this paper is, to the best of our knowledge, the first approach to solve an integrated multiple-period open-pit scheduling problem, across multiple mines and ports, where the grade and quality of the ore to be produced by each mine is not known *a priori*, but determined as part of the optimisation.

References

- Audet, C., J. Brimberg, P. Hansen, S. L. Digabel, N. Mladenovic. 2004. Pooling Problem: Alternate Formulations and Solution Methods. *Management Science* **50** 761–776.
- Bassett, M. H., J. F. Pekny, G. V. Reklaitis. 1996. Decomposition Techniques for the Solution of Large-Scale Scheduling Problems. *AIChE Journal* **42** 3373–3387.

- Bengtsson, J., D. Bredström, P. Flisberg, M. Rönnqvist. 2013. Robust planning of blending activities at refineries. *Journal of the Operational Research Society* **64** 848–863.
- Bitran, G. R., A. C. Hax. 1977. On The Design of Hierarchical Production Planning Systems. *Decision Sciences* **8** 28–55.
- Bley, A., N. Boland, G. Froyland, M. Zuckerberg. 2012. Solving mixed integer nonlinear programming problems for mine production planning with stockpiling. web.maths.unsw.edu.au/~froyland/bbfz.pdf .
- Blom, M. L., C. N. Burt, A. R. Pearce, P. J. Stuckey. 2014. A Decomposition-Based Heuristic for Collaborative Scheduling in a Network of Open-Pit Mines. *INFORMS Journal on Computing* **26** 658–676.
- Boland, N., I. Dumitrescu, G. Froyland, A. M. Gleixner. 2009. LP-based disaggregation approaches to solving the open pit mining production scheduling problem with block processing selectivity. *Computers & Operations Research* **36** 1064–1089.
- Carvalho, M., J. Pinto. 2006. A bilevel decomposition technique for the optimal planning of offshore platforms. *Brazilian Journal of Chemical Engineering* **23** 67 – 82.
- Cullenbine, C., R. K. Wood, A. Newman. 2011. A sliding time window heuristic for open pit mine block sequencing. *Optimization Letters* **5** 365–377.
- Dimitriadis, A. D., N. Shah, C. C. Pantelides. 1997. RTN-based Rolling Horizon Algorithms for Medium Term Scheduling of Multipurpose Plants. *Computers and Chemical Engineering* **21** 1061–1066.
- Elkamel, A., M. Zentner, F. Pekny, G. V. Reklaitis. 1997. A Decomposition Heuristic For Scheduling The General Batch Chemical Plant. *Engineering Optimization* **28** 299–330.
- Epstein, R., M. Goic, A. Weintraub, J. Catalán, P. S. nez, R. Urrutia, R. Cancino, S. Gaete, A. Aguayo, F. Caro. 2012. Optimizing Long-Term Production Plans in Underground and Open-Pit Copper Mines. *Operations Research* **60** 4–17.
- Espinoza, D., M. Goycoolea, E. Moreno, A. Newman. 2012. MineLib: a library of open pit mining problems. *Annals of Operations Research* **206** 93–114.
- Everett, J. E. 1996. Iron Ore Handling Procedures Enhance Export Quality. *Interfaces* **26** 82–94.
- Everett, J. E., M. Kamperman, T. J. Howard. 2002. Information for Decision Support, Information for Performance Evaluation – But Don't Mix Your Drinks. *Informing Science and IT Education Conference*. 427–439.
- Goodwin, G. C., M. M. Seron, R. H. Middleton, M. Zhang, B. F. Hennessy, P. M. Stone, M. Menabde. 2006. Receding horizon control applied to optimal mine planning. *Optimal Control Applications to Management Sciences* **8** 1337–1342.
- Gounaris, C. E., R. Misener, C. A. Floudas. 2009. Computational Comparison of Piecewise-Linear Relaxations for Pooling Problems. *Industrial and Engineering Chemistry Research* **48** 5742–5766.
- Haverly, C. A. 1978. Studies of the behaviour of recursion for the pooling problem. *ACM SIGMAP Bulletin* **25** 19–28.

