



PROJECT SCHEDULING AND STAFF ALLOCATION PROBLEM WITH TIME-DEPENDENT LEARNING EFFECT: A MIXED INTEGER NON-LINEAR PROGRAMMING APPROACH

Oğuzhan Ahmet ARIK^{1,*}

¹ Industrial Engineering Department, Engineering Faculty, Nuh Naci Yazgan University, Kayseri, Turkey

ABSTRACT

This paper proposes a mixed integer non-linear programming (MINLP) for project scheduling and staff allocation problem with time-dependent learning effect considering task similarities for learning. Resources of a project, especially staffs, can learn from previous tasks if those tasks and the current task are similar. Considering similarities among tasks for each staff can produce a more realistic schedule for a project. The objective function of the proposed problem is to minimize project completion time. The learning effect in this paper is dependent on the time when the project staff starts to do that task. If a project staff does repetitive and/or similar tasks then that staff can get experience and do tasks faster than planned durations. Therefore, task sequence is essential to assign tasks to project staffs in order to minimize project completion time by considering task similarities in view of learning.

Keywords: Time-dependent learning effect, Project scheduling, Mixed integer non-linear programming, Staff allocation

1. INTRODUCTION

Learning effect is a well-known phenomenon in scheduling problems. In project scheduling, learning effect or learning curve effect denotes a decrease in task durations by gaining experience due to repetitive tasks. Projects are complex and one-time endeavor. Furthermore, projects are limited by scarce resources such as working hours of qualified staffs. Utilization of these staffs is an essential part of project resource management. More realistic project schedule can be made by considering more realistic manpower planning. Any external or internal factor that directly effects task durations must be considered in project scheduling. Learning effect is one of these factors affecting task durations and efficiency of project staffs. This paper addresses time-dependent learning effect for project scheduling problems. Furthermore, this paper assumes that each task cannot be relevant to others in view of learning. If successive but irrelevant tasks are done by same resource or project staff then learning will not affect subsequent task and that staff cannot learn from the previous task. Therefore, this paper considers similarities of tasks with each other to present an efficient task sequence for project staffs and increase efficiencies of project staffs. Repetitive tasks directly influence the efficiency of the project staffs. Sometimes, some tasks are not repetitive but they are similar to each other. Any sequence including successive similar tasks may let project staff complete his/her tasks before initially planned duration.

The significance of project management has been getting more since the number of information technology (IT) projects increases. Results of IT projects are common interests of all departments in an organization. The most significant resource for IT projects is qualified project staffs such as developers, testers, and analyzers. Efficiencies and utilization of these qualified project staffs must be considered all internal and external factors such as learning effects.

*Corresponding Author: oaarik@nny.edu.tr

2. LITERATURE REVIEW

The most well-known learning curve was introduced by Wright [1] with the formula $Y = aX^b$ where Y is the cumulative average time (or cost) per unit, X is the cumulative number of units produced, a is time (or cost) required to produce the first unit and b is log of the learning rate/log of 2. An illustration for Wright's [1] learning curve can be seen in Figure 1.

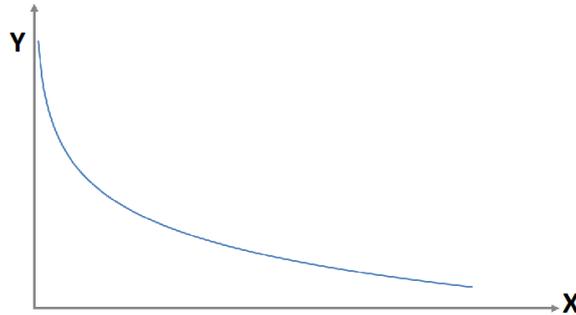


Figure 1. Well-known learning curve

The formula $Y = aX^b$ assures each of repetitive tasks are done by using less time than previous tasks. Biskup [2] was the first one who studied learning effect with job scheduling problems. He proposed a position-dependent learning effect for scheduling problem. He expressed actual processing time of a job is a function of the position that is used for assigned that job. Let P_r be basic processing time of the job or task assigned to position r , actual processing time $P_{[r]}$ under position-dependent learning effect can be calculated as follows:

