# Optimal Scheduling of Copper Concentrate Operations under Uncertainty

Pengfei Cheng[a,1], Pablo Garcia-Herreros[b], Mangalam Lalpuria[b], Ignacio E. Grossmann[a,*]

[a] *Center for Advanced Process Decision-Making, Department of Chemical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, United States*
[b] *Research, Development & Innovation Department, Aurubis AG, Hamburg 20539, Germany*

**Abstract**

We propose a continuous-time scheduling model for the logistic and blending operations of copper concentrates with uncertain composition. The formulation, based on the Multi-Operation Sequencing (MOS) model, gives rise to a large-scale nonconvex mixed-integer nonlinear programming (MINLP) model. We adopt a two-step MILP-NLP decomposition strategy and enhance the MILP relaxation to propose schedules that significantly reduce the optimality gaps. The bounded uncertainty in element composition of the concentrates is addressed by an extended robust MOS model, which combines robust optimization and flexibility analysis techniques. The effectiveness of the models and the solution strategy is validated with an illustrative example and an industrial case study.

*Keywords:* scheduling, copper operations, mixed-integer nonlinear porogramming, robust optimization

## 1. Introduction

Copper is used extensively throughout domestic and industrial applications because of its excellent corrosion resistance, malleability, and thermal conductivity (Langner, 2011). For this reason, copper demand is constantly growing due to the increasing needs on information

*Corresponding author
*Email address:* grossmann@cmu.edu
[1]Present address: School of Chemical and Biomolecular Engineering, Georgia Institute of Technology, Atlanta, GA 30332, United States

technology, energy supply, manufacturing, and other copper-intensive industries. Demand growth has been pushing the copper industry to improve its production efficiency, creating opportunities for mathematical programming techniques to be applied in the copper industry to systematically improve the performance of different stages of the process.

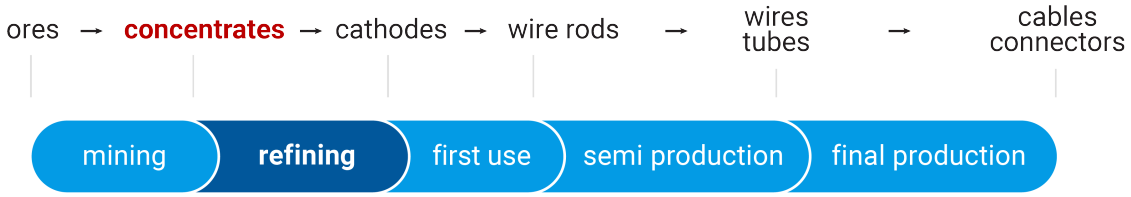


Figure 1: Value-added chain of copper industry

In this article, we optimize the logistic and blending operations of a copper smelting process, which is the first step of the refining process in the copper value-added chain shown in Figure 1. Copper concentrates are solid materials obtained from copper ores that undergo a concentration process. The smelting process transforms the copper concentrates to copper mattes, increasing the copper content from 25-35% to 50-70%. One important challenge in this process is to find the best mixture of various raw concentrates and recycling materials, such that the gross margin of the process is maximized, the environmental standards are met, and the quality of the products is guaranteed.

Scheduling problems are very prevalent in process industries such as chemicals, food, pharmaceuticals, and oil and gas (Harjunkoski et al., 2014). Generally, the objective of the scheduling problem is to maximize the profit, minimize the cost, or minimize the makespan. The decisions in scheduling problems usually include (Maravelias, 2012):

- selection and lot-sizing of tasks
- assigning tasks to resources (units)
- timing of tasks
- sequencing of tasks

Due to the complexity of the scheduling problem, mathematical programming approaches have been introduced to systematically improve the quality of the schedules. A number of review articles summarizing diverse mathematical models for scheduling have been pub-

lished in the past (Méndez et al., 2006; Maravelias, 2012; Harjunkoski et al., 2014). They identify two distinctive types of models according to time representation: discrete-time and continuous-time formulations. Discrete-time formulations divide the time horizon into uniform time intervals and often lead to large-size, tight formulations. Early works in process scheduling explored various discrete-time representations based on the State-Task Network model (STN) (Kondili et al., 1993) and the Resource-Task Network (RTN) model (Pantelides, 1994). Continuous-time formulations partition the time horizon using time-points, time intervals or event points, and the length of each time period is to be determined by the optimization model. These formulations are often smaller in size because they need fewer time periods to represent the schedule, but they tend to have weaker relaxations. Continuous-time models can be further divided into two types based on the coupling between task and unit events: single time grid (Castro et al., 2001) and multiple time grids (unit-specific event based) (Ierapetritou and Floudas, 1998a,b).

For the scheduling of copper concentrate operations, Song et al. (2018) used a discrete-time formulation to model the process as an mixed-integer nonlinear programming (MINLP) problem; they compared the performance of a split fraction model and a process network model for the same problem. Lalpuria (2017) developed two continuous-time formulations and demonstrated that the Multi-Operation Sequencing (MOS) model is superior to the Single-Operation Sequencing (SOS) model with respect to computational time. The MOS model is a continuous-time model based on the relationship between non-overlapping operations. It has proved to be efficient for crude oil scheduling problems (Mouret et al., 2011).

An important challenge related to copper concentrate operations arises from the uncertainty in their element composition. After being mined, copper ores go through a complex process in which they are transformed into concentrates; the process include crushing, grinding, flotation, and drying. The complexity of these processes causes the element compositions of each concentrate to fluctuate within certain ranges around their nominal value. These fluctuations are critical for the preparation of the concentrate blend, since the copper refining requires strict conditions on the composition of different elements. In order to comply

with these conditions, it is important to take the variability in element composition into account by treating it as a bounded uncertainty within the optimization framework.

Research on optimization under uncertainty has been an active area since 1970s (Grossmann et al., 2016). Two major uncertainty modeling techniques are robust optimization (Bertsimas et al., 2011) and stochastic programming (Birge and Louveaux, 2011). Robust optimization follows the idea of optimizing for the worst realization of the uncertainty while guaranteeing the feasibility of the solution. On the other hand, stochastic programming optimizes the expectation of an objective function based on the probability distribution of the uncertain parameters. Results from robust optimization are more conservative than those obtained from stochastic programming. Therefore, stochastic programming is suitable for long-term planning problems to maximize the expected profit, while robust optimization is better suited for process scheduling under uncertainty, as the feasibility of the solution is critical for short-term problems. Alonso-Ayuso et al. (2014) used stochastic programming to consider the effect of changing copper prices on the expected profit for a five-year copper extraction planning problem. Li and Ierapetritou (2008) reviewed several methods for process scheduling under uncertainty, including both robust optimization and stochastic programming. In addition to these two methods, flexibility analysis (Grossmann et al., 2014) can also be a suitable approach for scheduling under uncertainty, since it focuses on feasibility by accounting for recourse actions. Flexibility analysis is an uncertainty modeling technique that was originally developed for plant design under uncertainty, and it has been used in the process systems engineering community for more than thirty years. As shown in a recent article by Zhang et al. (Zhang et al., 2016), flexibility analysis can yield either identical or more rigorous results than robust optimization for linear problems, although it is computationally more expensive.

In this work, we propose a continuous-time scheduling formulation based on the MOS model. The goal is to schedule the logistic and blending operations necessary to feed the copper smelting process with suitable concentrate mixtures. The model includes robust optimization and flexibility analysis techniques to deal with the uncertainty in element composition of concentrates. A two-step MILP-NLP decomposition strategy is adopted and

enhanced to solve the scheduling models efficiently.

The remainder of this paper is organized as follows. The copper smelting process and the challenges derived from uncertainty in element composition are described in Section 2. The time representation and the mathematical formulation of the deterministic MOS model are presented in Section 3. In Section 4, the robust MOS model that considers uncertainty in element composition is developed. Section 5 presents the solution strategy based on a two-step MILP-NLP decomposition strategy. Finally, an illustrative example and an industrial case study are described in Section 6; the corresponding results are shown in Section 7.

## 2. Problem statement

### *2.1. Logistic and blending operations for copper smelting*

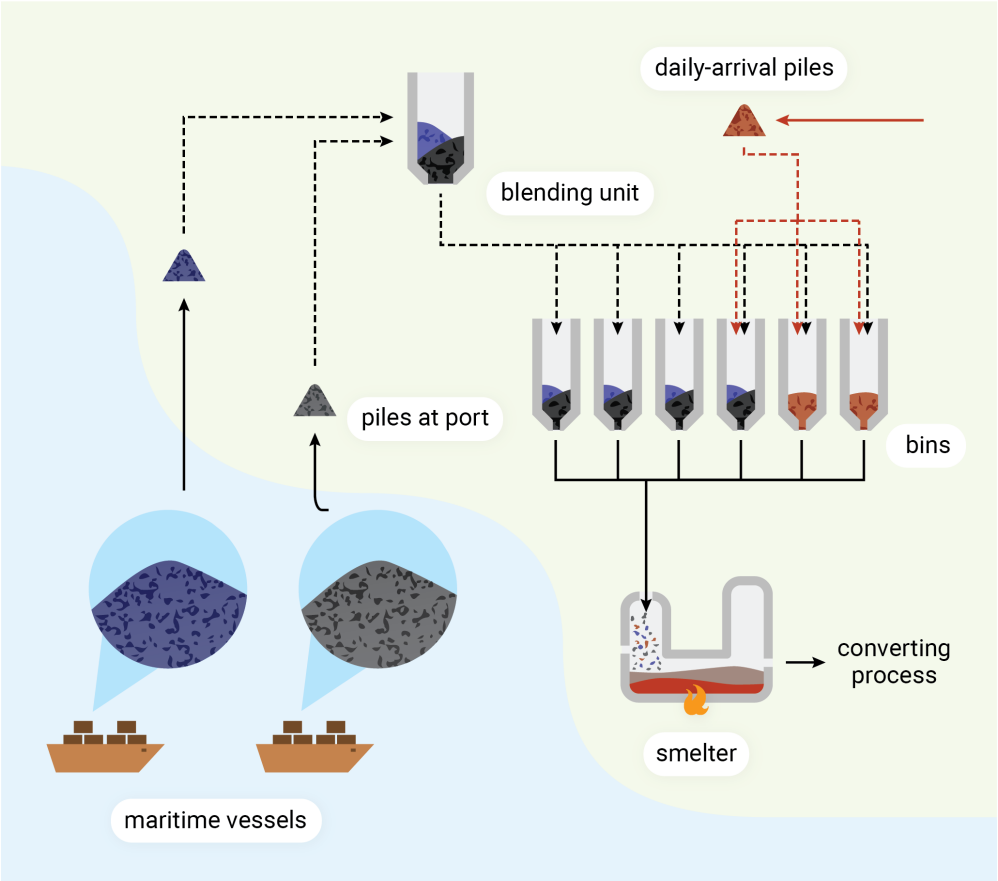


Figure 2: logistic and blending operations of copper smelting

The logistic and blending operations for a standard copper smelting process are represented in Figure 2. The concentrates available for smelting are located at storage facilities near the port since many of them arrive by maritime transportation. The concentrates are first unloaded from maritime vessels at the port and piled separately; then, they are transferred to a blending unit where an initial blend is prepared. The blends are then distributed to several bins, from where they are simultaneously discharged into the smelter. In addition to concentrates, daily-arrival materials are also transferred to bins and processed at the smelter. These daily-arrivals are a special raw material consisting mainly of process leftovers and scraps; processing these materials improves resource utilization and contributes to energy saving of the smelting process.

The following six parts are treated as units in the process: (1) maritime vessels, (2) piles at the port, (3) blending unit, (4) daily-arrival piles, (5) bins, (6) smelter. Correspondingly, six types of operations are allowed: (1) concentrate unloading from maritime vessels to piles at the port, (2) concentrate transfer from piles at the port to the blending unit, (3) transfer from blending unit to bins, (4) unloading of daily-arrival materials to daily-arrival piles, (5) transfer from daily-arrival piles to bins, (6) final transfer from bins to the smelter.

The following operation rules are specified for the units:

1. The piles at the port, the blending unit, and the bins have unlimited capacities.
2. The piles at the port and the blending unit cannot be charged and discharged simultaneously.
3. The bins and daily-arrival piles can be charged and discharged simultaneously.
4. The concentrates in all bins must be transferred to the smelter simultaneously.
5. The contribution of each bin feeding the smelter must be within specified ranges.
6. The total inventory of daily-arrival materials left in the process must be below a certain threshold at the end of time horizon.

The objective of the scheduling problem for the process described above is to maximize the gross margin for processing the concentrates over a given time horizon. In this maximization problem, the assignment, sizing, sequencing, and timing of each transfer operation need to be determined, while the specified operation rules must be followed. In addition

to the operation rules, there are several restrictions on the element composition for the blends feeding the smelter. These restrictions must be satisfied to guarantee the operating conditions necessary for the smelting and the successive downstream process.

### *2.2. Uncertainty in element composition*

The natural variability in ore grades and small fluctuations in the concentration conditions are the main sources of uncertainty in the element composition of concentrates. Since the smelting process is very sensitive to any changes in blend composition, this variability must be considered in the scheduling problem to avoid sub-optimal or even infeasible solutions yielding off-spec products.

For each concentrate, we consider a subset of elements whose composition fluctuates within a bound relative to its nominal value. Therefore, this variability can be formulated as a bounded symmetric uncertainty. By using $f_{c,k}$ to denote the mass fraction of element $k$ in concentrate $c$, the uncertainty set can be expressed as follows:

$$U = \left\{ \vec{F} \;\middle|\; f_{c,k} \in \left[(1-\delta)\bar{f}_{c,k}, (1+\delta)\bar{f}_{c,k}\right] \quad c \in C, k \in K_U \right\} \tag{1}$$

where $\bar{f}_{c,k}$ is the nominal mass fraction value of element $k$ in concentrate $c$, $\delta$ is the relative bound of the fluctuation, and $K_U$ is the set of elements whose composition fluctuate. The mass fraction ($f_{c,k}$) is related to various quality requirements of the process, which are formulated as a set of linear constraints in Section 3.

## 3. Multi-Operation Sequencing model

We use the MOS model to formulate the deterministic scheduling problem. The MOS model is a continuous-time, multiple-time-grid scheduling model that has been previously applied to crude oil scheduling (Mouret et al., 2011); it has proved to be computationally efficient compared to other common time formulations. In this section, we first describe its time representation, then explain the mathematical formulation in detail.

### *3.1. Time representation*

In the study by Mouret et al. (2011), the time representation of the MOS model is built on the concept of priority slot, and the computational efficiency of the model is based on the application of cliques. However, the connection between the clique and the time representation in the model was not explicitly presented. Here, an extensive time representation for the MOS model is given to explain in detail the function of the cliques.

The time representation of the MOS model is based on assigning a priority slot $i$ to each execution of operation $v$. A graphic representation of the MOS model is presented in Figure 3, where the execution of each operation is represented by a segment labeled with a certain priority slot. The total number of priority slots $|I|$ is specified *a priori*, which implies that each operation can be assigned to at most $|I|$ priority slots. The time span of a certain slot for different operations are independent of each other.

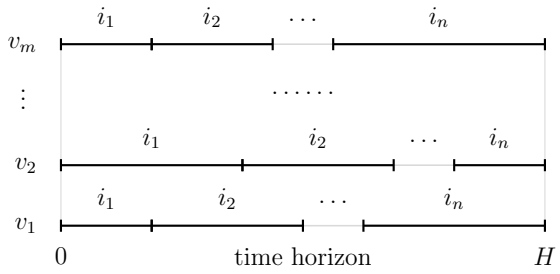


Figure 3: Time representation of MOS model

The priority slots specify the sequence in which the operations are executed. The priority mechanism works in two ways. For a single operation, the operation can be executed several times by assigning it to multiple priority slots. Each execution does not overlap with each other, and occurs in the order defined by the priorities. For multiple operations, the assignment of priority slots is based on the concept of cliques.

A clique represents a set of non-overlapping operations, i.e. operations that cannot be executed simultaneously, in the MOS model. In graph theory (Diestel, 2005), a clique refers to a subset of vertices in an undirected graph such that any two distinct vertices are connected to each other. In the MOS model, all operations can be mapped as vertices in a graph,

and two vertices will be connected if the operations they represent are non-overlapping. A small example of cliques is presented in Figure 4. Both Tank 1 and Tank 2 cannot be charged/discharged by more than one operation simultaneously, generating two sets of non-overlapping operations ($\{v_1, v_2, v_3\}$ for Tank 1 and $\{v_3, v_4, v_5\}$ for Tank 2). Correspondingly, there are two cliques $w_1$ and $w_2$ in the process.

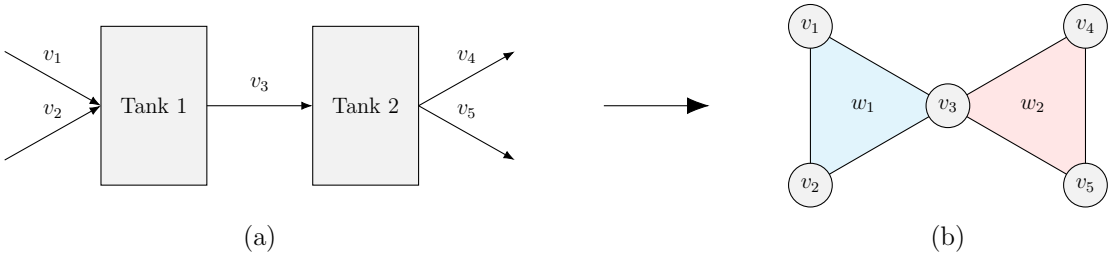


Figure 4: Clique in MOS model: (a) non-overlapping operations; (b) corresponding cliques

For all the operations in a clique, at most $|I|$ slots can be assigned to these operations, and each slot can only be assigned to a single operation. These operations are executed in the priority order, and their execution time cannot overlap with each other. As an example, for clique $w_1$ in Figure 4, a potential schedule is presented in Figure 5. The three non-overlapping operations are assigned to three different priority slots, and each slot is assigned to only one operation. This mechanism ensures that the executions of operations in the same clique do not overlap with each other.

