The effectiveness of resource levelling tools for Resource Constraint Project Scheduling Problem

A. Kastor a, K. Sirakoulis b,*

a Maliakos-Kleidi Construction JV, 2nd km. Itea – Gonnoi, 40004 Larissa, Greece

b Department of Project Management, Technological Education Institute of Larissa, TEI New Buildings, 41110 Larissa, Greece

Received 31 October 2007; received in revised form 23 July 2008; accepted 19 August 2008

Abstract

The PERT/CPM network techniques are based on the assumption that all needed resources will be available. The scarcity of resources is a usual reason for project delays. For the solution of the Resource Constrained Project Scheduling Problem (RCPSP) other methods have been applied. The objective of these methods consists in minimizing the project’s duration by considering both the precedence and resource constraints. Project Management software packages solve the resource conflicts using resource levelling. The paper evaluates the effectiveness of resource levelling tools of three popular packages by comparing the results when levelling two real construction projects as case studies.

© 2008 Elsevier Ltd and IPMA. All rights reserved.

Keywords: RCPSPs; Levelling; Software evaluation

1. Introduction

The CPM has been widely used for project scheduling, helping managers to guarantee the in time and on budget completion of the project. CPM provides useful information for the project, such as the critical path(s) and free and total float, which are essential for the efficient planning of a project. An advantage of CPM is the ability of managing by exceptions (critical and near critical activities) especially in large-scale projects [1]. However, CPM is based on the assumption that there are unlimited resources for the execution of the activities. Though in real projects, resources are not unlimited. Thus, scheduling without considering resource constraints gives unreliable schedules.

A survey conducted by Liberatore et al. [2] shows that 83% of professional project managers use project management software for planning and control, and that in construction industry resource levelling is used by 58% for planning and by 44% for project control. From the same survey it was derived that Primavera Project Planner and MS Project are the most popular software packages used for construction projects. By adding an open source package for comparing purposes, this paper examines the resource levelling capabilities of three PM software packages in two construction projects.

The paper is organized in five paragraphs. In the first, the theoretical background of the problem is formulated. Next paragraph presents the recent techniques, as they appear in the international literature, for finding more reliable schedules by considering resource constraints. After that, the objectives and the methodology of this paper are presented, and the paper closes with results and short conclusions.

2. Problem formulation

A project is represented by an Activity-On-Node network (AoN). It consists of \( n \) activities, where 1 and \( n \) are dummies and denote the start and the completion of the
When considering resource constraints in scheduling, some critical activities may be delayed because of resource unavailability. In that case, the sequence of the critical path is lost [6]. Critical sequence was proposed by Weist [6] to find critical activities by considering both precedence relations and resource constraints. Activities in the critical sequence affect the projects completion, like the critical path for the resource unconstrained case. Weist [7] proposed a heuristic approach for the solution of resource constrained problem with constant resource limits, and Woodworth and Shanahan [8] developed a method for the calculation of the resource constrained float. Li and Willis [9] used the critical sequence in their time/cost trade-off algorithm for the resource constrained problem, but they did not calculate the float. Bowers [1] proposed a set of heuristics based on the critical sequence for the calculation of the resource constrained float. These algorithms assumed a fixed critical sequence with an invariant resource allocation.

3. Literature review

However, a resource constrained project network may have a lot of equivalent schedules [1]. An example of equivalent schedules is given in Fig 2, where two equally good solutions of network A (Fig. 1) are depicted. Activities 3 and 4 have the same duration, resource requirement, predecessors and successors, but in schedule (a), Activity 3 is critical and Activity 4 has a significant float, while in schedule (b), Activity 4 is critical and Activity 3 has a significant float. Raz and Marshall [10] introduced the early and late scheduled dates, which are the early and late dates of the activities, by considering both precedence relations and resource constraints. A heuristic algorithm was applied for the calculation of the early scheduled dates. Inversing this algorithm and starting from the early scheduled finish of the project, they calculated the late scheduled dates. Then, they calculated the scheduled total and free float using the classic way. Goldratt [11] proposed a methodology for the identification of the Critical Chain, which is defined as “the longest chain of dependant steps”, where “dependencies between steps can be a result of a path or a result of a common resource” [11]. Critical Sequence and Critical Chain are based on the same concept. The difference is that critical chain uses reduced activity durations. Goldratt reduced the activity duration by assuming that it contains a protection against uncertainty and used this protection in some strategic positions in the schedule by adding time buffers.

