Journal of Construction Engineering, Management & Innovation
2020 Volume 3 Issue 2 Pages 131-140
https://doi.org/10.31462/jcemi.2020.02131140

www.goldenlightpublish.com

RESEARCH ARTICLE

A mixed integer programming method for multi-project resource leveling

Murat Altun, Rifat Sonmez, Asli Akcamete*

Middle East Technical University, Civil Engineering Department, Ankara, Turkey

Abstract

In the implementation of construction projects, efficient resource planning plays a prominent role in developing cost-efficient solutions. Therefore, the decision makers should level the project resources with respect to the planned project schedule to reduce the project costs. In general, they usually focus on each project separately to optimize the project's resource usage according to the intended resource objective function of the resource leveling problem. However, in real life, multiple projects with shared resources may be executed simultaneously. Hence, separate evaluations of resource leveling problem for each project may result in sub-optimal solutions due to negligence of the effects of the shared resources for the projects. Therefore, the projects using shared resources should be leveled together to reach the global optimum solution. In this study, an optimization model is developed using Mixed Integer Programming (MIP) to minimize peak requirements of different resource types in multiple construction projects. The performance of the proposed model is tested with four case study projects with different project settings. The solutions that are achieved with the proposed method are compared with the optimum solutions of the traditional leveling models which optimize the resource usage for each project separately. The results show that the proposed method considerably improves portfolio performance compared to the results of the traditional method based on separate leveling of projects.

Keywords

Multiple project; Resource leveling; Shared resource optimization; Scheduling; Mixed integer programming

Received: 05 June 2020; Accepted: 28 June 2020

ISSN: 2630-5771 (online)

1. Introduction

In construction project management, a schedule is generally created using Critical Path Method based on the precedence relationships between activities. The schedule is usually re-adjusted according to allocation of resources such as labors and equipment. To allocate resources efficiently, resource allocation problems are presented. These problems comprise of both resource constrained project scheduling problem (RCSP) and resource leveling problem (RLP). In RCSP, availability of the resources during the projects constrains the project schedule [1]. Therefore, the scheduler aims to reduce project duration without exceeding resource availability constraints [2]. On the other hand, in RLP, the scheduler aims to improve efficiency of the resource consumptions in the project by keeping the project duration fixed [3]. Considering the nature of the construction projects, in general, compared to the resource limitations, time is more influential parameter to complete project successfully. Hence, solutions offered to

* Corresponding author
Email: akcamete@metu.edu.tr

solve RLP has a significant contribution to the project success, by enabling efficient resource planning within the constraint of predefined project duration.

In RLP, the start and finish dates of the activities on the critical path cannot be changed to keep project duration fixed. Therefore, the scheduler focuses on non-critical activities and changes their start and finish dates in the interval of their possible early and late dates for efficient resource planning. The efficiency of the developed schedule alternatives is tested with various objective functions in the literature. These objectives are only one or combination of multiple ones of following objectives: (i) minimizing the maximum resource demand (peak minimization) at a time during project implementation, (ii) minimizing resource usage fluctuations during the projects, (iii) minimizing idle time of resources during the projects, especially for labor and equipment resources [4].

