Solving mixed integer nonlinear programming problems for mine production planning with stockpiling

Andreas Bley[1], Natashia Boland[2], Gary Froyland[3], and Mark Zuckerberg[4]

[1] Institute for Mathematics, Technical University Berlin, Berlin, Germany
[2] School of Mathematical & Physical Sciences, University of Newcastle, Callaghan, Australia
[3] School of Mathematics and Statistics, University of New South Wales, Sydney, Australia
[4] Resource and Business Optimization Group Function, BHP Billiton Ltd., Melbourne, Australia

Preprint 2012/31

August 28, 2012

Abstract. The open-pit mine production scheduling problem has received a great deal of attention in recent years, both in the academic literature, and in the mining industry. Optimization approaches to strategic planning for mine exploitation have become the industry standard, as the recent review of Newman *et al.* [19] highlights. However most of these approaches focus on extraction sequencing, and don't consider the material flow after mining. In particular, the use of stockpiling to manage processing plant capacity, and the interplay of material flows from mine to stockpile, mine to processing plant and stockpile to plant, has not been treated as an integrated part of mine schedule optimization.

One of the key reasons is that material of different grades becomes mixed on a stockpile, leading to difficult nonconvex, nonlinear optimization models. Here we show that the special structure of such models can be exploited to yield effective algorithms that incorporate post-mining material flows and stockpile management as an integrated part of mine production scheduling. The results give a more realistic assessment of the NPV that can be realized by a mining project than is possible with current approaches.

We address the solution of the open pit mine production scheduling problem (OPMPSP) with a single stockpile (OPMPSP+S). The addition of a stockpile adds a relatively small number of quadratic constraints to the formulation of the OPMPSP and turns the problem from a mixed-integer linear into a mixed-integer *non*linear program. We develop several extended formulations of the OPMPSP+S and discuss the strength of the linear outer approximations obtained by relaxing their nonlinear constraints. We also introduce an aggressive branching scheme that can force the violation of the quadratic stockpiling constraints to be arbitrarily close to zero and a primal heuristic that produces a fully feasible solution of OPMPSP+S from an integer feasible solution of OPMPSP which violates these constraints. Combining these two techniques with a branch-and-bound approach,we obtain an algorithm that yields fully feasible solutions of OPMPSP+S arbitrarily close to the optimum. Experimental results for realistic benchmark instances show that this algorithm is very efficient in practice.

Our methodology is easily extendable to multiple stockpiles.

1 Introduction

Open pit mine projects typically run for several decades and the optimisation of stategic plans is a crucial element for the successful planning of projects. A commonly used criterion for comparison of different strategic plans for the extraction of the valuable material from the ground is *net present value* (NPV). The NPV optimisation of strategic plans for open pit mines has a long history and has been approached from a number of different angles.

The two main classes of constraints are safe pit wall slopes and annual production limits. Material blending constraints are also sometimes included, and belong to the second category.

The former, which are modelled as precedence constraints, require binary variables to be modelled correctly, but the latter do not. Individually, it can be shown that these constraints do not pose computational challenges. In combination, however, they produce an NP-hard problem (see eg. [11]), often known as the Open Pit Mining Scheduling Problem (OPMPSP) in the literature.

The OPMPSP was modelled as a mixed integer linear program by Johnson [14], but at that time neither computing power nor suitable algorithms existed capable of solving even small instances. Classical approaches therefore attempted to derive an approximate solution by solving several passes of a problem containing only the precedence constraints, each parameterized differently. This was first suggested by Lerchs and Grossman [15], and has been implemented in Gemcom's popular Whittle [26] mine planning software. Lerchs and Grossman's seminal algorithm for the maximum weight closure problem, which is equivalent to the precedence constrained problem, is not a polynomial time algorithm in the worst case (it is pseudopolynomial). Hochbaum [12], however, derived a polynomial time version, and there are also various ways to reduce max closure to other graph theoretical problems for which highly efficient algorithms are known (see for example [14,21,23]).

Another approach of a similar flavor is to use Lagrangian relaxation of the constraints other than the precedence constraints in order to attempt to obtain a valid solution to the full problem by solving only precedence problems (Dagdelen and Johnson [9] and Akaike and Dagdelen [1]). Hoerger *et al.* [13] are one of the first to solve a MILP model without Lagrangian relaxation. Other authors have proposed various dynamic programming algorithms. Examples are Onur and Dowd [20], Wang [27] and Tolwinski and Underwood [25].

As computing power has increased and mixed integer programming algorithms have improved, MILP has emerged as a realistic, and arguably as the preferred, method for solving OPMPSP. In practical settings, a full sized problem can contain hundreds of thousands to millions of binary variables and millions to tens of millions of linear constraints, which still places it beyond the reach of commercial MILP solvers, but a number of techniques have been developed recently to speed up the solution process. These include aggregation [22], see also [16] and disaggregation [5,11] approaches, variable fixing and specialised cutting planes [3,4] and techniques for rapidly solving the LP relaxations [2,5,8,17]. Caccetta and Hill [7] have also published a paper describing a branch and cut based algorithm designed for large scale problems, but due to commercial confidentiality considerations they have not released details of their method. Fricke [10] contains a comprehensive survey of published work treating OPMPSP. A more recent survey is Newman *et al.* [19].

All of these approaches treat the pure OPMPSP, in which material mined is either processed or sent to waste, never to be reclaimed. In practice, however, *stockpiles* are used to hold valuable material that cannot be processed immediately due to process capacity limits. This valuable material is not lost, but may be reclaimed and processed at a future time when there is spare processing capacity. In exploring the discrepancy between short- and medium-term operational planning, for example, Yarmuch and Ortiz [28] schedule flows between the mine and two stockpiles, one high grade and one low grade, and the processing plant. They avoid the challenge of nonlinearity introduced by stockpiling by solving the problem period by period, re-calculating the stockpiles' grades at the end of each period.

In the context of mine project NPV optimization, while the ability to stockpile can significantly increase NPV, the introduction of stockpiles into MILP formulations of the OPMPSP have so far been *ad hoc* and approximate [7,24]. The main reason for this is that when material is sent to stockpile from one part of the mine, it is mixed (approximately homogeneously) with material from other parts of the mine. When material is reclaimed, it is not reclaimed at the quality which it entered the stockpile, but at the average quality of the stockpile in its current state. This gives rise to constraints that ensure that (the ratio of) the quality of material reclaimed from the stockpile and the quality of material in the stockpile is the same. These ratio constraints produce bilinear constraints, which are quadratic nonlinear constraints and challenging from a computation point of view.

We will call the OPMPSP model augmented with stockpiling constraints the OPMPSP+S model. In the present work, we treat a single stockpile, but our techniques are extendable to multiple stockpiles in a straightforward way. A summary of our approach is as follows. We take advantage of the techniques developed to speed up solution of the OPMPSP, including variable fixing, and specialised cuts [3,4]. The LP-relaxation of the OPMPSP formulation with the option to reclaim stockpiled material, but without the quadratic constraints ensuring the same quality of material in the stockpile and material taken from the stockpile, is a very weak relaxation of the OPMPSP+S. We therefore develop several extended formulations of OPMPSP+S and prove that they lead to tighter LP-relaxations. Using these tighter reformulations of OPMPSP+S we apply an additional branching procedure that can reduce the violation of the quadratic constraints to arbitrarily low levels. Finally, given a candidate solution of OPMPSP+S that is integer feasible but violates the quadratic constraints of OPMPSP+S we employ a primal heuristic to construct a fully feasible solution of OPMPSP+S, i.e., a solution satisfying both the integrality constraints and the quadratic constraint. Integrated into a branch-and-cut approach, these two techniques allow us to refine the outer linear relaxation of the problem adaptively via branching and to turn any integer feasible solution of a (refined) relaxation into a fully feasible solution of the original problem. Thus, our approach can produce fully feasible solutions of OPMPSP+S with objective values that are arbitrarily close to the optimal objective value.

Most aspects of our methodology can be applied to general mixed-integer nonlinear programs where the nonlinear constraints are bilinear functions that control ratios of quantities. In particular, some of the tighter reformulations we introduce, and the aggressive branching scheme, are very general techniques.

All aspects of the methodology are also easily extended to multiple stockpiles, with or without grade range specifications.

An outline of the paper is as follows. In Section 2 we describe the OPMPSP+S in greater detail. Section 3 introduces our tighter reformulations of OMPSPSP+S. Section 4 begins with an overview of how our proposed methodologies will be applied to the models from Section 3, and then proceeds to detail these methodologies, including an aggressive branching procedure and a primal heuristic. Section 5 details numerical experiments.

2 Problem Description and MINLP formulation

In this section we describe in detail our model of the *open pit mine production scheduling problem with stockpile* (OPMPSP+S). The region of earth in the ground that is under consideration for mining is called an *ultimate pit*. The ultimate pit provides an efficient excavation boundary within which decisions will be made to mine or not mine material. The ultimate pit is typically discretised into *blocks*, which represent the smallest possible size of material that may be selectively mined by the mining equipment. Note that there can often be millions of blocks in a single pit. While, in principle, one can try to optimise mining and processing decisions for each individual block, these blocks are often grouped together to form *aggregates* in order to reduce the size of the resulting OPMPSP. Various techniques exist for computing aggregates, for example, the fundamental tree method [22].

Geological estimates of the contents of the blocks, and hence the aggregates, are obtained by drilling. For each aggregate $i \in [N] := \{1, \ldots, N\}$ one obtains an estimate of total *rock* mass R_i and the mass of various attributes such as various mineral contents; see Table 4 at the end of the article for basic parameter notation. For simplicity, we consider only a single valuable attribute, namely a single valuable metal, and we denote the mass of this *metal* in aggregate i by A_i. It will turn out that our tightened formulations in Section 3 can automatically handle multiple attributes; we discuss this more later. The total amount of rock in aggregate i that is sufficiently valuable to consider processing is denoted O_i; such material is called *ore*. The standard OPMPSP model (c.f. [7]) assumes that the attributes are distributed homogeneously throughout the aggregate. Automatic disaggregation techniques such as those in [5] may be employed to mediate this assumption, but for simplicity of exposition, we have not done this here.

In addition to the spatial discretisation into aggregates, the life of the mine is discretised into time periods indexed by $t \in [T] := \{1, \ldots, T\}$. The durations of the time periods need not be the same. The mining operation consists of several steps. In the first step, rock is *mined* from the pit. We use two sets of variables to control the mining. For each aggregate $i \in [N]$ and each time period $t \in [T]$, we introduce the variables

$$x_{i,t} \in \{0,1\}\ , \qquad x_{i,t} = \begin{cases} 1, & \text{if aggregate } i \text{ has been \textit{completely} mined by the end} \\ & \text{of time period } t; \\ 0, & \text{otherwise,} \end{cases}$$

$$y_{i,t} \in [0,1]\ , \qquad y_{i,t} \text{ is the fraction of aggregate } i \text{ mined during time period } t.$$

The relation between these variables is expressed by the constraints

$$x_{i,t-1} \leqslant x_{i,t} \qquad \text{for } i \in [N],\ t \in \{2, \ldots, T\}, \tag{1}$$

$$x_{i,t} \leqslant \sum_{\tau=1}^{t} y_{i,\tau} \qquad \text{for } i \in [N],\ t \in [T]. \tag{2}$$

A limit on the fraction of mined material is enforced by

$$\sum_{t=1}^{T} y_{i,t} \leqslant 1 \qquad \text{for } i \in [N]. \tag{3}$$

Precedence constraints model the requirement that wall slopes are not too steep, ensuring the safety of the mine. Technically, these constraints demand that, before the mining of aggregate i may be started, each immediate predecessor aggregate $j \in \mathcal{P}(i)$ (see Table 4) must have been completely mined. A precedence-feasible extraction sequence is then guaranteed by the constraints

$$\sum_{\tau=1}^{t} y_{i,\tau} \leqslant x_{j,t} \qquad \text{for } i \in [N],\ j \in \mathcal{P}(i),\ t \in [T]. \tag{4}$$

Note that the set $\mathcal{P}(i)$ only contains the *immediate* predecessors aggregates of aggregate i. Aggregates that indirectly preceed i, i.e., that are predecessors of predecessors of i, are not contained in $\mathcal{P}(i)$.

Once rock has been mined, it may take one of three paths: it may be *processed* to extract the metal from the rock, it may be *stockpiled* on the ground nearby the pit for processing at a later time, or it may be sent to a waste dump, from which it is never reclaimed. For each aggregate $i \in [N]$ and each time period $t \in [T]$, we introduce two variables to model these options, namely

$z^{\mathrm{p}}_{i,t} \in [0,1]$, $z^{\mathrm{p}}_{i,t}$ is the fraction of aggregate i sent directly for processing in period t,

$z^{\mathrm{s}}_{i,t} \in [0,1]$, $z^{\mathrm{s}}_{i,t}$ is the fraction of aggregate i sent to the stockpile in time period t.

The fraction of aggregate i that is sent to the waste dump in period t is $y_{i,t} - z^{\mathrm{p}}_{i,t} - z^{\mathrm{s}}_{i,t}$. The constraints

$$z^{\mathrm{p}}_{i,t} + z^{\mathrm{s}}_{i,t} \leqslant y_{i,t} \qquad \text{for } i \in [N],\ t \in [T] \tag{5}$$

ensure that, for each aggregate, the sum of the fractions sent for processing and to the stockpile during one time period does not exceed the fraction mined.