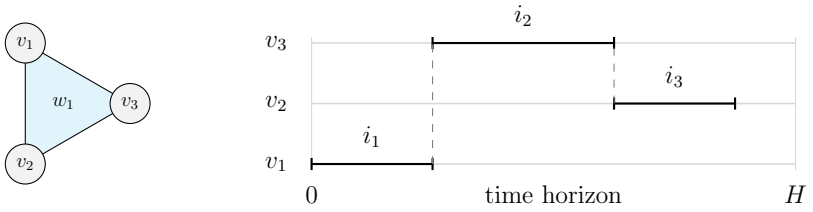


Figure 5: Priorities of multiple operations in one clique

For multiple operations in different cliques, each clique functions independently and the operations from different cliques can overlap with each other. In other words, each priority slot can be assigned to several operations from different cliques. Operations belonging to

multiple cliques connect these cliques, creating interdependencies in the execution priorities. For instance, Figure 6 shows a feasible schedule that considers the interaction between both cliques ($w_1$ and $w_2$) presented in Figure 4. The schedule of operations in clique $w_1$ is covered by blue dots, and schedule of operations in clique $w_2$ is covered by a red block. In Figure 6, it can be observed that 3 slots are assigned to operations $v_1, v_2$ and $v_3$ in clique $w_1$, and 2 slots are assigned to operations $v_3$ and $v_5$ in clique $w_2$. These assignments guarantee that execution of operations within each clique do not overlap with each other. Since $v_2$ and $v_5$ belong to two different cliques, they can both be assigned to slot $i_3$ and their execution time overlaps with each other. In this schedule, operation $v_3$ connects cliques $w_1$ and $w_2$. Notice that in this schedule, operation $v_4$ is not executed.

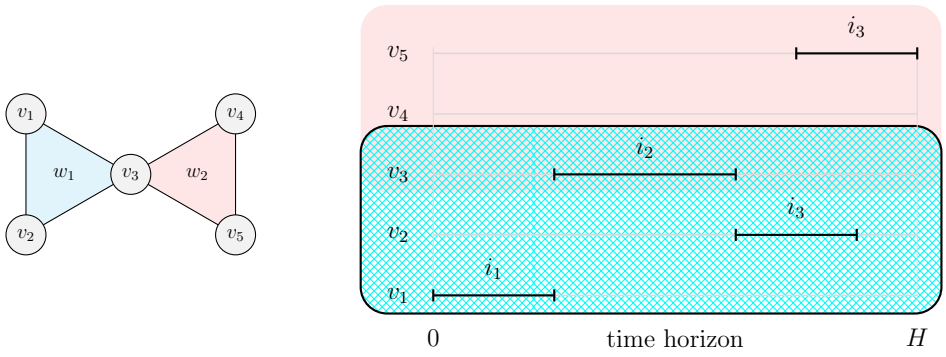


Figure 6: Priorities of multiple operations in different cliques

Based on the cliques, the sequential relationship among non-overlapping operations is established. The schedule for the whole process can be seen as a combination of the schedules of all cliques. In this way, each priority slot can be utilized by different cliques and the non-overlapping structure is captured in the time representation, leading to an efficient mathematical formulation.

### 3.2. Mathematical formulation

#### 3.2.1. Assignment constraints

Binary variable $Z_{i,v}$ denotes whether operation $v$ is assigned to priority slot $i$. In clique $W'$, at most one operation $v$ can be assigned to slot $i$.

$$\sum_{v \in W'} Z_{i,v} \leq 1 \qquad i \in I, W' \subset W \tag{2}$$

#### 3.2.2. Cardinality constraints

The cardinality of operations in clique $W'$ should be within its lower and upper bounds. For example, since vessels are required to unload all concentrates exactly once, the lower and upper bounds of cardinalities of unloading operations are both set to 1.

$$\underline{N}_{W'} \leq \sum_{i \in I} \sum_{v \in W'} Z_{i,v} \leq \bar{N}_{W'} \qquad W' \subset W \tag{3}$$

#### 3.2.3. Time constraints

The end time $E_{i,v}$ of the operation $v$ at slot $i$ is the sum of its start time $S_{i,v}$ and its duration $D_{i,v}$.

$$E_{i,v} = S_{i,v} + D_{i,v} \qquad i \in I, v \in V \tag{4}$$

If operation $v$ is assigned to slot $i$, then its start time, duration and end time should be within their lower and upper bounds respectively. Otherwise, the time variables should be zero.

$$\underline{S}_v \cdot Z_{i,v} \leq S_{i,v} \leq \bar{S}_v \cdot Z_{i,v} \qquad i \in I, v \in V \tag{5}$$

$$\underline{E}_v \cdot Z_{i,v} \leq E_{i,v} \leq \bar{E}_v \cdot Z_{i,v} \qquad i \in I, v \in V \tag{6}$$

$$\underline{D}_v \cdot Z_{i,v} \leq D_{i,v} \leq \bar{D}_v \cdot Z_{i,v} \qquad i \in I, v \in V \tag{7}$$

#### 3.2.4. Non-overlapping constraints

The non-overlapping constraints specify the time relationship between any two operations in a clique. In clique $W'$, if slot $i$ is assigned to an operation, then the gap between the end time ($E_{j,v}$) of operation at an earlier slot $j$ and the start time ($S_{i,v}$) of operation at slot $i$ should be at least the sum of durations corresponding to all slots between $i$ and $j$. According

to the assignment constraints (2) and time constraints (5)-(7), at most one operation in the clique $W'$ can have non-zero values for the time variables ($E_{i,v}$, $S_{i,v}$ and $D_{i,v}$) at each slot. Equation (8) is formulated as a big-M constraint, such that if slot $i$ is not assigned to any operation, the time variables at slot $i$ are all zero and the constraint is relaxed. Here, $H$ is the length of the time horizon and serves as a big-M parameter.

$$\sum_{v\in W'} E_{j,v} + \sum_{v\in W'} \sum_{\substack{i'\in I,\\ j<i'<i}} D_{i',v} \leq \sum_{v\in W'} S_{i,v} + H\cdot\left(1-\sum_{v\in W'} Z_{i,v}\right) \qquad i\in I, j\in I, i>j, W'\subset W \tag{8}$$

In addition, it is necessary to ensure that executions of the same operation at different priority slots do not overlap with each other. This condition is enforced with Equation (9).

$$E_{j,v} + \sum_{\substack{i'\in I,\\ j<i'<i}} D_{i',v} \leq S_{i,v} \qquad i\in I, j\in I, i>j, v\in V \tag{9}$$

*3.2.5. Mass balance constraints*

Variable $L^c_{i,r,c}$ denotes the accumulated level of concentrate $c$ in unit $r$ at slot $i$, and variable $M^c_{i,v,c}$ denotes the amount of concentrate $c$ in operation $v$ at slot $i$. The mass balance constraints are based on these two variables at the concentrate level. $L^c_{i,r,c}$ in unit $r$ at slot $i$ equals the sum of the initial level $L^{c,\text{initial}}_{r,c}$, the accumulation terms from inlet transfer operations, and the consumption terms from outlet transfer operations from all previous slots.

$$L^c_{i,r,c} = L^{c,\text{initial}}_{r,c} + \sum_{\substack{j\in I,\\ j<i}} \sum_{v\in V_r^{\text{in}}} M^c_{j,v,c} - \sum_{\substack{j\in I,\\ j<i}} \sum_{v'\in V_r^{\text{out}}} M^c_{j,v',c} \qquad i\in I, r\in R, c\in C \tag{10}$$

The level of concentrate $c$ in unit $r$ follows the priority sequence, i.e. the level of $c$ in unit $r$ is changing from $L^{c,\text{initial}}_{r,c}$ to $L^c_{i1,r,c}, L^c_{i2,r,c}, \ldots, L^c_{|I|,r,c}$. However, it is worth noting that only $M^c_{i,v,c}$ follows the time representation for operation $v$ at slot $i$, and there are no time variables calculated for $L^c_{i,r,c}$. In other words, there is no specific time information about the level variables, but only the relative sequence of them.

A disaggregated formulation of the mass balance constraints is presented in Equation (10′). The level of $c$ in unit $r$ at slot $i+1$ equals the sum of the original level, the accumulation

terms and the consumption terms $M^c_{i,v,c}$ at slot $i$.

$$L^c_{i+1,r,c} = L^c_{i,r,c} + \sum_{v \in V^{\text{in}}_r} M^c_{i,v,c} - \sum_{v' \in V^{\text{out}}_r} M^c_{i,v',c} \qquad i \in I, r \in R, c \in C \tag{10'}$$

The final level $L^{c,\text{Final}}_{r,c}$ equals the sum of the level, the accumulation terms and the consumption terms at the last slot.

$$L^{c,\text{Final}}_{r,c} = L^c_{i,r,c} + \sum_{v \in V^{\text{in}}_r} M^c_{i,v,c} - \sum_{v' \in V^{\text{out}}_r} M^c_{i,v',c} \qquad i = |I|, r \in R, c \in C \tag{11}$$

Variable $L^t_{i,r}$ denotes the total level of concentrates in unit $r$ at slot $i$, and variable $M^t_{i,v}$ denotes the total amount of concentrates in operation $v$ at slot $i$. They equal the sum of corresponding variables at the concentrate level, respectively.

$$M^t_{i,v} = \sum_{c \in C} M^c_{i,v,c} \qquad i \in I, v \in V \tag{12}$$

$$L^t_{i,r} = \sum_{c \in C} L^c_{i,r,c} \qquad i \in I, r \in R \tag{13}$$

Similarly to Equation (10), $L^t_{i,r}$ can also be represented as the sum of initial total level $L^{t,\text{initial}}_r$, accumulation terms and consumption terms from previous slots. This constraint is a linear combination of Equation (11)-(13) and is redundant to the model, but it can help to reduce the search space of the mixed-integer linear programming (MILP) subproblem.

$$L^t_{i,r} = L^{t,\text{initial}}_r + \sum_{\substack{j \in I, \\ j<i}} \sum_{v \in V^{\text{in}}_r} M^t_{j,v} - \sum_{\substack{j \in I, \\ j<i}} \sum_{v' \in V^{\text{out}}_r} M^t_{j,v'} \qquad i \in I, r \in R \tag{14}$$

Mass balances at the element level are implicitly included in the model. Since concentrates are solids and their element compositions remain constant through the process, the mass balance for the amount of each element can be formulated as a linear combination of mass balances at the concentrate level. This is different from the crude oil scheduling problem, because crude oils are liquids and blending will introduce bilinear terms in the mass balance constraints.

*3.2.6. Sequencing constraints*

Different from other units in the process, bins and daily-arrival piles are allowed to be charged and discharged simultaneously, which makes the clique and Equations (2), (3), (8) non-applicable to inlet/outlet transfer operations of these units. Therefore, new constraints need to be specified for the sequences of these operations.

The main rule ensures that outlet transfer operations of a particular blend are not executed before the preceding inlet operations. In order to simplify the model, outlet transfer operations are assigned to one slot after inlet transfer operations.

$$Z_{i,v} \geq Z_{i+1,v'} \qquad i \in I, v \in V_r^{\text{in}}, v' \in V_r^{\text{out}}, r \in R_{\text{bin}} \cup R_{\text{dapile}} \tag{15}$$

Furthermore, the start time of outlet transfer operations should be no earlier than the inlet transfer operations.

$$S_{i,v} \leq S_{i+1,v'} + H \cdot (1 - Z_{i+1,v'}) \qquad i \in I, v \in V_r^{\text{in}}, v' \in V_r^{\text{out}}, r \in R_{\text{bin}} \cup R_{\text{dapile}} \tag{16}$$

In addition, the inventory at these units must be non-negative throughout the time horizon. At slot $i$, the total outlet amount until the end time of the outlet operation is bounded by the maximum potential inlet amount, which can be calculated from the upper bounds of incoming flowrates ($\bar{F}_{v'}$) and the end time of the outlet operation ($E_{i,v}$).

$$\sum_{\substack{j \leq i, \\ j \in I}} \sum_{v'' \in V_r^{\text{out}}} M_{j,v''}^t \leq \sum_{v' \in V_r^{\text{in}}} \bar{F}_{v'} \cdot E_{i,v} + M \cdot (1 - Z_{i,v}) \qquad i \in I, v \in V_r^{\text{out}}, r \in R_{\text{bin}} \cup R_{\text{dapile}} \tag{17}$$

*3.2.7. Composition constraints*

The proportions of different concentrates in a blend must remain consistent in the subsequent transfer operations. The fraction of a concentrate in the blend being transferred in operation $v$ can be represented as $\dfrac{M_{i,v,c}^c}{M_{i,v}^t}$; similarly, the fraction of a concentrate in the blend being accumulated in unit $r$ can be represented as $\dfrac{L_{i,r,c}^c}{L_{i,r}^t}$. The consistency of the proportions can be achieved by setting these two terms equal to each other for the units and their corresponding outlet transfer operations, so that these proportions stay the same along blending

units and the downstream operations.

$$\frac{M^c_{i,v,c}}{M^t_{i,v}} = \frac{L^c_{i,r,c}}{L^t_{i,r}} \qquad i \in I, v \in V^{\text{out}}_r, r \in R_{\text{blender}} \cup R_{\text{bin}}, c \in C \tag{18*}$$

To avoid singularity, the fractions can be reformulated as bilinear equations:

$$M^c_{i,v,c} \cdot L^t_{i,r} = L^c_{i,r,c} \cdot M^t_{i,v} \qquad i \in I, v \in V^{\text{out}}_r, r \in R_{\text{blender}} \cup R_{\text{bin}}, c \in C \tag{18}$$

*3.2.8. Capacity constraints*

If operation $v$ is assigned to slot $i$, then $M^t_{i,v}$ should be within its lower and upper bounds. Otherwise, $M^t_{i,v}$ should be zero.

$$\underline{M}^t_v \cdot Z_{i,v} \leq M^t_{i,v} \leq \bar{M}^t_v \cdot Z_{i,v} \qquad i \in I, v \in V \tag{19}$$

If operation $v$ is assigned to slot $i$, then the total flowrate of $v$ should be within its lower and upper bounds.

$$\underline{F}^t_v \leq \frac{M^t_{i,v}}{D_{i,v}} \leq \bar{F}^t_v \qquad i \in I, v \in V \tag{20*}$$

This constraint can be reformulated as the linear inequality (20). If operation $v$ is not assigned to slot $i$, both $D_{i,v}$ and $M^t_{i,v}$ must be zero.

$$\underline{F}^t_v \cdot D_{i,v} \leq M^t_{i,v} \leq \bar{F}^t_v \cdot D_{i,v} \qquad i \in I, v \in V \tag{20}$$

The total level in unit $r$ at slot $i$ and at the end of time horizon must be within its lower and upper bounds.

$$\underline{L}^t_r \leq L^t_{i,r} \leq \bar{L}^t_r \qquad i \in I, r \in R \tag{21}$$

$$\underline{L}^t_r \leq L^{\text{t,final}}_r \leq \bar{L}^t_r \qquad r \in R \tag{22}$$

*3.2.9. Element concentration constraints*

As stated in previous sections, there are several quality requirements in this process, both for the main products and for the byproducts. These requirements are formulated as linear constraints that set restrictions on the ratios of different key elements in final blend feeding the smelter.