$$P_{[r]} = P_r * r^a \quad (1)$$

where a is learning effect coefficient. Time-dependent learning effect was introduced by Kuo and Yang [3] and this effect denotes that a job is assumed to be a function of total normal processing time of jobs scheduled in front of it. Let P_r be basic processing time of the job or task assigned to position r , actual processing time $P_{[r]}$ under time-dependent learning effect can be calculated as follows:

$$P_{[r]} = P_r * \left(1 + \sum_{k=1}^{r-1} P_{[k]} \right)^a \quad (2)$$

where $P_{[k]}$ is actual processing time of any job or task assigned before position r . The readers can find more details about how to use position-dependent and time-dependent learning effects in scheduling problems in Arık and Toksarı's studies [4]–[6] and Toksarı and Arık's study [7].

Learning effect has been considered in project scheduling problems for repetitive tasks and even repetitive projects. Shtub et al. [8] considered a problem of scheduling in programs involving the production of multiple units of the same product. By considering learning effect and due dates of each unit, they investigated how many units of the product to assign to available resources such as individuals, teams, and subcontractors. They compared simulated annealing, genetic algorithm and a pairwise swap algorithm for their problem and they stated that the pairwise swap algorithm outperforms others. Ash and Smith [9] considered simultaneously effects of learning, forgetting and relearning for multi-project scheduling problems. In their paper, the main purpose was to determine the impact of learning, forgetting and relearning (LFR) because of the allowance for task preemption.

As a conclusion of their simulation experiments, they stated that LFR effects are significant. They tested multiple priority rules for creating a new schedule of resources, and activities and they stated best performing rules are based on activity criticality and resource learning. Arditi et al. [10] have proposed an approach for formulating learning ratios and incorporating them into line-of-balance (LOB) accounts. They designed an S-type fuzzy membership function to change the sequential learning rates until an adjusted learning rate is achieved. They then use this adjusted learning rate to calculate the expected worker-hours and activity durations in each production unit. Lam et al. [11] investigated learning and forgetting events in recurrent construction operations and their impact on project productivity. They integrated the LOB technique for learning-forgetting so that they can easily predict the loss of productivity caused by learning-forgetting. In another study of Arditi et al., [12] they investigated computerized LOB calculations, including learning curve, to develop an optimal strategy to shorten the project duration by increasing the production rate of selected activities, performing cost optimization and improving the visual presentation of LOB diagrams. Moselhi and Hassanein [13] presented a model that is designed to optimize either project construction duration, total cost, or their combined impact for what is known as cost-plus-time bidding. Their model consisted of some interesting features supporting project staffs to work simultaneously on any activity. Hassanein and Moselhi [14] proposed a model to integrate the planning and scheduling stages of highway construction projects. Their model includes resource availability; multiple previous and subsequent activities; transverse hitches; activities involving work in varying amounts along the length of the road; the effect of harsh weather conditions on crew productivity; and the beneficial effect of the learning curve. Wu and Sun [15] proposed a MINLP model for project scheduling and staff allocation with learning consideration problems where the objective is to minimize outsourcing costs. They proposed a genetic algorithm to solve their problem. Yan and Wang [16] proposed a staff scheduling methodology including a GA approach in order to solve personnel efficiency problem under the learning-forgetting effect. Ammar and Abdel Maged [17] modified the basic equation of LOB technique for learning effect to adopt a straight-line learning curve model because of its applicability and simplicity. Zhang et al. [18] developed an improved learning curve instead of the log-linear learning curve in order to monitor the improvement of performance of workers. Then, they modified LOB technique by integrating their new learning curve and resource allocation mechanism. Zha and Zhang [19] investigated a project scheduling problem under multi-skill learning effect. They proposed a new model to deal with the problem considering both autonomous and induced learning. For optimal solution, they proposed a GA. Mehmanchi and Shadrokh [20] proposed a mathematical programming model in order to handle a multi-skill project scheduling problem with their new exponential learning function that assumes the efficiency of employees performing activities are dynamic. Yang et al. [21] investigated uncertainty and ambiguity in the product development process because of iteration and overlapping. They proposed a simulation model including four variables to characterize uncertainty: iteration probability, iteration length, number of iterations and activity's learning curve effect. Zheng and Wang [22] proposed a multi-agent optimization algorithm for solving a resource-constrained project scheduling problem by indicating that the evolution of agents can be achieved with four main elements including social behavior, autonomous behavior, self-learning, and environment adjustment. Van Peteghem and Vanhoucke [23] studied a project scheduling problem with learning effect and applied the learning effect to the discrete time/resource trade-off scheduling problem. Qin et al. [24] developed a workforce scheduling model for assigning tasks to multi-skilled workforce by considering learning of knowledge and requirements of project quality. They proposed a MINLP model by using piecewise linearization to learning curve. Ammar and Abdel-Maged [25] incorporated learning development effect into traditional LOB scheduling. They modified the basic equation of LOB technique to consider learning development effect, in which straight-line learning curve model is adopted.