To model the material flows through the stockpile, we introduce the following continuous variables for each $t \in [T]$:

$o^{\mathrm{p}}_t,\ a^{\mathrm{p}}_t \geqslant 0$, o^{p}_t (a^{p}_t) is the total amount of ore (resp. metal) sent from the stockpile to the processing plant during time period t, and

$o^{\mathrm{s}}_t,\ a^{\mathrm{s}}_t \geqslant 0$, o^{s}_t (a^{s}_t) is the total amount of ore (resp. metal) remaining in the stockpile during time period t (i.e., excluding material newly arriving in the stockpile in period t).

We assume that material reclaimed from the stockpile and sent to the processing plant during time period t is removed from the stockpile at the beginning of time period t, whereas material extracted from the pit and sent to the stockpile during time period t is put onto the stockpile at the end of time period t. Furthermore, we assume that the stockpile is empty at the beginning of the first time period and at the end of the planning horizon.

Following these assumptions, the material flow conservation constraints for the stockpile are

$$o^{\mathrm{s}}_{t-1} + \sum_{i=1}^{N} O_i z^{\mathrm{s}}_{i,t-1} = o^{\mathrm{s}}_t + o^{\mathrm{p}}_t \qquad \text{for } t \in \{2,\dots,T\}, \tag{6}$$

$$a^{\mathrm{s}}_{t-1} + \sum_{i=1}^{N} A_i z^{\mathrm{s}}_{i,t-1} = a^{\mathrm{s}}_t + a^{\mathrm{p}}_t \qquad \text{for } t \in \{2,\dots,T\}, \text{ and} \tag{7}$$

$$o^{\mathrm{s}}_1 = o^{\mathrm{p}}_1 = o^{\mathrm{s}}_T = a^{\mathrm{s}}_1 = a^{\mathrm{p}}_1 = a^{\mathrm{s}}_T = 0\ . \tag{8}$$

The limited mining and processing capacities (denoted M_t and P_t, respectively) in each period are reflected by the constraints

$$\sum_{i=1}^{N} R_i y_{i,t} \leq M_t \qquad \text{for } t \in [T], \text{ and} \tag{9}$$

$$\sum_{i=1}^{N} O_i z^{\mathrm{p}}_{i,t} + o^{\mathrm{p}}_t \leq P_t \qquad \text{for } t \in [T]. \tag{10}$$

As discussed in the introduction, material placed on the stockpile is assumed to be mixed homogeneously before possible later removal for processing. Hence, we need to ensure that, at the beginning of each time period, the ore-metal ratio of the material sent from stockpile to processing equals the ore-metal ratio in the stockpile itself. Otherwise, the profitable metal could be sent to processing while the ore, only incurring processing costs, could remain in the stockpile. The equality of these ratios can be enforced by

$$\frac{a_t^{\mathrm{p}}}{a_t^{\mathrm{s}} + a_t^{\mathrm{p}}} = \frac{o_t^{\mathrm{p}}}{o_t^{\mathrm{s}} + o_t^{\mathrm{p}}} \qquad \text{for } t \in [T].$$

To avoid the singularities from zero denominators, we reformulate these constraints as

$$a_t^{\mathrm{p}}(o_t^{\mathrm{s}} + o_t^{\mathrm{p}}) = o_t^{\mathrm{p}}(a_t^{\mathrm{s}} + a_t^{\mathrm{p}}) \qquad \text{for } t \in [T]. \tag{11}$$

Finally, the net present value of a mine schedule is calculated as

$$\begin{aligned} &NPV(y, z^{\mathrm{p}}, o^{\mathrm{p}}, a^{\mathrm{p}}) = \\ &\qquad \sum_{t=1}^{T} \delta_t \left[c \left(a_t^{\mathrm{p}} + \sum_{i=1}^{N} A_i z_{i,t}^{\mathrm{p}} \right) - p \left(o_t^{\mathrm{p}} + \sum_{i=1}^{N} O_i z_{i,t}^{\mathrm{p}} \right) - m \sum_{i=1}^{N} R_i y_{i,t} \right], \end{aligned}$$

where c is the sales price per unit of metal, p is the processing cost per unit of ore, m is the mining cost per unit of rock, and δ_t is the discount factor that applies to time period t. For homogeneous time periods and constant interest rate $q \geqslant 0$ per time period, the profit made in time period t is typically multiplied by a discount factor of $\delta_t = 1/(1+q)^t$.

All in all, we obtain a formulation of the open pit mine production scheduling problem with one attribute ("metal") and a single, infinite-capacity stockpile, which we will call the *natural formulation* (or *basic formulation*) of OPMPS+S throughout the paper:

$$\begin{aligned} \max \quad & NPV(y, z^{\mathrm{p}}, o^{\mathrm{p}}, a^{\mathrm{p}}) \\ \text{s.t.} \quad & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}) \text{ satisfies } (1) - (11), \\ & x \in \{0,1\}^{N \times T}, \\ & y, z^{\mathrm{p}}, z^{\mathrm{s}} \in [0,1]^{N \times T}, \\ & o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}} \in \mathbb{R}_{\geq 0}^{T}. \end{aligned} \tag{NF}$$

For notational convenience, we denote by S_{NF}, $S_{\text{NF-IP}}$, and $S_{\text{NF-LP}}$ the feasible solutions sets of (NF), of the mixed-integer linear relaxation obtained by dropping the bilinear constraints (11) from (NF), and of the linear relaxation obtained by dropping both the bilinear constraints (11) and the integrality constraints from (NF), i.e.,

$$\begin{aligned} S_{\text{NF}} := \big\{ & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}) \in [0,1]^{4(N \times T)} \times \mathbb{R}_{\geq 0}^{4T} : \\ & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}) \text{ satisfies } (1) - (11),\ x \in \{0,1\}^{N \times T} \big\}, \\ S_{\text{NF-IP}} := \big\{ & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}) \in [0,1]^{4(N \times T)} \times \mathbb{R}_{\geq 0}^{4T} : \\ & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}) \text{ satisfies } (1) - (10),\ x \in \{0,1\}^{N \times T} \big\}, \\ S_{\text{NF-LP}} := \big\{ & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}) \in [0,1]^{4(N \times T)} \times \mathbb{R}_{\geq 0}^{4T} : \\ & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}) \text{ satisfies } (1) - (10) \big\}, \end{aligned}$$

A more general setting comprising multiple attributes, multiple stockpiles, finite stockpiling capacity, initially non-empty stockpiles, or blending constraints can easily be modelled by minor extensions and modifications.

3 Stronger formulations

In this section we describe two techniques to strengthen the relaxations of OPMPSP+S. The first one individually tracks the fractions of each aggregate residing in the stockpile and reclaimed from the stockpile in each time period. This requires additional continuous variables, but produces considerably tighter relaxations. The second approach uses additional binary variables and linear constraints to produce a piece-wise linear outer approximation of the quadratic mixing constraints (11). Naturally, this approximation of the non-linear constraints leads to stronger relaxations than simply dropping the constraints.

3.1 The aggregate tracking formulation

In the (NF) formulation the material from *all* aggregates sent from the pit to the stockpile is aggregated into variables o^{s} and a^{s}, which only describe the total ore and metal content of the stockpile. Alternatively, we may track the material flow through the stockpile at an individual aggregate level. For this, we introduce for each aggregate i and time period t the additional variables

$z^{\mathrm{s,p}}_{i,t} \in [0,1]$, $z^{\mathrm{s,p}}_{i,t}$ is the fraction of aggregate i sent from stockpile for processing during time period t, and

$z^{\mathrm{s,s}}_{i,t} \in [0,1]$, $z^{\mathrm{s,s}}_{i,t}$ is the fraction of aggregate i remaining in the stockpile throughout time period t (ie. excluding material newly arriving in the stockpile in period t).

The concept of these variables is illustrated in Figure 1.

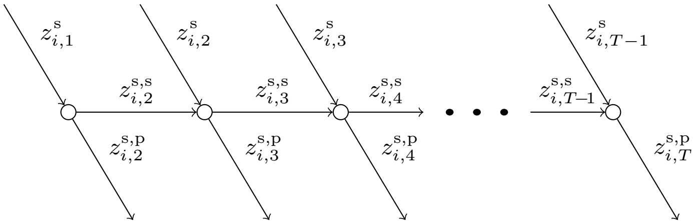

Fig. 1. Material flow over time in and out of the stockpile.

For each aggregate, these fractions must satisfy the aggregate-wise material conservation constraints

$$z^{\mathrm{s,s}}_{i,t-1} + z^{\mathrm{s}}_{i,t-1} = z^{\mathrm{s,s}}_{i,t} + z^{\mathrm{s,p}}_{i,t} \qquad \text{for } i \in [N],\ t \in \{2,\dots,T\}, \text{ and} \tag{12}$$

$$z^{\mathrm{s,s}}_{i,1} = z^{\mathrm{s,p}}_{i,1} = z^{\mathrm{s,s}}_{i,T} = 0 \qquad \text{for } i \in [N]. \tag{13}$$

It is not hard to see by induction that these constraints, together with the variable domains and (3) and (5), imply that $z^{\mathrm{s,s}}_{i,t} + z^{\mathrm{s,p}}_{i,t} \leqslant 1$ for all i, t. This is helpful to note for later use.

The total amounts of ore and metal remaining in the stockpile and taken from the stockpile during a time period are given by the aggregate fractions remaining in and taken from the stockpile in this period via the following equations:

$$o^{\mathrm{s}}_t = \sum_{i=1}^{N} O_i z^{\mathrm{s,s}}_{i,t} \qquad \text{for } t \in [T], \tag{14}$$

$$o^{\mathrm{p}}_t = \sum_{i=1}^{N} O_i z^{\mathrm{s,p}}_{i,t} \qquad \text{for } t \in [T], \tag{15}$$

$$a^{\mathrm{s}}_t = \sum_{i=1}^{N} A_i z^{\mathrm{s,s}}_{i,t} \qquad \text{for } t \in [T], \tag{16}$$

$$a^{\mathrm{p}}_t = \sum_{i=1}^{N} A_i z^{\mathrm{s,p}}_{i,t} \qquad \text{for } t \in [T]. \tag{17}$$

With the individual aggregate fractions in the stockpile at hand, we now can model the stockpile mixing constraint by demanding that, for each time period t, the ratio of material sent from the stockpile to processing to material remaining in the stockpile is equal for all aggregates $i \in [N]$. Introducing additional variables $f_t \in [0,1]$ for each time period t, called *out-fractions*, these constraints can be written as

$$\frac{z^{\mathrm{s,p}}_{i,t}}{z^{\mathrm{s,s}}_{i,t} + z^{\mathrm{s,p}}_{i,t}} = f_t \qquad \text{for } i \in [N],\ t \in [T],$$

or, avoiding zero denominators, as

$$z^{\mathrm{s,p}}_{i,t}(1 - f_t) = z^{\mathrm{s,s}}_{i,t} f_t \qquad \text{for } i \in [N],\ t \in [T]. \tag{18}$$

It is easy to verify that these constraints, together with equalities (14)–(17), imply the original mixing constraints (11); see Theorem 1 below. Hence, the total ore-metal mixing constraints (11) can be omitted from the model if the aggregate-wise mixing constraints (18) are given.

With these additional variables and constraints, we obtain the *aggregate tracking formulation*:

$$\begin{aligned} \max \quad & NPV(y, z^{\mathrm{p}}, o^{\mathrm{p}}, a^{\mathrm{p}}) \\ \text{s. t.} \quad & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, f) \text{ satisfies (1)–(10), (12)–(18),} \\ & \qquad\qquad x \in \{0,1\}^{N \times T}, \\ & y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}} \in [0,1]^{N \times T}, \\ & \qquad o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}} \in \mathbb{R}^{T}_{\geq 0}, \\ & \qquad\qquad f \in [0,1]^{T}. \end{aligned} \tag{AT}$$

Note that the variables o^{s}_t, o^{p}_t, a^{s}_t, and a^{p}_t describing the total ore and metal flow through the stockpile can be eliminated from the model using equalities (14)–(17). For simplicity of presentation, we nevertheless keep these variables and equalities (14)–(17) in the formulations.

As for the natural formulation, we denote by S_{AT} the solution set of (AT), by $S_{\text{AT-IP}}$ the solution set of the mixed-integer linear programming relaxation obtained by dropping constraints (18), and by $S_{\text{AT-LP}}$ the solution set of the linear relaxation of the latter, i.e.,

$$\begin{aligned}
S_{\mathrm{AT}} := \big\{ & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, f) \in [0,1]^{6(N\times T)} \times \mathbb{R}^{4T}_{\geq 0} \times [0,1]^T : \\
& (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, f) \text{ satisfies (1)–(10), (12)–(18),} \\
& x \in \{0,1\}^{N\times T} \ \big\} \\
S_{\text{AT-IP}} := \big\{ & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, f) \in [0,1]^{6(N\times T)} \times \mathbb{R}^{4T}_{\geq 0} \times [0,1]^T : \\
& (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, f) \text{ satisfies (1)–(10), (12)–(17),} \\
& x \in \{0,1\}^{N\times T} \ \big\} \\
S_{\text{AT-LP}} := \big\{ & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, f) \in [0,1]^{6(N\times T)} \times \mathbb{R}^{4T}_{\geq 0} \times [0,1]^T : \\
& (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, f) \text{ satisfies (1)–(10), (12)–(17)} \big\}
\end{aligned}$$

Note that the mixed-integer linear relaxation obtained by leaving out (18) is equivalent to a "warehouse" stockpile model, where each block may be stored and reclaimed individually.

