Automation in Construction 27 (2012) 89–98

Contents lists available at SciVerse ScienceDirect

Automation in Construction

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

Hybrid meta-heuristic methods for the multi-resource leveling problem with activity splitting

Hadeel Alsayegh, Moncer Hariga *

Engineering Systems Management Graduate Program, College of Engineering, American University of Sharjah, P O Box 26666, Sharjah, United Arab Emirates

ARTICLE INFO

Article history:
Accepted 16 April 2012
Available online 13 June 2012

Keywords:
Resource leveling
Activity splitting
Meta-heuristics
Particle swarm optimization
Simulated annealing

ABSTRACT

In this paper, we consider the multi-resource leveling problem with the objective of minimizing the total costs resulting from the variation of the resource utilization and the cost of splitting non-critical activities. We propose hybrid meta-heuristic methods which combine particle swarm optimization (PSO) and simulated annealing (SA) search procedures to generate near-optimal project schedules in less computational time than the exact optimization procedure. The PSO algorithms are based on different update mechanisms for the particles' velocities and positions. The cost and computation time performances of the combined PSO/SA search procedures are evaluated using a set of benchmark problems. Based on the results of the computational experiments, we suggest one of the proposed heuristic procedures to be used for solving the multi-resource leveling problem with activity splitting.

1. Introduction

Resource leveling is a resource management technique to minimize total deviations of resource requirements over a fixed project duration. Most resource leveling models proposed in the literature assume that activities may not be split, indicating that once an activity starts, it will continue until the work is finished. However, in practice, it may be cost effective to interrupt an activity to release its resources and assign them to other activities (Karaa and Nasr [7]).

By allowing activities to be interrupted, the resulting resource leveling optimization model becomes mathematically complex as it introduces more decision variables and constraints (Hariga and El-Sayegh [4]). Furthermore, additional costs should be included in the objective function such as resource and activity dependent costs, which are incurred as a result of splitting activities. A resource-dependent cost involves the costs of acquiring or releasing the resource in a given period. On the other hand, activity-dependent costs are related to the stopping and restarting of the activity. Upon splitting an activity, the learning process of the resources is affected, and it will take some time for the resources to re-achieve the learning level just prior to splitting the activity. For more details about these two types of costs, interested readers can refer to Hariga and El-Sayegh [4].

Although extensive research works have been carried out on resource leveling without splitting, very little research is found on resource leveling with activity splitting. Hariga and El-Sayegh [4] developed a mathematical model having cost based objective function rather than the traditional utilization function for the multi-resource leveling problem with activity splitting. They formulated the problem as a mixed integer linear program model. Although their model is guaranteed to generate optimal project schedules, it requires a lot of computational effort for a large number of non-critical activities. Therefore, the purpose of this paper is to propose a cost and computational time efficient heuristic procedure for the resource leveling problem with activity splitting.

The remainder of the paper is organized as follows. In Section 2, we review the literature relevant to the resource leveling problem with activity splitting. In Section 3, we discuss the proposed PSO–SA search procedures. In Section 4, we present the experimental framework and the computation results. Finally, in Section 5 we conclude the paper.

2. Literature review

Both resource allocation and resource leveling problems have been challenging topics of extensive research in the project management area. Several optimization techniques have been used to solve such problems, which can be classified as exact procedures, heuristic procedures, and meta-heuristic procedures. In the following, we review the research works which are relevant to the problem addressed in this paper.

Harris [5] was the first to develop a simple heuristic procedure called the Minimum Moment algorithm for the resource leveling problem without activity splitting. Later, Hiyassat [6] modified Harris's procedure by considering the value of the activity's free float and its resource rate in the selection criterion. Easa [2] presented the first optimization model for the resource leveling problem. The objective of Easa's model

* Corresponding author.
E-mail addresses: g00034580@aus.edu (H. Alsayegh), mhariga@aus.edu (M. Hariga).

0926-5805/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.autcon.2012.04.017

is to minimize the deviations between the actual and desirable resource rates. Also, Ramlogan and Goulter [15] proposed a mixed integer model to level resources for project scheduling. Bandelloni et al. [1] followed the dynamic programming approach to level resources. Mattila and Abraham [14] formulated an integer linear programming model to smooth resources' usage for linear projects, which are characterized by having a set of activities that are repeated in different locations. Son and Matilla's paper [17] is one of the earliest papers in which activity splitting is allowed. Recently, Hariga and El-Sayegh [4] formulated as a mixed binary–integer programming to minimize the costs associated with the splitting of the non-critical activities.

