

RESEARCH ARTICLE

Resource allocation capabilities of commercial project management software packages for resource leveling and resource constrained project scheduling problems: A comparative study

Noor Hussein Farooq Albayati ¹, Saman Aminbakhsh ²

¹ Atilim University, Graduate School of Natural and Applied Sciences, Ankara, Türkiye

² Atilim University, Department of Civil Engineering, Ankara, Türkiye

Article History

Received 16 March 2023

Accepted 31 May 2023

Keywords

Resource allocation

Resource leveling problem

Resource constrained project
scheduling problem

Project management
software

Serial scheduling scheme

Abstract

In construction project management the critical path method (CPM) is the most used technique for project scheduling. Although this technique provides many advantages for project managers, it cannot efficiently deal with the allocation of the resources. Therefore, alternative techniques have been introduced to address resource allocation requirements of the projects. Of these techniques, Resource Leveling (RLP) aims to minimize the fluctuation in resource usage histograms while maintaining the duration obtained by CPM. Resource Constrained Project Scheduling Problem (RCPSP), on the other hand, aims to secure the shortest CPM duration without violating the resource constraints. RLP and RCPSP are vital for effective utilization of project resources (e.g., manpower, machinery, and equipment) as they help precluding intermittent usage or over-allocation of the resources. Keeping the resource usage at a relatively constant level through RLP would result in a decrease in the overall project cost as the additional costs required to demobilize and remobilize the resources will be minimized. Shortening the makespan while meeting the resource constraints through RCPSP would lead to improved resource utilization and cost savings as well. The main objective of this study is, therefore, to analyze effectiveness and efficiency of the most widely used commercial project management software packages in solving resource allocation problems. To this end, the most recent versions – as per the date of this study – of three software packages, namely, Microsoft Project Professional 2019, Primavera P6 Professional 2019, and Asta Powerproject version 15.0.01.489 are examined. The performance of the practiced software is evaluated based on thirteen different priority rules over a set of problem instances available in the literature. The practiced problems include 640 instances providing a diverse combination of network complexity, activity number, and resource type number. Results obtained by the software for RCPSP are also compared with the solutions provided by the Serial Scheduling Scheme – a heuristic method. The findings of this study reveal that whilst all the three software packages manage to provide comparable results, Asta PowerProject transpire to be the all-round best performing method while Primavera sports the fastest leveling module. This study also sheds light on the challenges and practical hurdles to utilization of the aforementioned software for resource allocation purposes.

1. Introduction

Resources are commonly classified into two classes: renewable resources and non-renewable resources. The difference between the two main categories of resources is shown in Fig. 1. Non-renewable resources refer to consumable resources, such as money, fuel, energy, and raw materials. The objective is minimizing the total utilization value of non-renewable resources in the available range of project durations. In contrast with the non-renewable resources, renewable resources like manpower, machines, and various other capital equipment are necessary for the execution of the project [1].

In construction projects, the Critical Path Method (CPM) suggested by Kelley and Walker [2] and the Program Evaluation and Review Technique (PERT) introduced by Malcolm et al. [3] have widely been used for planning and controlling of projects. Such techniques mainly focus on timely completion of projects without exceeding a given budget and assume that the duration of activities is known or have a predetermined probability distribution [4]. In these classical techniques, activity durations are the only variables and the availability of resources which can potentially

affect the resource allocation are not considered [5]. In practice, though, there are many cases where these conditions are not met. Generally, these network techniques assume that each activity starts as early as possible and that all the required resources are available in unlimited quantities. However, in real projects, resource availability need to be considered due to the limitations on the number of required resources. In fact, disregarding the limitations on resource quantities can lead to unrealistic schedules [6]. Accordingly, there have been several studies focusing on various methods for more efficient handling of resources and scheduling of projects that can be classified as resource leveling (RLP) and resource constrained scheduling problem (RCPSP) [7]. Resource leveling is a technique used to balance resource usage over project span which aims to achieve a more even distribution of resources while maintaining the duration determined by the CPM [8]. The resource-constrained scheduling problem refers to the computational problem of minimizing the project makespan by taking into account the precedence relationships between the activities and the limitations on the availability of the project resources [9–11].

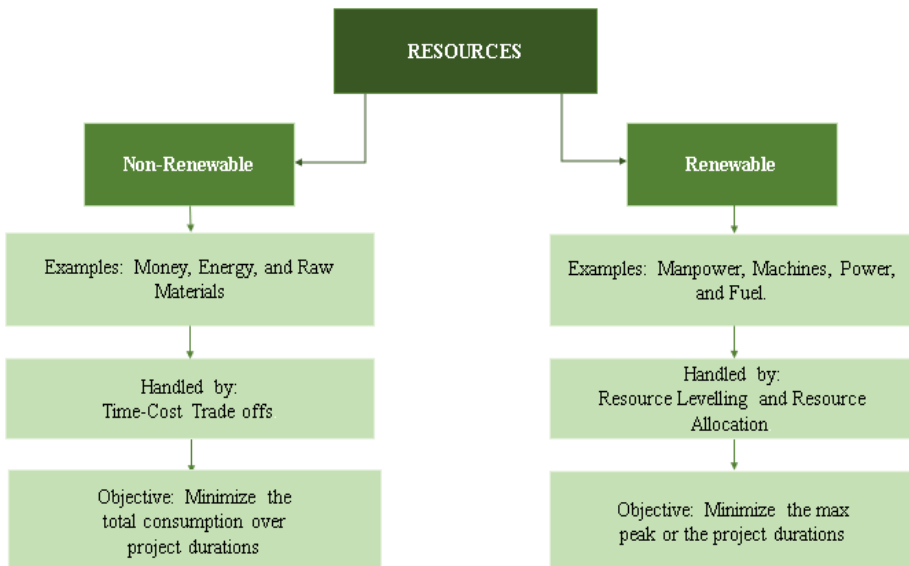


Fig. 1. Classification of resources

RLP and RCPSP are well-known and extensively studied problems in the field of project management. RCPSP involves allocating limited resources, such as labor, materials, and equipment, to a set of tasks that must be completed within a given timeframe. The RCPSP is NP-hard, meaning that it is computationally intractable to solve exactly for large instances. Therefore, numerous heuristic and metaheuristic approaches have been proposed to tackle this problem. These methods include genetic algorithms, simulated annealing, tabu search, ant colony optimization, and particle swarm optimization, among others.

A wide range of research has been conducted in the area of resource allocation, which has led to the development of numerous algorithms, models, and solution techniques. Several studies have focused on finding effective approaches to solve the resource allocation problems including various extensions and variants such as the multi-mode RCPSP, the resource-constrained multi-project scheduling problem, and the stochastic RCPSP. Likewise, RLP is somewhat dual to RCPSP [12] and comes under the category of RCPSP [13]. Brucker et al. [14] provided a comprehensive review of the notation, classification, models, and methods for solving the RCPSP. Kolisch and Hartmann [10] conducted an experimental investigation of heuristics for resource-constrained project scheduling and provided an update on the state-of-the-art. Blazewicz et al. [15] discussed the classification and complexity of scheduling problems subject to resource constraints. Garey and Johnson [16] showed that the RCPSP is NP-hard and provided a guide to the theory of NP-completeness. Demeulemeester et al. [17] reviewed recent developments in the RCPSP, including models, algorithms, and solution techniques. Valls et al. [18] proposed a hybrid genetic algorithm with a local search for solving the RCPSP. Beşikci et al. [19] reviewed the multi-project scheduling problem, which is an extension of the RCPSP. Bruni et al. [20] and Zhou et al. [21] reviewed the stochastic RCPSP, which considers uncertainty in resource availability and task duration. In recent years, there has been a growing interest in applying

artificial intelligence (AI) techniques, such as machine learning and deep learning, to solve the RCPSP. Sung et al. [22] and Sallam et al. [23] reviewed the application of machine learning technique to solve the RCPSP, including reinforcement learning method. They showed that machine learning approaches have the potential to significantly improve the efficiency and effectiveness of project scheduling. Overall, the RCPSP remains an active and important area of research in project management, and its solutions are crucial for effective project planning and scheduling in various domains, including construction, manufacturing, and software development.