Chen et al. [26] investigated a multi-skill project scheduling problem for an IT product development project. They separated their project into sub small projects because of the complexity of original one. In their study, each task assumed to be done by a certain staff that has specific skills and these staffs' skill efficiencies are changeable because of learning and forgetting effects. Their proposed problem has three real life objective functions that are skill efficiency gain, project life cycle time and costs. In order to solve their problem, they proposed a MINLP model and a Non-dominated Sorted Genetic Algorithm II (NSGA-II). Guo et al. [27] improved a model describing the dynamic relationship between the learning/forgetting effect and the employees' skill levels in an IT project and they proposed a multi-objective software project optimization problem. They used an improved NSGA-II method for solution of the problem. Guo et al. [28] proposed multi-objective firework algorithm with a novel encoding scheme for the problem of Guo et al. [27] and they stated that their proposed method outperforms NSGA-II of Guo et al. [27]. Chen et al. [29] investigated repetitive scheduling projects and they proposed an algorithm to increase worker's efficiency under learning effect. They gave a real-life example of a concrete bridge construction project. Rzepecki and Biruk [30] investigated linear projects such as highways, multi-store buildings or pipelines where some tasks are repeatedly implemented. Rzepecki and Biruk [30] developed a simulation method for scheduling linear construction projects under learning and forgetting effects. Hosseinian et al. [31] proposed a new mixed-integer formulation for the time-dependent multi-skilled resource-constrained project scheduling problem considering learning effect. They integrated the concept of diffusion maximization in social networks into their proposed model in order to build a network of workforces who share common skills. Hosseinian et al. [31] proposed an improved NSGA-II with new genetic operators that help to spread dexterity among workforces. For the readers who want to investigate the literature about learning effect and project scheduling problems in more details, the paper of Hosseinian et al. [31] can be a guide as the latest paper about the topic as far as we know.

When the literature for project scheduling problems are investigated, most of the papers [25]–[30] have used the phenome of Wright [1] for learning effects in project scheduling. Wright's [1] formula is to show the effects of the learning by doing repeating a single task in large amounts. When two or more tasks are done successively and there are similarities among these tasks in view of learning, a new way to calculated effects of the learning is needed. Therefore, we suggest using a time-dependent learning effect [3] that is similar to the phenome of Wright [1] and it is also different approach considering discrete and similar tasks done by same staffs. Furthermore, the proposed learning effect calculation method considers similarities of tasks in view of learning and how much of a task will be done by a certain project member. As far as we known, this study is the first paper that investigates the time-dependent learning effect [3] and tasks similarities in view of learning.

3. MIXED INTEGER NON-LINEAR PROGRAMMING MODEL

In this section, a MINLP model is introduced for a project scheduling problem where the objective is to minimize project duration when the resources of that project learn from previous similar tasks. A resource can learn from previous similar tasks and the sequence of tasks for that resource is so significant in order to decrease project initially planned duration considering learning relationship among tasks. There are some assumptions for the following model. These are; preemption for tasks is not allowed, all resources are ready at the beginning, and a resource can do only a task at the same time. The notations, parameters, decision variables, objective function and constraints for proposed MINLP are as follows:

i: index for tasks

r: index for resources (project staffs)

p : index for position numbers in each resources

t_i : duration of task i

$Prec_{i,j} \in \{0,1\} \forall i, j$: Precedence relationship between task i and j ($i \neq j$). If task i is a predecessor of task j , then $Prec_{i,j} = 1$, otherwise it is zero.

$SC_{i,j}$: Similarity coefficient between task i and j . If task i is related with task j in view of learning, then $SC_{i,j} = 1$, otherwise it is zero.