In practice, solutions offered by widely used construction planning and management programs such as Microsoft Project 2016 and Primavera P6 for RLPs is very limited in generating competitive solutions for efficient resource leveling [5]. Therefore, the researchers developed various optimization models and algorithms to apply on RLPs for generating better results. The previous studies in the literature mainly focus on single or multiple resource based leveling problem in a single construction project. Early efforts on these studies were based on exact optimization methods [6–8]; however, as the number of activities and resources increase, the time spent by exact optimization methods to solve the problems also increases significantly. Moreover, this increase in the complexity of the problem limits the capabilities of the exact methods to reach optimum or near optimum solutions in RLPs [9]. Therefore, to improve the solution, especially to find near optimum solutions, heuristic optimization approaches were developed. However, in general, these heuristics were developed as problem-specific and depended on a solution procedure considering priority rules for the activities [10–13]. Hence, these solution procedures may not produce same efficiency in all RLP problems. Moreover, due to their deterministic characteristic, all independent runs for the same problem generates same solution for the deterministic heuristics. In other words, if the selected heuristic algorithm is inefficient to solve the selected RLP problem, the researcher needs to change the heuristic and select a more efficient algorithm to find a better solution. Therefore, considering this limitation of the heuristics on RLPs, metaheuristic algorithms that have capabilities to generate different solutions due to their stochastic natures were applied to the RLPs [14]. According to "Free Lunch Theory", no metaheuristics can show superior performance in all types of problems on other metaheuristic algorithms [15]. Therefore, different metaheuristics such as Genetic Algorithm [1,4,5,16] Particle Swarm Optimizer [17,18], Ant Colony Optimization [19,20], Harmony Search Algorithm [21], Colliding Bodies Optimization and Charged System Search [22], Differential Evolution Algorithm [23,24] were applied to RLPs to optimize resource usage and cost efficiency in the projects.

Application of optimization algorithms on single project RLPs, was widely studied; however, in real life, multiple projects may be performed in the same period. For these projects, same types of resources, especially labors and equipment, are expected to be shared; however, if the decision makers plan the schedule and resource allocation separately for each project, this leveling approach will most probably generate sub-optimal solutions. This happens because, resources for shared resource types are also leveled specifically for each project and the interactions between projects due to common resource usage are not taken into consideration. Therefore, in order to eliminate sub-optimal solutions and generate optimum or near optimum solutions for efficient portfolio resource planning, multiple projects should be evaluated together and their resources should be leveled together. In the literature, the studies on multi-project multi-resource leveling are very limited. In these studies, Wang et al. [25] developed an RLP model integrated with Differential Evolution Algorithm to minimize the variance of total resource consumption in the multiple projects. Guo et al. [26] minimized fluctuations for multiple resources in the multiple projects using Particle Swarm Optimization, whereas Cheng et al. [27] applied Discrete Symbiotic Organisms Search to the same problem. In all of these studies, two small-scale projects were taken into consideration. However, these projects are conceptual projects to show the efficiency of the developed metaheuristics on multi-project RLPs. On the other hand, when the size of real construction projects is considered, there is a gap on the application of optimization methods on multi-project RLPs with higher number of activities to see the performance of the developed methods on more realistic RLPs.

In this study, the multi-project multi-resource leveling problem (MPMRL) is modeled to level common resources of multiple projects together in a portfolio to reduce the required resource amount. Within this context, Section 2 introduces the proposed MPMRL model using Mixed Integer Programming to minimize the sum of peak usage of multiple resources for multiple construction projects. This section summarizes description of the problem, assumptions of the model, and the exact model. Section 3 is devoted to computational experiments for testing the proposed model with four different case studies. To compare the performance of the proposed model, an optimization model is constructed to optimize resource leveling in each project of the case study separately. Finally, Section 4 summarizes prominent findings of this study.

2. Multi-project multi-resource leveling optimization model

2.1. Problem description

In MPMRL problems, a set J={1,2,…m) of projects are evaluated together to optimize resource efficiency of the whole portfolio. Therefore, each project j is assumed to be started at a certain time $t_{0,j}$. Moreover, project j consists of a set of I_j={1,2,..,n} of activities excluding start (activity 0) and finish (activity n+1) dummy activities. Additionally, each activity uses at most K={1,2,..,k} different resource types. The precedence relationship between activities in project j is constructed by a set $E_j \subseteq I_j \times I_j$ in which each pair of activities (p,i) indicates that activity p is a predecessor of activity i. In order to develop the initial schedule, forward pass of CPM is followed to calculate project completion time. Therefore, precedence relations, start and completion time of the projects are used in the constraints of the model to optimize resource usage efficiency.