In addition to the algorithms proposed in the literature, commercial project management software packages (PMSP) such as Microsoft Project, Primavera P6, and Asta PowerProject also incorporate features for addressing RLP and RCPSP. The construction industry is flexible and absorbent to employ such planning software packages [24]. The widespread use of such programs by the planners also the ease of access to the resource allocation modules of these software, motivated the authors to carry out a comparative study on the RLP and RCPSP performance of the aforesaid software. Accordingly, this study aims to analyze effectiveness and efficiency of the most widely used commercial PMSPs in solving RLP and RCPSP. To this end, the most recent versions – as per the date of this study – of three software packages, namely, Microsoft Project Professional 2019, Primavera P6 Professional 2019, and Asta Powerproject version 15.0.01.489 are examined. The performance of the practiced software are evaluated and guidelines are provided for the project managers for selection of the right software package(s) for their real-life applications. Selecting the suitable tool is crucial because real-life projects can include significant number of activities and resources, rendering the resource optimization process significantly more complex and computationally costly.

The remainder of this paper is organized as follows. Section 2 describes the relevant literature.

In Section 3 the methodology is outlined. Section 4 performance evaluations and the associated results. Findings of the study is discussed in Section 5. Finally, concluding remarks on the present work are given in Section 6.

2. Literature review

Commercial project management software packages are widely used by organizations to manage project scheduling, resource allocation, and budgeting. These software packages are designed to automate project management tasks, improve collaboration among team members, and increase productivity. One critical aspect of project management is resource allocation, which involves assigning resources to tasks and ensuring that resources are used efficiently. Previous studies have investigated the resource allocation capabilities of commercial software and compared them based on various criteria as follows.

Johnson [25], studied the performance of commercial software for solving RCPSP. 110 instance examples with number of activities ranging from 7 to 51 and resources types from 1 to 3 were used. The capabilities of seven different software packages were tested, including Super Project 1.0 and 2.0, Timeline 2.0 and 4.0, Primavera 4.00, 4.1 and 5.0, Harvard Total Project Manager II, Harvard Project Manager 3, Hornet, Pertmaster, Microsoft Project 1.0 and 3.0. The best performance was found to be for Timeline 2.0 and the worst performance was noted for Microsoft Project 1.0. Maroto and Tormos [26], studied the performance of different software packages in solving RCPSP. A single instance problem consisting of 51 activities and three resource types was used for evaluation purposes. The researchers used seven different software packages of CA-Super Project 2.00A, Insta Plan 3.00B, Micro Planner for Windows 6.24A, Micro Planner Professional 7.3B, Microsoft Project for Windows 1.0, Microsoft Project for Windows 3.0, and Project Scheduler 1.0. The best solution was reported for CA-Super Project and Microsoft Project 3.0 and the worst solution was recorded for the Microsoft Project 1.0. Kolisch et al. [27], used seven project

management software packages for RCPSP too. The authors used a set of 160 instance problems generated by ProGen and ProGen/max. The number of activities was listed as 10, 20, and 30 with 1 to 3 resource types. A comparison was made among seven software packages including Artemis Schedule Publisher 4.1, CA-Super Project 3.0C, Microsoft Project 4.0, Primavera Project Planner 1.0, Project Manager Workbench 1.1.02w, Project Scheduler 6.0 1.02, and TimeLine 6.0.0. The best solution was found to be of Timeline 6.0.0 and the worst performing software was discovered to be Artemis Schedule Publisher 4.1.

Mellentien and Trautmann [28], evaluated the performance of five commercial software packages in solving the RCPSP. Acos Plus.1 8.2, CA-SuperProject 5.0A, CS Project Professional 3.0, Microsoft Project 2000, Scitor Project Scheduler 8.0.1 were tested. A set of 1,560 instance problems were used with 30, 60 and 120 number of activities and 4 resource types. The best solutions were found by Scitor Project Scheduler 8.0.1 and Acos Plus.1 8.2. Hekimoglu [29], studied the performance of Primavera Enterprise Project Management 4.1 (using two priority rules of minimum total slack and late finish time) and Microsoft Project 2003 software packages for RCPSP. They used a set of 2,040 instance problems with 30, 60, 90, and 120 number of activities and 4 resource types. The results showed that for small problems Microsoft Project was performing better than Primavera, and for large problems Primavera using late finish time priority was recommended. Kastor and Sirakoulis [6], discussed the RCPSP capabilities of three software packages of Primavera P6.0, Microsoft Project 2007, and Open Workbench 1.1.6. They used two real construction project examples. The first one consisted of 98 activities with one resource type and the second included 668 activities with 7 resource types. The results revealed better performance by Primavera P6.0 [6]. Cekmece [30], used Primavera Enterprise Project Management P6.0 and Microsoft Project 2007 software packages for RCPSP. 45 instance problems used with 30, 60, and 120 number of activities that included 4 resource types. In this study P6.0 provided better

results than Microsoft Project 2007. Furthermore, it was argued that Hekimoglu [29] preferring Microsoft Project to Primavera over the small projects was not justifiable since they both provided relatively the same results.

Son and Mattila [31], proposed a linear programming method and compared the results with the capabilities of two commercial software packages of SureTrak Project Manager 3.0 and Primavera Project Planner P3.0 for solving the resource leveling problem. Binary decision variables were used to level the resources with allowing to split the activities (stop and restart some of the activities) for the first time in the literature. The formulation was tested and developed on the CPM and RLP was studied by incorporating three assumptions as: activities can split; only some of the activities can split; all the activities are allowed to split. Two examples were used consisting of 10 and 11 activities and one resource type and the best solution was found for the case in which all the activities were allowed to stop and restart. Iranagh and Sonmez [32], made a comparison between the performance of Microsoft Project 2010 and genetic algorithm solution of RLP. A set of 16 problems having up to 20 activities and a single resource was used. The results of resource leveling revealed superiority of the performance of the proposed genetic algorithm over Microsoft Project 2010. Rezvan Khan [33], made a comparison among Primavera P6.0 Professional R8.3, Microsoft Project Professional 2013, and Asta PowerProject V.12.5 software packages for RLP. A set of 640 instance problems were used for this purpose with 50, 100, 200, and 500 number of activities including 1, 5, 10, and 15 resource types. The results of the practiced software were compared with those obtained by Burgess and Killebrew [34] heuristic method. They discussed Burgess and Killebrew's algorithm required less computational time to solve the problems. They even indicated that the heuristic method was able to provide better solutions than the resource leveling module of the experimented software. Kuhlman [35] evaluated the portfolio management as well as the resource leveling

capabilities of four commercial PMSPs, including JDA, Planisware 5, Primavera P6, and HP. This study found that Planisware 5 had slightly better resource allocation capabilities than other software packages.

Gharaibeh [36] following a questionnaire survey concluded that Primavera could perform much better than MS Project, especially in relation to resource allocation and leveling. Farid and Manoharan [37] directed a comparative analysis of several project management software tools including Microsoft Project 3.0, Primavera Project Planner, Project Scheduler 5.0, and Time Line. Maroto et al. [38] generated 96 projects having 30 and 60 activities with resource requirements varying between 1 and 6. These instances were used to evaluate the performance of six different software of CA Superproject, Microsoft Project, Project Scheduler 6, Time Line 6, Primavera 2, and Artemis Schedule Publisher 4.2. The results obtained by these software packages were also compared with the solutions of Demeulemeester and Herroelen [39]. In a similar fashion, Kolisch [40] by using 160 instances, compared the results of seven software including Artemis Schedule Publisher 4.1, CA Superproject 3, Microsoft Project, Primavera Project Planner 1, Project Manager Workbench, Project Scheduler 6, and Time Line 6 with those of Demeulemeester and Herroelen [39]. Trautmann and Baumann [41] evaluated the RCPSPP capabilities of Acon Plus. 1, AdeptTracker Professional, CS Project Professional, Microsoft Office Project 2007, Primavera P6, Sciforma PS8, and Turbo Project Professional. They employed the benchmark test set of Mellentien and Trautmann [28] which included 1,560 instances with 30, 60, and 120 activities and 4 resource types for each. They concluded that while Sciforma PS8, AdeptTracker Professional, and Microsoft Project can provide shorter project makespans, none of them can compete with the state-of-the-art algorithms found in the literature. A summary of literature focusing on resource allocation capabilities of various commercial software packages is given in Table 1.

Table 1. Summary of existing studies on resource allocation capabilities of commercial PMSPs.