Bowers [12] used some perturbations of the network inorder to generate the alternative schedules. Tormos and Lova [13] complement the concept of Backward and Forward Free Slack from Lova et al. [14] with the Backward and Forward Total Slack for the calculation of Resource Constrained Activity Criticality Index (RC ACI), and they integrated their method into Microsoft Project ‘98. Forward Free Slack of an activity is the amount of time in
which the activity can be right-shifted allowing the successors to start on their scheduled dates. Backward Free Slack of an activity is the amount of time in which the activity can be left-shifted allowing the predecessors to start on their scheduled dates. Forward and Backward Free Slacks do not take into account resource constraints. Respectively, Forward and Backward Total Slack of an activity is amount of time in which the activity can be executed earlier or later, maintaining both the feasibility and the project completion time.

Fondahl [15] remarks that, as soon as resources are considered in a project, the original calculated network data, which are used as priority rules, are activity attributes and may have no meaning in resource constrained scheduling. For that reason Lu and Li [16] proposed the Resource-Activity Critical Path Method (RACPM), in which “the dimension of resources is considered in addition to activity and time” [16]. The RACPM is a serial path heuristic method based on the knowledge-based system of Waugh and Froese [17], which can handle both renewable and non-renewable resources and uses work content as a priority rule.

Kim et al. [18] developed the Resource Constrained Critical Path Method (RCPM) for the calculation of the resource constrained float without the phantom float. Phantom Float is called the difference between the theoretical remaining total float and the actual remaining total float [18]. This difference is the result of using the CPM for the backward scheduling after the resource constrained forward scheduling. This Phantom Float exists in project management software packages when executing resource levelling. An example of phantom float is shown in Fig. 3. Activities A–D share the same resource with constant availability of 1. Each activity requires 1 per day. When levelling the CPM schedule activities A and B seem to have a total float of 2 periods, but if activity B is delayed, the entire project will be delayed due to the use of a common resource. Hence, the float of the 2 periods does not exist. The advantages of RCPM are:

- Gives more realistic schedules since it takes into account resource availability.
- Identifies the critical path and calculates floats correctly.
- Provides a stable schedule in a certain required level throughout the project duration.

At a specific time period of a project, a pair of activities may be critical (limit the advancement of the project). This
means that if either of them is delayed, the project completion would not be affected. But, any delay in both of them would delay the project. This pair of activities consists a critical set \([19]\). If there is more than one critical set within the same time period, then the project’s completion is limited by a critical cloud \([19]\). Rivera and Duran \([19]\) developed an algorithm for identifying critical sets and clouds.

Evaluations of resource levelling capabilities of project management software were performed by Johnson \([20]\), Maroto and Tormos \([21]\), Burley \([22]\), Farid and Manoharan \([23]\), Maroto et al. \([24]\) and Kolisch \([25]\). Johnson \([20]\) used the 110 problems of Patterson \([26]\) to compare 13 versions of 7 project management software packages. Johnson compared the results given from the software packages with the optimal solutions of the problems as given by Demeulemeester and Herroelen \([27]\). Maroto and Tormos \([21]\) used one problem with 51 activities and three resources to compare 7 versions of 6 software packages. Burley \([22]\) compared Microsoft Project 3.0, Project Manager Workbench/w and Timeline 6.0, and Farid and Manoharan \([23]\) compared Microsoft Project 3.0, Primavera Project Planner, Project Scheduler 5.0 and Time Line. Maroto et al. \([24]\) evaluated six software packages: CA Superproject v.4.0, Microsoft Project v.4.0, Project Scheduler 6 v1.5, Time Line v.6, Primavera Project Planner v.2 and Artemis Schedule Publisher v.4.2. Ninety-six projects were solved from the six software packages, and the results were compared against the optimal solutions obtained by the Branch and Bound procedure of Demeulemeester and Herroelen \([27]\), 2 heuristics (MINLFT, FCFS) and the TRC Method. Kolisch \([25]\) compared the solutions given by seven software packages: Artemis Schedule Publisher v.4.1, CA Superproject v.3.0 C, Microsoft Project v.4.0, Primavera Project Planner 1.0, Project Manager Workbench v.1.1.02w, Project Scheduler 6.0 v.1.02 and Time Line v.6.0 with the solution given by the Branch and Bound procedure of Demeulemeester and Herroelen \([27]\) on 160 instances.