The proportion of key elements in the final blend of concentrates transferred into the smelter is restricted as follows:

$$\frac{\sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}}{\sum_{v\in V_{\text{Final}}} M^t_{i,v}} \le \bar{f}_k \qquad i\in I, k\in K \tag{23*}$$

where $V_{\text{Final}}$ denotes the set of final operations that transfer concentrates from bins to the smelter, the numerator denotes the total amount of element $k$ in the final mixture, and $\bar{f}_k$ denotes the upper bounds of the fraction of element $k$ in the final mixture. This constraint can be rearranged as a linear inequality:

$$\sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c} \le \bar{f}_k \cdot \sum_{v\in V_{\text{Final}}} M^t_{i,v} \qquad i\in I, k\in K \tag{23}$$

*3.2.10. Interdependency constraints*

The ratio between two key elements $k$ and $k'$ can be bounded as follows:

$$\underline{m}_{k,k'} \le \frac{\sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k'}M^c_{i,v,c}}{\sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}} \le \bar{m}_{k,k'} \qquad i\in I,\ (k,k')\in KK' \tag{24*}$$

This constraint can be rearranged as a linear inequality:

$$\underline{m}_{k,k'}\cdot \sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c} \le \sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k'}M^c_{i,v,c} \le \bar{m}_{k,k'}\cdot \sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c} \qquad i\in I,\ (k,k')\in KK' \tag{24}$$

Additional constraints are necessary to guarantee that the weighted proportion of a certain element $k$ does not exceed its upper bound $\text{UE}_k$,

$$\frac{\sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}\cdot \text{KE}_k}{\sum_{v\in V_{\text{Final}}}\sum_{c\in C}\sum_{k'\in K} f_{c,k'}M^c_{i,v,c}\cdot \text{KE}_{k'}} \le \text{UE}_k \qquad i\in I, k\in K \tag{25*}$$

where $\text{KE}_k$ denotes the weighting coefficient for the elements. This constraint can also be rearranged as a linear inequality:

$$\sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}\cdot \text{KE}_k \le \text{UE}_k \cdot \left(\sum_{v\in V_{\text{Final}}}\sum_{c\in C}\sum_{k'\in K} f_{c,k'}M^c_{i,v,c}\cdot \text{KE}_{k'}\right) \qquad i\in I, k\in K \tag{25}$$

*3.2.11. Operation rules for daily-arrival materials*

The daily-arrival materials must be transferred to the daily-arrival piles continuously throughout the time horizon.

$$\sum_{i \in I} D_{i,v} = H \qquad v \in V_r^{\text{in}}, r \in R_{\text{dapile}} \tag{26}$$

The total inventory of daily-arrival materials at the end of time horizon is restricted. The sum of inventories of these materials in all the daily-arrival piles and bins should not exceed a given bound $\bar{L}_{\text{daily}}$.

$$\sum_{c \in C_{\text{da}}} \sum_{r \in R_{\text{dapile}} \cup R_{\text{bin}}} L_{r,c}^{\text{Final}} \leq \bar{L}_{\text{daily}} \tag{27}$$

*3.2.12. Operation rules for bins*

The concentrates in all bins must be transferred to the smelter simultaneously. Therefore, the start time and the duration of final transfer operations ($v \in V_{\text{Final}}$) should be the same in each slot $i$. If any final operation $v$ is assigned to slot $i$, then other final operations $v'$ must also be assigned to the same slot $i$. Otherwise, $S_{i,v'} = 0$ and other final operations $v'$ will not be assigned to slot $i$.

$$S_{i,v} - S_{i,v'} = 0 \qquad i \in I, v \in V_{\text{Final}}, v' \in V_{\text{Final}}, v \neq v' \tag{28}$$

$$D_{i,v} - D_{i,v'} = 0 \qquad i \in I, v \in V_{\text{Final}}, v' \in V_{\text{Final}}, v \neq v' \tag{29}$$

An additional constraint is necessary to ensure that the contribution of each bin feeding the smelter is maintained within a certain range.

$$\underline{r}_{\text{bin}} \leq \frac{M_{i,v}^t}{\sum_{v' \in V_{\text{Final}}} M_{i,v'}^t} \leq \bar{r}_{\text{bin}} \qquad i \in I, v \in V_{\text{Final}} \tag{30*}$$

This constraint can be rearranged as the following linear inequality:

$$\underline{r}_{\text{bin}} \cdot \sum_{v' \in V_{\text{Final}}} M_{i,v'}^t \leq M_{i,v}^t \leq \bar{r}_{\text{bin}} \cdot \sum_{v' \in V_{\text{Final}}} M_{i,v'}^t \qquad i \in I, v \in V_{\text{Final}} \tag{30}$$

*3.2.13. Operation rules for smelter*

The operation of the smelter is constrained by its processing capacity. In this context, the total flowrate of concentrates transferred to the smelter should be below its upper bound.

$$\frac{\sum_{v' \in V_{\text{Final}}} M^t_{i,v'}}{D_{i,v}} \leq \bar{F}_{\text{smelter}} \qquad i \in I, v \in V_{\text{Final}} \tag{31*}$$

This constraint can be rearranged as a linear inequality:

$$\sum_{v' \in V_{\text{Final}}} M^t_{i,v'} \leq \bar{F}_{\text{smelter}} \cdot D_{i,v} \qquad i \in I, v \in V_{\text{Final}} \tag{31}$$

*3.2.14. Symmetry breaking constraints*

A general symmetry breaking constraint can be applied to the MOS model (Mouret et al., 2011). The logic implies that an operation $v$ should not be assigned to $i$ if no other non-overlapping operations $v'$ are assigned to $i-1$. This is because if the slot $i-1$ is empty, then $v$ can be assigned directly to $i-1$ instead of $i$.

$$Z_{i,v} \leq \sum_{v' \in W'} Z_{i-1,v'} \qquad i \in I, i > 1, v \in W', v \neq v', W' \subset W \tag{32}$$

*3.2.15. Objective function*

The objective function is to maximize the gross margin of concentrates transferred to the smelter as presented in Equation (MOS),

$$\max \sum_{i \in I} \sum_{v \in V_{\text{Final}}} \sum_{c \in C} G_c M^c_{i,v,c} \tag{MOS}$$

where $G_c$ denotes the unit price of concentrate $c$ (\$/ton).

The complete formulation for the deterministic MOS model is obtained by maximizing the objective function (MOS) subject to constraints (2)-(32). This model is a nonconvex MINLP problem due to the bilinear constraint (18).

## 4. Robust MOS model

As mentioned earlier, there is inherent uncertainty in element composition of concentrates due to the upstream processes in which they are produced. This uncertainty is crucial to

the quality requirements of the smelting process modeled with linear constraints (23)-(25). If the MOS model is solved using the nominal values of element composition, there is a good chance that some of the quality requirements be violated, and the solution is no longer feasible. Therefore, it is necessary to extend the MOS model to gain robustness against the impact of variability in the composition of concentrates.

In our process, the element composition of concentrates is unknown through the entire time horizon. Therefore, a static robust optimization technique, i.e. robust counterpart, is suitable to account for the uncertainty by optimizing under its worst realization (Bertsimas and Sim, 2003). In most robust optimization approaches, the value of the uncertainty corresponding to the worst case is usually dualized and eliminated in explicit form. Such information is valuable for the robust model, as it can be used to verify the model validity based on our existing knowledge of the process. In this context, the flexibility test problem (Grossmann et al., 2014) can be included in the robust framework as a complement to reveal the values of element composition obtained in the robust counterpart.

Since there are multiple quality requirements corresponding to different products at different stages of the process, it becomes impossible to satisfy all of them simultaneously when the uncertainty range is too large, leading to violations in some of the quality requirements. In these cases, the traditional robust model cannot obtain feasible solutions, or determine which requirements would be violated under this circumstance. To deal with this situation, we build a bi-criterion robust MOS model that allows violations of quality requirements to keep the model feasible under high uncertainty, and to quantify the violations. The information of quantified violations should guide response actions in the real process.

In this section, we first introduce and derive the robust counterpart of our MOS model. Then, we present the flexibility test problem. Finally, we discuss the robust MOS model that allows constraint violations.

### *4.1. Robust counterpart*

The basic idea of the robust counterpart is to guarantee that the solution is feasible for any realization in the uncertainty set, which is enforced with a deterministic equivalent of

their worst case (Soyster, 1973). In order to avoid the solution to be overly conservative, a budget parameter $\Gamma$, developed by Bertismas and Sim (Bertsimas and Sim, 2003), can be adopted in the formulation to constrain the uncertainty set.

The robust counterpart formulation is a constraint-wise method, which implies that every constraint containing uncertainty is treated independently. Therefore, the robust counterpart formulation for each quality requirements (Equations (23)-(25)) needs to be derived separately. We use the element concentration constraint (23) to show the four steps involved in the derivation of the robust counterpart. The derivations for the other quality requirements (Equations (24)-(25)) are presented in Appendix A.

*4.1.1. Defining uncertainty set*

The uncertainty set is defined in Equation (1). The original set is a box uncertainty set, in which the worst case occurs when all uncertain parameters reach one of their extreme values at the same time. In the smelting process, this case corresponds to the situation in which the composition of every element in each concentrate deviates $\delta$ from its nominal value. This case is very pessimistic and it can cause the solution to be overly conservative. To avoid the extreme case, a budget parameter $\Gamma$ is introduced with the following idea: for certain element $k$, there can be at most $\Gamma$ concentrates whose composition of $k$ can reach the worst-case values simultaneously. By normalizing the deviation of $f_{c,k}$ as $p_{c,k}$, the budget parameter can be implemented in the uncertainty set by requiring that the sum of normalized deviations be less than $\Gamma$ for each $k$:

$$U = \left\{ \vec{F} \;\middle|\; \begin{array}{ll} \bar{f}_{c,k} \cdot (1 - \delta p_{c,k}) \leq f_{c,k} \leq \bar{f}_{c,k} \cdot (1 + \delta p_{c,k}) & c \in C, k \in K_U \\ 0 \leq p_{c,k} \leq 1 & c \in C, k \in K_U \\ 0 \leq \sum_{c \in C} p_{c,k} \leq \Gamma & k \in K_U \end{array} \right\} \tag{33}$$

By adjusting $\Gamma$, the range of the uncertainty is controlled. When $\Gamma = 0$, all parameters in the uncertainty set stay at their nominal values, and the model is equivalent to the deterministic model. When $\Gamma = |C|$, all parameters in the set can reach the worst-case values at the same time, and the model is equivalent to the classical robust formulation with the box uncertainty set.

*4.1.2. Building worst case*

The objective of the problem is to maximize the gross margin of the process, which is a linear function of the variables $M^c_{i,v,c}$ that are restricted by the element concentration constraint (23). Therefore, the worst case is that the variables in the objective functions ($M^c_{i,v,c}$) are most constrained by the uncertain parameters, resulting in a lower objective value.

The worst cases can be achieved by maximizing the positive terms on the left hand side of the inequality (23). The corresponding worst case is presented in Equation (34).

$$\max_{\vec{F}\in U}\left(\sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}\right) \le \bar{f}_k \cdot \sum_{v\in V_{\text{Final}}} M^t_{i,v} \qquad i\in I, k\in K_U \tag{34}$$

*4.1.3. Building auxiliary optimization problem*

The worst case in Equation (34) transforms the model into a bi-level optimization problem. To make the model tractable, the maximized term in the inequality above can be reformulated by building an auxiliary problem. Considering the maximization term in (34), the reformulation for every individual inequality ($i \in I$, $k \in K_U$) is given by (35):

$$\begin{aligned}
&\max_{\vec{F}\in U}\left(\sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}\right)\\
\equiv\ &\max_{\vec{F}\in U}\left[\sum_{v\in V_{\text{Final}}}\sum_{c\in C} \bar{f}_{c,k}(1+\delta p_{c,k})M^c_{i,v,c}\right]\\
\Leftrightarrow\ &\max \sum_{v\in V_{\text{Final}}}\sum_{c\in C} \bar{f}_{c,k}\cdot p_{c,k}M^c_{i,v,c}\\
&\text{s.t. } 0\le p_{c,k}\le 1 \qquad\qquad c\in C\\
&\sum_{c\in C} p_{c,k}\le \Gamma
\end{aligned} \tag{35}$$

*4.1.4. Dualizing auxiliary problem*

The auxiliary optimization problem can be dualized to transform the model back to a single-level problem. For the auxiliary problem (35), with subscripts $i \in I$, $k \in K_U$, the corresponding maximization problem is equivalent to:

$$\min \Gamma q_{i,k} + \sum_{c\in C} s_{i,c,k} \tag{36a}$$

$$\text{s.t. } s_{i,c,k} + q_{i,k} \geq \sum_{v \in V_{\text{Final}}} \bar{f}_{c,k} M^c_{i,v,c} \qquad c \in C \tag{36b}$$

$$s_{i,c,k} \geq 0 \qquad c \in C \tag{36c}$$

$$q_{i,k} \geq 0 \tag{36d}$$

where $s_{i,c,k}$ and $q_{i,k}$ are dual variables corresponding to the first and the second constraints of problem (35), respectively. By strong duality, the dual problem is bounded and the optimal objective of the dual problem equals the optimal objective value of the auxiliary problem (35).

Substituting the dualized problems into Equation (23), the model will automatically minimize $s_{i,c,k}, q_{i,k}$ from the dualized problem. The robust counterpart formulation is then given by Equation (37).

$$\left[\sum_{i \in I} \sum_{v \in V_{\text{Final}}} \sum_{c \in C} \bar{f}_{c,k} M^c_{i,v,c} + \delta \left( \Gamma q_k + \sum_{c \in C} s_{c,k} \right)\right] - \bar{f}_k \cdot \sum_{i \in I} \sum_{v \in V_{\text{Final}}} M^t_{i,v} \leq 0 \qquad i \in I, k \in K_U \tag{37}$$

295 Compared with the original constraint (23), the uncertainty is taken into consideration in the robust counterpart by adding the new terms with the dual variables ($q_{i,k}$ and $s_{i,c,k}$) in the original constraint and including new constraints from the dualized auxiliary problem.

Similarly, the robust counterparts for Equations (24) with subscripts $i \in I, (k, k') \in KK', k, k' \in K_U$ are:

$$\begin{aligned} \underline{m}_{k,k'} \cdot \left[\sum_{v \in V_{\text{Final}}} \sum_{c \in C} \bar{f}_{c,k} M^c_{i,v,c} + \delta \left( \Gamma q_{i,k} + \sum_{c \in C} s_{i,c,k} \right)\right] \\ \leq \left[\sum_{v \in V_{\text{Final}}} \sum_{c \in C} \bar{f}_{c,k'} M^c_{i,v,c} - \delta \left( \Gamma q_{i,k'} + \sum_{c \in C} s_{i,c,k'} \right)\right] \end{aligned} \tag{38}$$

$$\begin{aligned} \left[\sum_{v \in V_{\text{Final}}} \sum_{c \in C} \bar{f}_{c,k'} M^c_{i,v,c} + \delta \left( \Gamma q_{i,k'} + \sum_{c \in C} s_{i,c,k'} \right)\right] \\ \leq \bar{m}_{k,k'} \cdot \left[\sum_{v \in V_{\text{Final}}} \sum_{c \in C} \bar{f}_{c,k} M^c_{i,v,c} - \delta \left( \Gamma q_{i,k} + \sum_{c \in C} s_{i,c,k} \right)\right] \end{aligned} \tag{39}$$

The robust counterpart for Equation (25) with subscripts $i \in I$, $k \in K_U$ is:

$$\begin{aligned}
&\mathrm{KE}_k \cdot (1 - \mathrm{UE}_k) \cdot \left[ \sum_{v \in V_{\mathrm{Final}}} \sum_{c \in C} \bar{f}_{c,k} M^c_{i,v,c} + \delta \left( \Gamma q_{i,k} + \sum_{c \in C} s_{i,c,k} \right) \right] \\
&\qquad \leq \mathrm{UE}_k \cdot \left[ \sum_{v \in V_{\mathrm{Final}}} \sum_{c \in C} \sum_{\substack{k' \in K, \\ k' \neq k}} \bar{f}_{c,k'} M^c_{i,v,c} \cdot \mathrm{KE}_{k'} - \delta \left( \sum_{\substack{k' \in K_U, \\ k' \neq k}} \Gamma q'_{i,k,k'} + \sum_{c \in C} \sum_{\substack{k' \in K_U, \\ k' \neq k}} s'_{i,c,k,k'} \right) \right]
\end{aligned} \tag{40}$$

$$q'_{i,k,k'} + s'_{i,c,k,k'} \geq \sum_{v \in V_{\mathrm{Final}}} \bar{f}_{c,k'} \cdot M^c_{i,v,c} \cdot \mathrm{KE_{k'}} \qquad c \in C, k' \in K_U, k' \neq k \tag{41a}$$

$$s'_{i,c,k,k'} \geq 0 \qquad c \in C, k' \in K_U, k' \neq k \tag{41b}$$

$$q'_{i,k,k'} \geq 0 \qquad k' \in K_U, k' \neq k \tag{41c}$$

A detailed derivation of the robust counterparts above is provided in Appendix A. The entire robust counterpart formulation includes both the deterministic MOS model and the robust counterparts presented above:

$$\begin{aligned}
&\max \sum_{i \in I} \sum_{v \in V_{\mathrm{Final}}} \sum_{c \in C} G_c M^c_{i,v,c} \\
&\text{s.t. (2)-(32)} \\
&\qquad \text{(36b)-(36d), (37)-(40), (41a)-(41c)}
\end{aligned} \tag{RO-MOS}$$

*4.2. Flexibility test problem*

As presented above, the robust counterpart dualizes the uncertain parameters in the constraints to transform the bi-level optimization problems into single-level ones. In the final formulation, the uncertain parameter $f_{c,k}$ is replaced with dual variables $q_{i,k}, q'_{i,k,k'}, s_{i,c,k}, s'_{i,c,k,k'}$, and there are no direct ways to relate these new variables to values of the uncertain parameters. In order to reveal which parameters are selected to fluctuate in the robust solution, we introduce the flexibility test problem to complement the robust counterpart.

The flexibility test problem was developed to quantify the capability of a system to achieve feasible solutions by manipulating control/recourse variables. For a detailed review

on flexibility analysis, see Grossmann et al. (2014). Because the uncertain parameters are explicitly modeled as variables in this problem, it can be used to yield the value of uncertain parameters that are most likely to cause quality requirements to be violated for the given solution from the robust counterpart, which may correspond to the critical operating conditions in the real process. Its full formulation (FT) is given in Appendix B, in which the uncertain parameter $f_{c,k}$ is modeled as variables, and the quality requirements (23)-(25) are reformulated to check if there is any potential constraint violation with the uncertainty set (33) for the solution from the robust counterpart. The solution of (FT) gives the value of $f_{c,k}$ that corresponds to the worst case in the robust counterpart.

### *4.3. Robust model framework*

The robust framework is shown in Figure 7. First, the robust counterpart obtains the optimal solution of the scheduling problem considering the worst case uncertainty in the element compositions. Then, the values of the continuous variables $M^t_{i,v}, M^c_{i,v,c}$ are passed to the flexibility test problem to obtain the explicit values of element composition corresponding to the worst case. The first step is a nonconvex MINLP model. The second step is a small MILP model.

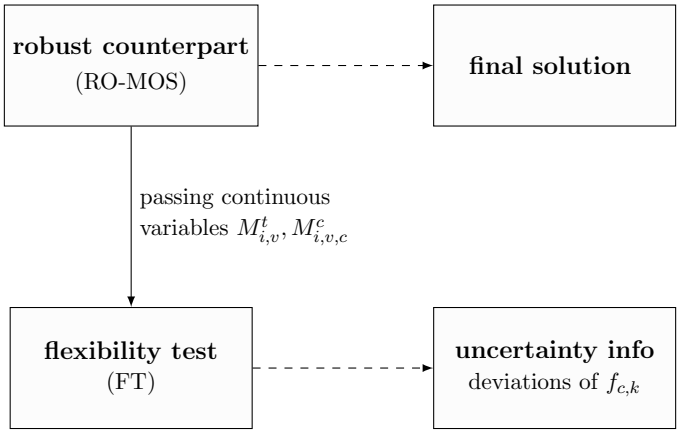


Figure 7: Robust model framework

### *4.4. Robust MOS model allowing constraint violations*

The robust model guarantees the feasibility of the solution while considering the worst case of the uncertainty. However, when the range of the uncertainty is large, it becomes unavoidable that the feasibility of the solution cannot be guaranteed no matter how the variables are adjusted. This issue can be demonstrated with the following motivating example.

