Scientia Iranica E (2023) 30(4), 1423–1434

SCIENTIA IRANICA

Sharif University of Technology
Scientia Iranica
Transactions E: Industrial Engineering
http://scientiairanica.sharif.edu

A combinatorial optimization solution for activity prioritizing problem

M.A. Hatefi* **and S.A. Razavi**

Department of Energy Economics and Management, Petroleum University of Technology (PUT), Tehran, Iran.

Received 8 December 2019; received in revised form 16 March 2021; accepted 17 May 2021

KEYWORDS
Activity prioritizing problem;
Project scheduling;
Mathematical programming;
Row generation;
Combinatorial optimization;
Branch-and-cut;
Oil and gas industry.

Abstract. This paper discusses a particular situation in project management in which an analyst attempts to prioritize several independent activities to handle all of them one by one in such a way that there would be no precedence relationships over the activities. The novelty of this research is that the structure of prioritized activities is linear in arrangement which can be considered as a combinatorial optimization problem. The paper formulates a mathematical model and applies it to two real cases in the oil and gas industry. In addition, a row generation procedure is developed to solve large-scale problems and the computational results for the problem instances of size up to 300 activities are reported. The results demonstrate the applicability and efficiency of the proposed methodology.

1. Introduction

As Gerardi [1] states in his book, project managers generally list several explanations as to why they love project management: 'they get to learn new things', 'work with new people', 'gain new skills', and 'are on the leading edge of innovation'. In line with this view, this paper concentrates on an interesting challenge in project planning named Activity Prioritizing Problem (APP), which is a branch of project scheduling problem. Project scheduling has attracted considerable attention because of its critical role in project resource management [2,3]. Consider a situation in which there are project activities with no precedence relationships over them. For instance, after scheduling a lot of project activities, a subset of them (with no inter-related dependencies) should be done in a limited period of time. In this situation, the problem is simply to assign priorities to such activities. It is clear that this challenge is not affected by duration of activities. Instead, several factors likely affect the priorities such as project' owner view, program and project manager' ideas, risk factors [4], etc. In this regard, the role of a project manager and project experts has been addressed as fundamental dimensions for the project success [5], which appears to be neglected in the literature [6,7]. On the other hand, Greek and Pullin [8] argued that many project managers did not focus on the criticality and urgency of project activities. However, a major concern in project management is the determination of a priority ranking for all activities that are candidates for assignment [9].

With the APP, there are many descriptive solutions and a few analytical models in the relevant literature. Among various descriptive guidelines (6-step process, Eisenhower matrix, Brian Tracy's method,

*. *Corresponding author. Tel.: +98 21 44222483*
E-mail addresses: Hatefi@put.ac.ir (M.A. Hatefi); Srazavi@put.ac.ir (S.A. Razavi)

doi: 10.24200/sci.2021.54883.3962

ABCDE method, Bubble sort technique, Moscow method, Pareto principle, Zen habit, Rag rating system, Scrum prioritization, 1-3-9 method, and so on), we will have minor explanation on only two methods. According to the Zen habit method [10], as a really simple prioritization guideline, the manager writes down one to three most important activities each time to be completed in a limited period of time and gets on with completing them. The Eisenhower matrix, popularized by Covey [11], works by dividing segregate activities into four quadrants based on urgency and importance: 'urgent and important', 'important but not urgent', 'urgent but not important', and 'neither urgent nor important'.

Despite the importance of weighting and prioritizing project activities, analytical models have been rarely addressed in the related literature [7,12]. In addition, most of the analytical studies use Multiple Attribute Decision Making (MADM) techniques to get a rank or weight vector of activities. Chang and Ibbs [9] introduced an expert system that uses possibility theory and the generalized modus ponens logic inference rule to prioritize activities in building construction projects. Alencar et al. [13] employed ELECTRE III technique to prioritize activities in a real-world construction project. Jha and Misra [14] conducted a survey to identify major factors in classifying and ranking construction coordination activities. An analytical effort was made by Mota et al. [15] to present a model for supporting project managers to focus on the main activities of a project network using a MADM technique, taking into account several, often contradictory, points of view. The mentioned model aims to assign project activities into three classes of managerial practice. In an article by Baykasoglu et al. [12], a practical fuzzy rating and ranking technique was proposed to prioritize project activities with fuzzy attributes. They considered prioritization of activities instead of their classification. In addition, their technique was in some aspects similar to that of Mota et al. [15], with substantial differences. Vanhoucke [16] introduced a scheme to measure sensitivity to the activities and network topological information to monitor the project time performance. Mota and de Almeida [17] proposed an ELECTRE-based model to help project managers make decisions about the key activities of a project. Golpira [18] made a comparison between conventional MADM and fuzzy MADM to determine weights and priorities of activities using Analytical Hierarchy Process (AHP) and Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) methods. Hadad et al. [19] prioritized the attributes of project activities using an MADM model. These attributes included activities' expected time, standard time deviation, expected cost, standard cost deviation, and the time and cost obtained through simulation. Hadad [20] proposed a technique to prioritize project activities using AHP and Data Envelopment Analysis (DEA) as research tools to select proper influencing factors. Kalayathankal et al. [21] applied a fuzzy technique to prioritize activities in a software development project with analogous activities in the electrical substation construction studied by Mota et al. [15]. They achieved better project output in terms of finance, personnel, equipment, space, time, etc. Berjis et al. [7] proposed an approach to determine the weights of project activities using DEA. They identified the parameters affecting the importance of activities through a review of the related literature and experts' opinions. The parameters included activity duration, activity cost, activity importance, activity difficulty, safety, communication rate, intellectual effort, physical effort, unfavourable work conditions, and work-related hazards.

In the current article, a novel optimization solution is proposed to deal with a particular situation of the APP. The idea in the proposed approach is completely different from previous works. The organization of the paper is as follows. Section 2 points out the problem characterization and related definitions. Section 3 develops a mathematical model to solve the problem. Section 4 includes two real cases studied from oil and gas industry. Section 5 discusses how to deal with large-scale problems followed by experimental analysis in Section 6. Finally, Section 7 summarizes the results achieved and provides future efforts and possibilities.