Study	Problem Type	Commercial Software Package	Test Problem(s)
Johnson [25]	RCPSP	Super project 1.0 & 2.0 Timeline 2.0 & 4.0 Primavera 4.00, 4.1 & 5.0 Harvard Total Project Manager II Harvard Project Manager 3.0 Hornet Pertmaster Microsoft Project 1.0 & 3.0	# of instances: 110 # of activities: 7 – 51 # of resources: 1-3
Maroto and Tormos [26]	RCPSP	CA-Super Project 2.00A Insta Plan 3.00B Micro Planner for Windows 6.24A Micro Planner Professional 7.3B Microsoft Project for Windows 1.0 Microsoft Project for Windows 3.0 Project Scheduler 1.0	# of instances: 1 # of activities: 51 # of resources: 3
Farid and Manoharan [37]	RCPSP	Microsoft Project 3.0 Primavera Project Planner Project Scheduler 5.0 Time Line	na
Kolisch [40]	RCPSP	Artemis Schedule Publisher V.4.1 CA Super Project V.3.0 C Microsoft Project V.4.0 Primavera Project Planner V.1.0 Project Manager Workbench V.1.1.02w Project Scheduler 6.0 V.1.02 Time Line V.6.0.0	# of instances: 160 # of activities: 10, 20 & 30 # of resources: 1 - 3
Maroto et al. [38]	RCPSP	CA Superproject Microsoft Project Project Scheduler 6 Time Line 6 Primavera 2 Artemis Schedule Publisher 4.2	# of instances: 96 # of activities: 30 & 60 # of resources: 1 - 6
Mallentien and Trautmann [28]	RCPSP	Acos Plus.1 8.2 CA SuperProject 5.0a CS Project Professional 3.0 MS Project 2000 Scitor Project Scheduler 8.0.1	# of instances: 1,560 # of activities: 30, 60 & 120 # of resources: 4
Son and Mattila [31]	RLP	SureTrak Project Manager V. 3.0 Primavera Project Planner (P3) V.3.0	# of instances: 2 # of activities: 10 & 11 # of resources: 1
Hekimoglu [29]	RCPSP	Primavera Enterprise V 4.1-Project Management Microsoft Project 2003	# of instances: 2,040 # of activities: 30, 60, 90 & 120 # of resources: 4
Kastor and Sirakoulis [6]	RCPSP	Primavera p6.0 Microsoft Project 2007 Open Workbench 1.1.6	# of instances: 2 # of activities: 98 & 668 # of resources: 1 & 7
Cekmece [30]	RCPSP	Primavera Enterprise V.6.0-Project Management (P6) Microsoft Project 2007	# of instances: 45 # of activities: 30, 60 & 120 # of resources: 4

Table 1. Continued

Trautmann and Baumann [41]	RCPSP	Acos Plus. 1 AdeptTracker Professional CS Project Professional Microsoft Office Project 2007 Primavera P6 Sciforma PS8 Turbo Project Professional	# of instances: 1,560 # of activities: 30, 60 & 120 # of resources: 4
Kuhlang [35]	RLP	JDA Planisware 5 Primavera P6 HP	na
Iranagh and Sonmez [32]	RLP	Microsoft Project 2010	# of instances: 1 # of activities: 5 – 20 # of resources: 1
Rezvan Khan [33]	RLP	Primavera P6.0 Professional R8.3 Microsoft Project Professional 2013 Asta PowerProject V.12.5	# of instances: 640 # of activities: 50 – 500 # of resources: 1 - 15
Gharaibeh [36]	RLP	Primavera Microsoft Project	na
Kolisch et al. [27]	RCPSP	Artemis Schedule Publisher V.4.1 CA Super Project V.3.0 C Microsoft Project V.4.0 Primavera Project Planner V.1.0 Project Manager Workbench V.1.1.02w Project Scheduler 6.0 V.1.02 Time Line V.6.0.0	# of instances: 160 # of activities: 10, 20 & 30 # of resources: 1 - 3

Overall, the literature on the resource allocation capabilities of commercial software packages suggests that Asta Powerproject, Microsoft Project, and Primavera P6 are among the most advanced software packages in terms of resource allocation. However, the performance of these software packages may vary depending on the specific context and requirements of a project. In addition, there exists no instance of a study focusing on both RLP and RCPSP capabilities of the aforesaid PMSPs in the literature. The main focus of this study is, therefore, to evaluate different software packages and to guide managers in choosing the one that best meets the resource allocation requirements and priorities of their particular projects.

3. Research methodology

In this section, the explanations about the instances adopted for this study and how they have originally been generated is covered. This section also

provides practical information as to how should the data be imported to the different software packages. Objective function will be discussed as well as the related formulation. The experimented software packages together with the Serial Scheduling Scheme (SSS) algorithm will be elucidated herein. Moreover, the leveling process and the assumption of the daily available resource value for RCPSP will be clarified.

3.1. Problem sets

In order to evaluate the RLP and RCPSP capabilities of project management software packages, 640 instances that were originally generated by Rezvan Khan [33] by using RanGen instance generator have also been used in this study. The researcher preferred RanGen to ProGen and ProGen/Max instance generators because of the ability to choose various parameters for generating a problem set [42]. In addition, RanGen is capable of generating samples with more complicated

networks that can resemble complexity of real-life projects. The parameters considered in the instance generation process included: number of activities, topology indicator or network complexity (Order Strength, i.e., OS), resource factor (RF), resource constrainedness (RC), and resource strength (RS). What follows is a brief explanation on how the original instances were generated by Rezvan Khah [33] and how they were slightly modified.

3.1.1. Activity number

This parameter defines the number of activities used in generating the problem set instances. Four levels of 50, 100, 200, and 500 activities are used in this study.

3.1.2. Topology indicator / Network complexity

Is an index that represents the network complexity named Order Strength (OS). Plainly, it is a parameter used to measure and set the number of precedence relationships in the network. Larger OS values indicate network activities are expected to include higher number of precedence relationships. Four levels of 0.1, 0.3, 0.5, and 0.7 are used for each problem set in this study.

3.1.3. Resource factor (RF)

The resource factor represents the average fraction of the resource type required per each activity. It can also be regarded as an index displaying the ratio of the resources used.

3.1.4. Resource constrainedness (RC)

The Resource Constrainedness parameter defines the demand for each of the resources. Resource demand per each resource increases from 0 to maximum available number as RC is increased from 0 to 1 for that specific resource. Resultantly, the value of RC is set as 0.9 for every resource type to generate more complex problems.

3.1.5. Resource strength (RS)

This parameter regulates resource availability. RS ratio ranges from 0 to 1 and larger values suggest greater resource availability. In the original study by Rezvan Khah [33], it is indicated that this parameter has not been considered while generation of the instances since in resource leveling the

demand for each resource is decided by the user. Though, in this study the values resulting from this parameter have been modified which is elucidated in section 3.1.6.

3.1.6. Problem set generation

In the study by Rezvan Khah [33], the RS parameter has not been considered while generation of the instances. As a result of this, the resource availabilities defined by RanGen for each resource in each instance have been set randomly without making any presumptions about their permissible ranges. For instance, in most of the sample problems the resource availabilities have been set randomly with values as high as 100, or sometimes even larger values are defined. Such large resource availabilities not only may lack practical relevance, but also can effectively remove the constrainedness of the resources. On the other hand, for each resource type, setting resource availability amount less than the largest daily utilization value of an activity would have increased the original duration of that specific activity during the leveling process. More specifically, the largest daily utilization value for any resource type is defined as 10 for any of the original instances generated; that is, setting any value smaller than 10 for the resource availability would have caused leveling to extend the original duration of the activities for the sake of satisfying the constraints on the number of the available resources. Since in this study the RCPSP results of the software packages are intended to be compared with those of Serial Scheduling Scheme – in which activity durations remain unchanged – any potential changes in the original activity durations are precluded in the leveling processes by equalizing the values of resource availability and daily utilization. For these very reasons, in order to adapt the original instances for resource leveling with resource constraints, some minor modifications are introduced herein as resource availability of 10 is assumed for each and every resource for all the instances. It should be pointed out that all the relationships among the activities are assumed to be Finish to Start (FS) meaning an activity may only start when all of its preceding activities finish. In addition, activities of each instance include two

dummies representing the start and the finish milestones.

3.2. Problem set conversion and resource allocation setup

All the generated instances are in text format which cannot be directly imported into any of the experimented software packages. Therefore, some adjustments must be made to re-arrange and convert the files in order to correctly input the data into the programs. In the first step, the data for each problem is imported to a Microsoft Excel file as it matches the mapping in Microsoft Project (MSP) program. After importing the Excel sheet into MSP, the maximum units for daily usage of each resource is defined as 10 by switching the active view to Resource Sheet because the maximum value of the daily resource utilization was assumed as 10 for all resource types for the whole instances. 'Level only within available slack' is once checked and once unchecked on Resource Leveling pane before running the Level All option located under Level group of the Resource tab. This is done to ensure leveling will ignore or respect the constraints on the availability of the resources for solving RLP or RCPSP, respectively. Resource leveling processes are then carried out and the results recorded. After each round of leveling/recording, all resources are returned back to their original state, i.e., they are unleveled by selecting the Clear Leveling option under the same menu on the ribbon. Resources are unleveled in order to import them to the other two software packages. It is observed and verified that the data can be imported easily from MSP to Asta PowerProject either as an '.mpp' or an '.xml' file; in contrast with Primavera P6 for which resource usage and maximum unit data were not imported properly. Therefore, benefitting from an intermediate file format converter, files with '.mpp' extension are converted to '.mpx' first. Similar to the procedure explained for MSP, necessary leveling options are configured before leveling the resources. More precisely, for P6 'Level resources only within activity Total Float' is unchecked on Level Resources window and 'Extend finish' is checked on Resource Leveller window for RCPSP.