In the interdependency constraint (24), the ratio of amount of element $k'$ represented by $Q'$ and amount of $k$ represented by $Q$, is set to be within certain bounds $[\underline{m}_{k,k'}, \bar{m}_{k,k'}]$. If the bounds are set as $[0.58, 0.64]$, and the element compositions for $k'$ fluctuate within $\pm 10\%$, then the interdependency constraint cannot be satisfied for the complete range of $Q'$. As shown in Figure 8, when $\frac{Q'}{Q} = 0.60$, the fluctuations of composition of $k$ will cause the actual value of $\frac{Q'}{Q}$ to vary between 0.54 and 0.66, violating the lower and upper bounds.

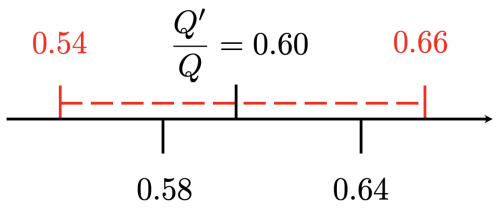


Figure 8: Illustration of constraint violation

For the given uncertainty range $\pm 10\%$ for $k'$, the fluctuation of the ratio is so large that any value within $[0.58, 0.64]$ will violate at least one of the bounds in the extreme cases. In other words, the violation of constraint (24) will always occur for the given uncertainty range in the robust model. When such a case occurs, the model will be infeasible, since there is no solution that satisfies both bounds for the ratio simultaneously. To make the problem tractable, and to quantify the violations, the robust counterpart can be further modified. We introduce a violation term, $\varepsilon$, to the robust counterpart (37)-(40) to allow violations of these terms:

$$\left[\sum_{i\in I}\sum_{v\in V_{\text{Final}}}\sum_{c\in C}\bar{f}_{c,k}M^{c}_{i,v,c} + \delta\left(\Gamma q_k + \sum_{c\in C} s_{c,k}\right)\right] - \bar{f}_k \cdot \sum_{i\in I}\sum_{v\in V_{\text{Final}}} M^{t}_{i,v} \leq \varepsilon^{1}_{i,k} \quad i \in I,\ k \in K_U \tag{42}$$

$$\underline{m}_{k,k'} \cdot \left[ \sum_{v \in V_{\text{Final}}} \sum_{c \in C} \bar{f}_{c,k} M^c_{i,v,c} + \delta \left( \Gamma q_{i,k} + \sum_{c \in C} s_{i,c,k} \right) \right]$$
$$\leq \left[ \sum_{v \in V_{\text{Final}}} \sum_{c \in C} \bar{f}_{c,k'} M^c_{i,v,c} - \delta \left( \Gamma q_{i,k'} + \sum_{c \in C} s_{i,c,k'} \right) \right] + \varepsilon^2_{i,k,k'} \qquad i \in I,\ (k,k') \in KK', k, k' \in K_U \tag{43}$$

$$\left[ \sum_{v \in V_{\text{Final}}} \sum_{c \in C} \bar{f}_{c,k'} M^c_{i,v,c} + \delta \left( \Gamma q_{i,k'} + \sum_{c \in C} s_{i,c,k'} \right) \right]$$
$$\leq \bar{m}_{k,k'} \cdot \left[ \sum_{v \in V_{\text{Final}}} \sum_{c \in C} \bar{f}_{c,k} M^c_{i,v,c} - \delta \left( \Gamma q_{i,k} + \sum_{c \in C} s_{i,c,k} \right) \right] + \varepsilon^3_{i,k,k'} \qquad i \in I,\ (k,k') \in KK', k, k' \in K_U \tag{44}$$

$$\text{KE}_k \cdot (1 - \text{UE}_k) \cdot \left[ \sum_{v \in V_{\text{Final}}} \sum_{c \in C} \bar{f}_{c,k} M^c_{i,v,c} + \delta \left( \Gamma q_{i,k} + \sum_{c \in C} s_{i,c,k} \right) \right]$$
$$\leq \text{UE}_k \cdot \left[ \sum_{v \in V_{\text{Final}}} \sum_{c \in C} \sum_{\substack{k' \in K, \\ k' \neq k}} \bar{f}_{c,k'} M^c_{i,v,c} \cdot \text{KE}_{k'} - \delta \left( \sum_{\substack{k' \in K_U, \\ k' \neq k}} \Gamma q'_{i,k,k'} + \sum_{c \in C} \sum_{\substack{k' \in K_U, \\ k' \neq k}} s'_{i,c,k,k'} \right) \right] + \varepsilon^4_{i,k}$$
$$i \in I, k \in K_U \tag{45}$$

Although unavoidable, the total violation of quality requirements should be reduced as much as possible so that the quality of products are less affected. This can be formulated as a second objective function:

$$\min \|\varepsilon\|_1 \tag{46}$$

where $\|\varepsilon\|_1$ stands for the $\ell_1$ norm of the violation terms,

$$\|\varepsilon\|_1 = \sum_{i \in I} \left[ \sum_{k \in K_U} \left( \varepsilon^1_{i,k} + \varepsilon^4_{i,k} \right) + \sum_{(k,k') \in KK'} \left( \varepsilon^2_{i,k,k'} + \varepsilon^3_{i,k,k'} \right) \right]$$

The new objective along with the modified quality requirements can be used to formulate a bi-criterion robust MOS model:

$$
\begin{cases}
\max \sum_{i \in I} \sum_{v \in V_{\text{Final}}} \sum_{c \in C} G_c M^c_{i,v,c} \\
\min \|\varepsilon\|_1 \\
\text{s.t. (2)-(32)} \\
\quad \text{(36b)-(36d), (41a)-(41c), (42)-(45)}
\end{cases}
$$

The bi-criterion model can be reformulated as a single-objective problem using the $\varepsilon$-constraint method (Cohon, 1978):

$$
\begin{aligned}
&\max \sum_{i \in I} \sum_{v \in V_{\text{Final}}} \sum_{c \in C} G_c M^c_{i,v,c} \\
&\text{s.t. } \|\varepsilon\|_1 \leq w \cdot \|\bar{\varepsilon}\|_1 \\
&\quad \text{(2)-(32)} \\
&\quad \text{(36b)-(36d), (41a)-(41c), (42)-(45)}
\end{aligned}
\qquad \text{(BC-RO-MOS)}
$$

in which $\|\bar{\varepsilon}\|_1$ is the maximal value of $\|\varepsilon\|_1$ that can be obtained by solving the bi-criterion model only with the objective of maximizing the gross margin, and $w$ is the coefficient that determines the upper bound of $\|\varepsilon\|_1$. It is obvious that the gross margin is positively correlated to to the violation term $\|\varepsilon\|_1$, as the smaller violation indicates that the solution is more restricted by the quality requirements.

By introducing $\varepsilon$, the bi-criterion robust model is capable of handling large range of uncertainty by relaxing the quality requirements. In addition, a Pareto-front between the gross margin and the violations can be obtained by adjusting the value of $w$, indicating the trade-offs that can be achieved between these two criteria.

### *4.5. Remarks*

1. The constraints to which the robust counterpart formulation is applied are all linear. Although the deterministic MOS model is an MINLP model, the quality requirements where the robust counterpart is implemented are all linear constraints.

2. The robust counterpart formulation does not introduce any integer variables or nonlinearity. The increase of model size is moderate, since it only introduces dual variables,

whose sizes are $O(|I| \cdot |C| \cdot |K_U|^2 + |I| \cdot |C| \cdot |KK'|)$.

3. The $\ell_1$ norm of the violation terms can be further modified to weigh each violation term to indicate the total economic cost of dealing with these violations.

## 5. Solution strategy

### *5.1. Two-step MILP-NLP decomposition strategy*

Solving the robust counterpart formulation directly using MINLP solvers is computationally expensive due to its size. In previous works (Mouret et al., 2009, 2011; Song et al., 2018), a two-step MILP-NLP decomposition strategy was developed to tackle such problems. Its scheme is outlined in Figure 9.

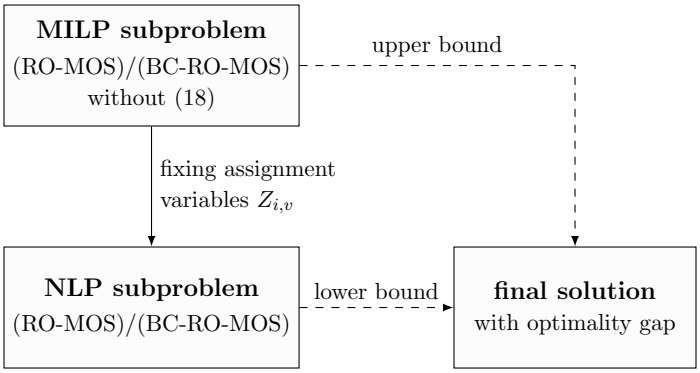


Figure 9: MILP-NLP decomposition strategy

The original MINLP problem ((RO-MOS) or (BC-RO-MOS)) is decomposed into an MILP subproblem and an nonlinear programming (NLP) subproblem. The MILP subproblem includes all the constraints except the nonlinear composition constraints (18). The NLP subproblem includes all the constraints, with assignments variable $Z_{i,v}$ fixed. The MILP subproblem is solved by CPLEX. The NLP subproblem is solved by the local NLP solver CONOPT, with fixed $Z_{i,v}$.

Because the MILP subproblem is a relaxation of the original maximization problem, its optimal objective value yields an upper bound for the original problem. With the binary variables fixed, the locally optimal objective value of the NLP subproblem yields a lower

bound for the original problem. The final solution is the solution of the NLP subproblem. The method provides an optimality gap, which is the relative difference between objective values of MILP and NLP solutions.

It is worth noting that feasibility and optimality of the final solution are not guaranteed with this approach. By passing the fixed assignment variables, the sequences of operations determined in the MILP subproblem is fixed in the NLP subproblem. These sequences may be infeasible to the NLP subproblem if the nonlinear composition constraints (18) cannot be satisfied. Also, since CONOPT is a local NLP solver, the NLP solution may not be the global optimum. Even if it is the global optimum to the NLP subproblem, it may not be the global optimum to the MINLP problem. Nevertheless, this strategy tends to give solutions with small or no optimality gaps with short computational time (Mouret et al., 2009, 2011; Song et al., 2018).

### *5.2. Enhancement of decomposition strategy*

The original MILP-NLP decomposition strategy can be further enhanced to reduce the optimality gaps. By comparing the MILP subproblem and the NLP subproblem in Figure 9, the presence of concentration constraints (18) is the main difference between them. Therefore, a set of linear constraints are formulated to model in a linear fashion some of the characteristics on NLP-feasible blends implied by composition constraints (18). Excluding the concentration constraints from the MILP subproblem has two effects in the operations leaving blending units: (1) inconsistent proportions of different concentrates, (2) inconsistent concentrate types in concentrate blends. These effects are illustrated in the following example.

The schedule of the MILP solution of a small case is shown in Figure 10. In this schedule, there are 2 batches of $c_1$ and 2 batches of $c_2$ at pile 1 and pile 2, respectively. Both batches are assumed to have the same size. All batches of $c_1$ and $c_2$ are transferred into the blending unit at slot $i_2$. Since the composition constraint is absent, the transfer of the blend from the blending unit to the smelter can be conducted as shown in Figure 10: 2 batches of $c_1$ and 1 batch of $c_2$ are transferred at slot $i_4$; later, 1 single batch of $c_2$ transferred at slot $i_6$. In this

way, 2 different blends of concentrates (highlighted in blue) are transferred to the smelter in two rounds; we assume that both of them satisfy quality requirements (23)-(25).

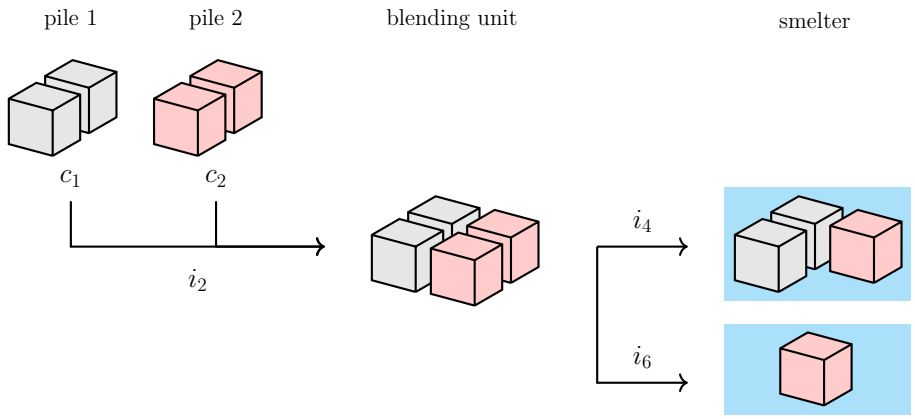


Figure 10: MILP solution

When the schedule of the MILP solution in the previous example is passed to the NLP subproblem, the corresponding NLP schedule is shown in Figure 11. Here, the composition constraints force the proportions of both blends to be the same, changing the amount of $c_1$ and $c_2$ in each blend (highlighted in red). These changes are likely to violate the quality requirements, which are satisfied in the original MILP solution.

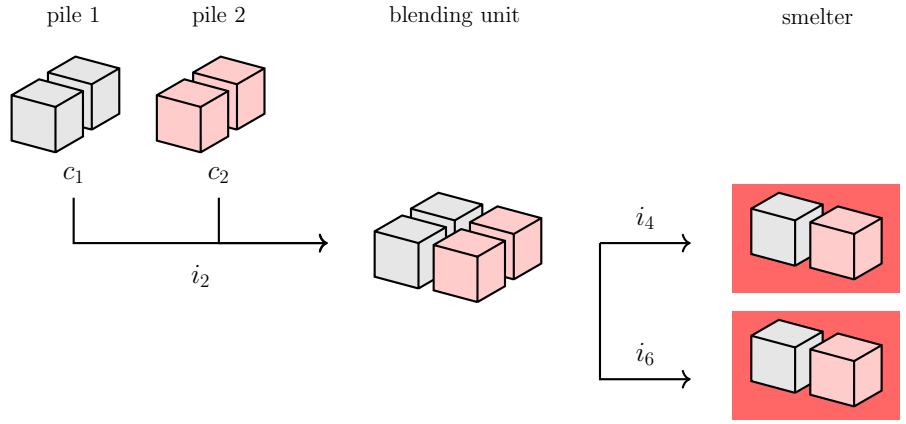


Figure 11: Scheme of inferior NLP solution

One way to enhance the decomposition strategy is to maintain the concentrate types before and after blending to eliminate improper MILP solutions like in Figure 10. We

introduce a new binary variable $Y_{i,v,c}$ to indicate if concentrate $c$ is transferred in operation $v$ in slot $i$:

$$-U \cdot (1 - Y_{i,v,c}) + \varepsilon \leq M^c_{i,v,c} \leq U \cdot Y_{i,v,c} \qquad i \in I, v \in V, c \in C \tag{47}$$

in which $\varepsilon$ is a sufficiently small parameter to enforce positive values.

The new binary variable has the following relationship with the original binary variable $Z_{i,v}$:

$$Z_{i,v} \geq Y_{i,v,c} \qquad i \in I, v \in V, c \in C \tag{48}$$

The concentrate types arriving and leaving the blending unit can be related by specifying $Y_{i,v,c}$ for these two parts. There are various ways to construct such relationships. Here, we assume that the transfer of blends to the smelter occurs exactly 2 slots after the transfer of concentrates to the blending unit, because "2 slots" is the minimum slots needed to transfer blends from the blending unit (blending unit $\rightarrow$ bins $\rightarrow$ smelter). Then the concentrate types in the former operations should be identical to those in the latter operations. This relationship can be formulated in aggregated form as follows:

$$\sum_{\substack{i' \in I, \\ i' \leq i-2}} \sum_{v' \in V^{\text{in}}_r} \sum_{r \in R_{\text{blender}}} Y_{i',v',c} = \sum_{\substack{i'' \in I, \\ i'' \leq i}} Y_{i'',v,c} \qquad i \in I, i > 2, v \in V_{\text{Final}}, c \in C \setminus C_{\text{da}} \tag{49}$$

We also assume that the amount of each concentrate, $M^c_{i,v,c}$, is the same between transfer of concentrates to the blending unit at slot $i$ and the transfer of blends to the smelter at $i + 2$:

$$\sum_{\substack{i' \in I, \\ i' \leq i-2}} \sum_{v' \in V^{\text{in}}_r} \sum_{r \in R_{\text{blender}}} M^c_{i',v',c} = \sum_{\substack{i'' \in I, \\ i'' \leq i}} \sum_{v \in V_{\text{Final}}} M^c_{i'',v,c} \qquad i \in I, i > 2, c \in C \setminus C_{\text{da}} \tag{50}$$

After applying these new constraints, the previous example will have a new MILP solution (Figure 12), in which the sole $c_2$ mixture transferred to the smelter at slot $i_6$ leads to an additional transfer of $c_2$ to the blending unit at slot $i_4$. The new MILP solution will have exactly the same corresponding NLP solution since there is no inconsistent proportion, thus eliminating the optimality gap.