$Sm_{i,r} \in \{0,1\} \forall i, r$: If task i must be done by resource r then $Sm_{i,r} = 1$, otherwise it is zero. Each project task has to be executed by a resource at least. However, a task may be needed to be executed by multiple resources than $\sum_r^R Sm_{i,r}$ may be greater than 1.

$L_{i,r}$: Loading coefficient of t_i for resource r .

α_r : Time-dependent learning effect coefficient for resource r .

C_{max} : Project duration (makespan).

$C_{i,r}$: Completion time of resource r for task i .

$S_{i,r}$: Starting time of resource r for task i .

$P_{[r],p}$: Actual task duration of the task assigned at position p of resource r under time-dependent learning effect.

$C_{[r],p}$: Completion time of the task assigned at position p of resource r .

$S_{[r],p}$: Starting time of the task assigned at position p of resource r .

$X_{i,r,p} \in \{0,1\} \forall i, r, p$: if task i is assigned at position p of resource r , then $X_{i,r,p} = 1$, else it is zero.

The objective function (3) of proposed MINLP is to minimize project duration C_{max} or longest task completion time of any task by any resources. C_{max} is also the duration of the critical path of the project.

$$Min z = C_{max} \quad (3)$$

Constraint (4) assures that project duration C_{max} must be greater than the longest project task completion time by any resources.

$$C_{max} \geq C_{[r],p} \quad \forall r, p \quad (4)$$

Constraint (5) shows that completion time of the task assigned to position p of resource r equals to the sum of its starting time and actual task duration under time-dependent learning effect.

$$C_{[r],p} = S_{[r],p} + P_{[r],p} \quad \forall r, p \quad (5)$$

Constraint (6) assures that actual task duration of the task assigned to position p of resource r is dependent on the previous tasks which are similar to the current task in view of learning.

$$P_{[r],p} = \sum_{i=1}^n \left[(X_{i,r,p} * t_i * L_{i,r}) * \left(1 + \sum_{k=1}^{p-1} \sum_{\substack{j=1 \\ j \neq i}}^n X_{i,r,p} * Sc_{i,j} * P_{[r],k} \right)^{\alpha_r} \right] \forall r, p \quad (6)$$

Constraint (7) shows that starting time of the current position must be greater than or equal to the previous position's completion time.

$$S_{[r],p} \geq C_{[r],p-1} \forall r \text{ and } p = 2,3,4,\dots,\bar{p} \quad (7)$$

\bar{p} is maximum number of positions in each sequence for each resource and $\bar{p} = \max(\sum_{i=1}^n Sm_{i,r} \forall r)$. Constraint (8) assures that if a task must be executed by any resource r , then that task must be assigned to only one position of that resource.

$$\sum_{p=1}^{\bar{p}} X_{i,r,p} = Sm_{i,r} \forall i, r \quad (8)$$

Constraint (9) shows that any position of any resource can be used for only one task at most.

$$\sum_{i=1}^n X_{i,r,p} \leq 1 \forall r, p \quad (9)$$

Constraints (10) and (11) are to determine each task's completion and starting time in each resource.

$$C_{i,r} = \sum_{p=1}^{\bar{p}} X_{i,r,p} * C_{[r],p} \forall i, r \quad (10)$$

$$S_{i,r} = \sum_{p=1}^{\bar{p}} X_{i,r,p} * S_{[r],p} \forall i, r \quad (11)$$

Constraint (12) assures that if task i is a predecessor of task j ($j \neq i$), then starting time of task j by any resource r must be greater than or equal to then completion time of task i by any resource k ($k \neq r$). In order to have computational fitness of this constraint to the model, some conditions as follows:

$$\begin{aligned} S_{j,r} &\geq Prec_{i,j} * C_{i,k} \forall i, j, r, k \text{ where } i \neq j, k \neq r, \\ Prec_{i,j} &= 1, Sm_{j,r} = 1, Sm_{i,k} = 1 \end{aligned} \quad (12)$$