Furthermore, note that the f-variables occur neither in the objective function $NPV(y, z^{\mathrm{p}}, o^{\mathrm{p}}, a^{\mathrm{p}})$ nor in the constraints (1)–(10) and (12)–(17). Hence, these variables can be eliminated from the mixed-integer linear relaxation and from the linear relaxation of (AT) when maximizing $NPV(y, z^{\mathrm{p}}, o^{\mathrm{p}}, a^{\mathrm{p}})$.

As mentioned above, the aggregate-wise mixing constraints, together with the equalities defining the total ore and metal flows through the stockpile, imply the original mixing constraints.

Theorem 1. *Let* $(z^{s,p}, z^{s,s}, o^s, a^s, o^p, a^p, f) \in [0,1]^{2(N\times T)} \times \mathbb{R}^{4T}_{\geq 0} \times [0,1]^T$ *such that* (14)–(18) *hold. Then* (o^s, a^s, o^p, a^p) *satisfies* (11).

Proof. Multiplying each equality (18) (indexed by i and t) with its corresponding ore value O_i and summing up the resulting equalities over all aggregates yields the equalities

$$\sum_{i=1}^{N} O_i z^{\mathrm{s,p}}_{i,t}(1 - f_t) = \sum_{i=1}^{N} O_i z^{\mathrm{s,s}}_{i,t} f_t \qquad \text{for } t \in [T].$$

Analogously one obtains

$$\sum_{i=1}^{N} A_i z^{\mathrm{s,p}}_{i,t}(1 - f_t) = \sum_{i=1}^{N} A_i z^{\mathrm{s,s}}_{i,t} f_t \qquad \text{for } t \in [T].$$

With (14)–(17), these equalities yield

$$\begin{aligned}
o^{\mathrm{p}}_t(1 - f_t) &= o^{\mathrm{s}}_t f_t && \text{for } t \in [T], \text{ and} \\
a^{\mathrm{p}}_t(1 - f_t) &= a^{\mathrm{s}}_t f_t && \text{for } t \in [T],
\end{aligned}$$

which directly imply (11). □

Theorem 1 implies that each feasible solution of (AT) defines a feasible solution of (NF) by projecting out the variables $z^{\mathrm{s,s}}$, $z^{\mathrm{s,p}}$, and f. It is obvious that the same holds for the solutions of the corresponding mixed-integer and linear relaxations. The following theorem shows that each solution of (NF) can in fact be obtained by projecting from a solution of (AT).

Theorem 2. *Let $(x, y, z^p, z^s, o^s, a^s, o^p, a^p) \in S_{NF}$. Then there exist $(z^{s,p}, z^{s,s}, f) \in [0,1]^{2(N\times T)+T}$ such that $(x, y, z^p, z^s, z^{s,p}, z^{s,s}, o^s, a^s, o^p, a^p, f) \in S_{AT}$.*

Proof. Given $(x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}) \in S_{\mathrm{NF}}$, we define

$$f_t := \begin{cases} \frac{o^{\mathrm{p}}_t}{o^{\mathrm{p}}_t + o^{\mathrm{s}}_t} & \text{if } o^{\mathrm{p}}_t + o^{\mathrm{s}}_t > 0 \\ 1 & \text{otherwise} \end{cases} \qquad \text{for } t \in [T],$$

$$z^{\mathrm{s,s}}_{i,t} := \begin{cases} 0 & \text{for } t = 1 \\ (1 - f_t)(z^{\mathrm{s,s}}_{i,t-1} + z^{\mathrm{s}}_{i,t-1}) & \text{for } t \in \{2, \ldots, T\} \end{cases} \quad \text{and}$$

$$z^{\mathrm{s,p}}_{i,t} := \begin{cases} 0 & \text{for } t = 1 \\ f_t(z^{\mathrm{s,s}}_{i,t-1} + z^{\mathrm{s}}_{i,t-1}) & \text{for } t \in \{2, \ldots, T\}. \end{cases}$$

Obviously $(z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, f) \in [0,1]^{2(N\times T)+T}$. Also, one easily verifies that (12)–(18) hold for $(x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, f)$, which implies the claim. □

Denoting by $proj_{NF}(X)$ the projection of a set $X \subseteq S_{\mathrm{AT}}$ to the variable space of (NF), Theorems 1 and 2 imply the following proposition.

Proposition 3. *For any instance of OPMPSP+S we have*

(i) $proj_{NF}(S_{AT}) = S_{NF}$
(ii) $proj_{NF}(S_{AT\text{-}IP}) \subseteq S_{NF\text{-}IP}$
(iii) $proj_{NF}(S_{AT\text{-}LP}) \subseteq S_{NF\text{-}LP}$

In consequence, the mixed-integer linear relaxation and the linear relaxation of the aggregate tracking formulation (AT) are at least as strong as their corresponding counterparts for the natural formulation (NF). Denoting the maximum value of $NPV(y, z^{\mathrm{p}}, o^{\mathrm{p}}, a^{\mathrm{p}})$ over a solution set X by z^*_X, we obtain the following corollary.

Corollary 4. *For any instance of OPMPSP+S we have*

(i) $z^*_{AT} = z^*_{NF}$
(ii) $z^*_{LP-AT} \leqslant z^*_{LP-NF}$
(iii) $z^*_{MIP-AT} \leqslant z^*_{MIP-NF}$

Note that Theorem 2 does *not* hold for the solutions of the mixed-integer linear relaxations or those of the linear relaxations of (NF) and (AT). In fact, it is not difficult to construct instances where both the linear and the mixed-integer linear relaxation of the aggregate tracking formulation (AT) are strictly stronger than their counterparts for (NF). If there are no restrictions on the problem data, one can even construct problem instances where the ratio between the two LP relaxation values or between the two MIP relaxation values is arbitrarily large.

Proposition 5. *There are instances of OPMPSP+S where*

(i) $proj_{NF}(S_{AT\text{-}IP}) \neq S_{NF\text{-}IP}$ *and* $z^*_{MIP-AT} < z^*_{MIP-NF}$
(ii) $proj_{NF}(S_{AT\text{-}LP}) \neq S_{NF\text{-}LP}$ *and* $z^*_{LP-AT} < z^*_{LP-NF}$

Our numerical results in Section 5 show that the difference between the LP relaxation values of the natural formulation and of the aggregate tracking formulation may be very large also for real-world instances.

3.2 The discretised out-fraction formulations

Another technique to tighten either (NF) or (AT) is to introduce a (rough) *a priori* discretisation of the out-fractions f_t and to produce a piece-wise linear outer approximation of the non-linear mixing constraints. For this, we choose a series of fixed ratio levels $0 = \phi_0 < \phi_1 < \ldots < \phi_L < \phi_{L+1} = 1$. Let $[L] := \{1, \ldots, L\}$ and $\Delta_l = \phi_l - \phi_{l-1}$ for $l \in \{1, \ldots, L+1\}$. Using auxiliary binary variables, we then enforce the ratios of material taken from the stockpile and material remaining in the stockpile during a time period to be (at least) in the same interval $[\phi_{i-1}, \phi_i]$, either for all individual aggregates or for the total ore and the total metal amounts. For each time period $t \in [T]$ and each level $l \in [L]$, we introduce the binary variable

$$u_{t,l} \in \{0,1\}, \quad u_{t,l} = \begin{cases} 1 & \text{if } f_t \geqslant \phi_l, \\ 0 & \text{otherwise,} \end{cases}$$

for which we demand

$$u_{t,l} \leqslant u_{t,l-1} \text{ for } l \in \{2, \ldots, L\},\ t \in [T]. \tag{19}$$

We begin by describing how this discretisation can be used to tighten (AT) and later show how the technique applies to (NF).

The connection between the auxiliary variables u and the aggregate-wise material flow variables is established via the inequalities

$$z_{i,t}^{\mathrm{s,p}}(\phi_l - 1) + z_{i,t}^{\mathrm{s,s}}\phi_l \leqslant \sum_{l'=1}^{l} \Delta_{l'}(1 - u_{t,l'}) \qquad \text{for } i \in [N],\ l \in [L],\ t \in [T], \text{ and} \tag{20}$$

$$z_{i,t}^{\mathrm{s,p}}(1 - \phi_l) - z_{i,t}^{\mathrm{s,s}}\phi_l \leqslant \sum_{l'=l}^{L} \Delta_{l'+1} u_{t,l'} \qquad \text{for } i \in [N],\ l \in [L],\ t \in [T]. \tag{21}$$

The following lemma shows that, for any binary vector $u \in \{0,1\}^{T \times L}$, these linear constraints force the out-fraction of each individual aggregate i in time period t to lie in the interval $[\alpha^t, \beta^t] = [\sum_{l=1}^{L} \Delta_l u_{t,l},\ \Delta_1 + \sum_{l=1}^{L} \Delta_{l+1} u_{t,l}]$. We note that these constraints are also valid in the in the converse sense: if the z and f variables are feasible for the (AT) formulation, then binary u can be found satisfying these constraints, where the proof relies on $z_{i,t}^{\mathrm{s,s}} + z_{i,t}^{\mathrm{s,p}} \leqslant 1$ for all i, t.

Lemma 6. *Let $(z^{s,p}, z^{s,s}) \in [0,1]^{2(N \times T)}$ and $u \in \{0,1\}^{T \times L}$ such that (19) holds. Furthermore, let $\alpha^t := \sum_{l=1}^{L} \Delta_l u_{t,l}$ and $\beta^t = \Delta_1 + \sum_{l=1}^{L} \Delta_{l+1} u_{t,l}$. If both (20) and (21) hold, we have*

$$z_{i,t}^{s,p} \geqslant \alpha^t (z_{i,t}^{s,s} + z_{i,t}^{s,p}) \quad \text{and} \tag{22}$$

$$z_{i,t}^{s,p} \leqslant \beta^t (z_{i,t}^{s,s} + z_{i,t}^{s,p}) \quad . \tag{23}$$

In the case of $z_{i,t}^{s,s} + z_{i,t}^{s,p} > 0$, this is equivalent to $f_{i,t} := z_{i,t}^{s,p}/(z_{i,t}^{s,s} + z_{i,t}^{s,p}) \in [\alpha^t,\ \beta^t]$.

Proof. We first consider constraint (22). Let $i \in [N]$, $t \in [T]$ and $l^* \in [L]$ be the maximal l such that $u_{t,l} = 1$. If there is no l with $u_{t,l} = 1$, then (22) reduces to $z_{i,t}^{\mathrm{s,p}} \geqslant 0$, which holds trivially.

Otherwise, constraint (20) for $l = l^*$ is

$$z_{i,t}^{\mathrm{s,p}}(\phi_{l^*} - 1) + z_{i,t}^{\mathrm{s,s}}\phi_{l^*} \leqslant \sum_{l'=1}^{l^*} \Delta_{l'}(1 - u_{t,l'}) = 0.$$

With $\alpha^t = \sum_{l=1}^{L} \Delta_l u_{t,l} = \phi_{l^*}$ this implies

$$z_{i,t}^{\mathrm{s,p}}(\alpha^t - 1) + z_{i,t}^{\mathrm{s,s}}\alpha^t \leqslant 0,$$

which is is equivalent to (22). For each $l < l^*$, constraint (20) is equivalent to

$$z_{i,t}^{\mathrm{s,p}} \geqslant \phi_l(z_{i,t}^{\mathrm{s,s}} + z_{i,t}^{\mathrm{s,p}}),$$

which is implied by (22) and thus redundant. For each $l > l^*$, constraint (20) is equivalent to

$$z_{i,t}^{\mathrm{s,p}} \geqslant \phi_l(z_{i,t}^{\mathrm{s,s}} + z_{i,t}^{\mathrm{s,p}}) - \sum_{l'=l^*+1}^{l} \Delta_{l'}.$$

Since $z_{i,t}^{\mathrm{s,s}} + z_{i,t}^{\mathrm{s,p}} \leqslant 1$, this is implied by

$$\begin{aligned} z_{i,t}^{\mathrm{s,p}} &\geqslant \big(\phi_l - \sum_{l'=l^*+1}^{l} \Delta_{l'}\big)(z_{i,t}^{\mathrm{s,s}} + z_{i,t}^{\mathrm{s,p}}) \\ &= \phi_{l^*}(z_{i,t}^{\mathrm{s,s}} + z_{i,t}^{\mathrm{s,p}}) = \alpha^t(z_{i,t}^{\mathrm{s,s}} + z_{i,t}^{\mathrm{s,p}}), \end{aligned}$$

and thus redundant as well. Hence, inequality (20) is non-redundant only for $l = l^*$, when it reduces to (22).

Analogously, it follows that inequality (21) is non-redundant only for $l = l^* + 1$, when it reduces to (23). □

Note that by forcing, for each $t \in [T]$, the individual out-fractions $f_{i,t}$ of all aggregates to lie in the same interval $[\alpha^t, \beta^t]$, inequalities (20) and (21) implicitly also force the common out-fraction variable f_t in (AT) to lie in this interval.