2.2. Model assumptions

The assumptions of the constructed model are as follows:

1. An activity can only start after its all predecessors are finished.
2. An activity must be performed without any interruption through its execution period.
3. Each project starts at a predetermined time and is performed in an overlapping period.
4. Common renewable resource types are utilized for all projects and can be shareable between the projects.
5. The site locations of the projects are assumed to be very near to each other. Therefore, considering proximity of the sites, no extra delay time is required to mobilize a resource from one site to the other one.

2.3. Exact optimization model

The objective of the model is to minimize weighted peak resource usage in the portfolio. Therefore, objective function is formulated as in Eq. (1).

$$\min \quad z = \sum_k w_k R_k \tag{1}$$

where z represents the objective function whereas w_k is weight coefficient of resource k and R_k is decision variable of peak usage amount of resource k.

The model is subject to the various constraints which are explained below one by one in Eqs. (2-9).

The first constraint satisfies the precedence relationship between the activities. The successor activity can only start with at least one day later than the finish time of its predecessor activity for Finish-to-Start activity relations (Eq. (2)).

$$F_{pj} \leq S_{ij} + 1 \qquad (p,i) \in E_j, \forall j \in J \tag{2}$$

The next four constraints determine start (S_{ij}) and finish (F_{ij}) time decision variables of the activity i in each project j (Eqs. (3-6)).

$$S_{ij} \leq tD_{ijt} + M\left(1 - D_{ijt}\right) \; \forall i \in I, \forall j \in J, \forall t \in T \tag{3}$$

$$F_{ij} \geq tD_{ijt} - M\left(1 - D_{ijt}\right) \; \forall i \in I, \forall j \in J, \forall t \in T \tag{4}$$

$$S_{ij} \geq t_{0,j} \qquad \forall i \in I, \forall j \in J \tag{5}$$

$$F_{ij} \leq PD_j \qquad \forall i \in I, \forall j \in J \tag{6}$$

where t, M, and PD_j present time, arbitrarily large number, and pre-calculated project completion time of project j. Moreover, D_{ijt} is binary variable to determine whether activity i of project j is performed in time t or not.

$$S_{ij} + Dur_{ij} - 1 = F_{ij} \qquad \forall i \in I, \forall j \in J \tag{7}$$

$$Dur_{ij} = \sum_{t} D_{ijt} \qquad \forall i \in I, \forall j \in J \tag{8}$$

Dur_{ij} represents duration of activity i in project j. Using Eq. (8), Finish time variables can be eliminating by replacing F_{ij} using S_{ij} and Dur_{ij} in Eqs (2,4,6) in practice.

The final constraint presents peak usage amount constraint for resource k (Eq. (9)).

$$\sum_{j}\sum_{i} D_{ijt} r_{ijk} \leq R_k \qquad \forall k \in K \tag{9}$$

where r_{ijk} presents daily usage amount of resource k for activity i in project j.

3. Computational experiments and discussions

The proposed model is tested with four case studies to investigate its performance for different portfolio settings. The projects in each case were generated using RanGen software which was specifically developed for resource constrained scheduling problems [28]. The details of the projects in each case are available at the website (https://blog.metu.edu.tr/mualtun/publication-sources/). The test projects were implemented in Matlab environment to construct the CPM schedule and calculate the completion time of each project which is given as a parameter in the constraints (Eq. (6)). The optimization problem was modeled in GAMS. The number of constraints in the model was too high to be entered manually. Therefore, considering the modeling format in GAMS, the model structure was constructed via Matlab code and conveyed to GAMS. All test cases are run by CPLEX MIP Solver 12.6 in GAMS on a PC with i7-3770 CPU, 8GB RAM memory, and Windows 10 (64 bit) operating system. All projects in each case are assumed to start at time 0. Moreover, each resource is equally weighted in the objective function where weight is taken as 250.