As a result of these settings, leveling, would be able to satisfy the constrained number of resources by shifting the start times of the activities and freely extending the duration of the project.

As discussed in section 4, objective function calculations are carried out externally by using Microsoft Excel for the leveled schedules. For this, a major bottleneck is experimented with Asta PowerProject especially for more complicated instances with higher number of work items as it takes quite significant time to export the leveled daily resource consumptions. It takes so long to either copy or export resource usage data as '.csv' files that it removes away practicality of the obtained leveled values for post-processing purposes. Though, to walk around this issue, the authors have discovered an effective yet simple technique. The leveled schedules first need to be exported as '.mpp' files, then opened and exported as Excel files using Microsoft Project software.

3.3. Objective functions

Two groups of objective functions exist. The first group includes methods widely used for evaluation of the resource leveling capabilities of different approaches whereas the second group includes the metrics frequently used for analyzing the performance of methods for resource leveling capabilities of projects with constrained resources. Objective functions for resource leveling include Sum of Squares Metric (SSQR), Absolute Deviation Metric (ABSDEV), Overload Metric (OVERLOAD), and the Idle Days and Maximum Daily Resource Demand Metric (RID-MRD) all of which push the solution procedure to generate a flat resource usage histogram where variations are minimized. Objective functions for resource constrained project scheduling problem include but is not limited to Makespan Minimization and Net Present Value. They are chiefly used to analyze makespan minimization capabilities of different leveling approaches. One per each of the two distinct objective function groups are exercised in this study to not only assess the leveling performances but also makespan minimization capabilities of the experimented approaches. From

the first group, SSQR is used mainly because it demonstrates the strongest capability in peak minimization of resource utilizations due to squaring of the deviations, thereby penalizing the deviated utilizations even more so than the other methods. This metric captures daily resource usage both over and under the average resource demand. Formulation of SSQR is presented in Eq. (1).

$$f_{SSQR} = \sum_{k=1}^K w_k \sum_{t=1}^T r_{kt}^2 \quad (1)$$

where; f_{SSQR} is the objective function to be minimized; K is the total number of resource types; k denotes the resource type; w_k is the weight of k -th resource; T denotes the total project duration; t is a day in the project span; r_{kt} denotes the resource usage of k -th resource on t -th day.

Yet, RCPSP generally tend to increase the overall duration of the project due to shifting forward and decreasing the overlapping segments of the parallel activities requiring the same type of resources – more than the available units – at the same time. Obviously, securing the shortest makespan is a desired outcome of the RCPSP and to analyze this, Makespan Minimization metric from the second group of objective functions is employed in this study.

All the experimented commercial software packages report the resource usage data in man-hours per day. Thus, SSQR objective function calculations are carried out externally by using Microsoft Excel for the leveled schedules. As the daily working hours were set as 8 hours, the outcomes were divided by 8 for all the values of the daily resource in order to make them comparable with the results of Serial Scheduling Scheme.

3.4. Experimented commercial software packages

The suitable selection of software packages is vital since each program has their own merits and demerits. In this study though, resource allocation capabilities of some the more widely practiced software are addressed. This study evaluates effectiveness and efficiency of three software

packages of Microsoft Project Professional 2019, Primavera P6 Professional 2019, and Asta PowerProject version 15.0.01.489 in tackling RLP and RCPSP.

3.5. Serial scheduling scheme (SSS)

As mentioned earlier, results obtained by the software are also compared with the solutions of Serial Scheduling Scheme (SSS) which is a heuristic algorithm for RCPSP and was first introduced by Kelly [43]. SSS aims to minimize the project total duration while satisfying precedence and resource constraints. The results obtained by the aforesaid software packages are also compared with the solutions provided by this heuristic method. SSS is implemented in MATLAB 2019 and the activity ID is used as the activity leveling priority. To be consistent with the other attempts, the maximum daily available resources of 10 is used for every resource type over all the practiced instances. SSS involves the four steps explained below:

- Step 1:* Schedule the activities according to the chosen priority which is activity ID in this case;
- Step 2:* The first activity from the prioritized list of activities is selected with the condition that all of its predecessors been already scheduled;
- Step 3:* The selected activity is scheduled according to its possible Early Start (ES) date such that both of the precedence and resource constraints are satisfied;
- Step 4:* Step 2 and 3 are repeated for the next activity on the prioritized list. This process is repeated for all the activities on the list until all the activities are scheduled.

4. Comparative performance evaluations

This section summarizes the results of the comparative study on the resource allocation capabilities of PMSPs. Detailed results for each of the individual instances can be found in Albayati [44] which is not repeated here for the sake of brevity. Performance evaluations also involve comparisons with the results obtained by SSS. As discussed in section 3.1, 640 instances having up to 500 activities with up to 15 resource types have

been employed. Unleveled durations of the practiced instances ranges from 16 to 2,146 days. A total of 14 different method/priority combinations – including activity ID for SSS – are experimented. All of the experiments are performed on a Laptop computer running on an Intel® Core™ i7-5500 CPU at 2.40 GHz, with an operating system of 64 bit, and 12 GB of RAM.

4.1. Exercised leveling priorities

Leveling priority simply indicates the order of the leveling process. As illustrated in Table 2, 14 of the most common leveling priorities that are frequently implemented in the literature and are used in practice are exercised in this study. As seen in Table 2, the Standard priority is used for MSP while for Primavera P6 four different priorities coupled with two distinct sorting orders are applied.

Table 2. List of priorities selected for each leveling process

Software Method	Exercised Priority	
	Denotation	Explanation
MSP 2019	S	Standard
Primavera P6	ID (Asc.)	Activity ID-Ascending
	ID (Desc.)	Activity ID-Descending
	TF (Asc.)	Total Float-Ascending
	TF (Desc.)	Total Float-Descending
	ES (Asc.)	Early Start-Ascending
	ES (Desc.)	Early Start-Descending
	LF (Asc.)	Late Finish-Ascending
	LF (Desc.)	Late Finish-Descending
Asta PowerProject	TF	Total Float
	ID	Activity ID
	TS	Task Start Date
	MP	Multi Priority
Serial Scheduling Scheme	ID	Activity ID

4.2. Computation time for RCPSP

Computation times for every individual instance under each of the practiced priority schemes are measured. As mentioned earlier in section 3.1, 10 instances for each problem configuration of: activity number (50, 100, 200, and 500), resource type number (1, 5, 10, and 15), and OS (0.1, 0.3, 0.5, and 0.7) were generated totaling 640 sample problems. Table 3 presents the average CPU times for each 10 similarly-configured problems. With respect to the results, Primavera P6 is experienced to be the fastest method with a total average of 1 second which is followed by MSP with a total average of 116 seconds. The solution times of the Asta PowerProject and Serial Scheduling Scheme algorithm are significantly higher compared to those of Primavera P6 and MSP.

Since in real-life projects the number of activities is very high and are typically more than 300 work items [45], the duration of the leveling process can play a major role and can be regarded as one of the chief deciding factors in preference of a software package in practice. The results of this study indicate that all software packages require relatively the same amount of computation time to level small-scale instances with 50 activities. For medium-scale instances including 100 to 200 activities Primavera P6 is shown to be the fastest method compared to the other approaches. For large-scale instances with 500 activities only Primavera P6 is experienced to achieve results in reasonable processing times as the other programs were significantly slow in leveling the problems.

4.3. Leveling performances of experimented approaches

Since no optimal solution is available in the literature for any of the instances used, the percent deviations from the best solutions with the smallest objective function values found – denoted by Upper Bound (UB) – are used instead to evaluate the effectiveness of the methods for solution of RLPs. That is, UB is decided by determining the best objective function found through the 13 different method/priority combinations (Table 2).