Making use of Lemma 6 we obtain the *out-fraction discretised aggregate tracking formulation*

$$\begin{aligned} \max\quad & NPV(y, z^{\mathrm{p}}, z^{\mathrm{s,p}}) \\ \text{s.t.}\quad & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, f, u) \text{ (1)–(10), (12)–(18), (19)–(21)}, \\ & x \in \{0,1\}^{N\times T}, \\ & y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}} \in [0,1]^{N\times T}, \\ & f \in [0,1]^T, \\ & u \in \{0,1\}^{T\times L}. \end{aligned} \tag{DAT}$$

As before, we denote the solution sets of (DAT), of its mixed-integer linear relaxation, and of its linear relaxation by S_{DAT}, $S_{\text{DAT-IP}}$, and $S_{\text{DAT-LP}}$, respectively. Since all variables and constraints of (DAT) and its relaxations are contained in the respective discretised out-fraction model and relaxations, we trivially get the following result.

Proposition 7. *For any instance of OPMPSP+S we have*

(i) $\mathit{proj}_{AT}(S_{DAT}) \subseteq S_{AT}$
(ii) $\mathit{proj}_{AT}(S_{DAT\text{-}IP}) \subseteq S_{AT\text{-}IP}$
(iii) $\mathit{proj}_{AT}(S_{DAT\text{-}LP}) \subseteq S_{AT\text{-}LP}$

Again, one easily verifies that the two mixed-integer non-linear models (DAT) and (AT) are equivalent, while the two relaxations of (DAT) may be strictly stronger than their corresponding counterparts for (AT).

Proposition 8. *For any instance of OPMPSP+S we have*

(i) $\mathit{proj}_{AT}(S_{DAT}) = S_{AT}$ *and* $z^*_{DAT} = z^*_{AT}$.

There are instances of OPMPSP+S with

(i) $\mathit{proj}_{AT}(S_{DAT\text{-}IP}) \neq S_{AT\text{-}IP}$ *and* $z^*_{DAT-MIP} < z^*_{AT-MIP}$,

(ii) $\mathit{proj}_{AT}(S_{DAT\text{-}LP}) \neq S_{AT\text{-}LP}$ *and* $z^*_{DAT-LP} < z^*_{AT_LP}$.

Analogous to the out-fractions $f_{i,t}$ of the individual aggregates considered above, we can define the out-fraction of the total ore mass and of the total metal masses as

$$f_{a,t} := \frac{a^{\mathrm{p}}_t}{a^{\mathrm{s}}_t + a^{\mathrm{p}}_t} \quad \text{and} \quad f_{o,t} := \frac{o^{\mathrm{p}}_t}{o^{\mathrm{s}}_t + o^{\mathrm{p}}_t} \qquad \text{for } t \in [T]$$

if $a^{\mathrm{s}}_t + a^{\mathrm{p}}_t$ and $o^{\mathrm{s}}_t + o^{\mathrm{p}}_t$ are strictly positive or, in a more general way, as (not-necessarily unique) values $f_{a,t}, f_{o,t} \in [0,1]$ that satisfy

$$\begin{aligned} a^{\mathrm{p}}_t(1 - f_{a,t}) &= a^{\mathrm{s}}_t f_{a,t} && \text{for } t \in [T], \text{ and} \\ o^{\mathrm{p}}_t(1 - f_{o,t}) &= o^{\mathrm{s}}_t f_{o,t} && \text{for } t \in [T]. \end{aligned}$$

Obviously, the original mixing constraints (11) are satisfied if and only if there exist such values that, in addition, satisfy $f_{o,t} = f_{a,t}$ for all $t \in [T]$.

Let $\bar{A}_t$ and $\bar{O}_t$ be valid upper bounds on the maximum total amounts of metal and ore that may be contained in the stockpile at the beginning of period t, i.e., such that $\bar{A}_t \geqslant a^{\mathrm{p}}_t + a^{\mathrm{s}}_t$ and $\bar{O}_t \geqslant o^{\mathrm{p}}_t + o^{\mathrm{s}}_t$ hold for all valid solutions. Analogous to inequalities (20) and (21), the following inequalities ensure that, for any binary vector $u \in \{0,1\}^{T \times L}$, the values $f_{o,t}$ and $f_{a,t}$ both lie in the interval $[\alpha^t, \beta^t]$:

$$a^{\mathrm{p}}_t(\phi_l - 1) + a^{\mathrm{s}}_t \phi_l \leqslant \sum_{l'=1}^{l} \bar{A}_t \Delta_{l'}(1 - u_{t,l'}) \qquad \text{for } i \in [N],\ l \in [L],\ t \in [T], \tag{24}$$

$$o^{\mathrm{p}}_t(1 - \phi_l) - o^{\mathrm{s}}_t \phi_l \leqslant \sum_{l'=l}^{L} \bar{O}_t \Delta_{l'+1} u_{t,l'} \qquad \text{for } i \in [N],\ l \in [L],\ t \in [T]. \tag{25}$$

Adding these constraints together with the u-variables to the natural formulation (NF), we obtain the *out-fraction discretised natural formulation*:

$$\begin{aligned} \max \quad & NPV(y, z^{\mathrm{p}}, o^{\mathrm{p}}, a^{\mathrm{p}}) \\ \text{s.t.} \quad & (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, u) \text{ satisfies (1) – (11), (19), (24)–(25),} \\ & \qquad x \in \{0,1\}^{N \times T}, \\ & \qquad y, z^{\mathrm{p}}, z^{\mathrm{s}} \in [0,1]^{N \times T}, \\ & o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}} \geqslant 0, \\ & \qquad f \in [0,1]^T, \\ & \qquad u \in \{0,1\}^{T \times L}. \end{aligned} \tag{DNF}$$

Again, we denote the solution sets of (DNF), of its mixed-integer linear relaxation, and of its linear relaxation by S_{DNF}, $S_{\text{DNF-IP}}$, and $S_{\text{DNF-LP}}$, respectively. Analogously to Propositions 7 and 8, we have the following relations between the solution sets and optimal solution values.

Proposition 9. *For any instance of OPMPSP+S we have*

(i) $proj_{NF}(S_{DNF}) = S_{NF}$ *and* $z^*_{DNF} = z^*_{NF}$,
(ii) $proj_{NF}(S_{DNF\text{-}IP}) \subseteq S_{NF\text{-}IP}$ *and* $z^*_{DNF-MIP} \leqslant z^*_{NF-MIP}$,
(iii) $proj_{NF}(S_{DNF\text{-}LP}) \subseteq S_{NF\text{-}LP}$ *and* $z^*_{DNF-LP} \leqslant z^*_{NF-LP}$.

There are instances of OPMPSP+S with

(iv) $proj_{AT}(S_{DNF\text{-}IP}) \neq S_{AT\text{-}IP}$ *and* $z^*_{DNF-MIP} < z^*_{NF-MIP}$,
(iiv) $proj_{AT}(S_{DNF\text{-}LP}) \neq S_{AT\text{-}LP}.$ *and* $z^*_{DNF-LP} < z^*_{NF-LP}$.

Analogous to the proofs in Section 3.1 one can also show that, for the same discretisation ϕ, the mixed-integer linear relaxation and linear relaxations of (DAT) are stronger than those of (DNF).

Proposition 10. *For any instance of OPMPSP+S we have*

(i) $proj_{DNF}(S_{DAT}) = S_{DNF}$ *and* $z^*_{DAT} = z^*_{DNF}$,
(ii) $proj_{DNF}(S_{DAT\text{-}IP}) \subseteq S_{DNF\text{-}IP}$ *and* $z^*_{DAT-MIP} \leqslant z^*_{DNF-MIP}$,
(iii) $proj_{DNF}(S_{DAT\text{-}LP}) \subseteq S_{DNF\text{-}LP}$ *and* $z^*_{DAT-LP} \leqslant z^*_{DNF-LP}$.

There are instances of OPMPSP+S with

(iv) $proj_{DNF}(S_{DAT\text{-}IP}) \neq S_{DNF\text{-}IP}$ *and* $z^*_{DAT-MIP} \leqslant z^*_{DNF-MIP}$,
(iiv) $proj_{DNF}(S_{DAT\text{-}LP}) \neq S_{DNF\text{-}LP}$ *and* $z^*_{DAT-LP} \leqslant z^*_{DNF-LP}$.

This shows that the individual aggregate tracking and the out-fraction discretisation both independently strengthen the relaxations of our formulation.

4 Solution approach

4.1 Overview

In this section, we describe the approach we developed and implemented to solve OPMPSP+S.

Our approach is based on a branch-and-bound algorithm solving the mixed-integer linear programming relaxation of one of the models presented in the previous section. In addition to the standard mixed-integer branching schemes, which enforce the integrality of all integer variables in a solution, we also employ a specialised spatial branching scheme to reduce the maximum violation of the relaxed nonlinear mixing-constraints. Using this branching scheme, we can force maximum violation of the nonlinear constraints arbitrarily close to zero in the created subproblems. Thus, our algorithm can compute upper bounds that are arbitrarily close to the global optimum solution value of the full mixed-integer nonlinear problem formulation.

Furthermore, we use a specialized primal heuristic to turn solution candidates that are integer feasible but still violate the mixing constraints into solutions that also satisfy these nonlinear constraints and, thus, are feasible for the respective full mixed-integer nonlinear formulation. As the deterioration of the objective function value is linearly bounded by the violation of nonlinear constraints by the given candidate solution, this allows us to also compute fully feasible solutions arbitrary close to a globally optimal solution of the full mixed-integer nonlinear problem formulation.

As our branching scheme only requires the addition of linear constraints in the definition of the subproblems, only linear relaxations need to be solved during the branch-and-bound process, which leads to a practically efficient algorithm.

In the following, we describe the branching scheme and the primal heuristic in more detail.

4.2 Branching

First, we consider the case in which we wish to solve the basic formulation (NF) using the (mixed-integer) linear relaxation R obtained by dropping constraint (11).

Suppose we are given a (fractional) solution $(x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}) \in S_{\text{NF-LP}}$ that violates constraint (11) for some time period $t \in [T]$. Then the metal fraction taken out of the stockpile differs from the ore fraction taken out. Suppose that the metal fraction exceeds the ore fraction, which will be the case for the earliest period where the fractions differ if the given solution is an optimal LP solution and all revenue and cost values are positive. Then there is a some ratio ϕ with

$$\frac{o_t^{\mathrm{p}}}{o_t^{\mathrm{p}} + o_t^{\mathrm{s}}} < \phi < \frac{a_t^{\mathrm{p}}}{a_t^{\mathrm{p}} + a_t^{\mathrm{s}}}.$$

Using this ratio ϕ as threshold, two branches are created: one where both ratios are at least ϕ, i.e.

$$\text{both} \quad (1-\phi)o_t^{\mathrm{p}} \leqslant \phi o_t^{\mathrm{s}} \quad \text{and} \quad (1-\phi)a_t^{\mathrm{p}} \leqslant \phi a_t^{\mathrm{s}}, \tag{26}$$

and one where neither ratio exceeds ϕ, i.e.

$$\text{both} \quad (1-\phi)o_t^{\mathrm{p}} \geqslant \phi o_{t-1}^{\mathrm{s}} \quad \text{and} \quad (1-\phi)a_t^{\mathrm{p}} \geqslant \phi a_{t-1}^{\mathrm{s}} \tag{27}$$

hold. Clearly, the given solution violates both sets of constraints and thus is cut off in both branches. On the other hand, any solution satisfying (11) satisfies at least one of the two inequality sets (26) or (27). Adding the linear constraints (26) or (27) to the current (mixed-integer) linear relaxation $R \subseteq S_{\text{NF-LP}}$ thus yields two (mixed-integer) linear subproblems R_1 and R_2 of R such that each solution of R that satisfies (11) is also solution of R_1 or R_2. Furthermore, denote by $\varDelta_R := |\frac{o_t^{\mathrm{p}}}{o_t^{\mathrm{p}}+o_t^{\mathrm{s}}} - \frac{a_t^{\mathrm{p}}}{a_t^{\mathrm{p}}+a_t^{\mathrm{s}}}| \leqslant 1$ the maximum difference between the metal fraction and the ore fraction taken out of the stockpile that may be attained by a solution of relaxation R. With ϕ chosen strictly between $\frac{o_t^{\mathrm{p}}}{o_t^{\mathrm{p}}+o_t^{\mathrm{s}}}$ and $\frac{a_t^{\mathrm{p}}}{a_t^{\mathrm{p}}+a_t^{\mathrm{s}}}$, we immediately get $\varDelta_{R_1} < \varDelta_R$ and $\varDelta_{R_2} < \varDelta_R$, that is, the maximum violation of constraint (11) in the newly created subproblems strictly reduces. Choosing $\phi := \frac{o_t^{\mathrm{p}}}{2(o_t^{\mathrm{p}}+o_t^{\mathrm{s}})} + \frac{a_t^{\mathrm{p}}}{2(a_t^{\mathrm{p}}+a_t^{\mathrm{s}})}$, the actual violation of (11) will be forced to reduce by a constant factor with each application of this branching scheme for period t. Applying this branching scheme repeatedly whenever some constraint (11) is violated, we can thus force the violation of all constraints (11) to become arbitrarily close to 0.

Similarly, for the aggregate tracking formulation (AT), suppose we are given an (already partially tightened) (mixed-integer) linear relaxation $R \subseteq S_{\text{AT-LP}}$ of (AT) and some (fractional) solution $(x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, f) \in R$ that violates constraint (18) for some time period $t \in [T]$.