Performance of the proposed portfolio model is compared with the results of separately optimized projects according to same objective function. For a case study with M projects, M+1 models are constructed to generate three different results for the comparison. The first result is held by optimizing objective fitness value of the case portfolio with the proposed model. Secondly, summation of fitness values for separately optimized M projects gives the second fitness result as an upper bound for the portfolio optimization. Finally, it is assumed that although resources are leveled separately for each project, these projects can exchange resources if needed. Therefore, daily resource usage in the optimal solutions of the projects is summed to calculate daily portfolio resource needs for each resource. Hence, peak values of each daily portfolio resource usage amount are summed to determine third result.

3.1. Case example 1: Five 30-activity projects with two common resources

In the first case, efficiency of the model is analyzed with five different projects to investigate effect of number of projects on the portfolio performance. Therefore, projects were selected among small-sized ones as 30-activity projects whereas utilization of common resource types was limited with two for each project. The schedule network in each project is constructed using forty-eight precedence relationships. Projects in the case are completed in between 31 and 55 days (38, 42, 43, 55, 31 respectively) while number of activities on the critical path varies 7 to 9 for the projects. The optimization model is constructed with 150 Start/Finish decision variables, 2 resource variables, and 6270 binary variables.

The optimum solutions generated by integrated MPMRL model and MRL models of five different projects are presented in Table 1. The result shows that integrated model that levels resources of multiple project together generates significantly better solutions compared to separate evaluation of RLP for each project in total. The optimal solution of the MPMRLP model constructed via MIP used a maximum of 29 and 31 resources from different resource types respectively that costs 15000 in objective fitness. On the other hand, summation of peak resource requirements of the independently optimized projects is equal to 55 for resource type 1 and 64 for resource type 2 with a cost of 29000. Therefore, compared to the upper bound, the integrated model reduces the cost up to 14000, which provides 48.3% improvement in the objective fitness. Similarly, the results obtained using daily peak resource requirements of the portfolio according to optimum solution of separately optimized projects shows that 39 of resource type 1 and 47 of resource type 2 are required to complete projects successfully within cost of 21500. Compared to this result, the integrated solution improves the result within 30.2%.

3.2. *Case example 2: Two 300-activity projects with two common resources with tight schedule*

In the second case, performance of the model is tested with the projects that are performed in a short time horizon. Therefore, number of the projects in the case is decreased to 2 whereas number of the activities in each project is increased to 300. The number of common resource types is kept same. First project is completed in 44 days while the second one is performed in 41 days. In the first project, 5208 precedence relations exist whereas this value is 5240 in the second project; however, compared to the network size, the number of activities in the critical path is quite limited such as 6 activities in the first project and 5 activities in the second one. The optimization model is constructed with 600 Start/Finish decision variables, 2 resource variables, and 25500 binary variables.

The optimum solutions of the proposed model and separately analyzed single projects model are shown in Table 2. The results show that the proposed model slightly improves the performance of the summation of peak resource needs for independently optimized single projects.

Table 1. Optimization results of case example 1

Solution Approaches	Resource 1	Resource 2	Objective Fitness	Time Elapsed (sec)
MPMRL model				
Proposed solution	**29**	**31**	**15000**	**236.16**
MRL models for single projects				
Approach 1: Project peaks	55	64	29000	
Project 1	10	16	6500	0.13
Project 2	17	9	6500	0.17
Project 3	12	8	5000	0.19
Project 4	7	13	5000	0.20
Project 5	9	18	6750	0.25
Approach 2: Daily peaks	39	47	21500	

Table 2. Optimization results of case example 2

Solution Approaches	Resource 1	Resource 2	Objective Fitness	Time Elapsed (sec)
MPMRL model				
Proposed solution	**37**	**37**	**18500**	**36.56**
MRL models for single projects				
Approach 1: Project peaks	39	38	19250	
Project 1	19	17	9000	73.05
Project 2	20	21	10250	89.53
Approach 2: Daily peaks	39	38	19250	

On the other hand, due to higher parallelism in the project networks, using only the peak results of the independent projects and evaluating the daily resource requirements of separately optimized projects for portfolio peak resource needs reach to the same solution. Therefore, in both cases, the proposed model decreases resource needs per 2 of resource type 1 and 1 of resource type 2 within 3.9 % improvement which is quite smaller compared to the improvement rate for the first case. Therefore, even the integrated model finds optimum solution as is in this case, leveling the resources of numerous activities in a short project horizon increases the lower bound for the integrated solution and this results in smaller improvements in the solutions.