Table 3. Average computation time for each problem set

No. of Acts.	No. of Res.	Avg. CPU Time (Sec.)													
		MSP 2019	PRIMAVERA P6 2019								ASTA 2019				SSS
		S	ID (Asc.)	ID (Desc.)	TF (Asc.)	TF (Desc.)	ES (Asc.)	ES (Desc.)	LF (Asc.)	LF (Desc.)	TF	ID	TS	MP	ID
50	1	0.6	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	1	1	1	1	32
	5	1	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	2	2	2	2	58
	10	2	1	1	1	1	1	1	1	1	16	16	11	15	72
	15	5	1	1	1	1	1	1	1	1	40	40	35	41	72
100	1	1.8	1	1	1	1	1	1	1	1	2	2	1.5	2	81
	5	6	1	1	1	1	1	1	1	1	18	18	16	18	184
	10	12	1	1	1	1	1	1	1	1	22	22	19	22	232
	15	20	1	1	1	1	1	1	1	1	31	31	26	31	249
200	1	9	1	1	1	1	1	1	1	1	19	19	10	19	244
	5	36	1	1	1	1	1	1	1	1	49	48	35	49	440
	10	76	1	1	1	1	1	1	1	1	102	99	72	102	450
	15	120	1	1	1	1	1	1	1	1	207	186	127	208	483
500	1	58	1	1	1	1	1	1	1	1	54	54	29	52	1528
	5	300	1	1	1	1	1	1	1	1	414	409	301	417	2983
	10	523	1	1	1	1	1	1	1	1	967	905	811	917	2804
	15	692	1.15	1.15	1.15	1.15	1.15	1.15	1.15	1.15	7875	7449	2450	7971	14822
Avg.		116	1	1	1	1	1	1	1	1	614	581	247	617	1546

After setting the UB values, deviations from UB for every leveling attempt are measured in percentages. Insight is provided into the obtained results as follows.

4.3.1. Performance comparisons based on number of activities

For performance comparisons, initially, the average percent deviations from UB are analyzed with respect to the number of project activities and inferences are made. With respect to total average deviations given in Table 4, Task Start Date priority used for Asta PowerProject happen to provide the best results with a deviation of 6.65% while Primavera P6 with Total Float-Ascending priority with an average deviation of 27.45% turns out to perform the worst. For instances with 50 activities, the best solution with an average deviation of

2.62% was found by Primavera P6 when TF (Desc.) priority was used and it was followed by the rest of priorities used with the same software. When Primavera is set aside, Asta PowerProject is experienced to provide the lowest average deviation of 9.92% when TF priority is used and next in order is the solutions found by using other priorities with PowerProject. Succeeding PowerProject, MSP with an average deviation of 14.08% is discovered to perform poorer than the first two software. For the rest of the instances with 100, 200, and 500 activities, the best solutions with the lowest average deviations were found by PowerProject when TS priority was used. In this leveling attempt, average deviations of 4.69%, 9.60%, and 0.76% were experimented for problems including 100, 200, and 500 activities, respectively.

Table 4. Average deviation from UB based on activity numbers

		Avg. Deviation based on Activity Numbers (%)												
No. of Acts.	No. of Res.	MSP 2019		PRIMAVERA P6 2019						ASTA 2019				
		S	ID (Asc.)	ID (Desc.)	TF (Asc.)	TF (Desc.)	ES (Asc.)	ES (Desc.)	LF (Asc.)	LF (Desc.)	TF	ID	TS	MP
50	1	5.77	11.69	10.85	80.22	8.80	12.93	11.30	14.50	9.00	1.93	4.32	2.95	2.72
	5	15.36	1.87	1.60	3.16	0.71	2.79	1.71	3.81	1.20	12.63	13.76	12.58	12.97
	10	17.95	0.85	1.07	1.46	0.95	0.81	1.15	1.38	0.65	12.82	15.65	13.52	13.11
	15	17.23	0.04	0.04	0.22	0.04	0.04	0.04	0.16	0.04	12.29	15.13	12.91	12.31
	Avg.	14.08	3.61	3.39	21.27	2.62	4.14	3.55	4.96	2.72	9.92	12.22	10.49	10.28
100	1	5.78	9.96	5.95	6.73	6.91	9.02	6.81	11.62	4.55	1.40	3.70	1.68	1.43
	5	13.07	3.45	2.14	3.42	1.68	2.76	2.57	4.01	1.63	10.79	10.57	7.82	11.28
	10	12.76	1.18	0.84	1.70	0.71	0.88	0.92	1.55	0.72	10.80	11.20	8.82	11.24
	15	129.63	102.27	102.27	103.58	102.10	102.10	102.37	102.85	102.10	13.56	14.92	0.43	13.35
	Avg.	40.31	29.21	27.80	28.86	27.85	28.69	28.17	30.00	27.25	9.14	10.10	4.69	9.33
200	1	95.90	107.01	100.88	103.57	98.36	107.74	100.93	109.52	98.45	0.97	20.37	14.72	19.91
	5	12.05	2.32	1.28	2.42	0.99	2.15	1.51	3.11	1.03	13.39	12.28	10.62	12.59
	10	12.64	0.22	0.09	0.64	0.07	0.19	0.19	0.76	0.04	12.60	13.10	9.42	11.72
	15	12.26	0.52	0.48	0.89	0.48	0.49	0.52	0.83	0.48	11.42	10.51	7.94	11.20
	Avg.	33.21	27.52	25.68	26.88	24.97	27.64	25.79	28.55	25.00	9.60	14.07	10.67	13.86
500	1	3.04	6.73	2.84	5.42	2.19	6.25	3.91	7.48	2.27	1.38	2.16	0.63	1.59
	5	5	32	31	33	31	32	31	33	31	4	3	1	4
	10	7.06	47.22	47.02	48.78	47.01	47.19	47.25	48.81	46.99	4.02	4.71	0.93	4.84
	15	5.79	40.77	40.82	44.26	40.76	40.77	41.35	43.94	40.76	3.29	4.17	0.57	4.13
	Avg.	5.30	31.72	30.37	32.81	30.13	31.51	30.92	33.42	30.16	3.11	3.62	0.76	3.74
Total Avg.		23.23	23.02	21.81	27.45	21.39	23.00	22.11	24.24	21.28	7.94	10.00	6.65	9.30

4.3.2. Performance comparisons based on number of resource types

Second to number of project activities, the effect of resource type numbers is also analyzed on the performance of the practiced PMSPs. Considering the total average deviations given in Table 5, it can be clearly observed that the number of resource types has a significant impact on the leveling performance of all the experimented leveling approaches. When results are evaluated based on the number of different resources, it is realized that Asta PowerProject with TS priority provides the best solutions with a total average deviation of

6.64%. On the other hand, Primavera P6 with Total Float-Ascending priority with a total average deviation of 27.32% is observed to perform the worst. For instances with a single resource type, the best solution with an average deviation of 1.42% was obtained by PowerProject when TF priority was used. For the rest of the problems with 5, 10, and 15 resource types, the best solutions with the lowest average deviations were achieved by PowerProject when TS priority was selected. In this leveling attempt, average deviations of 7.98%, 8.13%, and 5.46% were experimented for problems including 5, 10, and 15 resource types, respectively.

Table 5. Average deviation from UB based on resource type numbers

		Avg. Deviation based on Resource Type Numbers (%)												
No. of Acts.	No. of Res.	MSP 2019		PRIMAVERA P6 2019						ASTA 2019				
		S	ID (Asc.)	ID (Desc.)	TF (Asc.)	TF (Desc.)	ES (Asc.)	ES (Desc.)	LF (Asc.)	LF (Desc.)	TF	ID	TS	MP
50	1	5.77	11.69	10.85	80.22	8.80	12.93	11.30	14.50	9.00	1.93	4.32	2.95	2.72
100		5.78	9.96	5.95	6.73	6.91	9.02	6.81	11.62	4.55	1.40	3.70	1.68	1.43
200		95.90	107.01	100.88	103.57	98.36	107.74	100.93	109.52	98.45	0.97	20.37	14.72	19.91
500		3.04	6.73	2.84	5.42	2.19	6.25	3.91	7.48	2.27	1.38	2.16	0.63	1.59
	Avg.	27.62	33.85	30.13	48.99	29.07	33.98	30.74	35.78	28.57	1.42	7.64	4.99	6.41
50	5	15.36	1.87	1.60	3.16	0.71	2.79	1.71	3.81	1.20	12.63	13.76	12.58	12.97
100		13.07	3.45	2.14	3.42	1.68	2.76	2.57	4.01	1.63	10.79	10.57	7.82	11.28
200		12.05	2.32	1.28	2.42	0.99	2.15	1.51	3.11	1.03	13.39	12.28	10.62	12.59
500		5	32	31	33	31	32	31	33	31	4	3	1	4
	Avg.	11.45	9.95	8.96	10.45	8.48	9.89	9.23	11.09	8.62	10.14	10.02	7.98	10.31
50	10	17.95	0.85	1.07	1.46	0.95	0.81	1.15	1.38	0.65	12.82	15.65	13.52	13.11
100		12.76	1.18	0.84	1.70	0.71	0.88	0.92	1.55	0.72	10.80	11.20	8.82	11.24
200		12.64	0.22	0.09	0.64	0.07	0.19	0.19	0.76	0.04	12.60	13.10	9.42	11.72
500		7.06	47.22	47.02	48.78	47.01	47.19	47.25	48.81	46.99	4.02	4.71	0.93	4.84
	Avg.	12.37	11.88	11.60	12.60	11.44	11.79	11.75	12.72	11.41	10.08	10.93	8.13	10.24
50	15	17.23	0.04	0.04	0.22	0.04	0.04	0.04	0.16	0.04	12.29	15.13	12.91	12.31
100		129.63	102.27	102.27	103.58	102.10	102.10	102.37	102.85	102.10	13.56	14.92	0.43	13.35
200		12.26	0.52	0.48	0.89	0.48	0.49	0.52	0.83	0.48	11.42	10.51	7.94	11.20
500		5.79	40.77	40.82	44.26	40.76	40.77	41.35	43.94	40.76	3.29	4.17	0.57	4.13
	Avg.	41.23	35.90	35.90	37.24	35.84	35.85	36.07	36.95	35.84	10.14	11.18	5.46	10.25
Total Avg.		23.17	22.90	21.65	27.32	21.21	22.88	21.95	24.13	21.11	7.94	9.94	6.64	9.31