- Hax, A. C., H. C. Meal. 1973. Hierarchical integration of production planning and scheduling. Tech. rep., Sloan School of Management.
- Iyer, R. R., I. E. Grossmann. 1998. Optimal Planning and Scheduling of Offshore Oil Field Infrastructure Investment and Operations. *Industrial and Engineering Chemistry Research* **37** 1380–1397.
- Janak, S. L., C. A. Floudas, J. Kallrath, N. Vormbrock. 2006. Production Scheduling of a Large-Scale Industrial Batch Plant. I. Short-Term and Medium-Term Scheduling. *Industrial and Engineering Chemical Research* **45** 8234–8252.
- Kumral, M. 2006. Bed blending design incorporating multiple regression modelling and genetic algorithms. *Journal of the South African Institute of Mining and Metallurgy* **106** 229–237.
- Lambert, W. B., A. Brickey, A. M. Newman, K. Eurek. 2014. Open-Pit Block-Sequencing Formulations: A Tutorial. *Interfaces* **44** 127–142.
- Lambert, W. B., A. M. Newman. 2013. Tailored Lagrangian Relaxation for the open pit block sequencing problem. *Annals of Operations Research* 1–20.
- Maravelias, C. T., C. Sung. 2009. Integration of production planning and scheduling: Overview, challenges, and opportunities. *Computers and Chemical Engineering* **33** 1919–1930.
- Màs, R., J. M. Pinto. 2003. A Mixed-Integer Optimization Strategy for Oil Supply in Distribution Complexes. *Optimization and Engineering* **4** 23–64.
- McCormick, G. P. 1976. Computability of global solutions to factorable nonconvex programs: Part i—convex underestimating problems. *Mathematical Programming* **10** 147–175.
- Méndez, C. A., J. Cerdá, I. E. Grossmann, I. Harjunkoski, M. Fahl. 2006. State-of-the-art review of optimization methods for short-term scheduling of batch processes. *Computers and Chemical Engineering* **30** 931–946.
- Minnitt, R. C. A., F. F. Pitard. 2008. Application of variography to the control of species in material process streams: *The Journal of The Southern African Institute of Mining and Metallurgy* **108** 109–122.
- Misener, R., C. A. Floudas. 2009. Advances for the pooling problem: modelling, global optimisation, and computational studies. *Applied Computational Mathematics* **1** 3–22.
- Müller, K. P. 2010. Stacking, reclaiming and blending effects. *Mechanical Technology* 14–17.
- Newman, A. M., M. Kuchta. 2007. Using aggregation to optimize long-term production planning at an underground mine. *European Journal of Operational Research* **176** 1205–1218.
- Newman, A. M., E. Rubio, R. Caro, A. Weintraub, K. Eurek. 2010. A Review of Operations Research in Mine Planning. *Interfaces* **40** 222–245.
- Osanloo, M., J. Gholamnejad, B. Karimi. 2008. Long-term open pit mine production planning: a review of models and algorithms. *International Journal of Mining, Reclamation, and Environment* **22** 3–35.

- Ramazan, S. 2007. The new fundamental tree algorithm for production scheduling of open pit mines. *European Journal of Operational Research* **177** 1153–1166.
- Reddy, P. P., I. Karimi, R. Srinivasan. 2004. A new continuous-time formulation for scheduling crude oil operations. *Chemical Engineering Science* **59** 1325 – 1341.
- Rodrigues, M. T. M., L. Gimeno, C. A. S. Passos, M. D. Campos. 1996. Reactive scheduling approach for multipurpose chemical batch plants. *Computers and Chemical Engineering* **20** 1215–1220.
- Rogers, D., R. D. Plante, R. T. Wong, J. R. Evans. 1991. Aggregation and Disaggregation Techniques and Methodology in Optimization. *Operations Research* **39** 553–582.
- Shah, N. 1996. Mathematical programming techniques for crude oil scheduling. *Computers and Chemical Engineering* **20, Supplement 2** S1227 – S1232.
- Singh, G., R. García-Flores, A. Ernst, P. Welgama, M. Zhang, K. Munday. 2013. Medium-Term Rail Scheduling for an Iron Ore Mining Company. *Interfaces, Articles in Advance* 1–19.
- Sundar, D. K., D. Acharya. 1995. Blast schedule planning and shiftwise production scheduling of an opencast iron ore mine. *Computers and Industrial Engineering* **28** 927–935.
- Tabesh, M., H. Askari-Nasab. 2011. Two-stage clustering algorithm for block aggregation in open pit mines. *Mining Technology* **120** 158–169.
- Thomas, A., G. Singh, M. Krishnamoorthy, J. Venkateswaran. 2012. Distributed optimisation method for multi-resource constrained scheduling in coal supply chains. *International Journal of Production Research* **51** 2740–2759.
- Thomas, A., J. Venkateswaran, G. Singh, M. Krishnamoorthy. 2013. A resource constrained scheduling problem with multiple independent producers and a single linking constraint: A coal supply chain example. *European Journal of Operational Research* **In Press**.
- van den Heever, S. A., I. E. Grossmann. 2000. An Iterative Aggregation/Disaggregation Approach for the Solution of a Mixed-Integer Nonlinear Oilfield Infrastructure Planning Model. *Industrial and Engineering Chemistry Research* **39** 1955–1971.
- Weintraub, A., M. Pereira, X. Schultz. 2008. A Priori and A Posteriori Aggregation Procedures to Reduce Model Size in MIP Mine Planning Models. *Electronic Notes in Discrete Mathematics* **30** 297–302.
- Wenkai, L., C.-W. Hui, B. Hua, Z. Tong. 2002. Scheduling crude oil unloading, storage, and processing. *Industrial and Engineering Chemistry Research* **41** 6723–6734.

