Computers & Industrial Engineering 110 (2017) 264–276

Contents lists available at ScienceDirect

Computers & Industrial Engineering

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

A max-min ant system for the finance-based scheduling problem

Sameh T. Al-Shihabi [a], Mohammad M. AlDurgam [b,*]

[a] Industrial Engineering Department, The University of Jordan, Amman 11942, Jordan
[b] Systems Engineering Department, King Fahd University of Petroleum & Minerals, P.O. Box 1743, Dhahran 31261, Saudi Arabia

ARTICLE INFO

Article history:
Received 4 September 2016
Received in revised form 30 May 2017
Accepted 10 June 2017
Available online 11 June 2017

Keywords:
Ant colony optimization
Man-min ant system
Finance-based scheduling
Project scheduling
Cash flow

ABSTRACT

Construction contractors depend on bank overdrafts to finance their expenses; however, these overdrafts cannot exceed an imposed Credit Line (CL). The Finance-Based Scheduling Problem (FBSP) is about scheduling activities without exceeding the CL. In this paper, we provide a more eloquent formulation of the FBSP and list its different variants. Three Max-Min Ant System (MMAS) algorithms, which use different heuristic information when generating solutions, are then developed to solve the FBSP. To test the MMAS algorithms, we generate 60 instances that are used to tune the MMAS algorithms and then use these algorithms to solve the generated instances. The found solutions are compared with the best bounds found using a Branch and Bound (B&B) algorithm. A 0.6% improvement is achieved by the B&B algorithm when compared to the best results found by the MMAS algorithms; moreover, the comparison shows that using the number of successors as heuristic outperformed other heuristics. Furthermore, the MMAS algorithm outperformed other meta-heuristics that use repair operators or penalize infeasible solutions in terms of computation time while having comparable solution values.

1. Introduction

Project scheduling is about sequencing activities so that the activities' execution times satisfy a set of precedence relations and resource limits while maximizing or minimizing a presumed objective function, e.g., minimizing the project duration (Davis, 1973). In practice, however, and from a contractor's perspective, project success needs to be translated into a financial reality, which largely depends on managing the project cash flows (Russell, 1970). Not having enough liquidity is a common problem that faces contractors, as explained in Singh and Lokanathan (1992). This lack of cash forces contractors to delay the execution of some tasks; consequently, contractors cannot finish their projects on time and usually need to pay lateness fees. Indeed, it was reported in Russell (1991) that 60% of construction contractors' failures are due to financial problems. Similar conclusions are found in Kangari (1988) and Pate-Cornell, Tagaras, and Eisenhardt (1990). Thus, forecasting and managing cash flows are highly critical in any project (Barbosa & Pimentel, 2001; Kaka & Price, 1991; Khosrowshahi & Kaka, 2008).

Managing a project cash flow is not an easy task due to many factors such as the long duration of construction projects, cash retainage from the clients, cost estimates and efficiency problems (Park, Han, & Russell, 2005). Contractors do not wait until the end of the project to receive their full bid amount; instead, they are partially paid as the project progress. A reimbursement period is chosen in the contract, and contractors submit invoices that cover the work executed during each reimbursement period. Clients do not pay these bills in full; instead, they retain a certain percentage, Retainage Percentage (RP), of the invoices to be paid at the end of the project. Moreover, contractors do not receive their money instantly; they need to wait for the next reimbursement period to be paid. A typical cash flow of a construction company is shown in Fig. 1 (Hendrickson, 2008).

Fig. 1 shows that a contractor has a negative cash balance throughout the project, except when the project is closed and the sum of the retained amounts, from all the previous reimbursement periods, is paid. The contractor pays the expenditures by having an account with a bank so that the contractor withdraws money from this account but needs to deposit money back into the same account to remain below a Credit Limit (CL) that is approved and imposed by the bank (Ahuja, 1976). By maintaining a CL, we mean that the negative cash flow will never exceed this limit. Thus, it is important to manage cash inflows and outflows in order not to exceed the bank CL, as shown in Elazouni and Gab-Allah (2004). Henceforth, we will denote the condition of keeping the bank balance below the CL by the term CL-constraint.

The problem of scheduling project activities subject to the CL-constraint is referred to as the Finance-Based Scheduling Problem (FBSP) in Elazouni and Gab-Allah (2004), where an Integer Pro-

* Corresponding author.
E-mail addresses: s.shihabi@ju.edu.jo (S.T. Al-Shihabi), aldurgam@kfupm.edu.sa (M.M. AlDurgam).

http://dx.doi.org/10.1016/j.cie.2017.06.016
0360-8352/

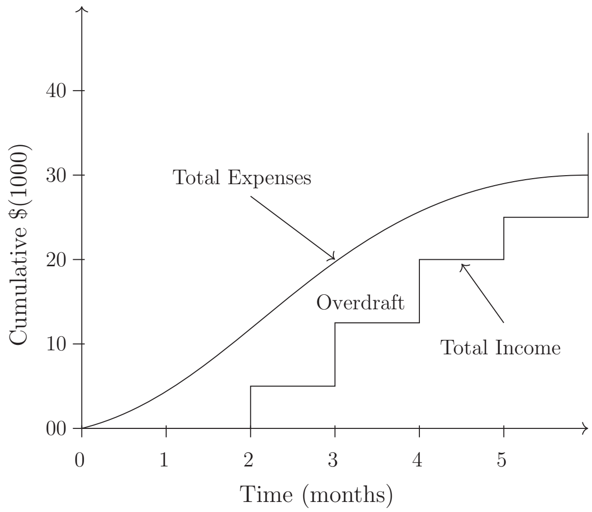

Fig. 1. Typical expense versus income profile.

gramming (IP) formulation is introduced to model the FBSP. As shown in Elazouni and Gab-Allah (2004), the actual start times of activities might be equal to or greater than their start times as found by the Critical Path Method (CPM) schedule (Kelly, 1961), which ignores the CL-constraint. This delay -also called extension throughout this paper- is needed to satisfy the CL-constraint. In other words, the contractor will delay the execution start times of some activities until enough money is accumulated in the project's bank account.