4. Objectives and methodology

The objective of this paper is to analyze the quality of resource constrained scheduling of project management software, by evaluating two widely used software packages \([2]\), Primavera P6.0 and Microsoft Project 2007, and one open source, Open Workbench 1.1.6. Furthermore, the paper compares the results with the previous versions of these software packages, e.g. Primavera Project Planner v.3.1, Microsoft Project 2003 and Open Workbench 1.1.3.

Project management software packages usually use priority rule-based heuristic algorithms for the resource leveling, but they do not offer other information about the details of the algorithm, e.g. which scheduling scheme or which priority rule they use, if it is static or dynamic. Primavera Project Planner gives the user the opportunity to choose the priority rule but it does not illustrate to the user the possible usefulness of changing the priority rule or using more than one. These criteria can cause many possible methods which can end up to a different schedule. Therefore, the paper examines the effectiveness...
of resource levelling of the software packages on two real construction projects to show that the duration of the project, when considering resource constraints, depends on the software and/or the method used.

The first benchmark instance is a real housing project which consisted of 96 houses. The focus was on the concreting of these 96 houses because there was problem. The project has 98 activities, and 1 renewable resource (concrete) was considered. According to the projects initial plan, the average daily concrete requirement, during the projects horizon, was 112.28 m³ with a maximum of 126.4 m³ but the constructor could procure only 80 m³. The following parameters were used inorder to determine the complexity of the problem [28]: The Network Complexity (NC), which measures the average number of precedence relations per activity. The Resource Factor (RF), which defines the average number of constrained resources requested by an activity. RF is normalized to the interval [0,1]. For RF=1, each activity requires every one of the R resources, and for RF= 0, none of the activities require any of the resources. The resource strength (RS), which measures the proportion of resource demand and availability RS, is also normalized to the interval [0,1], and for RS=0 the problem is highly resource constrained, while for RS=1, the project is resource unconstrained. For this instance, the values are NC=2.6, RF=1 and RS=0.62.

The second instance is the construction of a shopping mall consisting of 19 buildings. When the resource levelling was applied, the project was in progress and the contractor wanted to program the remaining work, especially electromechanical and activities requiring specialised work, by considering the availability of workgroups. The timeschedule consists of 668 activities and 7 renewable resources. The complexity factors for this instance are NC =0.86, RF =0.05, and RS equals 0 for 5 of the seven resources and 0.4 and 0.5 for the other two.

Microsoft Project and Open workbench use a standard built-in procedure so there is one result for each. For P6, besides the default resource levelling, six priority rules and all possible combinations between them were used, so P6 obtained 31 results. The priority rules used are Late Start Time (LST), Late Finish Time (LFT), Minimum Slack (MSLK), Ranked Positional Weighted Method (RPWM) and Enhanced Positional Weighted Method (EPWM). The last two rules do not exist in Primavera, but were computed and imported to Primavera as Custom Data Items. The comparison of these results made using the percentage deviation from the project duration without resource constraints, which is 464 days for instance 1 and 238 for instance 2.

5. Results

On comparing the default options given from the software (Table 1), Primavera P6 gives the best results and is followed by MS Project in both the instances. Open Workbench has a great deviation especially in the second instance.

On comparing all the results (Table 2), Primavera P6 default and by LST rule give the best solution for both instances with an average deviation of 41.11%. MS Project is ranked 3 (46.14%) and is followed by P6 by RPWM and by LFT (47.19%). The worst result was given by Open Workbench with an average deviation of 167.79%. Especially in the second instance, Open Workbench gives a duration that is over the double from CPM duration. A point to be noted is that P6 by EPWM rule gives the best solution for the second instance, which is not the case for the first instance. Also MS Project in the first instance gives the same result with P6 by PWM and LFT, but in the second instance gives slightly better solution.

The evaluated software can be divided into three groups according to the quality of the schedules they produce. The first group consisted of those who have a better performance (average deviation less than 50%), which are P6 default, P6 by LST, MS Project, P6 by LFT and P6 by PWM. The second group contains the medium quality software (average deviation 50% to 60%), which are P6 by EPWM and P6 by MSLK, and the third group is formed by the worst quality software (average deviation over 60%), which is P6 by SPT, and the result of Open Workbench is not included as it is an outlier (Fig. 4).