Then there exist at least two aggregates $i, j \in [N]$ with different out-fractions, thus there is a ratio ϕ with

$$\frac{z^{\mathrm{s,p}}_{i,t}}{z^{\mathrm{s,s}}_{i,t} + z^{\mathrm{s,p}}_{i,t}} < \phi < \frac{z^{\mathrm{s,p}}_{j,t}}{z^{\mathrm{s,s}}_{j,t} + z^{\mathrm{s,p}}_{j,t}}.$$

This again gives rise to two branches, one with the additional constraints

$$(1-\phi) z^{\mathrm{s,p}}_{i,t} \leqslant \phi z^{\mathrm{s,s}}_{i,t} \qquad \text{for } i \in [N], \tag{28}$$

forcing the out-fractions of *all* aggregates in time period t to be no more than ϕ, and the other branch with constraints

$$(1-\phi) z^{\mathrm{s,p}}_{i,t} \geqslant \phi z^{\mathrm{s,s}}_{i,t} \qquad \text{for } i \in [N], \tag{29}$$

forcing the out-fractions of *all* aggregates in time period t to be no less than ϕ.

Again, the current solution is cut off in both branches, while any solution satisfying (18) satisfies at least one of the two inequality sets (28) or (29). Also, the maximum difference $\varDelta_R := \max\{\frac{z^{\mathrm{s,p}}_{i,t}}{z^{\mathrm{s,s}}_{i,t}+z^{\mathrm{s,p}}_{i,t}} : i \in [N]\} - \min\{\frac{z^{\mathrm{s,p}}_{i,t}}{z^{\mathrm{s,s}}_{i,t}+z^{\mathrm{s,p}}_{i,t}} : i \in [N]\}$ between the aggregate out-fractions will be strictly reduced in the two newly created subproblems. Choosing ϕ as the mean of the minimum and the maximum aggregate out-fraction, the violation of constraint (18) for time period t can be forced to become arbitrarily small by applying this second branching scheme repeatedly whenever (18) is violated. In practice, however, choosing ϕ equal to the out-fraction of the total ore amount proved to be more efficient. One easily verifies that the out-fraction of the total ore tonnage is always between the minimum and the maximum aggregate out-fractions, and strictly between those values if the aggregates attaining these values have different grades (metal to ore ratios).

Note that, in both branching schemes, the branches are created by adding *multiple* linear inequalities. Such an aggressive branching strategy is usually invalid for general MINLP solvers, since it cuts off feasible solutions. In our special application it is feasible because of our additional knowledge, that the out-fractions of each attribute and of each individual aggregate must be equal. Also note that the branches are created by adding *linear* inequalities to the current branch-and-bound node's relaxation. Thus, only linear programs need to be solved during the branch-and-bound algorithm, and efficient warm-starts using the dual simplex algorithm are possible.

In order to enforce integrality for all integer variables and arbitrarily small violation of the nonlinear mixing constraints, we combine the branching schemes described above with the standard branching on fractional variables that must be integer. For this, we need to decide at each node of the branch-and-bound tree what type of branching to apply.

Our goal is to obtain a small branch-and-bound tree. In order to select a good branching, we evaluate a number of potential branchings on violated mixing constraints and compare the predicted effect these branches would have on the value of the relaxation to the predicted effect of the branching on the fractional variable proposed by the built-in variable brancher of the branch-and-bound framework. For the optimal solution of the current node's relaxation and for each time period $t \in [T]$ where the corresponding mixing constraints (11) or (18) are violated, we consider the branching defined by the threshold $\phi_t = \frac{o^{\mathrm{p}}_t}{2(o^{\mathrm{p}}_t+o^{\mathrm{s}}_t)} + \frac{a^{\mathrm{p}}_t}{2(a^{\mathrm{p}}_t+a^{\mathrm{s}}_t)}$ when solving (NF) or (DNF) or by the threshold $\phi_t = \frac{o^{\mathrm{p}}_t}{o^{\mathrm{p}}_t+o^{\mathrm{s}}_t}$ when solving (AT) or (DAT).

For each of these branchings, we estimate the effect of the branching as the difference of the objective values of the current relaxation's solution and the solution obtained as follows. For period t, we set the out-fraction of the total metal amount to ϕ_t and adjust the corresponding amounts of metal remaining in the stockpile during period t. For all later periods, we successively set the out-fraction of metal equal to that of ore. This procedure shifts the metal that was 'illegally' taken from the stockpile in period t to the next period and, for all following time periods, adjusts the metal flows through the stockpile to the ore flows. The deterioration observed in the objective function when doing these adjustments proved to be good measure for the effect of the proposed branching. (Note that our estimate implicitly depends on the threshold value chosen in the branching. For different threshold choices, different estimates may be more appropriate.) Among all possible branchings on violated mixing constraints, we pre-select the one where this estimate is maximized.

Finally, the estimate of the pre-selected branching is compared to the estimation of the effect of the proposed non-integer variable branch, which is given via the pseudo-cost associated with that variable. If the estimate of the pre-selected branching on a mixing constraint is greater than the minimum predicted deterioration in the two subproblems proposed by the variable branching, then we perform the branching on the mixing constraint. Otherwise, we perform the proposed non-integer variable branching. In order to speed up the finding of good feasible solution, we also perform the proposed branching on the non-integer variable whenever the pseudo-cost prediction tells us that

(i) both subproblems proposed by the variable branching close more than 30% of the current node's optimality gap,
(ii) one of them closes more than 90% of this gap,
(iii) the number of integer-infeasible variables is below 10, or
(iv) the difference between the two subproblems suggests that the branch-and-bound algorithm is diving for a feasible solution.

4.3 Primal heuristic

As discussed above, our specialized branching schemes can be used to force the violation of the nonlinear mixing constraint to become arbitrarily small. However, the optimal solutions of the branch-and-bound nodes' linear relaxations will never perfectly satisfy the mixing constraints but violate them within the local limits, unless the grades of all aggregates put in the stockpile are equal. In contrast, the branching on integer variables, while we know beforehand that their number is finite and, thus, finitely many branches suffice to ensure integer feasibility, our branching scheme for the mixing constraints cannot ensure that these constraints will be satisfied after finitely many branchings. With finitely many branchings, our branch-and-bound algorithm thus can ensure integer feasibility, but not fully feasibility with respect to the nonlinear constraints.

A candidate solution that is integer feasible but violating the mixing constraint, however, can be easily turned into a fully feasible solution by modifying the material flows through the stockpile.

Algorithm Rebalance shown in Figure 2 illustrates how this can be done for the aggregate tracking formulation (AT). For each period t from 2 to T, it iteratively sets the out-fractions of the total metal amount and the out-fractions of all individual aggregates to the out-fraction of the total ore amount for that period and updates the respective amounts remaining in the stockpile for the next period.

Note that Algorithm Rebalance only modifies the total metal amount and the aggregate fractions reclaimed from and remaining in the stockpile. The aggregate fractions put in the stockpile

```
   REBALANCE
   Input: $v = (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, f) \in S_{\text{AT-LP}}$
   Output: $\tilde{v} = (x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, z^{\mathrm{s,p}}, z^{\mathrm{s,s}}, o^{\mathrm{s}}, a^{\mathrm{s}}, o^{\mathrm{p}}, a^{\mathrm{p}}, f)$ at the end of the algorithm
 1 for $t := 2$ to $T$ do
 2   $S := o^{\mathrm{s}}_{t-1} + \sum_{i\in[N]} O_i z^{\mathrm{s}}_{i,t-1}$
 3   $O := o^{\mathrm{p}}_t$
 4   if $S > 0$ then $f_t := O/S$, else $f_t := 1$
 5   $a^{\mathrm{p}}_t := f_t\big(a^{\mathrm{s}}_{t-1} + \sum_{i\in[N]} A_i z^{\mathrm{s}}_{i,t-1}\big)$
 6   $a^{\mathrm{s}}_t := (1 - f_t)\big(a^{\mathrm{s}}_{t-1} + \sum_{i\in[N]} A_i z^{\mathrm{s}}_{i,t-1}\big)$
 7   for $i := 1$ to $N$ do
 8     $z^{\mathrm{s,p}}_{i,t} := f_t(z^{\mathrm{s,s}}_{i,t-1} + z^{\mathrm{s}}_{i,t-1})$
 9     $z^{\mathrm{s,s}}_{i,t} := (1 - f_t)(z^{\mathrm{s,s}}_{i,t-1} + z^{\mathrm{s}}_{i,t-1})$
10   end-do
11 end-do
```

Fig. 2. Rebalancing heuristics (for the aggregate tracking formulation)

as well as the total ore amounts reclaimed from and remaining in the stockpile remain unchanged. Hence, the modified solution will still satisfy the processing capacity constraints. More precisely, we have the following theorem.

Theorem 11. *Let* $v := (x, y, z^p, z^s, z^{s,p}, z^{s,s}, o^s, a^s, o^p, a^p, f) \in S_{\mathit{AT\text{-}LP}}$ *and denote by* $\tilde{v} := (\tilde{x}, \tilde{y}, \tilde{z^p}, \tilde{z^s}, \tilde{z^{s,p}}, \tilde{z^{s,s}}, \tilde{o^s}, \tilde{a^s}, \tilde{o^p}, \tilde{a^p}, \tilde{f})$ *be the solution constructed by algorithm* REBALANCE *from* v*. Then* $\tilde{v}$ *satisfies* (1)–(10)*,* (12)–(17) *and* (18) *(as well as* (11)*).*

Proof. Since algorithm REBALANCE only modifies the variables f, a^{p}, a^{s}, $z^{\mathrm{s,p}}$, and $z^{\mathrm{s,s}}$, we have $(x, y, z^{\mathrm{p}}, z^{\mathrm{s}}, o^{\mathrm{s}}, o^{\mathrm{p}}) = (\tilde{x}, \tilde{y}, \tilde{z^{\mathrm{p}}}, \tilde{z^{\mathrm{s}}}, \tilde{o^{\mathrm{s}}}, \tilde{o^{\mathrm{p}}})$. With $v \in S_{\text{AT-LP}}$, it follows immediately that $\tilde{v}$ satisfies (1)–(6) and (8)–(10). Furthermore, the variables f, a^{p}, a^{s}_t, $z^{\mathrm{s,p}}$, and $z^{\mathrm{s,s}}$ are modified only for time periods $t \geqslant 2$, we also have $(f_1, a^{\mathrm{p}}_1, a^{\mathrm{s}}_1, z^{\mathrm{s,p}}_1, z^{\mathrm{s,s}}_1) = (\tilde{f_1}, \tilde{a^{\mathrm{p}}_1}, \tilde{a^{\mathrm{s}}_1}, \tilde{z^{\mathrm{s,p}}_1}, \tilde{z^{\mathrm{s,s}}_1})$ which implies that (8) and (13) hold for $\tilde{v}$.

Step 4 of the algorithm REBALANCE sets the out-fraction variable f_t for each period to the total ore out-fraction (or to 1, if the stockpile is empty at the beginning of period t). Steps 5 and 6 then ensure that the out-fraction of the total metal amount is equal to f_t for each t, which implies that the total metal flow conservation constraints (7) and the total mixing constraints (11) are satisfied by $\tilde{v}$. Analogously, Steps 7 to 10 ensure that the out-fraction of each individual aggregate is equal to f_t for each period t, which implies that (12) and (18) hold for $\tilde{v}$. Together with the total and aggregate-wise material conservation constraints (6), (7), and (12), the equalities defining $z^{\mathrm{s,p}}_{i,t}$ and $z^{\mathrm{s,s}}_{i,t}$ in Steps 8 and 9 of the algorithm finally imply that also the equalities (14)–(17) linking the total and aggregate-wise material flows through the stockpile are satisfied by $\tilde{v}$. □

Corollary 12. *Applying algorithm* REBALANCE *to an integer-feasible solution* v *of the mixed-integer linear relaxation of* (AT) *yields a fully feasible solution* $\tilde{v}$ *of* (AT)*.*

Omitting steps 7 to 10 from algorithm REBALANCE, one obtains an analogous algorithm to turn solutions of the (mixed-integer) linear relaxation of the natural formulation (NF) into solution candidates that, in addition to the constraints of the relaxation, also satisfy the nonlinear mixing constraints (11).

As algorithm REBALANCE modifies the amount of metal that is contained in the material reclaimed from the stockpile in each time period, the constructed solution will have a different objective value than the candidate solution passed to the algorithm as input. The following theorem provides a bound for the possible deterioration of the objective value.