The FBSP might appear as a variant of the Resource Constrained Project Scheduling Problem (RCPSP) (Özdamar & Ulusoy, 1995); however, resources in the RCPSP are classified into three types, according to Özdamar and Ulusoy (1995):

1. Renewable resources, which are constrained on a period-by-period basis. Labor is a typical example of a renewable resource since it is used on a daily basis and there is a fixed amount of it every day.
2. Non-renewable resources, which are constrained on a project basis. A project budget is an example of this type of constraint since a limited budget is available for the whole project.
3. Doubly-constrained resources, which are constrained by both period and project bases. Daily cash expenditures can be limited for the whole project and a daily cap can be imposed.

In the FBSP, the CL-constraint does not belong to any of the previous three classifications. The CL is not a non-renewable resource because the limit is imposed on a daily basis, not on a project basis. The CL might appear as a renewable resource since it has a fixed value during the project life; however, the actual cash available to spend changes from one period to the other based on the actual cash balance in the bank account. Classifying the daily cash expenditures as doubly-constrained resources in RCPSP ignores the dynamics of cash flows: it only considers expenditures without regard to payments and bank balance.

A Max-Min Ant System (MMAS) algorithm (Stützle & Hoos, 2000) that has three variants to solve the FBSP is suggested in this paper. These variants differ in the type of heuristic information used when generating solutions. As a set of test instances, we generate 60 instances by modifying instances that are used to benchmark algorithms targeting the (RCPSP). This is done by imposing a CL to the projects and ignoring other types of resources, except for the precedence relations. These instances are then solved using the three MMAS variants and their solutions are compared to the bounds found using the default Branch and Bound (B&B) algorithm of CPLEX12.6. We also compare our MMAS algorithm with other meta-heuristic algorithms by solving the already published instances Elazouni, Alghazi, and Selim (2015).

In summary, the contributions of this work are threefold. First, we enhance the IP model suggested in Elazouni and Gab-Allah (2004). Second, we introduce a benchmarking set that has 60 instances that researchers can use to compare algorithms targeting the FBSP. Third, we solve the newly generated instances using the three MMAS algorithms; and to benchmark the MMAS algorithms, the same instances are solved using an exact method. We also solve a limited number of instances used previously by researchers to compare our MMAS algorithm to other meta-heuristics.

The rest of the paper is organized as follows. In the next section, we discuss the literature related to the FBSP, while we present the IP model in Section 3 and our MMAS algorithms in Section 4. All of the conducted experiments are covered in Section 5. We end this paper with our conclusions and suggestions for future work.

2. Literature review

Researchers have suggested numerous optimization models in which the cash-flow management issues are incorporated. These models had different degrees of modeling complexity: Linear Programming (LP), as in Barbosa and Pimentel (2001); (IP), as in Elazouni and Gab-Allah (2004); and Mixed Integer Nonlinear Programming (MINLP), as in Chiu and Tsai (2002). Researchers have also considered different objectives to maximize or minimize, e.g., profit maximization, Liu and Wang (2008) and Barbosa and Pimentel (2001); Net Present Value (NPV) maximization, Elazouni and Metwally (2005); and project completion time minimization, Elazouni and Gab-Allah (2004). Researchers have also studied the case of multi-objective functions, e.g., Liu and Wang (2009), Fathi and Afshar (2010), Jiang, Issa, and Malek (2011a) and Elazouni and Abido (2014). Moreover, researchers did not ignore the practical problem of managing a portfolio of projects (Elazouni, 2010; Liu & Wang, 2010).

To solve these models, researchers have tried a myriad of optimization techniques. Thus, researchers have suggested using exact methods in Barbosa and Pimentel (2001), Elazouni and Gab-Allah (2004) and Chiu and Tsai (2002), to solve the LP, IP, and MINLP models, respectively. Researchers have also used Constraint Programming (CP) in Liu and Wang (2008), while the majority of research efforts were concerned with using heuristics and meta-heuristics for solving the FBSP, which is an NP-hard problem according to Alghazi, Selim, and Elazouni (2012). A Genetic Algorithm (GA) was used in Abido and Elazouni (2009) to solve the FBSP which was later improved in Alghazi, Elazouni, and Selim (2013). In Alghazi et al. (2012) and Elazouni et al. (2015), a Shuffled Frog Leaping Algorithm (SFLA) is compared to a GA and a Simulated Annealing (SA) algorithm to solve the FBSP, considering the maximization of the NPV and minimization of the project duration, respectively.

The drawback with the three meta-heuristics compared in Alghazi et al. (2012) and Elazouni et al. (2015), namely, SA, GA, and SFLA, is that infeasible solutions might result due to the mutation and crossover operations in the GA algorithm or to the neighborhood search using swapping and insertion in the SA and SFLA algorithms. Consequently, infeasible solutions generated while executing these algorithms were either penalized or repaired to make them feasible. More execution time is needed to repair infeasible solutions; however, and as shown in Elazouni et al. (2015), the quality of the results obtained by repairing infeasible solutions is better than the ones found by penalizing infeasible solutions. Alternatively, in this research, we try to use a constructive meta-