3.3. Case example 3: Two 300-activity projects with two common resources with broader schedule

In third case, how the serializability of the schedule impacts the solution of the constructed model is analyzed. Therefore, the projects having more serial networks compared to the ones in Case 2 were selected. Hence, number of activities and resource types in the projects are kept same while number of precedence relations is decreased and project horizon is elongated compared to the projects in Case 2. In this case, projects consist of 971 and 900 precedence relations whereas they are completed in 505 and 582 days, respectively. The optimization model is constructed with 600 Start/Finish decision variables, 2 resource variables, and 326100 binary variables. Therefore, the rate of number of activities in the critical path to total number of activities for the projects is increased up to around 30%, having 90 critical activities in the first project and 96 critical activities in the second one.

The optimum solutions of the portfolio and single projects are demonstrated in Table 3. The summation of peak resource needs of each project according to two resource types constraints the upper bound of integrated solutions with 44 of both resource types with a cost of 22000. On the other hand, daily peaks of the case portfolio according to independent leveling of the projects are 42 of resource type 1 and 41 of resource type 2 with a cost of 20750. The proposed solution decreased the resource peaks to 34 for resource type 1 and 28 for resource type 2 and that costs 15500. Therefore, compared to fitness values of projects peaks and daily portfolio peaks of the case, optimum solution of the integrated model improves the result with a rate of 29.5% and 25.5%, respectively. According to restricted comparison of improvement rate between the last two case examples, in the case portfolio whose projects have more serializable schedule network, the difference between upper and lower bound is quite higher than the other case. Therefore, the portfolios consisting of projects with serializable networks are open to a greater improvement rate for the integrated solution.

3.4. Case example 4: Two 300-activity projects with four common resources with broader schedule

In the final case, effects of number of resource types on the performance of the constructed model is examined.

Table 3. Optimization results of case example 3

Solution Approaches	Resource 1	Resource 2	Objective Fitness	Time Elapsed (sec)
MPMRL model				
Proposed solution	**34**	**28**	**15500**	**17.44**
MRL models for single projects				
Approach 1: Project peaks	44	44	22000	
Project 1	19	21	10000	11.94
Project 2	25	23	12000	107.16
Approach 2: Daily peaks	42	41	20750	

Therefore, same two projects in the third case were utilized to optimize resource levelling problem; however, two new resource types were added into optimization process The optimization model is constructed with 600 Start/Finish decision variables, 4 resource variables, and 326100 binary variables.

The optimization results shown in Table 4 indicates integrated solution improves the resources peaks in a considerable amount compared to the results of resource leveling of the projects independently. The integrated solutions reduce number of required resources in an amount of 47 and 33 for projects peaks and portfolio daily peaks, respectively, which results in 11750 and 8250 improvement in the objective fitness. In other words, the constructed model provides 27.6 % and 21.2 % savings, respectively. Moreover, additional resource types have impacts on the peak requirements of the resource types in the projects, as it can be seen in the resource type 2 of project 2. Due to the utilization of different resource types for the same activity, as the number of resources increases, tradeoff between the peaks of the resource types plays a prominent role to determine performing time of the activity in the non-critical path, and this may result in a change in the peak usage of the resource types.

