Production scheduling at LKAB’s Kiruna Mine using mixed-integer programming

Introduction

Both heuristic and exact approaches, specifically mathematical programming, have been used for production scheduling in underground mines. While heuristic algorithms may produce usable schedules, there is no easy way to deduce the quality of these schedules relative to the best (cost-minimizing) schedule. Often, the quality of a heuristic schedule is assessed by comparing it to the current status operation. A mathematical programming technique, mixed-integer programming (MIP), has been used to produce optimal production schedules for underground mines, e.g., schedules that result in a minimum cost, for limited cases. The practical use of MIP has been hindered because models must incorporate several decision variables, with many of these restricted to assume integer values. The large number of integer variables required for model formulation results in commensurately long solution times that may be unacceptable for practical planning purposes. By preprocessing the production data and through careful model formulation, it is possible to reduce the number of integer variables and thus reduce solution times. This paper summarizes a new mixed-integer programming model that developed for the Kiruna Mine.

The Kiruna Mine

LKAB’s Kiruna Mine is located in northern Sweden, produces about 24 Mta (23.6 million tons) of iron ore using sublevel caving. For mill efficiencies, the mine must deliver planned quantities of three ore products. Mixed-integer programming is used to schedule Kiruna’s operations. Specifically, this involves determining which production blocks should be mined and when they should be mined to minimize deviations from planned production quantities while adhering to mining constraints. A new production block database, and a well-formulated mixed-integer programming model dramatically reduce the size of the original problem. In less than ten minutes, five-year schedules with time fidelity of months are generated.

LKAB’s Kiruna Mine, located in northern Sweden, produces about 24 Mta (23.6 million tons) of iron ore using sublevel caving. For mill efficiencies, the mine must deliver planned quantities of three ore products. Mixed-integer programming is used to schedule Kiruna’s operations. Specifically, this involves determining which production blocks should be mined and when they should be mined to minimize deviations from planned production quantities while adhering to mining constraints. A new production block database, and a well-formulated mixed-integer programming model dramatically reduce the size of the original problem. In less than ten minutes, five-year schedules with time fidelity of months are generated.

Abstract

LKAB’s Kiruna Mine, located in northern Sweden, produces about 24 Mta (23.6 million tons) of iron ore using sublevel caving. For mill efficiencies, the mine must deliver planned quantities of three ore products. Mixed-integer programming is used to schedule Kiruna’s operations. Specifically, this involves determining which production blocks should be mined and when they should be mined to minimize deviations from planned production quantities while adhering to mining constraints. A new production block database, and a well-formulated mixed-integer programming model dramatically reduce the size of the original problem. In less than ten minutes, five-year schedules with time fidelity of months are generated.

Towards the hangingwall, but occasionally occurs in the middle or along the footwall. Phosphorous (P) and potassium (K) are the major ore contaminants. The phosphorous grade in the D ore varies considerably and averages about 2%. The average grade for the best-quality B ore is about 0.025% phosphorous with an iron (Fe) grade of about 66%. In some areas, the B ore is highly brecciated, resulting in lower iron grades and higher potassium grades.

From the two major in situ ore types, the mine delivers three raw ore products: B1, B2 and D3 (Table 1). These three ore products are classified according to the phosphorous content. The B product generally cannot be mined directly from the in situ ore, but rather it is produced almost entirely from the blending of high-phosphorous D ore with low-phosphorous B ore during extraction. Kicheta (1999) explains the blending process using principles of gravitational flow. The mining method is large-scale sublevel caving (Fig. 1). Transverse sublevel caving is normally used with mining proceeding from the hanging wall to the footwall, but longitudinal sublevel caving is used in a few areas of the mine. The spacing between sublevels is about 27 m (89 ft) and the spacing between crosscuts is about 25 m (82 ft). The
production drifts are 7-m (23-ft) wide and 5-m (16-ft) high. From the production drifts, rings of holes are drilled upward in a fan-shaped pattern, with spacing between rings of 3 to 3.5 m (9.8 to 11.5 ft). Rings are usually drilled at an inclination of 10° forward (towards the hangingwall). Each production ring contains around 10 11th (9,500 t) of ore.

The orebody is now being mined from the 1,045-m (3,428-ft) main transportation level (Fig. 2). The mine is divided into 10 production areas, each with its own group of ore passes and ventilation shafts. Each of the 10 production areas is about 400 to 500 m (1,310 to 1,640 ft) in length. The ore passes are located at the center of the production areas. Mining begins at the uppermost sublevel and proceeds sequentially downward. One to two 25-kW (27-kW) electric load haul dump (LHD) units operating on a sublevel within each production area transport the ore from the production drifts to the ore passes. The ore passes extend down to the 1,045-m (3,430-ft) transportation level, where the ore is transported by large trains to the main chutes and then hoisted to the surface through a series of vertical shafts.

