Expert Systems With Applications 43 (2016) 59–65

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

New graph-based algorithms to efficiently solve large scale open pit mining optimisation problems

Shi Qiang Liu, Erhan Kozan*

Decision Science Discipline, Mathematical Sciences School, Queensland University of Technology, 2 George St GPO Box 2434, Brisbane, Qld 4001, Australia

ARTICLE INFO

Keywords:
Mine optimisation algorithms
Planning and scheduling
Mine block sequencing
Ultimate pit limit
Constrained pit limit
Network flow graph

ABSTRACT

In the mining optimisation literature, most researchers focused on two strategic-level and tactical-level open-pit mine optimisation problems, which are respectively termed *ultimate pit limit* (UPIT) or *constrained pit limit* (CPIT). However, many researchers indicate that the substantial numbers of variables and constraints in real-world instances (e.g., with 50–1000 thousand blocks) make the CPIT's mixed integer programming (MIP) model intractable for use. Thus, it becomes a considerable challenge to solve the large scale CPIT instances without relying on exact MIP optimiser as well as the complicated MIP relaxation/decomposition methods. To take this challenge, two new graph-based algorithms based on network flow graph and conjunctive graph theory are developed by taking advantage of problem properties. The performance of our proposed algorithms is validated by testing recent large scale benchmark UPIT and CPIT instances' datasets of MineLib in 2013. In comparison to best known results from MineLib, it is shown that the proposed algorithms outperform other CPIT solution approaches existing in the literature. The proposed graph-based algorithms lead to a more competent mine scheduling optimisation expert system because the third-party MIP optimiser is no longer indispensable and random neighbourhood search is not necessary.

1. Introduction

Modern mining is a complicated procedure that may sustain over several decades and necessitate huge investment in billions of dollars. To prepare for a feasibility study report at the exploration phase, a tentative strategic mine production plan/schedule should be optimised, that is: which part of orebody should be selected; and in which time period (when) the subset of blocks in this part should be extracted. The first of these questions is answered by the ultimate pit limit (UPIT) problem in the mining literature. As pioneers, Lerchs and Grossmann (1965) presented to the mining community a dynamic programming method known as the Lerchs–Grossmann approach for UPIT. Caccetta and Giannini (1988) proposed several mathematical theorems to improve the Lerchs–Grossmann approach. Underwood and Tolwinski (1998) developed a dual simplex approach to solve the UPIT model. Hochbaum and Chen (2000) presented a push-relabel algorithm for UPIT based on the network flow graph theory. Nowadays, the UPIT problem has been well defined and computationally tractable to be solved even for the very large UPIT instances in today's computer technology.

After the determination of the ultimate pit contour, the next widely-studied mine optimisation problem type is to answer the second question: when the blocks should be extracted over time periods so that the total net present value is maximised. In the mining community, this problem type is called mine production scheduling Bley, Boland, Fricke, & Froyland, 2010; Boland, Dumitrescu, Froyland, & Gleixner, 2009; Caccetta & Hill, 2003; Chicoisne, Espinoza, Goycoolea, Moreno, & Rubio, 2012), or open-pit block sequencing (Cullenbine, Wood, & Newman, 2011; Lambert, Brickey, Newman, & Eurek, 2014), or constrained pit limit (Espinoza, Goycoolea, Moreno, & Newman, 2013; MineLib, 2013). For convenience, we use the term **Constrained Pit Limit** (**CPIT**) to call this problem type in this paper. The following leading papers contribute to CPIT solution approaches in the mining optimisation literature. As a pioneer, Caccetta and Hill (2003) proposed a branch-and-cut algorithm embedded LP relaxation and MIP optimiser to solve CPIT. However, due to software commercialisation and confidentiality agreements, they only summarise some important features and the full details of all aspects of their proposed branch-and-cut algorithm are not provided in this paper. Ramazan (2007) proposed a "Fundamental Tree Algorithm" to aggregate the blocks for reducing the number of variables and constraints in the MIP model. Boland et al. (2009) developed a LP-based relaxation approach to solve large-size CPIT instances. Bley et al. (2010) improved

Abbreviations: UPIT, ultimate pit limit; CPIT, constrained pit limit; PCPSP, precedence constrained production scheduling problem; MineLib, a public online library of benchmark instances' data files and best known results of mine optimisation problems including UPIT, CPIT and PCPSP.

* Corresponding author. Tel.: +61 7 3138 1029; fax: +61 7 3138 2310.
E-mail addresses: sq.liu@qut.edu.au (S.Q. Liu), e.kozan@qut.edu.au (E. Kozan).

http://dx.doi.org/10.1016/j.eswa.2015.08.044
0957-4174/

the CPIT formulation by adding inequalities derived by generating the union predecessor set to replace the immediate predecessor set. Bienstock and Zuckerberg (2010) developed linear programming (LP) relaxation approaches to solve CPIT efficiently. Cullenbine et al. (2011) developed a sliding-time-window algorithm in which the relaxed CPIT formulation models are iteratively solved by MIP optimiser over divided time windows. Chicoisne et al. (2012) proposed a decomposition method to solve the relaxed CPIT formulation model period by period, in which there is a single capacity constraint per period. Then, the feasible solutions of more general CPIT formulation model with multiple capacity constraints are obtained by adding a rounding algorithm based on a topological sorting algorithm. Espinoza et al. (2013) presented a library (i.e., benchmark data and results of varied-size instances) of open-pit mining optimisation problems such as UPIT and CPIT to the mining community. Lambert and Newman (2014) employed a tailored Lagrangian relaxation to efficiently solve the CPIT formulation model. Lambert et al. (2014) concluded a tutorial of several CPIT mathematical formulation models developed in the literature.