Appendix A: Modelling Notation

$\mathcal{O}_m, \mathcal{O}_\Pi$ mine-side and port-side subproblems

Sets and Indices

m, \mathcal{M}	mines	π, Π	ports
b, \mathcal{B}_m	blocks at mine m	\mathcal{B}_m^g	grade blocks at mine m

\mathcal{B}_m^b	blast blocks at mine m	l, \mathcal{L}	granularities (lump/fines)
q, \mathcal{Q}	quality attributes	t	time period
δ, Δ_m	waste dumps at mine m	k, hi, lo, w	index k refers to high grade (<i>hi</i>), low grade (<i>lo</i>), or waste (<i>w</i>)
θ, Θ_m	high grade stockpiles at mine m	λ, Λ_m	low grade stockpiles at mine m
s, \mathcal{S}_m	set of material sources at mine m (blocks and stockpiles)	d, \mathcal{D}_m	set of material destinations at mine m (stockpiles and plants)
κ, ω	dry and wet processing plant	$n, N_{\pi, l}$	set of products, of granularity $l \in \mathcal{L}$, formed by port π
$\mathcal{A}_{m, b}^\wedge$	set of blocks that must be removed before block $b \in \mathcal{B}_m$ can be accessed	$\mathcal{A}_{m, b}^\vee$	set of blocks, one of which must be removed, before $b \in \mathcal{B}_m$ can be accessed
i	iteration of the decomposition-based algorithm	\vec{s}_{best}	best solution to the MTP-MMPP found by our algorithm, defines values of variables in the MINLP of Section 4
\vec{s}_i	solution to the MTP-MMPP found by our algorithm in iteration i	Ω_m^i	set of extraction schedules generated by \mathcal{O}_m in iteration i of the algorithm
$\vec{\phi}_m, \vec{\phi}_m^i$	grade and quality target used by \mathcal{O}_m , in iteration i , to generate Ω_m^i	$\vec{\sigma}_m, \vec{\sigma}_m^i$	set of standard deviations used by \mathcal{O}_m , in iteration i , to generate Ω_m^i
$\phi_{l, q}^{m, t, i}$	grade/quality target in $\vec{\phi}_m^i$ corresponding to granularity $l \in \mathcal{L}$, attribute $q \in \mathcal{Q}$, and time period t	$\sigma_{l, q}^{m, t, i}$	standard deviation in $\vec{\sigma}_m^i$ corresponding to granularity $l \in \mathcal{L}$, attribute $q \in \mathcal{Q}$, and time period t
$j, \vec{s}_{m, j}^x$	the j^{th} schedule in a set of schedules	$\vec{s}_{best, m}$	schedule to be implemented at mine m in the best solution to the MTP-MMPP found by our algorithm, \vec{s}_{best}
\mathcal{B}_m^A	a set of aggregates of blocks in \mathcal{B}_m	$IN(a), a \in \mathcal{B}_m^A$	set of grade or blast blocks forming part of aggregate $a \in \mathcal{B}_m^A$
$\overline{\mathcal{B}}_m^A$	set of grade or blast block aggregates mined (partially or completely) in an extraction schedule, for mine m , formed during the solving of \mathcal{O}_m	\mathcal{B}_m^*	set of grade blocks at mine m , together with all grade and blast blocks forming part of an aggregate in $\overline{\mathcal{B}}_m^A$