2. Characterization of the problem

Herein, a special situation of the APP is introduced and it is an interesting issue that the authors encountered in real projects. Indeed, real cases (described in Section 4) were the root of the idea of the problem discussed in this paper. As a result, the need for the proposed model (described in Section 3) originates from real-world projects and it, thus, can be applied to actual challenges.

Suppose a situation in which there are a variety of project activities and the aim is to prioritize activities to handle all of them one after another. Indeed, all activities need a unique resource available only for one activity at any point of time. Taking this fact into account, we seek to sequentially perform activities from the most important to the least important. Thus, an arrangement structure of activities needs to be formed. That special resource may be a unique montage platform, a unique consultant, a unique working front, etc. The solution is influenced by some different "adjacency factors". An adjacency factor indicates a reason according to which some activities need to be assigned locations in the arrangement structure close to each other. Obviously, any linear permutation including all activities is a valid solution to the problem. Thus, the idea of the current paper arises from this point of view and it is a linear arrangement in the prioritizing challenge explained above; therefore, we have a combinatorial optimization problem in which activities should be arranged from the most important to the least important.

Let us define the problem in a general form. The problem has two groups of elements: "activities" and "adjacency factors" (henceforth referred to as factors). There are m activities $T_1, T_2, \ldots, T_m$ to be prioritized and n factors having positive weights $\gamma_1, \gamma_2, \ldots, \gamma_n$. In view of an assumed factor, the relevant activities should be placed one after another in the overall arrangement structure. The activities respecting the jth factor are shown by subset $f_j \subset \{T_1, \ldots, T_m\}$. An assumed factor may be proposed by any of o experts ($k = 1, 2, \ldots, o$). In addition, each expert may suggest various factors. The factors are actually derived from similarity of the complexity of activities, relationships among activities, equality in urgency of activities, etc. The weight of a given factor is calculated via Eq. (1), which is a weighted geometric mean stressed by many authors to aggregate the individual weights [22–25]. In this equation, w_{jk} is the weight of the jth factor preferred by the kth expert. This individual weight is a number between 1 and 10. If an expert has no idea about a factor, the respected items should be dropped out from Eq. (1). The power u_k, a number between 0 and 1, indicates the importance of the kth expert's comments. This power depends on the expert's skills, experiences, abilities, or his/her area of educations:

$$\gamma_j = \left(\prod\nolimits_{k=1}^{o} {w_{jk}}^{u_k}\right)^{1/\sum_{k=1}^{o} u_k}, \qquad j = 1, \ldots, m. \qquad (1)$$

In view of the jth factor, the best case is comprised of a contiguous chain of all the activities in f_j. In this case, complete weight γ_j will be added to the objective value of the mathematical model; otherwise, a proportion of γ_j would be met. By taking this into consideration, the problem is to arrange activities so as to maximize the sum of proportions of factors' weights. The value of each proportion depends on the adjacency of relevant activities in the overall arrangement structure.

Truly, the introduced challenge falls into the combinatorial optimization problem. Although a wide range of models and algorithms exist to deal with famous graph-theoretic and combinatorial optimization problems, the introduced issue is not like any well-known problems. In particular, it is completely different from cell formation problem [26,27], job-shop scheduling problem [28], project scheduling problem [2], single-row facility layout problem [29], linear ordering problem [30], minimum linear arrangement problem [31,32], adjacency problem [33], MADM problem [22,34], and linear assignment problem [35].

3. Modeling the problem

First, let us define the following elements:

- T_i refers to the ith activity ($i = 1, \ldots, m$);
- f_j stands for the set of activities respecting the jth factor ($j = 1, \ldots, n$);
- γ_j indicates the weight of the jth factor ($j = 1, \ldots, n$);
- θ_{pq} ($p < q$) shows that a binary variable equals one if activity T_p is placed exactly close to activity T_q;
- $s()$ shows size of a set.

Now, the formulation of the problem is written as in Model (2)–(5):

$$\max \sum_{j=1}^{n} \left(\frac{\sum_{T_p \& T_q \in f_j | p<q} \theta_{pq}}{s(f_j) - 1} \right) \gamma_j, \qquad (2)$$

s.t.:

$$\sum\nolimits_{q=1,\ldots,m|i<q} \theta_{iq} + \sum\nolimits_{p=1,\ldots,m|p<i} \theta_{pi} \leq 2, \quad i = 1, \ldots, m, \qquad (3)$$

$$\sum\nolimits_{T_p \& T_q \in B|p<q} \theta_{pq} \leq s(B) - 1, \quad B\{T_1, \ldots, T_m\}, 2 < s(B), \qquad (4)$$

$$\forall \theta_{pq} \in (0, 1). \qquad (5)$$

To ensure a better understanding, an explicit formulation for a small problem with the corresponding LINGO code was provided in Appendix A.

The rule to formulate Function (2) is that "If two activities of f_j are placed right beside each other in the overall string of activities, then "1" is added to objective function coefficient of γ_j". Let us consider some cases. For instance, suppose that there are 4 activities and 3 factors and we want to determine the objective function coefficients of γ_1, γ_2, and γ_3 for the given string $T_4\ T_1\ T_3\ T_2$. The set of activities respecting the factors includes $f_1 = \{T_1, T_3, T_4\}$, $f_2 = \{T_2, T_4\}$, and $f_3 = \{T_1, T_2, T_4\}$. For a given factor f_j, in the best case in which each activity f_j has at least one adjacent activity from f_j, $\sum_{T_p \& T_q \in f_j | p<q} \theta_{pq}$ equals $s(f_j) - 1$; as a result, the objective function coefficient of γ_j equals 1. For example, in the above instance, considering f_1, T_4 is located next to T_1 and T_1 is adjacent to T_3; thus, $\sum_{T_p \& T_q \in f_1 | p<q} \theta_{pq} = 2$ and coefficient of γ_1 equals 1. In the worst case, none of activities of f_j is placed right next to another activity of f_j; consequently, $\sum_{T_p \& T_q \in f_j | p<q} \theta_{pq} = 0$. Thus, the objective function

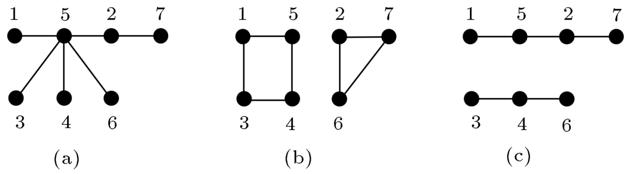

Figure 1. Infeasible patterns containing (a) unacceptable tree, (b) cycle, and (c) non-connected tree.

coefficient for γ_j will be equal to zero. For example, considering the second factor in the above instance, T_2 is not located next to T_4; thus, $\sum_{T_p \& T_q \in f_2 | p<q} \theta_{pq} = 0$ and the coefficient of γ_2 equals 0, too. For example, let us consider the third factor. Given that T_4 and T_1 are adjoining, but T_2 is not located next to T_4 or T_1, $\sum_{T_p \& T_q \in f_3 | p<q} \theta_{pq} = 1$, i.e., coefficient of γ_3 equals 0.5.

