Expert Systems with Applications 41 (2014) 1830–1846

Contents lists available at ScienceDirect

Expert Systems with Applications

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

A new multi-objective multi-mode model for solving preemptive time–cost–quality trade-off project scheduling problems

Madjid Tavana [a,b,*], Amir-Reza Abtahi [c], Kaveh Khalili-Damghani [d]

[a] Business Systems and Analytics Department, Lindback Distinguished Chair of Information Systems and Decision Sciences, La Salle University, Philadelphia, PA 19141, USA
[b] Business Information Systems Department, Faculty of Business Administration and Economics, University of Paderborn, D-33098 Paderborn, Germany
[c] Department of Knowledge Engineering and Decision Sciences, University of Economic Sciences, Tehran, Iran
[d] Department of Industrial Engineering, South-Tehran Branch, Islamic Azad University, Tehran, Iran

ARTICLE INFO

Keywords:
Project scheduling problem
Discrete time–cost trade-off problem
Multi-objective evolutionary algorithm
Activity preemption
Generalized precedence relations

ABSTRACT

Considering the trade-offs between conflicting objectives in project scheduling problems (PSPs) is a difficult task. We propose a new multi-objective multi-mode model for solving discrete time–cost–quality trade-off problems (DTCQTPs) with preemption and generalized precedence relations. The proposed model has three unique features: (1) preemption of activities (with some restrictions as a minimum time before the first interruption, a maximum number of interruptions for each activity, and a maximum time between interruption and restarting); (2) simultaneous optimization of conflicting objectives (i.e., time, cost, and quality); and (3) generalized precedence relations between activities. These assumptions are often consistent with real-life projects. A customized, dynamic, and self-adaptive version of a multi-objective evolutionary algorithm is proposed to solve the scheduling problem. The proposed multi-objective evolutionary algorithm is compared with an efficient multi-objective mathematical programming technique known as the efficient ε-constraint method. The comparison is based on a number of performance metrics commonly used in multi-objective optimization. The results show the relative dominance of the proposed multi-objective evolutionary algorithm over the ε-constraint method.

1. Introduction

Project scheduling problems (PSPs) have received significant attention and played a vital role in managing organizational resources. A PSP is defined by its activities (each with a specific execution time) and by the precedence relations among them. The overall goal in project scheduling is to optimize a set of measurement functions subject to a set of precedence and resource constraints (Singh & Ernst, 2011). PSPs are some of the most intractable problems in operations research, and have therefore become a popular playground for the latest optimization techniques (Baptiste & Demassey, 2004; Möhring, Schulz, Stork, & Uetz, 2003). PSPs are complex scheduling problems with limited resources. This extension of the PSP is called the resource-constrained PSP (RCPSP). Two types of constraints are usually present in a RCPSP: precedence constraints and resource constraints. Precedence constraints establish a specific sequence for some pairs of activities and resource constraints model the resource requirements of the activities assuming a limited resource supply (Correia, Lourenço, & Saldanha-da-Gama, 2012). Evolutionary optimization approaches such as particle swarm optimization (PSO) and genetic algorithms (GA) have been successfully used to solve the RCPSPs (Chen, 2011; Chen, Wu, Wang, & Lo, 2010; Hartmann, 2002; Van Peteghem & Vanhoucke, 2010.

In project scheduling, it is often possible to reduce the duration of some activities and thereby expedite the project duration with some additional costs. Project expedition decisions has traditionally involved time and cost trade-off considerations. However, it was recently suggested that the quality of a project should also be taken into consideration (Iranmanesh, Skandari, & Allahverdiloo, 2008; Tareghian & Taheri, 2006).

In the continuous trade-off problems, there are functions which correlate the time, cost, and quality objectives. As research efforts progressed in the field and practical needs arose, researchers began to focus on the development of procedures for solving the discrete time–cost trade-off problems (DTCTPs) (Hazır, Erel, & Günalay, 2011; Sonmez & Bettemir, 2012; Wuliang & Chengen, 2009; Xu, Zheng, Zeng, Wu, & Shen, 2012). In the discrete variant, the relationships between the objectives in a project are defined at discrete points. In this case, each activity can be executed in several

---

* Corresponding author at: Business Systems and Analytics Department, Lindback Distinguished Chair of Information Systems and Decision Sciences, La Salle University, Philadelphia, PA 19141, USA. Tel.: +1 215 951 1129; fax: +1 267 295 2854.

E-mail addresses: tavana@lasalle.edu (M. Tavana), amir_abtahi@yahoo.com (A.-R. Abtahi), kaveh.khalili@gmail.com (K. Khalili-Damghani).

URLs: http://tavana.us/ (M. Tavana), http://www.kaveh-khalili.webs.com (K. Khalili-Damghani).

0957-4174/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2013.08.081

modes. Hence, the feasible solution space of the problem exponentially increases for medium and large size problems. These trade-off problems are known as non-deterministic polynomial-time hard (NP-Hard) (De, Dunne, Ghosh, & Wells, 1997).