Parameters

T	number of periods in planning horizon	N	maximum number of schedules formed by each mine-side subproblem, \mathcal{O}_m
$T_k^{t=1}(b)$	tons of material k in block b at the start of the planning horizon	$T^{t=1}(s)$	tons of material on stockpile s at the start of the planning horizon
MAX_i	cap on the number of iterations of the algorithm performed	$G_{b, l, q}^{m, k}$	percentage of $q \in \mathcal{Q}$ in the granularity $l \in \mathcal{L}$ within material k of block $b \in \mathcal{B}_m$
$S_{b, l}^{m, k}$	percentage of granularity $l \in \mathcal{L}$ in the material k within block $b \in \mathcal{B}_m$	$G_{s, l, q}^m$	percentage of $q \in \mathcal{Q}$ in granularity $l \in \mathcal{L}$ of stockpile s at mine m
$S_{s, l}^m$	percentage of granularity $l \in \mathcal{L}$ in stockpile s at mine m	$R_{l, q}^{m, \omega}$	percentage of $q \in \mathcal{Q}$ in the granularity $l \in \mathcal{L}$ of any source $s \in \mathcal{S}_m$ that will be recovered after wet processing
$Y_l^{m, \omega}$	percentage of $l \in \mathcal{L}$ in any source $s \in \mathcal{S}_m$ that will be recovered after wet processing	C_τ^m	tons of material haulable by truck, per time period, at mine m

C_π	tons of material that can pass through port $\pi \in \Pi$, per time period	γ	factor by which standard deviations are increased/reduced during our decomposition-based algorithm
C_e^m	tons of material that can be extracted from blocks at mine m , per time period	C_κ^m, C_ω^m	tons of material that can be processed by the dry and wet plants at m , per period
T_R	tons of ore in a trainload	Δ_q^+, Δ_q^-	significant, and insignificant, change in the percentage of attribute $q \in \mathcal{Q}$
$U_{n,q}^{\pi,l,t}$	upper bound on $q \in \mathcal{Q}$ percentage in product $n \in N_{\pi,l}$ formed by port π in t	$L_{n,q}^{\pi,l,t}$	lower bound on the percentage of $q \in \mathcal{Q}$ in product $n \in N_{\pi,l}$ formed by port π in period t
Φ_ω^m	binary parameter equal to 1 if and only if mine m has a wet processing plant	TB_m	time period, prior to which, blast blocks at mine m cannot be mined
M_A	maximum number of blocks permitted in an aggregate	$\vec{\sigma}^+, \vec{\sigma}^-$	set of upper, and lower, bounds on the standard deviation set $\vec{\sigma}_m^i, \forall i, m \in \mathcal{M}$
Ξ_m	grade/quality target to be achieved by m , over the planning horizon, as stated in a medium-term (five year) plan		

Functions

$f(t)$	equal to $\max(1, t - 1)$	$g(t)$	equal to $\min(t + 1, T)$ where T is the number of periods in the horizon
$v_{l,q}^{m,t}(\vec{s})$	percentage of $q \in \mathcal{Q}$ in the ore of granularity $l \in \mathcal{L}$ produced by mine m in period t of the MTP-MMPP solution, \vec{s}	$\rho_m(\vec{s}, t)$	productivity of mine m , in period t , in a solution to the MTP-MMPP, \vec{s}

Decision Variables and Expressions: Monolithic MINLP

$x_{s,d}^{m,t}$	tons of source $s \in \mathcal{S}_m$, at mine m , hauled to destination $d \in \mathcal{D}_m$, in period t	$r_{m,\pi,t}^{l,n,t'}$	(floating point) number of trains of granularity $l \in \mathcal{L}$, produced by mine m in period t' , railed to port π to form part of product $n \in N_{\pi,l}$ in period t
$v_{l,q}^{m,t}$	percentage of $q \in \mathcal{Q}$ in the ore of granularity $l \in \mathcal{L}$ produced at m in t	$\bar{r}v_{\pi,n,t}^{m,l,q,t'}$	tons of attribute $q \in \mathcal{Q}$, from ore produced by mine m in period t' , blended into product $n \in N_{\pi,l}$ in period t
$\tau_l^{m,t}$	tons of granularity $l \in \mathcal{L}$ produced at mine m in period t	\vec{x}	set of $x_{s,d}^{m,t}$ variables for all mines m , periods t , material sources $s \in \mathcal{S}_m$, and material destinations $d \in \mathcal{D}_m$
$\rho_m(\vec{x}, t)$	measure of productivity (equipment usage) at mine m in period t	$\eta(\vec{x}, \vec{r}, t)$	measure of deviation present between port product composition and desired bounds, across all ports, in period t
$y_{m,t}^{\sigma,b}$	binary, taking value 1 if and only if block $b \in \mathcal{B}_m$ has been mined (partially or completely) by, or in, period t , at m	$y_{m,t}^{\tau,b}$	binary, taking value 1 if and only if block $b \in \mathcal{B}_m$ has been completely mined by, or in, period t , at m
$o_s^{m,t}$	tons of material on stockpile s , at mine m , at the start of period t	$\nu_{l,q}^{m,t}$	tons of $q \in \mathcal{Q}$ in the ore of granularity $l \in \mathcal{L}$ produced by mine m in period t