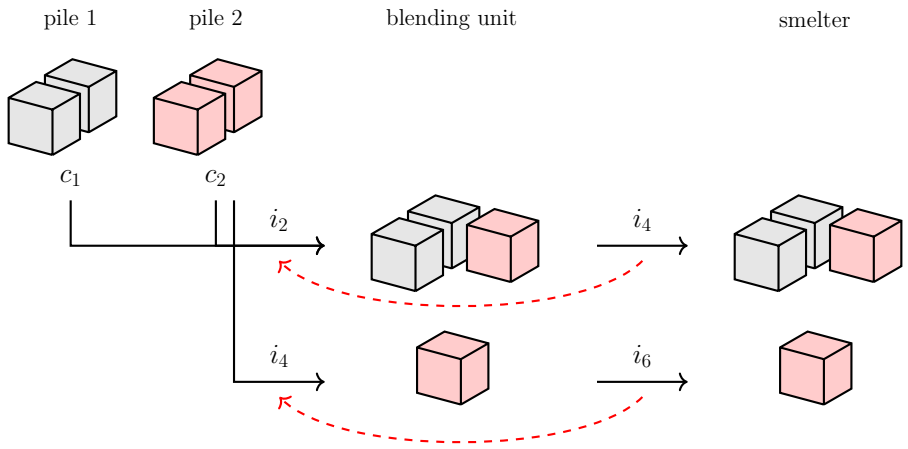


Figure 12: Scheme of MILP solution with new assignment constraints

For daily-arrival materials, a similar constraint can be formulated to maintain their amounts between inlet and outlet operations of bins. We assume that the transfer of daily-arrival materials to the smelter is exactly one slot after the transfer to the bins (minimum slots needed), then the amount of $c$ transferred to the bins at $i$ should be no less than its amount transferred to the smelter at $i+1$:

$$\sum_{\substack{i' \in I, \\ i' \leq i-1}} \sum_{v \in V_r^{\text{out}}} \sum_{r \in R_{\text{dapile}}} M^c_{i',v,c} \geq \sum_{\substack{i'' \in I, \\ i'' \leq i}} \sum_{v \in V_{\text{Final}}} M^c_{i'',v,c} \qquad i \in I, i > 1, c \in C_{\text{da}} \tag{51}$$

The new assumptions for constraints (49)-(51) together imply a specific transfer pattern for the blending unit: the unit can be filled with new concentrates only when it is empty. As the same amount of concentrates are required to be discharged from the blending unit to the smelter, there will be no concentrate left in the blending unit after the outlet operations and before the next inlet operations. This pattern eliminates some possible ways of transferring blends from the blending unit to the smelter. This may affect the objective value for the MILP subproblem, but will reduce optimality gap and the computational time (see Section 7).

The scheme of the enhanced solution strategy is outlined in Figure 13 where the new linear constraints (47)-(51) are added to the MILP subproblem.

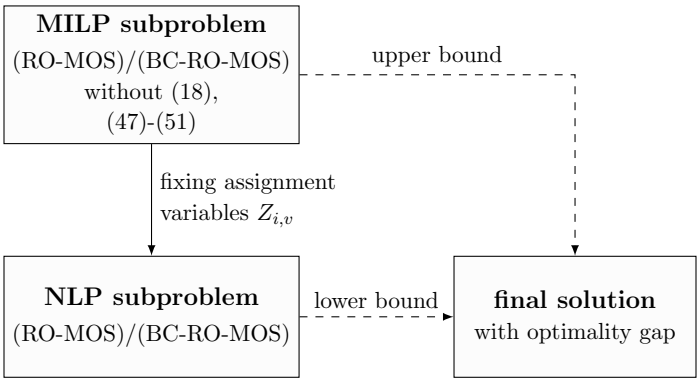


Figure 13: Enhanced MILP-NLP decomposition strategy

## 6. Case studies

We address an illustrative example and an industrial case study to evaluate the performance of the MOS model, the robust formulations and the decomposition strategies. The topology of the illustrative example is presented in Figure 14. This example considers 6 types of concentrates and 4 key elements within a 10-day horizon. The process consists of 2 maritime vessels, 5 piles at the port, 1 blending unit, 1 daily-arrival pile, 3 bins, and 1 smelter. To simplify the model, interdependency constraints (24) and (25) are excluded from the instance. The corresponding data of the process and the concentrates (profit, element compositions, etc.) are presented in Tables C.1 and C.2 in Appendix C. The set of cliques $W$ is shown in Figure C.1 in Appendix C.

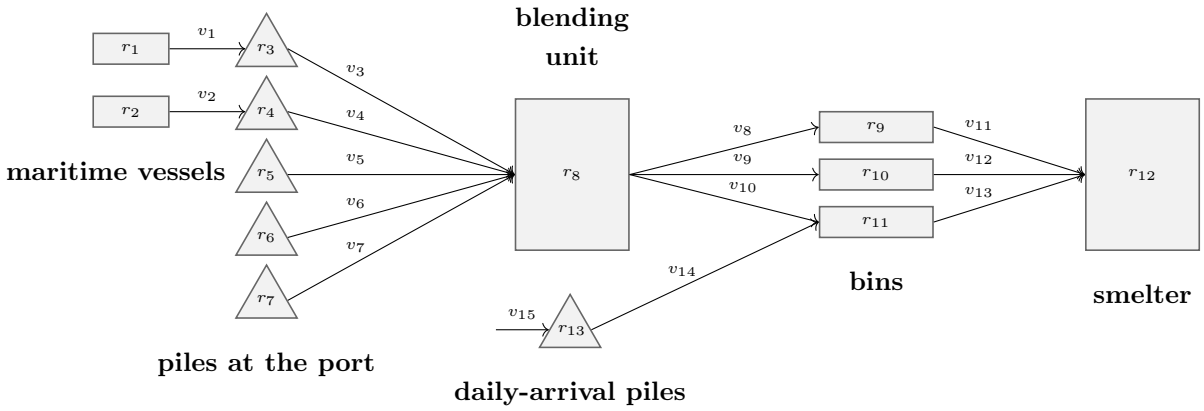


Figure 14: Topology of the illustrative example

The topology of the industrial case study is shown in Figure 15. The industrial case study considers 14 types of concentrates with 9 key elements and a time horizon of 15 days. The process consists of 5 maritime vessels, 11 piles at the port, 1 blending unit, 3 daily-arrival piles, 6 bins, and 1 smelter. All daily-arrival materials in piles $r_{25}$, $r_{26}$ and $r_{27}$ can be discharged to bins $r_{21}$, $r_{22}$, and $r_{23}$. The data for the process and the set of cliques are presented in Tables C.3 and C.4 and Figure C.2 in Appendix C.

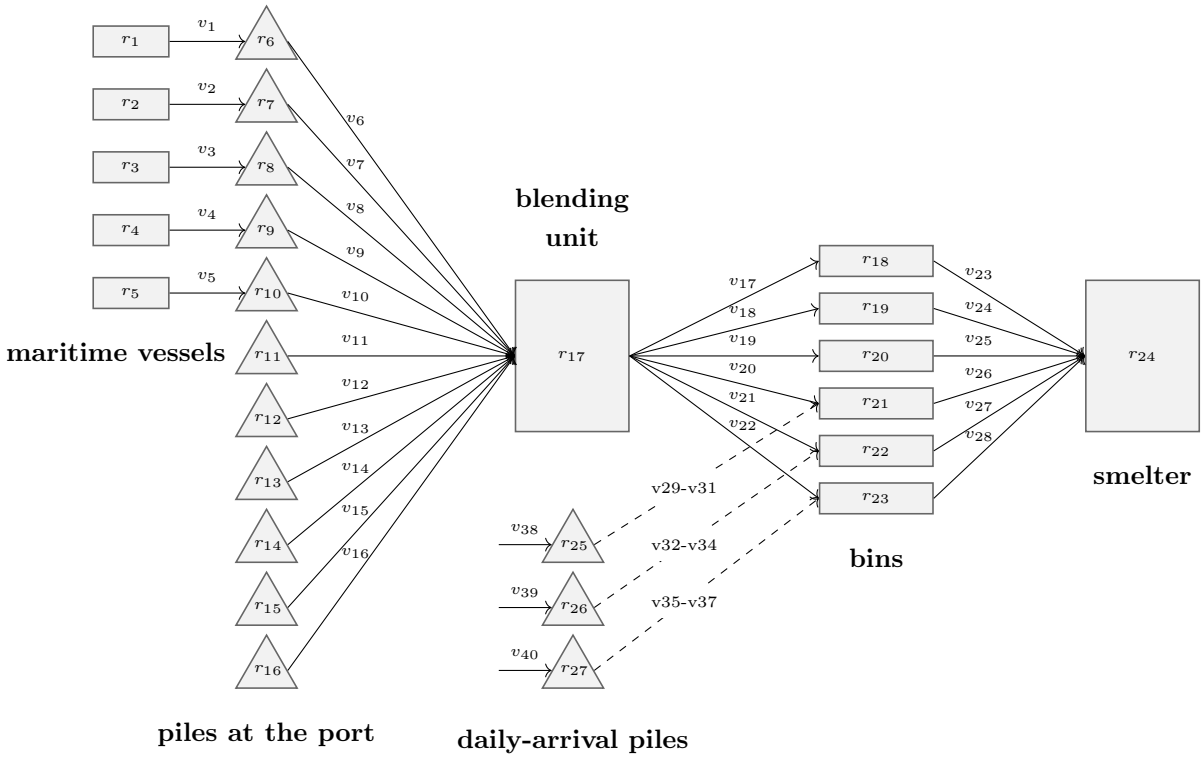


Figure 15: Topology of the industrial case study

Different ranges of uncertainty are set to show the effectiveness of the robust MOS model ((RO-MOS)- (FT)) and the bi-criterion robust model (BC-RO-MOS). The relative bounds for the uncertainty are set to be $\pm 1\%, \pm 2\%, \pm 3\%, \pm 5\%$ and $\pm 10\%$ of the nominal mass fraction values. For the illustrative example, the uncertainty elemental set is $K_U = \{k_1, k_2\}$ and the budget parameter $\Gamma$ is set to 2. For the industrial case study, $K_U = \{k_1, k_2, k_3\}$ and $\Gamma$ is set to 5. The equations involved in each case and each model are shown in Table 1.

The two cases are solved on an Intel Core i7 (6 cores) 2.60 GHz processor with 16 GB

Table 1: Equations for each case and each model

| | illustrative example | industrial case study |
|---|---|---|
| deterministic MOS model | (2)-(23), (26)-(MOS) | (2)-(MOS) |
| robust MOS model | (2)-(23), (26)-(MOS), (37), (36b)-(36d) | (2)-(MOS), (37), (38), (39), (40), (36b)-(36d), (41a)-(41c) |
| bi-criterion robust MOS model | (2)-(23), (26)-(MOS), (42), (36b)-(36d) | (2)-(MOS), (42), (43), (44), (45), (36b)-(36d), (41a)-(41c) |

RAM, with GAMS 27.2.0 (GAMS Development Corporation, 2019) as the modeling system, CPLEX 12.9 as the MILP solver, CONOPT 4.12 as the local NLP solver. The relative tolerances for CPLEX and CONOPT are set to 0.001 and 0 respectively. The number of priority slots in all models is set to 10.

## 7. Computational results

### *7.1. Illustrative example*

The deterministic MOS model is first solved for the illustrative example with all element compositions of concentrates at their nominal values. The statistics of the model with both original and enhanced solution strategies are given in Table 2.

Table 2: Model statistics for the deterministic MOS model of the illustrative example

| # | | continuous variables | discrete variables | equations |
|---|---|---|---|---|
| original solution strategy | MILP subproblem | 2,571 | 144 | 5132 |
| | NLP subproblem | 2,571 | - | 5,492 |
| enhanced solution strategy | MILP subproblem | 2,571 | 1,044 | 7,322 |
| | NLP subproblem | 2,571 | - | 7,682 |

The model using the enhanced solution strategy has a significantly larger number of discrete variables as a result of introducing the binary variable $Y_{i,v,c}$. The number of equations also increases, as the linear constraints (47)-(51) are added to reduce the optimality

gap. However, larger model size does not necessary lead to longer computational time (see Table 3).

Table 3: Computational results for the deterministic MOS model of the illustrative example

| solution strategy | enhanced | original | BARON |
|---|---|---|---|
| MILP obj. value (k$) | 9924.198 | 10535.624 | 9383.527 |
| NLP obj. value (k$) | 9924.198 | 9837.614 | |
| optimality gap | 0.00% | 7.10% | - |
| MILP time (s) | 3.199 | 9.363 | - |
| NLP time (s) | 0.603 | 1.67 | - |
| total time (s) | 3.802 | 11.033 | 3000.860 |

The enhanced solution strategy yields the same objective values in the MILP and NLP subproblems. The original solution strategy obtains a higher objective value in the MILP subproblem, but its objective value in the NLP subproblem is lower than the one from the enhanced solution strategy, with a 7.10% optimality gap. The MILP subproblems takes most of the computational time in both cases. However, the MILP and NLP subproblems using the enhanced solution strategy are faster to solve compared with the ones using the original solution strategy. The global MINLP solver BARON is also used to directly solve the model. It reaches $9383.527k after 3000 s, which is inferior to the results obtained with the proposed MILP-NLP decomposition strategies.

The lower objective value in the MILP subproblem for the enhanced solution strategy is due to the transfer pattern implied by the added constraints (49)-(51). This specific pattern excludes some solutions from the MILP subproblem with a higher objective value, which is obtained in the original solution strategy. However, such solutions may have a larger optimality gap and a lower value for the NLP subproblem.

The model statistics for the robust MOS model using both solution strategies are given in Table 4. The number of continuous variables and equations in the model is slightly higher than the ones in the deterministic model because of the addition of the robust counterparts. The flexibility test problem is a small MILP problem compared with the main MOS model.

Table 4: Model statistics for the robust MOS model of the illustrative example

| # | | continuous variables | discrete variables | equations |
|---|---|---|---|---|
| original solution strategy | MILP subproblem | 2,733 | 144 | 5,274 |
| | NLP subproblem | 2,733 | - | 5,634 |
| enhanced solution strategy | MILP subproblem | 2,733 | 1,044 | 7,464 |
| | NLP subproblem | 2,733 | - | 7,824 |
| flexibility test | | 129 | 40 | 157 |

The computational results for the robust model are presented in Table 5.

Table 5: Computational results for the robust MOS model of the illustrative example

| $\delta$ | solution strategy | obj. value (k$) | optimality gap | total time (s) |
|---|---|---|---|---|
| 0 (deterministic) | enhanced | 9924.198 | 0.00% | 3.802 |
| 0.01 | | 9826.972 | 0.00% | 5.413 |
| 0.02 | | 9740.942 | 0.00% | 1.665 |
| 0.03 | | 9656.447 | 0.00% | 5.165 |
| 0.05 | | 9492.017 | 0.00% | 5.357 |
| 0.1 | | 9114.389 | 0.00% | 5.228 |
| 0 (deterministic) | original | 9837.614 | 7.10% | 11.033 |
| 0.01 | | 9707.546 | 7.39% | 8.066 |
| 0.02 | | 9638.452 | 7.11% | 10.627 |
| 0.03 | | 9429.985 | 8.42% | 14.136 |
| 0.05 | | 9424.313 | 6.47% | 11.673 |
| 0.1 | | 8896.029 | 7.93% | 13.005 |

The enhanced solution strategy manages to obtain objective values with zero-gaps for all cases with different values of $\delta$. The original solution strategy achieves the objectives with optimality gaps around 7%. The enhanced solution strategy also yields smaller computational time for all cases compared with the original solution strategy. The MILP subproblems

still take most of the total computational time, whereas the flexibility test problem usually takes less than 0.1 s to solve. The objective profile is outlined in Figure 16. The enhanced solution strategy always yields a higher gross margin than the original solution strategy. In addition, the gross margin tends to decrease monotonically with the increase of the given uncertainty bounds, because the higher uncertainty makes the feasible region of the model smaller.

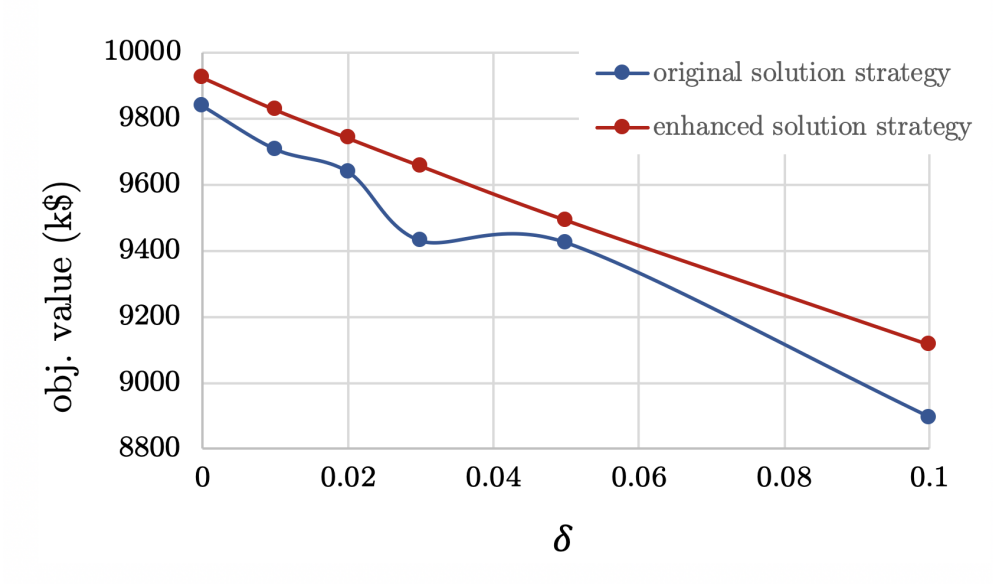


Figure 16: Objective value profile for the robust MOS model of illustrative example

The values of element composition obtained from the flexibility test problem for the case with $\delta = 0.1$ are given in Table 6 in the form of deviation from their nominal values. From the table, it is clear that the worst case indicated in the robust model corresponds to concentration of $k_1$ increasing in $c_2$ and $c_6$, together with a drop in the concentration of $k_2$ in $c_4$ and $c_5$. The number of deviations in each column equals 2, corresponding to the value of the budget parameter $\Gamma$.

### *7.2. Industrial case*

The robust MOS model is first solved for the industrial case using the enhanced solution strategy. The model statistics are given in Table 7. The model is much larger than the

Table 6: Deviation of element composition for $\delta = 0.1$ in the illustrative example

| | $k_1$ | $k_2$ |
|---|---|---|
| $c_1$ | 0 | 0 |
| $c_2$ | 0.070 | 0 |
| $c_3$ | 0 | 0 |
| $c_4$ | 0 | -0.037 |
| $c_5$ | 0 | -0.026 |
| $c_6$ | 0.021 | 0 |

illustrative example, as more concentrates and operations are involved. The computational results are presented in Table 8.