4.3.3. Performance comparisons based on network complexity (OS)

Eventually, the impact of Network Complexity (OS) on the performance of the resource leveling approaches is also evaluated. With regard to the total average deviation amounts summarized in Table 6, the significance of the effect of this factor on the results can also be verified. When solutions are analyzed with respect to complexity of the networks, it is concluded that Asta PowerProject when ran using TS priority achieves the best results with a total average deviation of 6.65%. Whereas,

Table 6. Average deviation from UB based on OS factor

Primavera P6 with Total Float-Ascending priority with a larger total average deviation of 25.53% is experienced to provide the worst solutions. For the instances with OS factors of 0.1, 0.3, and 0.5, respectively, the best solutions with average deviation values of 3.47%, 5.09%, and 6.39% were achieved by PowerProject when TS priority was chosen. In addition, for instances with the greatest OS value of 0.7, the best solution with an average deviation of 7.74% was located by PowerProject when TF priority was selected.

Table 6. Average deviation from UB based on OS factor

		Avg. Deviation based on OS (%)												
No. of Acts.	OS	MSP 2019		PRIMAVERA P6 2019						ASTA 2019				
		S	ID (Asc.)	ID (Desc.)	TF (Asc.)	TF (Desc.)	ES (Asc.)	ES (Desc.)	LF (Asc.)	LF (Desc.)	TF	ID	TS	MP
50	0.1	14.58	4.16	4.12	5.32	2.96	4.53	4.50	5.39	3.27	8.75	12.34	9.81	9.42
100		13.28	8.67	6.83	9.65	6.39	7.25	7.71	9.75	6.38	8.26	7.73	1.90	8.51
200		6.50	0.99	0.86	1.74	0.84	0.90	1.04	1.80	0.82	7.83	6.69	2.01	5.97
500		4.13	44.03	43.42	48.15	43.28	43.88	44.52	48.03	43.29	4.43	4.01	0.17	4.62
Avg.		9.62	14.46	13.81	16.21	13.37	14.14	14.44	16.24	13.44	7.32	7.69	3.47	7.13
50	0.3	14.31	3.49	3.47	4.68	2.81	3.59	3.75	4.77	2.55	10.08	12.58	10.44	10.36
100		23.89	15.55	13.98	14.98	14.27	14.63	14.81	16.01	13.78	9.22	9.11	3.69	9.73
200		9.69	2.32	1.04	2.08	1.04	2.23	1.63	2.74	1.08	7.88	8.31	6.24	8.47
500		4.87	35.82	34.75	36.79	34.50	35.64	35.39	37.17	34.61	4.54	4.32	0.00	5.30
Avg.		13.19	14.30	13.31	14.63	13.16	14.02	13.90	15.17	13.00	7.93	8.58	5.09	8.47
50	0.5	13.39	3.34	2.89	3.60	2.48	1.20	2.85	4.73	2.69	10.55	12.17	11.05	10.93
100		49.66	36.16	34.70	35.12	35.11	36.04	34.87	36.74	34.06	11.20	12.66	5.89	11.85
200		11.33	3.08	1.58	2.93	1.34	2.90	1.58	3.76	1.25	10.17	10.27	8.53	9.97
500		6.03	27.77	26.20	28.25	25.94	27.46	26.48	29.04	25.91	3.12	4.05	0.08	4.32
Avg.		20.10	17.59	16.34	17.48	16.22	16.90	16.45	18.57	15.98	8.76	9.79	6.39	9.27
50	0.7	14.03	3.46	3.07	40.69	2.24	4.43	3.10	4.96	2.39	10.29	11.77	10.65	10.40
100		74.41	56.48	55.68	55.68	55.62	56.83	55.29	57.52	54.77	7.86	10.88	7.28	7.21
200		105.29	103.63	99.21	100.73	96.64	104.49	98.86	105.87	96.82	12.46	30.94	25.88	30.96
500		6.20	19.27	17.13	18.06	16.81	19.08	17.27	19.45	16.84	0.35	2.11	2.78	0.73
Avg.		49.98	45.71	43.77	53.79	42.83	46.21	43.63	46.95	42.71	7.74	13.93	11.65	12.33
Total Avg.		23.22	23.01	21.81	25.53	21.39	22.82	22.10	24.23	21.28	7.94	10.00	6.65	9.30

4.3.4. Makespan minimization capabilities of experimented approaches

In addition to the leveling capabilities, makespan minimization abilities are also considered when solving RCPSP. Similar to UB, minimum makespan over the 14 different method/priority combinations (Table 2) is determined for each individual problem. Afterwards, deviations from this amount for every leveling attempt are measured

in percentages. Table 7 shows the average percent deviations from minimum makespan for each 10 similarly-configured problems. When total average percent deviations from minimum levelled makespans are considered, it can be concluded that Microsoft Project 2019 is able to achieve the best results with a total average deviation of 4.63%. Whereas, SSS with a large total average deviation of 23.14% provides the least favorable solutions.

Table 7. Average deviation from minimum makespan for each problem set

No. of Acts.	No. of Res.	Avg. Deviation from Min. Makespan (%)													
		MSP 2019	PRIMAVERA P6 2019								ASTA 2019				SSS
		S	ID (Asc.)	ID (Desc.)	TF (Asc.)	TF (Desc.)	ES (Asc.)	ES (Desc.)	LF (Asc.)	LF (Desc.)	TF	ID	TS	MP	ID
50	1	16.89	4.92	6.48	3.35	10.45	3.03	5.62	1.36	9.86	24.35	20.01	22.53	24.28	40.17
	5	1.90	27.82	27.89	23.98	30.84	24.99	28.69	22.28	29.83	4.73	3.79	5.01	4.65	37.52
	10	0.27	30.61	30.66	29.84	30.78	30.38	30.40	29.99	30.87	7.82	3.44	6.73	6.75	31.35
	15	0.66	32.31	32.24	32.01	32.31	32.31	32.30	31.99	32.31	8.66	3.50	7.11	7.69	32.31
	Avg.	4.93	23.92	24.32	22.30	26.10	22.68	24.25	21.41	25.72	11.39	7.69	10.35	10.84	35.34
100	1	9.30	3.33	9.89	4.66	12.04	4.62	8.30	0.32	12.25	19.13	14.56	18.28	19.28	32.36
	5	4.97	18.16	22.38	19.08	23.54	20.30	21.12	17.38	23.67	2.89	5.39	11.61	2.86	25.56
	10	6.12	22.47	22.95	22.10	23.30	22.79	22.87	21.88	23.25	3.39	2.65	10.09	2.05	23.25
	15	5.56	21.64	21.62	21.01	21.77	21.77	21.59	21.24	21.77	3.05	1.90	9.94	3.10	21.77
	Avg.	6.49	16.40	19.21	16.71	20.16	17.37	18.47	15.21	20.24	7.12	6.13	12.48	6.82	25.74
200	1	7.04	1.76	7.05	3.85	8.72	1.97	6.09	0.27	8.43	13.84	10.70	13.06	13.83	7.29
	5	2.59	16.63	19.56	17.17	20.15	17.03	18.92	15.11	19.98	2.71	3.63	7.72	2.01	20.59
	10	3.34	19.15	19.56	18.97	19.68	19.26	19.37	18.60	19.67	1.97	1.53	7.16	0.89	19.74
	15	3.97	18.94	18.99	18.65	19.00	18.95	18.94	18.70	19.00	2.56	2.03	7.22	2.15	19.00
	Avg.	4.24	14.12	16.29	14.66	16.89	14.30	15.83	13.17	16.77	5.27	4.47	8.79	4.72	16.66
500	1	6.99	1.44	7.77	3.28	9.21	2.09	5.93	0.05	9.02	10.92	9.02	12.29	10.22	7.77
	5	2.00	15.00	18.00	16.00	19.00	16.00	18.00	14.00	19.00	2.00	3.00	6.00	1.00	16.00
	10	1.42	18.12	18.57	17.80	18.60	18.19	18.45	17.59	18.61	2.53	1.37	5.32	1.54	18.61
	15	1.40	16.93	16.97	16.26	16.98	16.93	16.87	16.31	16.98	2.49	0.65	5.21	1.63	16.98
	Avg.	2.95	12.87	15.33	13.34	15.95	13.30	14.81	11.99	15.90	4.49	3.51	7.21	3.60	14.84
Total Avg.		4.63	16.85	18.82	16.75	19.77	16.91	18.33	15.47	19.64	7.08	5.43	9.72	6.51	23.14