Decision Variables, Expressions, and Constants: \mathcal{O}_{Π}

$s_{m,j}$	binary variable with value 1 if and only if \mathcal{O}_{Π} selects the j^{th} schedule in set $\Omega_m^i \cup \{\vec{s}_{best,m}\}$, $\vec{s}_{m,j}$, to be enacted at mine m	$v_{l,q}^{m,t}(\vec{s}_{m,j})$	the percentage of $q \in \mathcal{Q}$ in the ore of granularity $l \in \mathcal{L}$ produced by mine m in period t of schedule $\vec{s}_{m,j}$
$\tau_l^m(\vec{s}_{m,j}, t)$	the tons of granularity $l \in \mathcal{L}$ produced by mine m in period t of schedule $\vec{s}_{m,j}$	\mathcal{P}_j^m	the productivity achieved by mine m in schedule $\vec{s}_{m,j}$
N_m	number of candidate schedules available for \mathcal{O}_{Π} to choose amongst, for mine m , in a iteration i ($N_m = \Omega_m^i \cup \{\vec{s}_{best,m}\} $)	$r_{\pi,n,t}^{m,l,j,t'}$	(floating point) number of trainloads of granularity l produced by mine m in period t' of schedule $\vec{s}_{m,j}$, blended into product $n \in N_{\pi,t}$ at port π in period t
\vec{r}^m	set of all $r_{\pi,n,t}^{m,l,j,t'}$ variables for $m \in \mathcal{M}$, $l \in \mathcal{L}$, $j \in \{1, 2, \dots, N_m\}$, $t, t' \in \{1, 2, \dots, T\}$	$\eta(\vec{r}^m, t)$	measure of deviation present between composition of port products and desired bounds, in period t

Decision Variables, Expressions, and Constants: \mathcal{O}_m

$\mathcal{O}_{m,1}$	two-time-period MIP, solved as part of the rolling horizon heuristic for schedule generation at a mine m , which minimises deviation between the grade of production at m and a grade target	$\mathcal{O}_{m,2}$	two-time-period MIP, solved as part of the rolling horizon heuristic for schedule generation at a mine m , which maximises productivity
h_1, h_2	periods scheduled in $\mathcal{O}_{m,1}$ and $\mathcal{O}_{m,2}$	$\Delta_{l,q}^{j,t}$	random real number generated from a normal distribution
$L_{l,q}^{m,t,j}$	lower bound on the percentage of $q \in \mathcal{Q}$ in the ore of granularity $l \in \mathcal{L}$ produced by mine m , in period t , of the j^{th} schedule formed by \mathcal{O}_m	$U_{l,q}^{m,t,j}$	upper bound on the percentage of $q \in \mathcal{Q}$ in the ore of granularity $l \in \mathcal{L}$ produced by mine m , in period t , of the j^{th} schedule formed by \mathcal{O}_m
$\tilde{x}_{s,d}^{m,h}$	tons of source s hauled to destination d , at m , in period h of $\mathcal{O}_{m,1}$ and $\mathcal{O}_{m,2}$	$L_{l,q}^{m,h}, U_{l,q}^{m,h}$	bounds on the percentage of $q \in \mathcal{Q}$ in granularity $l \in \mathcal{L}$ produced by m , in period h , when solving $\mathcal{O}_{m,1}$ and $\mathcal{O}_{m,2}$
$\tilde{\mathbf{x}}_m$	set of all $\tilde{x}_{s,d}^{m,h}$ variables, for $m \in \mathcal{M}$, $h \in \{h_1, h_2\}$, source s , destination d	$\tilde{Q}_{l,q}^{m,h}$	percentage of $q \in \mathcal{Q}$ in the ore of granularity $l \in \mathcal{L}$ produced by mine m , in period h , of a solution to $\mathcal{O}_{m,1}$
$\nu_{l,q}^m(\tilde{\mathbf{x}}_m, h)$	tons of $q \in \mathcal{Q}$ in the ore of granularity $l \in \mathcal{L}$ formed by mine m , in period h	$\eta(\tilde{\mathbf{x}}_m, h)$	measure of total deviation present between composition of production at mine m , in period h , and bounds given by $[L_{l,q}^{m,h}, U_{l,q}^{m,h}]$, for all $l \in \mathcal{L}$, $q \in \mathcal{Q}$
$\tau_l^m(\tilde{\mathbf{x}}_m, h)$	tons of granularity $l \in \mathcal{L}$ produced by mine m , in period h		