Each activity might have at most two neighbourhoods in any solution. This limitation is observed by Inequality (3). In terms of the mathematical field of graph theory, these constraints create a condition in which Model (2)–(5) generate only simple trees instead of unacceptable trees, as seen in Figure 1(a).

The solution of the problem should be a contiguous chain of all the activities. As a result, no cycle is accepted. Constraint (4) prevent generating cyclic patterns like those in Figure 1(b).

After solving Model (2)–(5), two issues should be handled. First, the solution may be non-connected trees as in Figure 1(c) in which we need to join them to make a linear arrangement of all activities. Of course, it is possible to add Constraint (6) to Model (2)–(5) to prevent the generation of non-connected trees. However, adding this constraint is not advised, because it causes the model to randomly connect the parts of a non-connected tree. Instead, we advise the experts to compare the tail-end activities of non-connected parts of trees and then, determine the most similar activities as joining points. As the second issue, we need to determine the direction of the arranged structure of activities from the most important to the least important. This objective can be reached by comparing two tail-end activities of the structure using the expert's opinions.

$$\sum_{p=1}^{m} \sum_{q=1}^{m} \theta_{pq} = m - 1. \tag{6}$$

4. Cases studied

4.1. Transformer oil commercialization

The focus of this case concerns a project of the know-how commercialization of a technology known in the industry as transformer oil. This technology is a highly-refined mineral oil that is stable at high temperatures and has excellent electrical insulating properties. It is used in oil-filled transformers, some high-voltage capacitors, fluorescent lamp ballasts, and some types of high-voltage switches and circuit breakers. Transformer oil is used in oil-immersed transformers, high-voltage capacitors, tap changers, fluorescent lamp ballasts, and some switches and circuit breakers.

One of the project modules requires prioritizing seven major independent activities, as given in Table 1. Two experts were involved in this case. The importance of the experts was identified as $u_1 = 1$ and $u_2 = 0.5$. The experts recommended five factors, #1 to #5. The first expert proposed the factors' weights 8, 8, 5, 4, and 7, respectively. The latter only conferred Factors #2 and #5 with $w_{22} = 3.5$ and $w_{52} = 9$, respectively. The final weights of the factors were obtained as $\gamma_1 = \left(8^1\right)^{1/1} = 8$, $\gamma_2 = \left(8^1 \times 3.5^{0.5}\right)^{1/1.5} = 6.07$, $\gamma_3 = \left(5^1\right)^{1/1} = 5$, $\gamma_4 = \left(4^1\right)^{1/1} = 4$, and $\gamma_5 = \left(7^1 \times 9^{0.5}\right)^{1/1.5} = 7.61$. Model (2)–(5) were formulated with 21 variables and 105 constraints. This model was solved by means of the LINGO release 8.0 on a personal computer. The optimal arrangement of

Table 1. The activity-factor matrix for the case studied.

Activities		Factors 1	2	3	4	5
T_1	Specifying electrically insulate requirements.	*	*	–	–	–
T_2	Specifying suppressed corona and arcing equipment.	–	*	*	*	–
T_3	Studying coolant features	–	*	*	–	*
T_4	Determining resist oxidation items	*	*	–	–	–
T_5	Determining deposit formation items	*	–	–	–	–
T_6	Studying the overall system performance	–	*	–	*	*
T_7	Creating a manual for installation of the system	*	–	–	–	–

the activities was obtained as T_7 T_5 T_4 T_1 T_2 T_3 T_6 corresponding to the objective value 26.68. Therefore, Factor #4 is the only one whose activities include a gap. The experts determined T_6 as the most important activity.

4.2. Planning at NIOC

National Iranian Oil Company (NIOC), according to its mission on oil production from the country's reservoirs, compiles five-year productions and investment plans for all oil and gas fields in the country. This task is the responsibility of the project planning management deputy of the NIOC. In this regard, the deputy receives final reports of several studies made by other deputies (such as deputy for hydrocarbon reservoirs, deputy for economic feasibility and investment, deputy for oil and gas projects, and so on) and they are analyzed to make investment plans.

The current case reports a real project at the NIOC in which the output of eight studies is received for analysis by the relevant team. Therefore, eight activities are defined as follows:

- T_1: Analyzing reservoir study;
- T_2: Analyzing surface and downstream study;
- T_3: Analyzing the study of target markets;
- T_4: Analyzing the study of energy capital financing;
- T_5: Analyzing the study of the type of development and maintenance contracts;
- T_6: Analyzing oil transfer and transportation study;
- T_7: Analyzing similar projects;
- T_8: Analyzing the production of crude oil.

Due to the limited human resources in the team (i.e., technical and planning experts), the activities must be considered sequentially. Moreover, based on the four factors defined by the planning management of the NIOC, the following groups of activities were determined to be done continuously:

- Activities T_1 and T_2 (due to engineering and reservoir factor, with weight 100);
- Activities T_3, T_6, and T_8 (due to international market factor, with weight 40);
- Activities T_4 and T_5 (due to economic feasibility factor, with weight 70);
- Activities T_4 and T_7 (due to technology factor, with weight 40).

Using the proposed model, the optimal solution was obtained including three non-connected trees T_5 T_4 T_7, T_1 T_2 and T_6 T_3 T_8. Finally, according to the expert's opinions, a linear arrangement of the activities was determined as T_1 T_2 T_7 T_4 T_5 T_6 T_3 T_8 with the best objective value 250.

