Automation in Construction 29 (2013) 161–172

Contents lists available at SciVerse ScienceDirect

Automation in Construction

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

The Resource Leveling Problem with multiple resources using an adaptive genetic algorithm

Jose Luis Ponz-Tienda [a], Víctor Yepes [b,*], Eugenio Pellicer [c], Joaquin Moreno-Flores [a]

[a] School of Building Eng., Universitat Politècnica de València, Camino de Vera sn, 46022, Valencia, Spain
[b] ICITECH, Universitat Politècnica de València, Camino de Vera sn, 46022, Valencia, Spain
[c] School of Civil Eng., Universitat Politècnica de València, Camino de Vera sn, 46022, Valencia, Spain

ARTICLE INFO

Article history:
Accepted 3 October 2012
Available online 29 October 2012

Keywords:
Project scheduling
Resource leveling
Genetic algorithms
Benchmarking

ABSTRACT

Resource management ensures that a project is completed on time and at cost, and that its quality is as previously defined; nevertheless, resources are scarce and their use in the activities of the project leads to conflicts in the schedule. Resource leveling problems consider how to make the resource consumption as efficient as possible. This paper presents an Adaptive Genetic Algorithm for the Resource Leveling Problem, and its novelty lies in using the Weibull distribution to establish an estimation of the global optimum as a termination condition. The extension of the project deadline with a penalty is allowed, avoiding the increase in the project criticality. The algorithm is tested with the Project Scheduling Problem Library PSPLIB. The proposed algorithm is implemented using VBA for Excel 2010 to provide a flexible and powerful decision support system that enables practitioners to choose between different feasible solutions to a problem in realistic environments.

1. Introduction

Project management is the process of the coordination and integration of activities in an efficient and effective manner using limited resources. It consists of linking resources to their respective deliverables and assembling them into the whole project [1]. Resource management is an intrinsic element of project management [2–4]; resource management ensures that the project is completed on time and at cost and that the quality is as previously defined [5–7]. This is even more necessary for project-based companies such as contractors [3,8,9]. In fact, project scheduling problems are one of the most important problems that practitioners deal with in scheduling, especially when they need to achieve the most efficient resource consumption without increasing the prescribed makespan of the project.

However, because resources are scarce, the use of resources in the activities of the project leads to conflicts in the schedule [10]. Project scheduling problems comprise not only resource-constrained problems but also Resource Leveling Problems, among others [11]. These two kinds of problem consider resource consumption in two different ways: in the former it is seen as a constraint, and in the latter the problem is to make it as efficient as possible. Even though these two approaches may seem similar, they are conceptually different. Both have been widely studied by researchers and applied by practitioners, although these two groups are unaware of the differences between the approaches and the serious limitations imposed by the heuristics used in the commercial software.

These two problems are defined as non-deterministic polynomial-time hard (NP-hard) problems [12]. The first approach is a regular problem known as the Resource Constrained Project Scheduling Problem; its objective is to reduce the makespan without exceeding the constraints of resource availability [13,12]. The second, known as the Resource Leveling Problem (from now on, RLP) is a non-regular problem; its objective is to achieve the most efficient resource consumption without increasing the prescribed makespan of the project [14,12]. The two problems can be combined together as a multi-objective optimization problem, but there is always one main objective (usually the makespan); the other objective (usually the efficient resource consumption) is secondary.

Nevertheless, conventional analytical and heuristic methods are neither flexible nor productive when solving the RLP [15]. Some reasons for this inefficiency are, on the one hand, that exact procedures simplify the real problems so are not useful at offering optimal solutions with acceptable computational effort [16] and, on the other hand, that heuristics offer solutions which are far from optimal, so that it is necessary to apply metaheuristic algorithms to complex and realistic projects [17]. Recently, important approaches have been made by researchers to improve the efficiency of resource consumption, proposing different heuristics which are applicable to small projects; simple examples try to show the merits of a particular algorithm, without establishing

* Corresponding author.
E-mail addresses: jopontie@csa.upv.es (J.L. Ponz-Tienda), vyepesp@cst.upv.es (V. Yepes), pellicer@upv.es (E. Pellicer), jmflores@mat.upv.es (J. Moreno-Flores).

0926-5805/$ – see front matter
http://dx.doi.org/10.1016/j.autcon.2012.10.003

clear criteria for a performance comparison between the different algorithms [18].

Following this line of work, Liao et al. [11] proposed some ideas to advance the RLP in realistic environments; these authors made several proposals for the development or the improvement of the RLP. Regarding resource allocation, these authors proposed the use of a decision support system to assist project managers, as well as the development of benchmarking tests for performance assessment and comparison [11]. Concerning resource leveling, they suggested the use of multiple resources allowing the extension of the project deadline with a penalty [11]. We take these proposals as challenges to be overcome in this paper, contributing a little to the corpus of knowledge in this field.