Appendix B: A MINLP model of the MTP-MMPP

Constraints (39)–(42) ensure that, in each period t : the capacity of dig units at each mine m , C_e^m , is not exceeded; material hauled is limited by C_r^m ; the processing of ore at each plant is equal, within a tolerance ϵ , to C_d^m for $d \in \{\kappa, \omega\}$; and the tons of ore railed to each port π does not exceed C_{π} . Constraint (43) ensures that all ore produced by each mine m in each period t is transported to a port by the end of period $t + 1$.

$$\sum_{b \in \mathcal{B}_m} \sum_{d \in \mathcal{D}_m} x_{b,d}^{m,t} \leq C_e^m \quad \forall m \in \mathcal{M}, t \in \{1, 2, \dots, T\} \quad (39)$$

$$\sum_{s \in \mathcal{S}_m} \sum_{d \in \mathcal{D}_m} x_{s,d}^{m,t} \leq C_\tau^m \quad \forall m \in \mathcal{M}, t \in \{1, 2, \dots, T\} \quad (40)$$

$$C_d^m - \epsilon \leq \sum_{s \in \mathcal{S}_m} x_{s,d}^{m,t} \leq C_d^m + \epsilon \quad \forall m \in \mathcal{M}, d \in \{\kappa, \omega\}, t \in \{1, 2, \dots, T\} \quad (41)$$

$$\sum_{m \in \mathcal{M}} \sum_{l \in \mathcal{L}} \sum_{n \in N_\pi^l} \sum_{t'=f(t)}^t T_R r_{\pi,n,t}^{m,l,t'} \leq C_\pi \quad \forall \pi \in \Pi, t \in \{1, 2, \dots, T\} \quad (42)$$

$$\sum_{t=t'}^{g(t')} \sum_{\pi \in \Pi} \sum_{n \in N_{\pi,l}} T_R r_{\pi,n,t}^{m,l,t'} = \tau_l^{m,t'} \quad \forall m, l, t' \in \{1, 2, \dots, T\} \quad (43)$$

Constraint (44) constrains the value of binaries $y_{m,t}^{\sigma,b}$ (1 if the mining of b has been scheduled by, or during, t) and $y_{m,t}^{\tau,b}$ (1 if b is scheduled to be entirely extracted by, or during, t). Vertical and disjunctive block precedences are respectively expressed in Constraints (45)–(46).

$$y_{m,t}^{\tau,b} \sum_k T_k^{t=1}(b) \leq \sum_{t'=1}^t \sum_{d \in \mathcal{D}_m} x_{b,d}^{m,t'} \leq y_{m,t}^{\sigma,b} \sum_k T_k^{t=1}(b) \quad \forall m \in \mathcal{M}, b \in \mathcal{B}_m, t \in \{1, 2, \dots, T\}, \quad (44)$$

$$y_{m,t}^{\tau,b'} \geq y_{m,t}^{\sigma,b} \quad \forall m \in \mathcal{M}, b \in \mathcal{B}_m, b' \in \mathcal{A}_{m,b}^\wedge, t \in \{1, 2, \dots, T\} \quad (45)$$

$$\sum_{b' \in \mathcal{A}_{m,b}^\vee} y_{m,t}^{\tau,b'} \geq y_{m,t}^{\sigma,b} \quad \forall m \in \mathcal{M}, b \in \mathcal{B}_m, t \in \{1, 2, \dots, T\}, \quad (46)$$

Constraint (47) ensures that no blast blocks are mined prior to a specific time period, TB_m , at each mine $m \in \mathcal{M}$. Prior to TB_m , only grade blocks can be extracted at mine m . Prior to TB_m , only grade blocks can be extracted at m . Constraints (48)–(51) ensure that stockpile capacities are respected and that no more than $o_s^{m,t}$ tons (the tons of ore on stockpile s at the start of t) can be extracted from $s \in \Theta_m \cup \Lambda_m$ in t .

$$y_{m,t}^{\sigma,b} = 0 \quad \forall m \in \mathcal{M}, b \in \mathcal{B}_m^b, t < TB_m, \quad (47)$$

$$o_s^{m,t} = T^{t=1}(s) \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, t = 1, \quad (48)$$

$$o_s^{m,t} = o_s^{m,t-1} - \sum_{d \in \{\kappa, \omega\}} x_{s,d}^{m,t-1} + \sum_{b \in \mathcal{B}_m} x_{b,s}^{m,t-1} \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, t \in \{2, 3, \dots, T\}, \quad (49)$$

$$o_s^{m,t} - \sum_{d \in \{\kappa, \omega\}} x_{s,d}^{m,t} + \sum_{b \in \mathcal{B}_m} x_{b,s}^{m,t} \leq C_s^m \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, t \in \{2, 3, \dots, T\}, \quad (50)$$

$$o_s^{m,t} - \sum_{d \in \{\kappa, \omega\}} x_{s,d}^{m,t} \geq 0 \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, t \in \{2, 3, \dots, T\}, \quad (51)$$