Constraints (13-16) define the necessary domains of decision variables as follows:

$$X_{i,r,p} \in \{0,1\} \forall i, r, p \quad (13)$$

$$S_{i,r}, C_{i,r} \geq 0 \forall i, r \quad (14)$$

$$S_{[r],p}, C_{[r],p}, P_{[r],p} \geq 0 \forall r, p \quad (15)$$

$$C_{max} \geq 0 \quad (16)$$

Completion times of same tasks done by different resources can be different in this proposed model. However, the decision maker can make them equal to each other by using constraint (17), if necessary.

$$C_{i,r} = C_{i,k} \forall i, r, k \text{ where } k \neq r, Sm_{i,r} = 1, Sm_{i,r1} = 1 \quad (17)$$

4. NUMERICAL EXAMPLE

In this section of the paper, a numerical example for the proposed mathematical model is presented. Let's consider a software project that has 10 tasks must be done by three different resources (a developer, a tester, and an analyzer). These project resources prefer some tasks do earlier or later so

that they can be faster than planned task durations because of learning. Therefore, the sequence of tasks done by the same resource is significant to have a faster project completion. For instance, the tester can arrange the sequence of his/her tasks by considering precedence relations among his/her tasks and other tasks and learning similarity of those tasks in order to shorten initially planned project duration. The precedence relations, task durations, and resource requirements are given in Table 1. Figure 2 shows also the activity-on-node diagram of the numerical example. Similarity coefficients of tasks are given in Table 2. Loading coefficients of task durations to resources are given in Table 3. Resources in Table 1 are coded as R1, R2, and R3 for the developer, the tester, and the analyzer, respectively. Learning effect coefficients of resources are -0.08, -0.1 and -0.12 respectively.

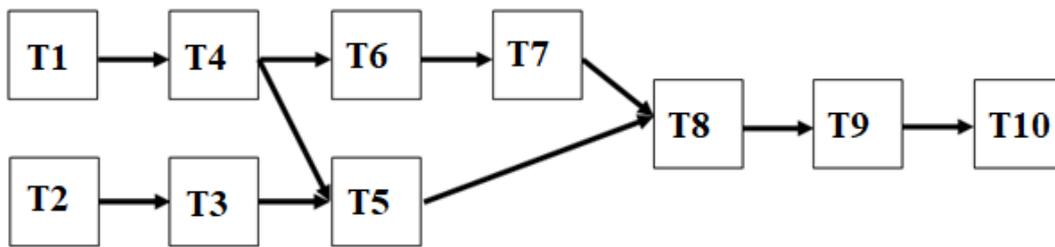


Figure 2. Activity-on-Arrow diagram of the numerical example

Table 1. Data for numerical example

Task Code	Predecessors	Duration (weeks)	Resources
T1	-	5	R1-R2
T2	-	6	R2-R3
T3	T2	3	R2-R3
T4	T1	13	R1-R3
T5	T3-T4	9	R3
T6	T4	2	R2
T7	T6	11	R1-R2
T8	T7-T5	9	R2
T9	T8	7	R2-R3
T10	T9	2	R3

Table 2. Similarities of tasks in view of learning

<i>i/j</i>	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
T1										
T2										
T3		✓								
T4	✓									
T5			✓	✓						
T6				✓						
T7						✓				
T8					✓		✓			
T9								✓		
T10									✓	

Table 3. Loading coefficients of task durations to resources

<i>i/r</i>	R1	R2	R3
T1	1	0.7	0
T2	0	1	0.8
T3	0	0.2	1
T4	1	0	0.4
T5	0	0	1
T6	0	1	0
T7	1	0.8	0
T8	0	1	0
T9	0	0.6	1
T10	0	0	1

This numerical example is solved with GAMS 21.6 software with DICOPT solver. Figure 3 shows the schedule of the model without constraint (17). The project completion time of the schedule in Fig. 1 is 45.056 weeks. If the model considered without learning effect, constraint (6) is changed as follows:

$$P_{[r],p} = \sum_{i=1}^n [(X_{i,r,p} * t_i * L_{i,r})] \quad \forall r,p \tag{18}$$

If we solve the problem by replacing constraint (6) with constraint (18), then the completion time of the project increases to 49 weeks. Considering learning effect among similar tasks in view of learning decreases project duration almost 4 weeks without changing the task sequence and task-project member pairs. Therefore a similar schedule as in Figure 3 can be obtained with additional 4-week duration for the same numerical example without learning effect.