In RLPs, analysis time of the optimization process depends on various factors such as number of activities and their precedence relations, number of resource types, floats of the activities, tradeoffs between peak requirements of resource types. Therefore, combination of various factors determines the required time to optimize the problem. For instance, in the first case example, although RLP was optimized in less than a second for each project, the integrated model of the same projects was solved in 236.16 seconds. In other words, although the number of constraints increased arithmetically, the analysis time showed exponential increase because of the tradeoffs between peaks of the two resource types for the projects with different project durations. On the other hand, for the second case, the integrated model reached the optimum solution in half the time required to solve RLP for one of the projects. In that case, tradeoff process is accelerated due to the evaluation of the multiple projects together where compelling tradeoff for the single projects' peaks is overcame with the ones of the multiple projects. The third and fourth cases show that solutions of the resource leveling problems require quite different time for projects generated with the same settings. Therefore, the precedence relations and their effect on the tradeoffs in the objective function have impacted the analysis time significantly. Furthermore, according to those cases, while the number of resource types correlated positively with the solution time of the integrated model, increasing the number of resource types decreased the analysis time of the problem for Project 1. Therefore, all of these case studies show that it is quite difficult to correlate the parameters/settings to the time required to optimize RLPs, because these parameters accelerate or decelerate the tradeoff process depending on the problem characteristics in projects.

Table 4. Optimization Results of Case Example 4

Solution Approaches	Resource 1	Resource 2	Resource 3	Resource 4	Objective Fitness	Time Elapsed (sec)
MPMRL model						
Proposed solution	**34**	**28**	**30**	**31**	**30750**	**93.70**
MRL models for single projects						
Approach 1:						
Project peaks	45	44	41	40	42500	
Project 1	19	21	25	19	21000	8.72
Project 2	26	23	16	21	21500	202.73
Approach 2:						
Daily peaks	45	40	33	38	39000	

4. Conclusion

In this study, an optimization model for the resource leveling problem of multiple resources in multiple projects is presented. The model is constructed using Mixed Integer Programming to minimize peak usage of resources by leveling the resources needed in different projects together. The performance of the constructed model was tested with four different case portfolios. The result shows that the model has superior performance for the case projects for finding the optimal solution for multi-project resource leveling. Moreover, the efficiency of the model was evaluated by comparing result of its optimal solution with two different results of optimum solutions for resource leveling problem optimized separately for each project; using the project specific peaks for resources and the peak resource requirements of daily resource requirements of the projects together. Apart from the case project with tight schedule, in all other three cases, the peak resource requirements of the case portfolio were reduced significantly. Therefore, these analyses reveal that the integrated model has a great potential to generate better solutions compared to the sub-optimum upper bound solutions of the individual optimization of each project.

In future studies, it is planned to test the effectiveness of the proposed solution approach with more complex project settings such as more number of projects, including more number of activities with higher number of resource types. In addition, heuristic approaches and hybrid models of heuristics with metaheuristics will be applied to mid-size and large-scale multi-project multi-resource leveling problems, to compare their performance in complex projects with the exact model constructed in this study in terms of time efficiency and optimal solutions.

Declaration of conflicting interests

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

Note

The preliminary version of this paper was presented to 5th International Project and Construction Management Conference (IPCMC2018) at Cyprus International University, Faculty of Engineering, Civil Engineering Department, North Cyprus.

References

[1] Hegazy T (1999). Optimization of resource allocation and leveling using genetic algorithms. Journal of Construction Engineering and Management 125:167–75.

[2] Ammar MA, Mohieldin YA (2002). Resource constrained project scheduling using simulation.

Construction Management and Economics 20:323–30.

[3] Leu S, Yang C, Huang J (2000). Resource leveling in construction by genetic algorithm-based optimization and its decision support system application. Automation in Construction 10:27–41.

[4] El-Rayes K, Jun DH (2009). Optimizing resource leveling in construction projects. Journal of Construction Engineering and Management 135:1172–80.

[5] Iranagh MA, Sonmez R (2012). A genetic algorithm for resource leveling of construction projects. 28th Annu. ARCOM Conf., vol. 2, p. 1047–1055.

[6] Easa SM (1989). Resource leveling in construction by optimization. Journal of Construction Engineering and Management 115:302–16.