5. Dealing with large scale problems

Model (2)–(5) include $m^2/2-m/2$ binary variables and $2^m - m^2/2 + m/2$ constraints ($m > 2$). Hence, the constraints exponentially grow in number by increasing m. For example, there are 22 constraints for $m = 5$, 979 constraints for $m = 10$, and 32663 constraints for $m = 15$. The problem was modeled as binary programming which is a subset of integer programming. Since integer programming is NP-hard [36,37], the problem must be hard. Due to the involved NP-hardness of the problem, heuristic and meta-heuristic approaches may strike us. However, the current paper aims at looking for a method that provides a global optimal solution. To solve large-scale models, this paper suggests a relaxation that removes Constraint (4) from Model (2)–(5), resulting in a relaxed model with only m constraints. Needless to say, the solution pattern of the relaxed model may include cycles and non-connected sub-graphs. Definitely, the output pattern may be one of the two cases: (a) the solution pattern, i.e., a linear layout of all activities, or (b) a non-connected graph which includes at least one cycle. Accordingly, at first, we should identify any cycles, if those exist. With this purpose in mind, a pseudo code is designed as in Figure 2. This code, in each round, arbitrarily chooses an initial vertex named "root" and then, tries to search for the next vertex connected to the initial vertex. The algorithm continues to search for the next vertex connected to the previous one found until getting the root or failing to find the next connected vertex. The former shows a cycle, while the latter indicates a chain.

Obviously, if there are no cycles (i.e., $t = 0$), the current pattern is the optimal solution; otherwise, there will be one or more cycles. Therefore, in order to prevent generating cycles, by using a row generation approach, new Constraint (7) will be added to Model (2)–(5), where V_r is the set of vertices (i.e., activities) for the rth cycle. This approach is often called branch-and-cut algorithm. Moreover, it should be noted that the row generation approach is a dual form of the column generation approach [38]:

$$\sum\nolimits_{T_p \& T_q \in V_r | p<q} \theta_{pq} \leq s\left(V_r\right) - 1, \qquad \forall r = 1, 2, \ldots, t. \tag{7}$$

To sum up, let us review the proposed procedure. At the first iteration, Model (2)–(5) are solved without Constraint (4) (i.e., $t = 0$). Now, subject to the presence of any cycle in the generated pattern, Constraint (7) would be added to the model. Admittedly, this version of the model includes Constraint (7) instead of Constraint (4). This model should be solved and rechecked in the presence of cycles again. This iterative process continues until no cycle can be found.

```
Inputs: Set of θ_pq values, came from solution of the relaxed mathematical model
Method:
{
CT ← {T_i, ··· , T_m}   !the activities to be checked
t ← 0
Next cycle:
  Take an element with smallest numerator from CT, named root
  x ← numerator of root
  Cycle ← {T_x}
  previous← {}
  CT←CT-{T_x}
  Next search:
    Search ({T_1, ··· , T_m}-{previous}) for an element T_y in the way that

    (θ_xy = 1 (if x < y) or θ_xy = 1 (if y < x ); then:
        On success do:      !an activity connected T_x to was found
            {
            CT ← CT − {T_y}
            Cycle← Cycle+ {T_y}
            If T_y = root then put t ← t + 1 & V_t ←Cycle; after that go to Next cycle  !a cycle was found
            previous← T_x
            x ← y
            Go to Next search
            }
        On failure, if CT = {} then stop; otherwise go to Next cycle
}
Outputs: Sets of cycles: there are t cycles ( r = 1, 2, ··· , t ); V_r: Set of vertices in the rth cycle.
```

Figure 2. The pseudo code for cycle identification.

6. Experimental evaluation

As a preliminary assumption, there are three parameters affecting running time: the number of activities (m), the number of factors (n), and the size of factors (s). Consequently, a typical problem is addressed by (m,n,s) i.e., a problem with m activities and n factors by size s. Given that the current study is the first research to consider the introduced problem, there is naturally no benchmark instance in the literature. Hence, the instance problems (i.e., data set) were randomly generated by a macro module program in Excel software. LINGO software version 8.0 was employed to solve the mathematical models along with an Excel macro to identify cycles. The procedure was run on an ASUS Core (i7-7500) 2.7 GHz with 12.00 GB memory.

6.1. Sensitivity analysis

For the purpose of sensitivity analysis, a study is conducted on several instance problems with the number of activities ranging from $m = 25$ to $m = 300$, the number of factors ranging from $n = 100$ to $n = 2500$, and the sizes of factors ranging from $s = 3$ to $s = 75$. Figure 3(a)–(f) portray running time plots resulting from solving instance problems. All the reported times are average of ten times that the algorithm runs on each level of the parameters.

Figure 3(a), (b), and (c) show the running time variation with s, each curve under a fixed m and n. These figures indicate that increasing s often results in excessive increase in the running time. Interestingly, we found that at $n = 500$ and $n = 1000$, a peak occurs for s around 10. Figure 3(d) and (e) present how the running time varies as n is grown. These figures are made with fixed m (i.e., $m = 100$ and $m = 150$) and three levels $s = 10$, $s = 20$, and $s = 30$. The curves in Figure 3(d) show the slowly incremental effect of n on the running time of instance problems, even up to 2500 factors. On the contrary, Figure 3(e) does not follow the same trends as curves in Figure 3(d). The curves have an increasing trend up to around $n = 1550$ followed by an abrupt drop, such that there is a rather uniform trend after $n = 1600$. Figure 3(b) confirms this phenomenon, in which the curve of $n = 2000$ is quite below the curve of $n = 1500$. Figure 3(f) shows the running time variation with n. This figure is drawn based on instance problems by $n = 500$ and four levels of s. The results of analysis illustrate that the curves as in Figure 3(f) are the only ones with exponential growth. Almost all the curves show an exponential growth in the running time as m increases. Except curve with $s = 3$, on the other cases there were large scale instance problems (about $m = 250$ to $m = 300$ activities) with a huge running time, even more than 24 hours. Note that these long running times were achieved only at the first iteration of the solving procedure.

Regarding unusual variations shown in Figure 3(c) and (e), we could not find a definite interpretation except that they may result from intrinsic connections among factors. It should be noted that all the solved instance problems obtained an optimal solution through 3 to 12 iterations. Further, as shown in the figures, running times range between 1 and 3500 seconds. In sum, the running times are very reactive (i.e., alert) to m and slightly sensitive to s and n.