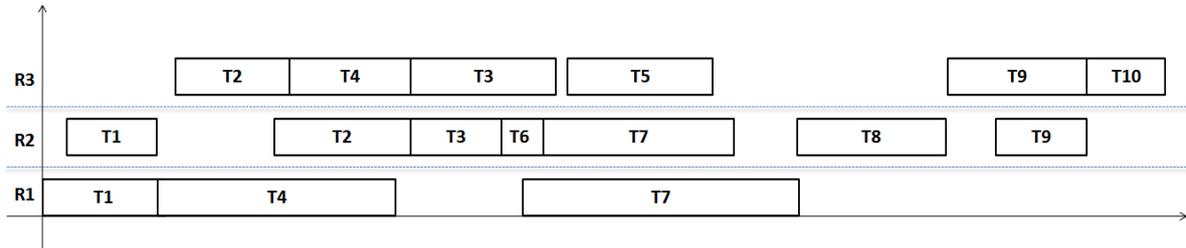


Figure 3. The schedule of the numerical example by not considering the constraint (17)

If the decision maker wants to complete the same task by different resources at the same time, the constraint (17) has to be considered in the model. The schedule in Figure 4 has the same project completion time with the model without the constraint (17).

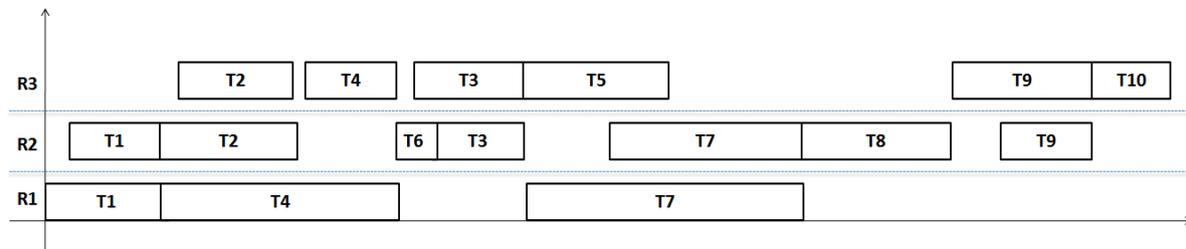


Figure 4. The schedule of the numerical example by considering the constraint (17)

As seen in Figure 4, a task is completed by different resources at the same time when the constraint (17) is considered in the model. Both of the schedules have same project completion time but the second model's execution time is more than the first one. The problems with or without constraint (17)

may not always have the same completion time. For instance, the problem is solved with constraint (17) by replacing constraint (6) with constraint (18), then the completion time of the project increases to 53.5 weeks. In case of using constraint (17), considering learning effect among similar tasks in view of learning decreases project duration more than 8 weeks without changing the task sequence and task-project member pairs. Therefore a similar schedule as in Figure 4 can be obtained with additional 8-week duration for the same numerical example without learning effect.

5. PERFORMANCE OF PROPOSED MINLP MODEL

In previous section, a small numerical example with ten tasks and three project staffs is illustrated for illustrating the proposed MINLP model. As seen from previous section, considering learning effect and similarities of tasks in view of learning shorten the project duration. Thus, we may say the proposed model has contribution to the project duration for a decision maker. Small problems like the numerical example in previous section can be solved in bearable and short times. Therefore, large instances are needed to show the contribution of the proposed model considering the amount of decrease in the project duration and time requirement for obtaining a solution. In order to validate effectiveness of proposed model, we used some benchmark problems of Alvarez and Tamarit [32]. We selected ST27-1, ST51-1 and ST103-1 problems as benchmark problems. There are 27 tasks and 6 resources in ST27-1 problem. There are 51 tasks and 6 resources in ST51-1 problem. There are 103 tasks and 6 resources in ST103-1 problem. These instances do not give information for similarities among tasks in view of learning and there is no additional information about loading coefficients like in Table 3. Therefore learning effect coefficients of resources, loading coefficients and similarities among tasks in instances are generated arbitrarily. Task durations of the instances are calculated by taking sum of all resources' task durations. Table 4 shows solutions of classical approach (without learning effect) and the proposed approach by not using constraint (17).