In addition to the above exact formulations for the resource leveling problem, several authors proposed meta-heuristic procedures to generate near-optimal schedules. Senouci and Eldin [16] proposed a model based on genetic algorithm (GA) for resource scheduling. This model performs resource leveling along with resource allocation simultaneously. Son and Skibniewski [18] developed a technique for resource leveling, based on a local optimizer procedure and a hybrid procedure, with the objective of minimizing the difference between the required resources and the desired resource profile. In a recent paper, Liao et al. [13] provided a comprehensive review of previous research works using meta-heuristics to address project management problems and issues. Leu et al. [12] proposed a GA based optimization system to minimize the weighted total deviations of resources' requirements. Leu and Hung [11] solved the stochastic resource leveling problem using a combined simulation-GA model. A GA-based system was also used by Georgy [3] to perform resource leveling for linear projects. As it can be noticed from the review of approximate solution methods, none of the cited papers utilizes PSO based meta-heuristic procedures. Therefore, to the best of our knowledge, our paper is the first to propose an integrated meta-heuristic search procedure by combining PSO with simulated annealing for the multi-resource leveling problem with activity splitting (MRLP-AS) and considering a cost instead of a utilization based objective function.

3. PSO–SA method for MRLP-AS

In this section, we first review the concepts and techniques underlying different versions of PSO procedures. We then show how to implement PSO to the resource leveling problem with activity splitting. Stand-alone and combined PSO and SA search procedures are also presented in this section.

3.1. Review of PSO search procedures

Particle Swarm Optimization, as developed by Kennedy and Eberhart [9], is based on the social behaviors of animals and insects, such as bird flocks and fish schools. Swarms, or groups, of these animals and insects, tend to self-organize themselves in optimal spatial patterns. Their behaviors, such as speed and direction, are determined through the exchange of information between individuals, called particles. Each particle is represented by its position in an N-dimensional space and its velocity. The velocity corresponds to the speed and direction at which the particle is moving.

In the classical PSO method, the particles and velocities are represented by real numbers. Each particle, from a population of P particles in the swarm, is initialized with a random position and velocity. Next, the PSO procedure searches iteratively for the best position (near or optimum) by updating each particle's velocity and position using its own previous best position (cognitive learning) and best position of all particles (social learning). The local and global best positions are determined through the assessment of each particle's fitness values. The search continues until convergence which is attained either when the allowed maximum number of iterations, K, is exceeded or a relatively steady position is reached.

The particle's position and velocity are denoted by $X_i(k)$ and $V_i(k)$, for $i=1, 2, \ldots, P$ and $k=1, 2, \ldots, K$. The N-dimensional position for the ith particle at the kth iteration is represented by

$$X_i(k) = [x_{i1}(k), x_{i2}(k), \ldots, x_{iN}(k)]$$

where, $x_{ij}(k)$ represents the jth coordinate of the ith particle for $j=1,2, \ldots, N$.

Similarly, the velocity for the ith particle at the kth iteration is represented by

$$V_i(k) = [v_{i1}(k), v_{i2}(k), \ldots, v_{iN}(k)].$$

The updating mechanism of the ith particle's velocity and position at the kth iteration is performed using the following two equations, respectively

$$\begin{aligned} V_i(k) &= wV_i(k-1) + c_1 r_1 \left[X_1^L - X_i(k-1)\right] + c_2 r_2 \left[X^G - X_i(k-1)\right] \\ X_i(k) &= V_i(k) + X_i(k-1) \end{aligned} \tag{1}$$

where,

X_i^L is the local best position of the ith particle found after $(k-1)$ iterations.

X^G is the global best position among all particles in the swarm visited so far.

w is the inertia weight used to reduce the impact of previous velocities on the current velocity so that it does not go out of control.

c_1 and c_2 are two positive parameters representing the cognition and social learning factors, respectively. If c_1 is large, then the particles tend to move toward their own local best. On the other hand, if c_2 is large, then the particles tend to move toward the known global best so far.

r_1 and r_2 are random numbers between 0 and 1.

The velocity of any particle is restricted in the interval $[V_{\min}, V_{\max}]$. If the new velocity is smaller than $V_{\min}$, then it is set to $V_{\min}$. Similarly, if the new velocity is larger than $V_{\max}$, then it is set to $V_{\max}$.

Kennedy and Eberhart [8] modified their PSO method to handle binary variables. In this case, the velocities of the particle no longer represent the speed but rather represent either the probability of a position changing its value to one or the probability of a position being 0. Thus, the values of the velocities are restricted to the interval [0, 1].

In discrete PSO, the particle's velocity is updated in the same way as in the continuous PSO. However, a normalization function is used to transform the real numbers to binary numbers. This is done using the following sigmoid function:

$$v_{ij}(k) = sig\left(v_{ij}(k)\right) = \frac{1}{1+e^{-v_{ij}(k)}}. \tag{2}$$

The velocity is then used to update the position of the particle using the following equation:

$$x_{ij}(k+1) = \left\{ \begin{array}{l|c} 1 & \text{if } \ r_{ij} < sig(v_{ij}(k+1)) \\ 0 & \text{otherwise} \end{array} \right\} \tag{3}$$

where, r_{ij} is a random number between [0,1].

Yang et al. [19] presented another version of discrete PSO, which is based on quantum theory. In the quantum theory, the quantum particle position, $X_{ij}(k)$, consists of bits, where each bit holds the values of 0 or 1. A quantum particle vector, $V_i(k)$ denotes the particle's velocity, v_{ij}, which represents the probability that the jth bit of the ith