6.2. Efficiency analysis

In order to demonstrate the efficiency of the developed

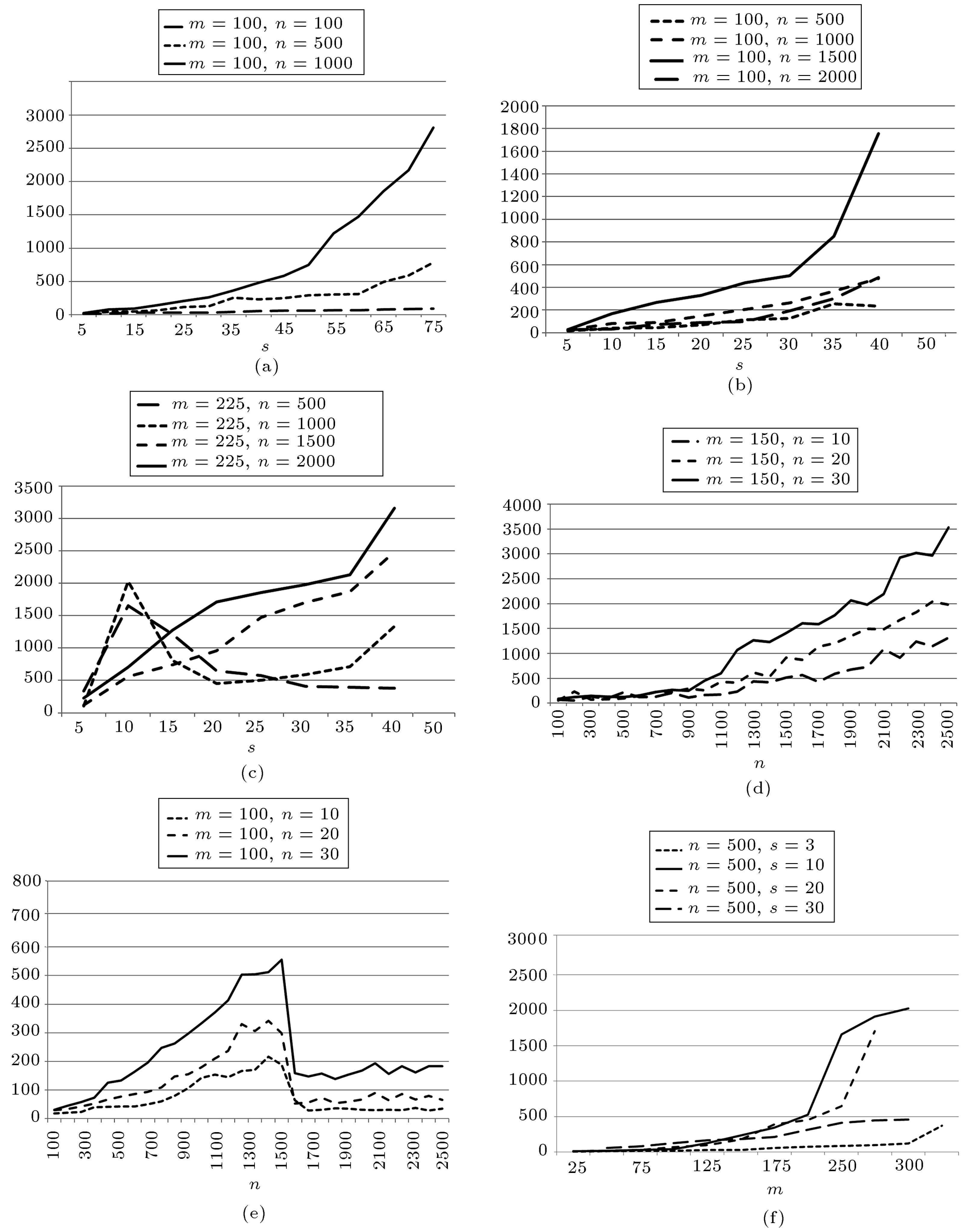

Figure 3. Comparative analysis of running times (in seconds).

row generation algorithm, the employed approach attempts to (a) solve the test problems using the un-relaxed Model (2)–(5) as much as possible depending on running time or computer memory limitations and then, (b) to solve the same test problems using the row generation algorithm. This will indicate the effectiveness of the row generation algorithm over the un-relaxed Model (2)–(5). In order to determine the values of such parameters as m, n, and s to generate the instance problems, the major result of the previous section, i.e., sensitivity analysis, was considered. In this respect, the number of activities (m) is considered as the main parameter that differs from other parameters at two levels of low ($n = m$ and $s \cong 0.25$ m) and high ($n = 3$ m and $s \cong 0.75$ m). Table 2 presents the characteristics and results of 10 test problems with the number of activities ranging from $m = 5$ to $m = 19$ for the un-relaxed Model (2)–(5), compared with the relaxed version. It was not possible to bring in the instances with more activities due to computer memory limitations (after verifying that the application of many activities did not affect the obtained results). In comparison, the two last columns in Table 2 were considered. The former, i.e., reduction of constraints,

Table 2. The characteristics and results of 10 instances for both un-relaxed and relaxed forms.

		The un-relaxed form				The relaxed form					Comparison	
Problem	(m,n,s)	Variables	Constraints	Optimal objective function value	Running time (seconds)	Variables	Constraints	Added constraints for loops	Final objective function value	Running time (seconds)	Decreasing in constraints (percentage)	Decreasing in running time (percentage)
P1	(5,5,3)	10	22	237.5	0.06	10	5	1	237.5	0.05	72.727%	16.667%
P2	(5,15,4)	10	22	692.66	0.07	10	5	3	692.66	0.05	63.636%	28.571%
P3	(10,10,3)	45	979	355	0.1	45	10	0	355	0.05	98.979%	50.000%
P4	(10,30,7)	45	979	1319.5	0.1	45	10	1	1319.5	0.08	98.876%	20.000%
P5	(15,15,4)	105	32663	536.66	3.32	105	15	4	536.66	0.07	99.942%	97.892%
P6	(15,45,11)	105	32663	1949.5	6.04	105	15	3	1949.5	0.07	99.945%	98.841%
P7	(17,17,4)	136	130936	467.33	66.13	136	17	9	467.33	0.1	99.980%	99.849%
P8	(17,51,13)	136	130936	2290.583	23.9	136	17	4	2290.583	0.09	99.984%	99.623%
P9	(19,19,5)	171	524117	627	121.3	171	19	2	627	0.08	99.996%	99.934%
P10	(19,57,14)	171	524117	2259	693	171	19	4	2259	0.12	99.996%	99.983%

is defined as one minus the ratio of the total number of constraints in the un-relaxed Model (2)–(5) to that in the relaxed model. The latter, i.e., reduction of running time, is similar to the former, except that it concerns the running time instead of the total number of constraints. Unquestionably, the results confirm the efficiency of the proposed algorithm.