Table 7: Model statistics for the robust MOS model of the industrial case

| # | continuous variables | discrete variables | equations |
|---|---|---|---|
| MILP subproblem | 13,485 | 5,986 | 33,308 |
| NLP subproblem | 13,485 | - | 34,988 |
| flexibility test | 653 | 200 | 788 |

Similar to the results of the robust model for the illustrative example, for $\delta = 0 \sim 0.03$, the objective value decreases monotonically with the increase of $\delta$. Interestingly, the enhanced solution strategy also yields zero-gaps for these cases. The uncertain case with $\delta = 0.01$ has a significantly larger computational time compared with the deterministic case. However, as $\delta$ increases, the computational time drops monotonically. This may be because the introduction of robust counterparts makes the solution process much harder at first; then, the larger $\delta$ requires the solution to be more conservative to be robust to larger ranges of uncertainty, leading to smaller feasible regions and shorter computational time. The cases with $\delta = 0.05$ and 0.1 are infeasible, indicating there are unavoidable violations in some of the quality requirements. In these cases, the robust MOS model cannot obtain any feasible solutions.

Table 8: Computational results for the robust MOS model of the industrial case

| $\delta$ | final obj. value (k$) | optimality gap | total time (s) |
|---|---|---|---|
| 0 (deterministic) | 15752.041 | 0.00% | 827.597 |
| 0.01 | 15685.341 | 0.00% | 4264.045 |
| 0.02 | 15615.281 | 0.00% | 2170.918 |
| 0.03 | 15550.795 | 0.00% | 801.466 |
| 0.05 | infeasible | - | 32.883 |
| 0.1 | infeasible | - | 5.525 |

The case with $\delta = 0.1$ is then solved with the bi-criterion robust MOS model. The two extreme values of the case (maximal gross margin and minimal violation) are first calculated (Table 9). The upper bound of the gross margin is the same as the objective value of the deterministic case. The minimal violation is 25.128, which explains the infeasibility of the robust MOS model. Then, a set of inner points are calculated, forming the Pareto-front shown in Figure 17. The Pareto-front shows the trade-off between gross margin and constraint violations.

Table 9: Extreme points for the bi-criterion robust MOS model of the industrial case

| gross margin (k$) | violation $\|\varepsilon\|_1$ | total computational time (s) |
|---|---|---|
| 15752.041 | 652.506 | 1378.944 |
| 1141 | 25.128 | 136.907 |

The Gantt chart for the case with $\delta = 0.03$ is shown in Figure 18; here, the x-axis represents time and the y-axis represent the operations. For each operation, the bars with different colors represents executions of the same operation at different priority slots. From the chart, the non-overlapping features are clearly shown for operations V6-V16 against V17-V22. Here, V6-V16 are the inlet operations for the blending unit and V17-V22 are its outlet operations. The inlet operations are only assigned to slots with odd indexes, whereas the outlet operations are only assigned the slots with even indexes. These two operation

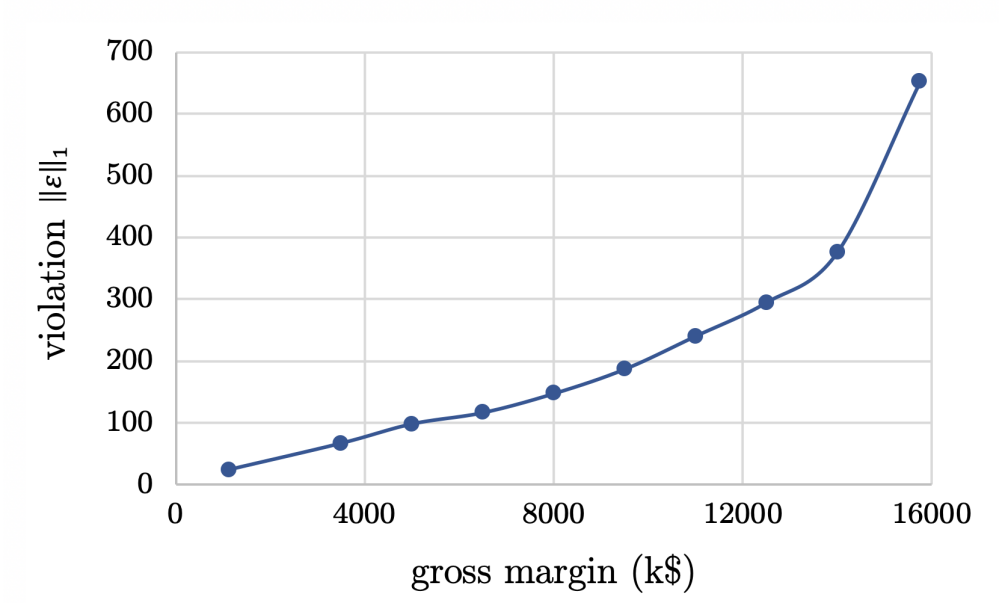


Figure 17: Pareto front for bi-criterion robust MOS model of the industrial case

sets never overlap with each other. The final operations V23-V29 also indicate that there are four rounds of blends being transferred to the smelter during the 15-day horizon.

## 8. Conclusions

In this work, we have addressed the optimal scheduling of logistic and blending operations for copper smelting process so that the gross margin associated to the concentrates being processed in the smelter are maximized. This problem is very difficult to solve due to the complexity of the process operations rules and the quality requirements for products, which are strongly affected by the uncertainty in element composition. We apply the MOS model to formulate the problem as a nonconvex MINLP problem, and we include a detailed explanation of its time representation. Based on the MOS model, we have developed a robust model combining the robust counterpart formulation and the flexibility test problem. We also present a bi-criterion robust model to allow constraint violations in the robust model. In order to solve the resulting MINLP model, we adopted and enhanced a two-step MILP-NLP decomposition strategy that significantly reduces the optimality gap.

The deterministic MOS model, the robust MOS model and the bi-criterion robust model

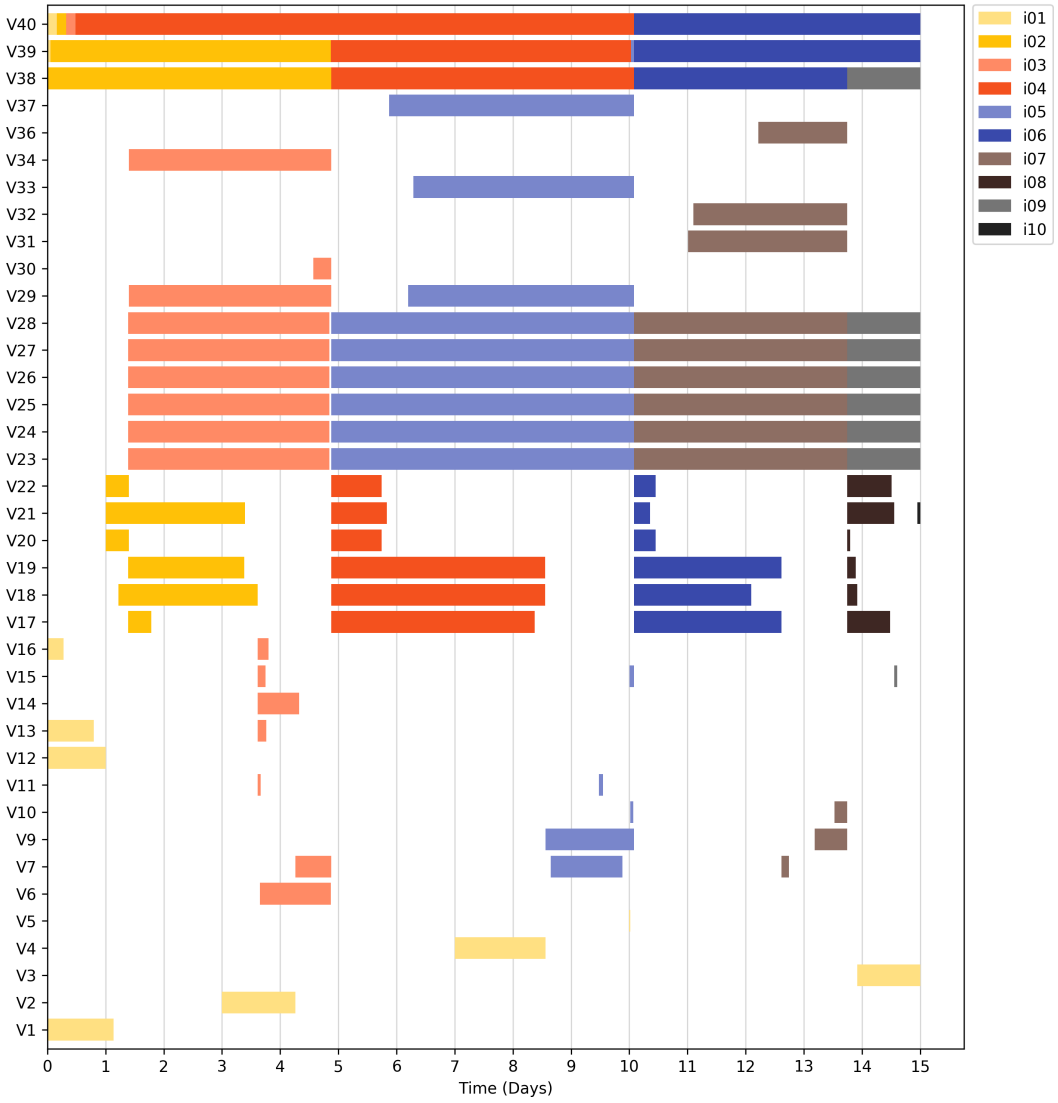


Figure 18: Gantt chart for the robust MOS model ($\delta = 0.03$) of the industrial case

are solved for an illustrative example and an industrial case with different ranges of uncertainty. The enhanced MILP-NLP decomposition strategy is shown to be superior to the original strategy, both in terms of computational time and solution quality. The robust MOS model is shown to be capable of obtaining optimal solutions that are guaranteed to satisfy quality requirements when the uncertainty is not too large. The explicit value of

element compositions can be obtained from the flexibility test problem. When the range of the uncertainty is large, the robust MOS model cannot obtain feasible solutions, while the bi-criterion robust model is able to obtain feasible solutions and determine the Pareto-front between the gross margin and total constraint violations.

**Acknowledgement**

The authors thank the support from the Research, Development & Innovation Department at Aurubis AG and the Center for Advanced Process Decision-making at Carnegie Mellon University for their involvement in the collaboration project.

## Appendix A. Derivation of robust counterpart

The robust counterpart formulation is constraint-wise, i.e. it is different for every constraint containing uncertainty based on the uncertain terms. The derivation of robust counterparts for constraints (24) and (25) are given here.

### *Appendix A.1. Defining uncertainty set*

This step is the same as the one shown in the paper. By introducing the budget parameter, the uncertainty set can be modified as:

$$U = \left\{ \vec{F} \;\middle|\; \begin{array}{ll} \bar{f}_{c,k} \cdot (1 - \delta p_{c,k}) \leq f_{c,k} \leq \bar{f}_{c,k} \cdot (1 + \delta p_{c,k}) & c \in C, k \in K_U \\ 0 \leq p_{c,k} \leq 1 & c \in C, k \in K_U \\ 0 \leq \sum_{c \in C} p_{c,k} \leq \Gamma & k \in K_U \end{array} \right\} \tag{33}$$

### *Appendix A.2. Building worst cases*

The worst cases can be obtained by maximizing the positive terms and minimizing the negative terms on the left hand side of the constraints. The corresponding worst cases are:

$$\underline{m}_{k,k'} \cdot \max_{\vec{F} \in U} \left( \sum_{v \in V_{\text{Final}}} \sum_{c \in C} f_{c,k} M^c_{i,v,c} \right) \leq \min_{\vec{F} \in U} \left( \sum_{v \in V_{\text{Final}}} \sum_{c \in C} f_{c,k'} M^c_{i,v,c} \right) \qquad i \in I,\ (k,k') \in KK', k, k' \in K_U \tag{A.1}$$

$$\max_{\vec{F}\in U}\left(\sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k'}M^c_{i,v,c}\right) \le \bar{m}_{k,k'}\cdot\min_{\vec{F}\in U}\left(\sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}\right) \qquad i\in I,\ (k,k')\in KK', k,k'\in K_U \tag{A.2}$$

$$\begin{aligned}&\text{KE}_{k,k}\cdot(1-\text{UE}_k)\cdot\max_{\vec{F}\in U}\left(\sum_{v\in V_{\text{Final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}\right)\\ &\qquad\le \text{UE}_k\cdot\min_{\vec{F}\in U}\left(\sum_{v\in V_{\text{Final}}}\sum_{c\in C}\sum_{\substack{k'\in K,\\ k'\neq k}} f_{c,k'}M^c_{i,v,c}\cdot\text{KE}_{k,k'}\right) \qquad i\in I, k\in K_U\end{aligned} \tag{A.3}$$

*Appendix A.3. Building auxiliary optimization problems*

To make the model tractable, the maximized and minimized terms in the inequalities above can be reformulated by building an auxiliary optimization problem. For the maximized term $\max_{\vec{F}\in U}\left(\sum_{v\in V_{\text{final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}\right)$ in (A.1) and (A.2), it is the same as the maximized term in (34), leading to the same auxiliary problem (35).

For the minimized term $\min_{\vec{F}\in U}\left(\sum_{v\in V_{\text{final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}\right)$ in (A.1) and (A.2), with subindices $(i\in I,(k,k')\in KK', k,k'\in K_U)$, the corresponding maximization problem is equivalent to:

$$\begin{aligned}&\min_{\vec{F}\in U}\left(\sum_{v\in V_{\text{final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}\right)\\ \equiv\ &\min_{\vec{F}\in U}\left[\sum_{v\in V_{\text{final}}}\sum_{c\in C}\bar{f}_{c,k}(1-\delta p_{c,k})M^c_{i,v,c}\right]\\ \Leftrightarrow\ &\min_{\vec{F}\in U}\left(-\sum_{v\in V_{\text{final}}}\sum_{c\in C}\bar{f}_{c,k}p_{c,k}M^c_{i,v,c}\right)\\ \equiv\ &-\max\sum_{v\in V_{\text{final}}}\sum_{c\in C}\bar{f}_{c,k}\cdot p_{c,k}M^c_{i,v,c}\\ &\text{s.t. } 0\le p_{c,k}\le 1 \qquad\qquad c\in C\\ &\sum_{c\in C} p_{c,k}\le\Gamma\end{aligned} \tag{A.4}$$

It is worth noting that $\max_{\vec{F}\in U}\left(\sum_{v\in V_{\text{final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}\right)$ and $\min_{\vec{F}\in U}\left(\sum_{v\in V_{\text{final}}}\sum_{c\in C} f_{c,k}M^c_{i,v,c}\right)$ are transormed to the same maximization problem in (35) and (A.4) respectively with opposite signs.

For the minimized term $\min_{\vec{F}\in U}\left(\sum_{v\in V_{\text{Final}}}\sum_{c\in C}\sum_{\substack{k'\in K,\\ k'\neq k}} f_{c,k'}M^c_{i,v,c}\cdot \text{KE}_{k,k'}\right)$ in (A.3) with subindices ($i \in I$, $k \in K_U$), the corresponding problem is equivalent to:

$$
\begin{aligned}
&\min_{\vec{F}\in U}\left(\sum_{v\in V_{\text{Final}}}\sum_{c\in C}\sum_{\substack{k'\in K,\\ k'\neq k}} f_{c,k'}M^c_{i,v,c}\cdot \text{KE}_{k,k'}\right)\\
\equiv\ &\min_{\vec{F}\in U}\left[\sum_{v\in V_{\text{final}}}\sum_{c\in C}\sum_{\substack{k'\in K,\\ k'\neq k}} \bar{f}_{c,k'}(1-\delta p_{c,k,k'})M^c_{i,v,c}\cdot \text{KE}_{k,k'}\right]\\
\Leftrightarrow\ &\min_{\vec{F}\in U}\left(-\sum_{v\in V_{\text{final}}}\sum_{c\in C}\sum_{\substack{k'\in K,\\ k'\neq k}} \bar{f}_{c,k'}p_{c,k,k'}M^c_{i,v,c}\cdot \text{KE}_{k,k'}\right)\\
\equiv\ &-\max \sum_{v\in V_{\text{final}}}\sum_{c\in C}\sum_{\substack{k'\in K,\\ k'\neq k}} \bar{f}_{c,k'}p_{c,k,k'}M^c_{i,v,c}\cdot \text{KE}_{k,k'}\\
&\text{s.t. } 0\le p_{c,k,k'}\le 1 && c\in C, k'\in K_U, k'\neq k\\
&\sum_{c\in C} p_{c,k,k'}\le \Gamma && k'\in K_U, k'\neq k
\end{aligned}
\tag{A.5}
$$

*Appendix A.4. Dualizing auxiliary problems*

The auxiliary optimization problem can be dualized to transform the model back to single-level problem. The auxiliary problem (A.4) has the same dualized problem as (35). For every problem with subindices ($i \in I$, $k \in K_U$), the corresponding maximization problem is equivalent to:

$$\min\ \Gamma q_{i,k} + \sum_{c\in C} s_{i,c,k} \tag{36a}$$

$$\text{s.t. } s_{i,c,k} + q_{i,k} \ge \sum_{v\in V_{\text{final}}} \bar{f}_{c,k}M^c_{i,v,c} \qquad c\in C \tag{36b}$$

$$s_{i,c,k} \geq 0 \qquad c \in C \tag{36c}$$

$$q_{i,k} \geq 0 \tag{36d}$$

For the auxiliary problem (A.5) with subindices ($i \in I,\ k \in K_U$), the corresponding maximization problem is equivalent to:

$$\min \sum_{\substack{k' \in K, \\ k' \neq k}} \Gamma q'_{i,k,k'} + \sum_{c \in C} \sum_{\substack{k' \in K, \\ k' \neq k}} s'_{i,c,k,k'} \tag{A.6a}$$

$$\text{s.t. } q'_{i,k,k'} + s'_{i,c,k,k'} \geq \sum_{v \in V_{\text{final}}} \bar{f}_{c,k'} \cdot M^c_{i,v,c} \cdot \mathrm{KE_{k,k'}} \qquad c \in C, k' \in K_U, k' \neq k \tag{41a}$$

$$s'_{i,c,k,k'} \geq 0 \qquad c \in C, k' \in K_U, k' \neq k \tag{41b}$$

$$q'_{i,k,k'} \geq 0 \qquad k' \in K_U, k' \neq k \tag{41c}$$

where $s'_{i,c,k,k'}$ and $q'_{i,k,k'}$ are auxiliary variables of the dualized problem. By strong duality, the optimal objective value of the dual problem is equal to the optimal objective value of (A.5).