Table 4. Comparison of solution approaches

Problem	Solution of Proposed Method	Solution Time of Proposed Method	Solution of Classical Method	Solution Time of Classical Method
ST27-1	305.08 time units	97.273 seconds	455.60 time units	18.918 seconds
ST51-1	491.308 time units	617.273 seconds	720.00 time units	25.795 seconds
ST103-1	1150295.467 time units	142.674 seconds	1244099.691 time units	42.067 seconds

Solutions in Table 4 were obtained by relaxing the models and using GAMS 21.6 software with DICOPT solver. Results of the proposed method that considers learning effects and similarities among tasks in view of learning are much better than the results of classical method that does not consider the learning effect. Although the solutions were obtained via relaxing the problems, time differences between solution approaches directly indicate that if the decision-maker who wants to take learning effects account into consideration in any large project, then he/she should use a valid and strong metaheuristic method for obtaining a near-optimal solution in bearable times.

6. CONCLUSION AND FUTURE RESEARCHES

In this paper, time-dependent learning effect considering task similarities for learning is investigated for resources (project staffs) to produce more realistic project scheduling and staff allocation. The objective function of the proposed problem is to minimize project completion time. The learning effect in this paper is dependent on the time when the project staff starts to do that task. If a project staff does repetitive and/or similar tasks then that staff can get experience and do tasks faster than planned durations. In this paper, a MINLP model and a numerical example are presented for the problem. Furthermore, some benchmark problems in the literature are used to validate the effectiveness of the proposed model. The proposed model can be used in linear projects such as highways, multi-store

buildings or pipelines where some tasks and/or similar tasks are repeatedly implemented. As a real-life example, some tasks of construction projects such as the pouring of concrete on floors can be similar with other tasks so the learning phenomenon can be considered. Future researchers can investigate time-dependent learning effect with forgetting effect for project scheduling and staff allocation problem. Furthermore, the complex structure of the proposed model can be handled with metaheuristics or heuristic methods for large size instances. We also suggest future scholars to investigate this proposed learning effect considering task similarities for other scheduling problems such as flow shop or job shop scheduling problems.

REFERENCES

- [1] Wright TP. Factors Affecting the Cost of Airplanes. *J Aeronaut Sci* 1936; 3: 122–128.
- [2] Biskup D. Single-machine scheduling with learning considerations. *Eur J Oper Res* 1999; 115: 173–178.
- [3] Kuo WH, Yang DL. Minimizing the total completion time in a single-machine scheduling problem with a time-dependent learning effect. *Eur J Oper Res* 2006; 174: 1184–1190.
- [4] Arik OA, Toksarı MD. Multi-objective fuzzy parallel machine scheduling problems under fuzzy job deterioration and learning effects. *Int J Prod Res* 2018; 56: 2488–2505.
- [5] Arik OA, Toksarı MD. Fuzzy Parallel Machine Scheduling Problem Under Fuzzy Job Deterioration and Learning Effects With Fuzzy Processing Times in *Advanced Fuzzy Logic Approaches in Engineering Science*, M. Ram, Ed. IGI Global, 2019, pp. 49–67.
- [6] Arik OA, Toksarı MD. Fuzzy chance constrained programming technique for single machine earliness/tardiness scheduling problem under effects of fuzzy learning and deterioration. *Sak. Univ J Sci* 2018; 22(2): 652–662.
- [7] Toksarı MD, Arik OA. Single machine scheduling problems under position-dependent fuzzy learning effect with fuzzy processing times. *J Manuf Syst*, 2017; 45:159-179.
- [8] Shtub A, LeBlanc LJ, Cai Z. Scheduling programs with repetitive projects: A comparison of a simulated annealing, a genetic and a pair-wise swap algorithm. *Eur J Oper Res* 1996; 88: 124–138.
- [9] Ash R, Smith-Daniels DE. The effects of learning, forgetting, and relearning on decision rule performance in multi project scheduling. *Decis Sci* 1999; 30: 47–79.
- [10] Arditi D, Tokdemir OB, Suh K. Effect of learning on line-of-balance scheduling. *Int J Proj Manag* 2001; 19: 265–277.
- [11] Lam KC, Lee D, Hu T. Understanding the effect of the learning-forgetting phenomenon to duration of projects construction. *Int J Proj Manag* 2001; 19: 411–420.
- [12] Arditi D, Tokdemir OB, Suh K. Challenges in line-of-balance scheduling. *J Constr Eng Manag* 2002; 128: 545–556.
- [13] Moselhi O, Hassanein A. Optimized scheduling of linear projects, *J Constr Eng Manag*, 2003; 129: 664–673.