6.3. Running time analysis

This section answers this question: "What size of the problems can be solved using the proposed branch-and-cut algorithm within a logical time?" Among many solved problems, the results for the average running time of 45 typical tested problems are shown in Table 3 and sorted in terms of running time. Problems P43 to P45 are typical instances with a very long running time. It can be concluded that the proposed procedure is undeniably essential to get the optimal solution to the problems including even up to 250, or some more activities, in 60 minutes at most. However, when the number of activities is 300 or more, the running time may unexpectedly jump to even more than 24 hours.

7. Concluding remarks

This study defined a particular problem of prioritizing activities, named Activity Prioritizing Problem (APP) with the aim of handling activities one after another by a serial form. Later, the notion of making a solution to the problem via a mathematical model was explained. The results of the state-of-the-art optimization and decision analysis illustrated that the proposed idea has not been studied before. By taking this fact into consideration, the first contribution of the research is the introduction of the idea of prioritizing activities as an optimization problem. In addition, the challenge of large-scale problems was discussed; next, a branch-and-cut procedure to deal with such situations was developed. Thereby, another contribution of the research is demonstrating how a row generation

Table 3. The running time (in seconds) of the selected instances.

Problem	m	(m,n,s)	Running time	Average
P1	150	(150,100,30)	98	946.61
P2		(150,500,30)	126	
P3		(150,500,10)	567	
P4		(150,1800,10)	651	
P5		(150,1900,10)	798	
P6		(150,200,20)	896	
P7		(150,2500,10)	917	
P8		(150,2000,20)	1064	
P9		(150,2200,10)	1099	
P10		(150,1500,30)	1134	
P11		(150,2100,10)	1295	
P12		(150,2500,10)	1526	
P13		(150,2000,30)	2135	
P14	200	(200,500,10)	105	443
P15		(200,500,3)	112	
P16		(200,500,20)	280	
P17		(200,500,30)	294	
P18		(200,500,30)	427	
P19		(200,500,20)	623	
P20		(200,500,10)	1260	
P21	225	(225,1500,5)	91	1726
P22		(225,500,5)	189	
P23		(225,2000,5)	217	
P24		(225,1000,10)	266	
P25		(225,2000,5)	322	
P26		(225,500,40)	378	
P27		(225,1000,20)	476	
P28		(225,1000,40)	910	
P29		(225,1500,20)	959	
P30		(225,2000,20)	1715	
P31		(225,500,5)	2065	
P32		(225,2000,30)	2163	
P33		(225,1000,10)	6636	
P34		(225,1000,10)	7777	
P35	250	(250,500,3)	105	1300.83
P36		(250,500,20)	252	
P37		(250,500,10)	693	
P38		(250,500,20)	1267	
P39		(250,500,10)	2492	
P40		(250,500,20)	2996	
P41	300	(300,500,3)	413	–
P42		(300,500,30)	651	
P43		(300,500,10)	More than 24 hours	
P44		(300,500,20)	More than 24 hours	
P45		(300,500,30)	More than 24 hours	

approach could be used to get the optimal solution of the APP. The evaluations and discussions presented the efficiency and usability of the methodology for large-scale problems, even for instances up to 250 activities. In conclusion, the proposed methodology has a promising future and enjoys practical applications in project decision-making challenges in a productive manner. Further research can be undertaken in (I) Applying the proposed method to complex and large-size real-world cases and (II) Developing heuristic algorithms to solve the problem.