The complexity of large scale CPIT problem and its variants led to development of numerous heuristic/metaheuristic algorithms. Kumral and Dowd (2005) developed a simulated annealing metaheuristic combined with Lagrangian relaxation. Ferland, Amaya, and Djuimo (2007) developed a particle swam optimisation metaheuristic for solving CPIT. Myburgh and Deb (2010) reported an application of evolutionary algorithm to solve CPIT, in which an initial feasible sequence of blocks represented as a chromosome is iteratively improved by genetic operators such as crossover and mutation. Souza, Coelho, Ribas, Santos, and Merschmann (2010) developed a hybrid heuristic approach for a CPIT-type problem with the consideration of operational constraints such as truck allocations. Martinez and Newman (2011) developed a heuristic decomposition scheme to efficiently obtain satisfactory CPIT solutions in a real-world implementation. Lamghari and Dimitrakopoulos (2012) presented a tabu search to solve a CPIT-type problem with the consideration of metal uncertainty. Alonso-Ayuso et al. (2014) developed a heuristic approach to solve a stochastic CPIT model with the consideration of ore prices. Lamghari, Dimitrakopoulos, and Ferland (2015) developed a two-phase approach to solve the CPIT problem, in which the first phase is to generate the initial solution by a series of linear programming models and the second phase is to apply a variable neighbourhood search procedure to improve the initial solution. Shishvan and Sattarvand (2015) developed an ant colony optimisation (ACO) metaheuristic to solve an extended CPIT-type problem applied in a copper–gold mine.

According to the above literature review, the following five ways in solving large scale CPIT instances are concluded and categorised below:

i. reduce the full model size by aggregating the blocks and periods;
ii. relax the full model complexity by decreasing the number of variables or constraints;
iii. decompose the full model into several sub-models so that much less number of constraints and variables become tractable;
iv. embed heuristics within MIP optimiser to accelerate the solution procedure; and
v. develop metaheuristics with neighbourhood search.

However, the above first four ways still rely on the use of third-party MIP optimiser software and sophisticated relaxation/decomposition approaches. The fifth way by the development of metaheuristics such as generic algorithm (i.e., neighbourhood search with random diversification mechanisms) may be not advanced enough because there exists unexpected randomness in the solution procedure and the critical CPIT problem's structural properties are not utilised. Hence, the purpose of this study is to develop new graph-based algorithms to outperform the existing CPIT solution approaches in the literature.

The development of advanced mining optimisation approaches is an active research topic in expert and intelligent systems, especially for Australian mining industry. Main Australian mining companies such as BHP Billiton, Rio Tinto, Xstrata and OZ Minerals, are keen to adopt expert systems (e.g., commercialised mining software such as Whittle Gemcom's strategic planning software; XPAC's mine block sequencing software; and Modular's truck fleet dispatching software) for improving their mining management systems. However, in our recent visit to Australian mine sites, we observed that these commercialised expert systems still lack the advanced solution approaches in optimisation engine. As the CPIT problem is NP-hard, the required computational time of a MIP exact optimiser is increased exponentially. For solving large scale CPIT instances without any relaxation/decomposition schemes, a MIP exact optimiser such as IBM ILOG-CPLEX cannot be implemented due to memory overflow or unacceptable computational effort. Another practical reason is that the mining company is not willing to buy the third-party optimiser because the licence for commercial use is costly. To fill this gap, this study contributes to extend the boundaries of developing innovative numerical methods to solve large scale mining optimisation problems in a more efficient and effective way.

The remainder of this paper is outlined as follows. In Section 2, five lemmas on problem properties and the detailed procedures of two new algorithms are presented. In Section 3, the computational results of benchmark UPIT and CPIT instances obtained by the proposed algorithms are reported and compared to the best known results in MineLib. In the last section, we conclude the contribution and significance of this research in the last section.

2. New algorithms

The following two fundamental mathematical programming models are given for showing the objective function and main constraints of the UPIT and CPIT problems respectively.

UPIT Model

Objective:

$$\text{Maximise}: \quad \sum_{b\in\mathcal{B}} x_b p_b \tag{1}$$

Subject to:

$$x_b \le x_{b'}, \quad \forall b \in \mathcal{B};\, b' \in \Psi_b | \Psi_b \subset \mathcal{B} \tag{2}$$

$$x_b \in \{0, 1\}, \quad \forall b \in \mathcal{B}; \tag{3}$$

where x_b is a binary decision variable that equals 1 if block b is selected; p_b is the value (positive or negative) if block b is to be mined; $\mathcal{B}$ is the set of total blocks for the whole orebody; Ψ_b is the subset of blocks that are the immediate predecessors of block b. Constraint (2) ensures that each block should be extracted after its predecessors. Constraint (3) defines that decision variables are binary.

CPIT Model

Objective:

$$\text{Maximise}: \quad \sum_{b\in\mathcal{B}} \sum_{t\in T} (y_{bt} - y_{b,t-1}) p_{bt} \tag{4}$$

Subject to:

$$y_{b,t-1} \le y_{bt}, \quad \forall b \in \mathcal{B};\, t \in T \tag{5}$$

$$y_{bt} \le y_{b't}, \quad \forall b \in \mathcal{B};\, b' \in \Psi_b | \Psi_b \subset \mathcal{B};\ t \in T \tag{6}$$

$$R_{rt}^{min} \le \sum_{b\in\mathcal{B}} (y_{bt} - y_{b,t-1}) u_{br} \le R_r t^m ax, \quad \forall t \in T;\, r \in \mathcal{R} \tag{7}$$

$$y_{bt} \in \{0, 1\}, \quad \forall b \in \mathcal{B};\ t \in T \tag{8}$$

$$y_{b0} = 0, \quad \forall b \in \mathcal{B} \tag{9}$$