The two instances that are also solved by Primavera, combining 2 priority rules and the average results are presented below (Table 3). The first rules are given in the rows and the second rules are given in the columns. Better results are given from LST, LFT and PWM when used as first rules. For tie breaking rule, better results are given by EPWM, PWM and LST. The combination of PWM and LST and the combination of LST and MSLK give the best solution for both the instances.

### Table 1
Duration and percentage deviation obtained by the project management software for default options

<table>
<thead>
<tr>
<th>Rule</th>
<th>1st Instance Duration</th>
<th>Percentage deviation from CPM (%)</th>
<th>2nd Instance Duration</th>
<th>Percentage deviation from CPM (%)</th>
<th>Average Percentage deviation from CPM (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P6 Default</td>
<td>709</td>
<td>52.80</td>
<td>308</td>
<td>29.41</td>
<td>41.11</td>
</tr>
<tr>
<td>MS Project standard</td>
<td>744</td>
<td>60.34</td>
<td>314</td>
<td>31.93</td>
<td>46.14</td>
</tr>
<tr>
<td>Open workbench</td>
<td>863</td>
<td>85.99</td>
<td>832</td>
<td>249.58</td>
<td>167.79</td>
</tr>
</tbody>
</table>
6. Conclusions

This paper evaluated the resource levelling capabilities of three project management software packages on two real construction problems. The results show that the project duration depends on the software or the method used. It could be supported that, for problems of that size and complexity and objective function the makespan minimization, there is an increase in makespan from 41.11% to 167.79% of the resource unconstrained schedule and that Primavera P6 outperforms MS Project and Open Workbench. Because of the big range of the results, project managers should not rely on the first result they get but try other rules or even software if possible.

Primavera P6 is the most effective as it allows the user to define theoretically unlimited number of criteria as priority rules. For the other software, the main problems are that in MS Project the user cannot have constraint on a material (user has to manage it as labor) and that Open Workbench was too slow when scheduling. Previous versions of the
same software programs, e.g. Primavera Project Planner v.3.1, MS Project 2003 and Open Workbench 1.1.3, gave exactly the same results.

Regarding the different methodology used in the previous evaluations, it is not easy to make an exact comparison of the results. Johnson [20] compared seven software packages, with MS Project (1.0 and 3.0) and Primavera (4.0, 4.1 and 5.0) among them, against Talbot’s Optimizer [29] and Patterson Heuristic [30] on the 110 Patterson’s problems. The results showed that Timeline was the most successful with an average of 5.03% over the optimal solution. MS Project was the least successful. Found the optimal solution for 10 projects, and had an average of 25.6% over the optimal solution. Primavera had an average of 7.45% over the optimal duration. Maroto and Tormos [21] compared the results from seven software packages against CPM on a project with 51 activities and three resources. They did not use the product of Primavera. MS Project 3.0 was within the first three that outperformed. Farid and Manoharan [23] used 10 of the 110 Patterson’s problems to compare five software packages against Talbot’s Optimizer [29] and Patterson Heuristic [30]. MS Project v.3.0 and Primavera Project Planner v.5.0 are ranked last (fourth and fifth, respectively). Maroto, Tormos and Lova [24] compared six software packages on 96 auto-generated problems against Branch and Bound procedure from Demeulemeester [31]. Primavera standard, Primavera by LFT and MS Project are placed in the group of medium quality. Primavera by MSLK was placed in the third group of worst performance. Kolisch [25] used seven software packages on 160 instances against Branch and Bound procedure from Demeulemeester and Kolisch [31]. Primavera Project Planner v.1.0 was placed in the first group of best performance, Microsoft project in the second and Project Manager Workbench v.1.1.02w in the third.

Our results seem to be consistent with the evaluation of Maroto et al. [24]. Primavera standard, Primavera by LFT and MS Project are placed in the same group (first for us second for Maroto et al.), while Primavera by MSLK is placed in the next one.

CPM is inappropriate for scheduling resource constrained projects, and project managers should not rely only on this method when dealing with limited resources [32]. Project management software packages should incorporate other more efficient methods for resource constrained scheduling and should try to illuminate some drawbacks such as Phantom Float. It would be also interesting if project management software could have other objectives, besides the makespan minimization, such as maximizing the net present value.

References