5. Discussion of findings

In the light of the performance evaluations discussed earlier, some guidelines will be provided herein for the project managers for selection of the appropriate software packages for their real-life resource allocation needs. Selecting the suitable tool is crucial because real-life projects can include a diverse number of activities and resources which makes the allocation process quite complicated. Guidelines are provided based on RLP and RCPSP capabilities for each of the project sizes (of 50, 100,

200, and 500 activities) as follows. For small RLP projects (~50 activities) with low resource type numbers (up to 5 resources), Primavera P6 2019 with Total Float-Descending priority is recommended. For medium RLP projects (~100 to ~200 activities) with small resource type numbers (up to 5 resources), Primavera P6 2019 with Late Finish-Descending priority is recommended. For the large RLP projects (~500 activities) with low resource type number (up to 5 resources), Asta PowerProject with Task Start Date priority is recommended. Ultimately, for all the RLP instances

regardless of the number of activities, with large resource type numbers (more than 5 resources), Asta PowerProject with Task Start Date priority is recommended. A summary of the RLP performance of the approaches is demonstrated in Fig. 2 while Fig. 3 depicts efficiency of the practiced methods for over RCPSP.

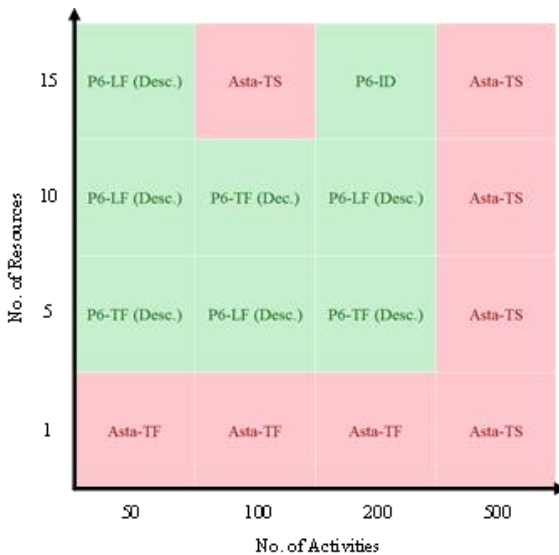


Fig. 2. Guideline for selection of suitable tool for RLP based on deviation from UB

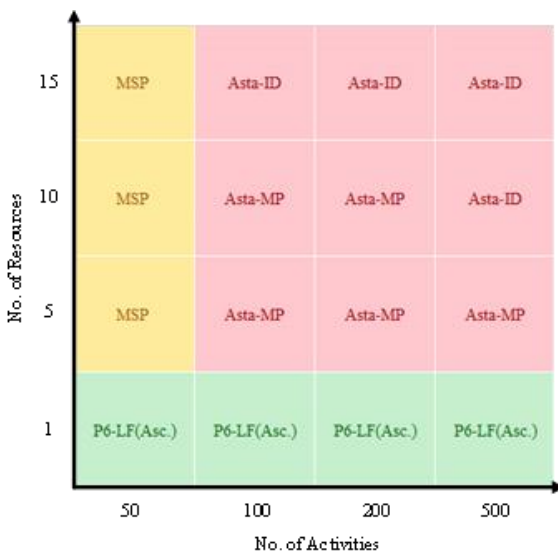


Fig. 3. Guideline for selection of suitable tool for RCPSP based on deviation from minimum makespan

6. Conclusions

Despite significance of resource allocation in real-life projects, studies focusing on the leveling capabilities of the commonly practiced commercial software packages is relatively scant. The widespread use of such programs, in addition, the ease of access to their resource leveling modules further substantiates the importance of this subject matter. Eventually, this study has taken a significant step in filling this gap by borrowing 640 examples from the literature. Minor adjustments were made to adapt the existing problems to leveling with resource availability considerations. The sample problems were fed into the latest versions – as per the date of this study – of three more widely used commercial project management software packages of Microsoft Project Professional 2019, Primavera P6 Professional 2019, and Asta Powerproject version 15.0.01.489. Comparative analyses were carried out over the foregoing software together with the Serial Scheduling Scheme algorithm, with the aim of experimenting their resource leveling as well as resource constrained project scheduling performances. Throughout the analyses, several leveling priorities that are frequently implemented in the literature and are used in practice were exercised. Microsoft Project was experimented by using the standard priority, Primavera P6 with eight different priorities, and Asta PowerProject by selecting four distinct priority rules.

The findings reveal while all the three software packages manage to provide comparable results, Asta PowerProject – especially with Task Start Date and Multi Priority priorities – transpire to be the all-round best performing method. Nevertheless, especially for large problems that include high resource type numbers, Asta PowerProject and Microsoft Project were found not to be as efficient as Primavera P6. The authors acknowledge that the CPU times reported could have some degree of measurement error due to inevitable human error, for, adopting a stopwatch for process-time measurement purposes. Nevertheless, since the same technique is used during all the attempts, the results are firmly comparable. Hence, if the critical parameter is the

computation time of the leveling process, project planners are recommended to use Primavera P6 because of its unmatched promptness. Taking into account the time required to level the resources by the commercial software packages, it is concluded that they, except for Primavera P6, fail to solve large-scale problems within reasonable computation times. Thus, integrating faster algorithms for solution of resource constrained leveling problems appear to be an area which needs further improvements. On the other hand, transferring data among the software is experienced to be an arduous and cumbersome endeavor. The challenges and practical hurdles to utilization of the software for resource leveling purposes as well as some practical information as to how should the data be imported to the different software packages are provided in this study. Nonetheless, further improvement of cross-platform data importing/exporting capabilities of the software is imperative to smoothness and simplicity of the leveling features they accommodate.

Declaration

Funding

This research received no external funding.

Author Contributions

N.H.F. Albayati: Software, Validation, Formal analysis, Investigation, Writing - Original Draft.; S. Aminbakhsh: Conceptualization, Methodology, Investigation, Writing - Original Draft, Writing - Review & Editing, Supervision.

Acknowledgments

Not applicable.

Data Availability Statement

The data presented in this study are available on request from the corresponding author.

Ethics Committee Permission

Not applicable.

Conflict of Interests

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

- [1] Abdel-Basset M, Ali M, Atef A (2020) Resource levelling problem in construction projects under neutrosophic environment. *The Journal of Supercomputing* 76(2): 964–988. <https://doi.org/10.1007/s11227-019-03055-6>
- [2] Kelley J E, Walker MM (1959) Critical-path planning and scheduling. In: *Proceedings of Eastern Joint IRE-AIEE-ACM Computer Conference*. <https://doi.org/10.1145/1460299.1460318>
- [3] Malcolm DG, Roseboom J, Clark CE, Fazar W (1959) Application of a technique for research and development program evaluation. *Operations Research* 7(5): 646–669. <https://doi.org/10.1287/opre.7.5.646>
- [4] Khanzadi M, Attar AH, Bagherpour M (2016) Finding optimum resource allocation to optimizing construction project time/cost through combination of artificial agents CPM and GA. *Periodica Polytechnica-Civil Engineering* 60(2): 169–180. <https://doi.org/10.3311/ppci.7883>
- [5] Kaveh A, Khanzadi M, Alipour M, Naraki MR (2015) CBO and CSS algorithms for resource allocation and time-cost trade-off. *Periodica Polytechnica-Civil Engineering*, 59(3), 361–371. <https://doi.org/10.3311/ppci.7788>
- [6] Kastor A, Sirakoulis K (2009) The effectiveness of resource levelling tools for Resource Constraint Project Scheduling Problem. *International Journal of Project Management* 27(5): 493–500. <https://doi.org/10.1016/j.ijproman.2008.08.006>
- [7] Altun M, Sonmez R, Akcamete A (2020) A mixed integer programming method for multi-project resource leveling. *Journal of Construction Engineering, Management & Innovation* 3(2): 131–140. <https://doi.org/10.31462/jcemi.2020.02131140>
- [8] Bettemir ÖH, Erzurum T (2019) Comparison of resource distribution metrics on multi-resource projects. *Journal of Construction Engineering, Management & Innovation* 2(2): 93–102. <https://doi.org/10.31462/jcemi.2019.02093102>
- [9] Kolisch R, Hartmann S (1999) Heuristic algorithms for the resource-constrained project scheduling