Constraints (52)–(55) define the tons of each granularity $l \in \mathcal{L}$ and attribute $q \in \mathcal{Q}$ residing on each stockpile $s \in \Theta_m \cup \Lambda_m$ at the start of period t .

$$o_{s,l}^{m,t} = S_{s,l}^{m,t=1} T^{t=1}(s) \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, l \in \mathcal{L}, t = 1, \quad (52)$$

$$o_{s,l}^{m,t} = o_{s,l}^{m,t-1} - \sum_{d \in \{\kappa, \omega\}} a_{s,l}^{m,t-1,d} + \sum_{b \in \mathcal{B}_m} S_{b,l}^m x_{b,s}^{m,t-1} \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, l \in \mathcal{L}, \quad (53)$$

$$o_{s,l,q}^{m,t} = S_{s,l}^{m,t=1} G_{s,l,q}^{m,t=1} T^{t=1}(s) \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, l \in \mathcal{L}, \quad (54)$$

$$o_{s,l,q}^{m,t} = o_{s,l,q}^{m,t-1} - \sum_{d \in \{\kappa, \omega\}} a_{s,l,q}^{m,t-1,d} + \sum_{b \in \mathcal{B}_m} S_{b,l}^m G_{b,l,q}^m x_{b,s}^{m,t-1} \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, l \in \mathcal{L}, q \in \mathcal{Q}, \quad (55)$$

$$t \in \{2, 3, \dots, T\},$$

Constraints (56)–(57) ensure that the composition of material leaving each stockpile $s \in \Theta_m \cup \Lambda_m$ for a processing plant $d \in \{\kappa, \omega\}$ in each period t is equal to that of the stockpile at the start of t .

$$a_{s,l}^{m,t,d} o_{s,d}^{m,t} = x_{s,d}^{m,t} o_{s,l}^{m,t} \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, d \in \{\kappa, \omega\}, l \in \mathcal{L}, t \in \{1, 2, \dots, T\} \quad (56)$$

$$a_{s,l,q}^{m,t,d} o_{s,l}^{m,t} = a_{s,l}^{m,t,d} o_{s,l,q}^{m,t} \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Lambda_m, d \in \{\kappa, \omega\}, l \in \mathcal{L}, q \in \mathcal{Q}, t \in \{1, 2, \dots, T\} \quad (57)$$

Constraints (58)–(62) prevent material movement along invalid pathways, and ensure that no more high (*hi*), low grade (*lo*), or waste (*w*) material is extracted from a block than exists at the start of $t = 1$.

$$x_{s,\kappa}^{m,t} = 0 \quad \forall m \in \mathcal{M}, s \in \Lambda_m \cup \Delta_m, t \in \{1, 2, \dots, T\} \quad (58)$$

$$x_{s,\omega}^{m,t} = 0 \quad \forall m \in \mathcal{M}, s \in \Theta_m \cup \Delta_m, t \in \{1, 2, \dots, T\} \quad (59)$$

$$\sum_{t=1}^T \sum_{d \in \mathcal{D}_m} x_{b,d}^{m,t} x_{b,\kappa}^{m,t} + \sum_{s \in \Theta_m} x_{b,s}^{m,t} \leq T^{t=1}(b) \quad \forall m \in \mathcal{M}, b \in \mathcal{B}_m, \quad (60)$$

$$\sum_{t=1}^T x_{b,\omega}^{m,t} + \sum_{s \in \Lambda_m} x_{b,s}^{m,t} \leq T^{t=1}(b) \quad \forall m \in \mathcal{M}, b \in \mathcal{B}_m, \quad (61)$$

$$\sum_{t=1}^T \sum_{s \in \Delta_m} x_{b,s}^{m,t} \leq T_w^{t=1}(b) \quad \forall m \in \mathcal{M}, b \in \mathcal{B}_m, \quad (62)$$

Variables $\tau_l^{m,t}$ and $\nu_{l,q}^{m,t}$ are defined in Constraints (64)–(65). The tons of $q \in \mathcal{Q}$ in the ore of granularity $l \in \mathcal{L}$ produced by mine m in t , denoted $\nu_{l,q}^{m,t}$, is defined in Equation (63).