[7] Neumann K, Zimmermann J (2000). Procedures for resource leveling and net present value problems in project scheduling with general temporal and resource constraints. European Journal of Operational Research 127:425–43.

[8] Rieck J, Zimmermann J, Gather T (2012). Mixed-integer linear programming for resource leveling problems. European Journal of Operational Research 221:27–37.

[9] Doğan B, Ölmez T (2015). A new metaheuristic for numerical function optimization: Vortex Search algorithm. Information Sciences 293:125–45.

[10] Burgess AR, Killebrew JB (1962). Variation in activity level on a cyclic arrow diagram. Journal of Industrial Engineering 13:76–83.

[11] Harris BRB (1990). Packing method for resource leveling (Pack). Journal of Construction Engineering and Management 116:331–50.

[12] Hiyassat MAS (2000). Modification of minimum moment approach in resource leveling. Journal of Construction Engineering and Management 126:278–84.

[13] Christodoulou SE, Michaelidou-Kamenou A, Ellinas G (2015). Heuristic methods for resource leveling problems. In: Schwindt C., Zimmermann J. (eds) Handbook on Project Management and Scheduling 1, pp. 389–407.

[14] Gogna A, Tayal A (2013). Metaheuristics: review and application. Journal of Experimental & Theoretical Artificial Intelligence 25:503–26.

[15] Wolpert DH, Macready WG (1997). No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation 1:67–82.

[16] Li H, Demeulemeester E (2016). A genetic algorithm for the robust resource leveling problem. Journal of Scheduling 19:43–60.

[17] Qi J, Wang Q, Guo X (2007). Improved particle swarm optimization for resource leveling problem. Proceedings of the Sixth International Conference on Machine Learning and Cybernetics, p. 896–901.

[18] Qiao J, Li Y (2018). Resource leveling using normalized entropy and relative entropy. Automation in Construction 87:263–72.

[19] Zhengjia W, Liping Z, Ying W, Kui W. Optimization for multi-resource allocation and leveling based on a self-adaptive ant colony algorithm. International Conference on Computational Intelligence and Security, 22 December 2008, Suzhou, China.

[20] Garmsiri M, Abassi MR (2012). Resource leveling scheduling by an ant colony-based model. Journal of Industrial Engineering International 8:1–9.

[21] Ponz-Tienda JL, Salcedo-Bernal A, Pellicer E, Benlloch-Marco J (2017). Improved Adaptive Harmony Search algorithm for the Resource Leveling Problem with minimal lags. Automation in Construction 77:82–92.

[22] Khanzadi M, Kaveh A, Alipour M, Aghmiuni HK (2016). Application of CBO and CSS for resource allocation and resource leveling problem. Iranian Journal of Science and Technology, Transactions of Civil Engineering 40:1–10.

[23] Cheng MY, Tran DH (2015). Opposition-based multiple objective differential evolution (OMODE) for optimizing work shift schedules. Automation in Construction 55:1–14.

[24] Altun M, Sonmez R, Akcamete A. An approach to resource balancing problem with priority differential evolution algorithm (ın Turkish). 4. Proje ve Yapım Yönetimi Kongresi, 3-5 November 2016, Eskisehir, Turkey.

[25] Wang H, Wang Z, Zhu M, Wang H, Wang Z (2017). Differential evolution algorithm for multi-project resource leveling problem. Journal of Interdisciplinary Mathematics 20:1383–1386.

[26] Guo Y, Li N, Ye T. Multiple resources leveling in multiple projects scheduling problem using particle swarm optimization. Fifth International Conference on Natural Computation, 14-16 August 2009, Tianjin, China.

[27] Cheng M-Y, Prayogo D, Tran D-H (2016). Optimizing multiple-resources leveling in multiple projects using discrete symbiotic organisms search.

Journal of Computing in Civil Engineering 30:04015036.

[28] Debels D, Vanhoucke M (2007). A decomposition-based genetic algorithm for the resource-constrained project-scheduling problem. Operations Research 55:457–69.