- problem: Classification and Computational Analysis. Springer eBooks, 147–178. https://doi.org/10.1007/978-1-4615-5533-9_7
- [10] Kolisch R, Hartmann S (2006) Experimental investigation of heuristics for resource-constrained project scheduling: An update. *European Journal of Operational Research* 174(1): 23–37. <https://doi.org/10.1016/j.ejor.2005.01.065>
 - [11] Kaveh A, Rajabi F (2021) Fuzzy-multi-mode resource-constrained discrete time-cost-resource optimization in project scheduling using ENSCBO. *Periodica Polytechnica-Civil Engineering* 66(1): 50–62. <https://doi.org/10.3311/ppci.19145>
 - [12] Waligóra G (2022) Genetic algorithm for resource leveling problems under various objective functions. In: *Proceedings of 26th International Conference on Methods and Models in Automation and Robotics, Międzyzdroje, Poland*. <https://doi.org/10.1109/mmarm55195.2022.9874286>
 - [13] Selvam G, Tadepalli TCM (2019) Genetic algorithm-based optimization for resource leveling problem with precedence constrained scheduling. *The International Journal of Construction Management*. <https://doi.org/10.1080/15623599.2019.1641891>
 - [14] Brucker P, Drexl A, Mohring R, Neumann K, Pesch E (1999) Resource-constrained project scheduling: notation, classification, models, and methods. *European Journal of Operational Research* 112(1): 3–41.
 - [15] Blazewicz J, Lenstra JK, Kan AR (1983) Scheduling subject to resource constraints: classification and complexity. *Discrete Applied Mathematics* 5(1): 11–24.
 - [16] Garey MR, Johnson DS (1979) *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman.
 - [17] Demeulemeester E, Herroelen W, Leus R (2008) Resource-Constrained Project Scheduling. In *ISTE eBooks*. <https://doi.org/10.1002/9780470611227>
 - [18] Valls V, Ballestín F, Quintanilla MS, (2008) A hybrid genetic algorithm for the resource-constrained project scheduling problem. *European Journal of Operational Research*, 185(2): 495–508. <https://doi.org/10.1016/j.ejor.2006.12.033>
 - [19] Beşikci U, Bilge Ü, Ulusoy G (2015) Multi-mode resource constrained multi-project scheduling and resource portfolio problem. *European Journal of Operational Research*, 240(1): 22–31. <https://doi.org/10.1016/j.ejor.2014.06.025>
 - [20] Bruni ME, Beraldi P, Guerriero F (2015) The Stochastic Resource-Constrained Project Scheduling Problem. In *Springer eBooks* (pp. 811–835). https://doi.org/10.1007/978-3-319-05915-0_7
 - [21] Zhou Y, Miao J, Yan B, Zhang Z (2021) Stochastic resource-constrained project scheduling problem with time varying weather conditions and an improved estimation of distribution algorithm. *Computers & Industrial Engineering* 157: 107322. <https://doi.org/10.1016/j.cie.2021.107322>
 - [22] Sung IK, Choi BJ, Nielsen P (2020) Reinforcement Learning for Resource Constrained Project Scheduling Problem with Activity Iterations and Crashing. *IFAC-PapersOnLine*, 53(2): 10493–10497. <https://doi.org/10.1016/j.ifacol.2020.12.2794>
 - [23] Sallam KM, Chakraborty RK, Ryan MJ (2021) A reinforcement learning based multi-method approach for stochastic resource constrained project scheduling problems. *Expert Systems With Applications* 169: 114479. <https://doi.org/10.1016/j.eswa.2020.114479>
 - [24] Azeem M, Ullah F, Thaheem MJ, Qayyum S (2020) Competitiveness in the construction industry: A contractor's perspective on barriers to improving the construction industry performance. *Journal of Construction Engineering, Management & Innovation*, 3(3): 193–219. <https://doi.org/10.31462/jcemi.2020.03193219>
 - [25] Johnson RV (1992) Resource Constrained Scheduling Capabilities of Commercial Project Management Software.
 - [26] Maroto C, Tormos P (1994) Project management: an evaluation of software quality. *International Transactions in Operational Research* 1(2): 209–221. <https://doi.org/10.1111/1475-3995.d01-22>
 - [27] Kolisch R, Schwindt C, Sprecher A (1999) Benchmark Instances for Project Scheduling Problems. *Springer US eBooks*, 197–212. https://doi.org/10.1007/978-1-4615-5533-9_9
 - [28] Mellentien C, Trautmann, N (2001) Resource allocation with project management software. *OR Spectrum* 23(3): 383–394. <https://doi.org/10.1007/pl00013358>
 - [29] Hekimoglu O (2007) Comparison of the resource allocation capabilities of project management software packages in resource constrained project scheduling problems. M.Sc. Thesis, Middle East Technical University, Turkey.

- [30] Çekmece K (2009) The resource allocation capabilities of commercial project management software packages for resource constrained project scheduling problem. M.Sc. thesis, Middle East Technical University, Turkey.
- [31] Son J, Mattila, KG (2004) Binary resource leveling model: activity splitting allowed. *Journal of the Construction Division and Management ASCE* 130(6): 887–894.
- [32] Iranagh MA, Sonmez R (2012) A genetic algorithm for resource levelling of construction projects. Smith, S.D (Ed) *Procs 28th Annual ARCOM Conference*, vol. 1047, pp. 1054, Edinburgh, UK.
- [33] Rezvan Khah E (2014) The Resource allocation capabilities of commercial construction project management software for the resource leveling problem. M.Sc. Thesis, Middle East Technical University, Turkey.
- [34] Burgess A, Killebrew JB (1962) Variation in activity level on a cyclical arrow diagram. *Journal of Industrial Engineering* 13(2): 76–83.
- [35] Kuhlang P (2011) Evaluation of project portfolio management software for ABB engineering centers, M.Sc. Thesis, TU Wien, Vienna, Austria.
- [36] Gharaibeh HM (2014) Evaluating Project Management Software Packages Using a Scoring Model—A Comparison between MS Project and Primavera. *Journal of Software Engineering and Applications* 07(7): 541–554. <https://doi.org/10.4236/jsea.2014.77050>
- [37] Farid F, Manoharan S (1996) Comparative analysis of resource-allocation capabilities of project management software packages. *Project Management Journal* 27(2): 35–44.
- [38] Maroto C, Tormos P, Lova A (1999) The Evolution of Software Quality in Project Scheduling. In: Węglarz, J. (eds) *Project Scheduling. International Series in Operations Research & Management Science*, vol 14. Springer, Boston, MA. https://doi.org/10.1007/978-1-4615-5533-9_11
- [39] Demeulemeester E, Herroelen W (1992) A branch-and-bound procedure for the multiple resource-constrained project scheduling problem. *Management Science* 38(12): 1803–1818. <https://doi.org/10.1287/mnsc.38.12.1803>
- [40] Kolisch R (1999) Resource allocation capabilities of commercial project management software packages. *Interfaces* 29(4): 19–31. <http://www.jstor.org/stable/25062501>
- [41] Trautmann N, Baumann PS (2009) Resource-allocation capabilities of commercial project management software: An experimental analysis. <https://doi.org/10.1109/iccie.2009.5223881>
- [42] Demeulemeester E, Vanhoucke M, Herroelen W (2003) RanGen: A random network generator for activity-on-the-node networks. *Journal of Scheduling* 6(1): 17–38.
- [43] Kelley JE (1963) The critical-path method: resource planning and scheduling. *Industrial scheduling*.
- [44] Albayati NHF (2021) A Comparative Study on Resource Leveling Capabilities of Commercial Project Management Software Packages for Projects with Resource Constraints, M.Sc. Thesis, Atilim University, Ankara, Turkey.
- [45] Liberatore MJ, Pollack-Johnson B, Smith CA (2001) Project management in construction: Software use and research directions. *Journal of Construction Engineering and Management ASCE*, 127(2): 101–107.