Therefore, in this paper we present an Adaptive Genetic Algorithm (AGA) for the RLP with multiple resources allowing the extension of the project deadline with a penalty; for this purpose, we use the Weibull distribution as a termination condition, establishing an estimation of the global optimum. The proposed algorithm is tested with the standard "project scheduling problem library" (PSPLIB) [18], presenting a complete set of benchmarking tests. A decision support system is also used in order to implement this algorithm. Without loss of generality, we consider the classical resource leveling objective function: the total squared utilization cost for a given schedule.

The remainder of this paper is organized as follows. Section 2 provides the classification and formulation of the RLP. Section 3 details the different solving procedures: exact, heuristic, and metaheuristic algorithms with the new use of the Weibull distribution as a termination condition. Section 4 describes the algorithm proposed for the RLP with multiple resources. Computational results and the benchmarking test are explained in Section 5. Finally, conclusions are drawn.

2. Classification and formulation of the Resource Leveling Problem

The general formulation of the RLP requires us to consider the following elements:

1. The set of activities, N:

$$N = \{j_1, j_2, \cdots, j_n\} \tag{1}$$

n being the total number of activities.

2. The set of durations, D:

$$D = \{d_1, d_2, \cdots, d_n\} \tag{2}$$

where d_i, $1 \le i \le n$ is the assigned duration for each activity.

3. The set of periods of time in which these activities have to be distributed:

$$T = \left\{t_1, t_2, \cdots, t_p\right\} \tag{3}$$

t_p being the deadline of the project, from now on denoted $\overline{T}$.

4. The set of resources, R:

$$R = \{r_1, r_2, \cdots, r_k\} \tag{4}$$

k being the total number of resources.

5. The set of availabilities of the resources, A:

$$A = \{a_{it}, 1 \le i \le k, 1 \le t \le p\} \tag{5}$$

where a_{it} is the availability of the resource r_i in the period t.

6. The set of costs, C:

$$C = \{c_1, c_2, \cdots, c_k\}. \tag{6}$$

7. The set SS: to distribute the performance of the activities along the elements of the set T one needs to allocate a starting time for each activity, given by the ordered set, SS:

$$SS = \{SS_1, SS_2, \cdots, SS_n\}. \tag{7}$$

SS_i, $1 \le i \le n$, is the starting time of the activity j_i. $\overline{T}$ can be considered as the starting time of a finish dummy activity SS_{finish}, and then SS becomes:

$$SS = \left\{SS_1, SS_2, \cdots, SS_n, SS_{finish}\right\}. \tag{8}$$

Obviously, the schedule SS is not unique; on the contrary, there are a large number of different possibilities, according to the logic and restrictions of the project to be performed. Each of these schedules has significant differences in the efficiency of resource consumption, and this is the reason for finding the values of SS which optimize this efficiency.

8. The functions $r_i(S,t)$, $1 \le i \le k$: given a schedule SS, the function $r_i(S,t)$ is defined as the consumption of the resource r_i in the period of time t, belonging to the set T, in such a way that the consumption of the resource r_i throughout the project is given by:

$$u_{i1} = r_i(S, t_1), u_{i2} = r_i(S, t_2), \cdots, u_{ip} = r_i\left(S, t_p\right). \tag{9}$$

9. The function f: Given a schedule SS, the efficiency of resource consumption depends on the layout of its use. Therefore, it becomes fundamental to establish an optimal criterion for the distribution of the resources. This is the role we want f to play in the development of the problem. Hence, the function f will be different for each optimization criterion to be considered.

Once we have the elements that compose the problem, a general formulation could be:

$$\text{Minimize} \sum_{i=1}^{k} c_i f[r_i(S,t)] \tag{10}$$

subject to:

$$SS_{finish} \le \overline{T} \tag{11}$$

$$SS_i + d_i + \gamma_{ij} \le SS_j \quad \text{, for all } i \text{ which are successors to } j \tag{12}$$

$$u_{ij} \le a_{ij} \tag{13}$$

where γ_{ij} is the lead/lag between i and j.

Having done this, the choice of the function f, which defines the criterion for the optimization of the resource consumption, provides different ways of solving the problem. In the case of the RLP, the optimization criterion focuses on getting the resource consumption as level as possible. Consequently, a suitable choice of f could be:

$$f[r_i(S,t)] = \sum_{t=1}^{\overline{T}} \frac{(u_{it} - a_{it})^2}{\overline{T}}. \tag{14}$$

And Eq. (10) turns into:

$$\text{Minimize} \sum_{i=1}^{k} c_i \cdot f[r_i(S,t)] = \sum_{i=1}^{k} \sum_{t=1}^{\overline{T}} c_i \frac{(u_{it} - a_{it})^2}{\overline{T}}. \tag{15}$$