References

1. Gerardi, B., *No-Drama Project Management: Avoiding Predictable Problems for Project Success*, 1st Edn., pp. 2–3, Apress, N.Y, USA (2011).
2. Birjandi, A., Mousavi, S.M., and Vahdani, B. "Designing a resource-constrained project scheduling model considering multiple routes for flexible project activities: Meta-heuristic algorithms", *Scientia Iranica, E*, **27**(5), pp. 2572–2591 (2020).
3. Namazian, A., Haji Yakhchali, S., and Rabbani, M. "Integrated bi-objective project selection and scheduling using Bayesian networks: A risk-based approach", *Scientia Iranica, E.*, **26**(6), pp. 3695–3711 (2019).
4. Dorfeshan, Y., Mousavi, S.M., Vahdani, B., et al. "Determining project characteristics and critical path by a new approach based on modified NWRT method and risk assessment under an interval type-2 fuzzy environment", *Scientia Iranica, E*, **26**(4), pp. 2579–2600 (2019).
5. Pheng, L.S. and Chuan, Q.T. "Environmental factors and work performance of project managers in the construction industry", *Int. J. Project Management*, **24**, pp. 24–37 (2006).
6. Turner, J.R. and Muller, R. "The project manager's leadership style as a success factor on projects: a literature review", *Project Management J.*, **36**(1), pp. 49–61 (2005).
7. Berjis, N., Shirouyehzad, H., and Jouzdani, J. "A new approach to determine the weights of project activities using data envelopment analysis: a case of Mobarakeh steel company", *Int. J. Managing Projects in Business*, **13**(6), pp. 1187–1217 (2020).
8. Greek, D. and Pullin, J. "Overrun, overspent, overlooked", *Professional Engineering*, **12**(3), pp. 27–8 (1999).
9. Chang, T.C. and Ibbs, C.W. "Priority ranking: A fuzzy expert system for priority decision making in building construction resource scheduling", *Building and Environment*, **25**, pp. 253–267 (1990).
10. Babauta, L., *Zen To Done: The Ultimate Simple Productivity System*, 1st Edn., LLC, Phatbits, UK (2012).
11. Covey, S.R., *7 Habits of Highly Effective People*, 25th Edn., Pocket books, UK (1999).
12. Baykasoglu, A., Gocken, T., and Kaplanoglu, V. "A practical approach to prioritize project activities through fuzzy ranking", *Cybernetics and Systems: An International Journal*, **42**(3), pp. 165–179 (2011).
13. Alencar, L.H., Mota, C.M.M., and Almeida, A.T. "Activities prioritization in a construction project management based on multi-criteria methods", *Third International Conference on Production Research*, Americas' Region, Federaço da Induústrias do Estado do Paraná, Brazil, July 30-August 2 (2006).
14. Jha, K.N. and Misra, S. "Ranking and classification of construction coordination activities in Indian projects", *Construction Management and Economics*, **25**, pp. 409–421 (2007).
15. Mota, C.M.M., Almeida, A.T., and Alencar, L.H. "A multiple criteria decision model for assigning priorities to activities in project management", *Int. J. Project Management*, **27**, pp. 175–181 (2009).
16. Vanhoucke, M. "Using activity sensitivity and network topology information to monitor project time performance", *Omega*, **38**(5), pp. 359–370 (2010).
17. Mota, C.M.M. and de Almeida, A.T. "A multi-criteria decision model for assigning priority classes to activities in project management", *Annals of Operations Research*, **199**(1), pp. 361–372 (2012).
18. Golpira, H. "Making comparison between FMCDM and MCDM to define project activities weights", *First International Scientific Conference on Project Management in the Baltic Countries*, February 8-9, Riga, University of Latvia, **35** (2012).
19. Hadad, Y., Keren, B., and Laslo, Z. "Applications of ranking indexes of project activities for project management tasks", *Challenges of IT Project Portfolio Selection*, **1392**, 1422 (2014).
20. Hadad, Y. "Multi-criteria methods for ranking project activities", *Yugoslav J. Operations Research*, **26**(2), pp. 201–219 (2016).
21. Kalayathankal, S.J., Varghese, J., and Abraham, J.T. "A modified fuzzy approach to prioritize project activities", *Int. J. Engineering and Technology*, **7**(2), pp. 158–162 (2018).
22. Hwang, C.L. and Yoon, K., *Multiple Attribute Decision Making: Methods and Applications: A State-of-the-art Survey*, 1st Edn., Springer, Berlin (1981).
23. Zavadskas, E.K. and Podvezko, V. "Integrated determination of objective criteria weights in MCDM", *Int. J. Information Technology and Decision Making*, **15**(2), pp. 267–283 (2016).
24. Vinogradova, I., Podvezko, V., and Zavadskas, E.K. "The recalculation of the weights of criteria in MCDM methods using the Bayes approach", *Symmetry*, **10**(205), pp. 2–18 (2018).
25. Hatefi, M.A. "A model for measuring success of an organisational management system", *Int. J. Advanced Operations Management*, **9**(2), pp. 73–87 (2017).

26. Pinheiro, R.G.S., Martins, I.C., Protti, F., et al. "A matheuristic for the cell formation problem", *Optimization Letters*, **12**(2), pp. 335–346 (2018).

27. Rabbani, M., Habibnejad-Ledari, H., Rafiei, H., et al. "A bi-objective mathematical model for dynamic cell formation problem considering learning eect, human issues, and worker assignment", *Scientia Iranica, E.*, **23**(5), pp. 2341–2354 (2016).

28. Leusin, M.E., Frazzon, E.M., Maldonado, M.U., et al. "Solving the Job-Shop Scheduling Problem in the Industry 4.0 Era", *Technologies*, **6**(107), pp. 2–27 (2018).

29. Palubeckis, G. "Single row facility layout using multi-start simulated annealing", *Computers and Industrial Engineering*, **103**, pp. 1–16 (2017).

30. Ceberio, J., Mendiburu, A., and Lozano, J.A. "The linear ordering problem revisited", *Eur. J. Operational Research*, **241**(3), pp. 686–696 (2015).

31. Horton, S.B., Gary Parker, R., and Borie, R.B. "On minimum cuts and the linear arrangement problem", *Discrete Applied Mathematics*, **103**, pp. 127–139 (2000).

32. Glover, F. and Rego, C. "New relationships for multi-neighbourhood search for the minimum linear arrangement problem", *J. Discrete Algorithms*, **46**47, pp. 16–24 (2017).

33. Seppanen, J. and Moore, J.M. "Facilities planning and graph theory", *Int. J. Management Science*, **17**, pp. 242–253 (1970).

34. Hatefi, M.A. "BRAW: block-wise rating the attribute weights in MADM", *Computers and Industrial Engineering*, **156**, 107274 (2021).

35. Hillier, F. and Lieberman, G., *Introduction to Operations Research*, 11th Edn., McGraw-Hill Education, NY, USA (2020).

36. Garey, M.R. and Johnson, D.S. "Computers and intractability: A guide to the theory of NP completeness", WH Freeman, NY, USA (1979).

37. Page, D.R. "Approximation algorithms for problems in make-span minimization on unrelated parallel machines", Ph.D. Thesis, University of Western Ontario (2019).

38. Hatefi, M.A. "Developing column generation approach to solve the rectangular two-dimensional single knapsack problem", *Scientia Iranica, E.*, **24**(6), pp. 3287–3296 (2017).

Appendix A

In order to illustrate how Model (2)–(5) are written, this appendix provides a numerical example for a small problem. This problem includes five activities, T_1 to T_5, and six factors with relevant sets as $f_1 = \{T_3, T_4\}$, $f_2 = \{T_1, T_3\}$, $f_3 = \{T_1, T_5\}$, $f_4 = \{T_1, T_2\}$, $f_5 = \{T_1, T_3, T_4, T_5\}$, and $f_6 = \{T_1, T_2, T_4\}$. The weights of factors are $\gamma_1 = 9$, $\gamma_2 = 7$, $\gamma_3 = 8$, $\gamma_4 = 2$, $\gamma_5 = 7$, and $\gamma_6 = 3$. The mathematical Model (2)-(5) for this problem are as follows:

$$\max 9\left(\theta_{34}\right) + 7\left(\theta_{13}\right) + 8\left(\theta_{15}\right) + 2\left(\theta_{12}\right) + 7\left(\frac{\theta_{13}+\theta_{14}+\theta_{15}+\theta_{34}+\theta_{35}+\theta_{45}}{3}\right) + 3\left(\frac{\theta_{12}+\theta_{14}+\theta_{24}}{2}\right),$$