$$\nu_{l,q}^{m,t} = \sum_{\lambda \in \Lambda_m} R_{l,q}^{m,\omega} a_{\lambda,l,q}^{m,t,\omega} + \sum_{\theta \in \Theta_m} a_{\theta,l,q}^{m,t,\kappa} \quad \forall m \in \mathcal{M}, t \in \{1, 2, \dots, T\}, l \in \mathcal{L}, q \in \mathcal{Q}, \quad (63)$$

$$+ \sum_{b \in \mathcal{B}_m} S_{b,l}^{m,hi} G_{b,l,q}^{m,hi} x_{b,\kappa}^{m,t} + S_{b,l}^{m,lo} G_{b,l,q}^{m,lo} R_{l,q}^{m,\omega} x_{b,\omega}^{m,t}$$

$$\tau_l^{m,t} = \sum_{\lambda \in \Lambda_m} Y_l^{m,\omega} a_{\lambda,l}^{m,t,\omega} + \sum_{\theta \in \Theta_m} a_{\theta,l}^{m,t,\kappa} \quad \forall m \in \mathcal{M}, t \in \{1, 2, \dots, T\}, l \in \mathcal{L}, \quad (64)$$

$$+ \sum_{b \in \mathcal{B}_m} S_{b,l}^{m,hi} x_{b,\kappa}^{m,t} + S_{b,l}^{m,lo} Y_l^{m,\omega} x_{b,\omega}^{m,t}$$

$$\nu_{l,q}^{m,t} \tau_l^{m,t} = \nu_{l,q}^{m,t} \quad \forall m \in \mathcal{M}, t \in \{1, 2, \dots, T\}, l \in \mathcal{L}, q \in \mathcal{Q}, \quad (65)$$

Constraint (66) defines variable $\bar{r}v_{\pi,n,t}^{m,l,q,t'}$, used in Equation (2) to compute the extent to which the composition of products formed at each port deviate from desired bounds.

$$T_R r_{\pi,n,t}^{m,l,t'} v_{l,q}^{m,t'} = \bar{r} v_{\pi,n,t}^{m,l,q,t'} \quad \forall m \in \mathcal{M}, t \in \{1, 2, \dots, T\}, t' \in \{f(t), t\}, l \in \mathcal{L}, q \in \mathcal{Q}, n \in N_{\pi,l}, \quad (66)$$

We define a set of separation constraints to prevent the extraction of blocks that are not immediately accessible on the mining face. Blom et al. (2014) provide a set of such constraints for the single time period MMPP, which we generalise to the multiple time period setting. A block $b \in \mathcal{B}_m$ is defined to lie on a mining face if $|\mathcal{A}_{m,b}^\vee| = 0$ (ie. no adjacent blocks must be mined before b can be accessed). The set $\mathcal{P}'(\mathcal{B}_m)$ contains all contiguous sets of blocks $\mathcal{B}'_m \subset \mathcal{B}_m$ for which $\nexists b' \in \mathcal{B}'_m$ such that $|\mathcal{A}_{m,b'}^\vee| = 0$. The set $\mathcal{N}(\mathcal{B}_m, \mathcal{B}'_m)$ contains all blocks $b'' \in \mathcal{B}_m \setminus \mathcal{B}'_m$ for which $\exists b' \in \mathcal{B}'_m$ such that $b'' \in \mathcal{A}_{m,b'}^\vee$ (ie. the ‘neighbours’ of the blocks in set \mathcal{B}'_m). Selected instances of Constraint set (67) are added to the MINLP via a separation algorithm.³ Each constraint in this set states that if a block in set \mathcal{B}'_m is mined in period t , then at least one of the neighbours of \mathcal{B}'_m must have been completely extracted by or during t .

$$\sum_{b'' \in \mathcal{N}(\mathcal{B}_m, \mathcal{B}'_m)} y_{m,t}^{\tau,b''} \geq \frac{1}{|\mathcal{B}'_m|} \sum_{b' \in \mathcal{B}'_m} y_{m,t}^{\sigma,b'} \quad \forall m \in \mathcal{M}, \mathcal{B}'_m \in \mathcal{P}'(\mathcal{B}_m), t \in \{1, 2, \dots, T\} \quad (67)$$

Indicators $y_{m,t}^{\tau,b}$ and $y_{m,t}^{\sigma,b}$, for all $m \in \mathcal{M}$, $b \in \mathcal{B}_m$, and $t \in \{1, 2, \dots, T\}$, are restricted to binary values, while all remaining variables are restricted to non-negative reals.

³ Instances of Constraint (67) are added to our implementation of this model via the use of lazy constraint callbacks. For brevity, further details of this separation algorithm have been omitted from this paper.