Theorem 13. *Let* $\tilde{v} := (\tilde{x}, \tilde{y}, \tilde{z^p}, \tilde{z^s}, z^{\tilde{s},p}, z^{\tilde{s},s}, \tilde{o^s}, \tilde{a^s}, \tilde{o^p}, \tilde{a^p}, \tilde{f})$ *be the solution constructed by algorithm* Rebalance *from* $v := (x, y, z^p, z^s, z^{s,p}, z^{s,s}, o^s, a^s, o^p, a^p, f) \in S_{AT\text{-}LP}$. *Furthermore, let* $\varDelta_t :=$ $\max\left\{\frac{z^{s,p}_{i,t}}{z^{s,s}_{i,t}+z^{s,p}_{i,t}} : i \in [N],\ z^{s,s}_{i,t} + z^{s,p}_{i,t} > 0\right\} - \min\left\{\frac{z^{s,p}_{i,t}}{z^{s,s}_{i,t}+z^{s,p}_{i,t}} : i \in [N],\ z^{s,s}_{i,t} + z^{s,p}_{i,t} > 0\right\}$ *be the difference between the maximum out-fraction and the minimum out-fraction over all aggregates in time period* t *or* $\varDelta_t := 0$ *if* $z^{s,s}_{i,t} + z^{s,p}_{i,t} = 0$ *for all* $i \in [N]$ *in period* t *in the given solution* v*. Then*

$$|NPV(y, z^p, o^p, a^p) - NPV(\tilde{y}, \tilde{z^p}, \tilde{o^p}, \tilde{a^p})| \leqslant c \sum_{t=2}^{T} \sum_{i=1}^{N} \big(\delta_t A_i \sum_{t'=2}^{t} \varDelta_{t'}\big) \ .$$

Proof. Clearly, the out-fraction of the total ore amount $f_t = \frac{o^{\mathrm{p}}_t}{o^{\mathrm{p}}_t + o^{\mathrm{s}}_t}$ is an average of the individual aggregate out-fractions in each time period t in the given solution v. Each aggregate's out-fraction thus is changed by at most $\varDelta_t$ in period t by algorithm Rebalance.

Adjusting the out-fractions iteratively for all periods 2 to $t-1$, the algorithm changes the total fraction of aggregate i in the stockpile at the beginning of period t by at most $\sum_{t'=1}^{t-1} \varDelta_{t'}$. Together with the out-fraction change of at most $\varDelta_t$ in period t, this yields

$$z^{\tilde{\mathrm{s}},\mathrm{p}}_{i,t} - z^{\mathrm{s},\mathrm{p}}_{i,t} \leqslant \sum_{t'=2}^{t} \varDelta_{t'} \quad \text{for all } i \in [N],\ t \in [T].$$

With (17), this implies that the objective function term $\sum_{t=1}^{T} \delta_t c a^{\mathrm{p}}_t$ for the revenues of processing the material reclaimed from the stockpile changes by at most $\sum_{t=2}^{T} \sum_{i=1}^{N} \big(\delta_t c A_i \sum_{t'=2}^{t} \varDelta_{t'}\big)$. As all other variables involved in the objective function are not modified by Rebalance, the claim follows. □

Analogously, one can show that the deterioration of the solution obtained by applying Rebalance with Steps 7 to 10 to a solution of the (mixed-integer) linear relaxation of (NF) is linearly bounded in the differences of the out-fractions of the total metal amount and the total ore amount.

Note that the bound given in the theorem is very weak and can be strengthened easily. The key statement of Theorem 13 is that the potential deterioration of the objective function value when applying Rebalance is bounded *linearly* in the violation of the mixing constraints. Applying Rebalance to a solution v of the (integer-linear) relaxation of (AT) (or (NF)) that is integer-feasible, we obtain a solution $\tilde{v}$ that is fully feasible for (AT) (or (NF)) and whose global optimality gap is linearly bounded in the optimality gap and the violation of the mixing constraints by the given solution. Combined with the branching scheme presented in the previous section, this allows us to compute fully feasible solutions that are arbitrarily close to the global optimal solution of OPMPSP+S.

We want to emphasis that algorithm Rebalance uses two important properties of the problem OPMPSP+S: (i) the time-expanded material flow network, in which the mixing constraints have to be enforced, is acyclic (c.f. Figure 1) and (ii) only one of the attributes associated with the material flow that are modified occurs in other side constraints. In our case, this attribute is the ore amount, and the only other constraints where it is involved are the processing capacity constraints. If we have these two properties, we can enforce the nonlinear constraints by adjusting the material flows iteratively in all nodes of the material flow network according to the single critical attribute without creating violations of any other side constraint.

Finally, we wish to remark that a solution obtained by applying Rebalance to an integer-feasible solution v of the mixed-integer linear relaxation of (AT) is optimal only among those fully feasible solutions, that differ from v only in the variables f, a^{p}, a^{s}, $z^{\mathrm{s,p}}$, $z^{\mathrm{s,s}}$. However, it is not necessarily optimal among those solutions that only coincide with v in the integer variables. In fact, if the given solution v violates the mixing constraints by a large amount, the constructed solution often can be improved by simple local exchanges. In the solution constructed by Rebalance it may, for example, happen that some of the material that is sent newly into the stockpile in period t has a better metal-to-ore grade than the material that is reclaimed from the stockpile and sent to processing during that period, because the real metal-to-ore grade of the reclaimed material was overestimated in the solution of the relaxation. In this case, one can send some of the better grade material directly to processing and instead retain more of the lower grade material in the stockpile, thereby increasing the objective value. Of course, the material flows in the stockpile need to be adjusted for the following periods to fulfil the material conservation constraints.

In our solution algorithm, we iteratively apply similar local improvements to handle the following types of non-optimal solutions:

(i) material sent to stockpile in period t has a higher grade than material reclaimed from stockpile in period t,
(ii) material sent to stockpile in period t has a higher grade than material sent directly to processing in period t,
(iii) material remaining in stockpile during period t has a higher grade than material sent directly to processing in period t, and
(iv) material remaining in stockpile in period t has a lower grade than material sent to waste in period t.

At the beginning of the branch-and-bound algorithm, where the violation of the nonlinear mixing constraints by the integer-feasible solutions that are given as input to the Rebalance algorithm is large, the effect of these local improvements is quite big. Later in the algorithm, when the relaxations at the branch-and-bound nodes only permit a relatively small violation of the mixing constraints, these local improvements rarely apply and their effect is negligible.

4.4 Cutting planes and variable fixing

In order to improve the practical performance of our branch-and-bound algorithm, we strengthen the linear relaxations of (NF), (AT), (DAT), or (DNF) by adding further valid inequalities.

As already observed in Section 3, one major drawback of the natural formulation is that it only tracks the total ore and the total metal amounts that are taken from the stockpile or remain in the stockpile. A solution of the (mixed-integer) linear relaxation, where the mixing constraints (11) linking these values are relaxed, in principle may reclaim pure metal from the stockpile, while leaving all ore in the stockpile. In such a solution, we not only violate the mixing constraints, but net revenue is obtained “for free” without using the processing capacity just by stockpiling the material in one period and extracting the pure metal in the next period. Of course, this is not possible in a solution that respects the mixing constraints. Clearly, the metal-to-ore grade of the material reclaimed from the stockpile cannot be better than that of the best aggregate put into the stockpile before or, more generally, of the best aggregate in the mine.

Let $g^{\max} := \max\{A_i/O_i \ : \ i \in [N],\ O_i > 0\}$ and $g^{\min} := \min\{A_i/O_i \ : \ i \in [N],\ O_i > 0\}$ be the maximum and the minimum grade of the aggregates in the mine. The above observation then immediately implies the following proposition.

Proposition 14. *Each solution of* (NF) *(or of* (DNF)*) satisfies the trivial grade bound inequalities*

$$\begin{aligned} a_t^p &\leqslant g^{\max} o_t^p && \textit{for } t \in [T], && (30)\\ a_t^p &\geqslant g^{\min} o_t^p && \textit{for } t \in [T], && (31)\\ a_t^s &\leqslant g^{\max} o_t^s && \textit{for } t \in [T], \textit{ and} && (32)\\ a_t^s &\geqslant g^{\min} o_t^s && \textit{for } t \in [T]. && (33) \end{aligned}$$

Adding these simple inequalities to the model leads to significantly better relaxation values for (NF) and for (DNF). As there are only $4T$ inequalities (30)–(33), we add all of them to the initial formulation when using (NF) or (DNF) in our branch-and-bound algorithm. For the aggregate tracking formulations (AT) and (DAT), inequalities (30)–(33) are also valid. However, they will never be violated by a fractional solution of the linear relaxation, as they are trivially implied by the model constraints of those formulations.

Now consider the aggregate tracking formulation (AT). The (mixed-integer) linear relaxation of this formulation does not permit the processing of material for free by sending it through the stockpile, but it permits individual reclamation of the different aggregates that have been put into the stockpile, as if they had been stored in individual stockpiles. Thus, a solution of the relaxation may reclaim all stockpiled material of aggregate i in period t, while leaving all stockpiled material of aggregate $j \neq i$ in the stockpile during this period. In a solution satisfying the mixing constraints, this is impossible of course. As (AT) tracks for each individual aggregate what fraction of this aggregate is reclaimed from the stockpile and what fraction is kept in the stockpile, we can at least forbid the extreme cases, where large fractions of two blocks are in the stockpile at the beginning of some period and one is reclaimed while the other is kept in the stockpile. The following theorem makes this more precise.

Theorem 15. *Each solution of* (AT) *(or* (DAT)*) satisfies the inequalities*

$$z_{i,t}^{s,p} + z_{j,t}^{s,s} \leqslant 1 \qquad \textit{for } i,j \in [N], t = 1,\dots,T. \qquad (34)$$

Proof. Consider a solution $(y, z^{\mathrm{p}}, z^{\mathrm{s,p}})$ of (AT) and two aggregates $i, j \in [N]$, and a time period t. The material conservation constraints (3), (5), (12) imply that $z_{i,t}^{\mathrm{s,p}} + z_{i,t}^{\mathrm{s,s}} \leqslant 1$ and $z_{j,t}^{\mathrm{s,p}} + z_{j,t}^{\mathrm{s,s}} \leqslant 1$. As $(y, z^{\mathrm{p}}, z^{\mathrm{s,p}})$ also satisfies the mixing constraint (18), we furthermore have $z_{i,t}^{\mathrm{s,p}} = f_t(z_{i,t}^{\mathrm{s,p}} + z_{i,t}^{\mathrm{s,s}}) \leqslant f_t$ and $z_{j,t}^{\mathrm{s,s}} = (1 - f_t)(z_{j,t}^{\mathrm{s,p}} + z_{j,t}^{\mathrm{s,s}}) \leqslant (1 - f_t)$, which immediately yields (34). □

Note that, again, there is only a polynomial number of inequalities (34). So, in principle, we could add them all to the initial formulation. In practice, however, adding all $N(N-1)T$ inequalities to the formulation that is actually solved is computationally prohibitive. Instead, we generate all inequalities and place them in a cut pool, from where we iteratively add only those that are violated to the relaxation that is actually solved.

Also note that in the case where $c > 0$ and the NPV discount factor decreases with time, i.e. $\delta_{t+1} \leqslant \delta_t$ for all t, none of the inequalities (34) with i, j such that the grade i is less than the grade of j, i.e. $A_i/O_i < A_j/O_j$, will be violated by an optimal solution of the linear relaxation. To see why, suppose otherwise, i.e. suppose that $A_i/O_i < A_j/O_j$ but $z^{\text{s,p}}_{i,t} + z^{\text{s,s}}_{j,t} > 1$ for some t where $z^{\text{s,p}}, z^{\text{s,s}}$ come from the an optimal solution over $S_{\text{AT-LP}}$. Note that in this case it must be that $z^{\text{s,p}}_{i,t}, z^{\text{s,s}}_{j,t} > 0$, so $z^{\text{s,s}}_{i,t}, z^{\text{s,p}}_{j,t} < 1$. Then we can construct a new feasible solution by "swapping" some portion of aggregate i sent for processing in period t with a portion of aggregate j, and restoring the material swapped in a later period. Mathematically this can be accomplished with the following steps: (1) find t^+, the first period after t in which some of aggregate j is sent for processing, i.e. with $z^{\text{s,p}}_{j,t^+} > 0$ and set $u := \min\{z^{\text{s,p}}_{j,t^+}, \frac{O_i}{O_j}(1 - z^{\text{s,p}}_{i,t^+})\} > 0$ where $z^{\text{s,p}}_{i,t^+} < 1$ follows from $z^{\text{s,p}}_{i,t} > 0$, taking $u := 1$ if there is no such period; (2) set $\epsilon := \min\{1 - z^{\text{s,p}}_{j,t}, \frac{O_i}{O_j}(1 - z^{\text{s,s}}_{i,t}), z^{\text{s,s}}_{j,t}, \frac{O_i}{O_j} z^{\text{s,p}}_{i,t}, u\} > 0$; (3) replace $z^{\text{s,p}}_{j,t}$ with $z^{\text{s,p}}_{j,t} + \epsilon$, adjusting $z^{\text{s,s}}_{j,t'}$ to maintain the material balance by subtracting ϵ for all $t' = t, \ldots, t^+ - 1$, or $t' = t, \ldots, T$ if $u = 1$, (this must be possible since $z^{\text{s,p}}_{j,t'} = 0$ so $z^{\text{s,s}}_{j,t'} \geqslant z^{\text{s,s}}_{j,t}$ for all such t'), and if $u < 1$, replacing $z^{\text{s,p}}_{j,t^+}$ with $z^{\text{s,p}}_{j,t^+} - \epsilon$; and (4) replace $z^{\text{s,p}}_{i,t}$ with $z^{\text{s,p}}_{i,t} - \epsilon \frac{O_j}{O_i}$, adjusting $z^{\text{s,s}}_{i,t'}$ to maintain the material balance by adding $\epsilon \frac{O_j}{O_i}$ for all $t' = t, \ldots, t^+ - 1$, or $t' = t, \ldots, T$ if $u = 1$, and if $u < 1$, replacing $z^{\text{s,p}}_{i,t^+}$ with $z^{\text{s,p}}_{i,t^+} + \epsilon \frac{O_j}{O_i}$. These adjustments keep the quantity of ore sent to processing constant in every period, but send more of the higher-grade ore (from aggregate j) to processing in period t than lower-grade ore (from aggregate i), with the reverse occurring in period t^+ if $u < 1$: the quantity of metal sent to processing in the new solution in period t is increased by $\gamma := \epsilon(A_j - A_i \frac{O_j}{O_i}) > 0$ over the metal processed in the original solution, and decreased in period t^+ by the same amount (or not at all if $u = 1$). However net present value costs favour earlier processing of higher grade material, and so the adjusted solution will have a better objective value: the objective value of the new solution minus the old is $c\gamma(\delta_t - \delta_{t+})$ (or $c\gamma\delta_t$ if $u = 1$), which is positive, contradicting the optimality of the original relaxation solution. Thus the objective drives satisfaction of the constraints when the grade of i is lower than that of j, and we add these inequalities only for those $i, j \in [N]$ with $A_i/O_i \geqslant A_j/O_j$ in realistic problem instances.