The site on which each LHD operates is also referred to as a "machine placement." Once started, mining is continuous within a machine placement until all available ore has been removed. Depending on production requirements, as many as 18 LHDs are active daily throughout the mine. Each machine placement is usually 200 to 500 m (660 to 1,640 ft) in length and contains from 3 to 3 Mt (1.08 to 2.9 million t) of ore. This corresponds to about 15 monthly production blocks per machine placement. At most, two machine placements can be simultaneously active per production area and for each such that the required tons of B1, B2, and D3 ore can be produced each month. The mine supplies one post processing plant (mill) with B1 ore, two mills with B2 ore, and one mill with D3 ore. Because the mine can stockpile only about 6 kt (5,900 t) of each day's, the mine must meet these production demands almost exactly, so that the mills can meet their respective production demands. The deviation from the specified production levels for each ore type in each month is minimized. Moreover, the mine must observe the following operational constraints:

- accounting constraints that track the amount of each ore type mined in each month;
- vertical sequencing constraints that preclude mining a machine placement under a given machine placement until at least 50% of the given machine placement has been mined;
- horizontal sequencing constraints that require adjacent machine placements on the same sublevel to be mined after 50% of the given machine placement has been mined; and
- shaft group constraints that restrict the number of LHDs active within a shaft group at any one time to a predetermined maximum, usually two or three.

**Model formulation**

The model formulation follows:

**Indices**

- \( a \) is the machine placement;
- \( b/b' \) is the production block;
- \( k \) is the ore type; i.e., B1, B2, D3;
- \( i \) is the time period (month); and
- \( s \) is the shaft group(s). i.e., 1, 10.

**Sets**

- \( \Omega \) is the set of eligible time periods in which production block \( b \) can be mined (restricted by production block location and the start date of other relevant production blocks);
- \( A \) is the set of production

---

**TABLE 1**

<table>
<thead>
<tr>
<th>Product</th>
<th>Fe %</th>
<th>P %</th>
<th>K₂O %</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>B1</td>
<td>68.0</td>
<td>0.0</td>
<td>0.160</td>
<td>Fine production</td>
</tr>
<tr>
<td>B2</td>
<td></td>
<td>0.2</td>
<td></td>
<td>Medium P pellets feed</td>
</tr>
<tr>
<td>D3</td>
<td></td>
<td>0.5</td>
<td></td>
<td>High P pellets feed</td>
</tr>
</tbody>
</table>
The deviations from the planned quantities of B1, B2 and D3 ores for each month are minimized. Constraints Eq. (1) calculate the tons of B1, B2, and D3 ore mined per time period and the corresponding deviations from the specified production levels. Constraints Eq. (2) comprise the vertical sequencing constraints between mining sublevels. Constraints Eq. (3) and (4) enforce horizontal sequencing constraints between adjacent production blocks. Constraints Eq. (5) limit the number of active production blocks within a shaft group. The last set of constraints enforces nonnegativity and integrality of variables, as appropriate.

Results

The mixed integer program is implemented using the AMPL programming language (Fourer et al., 1993) and the CPLEX solver, Version 7.0 (ILOG Corp., 2001), and the model instances are run on a Sun Ultra 10 machine with 256 MB RAM. A five-year schedule using 60 monthly time periods for the test case with current production requirements and planning data contains 56 machine placements.

The number of time periods, machine placements and corresponding production blocks for this scenario would have required 60×56×15 = 54,000 integer variables. About 12 production blocks can be aggregated into a single machine placement through careful model reformulation. The number of time periods and, hence, the number of variables under consideration can be further restricted by using both an earliest possible start date and a latest possible start date for each machine placement. Specifically, the mining sequence is highly restricted and precludes:

- the deeper-lying machine placements from being mined until at least 50% of the directly overlying machine placement has been mined, and
- a machine placement from starting to be mined so late that the underlying machine placements eventu-
alloy become "locked in," inhibiting the mine from producing the required tonnage.

Using early and late start values to restrict the eligible time periods in which a production block can be mined interval, the number of integral variables required for the complete five-year schedule is reduced to about 650.

Several additional constraints, while redundant with the original constraints, restrict the search space for the optimal solution, which reduces solution time. Specifically, several constraints are added that:

- require block b to start being mined at some point during the time horizon if its late start date falls within the time horizon and
- allow block b to start being mined once during the time horizon if its late start date occurs beyond the time horizon.