- [14] Hassanein A, Moselhi O. Planning and scheduling highway construction. *J Constr Eng Manag* 2004; 130: 638–646.
- [15] Wu MC, Sun SH. A project scheduling and staff assignment model considering learning effect. *Int J Adv Manuf Technol* 2006; 28: 1190–1195.
- [16] Yan JH, Wang ZM. GA based algorithm for staff scheduling considering learning-forgetting effect, in: 2011 IEEE 18th Int Conf Ind Eng Eng Manag IE EM 2011, 2011: 122–126.
- [17] Ammar M.A, Abdel-Maged AF, Scheduling of repetitive projects with learning development effect, in: *Adv Civ Eng Build Mater - Sel. Peer Rev Pap from 2012 2nd Int Conf Civ Eng Build Mater CEBM 2012*, 2012: pp. 341–346.
- [18] Zhang L, Zou X, Kan Z. Improved strategy for resource allocation in repetitive projects considering the learning effect. *J Constr Eng Manag* 2014; 140.
- [19] Zha H, Zhang L. Scheduling projects with multiskill learning effect. *Sci World J* 2014.
- [20] Mehmanchi E, Shadrokh S. Solving a new mixed integer non-linear programming model of the multi-skilled project scheduling problem considering learning and forgetting effect on the employee efficiency. in: *IEEE Int Conf Ind Eng Eng Manag 2014*: pp. 400–404.
- [21] Yang Q, Lu T, Yao T, Zhang B. The impact of uncertainty and ambiguity related to iteration and overlapping on schedule of product development projects. *Int J Proj Manag* 2014; 32: 827–837.
- [22] Zheng XL, Wang L. A multi-agent optimization algorithm for resource constrained project scheduling problem. *Expert Syst Appl* 2015; 42: 6039–6049.
- [23] Van Peteghem V, Vanhoucke M. Influence of learning in resource-constrained project scheduling. *Comput Ind Eng* 2015; 87: 569–579.
- [24] Qin S, Liu S, Kuang H. Piecewise Linear Model for Multiskilled Workforce Scheduling Problems considering Learning Effect and Project Quality. *Math Probl Eng* 2016.
- [25] Ammar MA, Abdel-Maged AF. Modeling of LOB scheduling with learning development effect. *Int J Constr Manag* 2018; 1–10.
- [26] Chen R, Liang C, Gu D and Leung J Y-T. A multi-objective model for multi-project scheduling and multi-skilled staff assignment for IT product development considering competency evolution. *Int. J. Prod. Res.* 2017; 55(21): 6207–6234.
- [27] Guo Y-N, Ji J-H, Ji J-J and Gong D.-W. Multi-objective software project scheduling optimization method with the learning and forgetting effect. *Kongzhi yu Juece/Control Decis.* 2018; 33(2): 203–210.
- [28] Guo Y, Ji, J, Ji, J, Gong D, Cheng J and Shen X., Firework-based software project scheduling method considering the learning and forgetting effect. *Soft Computing.* 2018; 1–16.
- [29] Chen H, Wang K, Du Y and Wang L. The algorithm for duration acceleration of repetitive projects considering the learning effect in 2018 2nd International Workshop on Advances in Energy Science and Environment Engineering, AESEE. 2018.

- [30] Rzepecki L, Biruk S. Simulation Method for Scheduling Linear Construction Projects Using the Learning-Forgetting Effect in 2nd Baltic Conference for Students and Young Researchers, BalCon. 2018.
- [31] Hosseinian AH, Baradaran V and Bashiri M. Modeling of the time-dependent multi-skilled RCPSP considering learning effect: An evolutionary solution approach. *Journal of Modelling in Management* 2019.
- [32] PROJECT SCHEDULING PROBLEM LIBRARY, “PROJECT SCHEDULING PROBLEM LIBRARY” 2019. [Online]. Available: <http://www.om-db.wi.tum.de/psplib/dataob.html>. [Accessed: 08-Mar-2019].