The two classes of inequalities described so far are concerned only with violations of the mixing constraints, but not with violations of the integrality constraints. They cut off both fractional and integer solutions that violate the mixing constraints and, thus, actually strengthen the mixed-integer linear relaxations of (NF) and (AT) (or of (DNF) and (DAT)). To further strengthen the linear relaxations of these formulations, we finally apply the variable fixing and cutting plane techniques introduced and analyzed in [3,4]. These techniques exploit the integrality of the variables x that describe the order in which the aggregates are mined.

For each period t, the corresponding mining capacity constraint (9) together with the precedence constraints (4) and the constraints (1)–(3) and (5) defines a so-called precedence constrained knapsack: The total amount of rock of all aggregates that have been completely mined until the end of period t must not exceed the total mining capacities over periods 1 to t and, furthermore, these aggregates must satisfy the precedence constraints implied by the wall slope (or other safety) restrictions. For each aggregate $i \in [N]$, let $\mathcal{P}^{cl}(i) \subseteq [N]$ be the set of all aggregates (including i itself) that directly or indirectly must precede aggregate i in the mining order, i.e., $\mathcal{P}^{cl}(i)$ is the transitively closed set of predecessors of i. Clearly, all aggregates in $\mathcal{P}^{cl}(i)$ must have been completely mined by the time period when i is completely mined.

We denote by $R^{cl}(i) := \sum_{k \in \mathcal{P}^{cl}(i)} R_k$ the total amount of rock contained in aggregate i and all of its (indirect) predecessors. The value $R^{cl}(i)$ is also called the *induced* amount of rock for aggregate i. If the induced amount of rock $R^{cl}(i)$ of some $i \in [N]$ exceeds the total mining capacity up to a period t, then aggregate i clearly cannot be mined completely by the end of period t. Consequently, we can fix the corresponding variables $x_{i,t}$ to zero.

Theorem 16 ([3,4]). *Each solution of* (NF), (AT), (DNF), *or* (DAT) *satisfies*

$$x_{i,t} = 0 \qquad\qquad \textit{for all } i \in [N],\ t \in [T] \textit{ with } R^{cl}(i) > \sum_{t'=1}^{t} M_{t'}. \tag{35}$$

Analogously, for each pair of aggregates $i \neq j$, we denote by $R^{cl}(ij) := \sum_{k \in \mathcal{P}^{cl}(i) \cup \mathcal{P}^{cl}(j)} R_k$ the total amount of rock contained in those aggregates k that preceed at least one of the aggregates i and j, which is called the commonly induced amount of rock for aggregates i and j. If the commonly induced rock $R^{cl}(ij)$ of some pair $i, j \in [N]$ exceeds the total mining capacity until the end of period t, both i and j cannot be mined completely. For each period t, these conflicts define a conflict graph $CG_t = ([N], E)$, which contains a node for each aggregate and and edge $\{i, j\} \in E$ if and only if the commonly induced rock exceeds the total mining capacity until period t, i.e., if and only if $R^{cl}(ij) > \sum_{t'=1}^{t} M_{t'}$. As any pair of aggregates i, j that are completely mined until the end of period t must be non-adjacent in this conflict graph, the set of all aggregates completely mined by the end of t must form a stable set in CG_t. Hence, the vector $(x_{i,t})_{i \in [N]}$ of an integer-feasible solution of (NF) or (AT) must satisfy all inequalities that are valid for the stable set polytope on the graph CG_t and, in particular, the clique inequalities (see Nemhauser and Wolsey [18], for example).

Theorem 17 ([3,4]). *Each solution of* (NF), (AT), (DNF), *or* (DAT) *satisfies*

$$\sum_{i \in C} x_{i,t} \leqslant 1 \qquad \textit{for all } t \in [T] \textit{ and each (inclusion-wise maximal) clique } C \textit{ in } CG_t. \tag{36}$$

As the number of potential variable fixings (35) is at most TN and the time to compute the induced rock amounts $R^{cl}(i)$ of *all* aggregates $i \in [N]$ is linear in N, all fixings (35) can be easily added to the initial formulation (NF), (AT), (DNF), or (DAT).

The conflict graphs CG_t for all $t \in [T]$ can be easily constructed in $O(N^3)$ time each. The maximum number of clique inequalities (36), however, may be exponentially large, so adding them all to the initial formulation is not a practical option. Furthermore, the separation problem for these inequalities is equivalent to a maximum weight clique problem, which is NP-hard in general. In our implementation of the algorithm, we therefore use the maximum weight clique algorithm by Borndörfer and Kormos [6] to separate the clique inequalities (36) heuristically. This algorithm is also used to separate the standard clique inequalities in general purpose ILP codes such as SCIP and proved to be very efficient in practice.

Note that, in contrast to the models discussed in [3], only the mining capacities define useful precedence constrained knapsacks that lead to strong valid inequalities. The processing capacities up to some period t impose no useful restrictions on the mining operations up to that period, as the material mined in early time periods can be stored in the stockpile for processing in later periods.

5 Computational results

5.1 Test instances

Obtaining real-world problem data from open pit mines is difficult, in part due to the high costs involved in gathering them. Public benchmark instances are typically not available. Our industry partner BHP Billiton Pty. Ltd. has provided us with realistic data from two open pit mines, which are used in our numerical experiments.

Data set *Marvin* is based on a block model provided with the Whittle 4X mine planning software [26], originally consisting of 8513 blocks which were aggregated to 85 so-called "panels", i.e. single layers of blocks without block-to-block precedence relations. These panels are used as aggregates in our experiments. On average, each aggregate has 2.2 immediate predecessor aggregates in this data set. The lifespan of this mine is 15 years.

Data set *Dent* is based on the block model of a real-world open pit mine in Western Australia, originally consisting of 96821 blocks which were aggregated to 125 panels. Each panel has an average of 2.0 immediate predecessor aggregates. The lifespan of this mine is 25 years.

The aggregations to panels, the cutoff grades (determining which blocks in each panel are immediately discarded as waste), and precedence relations between the panels were pre-computed by our industry partner. Scheduling periods are time periods of one year each with a discount rate of 10% per year. Realistic values for mining costs and processing profits as well as for mining and processing capacities per year were chosen by our industry partner.

Table 1 gives an overview over the size of the problem formulations presented in this paper for these instances. For discretised out-fraction formulations (DNF) and (DAT), we consider rough discretisations with $L = 5$ levels for both formulations and fine discretisations with $L = 250$ for (DNF) and with $L = 10$ for (DAT). The chosen level numbers in the fine discretisations ensure that a simpler algorithm, which first solves the mixed-integer linear relaxation of the formulation and then applies REBALANCE to the integer solution, finds a solution within 1% of global optimality.

5.2 Results

Our computational experiments were run single-threaded on a MacBookPro with 2.2 GHz Intel Core i7 CPU and 8 GB RAM. Our algorithm has been implemented in C++, using IBM CPLEX 12.3 as the branch-and-bound framework via its C callable library interface. The type of formulation to use and other parameters can be set via command line arguments.

The branching schemes and the REBALANCE algorithm presented in Section 4.2 have been added to the branch-and-bound framework via the branching-, incumbent- and heuristic-callback function interfaces of CPLEX. To obtain good fully feasible solutions early, we use a parameter setting that applies CPLEX's built-in heuristics more aggressively and thus produces more promising integer-feasible solution candidates to be post-processed by the REBALANCE algorithm.

Depending on the chosen formulation, either the trivial grade bound inequalities (30)–(33) for the basic formulation or the stronger inequalities (34) for the aggregate tracking formulation are created together with all other linear constraints of the respective formulation when building the initial linear relaxation. While the trivial grade bound inequalities (30)–(33) are added directly as normal inequalities to the active linear relaxation, inequalities (34) are added as so-called lazy constraints into a separate cut pool, from where they are moved to the active relaxation only if violated. This keeps the size of the active relaxation relatively small during the execution of the

Problem/	No. variables			No. constraints			Nonzeros
Formulation	total	bin.	cont.	total	linear	quadr.	
Marvin							
(NF)	5848	1445	4403	7631	7616	15	59920
(AT)	8753	1445	7308	10387	9112	1275	70680
(DNF), $L = 5$	5933	1530	4403	8039	8024	15	61468
(DNF), $L = 250$	10098	5695	4403	28864	28849	15	2102318
(DAT), $L = 5$	8838	1530	7308	24905	23630	1275	132951
(DAT), $L = 10$	8923	1615	7308	39440	38165	1275	270396
Dent							
(NF)	12600	3125	9475	15823	15800	23	161731
(AT)	18873	3125	15748	21875	19000	2875	185841
(DNF), $L = 5$	12725	3250	9475	16423	16400	23	164031
(DNF), $L = 250$	18850	9375	9475	47048	47025	23	3224081
(DAT), $L = 5$	18998	3250	15748	53225	50350	2875	320416
(DAT), $L = 10$	19123	3375	15748	84600	81725	2875	617541

Table 1. Size of formulations for instances *Marvin* and *Dent* (before presolving)

algorithm. Also, all variable fixings (35) are performed during initialization. The clique inequalities (36), on the other hand, are generated via a heuristic separation subroutine, which is included into the CPLEX branch-and-bound framework via the cut-callback function and executed only at the root node of the branch-and-bound tree.

In our test, we tried to solve the two test problems to a proven optimality gap of 1% and or 0.1% with the different formulations. Table 2 shows the results obtained with all four formulations within a time limit of 12 CPU hours when targeting an optimality tolerance of 1.0%. For the aggregate tracking formulations (AT) and (DAT), we also report the results for a target optimality tolerance of 0.1%. The column LP UB shows the upper bound obtained from the initial linear relaxation. The columns in the group Root node show the values of the upper bound, of the best solution found, and the resulting proven optimality gap after processing the root node of the branch-and-and bound tree (that is, after CPLEX completely processed the root node applying its build-in separation procedures and Rebalance has been applied to all integer-feasible solutions found by CPLEX's build-in heuristics at the root node). The columns in the group Final reports the same values at the end of the algorithm, when it either reached the time limit of 12 hours or the target optimality gap, together with the number of branch-and-bound nodes explored and the total CPU time. Results for fine discretisations of (DNF) and (DAT), with $L \geqslant 250$ and $L \geqslant 10$, respectively, are not reported in Table 2.

One easily observes that the aggregate tracking formulation (AT) and the discretised aggregate tracking formulation (DAT) perform much better than the natural formulations (NF) and (DNF). The main reason for this is that the bounds obtained from the linear relaxation of the natural formulations (NF) and (DNF) are very poor: For *Marvin*, the gap between the natural formulation's LP bound and the globally optimal solution is roughly 35%, while is only 3.2% for the aggregate tracking formulation. Also, this gap reduces only very slowly when branching on the violations of the natural formulation's mixing constraints (11). Typically, the differences between the out-fraction of metal and the out-fraction of ore observed for an optimal solution of a branch-and-bound node's

Problem/	LP	Root node			Final				
Formulation	UB	UB	LB	Gap(%)	UB	LB	Gap(%)	Nodes	Time(s)
Marvin									
(NF)	9.3847	9.3696	6.6778	28.73	7.8996	6.8618	13.14	876533	12 h
(DNF), $L = 5$	9.3478	9.2859	6.7492	27.32	7.7780	6.8493	11.94	790004	12 h
(AT)	7.1773	7.1534	6.8167	4.71	7.0138	6.9444	1.00	2594	162
					6.9567	6.9498	0.10	202877	9811
(DAT), $L = 5$	7.1738	7.1527	6.8814	3.79	7.0180	6.9487	1.00	2543	470
					6.9569	6.9500	0.10	134709	9006
Dent									
(NF)	5.4942	5.4887	3.2321	94.11	5.3150	4.8631	8.50	406057	12 h
(DNF), $L = 5$	5.4942	5.4898	3.2321	94.11	5.3173	4.8640	8.53	413680	12 h
(AT)	5.0307	4.9880	4.8806	2.15	4.9354	4.8866	1.00	450	140
					4.8927	4.8878	0.10	118275	9662
(DAT), $L = 5$	5.0300	4.9885	4.8796	2.18	4.9356	4.8873	1.00	599	361
					4.8927	4.8878	0.10	124604	23885

Table 2. Results for instances *Marvin* and *Dent*

linear relaxation are indeed as large as allowed by the branching constraints (26) and (27) at that node. Branching on the violations of the aggregate tracking formulation's mixing constraints (18) is much more effective. As both the out-fraction of ore and the out-fraction of metal are averages of the aggregate-wise out-fractions, these two out-fractions are typically much closer than the out-fractions of the individual aggregates, especially if the grades of the aggregates in the stockpile are similar. Also, the branching scheme for the aggregate tracking formulation is inherently more aggressive, as it adds N inequalities instead of 2 in each branching operation.