Substituting the dualized problems into Equations (24) and (25), the model will automatically minimizes the auxiliary variables. The robust counterpart formulations are presented in below:

$$\begin{aligned} &\underline{m}_{k,k'} \cdot \left[ \sum_{v \in V_{\text{final}}} \sum_{c \in C} \bar{f}_{c,k} M^c_{i,v,c} + \delta \left( \Gamma q_{i,k} + \sum_{c \in C} s_{i,c,k} \right) \right] \\ &\quad \leq \left[ \sum_{v \in V_{\text{final}}} \sum_{c \in C} \bar{f}_{c,k'} M^c_{i,v,c} - \delta \left( \Gamma q_{i,k'} + \sum_{c \in C} s_{i,c,k'} \right) \right] \qquad i \in I,\ (k,k') \in KK', k, k' \in K_U \end{aligned} \tag{38}$$

$$\begin{aligned} &\left[ \sum_{v \in V_{\text{final}}} \sum_{c \in C} \bar{f}_{c,k'} M^c_{i,v,c} + \delta \left( \Gamma q_{i,k'} + \sum_{c \in C} s_{i,c,k'} \right) \right] \\ &\quad \leq \bar{m}_{k,k'} \cdot \left[ \sum_{v \in V_{\text{final}}} \sum_{c \in C} \bar{f}_{c,k} M^c_{i,v,c} - \delta \left( \Gamma q_{i,k} + \sum_{c \in C} s_{i,c,k} \right) \right] \qquad i \in I,\ (k,k') \in KK', k, k' \in K_U \end{aligned} \tag{39}$$

$$
\begin{aligned}
&\mathrm{KE}_{k,k'} \cdot (1 - \mathrm{UE}_k) \cdot \left[ \sum_{v \in V_{\mathrm{final}}} \sum_{c \in C} \bar{f}_{c,k} M^c_{i,v,c} + \delta \left( \Gamma q_{i,k} + \sum_{c \in C} s_{i,c,k} \right) \right] \\
&\quad \leq \mathrm{UE}_k \cdot \left[ \sum_{v \in V_{\mathrm{Final}}} \sum_{c \in C} \sum_{\substack{k' \in K, \\ k' \neq k}} \bar{f}_{c,k'} M^c_{i,v,c} \cdot \mathrm{KE}_{k,k'} - \delta \left( \sum_{\substack{k' \in K_U, \\ k' \neq k}} \Gamma q'_{i,k,k'} + \sum_{c \in C} \sum_{\substack{k' \in K_U, \\ k' \neq k}} s'_{i,c,k,k'} \right) \right] \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad i \in I, k \in K^U
\end{aligned}
\tag{40}
$$

## Appendix B. Formulation of flexibility test problem

The full formulation of the flexibiity test problem is given below as (FT). In (FT), the variables $M^c_{i,v,c}$, $M^t_{i,v}$ in the MOS model are fixed and treated as parameters ($\left(M^c_{i,v,c}\right)^*$, $\left(M^t_{i,v}\right)^*$), while the uncertain parameter $f_{c,k}$ is formulated as ($\bar{f}_{c,k} + \theta_{c,k}$), in which $\theta_{c,k}$ denotes the absolute deviation of the mass fraction and is treated as variables. The quality requirements (23)-(25) are reformulated with ($\bar{f}_{c,k} + \theta_{c,k}$), slack variable $s^{\mathrm{FT}}$ and $u$. The big-M constraints and the constraint summing all $y^{\mathrm{FT}}$'s ensure that only one modified quality requirement is active ($s^{\mathrm{FT}} = 0$), and the rest are relaxed ($s^{\mathrm{FT}} = U$). In this way the flexibility test problem can determine if there is any potential constraint violation, represented by variable $u$, by varying the values of $\theta_{c,k}$ within the uncertainty set. Although the uncertain parameter is represented and constrained in a different way, the uncertainty set in (FT) is identical to (33) which the robust counterpart is derived from.

The solution ($\left(M^c_{i,v,c}\right)^*$, $\left(M^t_{i,v}\right)^*$) from the robust counterpart can be passed to the flexibility test problem. As the worst cases of the uncertainty are considered in the robust counterpart, its solution should be feasible and has no constraint violation for any realization of the uncertainty, which means that the objective $u$ of (FT) should be less than or equal to zero. Since $u$ is maximized in (FT), the value of $\theta_{c,k}$ in the solution corresponds to the deviation of element composition which leads to the smallest feasible region for $\left(M^c_{i,v,c}\right)^*$, $\left(M^t_{i,v}\right)^*$, which is identical to the worst cases for the uncertainty in the robust counterpart.

$$
\max u \tag{FT}
$$

$$
\begin{aligned}
\text{s.t.} \quad & \sum_{v\in V_{\text{Final}}}\sum_{c\in C}(\bar{f}_{c,k}+\theta_{c,k})\cdot\left(M^{c}_{i,v,c}\right)^{*}-\bar{f}_{k}\sum_{v\in V_{\text{Final}}}\left(M^{t}_{i,v}\right)^{*}+s^{\text{FT},1}_{i,k}=u && i\in I,k\in K_U\\
& \sum_{v\in V_{\text{Final}}}\sum_{c\in C}(\bar{f}_{c,k'}+\theta_{c,k'})\cdot\left(M^{c}_{i,v,c}\right)^{*}\\
& \quad -\bar{m}_{k,k'}\sum_{v\in V_{\text{Final}}}\sum_{c\in C}(\bar{f}_{c,k}+\theta_{c,k})\cdot\left(M^{c}_{i,v,c}\right)^{*}+s^{\text{FT},2}_{i,k,k'}=u && i\in I,(k,k')\in KK',k,k'\in K_U\\
& \underline{m}_{k,k'}\sum_{v\in V_{\text{Final}}}\sum_{c\in C}(\bar{f}_{c,k}+\theta_{c,k})\cdot\left(M^{c}_{i,v,c}\right)^{*}\\
& \quad -\sum_{v\in V_{\text{Final}}}\sum_{c\in C}(\bar{f}_{c,k'}+\theta_{c,k'})\cdot\left(M^{c}_{i,v,c}\right)^{*}+s^{\text{FT},3}_{i,k,k'}=u && i\in I,(k,k')\in KK',k,k'\in K_U\\
& \text{KE}_k\cdot\sum_{v\in V_{\text{Final}}}\sum_{c\in C}(\bar{f}_{c,k}+\theta_{c,k})\cdot\left(M^{c}_{i,v,c}\right)^{*}\\
& \quad -\text{UE}_k\left[\sum_{v\in V_{\text{Final}}}\sum_{c\in C}\sum_{k'\in K}(\bar{f}_{c,k'}+\theta_{c,k'})\cdot\left(M^{c}_{i,v,c}\right)^{*}\cdot\text{KE}_k\right]+s^{\text{FT},4}_{i,k}=u && i\in I,k\in K_U\\
& s^{\text{FT},1}_{i,k}-U(1-y^{\text{FT},1}_{i,k})\le 0 && i\in I,k\in K_U\\
& s^{\text{FT},2}_{i,k,k'}-U(1-y^{\text{FT},2}_{i,k,k'})\le 0 && i\in I,(k,k')\in KK',k,k'\in K_U\\
& s^{\text{FT},3}_{i,k,k'}-U(1-y^{\text{FT},3}_{i,k,k'})\le 0 && i\in I,(k,k')\in KK',k,k'\in K_U\\
& s^{\text{FT},4}_{i,k}-U(1-y^{\text{FT},4}_{i,k})\le 0 && i\in I,k\in K_U\\
& \sum_{i\in I}\left[\sum_{(k,k')\in KK'}\left(y^{\text{FT},2}_{i,k,k'}+y^{\text{FT},3}_{i,k,k'}\right)+\sum_{k\in K_U}\left(y^{\text{FT},1}_{i,k}+y^{\text{FT},4}_{i,k}\right)\right]=1\\
& \sum_{c\in C}\bar{\theta}_{c,k}\le\Gamma && k\in K_U\\
& -\delta\cdot\bar{f}_{c,k}\cdot\bar{\theta}_{c,k}\le\theta_{c,k}\le\delta\cdot\bar{f}_{c,k}\cdot\bar{\theta}_{c,k} && c\in C,k\in K_U\\
& \bar{\theta}_{c,k}\ge 0 && c\in C,k\in K_U
\end{aligned}
$$

## Appendix C. Data for case studies

Table C.1: Case study data for the illustrative example

| Time horizon | | 10 days | |
|---|---|---|---|
| Vessels | Arrival time (day) | Concentrate type | Amount (ton) |
| $r_1$ | 0 | $c_1$ | 11,380 |
| $r_2$ | 3 | $c_2$ | 10,800 |
| Piles at the port | | Concentrate type | Amount (ton) |
| $r_3$ | | - | - |
| $r_4$ | | - | - |
| $r_5$ | Unlimited capacity | $c_3$ | 5,017 |
| $r_6$ | | $c_4$ | 5,009 |
| $r_7$ | | $c_5$ | 4,719 |
| Pre-blending unit | Unlimited capacity | Inlet flowrate (ton/day) | [0, 5000] |
| Daily arrival piles | Concentrate type | Inlet flowrate (ton/day) | Outlet flowrate (ton/day) |
| $r_{13}$ | $c_6$ | 37.34 | [0, 50] |
| Bins | Unlimited capacity | Inlet flowrate (ton/day) | [0, 1300] |
| | | Outlet flowrate (ton/day) | [150, 900] |
| $[\underline{D}_v, \bar{D}_v]$ (day) | [0.05, 5] | $[\underline{E}_v, \bar{E}_v]$ (day) | [0, 10] |
| $W'$ for Equation (3) | $\{v_1\}, \{v_2\}$ | $[\underline{N}_{W'}, \bar{N}_{W'}]$ | $[1, 1]$ |
| $\bar{F}_{\text{smelter}}$ (ton/day) | 3000 | $\bar{L}_{\text{daily}}$ | 100 |
| $[\underline{r}_{\text{bin}}, \bar{r}_{\text{bin}}]$ | $[0.2, 0.4]$ | $[\underline{M}_v^t, \bar{M}_v^t]$ (ton) | $[0, 20000]$ |

Table C.2: Data for concentrates and elements for the illustrative example

| Copper concentrate | Key elements $k_1$ | $k_2$ | $k_3$ | $k_4$ | Profit ($/ton) |
|---|---|---|---|---|---|
| $c_1$ | $2.54 \cdot 10^{-1}$ | $2.97 \cdot 10^{-1}$ | $3.29 \cdot 10^{-1}$ | $1.20 \cdot 10^{-1}$ | 146 |
| $c_2$ | $7.00 \cdot 10^{-1}$ | $5.07 \cdot 10^{-3}$ | $1.85 \cdot 10^{-1}$ | $1.10 \cdot 10^{-1}$ | 1172 |
| $c_3$ | $2.50 \cdot 10^{-1}$ | $2.20 \cdot 10^{-1}$ | $2.70 \cdot 10^{-1}$ | $2.60 \cdot 10^{-1}$ | 319 |
| $c_4$ | $2.21 \cdot 10^{-1}$ | $3.70 \cdot 10^{-1}$ | $3.55 \cdot 10^{-4}$ | $5.50 \cdot 10^{-2}$ | 216 |
| $c_5$ | $2.41 \cdot 10^{-1}$ | $2.55 \cdot 10^{-1}$ | $3.15 \cdot 10^{-1}$ | $1.89 \cdot 10^{-1}$ | 264 |
| $c_6$ | $2.07 \cdot 10^{-1}$ | $3.50 \cdot 10^{-1}$ | $1.21 \cdot 10^{-1}$ | $3.22 \cdot 10^{-1}$ | 0 |
| Upper bounds of composition in final mixture | 0.4 | 0.285 | 0.31 | 1 | |

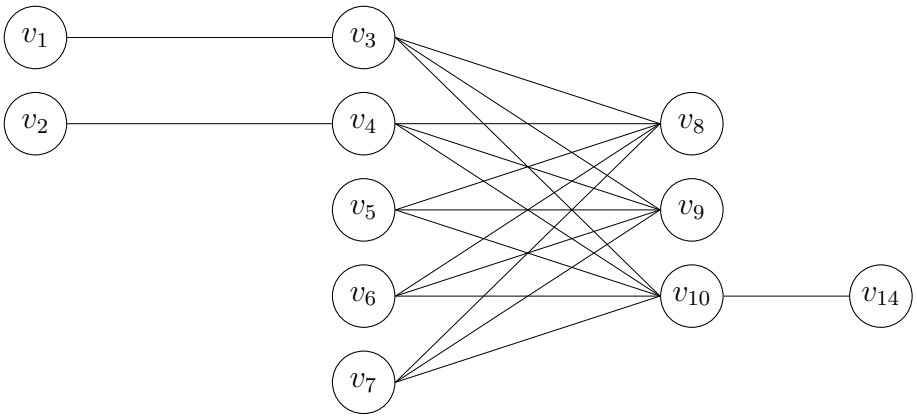


Figure C.1: Cliques in the illustrative example

Table C.3: Case study data for the industrial case

| Time horizon | | 15 days | |
|---|---|---|---|
| Vessels | Arrival time (day) | Concentrate type | Amount (ton) |
| $r_1$ | 0 | $c_1$ | 11,380 |
| $r_2$ | 3 | $c_2$ | 12,590 |
| $r_3$ | 6 | $c_3$ | 10,800 |
| $r_4$ | 7 | $c_4$ | 15,600 |
| $r_5$ | 10 | $c_5$ | 183 |
| Piles at the port | | Concentrate type | Amount (ton) |
| $r_6$ | | - | - |
| $r_7$ | | - | - |
| $r_8$ | | - | - |
| $r_9$ | | - | - |
| $r_{10}$ | Unlimited capacity | - | - |
| $r_{11}$ | | $c_6$ | 5,017 |
| $r_{12}$ | | $c_7$ | 5,009 |
| $r_{13}$ | | $c_8$ | 4,719 |
| $r_{14}$ | | $c_9$ | 3,543 |
| $r_{15}$ | | $c_{10}$ | 2,609 |
| $r_{16}$ | | $c_{11}$ | 10,000 |
| Pre-blending unit | Unlimited capacity | Inlet flowrate (ton/day) | [0, 5000] |
| Daily arrival piles | Concentrate type | Inlet flowrate (ton/day) | Outlet flowrate (ton/day) |
| $r_{25}$ | $c_{12}$ | 37.34 | [0, 50] |
| $r_{26}$ | $c_{13}$ | 36.05 | [0, 50] |
| $r_{27}$ | $c_{14}$ | 6.26 | [0, 15] |
| Bins | Unlimited capacity | Inlet flowrate (ton/day) | [0, 1300] |
| | | Outlet flowrate (ton/day) | [150, 900] |
| $[\underline{D}_v, \bar{D}_v]$ (day) | [0.05, 5] | $[\underline{E}_v, \bar{E}_v]$ (day) | [0, 10] |
| $W'$ for Equation (3) | $\{v_1\}, \{v_2\}, \{v_3\}, \{v_4\}, \{v_5\}$ | $[\underline{N}_{W'}, \bar{N}_{W'}]$ | $[1, 1]$ |
| $KK'$ for Equation (25) | $\{(k_2, k_7)\}$ | $[\underline{m}_{k,k'}, \bar{m}_{k,k'}]$ | $[0.58, 0.64]$ |
| $\bar{F}_{\text{smelter}}$ (ton/day) | 3000 | $\bar{L}_{\text{daily}}$ | 100 |
| $[\underline{r}_{\text{bin}}, \bar{r}_{\text{bin}}]$ | $[0.2, 0.4]$ | $[\underline{M}_v^t, \bar{M}_v^t]$ (ton) | $[0, 20000]$ |