DTCTP is inherently difficult to solve (Tareghian & Taheri, 2007). Prabudha, Dunne, Ghosh, and Wells (1995) have offered two reasons why interest in DTCTP should be revived. “First, discrete alternatives are common in practice and second discretization provides a convenient means for modeling any general time/cost relationship”. Several exact and approximation procedures are developed for small-size trade-off problems (Burns, Liu, & Feng, 1996; Demeulemeester, De Reyck, & Herroelen, 2000; Skutella, 1998; Sunde & Lichtenberg, 1995). The solution procedures to the DTCTPs are classified into three groups: (a) Exact algorithms, such as linear programming, integer programming, dynamic programming, branch-and-bound algorithms, etc. (Erenguc, Ahn, & Conway, 2001), (b) Heuristic algorithms (Vanhoucke, Debels, & Sched, 2007), and (c) Meta-heuristic algorithms (Afshar, Kaveh, & Shoghli, 2007; Azaron, Perkgoz, & Sakawa, 2005; Chao-guang, Shang, Yan, Yuan-min, & Zhen-dong, 2005; El-Rayes & Kandil, 2005; Wuliang & Chengen, 2009; Yang, 2011; Zhang & Xing, 2010).

Szmerekovsky and Venkateshanb (2012) proposed four integer programming formulations for the irregular costs PSP with time–cost trade-offs. Three formulations using the standard assignment type variables were tested against a more novel integer programming formulation. Their empirical tests showed that in many instances the new formulation performed best and could solve problems with up to 90 activities in a reasonable amount of time.

Heuristic and Meta-heuristic algorithms represented better results for medium and large-size trade-off problems (Rahimi & Iranmanesh, 2008). Meta-heuristics have also been successfully used to solve multi-objective trade-off problems. Other assumptions such as time-switch constraints have been introduced in the literature on the trade-off problems (Vanhoucke, 2005). Table 1 represents some relevant studies on the trade-off problems in the literature.

Real world tasks, including project activities and job scheduling, can be either preemptable or non-preemptable (Błażewicz, Ecker, Pesch, Schmidt, & Węglarz, 2007). An activity is preemptable if it can be preempted at any time and restarted later with no cost (Demeulemeester & Herroelen, 2002). Preemption may be either discrete or continuous. In discrete case, preemption is allowed at the end of the time period while in continuous preemption, activity preemption may occur at an arbitrary time instant. Considering the assumption that preemptive activities can be preempted at integer time instants and restarted later at no additional cost, an activity is split into a number of sub-activities with unit duration. This type of preemption was first introduced by Kaplan (1988) in preemptive RCPSPs. Kaplan (1988) used dynamic programming to formulate the preemptive RCPSP and showed that this type of preemption has no meaningful effect on project lengths when constant resource availability levels are defined and the exact procedures are used. Demeulemeester and Herroelen (1996) later improved the formulation of Kaplan (1988) through a branch-and-bound procedure. Preemption may occur subject to a maximum limitation. Although allowing activity interruption may reduce the duration of a project, the repeated stopping and starting of an activity may not be feasible in practice (Ballestín, Valls, & Quintanilla, 2008).

Lino (1997) conducted an extensive set of experiments on scheduling of projects with generalized precedence relations and no resource constraints. Lino (1997) considered three different assumptions for modeling preemption as follows: (a) no interruption, (b) any number of interruptions at integer time instants, and (c) a maximum of one interruption per activity. Lino (1997) used an extensive number of randomly generated instances and showed that if each activity is allowed to be interrupted just once, then a significant reduction in project length is obtained in comparison with the case of no interruption. He also detected that allowing more than one interruption instead of a maximum of one interruption per activity does not further reduce the project length in the majority of the instances – and that when a reduction happens it is very small.

In recent years, the applications of the RCPSP and its extensions have attracted increasing interest from researchers and practitioners (Demeulemeester & Herroelen, 2002). One of the extensions of the RCPSP that has received considerable attention has been the RCPSP with preemption. Yang and Chen (2000) investigated one type of time constraint called a time-switch constraint which assumes that an activity begins at a specific time interval in a cycle with some pairs of exclusive components. Yang and Chen (2000) developed polynomial time algorithms to find the longest path (or critical path) and analyzed the float of each arc in this time-constrained activity network. Their analysis showed that the critical path and float in the time-constrained activity networks differ from those of the traditional activity networks and the consideration of the time-switch constraints lead to effective use of budgets and resources.

Table 1
A summary of the studies on the TCQT problem.

Author(s)	Method	Main contributions
Babu and Suresh (1996)	Linear programming	Using three inter-related linear programming models and extending them into non-linear models
Khang and Myint (1999)	Linear programming	Applying the Babu and Suresh (1996) method to an actual cement factory and examining the method’s applicability, assumptions and limitations
El-Rayes and Kandil (2005)	Genetic algorithm	Applying their model to highway construction projects; quantifying quality with some quality indices and calculating the project quality based on an additive weighting method
Tareghian and Taheri (2006)	Integer programming	Developing a method to prune the activity execution modes
Pollack-Johnson and Liberatore (2006)	Goal programming	Conceptualizing the quality in projects; Quantifying the quality value of each activity execution mode with the Analytic Hierarchy Process (AHP) and developing a goal programming model with four objectives including: time, cost, minimum quality and mean of quality
Tareghian and Taheri (2007)	Electromagnetic scatter search	Validating and checking the applicability of their algorithm by solving a randomly generated large-scale problem with 19900 activities
Afshar et al. (2007)	Multi-colony ant algorithm	Solving an example and comparing their algorithm’s results with some other algorithms
Zhang and Xing (2010)	Particle swarm optimization	Considering construction methods instead of execution modes for each activity; Using fuzzy numbers to describe time, cost, and quality; Using fuzzy multi-attribute utility methodology and constrained fuzzy arithmetic operators to evaluate each construction method; Demonstrating the effectiveness of their algorithm by solving a bridge construction problem
Kim, Kang, and Hwang (2012)	Mixed integer linear programming	Focusing on minimizing quality loss cost instead of maximizing the individual activity quality of the projects; Validating their model by applying it to a robot type palletizing system installation project