$$\theta_{12}+\theta_{13}+\theta_{14}+\theta_{15} \leq 2,$$

$$\theta_{12}+\theta_{23}+\theta_{24}+\theta_{25} \leq 2,$$

$$\theta_{13}+\theta_{23}+\theta_{34}+\theta_{35} \leq 2,$$

$$\theta_{14}+\theta_{24}+\theta_{34}+\theta_{45} \leq 2,$$

$$\theta_{15}+\theta_{25}+\theta_{35}+\theta_{45} \leq 2,$$

$$\theta_{12}+\theta_{13}+\theta_{23} \leq 2,$$

$$\theta_{12}+\theta_{14}+\theta_{24} \leq 2,$$

$$\theta_{12}+\theta_{15}+\theta_{25} \leq 2,$$

$$\theta_{13}+\theta_{14}+\theta_{34} \leq 2,$$

$$\theta_{13}+\theta_{15}+\theta_{35} \leq 2,$$

$$\theta_{14}+\theta_{15}+\theta_{45} \leq 2,$$

$$\theta_{23}+\theta_{24}+\theta_{34} \leq 2,$$

$$\theta_{23}+\theta_{25}+\theta_{35} \leq 2,$$

$$\theta_{24}+\theta_{25}+\theta_{45} \leq 2,$$

$$\theta_{34}+\theta_{35}+\theta_{45} \leq 2,$$

$$\theta_{12}+\theta_{13}+\theta_{14}+\theta_{23}+\theta_{24}+\theta_{34} \leq 3,$$

$$\theta_{12}+\theta_{13}+\theta_{15}+\theta_{23}+\theta_{25}+\theta_{35} \leq 3,$$

$$\theta_{12}+\theta_{14}+\theta_{15}+\theta_{24}+\theta_{25}+\theta_{45} \leq 3,$$

$$\theta_{13}+\theta_{14}+\theta_{15}+\theta_{34}+\theta_{35}+\theta_{45} \leq 3,$$

$$\theta_{23}+\theta_{24}+\theta_{25}+\theta_{34}+\theta_{35}+\theta_{45} \leq 3,$$

$$\theta_{12}+\theta_{13}+\theta_{14}+\theta_{15}+\theta_{23}+\theta_{24}, +\theta_{25}+\theta_{34}+\theta_{35}+\theta_{45} \leq 4$$

$$\theta_{12},\theta_{13},\theta_{14},\theta_{15},\theta_{23},\theta_{24},\theta_{25},\theta_{34},\theta_{35},\theta_{45} \in (0,1).$$

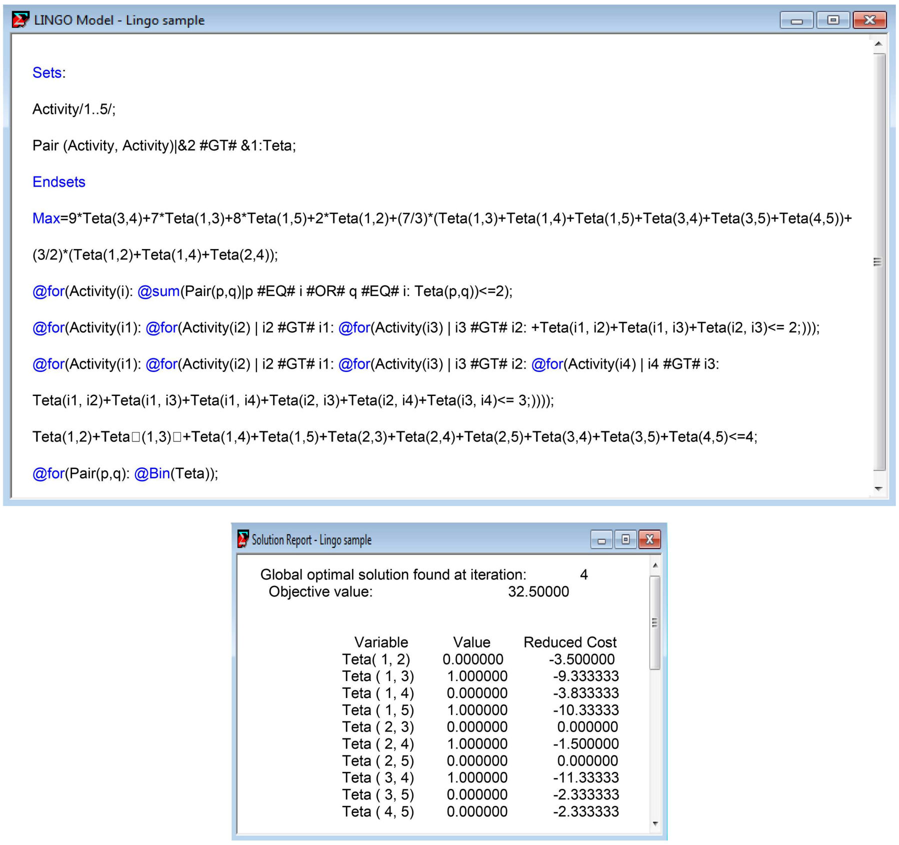

Figure A.1. The LINGO code and the related solution report for the numerical example.

In this example, the LINGO code and the solution are presented as in Figure A.1, leading to the optimal string as T_5 T_1 T_3 T_4 T_2. This corresponds to the objective value 32.5.

Biographies

Mohammad Ali Hatefi is an Associate Professor at the Department of Energy Economics and Management at Petroleum University of Technology (PUT). He graduated from Iran University of Science and Technology (IUST) with honour. His areas of interest are operations research, decision analysis, project management, and risk management. He is the author of several scientific publications in the area of decision analysis, operations research, and risk management. He is currently working for Journal of Petroleum Business Review (PBR) as an editorial board member.

Seyyed Abdollah Razavi is an Assistant Professor at the Department of Energy Economics and Management at PUT. He graduated from Ferdowsi University of Mashhad in the field of energy economics. His areas of interests and researches are energy management, energy economics, financial market, and political economy. He is presently working for Journal of Petroleum Business Review (PBR) as an executive manager.