Table C.4: Data for concentrates and elements for the industrial case

| Copper concentrate | Key elements $k_1$ | $k_2$ | $k_3$ | $k_4$ | $k_5$ | $k_6$ | $k_7$ | $k_8$ | $k_9$ | Profit ($/ton) |
|---|---|---|---|---|---|---|---|---|---|---|
| $c_1$ | $2.54 \cdot 10^{-1}$ | $2.97 \cdot 10^{-1}$ | $8.06 \cdot 10^{-5}$ | $2.06 \cdot 10^{-5}$ | $1.12 \cdot 10^{-4}$ | $2.30 \cdot 10^{-5}$ | $5.30 \cdot 10^{-2}$ | $3.29 \cdot 10^{-1}$ | $6.66 \cdot 10^{-2}$ | 146 |
| $c_2$ | $3.03 \cdot 10^{-1}$ | $3.07 \cdot 10^{-1}$ | $7.06 \cdot 10^{-5}$ | $4.55 \cdot 10^{-5}$ | $3.49 \cdot 10^{-5}$ | $4.57 \cdot 10^{-3}$ | $1.32 \cdot 10^{-2}$ | $3.45 \cdot 10^{-1}$ | $2.80 \cdot 10^{-2}$ | 1130 |
| $c_3$ | $7.00 \cdot 10^{-1}$ | $5.00 \cdot 10^{-3}$ | $5.00 \cdot 10^{-4}$ | $1.50 \cdot 10^{-6}$ | 0 | $6.00 \cdot 10^{-2}$ | 0.00 | $1.85 \cdot 10^{-1}$ | $4.95 \cdot 10^{-2}$ | 1172 |
| $c_4$ | $3.72 \cdot 10^{-1}$ | $1.76 \cdot 10^{-1}$ | $2.85 \cdot 10^{-4}$ | $2.02 \cdot 10^{-5}$ | $1.84 \cdot 10^{-4}$ | $5.97 \cdot 10^{-5}$ | $1.36 \cdot 10^{-1}$ | $1.30 \cdot 10^{-1}$ | $1.85 \cdot 10^{-1}$ | 412 |
| $c_5$ | $2.00 \cdot 10^{-2}$ | $2.75 \cdot 10^{-2}$ | $5.00 \cdot 10^{-3}$ | $3.70 \cdot 10^{-4}$ | 0 | $2.00 \cdot 10^{-5}$ | $7.30 \cdot 10^{-1}$ | $2.00 \cdot 10^{-3}$ | $2.15 \cdot 10^{-1}$ | 764 |
| $c_6$ | $2.50 \cdot 10^{-1}$ | $2.20 \cdot 10^{-1}$ | $1.00 \cdot 10^{-2}$ | $1.00 \cdot 10^{-7}$ | $5.00 \cdot 10^{-5}$ | $1.00 \cdot 10^{-4}$ | $5.80 \cdot 10^{-2}$ | $2.70 \cdot 10^{-1}$ | $1.92 \cdot 10^{-1}$ | 319 |
| $c_7$ | $2.21 \cdot 10^{-1}$ | $3.70 \cdot 10^{-1}$ | $5.38 \cdot 10^{-4}$ | $6.69 \cdot 10^{-7}$ | $8.85 \cdot 10^{-6}$ | $3.38 \cdot 10^{-5}$ | $4.67 \cdot 10^{-3}$ | $3.55 \cdot 10^{-1}$ | $4.94 \cdot 10^{-2}$ | 216 |
| $c_8$ | $2.41 \cdot 10^{-1}$ | $2.55 \cdot 10^{-1}$ | $1.57 \cdot 10^{-3}$ | $1.57 \cdot 10^{-6}$ | 0 | $2.43 \cdot 10^{-5}$ | $7.75 \cdot 10^{-2}$ | $3.15 \cdot 10^{-1}$ | $1.10 \cdot 10^{-1}$ | 264 |
| $c_9$ | $2.36 \cdot 10^{-1}$ | $6.99 \cdot 10^{-2}$ | $1.55 \cdot 10^{-3}$ | $7.50 \cdot 10^{-8}$ | 0 | $1.35 \cdot 10^{-5}$ | $1.25 \cdot 10^{-1}$ | $1.18 \cdot 10^{-1}$ | $4.50 \cdot 10^{-1}$ | 214 |
| $c_{10}$ | $2.19 \cdot 10^{-3}$ | $2.80 \cdot 10^{-1}$ | $9.30 \cdot 10^{-3}$ | $1.29 \cdot 10^{-4}$ | 0 | $4.22 \cdot 10^{-4}$ | $3.65 \cdot 10^{-1}$ | $1.55 \cdot 10^{-1}$ | $1.87 \cdot 10^{-1}$ | 437 |
| $c_{11}$ | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| $c_{12}$ | $2.41 \cdot 10^{-1}$ | $3.09 \cdot 10^{-1}$ | $1.13 \cdot 10^{-3}$ | $5.14 \cdot 10^{-6}$ | 0 | $3.22 \cdot 10^{-3}$ | $1.83 \cdot 10^{-1}$ | $8.88 \cdot 10^{-2}$ | $1.74 \cdot 10^{-1}$ | 0 |
| $c_{13}$ | $2.07 \cdot 10^{-1}$ | $3.50 \cdot 10^{-1}$ | $7.74 \cdot 10^{-4}$ | $4.29 \cdot 10^{-6}$ | 0 | $2.11 \cdot 10^{-3}$ | $2.40 \cdot 10^{-1}$ | $1.21 \cdot 10^{-1}$ | $7.86 \cdot 10^{-2}$ | 0 |
| $c_{14}$ | $4.28 \cdot 10^{-1}$ | 0.00 | $1.68 \cdot 10^{-2}$ | $2.78 \cdot 10^{-5}$ | 0 | 0 | 0 | 0 | $5.56 \cdot 10^{-1}$ | 0 |
| Upper bound of composition in final mixture | 1 | 0.285 | 0.0011 | 1 | 1 | 0.001 | 1 | 0.31 | 1 | |
| $UE_k$ | 1 | 1 | 0.001 | 0.0001 | 1 | 0.0033 | 1 | 1 | 1 | |
| $KE_k$ | 0.8680 | 0 | 0.1888 | 0.9078 | 0.4319 | 0 | 0 | 0 | 0.0002 | |

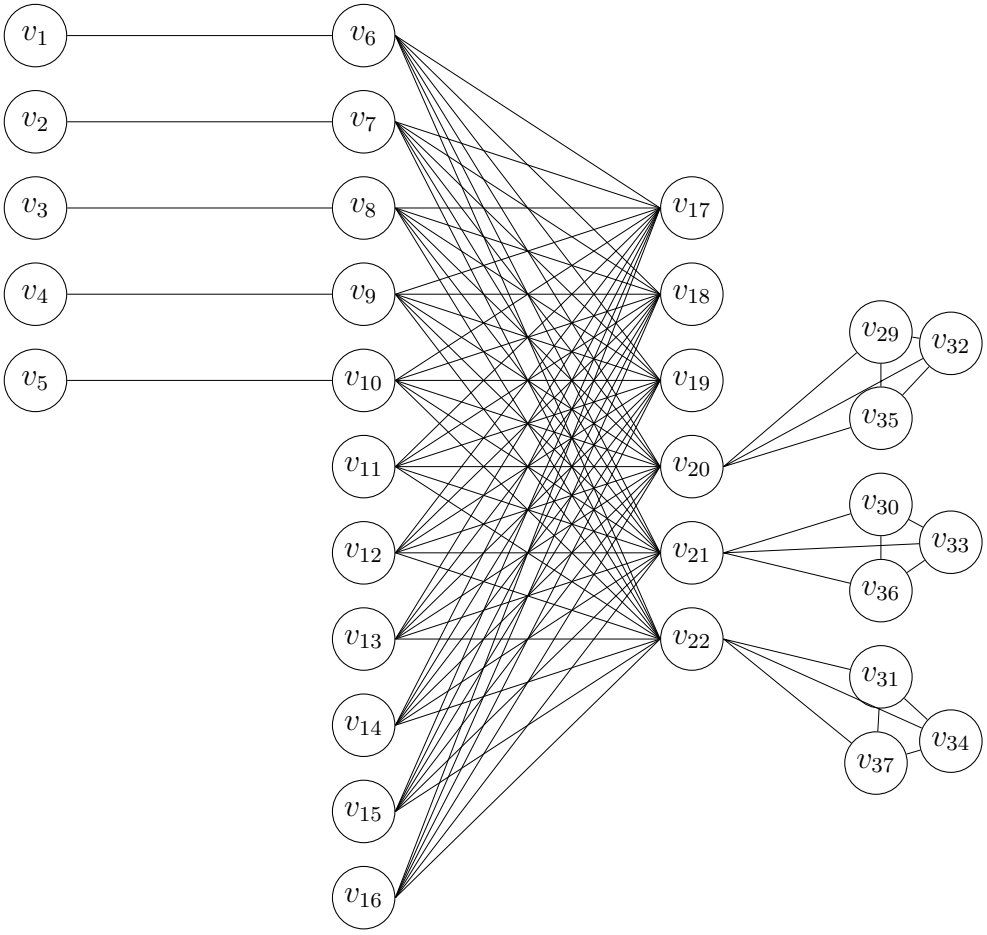


Figure C.2: Cliques in the industrial case

## Nomenclature

### Indices

| | |
|---|---|
| $c$ | Concentrates |
| $i, j, i', i''$ | Priority slots |
| $k, k'$ | Elements |
| $r$ | Units |
| $v, v', v''$ | Operations |

### Sets

| | |
|---|---|
| $C$ | Concentrates |

$C_{\text{da}}$ Daily-arrival concentrates

$I$ Priority slots

$K$ Elements

$K_U$ Elements whose composition fluctuates

$KK'$ Pairs of elements with composition interdependency

$R$ Units

$R_{\text{bin}}$ Bins

$R_{\text{blender}}$ Blending units

$R_{\text{dapile}}$ Daily-arrival piles

$V$ Operations

$V_r^{\text{in}}$ Inlet operations for unit $r$

$V_r^{\text{out}}$ Outlet operations for unit $r$

$V_{\text{Final}}$ Final operations that transfer concentrates from bins to the smelter

$W'$ Clique, set of non-overlapping operations

$W$ Set of cliques

**Parameters**

$\bar{f}_k$ Maximum mass fraction of element $k$ in concentrate mixture

$\bar{F}_{\text{smelter}}$ Upper bound of total flowrate of concentrates transferred to the smelter

$\bar{f}_{c,k}$ Nominal mass fraction of element $k$ in concentrate $c$

$\bar{L}_{\text{daily}}$ Upper bound of total daily-arrival material inventory at the end of time horizon

$\delta$ Relative bound of the fluctuation for the mass fraction of elements

$\Gamma$ Budget parameter for modifying the uncertainty set

$\text{KE}_k$ Distribution coefficient for element $k$ for interdependency constraint (25)

$\text{UE}_k$ Upper bound for element $k$ for interdependency constraint (25)

$\underline{D}_v, \bar{D}_v$ Lower and upper bounds of duration of operation $v$

$\underline{E}_v, \bar{E}_v$ Lower and upper bounds of end time of operation $v$

$\underline{F}_v, \bar{F}_v$ Lower and upper bounds of flowrate of operation $v$

$\underline{L}_r^t, \bar{L}_r^t$ Lower and upper bounds of capacity of unit $r$

$\underline{M}_v^t, \bar{M}_v^t$ Lower and upper bounds of total amount of concentrates in operation $v$

$\underline{m}_{k,k'}, \bar{m}_{k,k'}$ Lower and upper bounds of ratio between amount of element $k$ and element $k'$

$\underline{N}_{W'}, \bar{N}_{W'}$ Lower and upper bounds of cardinalities of operations in clique $W'$

$\underline{r}_{\text{bin}}, \bar{r}_{\text{bin}}$ Lower and upper bounds of the ratio of contribution of each bin

$\underline{S}_v, \bar{S}_v$ Lower and upper bounds of start time of operation $v$

$f_{c,k}$ Mass fraction of element $k$ in concentrate $c$

$G_c$ Gross margin of concentrate $c$

$H$ Time horizon

$L_{r,c}^{c,\text{initial}}$ Initial level of concentrate $c$ in unit $r$

$L_r^{t,\text{initial}}$ Initial total level of concentrates in unit $r$

$p_{c,k}$ Normalized deviation of mass fraction for element $k$ in concentrate $c$

$U$ Uncertainty set

**Variables**

$\bar{\theta}_{c,k}$ Normalized mass fraction deviation of element $k$ in concentrate $c$

$\theta_{c,k}$ Absolute mass fraction deviation for element $k$ in concentrate $c$

$\varepsilon_{i,k}^1, \varepsilon_k^2, \varepsilon_k^3, \varepsilon_{i,k}^4$ Violation terms in bi-criterion robust MOS model

$D_{i,v}$ Duration of operation $v$ assigned to slot $i$

$E_{i,v}$ End time of operation $v$ assigned to slot $i$

$L_{i,r,c}^c$ Accumulated level of concentrate $c$ in unit $r$ at slot $i$

$L_{i,r}^t$ Total accumulated level of concentrates in unit $r$ at slot $i$

$L_{r,c}^{c,\text{Final}}$ Final level of concentrate $c$ in unit $r$

$L_r^{t,\text{Final}}$ Final total level of concentrates in unit $r$

$M_{i,v,c}^c$ Amount of concentrate $c$ in operation $v$ at slot $i$

$M_{i,v}^t$ Total amount of concentrates in operation $v$ at slot $i$

$s'_{i,c,k,k'}, q'_{i,k,k'}$ Dual variables for the dual of inner maximization problem for constraint (25)

$s_{i,c,k}, q_{i,k}$ Dual variables for the dual of inner maximization problem for constraint (23),(24), (25)

$s^{\mathrm{FT},1}_{i,k}, s^{\mathrm{FT},2}_{i,k,k'}, s^{\mathrm{FT},3}_{i,k,k'}, s^{\mathrm{FT},4}_{i,k}$ Slack variables for flexibility test problem

$S_{i,v}$ Start time of operation $v$ assigned to slot $i$

$u$ Objective of flexibility test problem

$y^{\mathrm{FT},1}_{i,k}, y^{\mathrm{FT},2}_{i,k,k'}, y^{\mathrm{FT},3}_{i,k,k'}, y^{\mathrm{FT},4}_{i,k}$ Binary variables for flexibility test problem

$Y_{i,v,c}$ Assignment variable to denote if concentrate $c$ is in operation $v$ at slot $i$, binary variable

$Z_{i,v}$ Assignment variable to denote if operation $v$ is assigned to slot $i$, binary variable

## References


Alonso-Ayuso, A., Carvallo, F., Escudero, L. F., Guignard, M., Pi, J., Puranmalka, R., Weintraub, A., 2014. Medium range optimization of copper extraction planning under uncertainty in future copper prices. Eur. J. Oper. Res. 233 (3), 711–726.

Bertsimas, D., Brown, D. B., Caramanis, C., 2011. Theory and applications of robust optimization. SIAM review 53 (3), 464–501.

Bertsimas, D., Sim, M., 2003. Robust discrete optimization and network flows. Math. Program. 98 (1-3), 49–71.

Birge, J. R., Louveaux, F., 2011. Introduction to stochastic programming. Springer Science & Business Media.

Castro, P., Barbosa-Póvoa, A. P. F. D., Matos, H., 2001. An improved rtn continuous-time formulation for the short-term scheduling of multipurpose batch plants. Ind. & Eng. Chem. Res. 40 (9), 2059–2068.

Cohon, J. L., 1978. Multiobjective programming and planning. Academic Press.

Diestel, R., 2005. Graph theory. 2005. Grad. Texts Math 101.

GAMS Development Corporation, 2019. General Algebraic Modeling System (GAMS) Release 27.2.0. Fairfax, VA, USA.

URL http://www.gams.com/

Grossmann, I. E., Apap, R. M., Calfa, B. A., Garcia-Herreros, P., Zhang, Q., 2016. Recent Advances in Mathematical Programming Techniques for the Optimization of Process Systems under Uncertainty. Comput. Chem. Eng. 91, 3–14.

URL http://dx.doi.org/10.1016/j.compchemeng.2016.03.002

Grossmann, I. E., Calfa, B. A., Garcia-Herreros, P., 2014. Evolution of concepts and models for quantifying resiliency and flexibility of chemical processes. Comput. & Chem. Eng. 70, 22–34.

Harjunkoski, I., Maravelias, C. T., Bongers, P., Castro, P. M., Engell, S., Grossmann, I. E., Hooker, J., Méndez, C., Sand, G., Wassick, J., 2014. Scope for industrial applications of production scheduling models and solution methods. Comput. Chem. Eng. 62, 161–193.
URL http://dx.doi.org/10.1016/j.compchemeng.2013.12.001

Ierapetritou, M. G., Floudas, C. A., 1998a. Effective continuous-time formulation for short-term scheduling. 1. Multipurpose batch processes. Ind. Eng. Chem. Res. 37 (11), 4341–4359.

Ierapetritou, M. G., Floudas, C. A., 1998b. Effective continuous-time formulation for short-term scheduling. 2. Continuous and semicontinuous processes. Ind. Eng. Chem. Res. 37 (11), 4360–4374.

Kondili, E., Pantelides, C. C., Sargent, R. W. H., 1993. A general algorithm for short-term scheduling of batch operations-I. MILP formulation. Comput. Chem. Eng. 17 (2), 211–227.

Lalpuria, M., 2017. Optimal scheduling of copper concentrate operations using priority slots. Master thesis, Carnegie Mellon University.

Langner, B., 2011. Understanding Copper: Technologies, Markets, Business. Eigenverl.
URL https://books.google.com/books?id=X7lLMwEACAAJ

Li, Z., Ierapetritou, M., 2008. Process scheduling under uncertainty: Review and challenges. Comput. Chem. Eng. 32 (4-5), 715–727.

Maravelias, C. T., 2012. General framework and modeling approach classification for chemical production scheduling. AIChE J. 58 (6), 1812–1828.

Méndez, C. A., Cerdá, J., Grossmann, I. E., Harjunkoski, I., Fahl, M., 2006. State-of-the-art review of optimization methods for short-term scheduling of batch processes. Comput. Chem. Eng. 30 (6-7), 913–946.

Mouret, S., Grossmann, I. E., Pestiaux, P., 2009. A novel priority-slot based continuous-time formulation for crude-oil scheduling problems. Ind. Eng. Chem. Res. 48 (18), 8515–8528.

Mouret, S., Grossmann, I. E., Pestiaux, P., 2011. Time representations and mathematical models for process scheduling problems. Comput. Chem. Eng. 35 (6), 1038–1063.
URL http://dx.doi.org/10.1016/j.compchemeng.2010.07.007

Pantelides, C., 1994. Unified Frameworks for the Optimal Process Planning and Scheduling. Proceedings of the 2nd Conference on the Foundations of Computer Aided Operations. New York: Cache Publ., 253.

Song, Y., Menezes, B. C., Garcia-Herreros, P., Grossmann, I. E., 2018. Scheduling and Feed Quality Optimization of Concentrate Raw Materials in the Copper Refining Industry. Ind. Eng. Chem. Res. 57 (34), 11686–11701.
URL http://pubs.acs.org/doi/10.1021/acs.iecr.8b01512

Soyster, A. L., 1973. Convex programming with set-inclusive constraints and applications to inexact linear programming. Oper. research 21 (5), 1154–1157.

Zhang, Q., Grossmann, I. E., Lima, R. M., 2016. On the relation between flexibility analysis and robust optimization for linear systems. AIChE J. 62 (9), 3109–3123.