A complete five-year schedule is obtained in 300 seconds. Figure 4 depicts the first year of a complete five-year schedule.

Discussion

Both linear programming models have been applied to underground mine scheduling for a variety of mining methods (e.g., Williams et al., 1972) for sublevel stoping in a copper mine and Jawed (1993) for room-and-pillar in a coal mine. Others (Winkler, 1998 and Tang et al., 1993) combine linear programming with simulation. These underground mining models are primarily linear programming models that determine the amount of ore to be mined. The binary decisions of whether to mine a production block are either ignored or incorporated into the model ex post facto through, for example, simulation or an iterative technique to induce feasibility of an indirectly determined mining sequence.

Winkler (1996) points out the importance of using mixed-integer programming in (mine) production scheduling to account for fixed costs, logical conditions, sequencing constraints and machine use restrictions. Indeed, mixed integer programming cannot be ignored as an optimization tool if production block mining sequence is to be accounted for. However, Winkler de-
classes that the theoretical complexity of mixed-integer programs precludes their use for multiperiod mine scheduling and produces a schedule for underground coal mining operations for a single time period. His model minimizes fixed and variable extraction costs subject to quality constraints, and demonstrates that significant differences in solutions exist when fixed costs are ignored. Others (Smith 1988) in developing a model for a silver and gold surface mine, face the same intractability issues when using mixed integer programming. Trout (1995) actually formulates and attempts to solve a mixed-integer multiperiod production scheduling model for underground stoping operations for base metal (e.g., copper sulfide). Although the model produces a solution that is considerably better than what is now realized in practice, solution time exceeds 200 hours, without a guarantee of optimality.

In fact, several attempts at multiperiod scheduling using mixed-integer programming have been made at the Kumba Mine. So far, though, these have produced acceptable multiperiod production schedules. Specifically, Almgren (1994) provides the first attempt, Almgren divides the data into 100 (328-13) blocks (Fig. 3), where one to three of these blocks constitutes a machine placement. Several additional sets of binary variables account for whether a block, or a portion of a block, has been mined by a given time period. Continuous-valued variables track the amount of various ore types mined in given locations at a given time period. In general, because of the database from which model parameters are obtained and because of the resulting unnecessarily complicated formulation, the model is too large to be solved in its monolithic form. Hence, Almgren generates five-year schedules by running the model one month at a time for 60 monthly time periods. Schedules with acceptable deviations from the production demands can be produced in a few hours, although there is no guarantee of optimal schedule for the complete five-year period, especially because “end effects” between months are ignored.

Topal (1998) and Dageden et al. (1999) propose a modeling approach in which they derive the input data from the same database. However, their model contains fewer integer variables, specifically only binary variable representing the mined state of a block in a given time period. Their model also contains continuous-valued variables to track the amount of the various ore types in each production block that has been mined in each time period. Because, similarly, this model cannot be solved in its monolithic form, Topal (1998) and Dageden et al. (1999) reduce the number of integer variables by solving for yearly, rather than five-year, schedules. Each yearly model considers only those production blocks that could jointly be mined in the corresponding time frame and excludes those blocks that would already have been mined in a prior year. The authors subsequently obtain a monthly production schedule based on the yearly production schedule. However, there is no guarantee of an optimal schedule for the complete five-year period. Not, by first solving for a yearly schedule, is there a guarantee that production requirements can be met on a monthly basis. Cuddick and Cavers (2001) formulate a model in a vein similar to the one presented here for production planning of a sublevel stoping operation at Stillwater Mining. The model maximizes revenue from mining platinoids and palladium. The authors obtain near-optimal solutions but do not describe any special techniques to expedite solution time or improve solution quality.

The main advantage of the formulation presented here is that there is a substantial reduction in the number of variables compared to the previous models. Fewer variables results in a dramatic improvement in model tractability. A new database and careful formulation allows one to aggregate what may have been 12 production blocks into a single production block, in turn reducing the number of integer variables. When additional modifications are made (e.g., the addition of tightening constraints and eliminating variables using early and late start dates for machine placements) and the model is run with appropriate hardware and software, the model presented here produces an optimal schedule for a five-year time horizon in a matter of minutes.

Future research

Using robust methods, it should be possible to more precisely establish the latest possible start time for each machine placement. This, in turn, would further reduce the number of eligible time periods in which a block can be mined and, correspondingly, the number of binary variables.

Acknowledgments

The authors would like to thank LKAB for the opportunity to work on this challenging project and for permission to publish these results.

References