We also find that the discretised out-fraction models (DNF) and (DAT) do not perform significantly better than their counterparts (NF) and (AT) in general. The improvement of the LP relaxation value by adding a piecewise linear discretisation of the mixing constraints is negligible, while the substantially larger size of the formulation may lead to larger solution times.

In Table 3, we consider the effect on NPV of mine extraction scheduling with versus without the use of a stockpile. We also compare several heuristic approaches to finding feasible solutions to the former, by solving models without any mixing constraints, or with mixing constraints modelled only by discretization, and then repairing the best feasible solution found using the Rebalance method. Table 3 shows the objective values and computation times of (i) the best solution found with our algorithm for the OPMPSP+S problem, (in the rows labelled (AT)), (ii) the corresponding OPMPSP problem without a stockpile, (iii) the best solution that is obtained by applying Rebalance to the best solution found by MILP models with mixing constraints removed (IP(NF) and IP(AT)) and (iv) the best solution that is obtained by applying Rebalance to the best solution found by MILP models with mixing constraints modelled only by discretization (IP(DNF) and IP(DAT)). For the OPMPSP problem without a stockpile, and for the runs where Rebalancing was applied to postprocess the solutions of the mixed-integer linear relaxations of (NF) and (AT), the mixed-integer linear relaxations have been solved to an optimality tolerance of 0.01%, which is the CPLEX default setting, as we are mainly interested in seeing how good (or bad) the resulting solutions are compared to the globally optimal solution of the full mixed-integer non-linear model. The other three runs, i.e., our specialized branch-and-bound algorithm to solve (AT) and the two runs postprocessing the the solutions of the mixed-integer linear relaxations of (NF) and (AT), have been parametrized in such a way that the resulting solutions would be within 1% of the globally optimal solution of the full mixed-integer non-linear model. In our specialized algorithm, this is achieved by simply stopping the algorithm when the target gap of 1% is reached. For the other two runs, we have chosen a discretization of the out-fractions (with uniform δ_l for all levels and time periods) that was sufficiently fine to ensure that the difference between the objective function value of an integer-feasible solution and that of its postprocessed solution is no more than 0.5%. We then solved the resulting mixed-integer linear relaxation of (DNF) and (DAT) to an optimality tolerance of 0.5%. This ensures that the gap between the final solution and the globally optimal solution of the full non-linear problem is at most 1%. For the problem instances at hand, this required $L = 10$ levels for the discretised aggregate tracking formulations and $L = 250$ levels for the discretised natural formulation. Note that these level values have been determined experimentally by evaluating the objective function gaps introduced by Rebalance for the first 10 solutions produced by CPLEX when solving the integer-linear relaxations. To actually *guarantee* that the optimal solution of the integer-linear relaxation does not worsen by more than 0.5% when applying Rebalance to an integer solution of (DNF), we would need $L = 1000$ for *Marvin* and $L = 2400$ for *Dent*, which leads to formulations that are computationally not tractable.

Problem /	Branch-and-bound					Solution		
Solution method	UB	LB	Gap (%)	Nodes	Time (s)	Val	GapB (%)	GapO (%)
Marvin								
(AT)	7.0138	6.9444	1.0	2594	162	6.9444	1.0	0.2
OPMPSP	6.5883	6.5877	0.0	4930	67	6.5883	—	5.3
IP(NF) + Rebalance	9.2963	9.2954	0.0	343	11	5.5045	40.8	20.9
IP(AT) + Rebalance	7.0134	7.0128	0.0	3801	120	6.7972	3.1	2.3
IP(DNF), $L=250$ + Rebal	8.9626	6.7687	24.5	59387	12 h	6.7599	24.5	2.8
IP(DAT), $L=10$ + Rebal	6.9912	6.9565	0.5	7717	2187	6.9476	0.6	0.1
Dent								
(AT)	4.9354	4.8866	1.0	450	140	4.8866	1.0	0.1
OPMPSP	4.8709	4.8704	0.0	8900	323	4.8704	—	0.4
IP(NF) + Rebalance	5.4059	5.4054	0.0	11921	1194	4.3838	18.9	10.3
IP(AT) + Rebalance	4.8892	4.8887	0.0	509538	14321	4.8840	0.1	0.1
IP(DNF), $L=250$ + Rebal	5.4356	4.8380	11.0	43971	12 h	4.8344	11.1	1.1
IP(DAT), $L=10$ + Rebal	4.9123	4.8879	0.5	12590	5859	4.8878	0.5	0.0

Table 3. Objective values of OPMPSP+S versus objective values of OPMPSP without stockpile and of best solution obtained by applying Rebalance to the best IP solutions.

The columns in the group Branch-and-bound show the values of the upper bound, of the best solution found, the resulting proven gap between these two values, the number of branch-and-bound nodes explored, and the total CPU time at the end of the branch-and-bound algorithm, i.e., when it either reached the time limit of 12 hours or the target optimality gap. The branch-and-bound algorithm is either our specialised algorithm for (AT) or the standard branch-and-bound applied to the respective mixed-integer linear relaxation in the other cases. The columns in the group Solution show the values of the final solutions obtained after applying Rebalance to the best integer-feasible solution obtained in the branch-and-bound phase. Column Val reports the solution's value. Column GapB shows the gap between the solution value and the best bound proven by the respective method, while GapO shows the gap observed between the solution's value and the best bound proven by *any* method, which is (for the precision shown in the table) equal to the gap to the global optimal solution of the full non-linear model.

It can be seen from the Tables 2 and 3 that neither ignoring the stockpile, nor post-processing the best integer-feasible solution of a mixed-integer linear relaxation with relaxed mixing constraints, nor using a sufficiently fine *a priori* discretisation of the mixing constraints are practically competitive with our approach. Ignoring the stockpile or post-processing the best integer-feasible solution of the natural or aggregate tracking model without the mixing constraints may lead to very poor solutions that differ a lot from the real optimum. The sufficiently fine *a priori* discretisations, on the other hand, lead to very large mixed-integer linear programs, that are extremely hard to solve. For the fine discretisations of the natural formulation, the mixed-integer linear programs had optimality gaps of more than 10% after 12 hours computation time. Using rougher discretisations with less levels will certainly reduce the optimality gaps and the solution times of the mixed-integer relaxation, but then an overall gap of 1% cannot be guaranteed anymore. With the fine discretisatons of the aggregate tracking formulation we are able to solve the test instances with a 1% optimality gap guarantee, but the solution times exceed those of our specialized algorithm substantially.

Our branch-and-bound algorithm, which adaptively refines the relaxation where needed via its special branching schemes, proves to be the computationally most effective and efficient compromise.

Acknowledgements The authors are very grateful to Merab Menabde and Peter Stone (BHP Billiton) for their ongoing support and guidance on a variety of practical mining-related issues and for numerous technical suggestions and insightful feedback that improved the content and exposition of this work. This research is supported by the Australian Research Council under Linkage Project grants LP0348907 and LP0561744, and by BHP Billiton Limited and by the DFG Research Center Matheon – Mathematics for key technologies.

References

1. A. Akaike and K. Dagdelen. A strategic production scheduling method for an open pit mine. In *Proc. 28th Internat. Appl. Comput. Oper. Res. Mineral Indust. (APCOM) Symposium*, SME, Littleton, CO.
2. Daniel Bienstock and Mark Zuckerberg. Solving LP relaxations of large-scale precedence constrained problems. In *Integer Programming and Combinatorial Optimization: Proceedings of the 14th Conference on Integer Programming and Combinatorial Optimization (IPCO 2010), Lausanne, Switzerland*, number 6080 in LNCS, pages 1–14. Springer, 2010.
3. A. Bley, N. Boland, C. Fricke, and G. Froyland. A strengthened formulation and cutting planes for the open pit mine production scheduling problem. *Computers and Operations Research*, 37:1641–1647, 2010.
4. N. Boland, A. Bley, C. Fricke, G. Froyland, and R. Sotirov. Clique-based facets for the precedence constrained knapsack problem. *Mathematical Programming*, to appear.
5. Natashia Boland, Irina Dumitrescu, Gary Froyland, and Ambros M. Gleixner. LP-based disaggregation approaches to solving the open pit mining production scheduling problem with block processing selectivity. *Computers & Operations Research*, 36:1064–1089, 2009. doi:10.1016/j.cor.2007.12.006.
6. R. Borndörfer and Z. Kormos. An algorithm for maximum cliques. Unpublished working paper, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 1997.

7. Louis Caccetta and Stephen P. Hill. An application of branch and cut to open pit mine scheduling. *Journal of Global Optimization*, 27:349–365, 2003.
8. R. Chicoisne, D. Espinoza, M. Goycoolea, E. Moreno, and E. Rubio. A new algorithm for the open-pit mine scheduling problem. Technical report, 2009. Submitted for publication, available at mgoycool.uai.cl.
9. Kadri Dagdelen and Thys B. Johnson. Optimum open pit mine production scheduling by lagrangian parameterization. In *Proceedings 19th APCOM Symposium of the Society of Mining Engineers (AIME)*, pages 127–142, 1986.
10. Christopher Fricke. *Applications of Integer Programming in Open Pit Mining.* PhD thesis, University of Melbourne, August 2006.
11. Ambros M. Gleixner. Solving large-scale open pit mining production scheduling problems by integer programming. Master's thesis, Technische Universität Berlin, June 2008.
12. D. Hochbaum. A new-old algorithm for minimum-cut and maximum flow in closure graphs. *Networks*, 37(4):171–193, 2001.
13. S. Hoerger, L. Hoffman, and F. Seymour. Mine planning at newmont's nevada operations. *Mining Engineering*, 51(10):26–30, 1999.
14. T.B. Johnson. *Optimum open pit mine production scheduling.* PhD thesis, Operations Research Department, University of California, Berkeley, 1968.
15. Helmut Lerchs and Ingo F. Grossmann. Optimum design of open-pit mines. *Transactions CIM*, LXVIII:17–24, 1965.
16. Merab Menabde, Gary Froyland, Peter Stone, and Gavin Yeates. Mining schedule optimisation for conditionally simulated orebodies. In *Proceedings of Orebody Modelling and Strategic Mine Planning – Uncertainty and Risk Management*, pages 357–342, Perth, WA, November 2004.
17. E. Moreno, D. Espinoza, and M. Goycoolea. Large-scale multi-period precedence constrained knapsack problem: A mining application. *Electronic Notes in Discrete Mathematics*, 36:407–414, 2010.
18. G.L. Nemhauser and L.A. Wolsey. *Integer and Combinatorial Optimization.* Wiley-Interscience, New York, 1988.
19. A. Newman, E. Rubio, A. Caro, A. Weintraub, and K. Eurek. A review of operations research in mine planning. *Interfaces*, 40(3):222–245, 2010.
20. Ahmet H. Onur and Peter A. Dowd. Open-pit optimization – part 2: Production scheduling and inclusion of roadways. *Transactions of the Institution of Mining and Metallurgy (Section A: Mining Industry)*, 102:A105–A113, 1993.
21. J.C. Picard. Maximal closure of a graph and applications to combinatorial problems. *Management Science*, 22:1268–1272, 1976.
22. Salih Ramazan. The new fundamental tree algorithm for production scheduling of open pit mines. *European Journal of Operational Research*, 177(2):1153–1166, 2007.
23. J.M.W. Rhys. A selection problem of shared fixed costs and network flows. *Management Science*, 17:200–207, 1970.
24. Martin L. Smith. Optimizing inventory stockpiles and mine production: An application of separable and goal programming to phosphate mining using AMPL/CPLEX. *CIM Bulletin*, 92:61–64, 1999.
25. Boleslaw Tolwinski and Robert Underwood. A scheduling algorithm for open pit mines. *IMA Journal of Mathematics Applied in Business & Industry*, 7:247–270, 1996.
26. Gemcom software international. Vancouver, BC, Canada.
27. Q. Wang. Long-term open-pit production scheduling through dynamic phase-bench sequencing. *Transactions of the Institution of Mining and Metallurgy (Section A: Mining Industry)*, 105:A99–A104, 1996.
28. J. Yarmuch and J. Ortiz. A novel approach to estimate the gap between the middle- and short-term plans. In *Proc. 35th Appl. Comput. Oper. Res. Mineral Indust. (APCOM) Symposium*, pages 419–425, Wollongong, NSW, Australia, 2011.

A List of Notation

T	number of time periods
N	number of aggregates
$\mathcal{P}(i)$	set of immediate predecessors of aggregate i
R_i, O_i	rock and ore tonnage of aggregate i, respectively [tonnes]
A_i	tonnage of attribute (metal) in aggregate i (A_i for a single attribute) [tonnes]
c	sales price of attribute (metal) [\$m/tonne]
m	mining cost (per tonne of rock) [\$m/tonne]
p	processing cost (per tonne of ore) [\$m/tonne]
M_t	mining capacity for time period t [tonnes of rock]
P_t	processing capacity for time period t [tonnes of ore]
δ_t	discount factor for time period t

Table 4. List of notation