Scientia Iranica E (2020) 27(1), 361–376

SCIENTIA IRANICA

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

Robust optimization for the resource-constrained multi-project scheduling problem with uncertain activity durations

E. Nabipoor Afruzi*, **A. Aghaie**, **and A.A. Najafi**

Faculty of Industrial Engineering, K.N. Toosi University of Technology, Tehran 1999143344, Iran.

Received 13 June 2017; received in revised form 12 March 2018; accepted 6 August 2018

KEYWORDS
Multi-project scheduling problem;
Resource sharing policy;
Robust optimization;
Resource constraint;
Uncertain activity duration.

Abstract. This paper studies the multi-project scheduling problem that involves multiple projects with different importance weights, predefined assigned due dates, activities with uncertain durations, and renewable constrained resources. The resource sharing policy is applied to share resources among projects. Due to the environmental rapid changes and, also, the uniqueness of projects, the probability distribution function of uncertain durations cannot be estimated with confidence. Besides, the multi-project scheduling problem with its large-scale investment dictates a conservative approach to deal with the existing uncertainty. Therefore, the Robust Resource-Constrained Multi-Project Scheduling Problem (RRCMPSP) is studied in this paper, while the maximum total weighted tardiness of the projects should be minimized. A scenario-relaxation algorithm is implemented, which results in optimal solutions for the RRCMPSP. The aim is to find an optimal structure that contains all of the projects such that it transfers the resources between the activities based on the resource sharing policy, while the maximum weighted differences between the projects' finish times and their assigned due dates will be minimum.

1. Introduction

The Resource-Constrained Project Scheduling Problem (RCPSP) aims to minimize the project makespan while considering precedence and resource constraints [1]. This problem is one of the most well-known problems to which researchers have devoted considerable efforts over the past decade.

The RCPSP is applicable in many areas such as make to order industries, construction, software development, etc. In modern enterprises in which a large number of projects are set up to achieve the product innovation, the key resource is mostly manpower, which belongs to renewable resources. In contrast to the importance of renewable resources and its role in project management success, the renewable resources have not attained sufficient consideration in the literature [2]. As a brief definition, the renewable resources are those resources such as manpower, machines, etc. that are constrained and that there is a certain available capacity of this kind of resources in each time period. By finishing one activity, its required renewable resources can be released and applied to other activities. In this paper, the project scheduling problem is investigated under the renewable resource-constraint condition.

The Resource-Constrained Multi-Project Sched-

*. *Corresponding author. Tel.: +98 21 84063363; Fax:+98 21 88674858*
E-mail addresses: enabipoorafruzi@mail.kntu.ac.ir (E. Nabipoor Afruzi); aaghaie@kntu.ac.ir (A. Aghaie); aanajafi@kntu.ac.ir (A.A. Najafi)

doi: 10.24200/sci.2018.20801

uling Problem (RCMPSP) as the extension of the RCPSP is considered as the simultaneous scheduling of two or more projects, which demand the same scarce sources [3]. Multi-project management is a major way of doing business both in manufacturing and services, and being a large-scale complex problem constitutes an important research area [4]. According to the study by Payne [5], up to 90% of all projects in the world are executed in a multi-project management environment. It is notable that the management of multiple projects presents challenges that are fundamentally different from single project management [6]. Thus, managing the multi-project problem is not simply an aggregate of single project efforts. In this paper, the multi-project management problem is investigated.

During the project execution in an indeterminate environment, the projects are subject to considerable uncertainty. In other words, due to unavailable resources, delays in the delivery of materials, absent employees, bad weather conditions, accidents, and many other uncontrollable factors, some project activities may last longer than expected, threatening the operational viability of the planned schedule [7]. Therefore, the obtained results of the project scheduling model with deterministic parameters are no longer valid. In other words, when the project parameters take realized values, the usability of any result of the deterministic models is under question. Therefore, it is conceivable that as the data takes values different from the nominal ones, several constraints may be violated and the optimal solution found using the nominal data may no longer be optimal or even feasible [8]. In this paper, the uncertainty of the activities' duration is under study.

There are several approaches to scheduling projects under uncertainty. In order to select an appropriate approach for dealing with uncertainty in problems, first of all, we should investigate the nature and characteristic of the studied problem. The fundamental approaches to scheduling projects under uncertainty are reactive scheduling, stochastic scheduling, scheduling under fuzziness, proactive (robust) scheduling, and sensitivity analysis [9].

Considering the uniqueness of each project in the real world, it is not uncommon that its activities are seldom or even never have been executed before. Therefore, these indeterminacies cannot be treated as fuzziness, probability, roughness, ambiguity or entropy. Instead, uncertainty theory can be a useful tool [1]. Robust Optimization is an appropriate approach that is totally compatible with the nature of the project scheduling problem and is applied in this paper. Robust optimization belongs to an important methodology for dealing with optimization problems with data uncertainty. In this type of the method, a deterministic data set is defined within the uncertain space, and the best solution, which is feasible for any realization of the data uncertainty in the given set, is computed through the solution of the robust counterpart optimization problem [10].

The major advantage of robust optimization compared to stochastic programming is that no assumptions regarding the underlying probability distribution of the uncertain data are required [11]. It is also true when comparing the robust optimization approach with the fuzzy approach because there is no need for RO to define the membership function for the uncertain parameter.

On the other hand, in this paper, the multi-project scheduling problem is investigated that requires time, cost, resources, etc. in a large-scale quantity. Therefore, it seems that a conservative approach is essential that can immunize the project scheduling problem against data uncertainty. It is exactly the characteristics of the robust optimization approach that is applied for dealing with uncertainty in this paper.

In this paper, the robust optimization approach is applied to the multi-project scheduling problem under resource constraint and uncertain activities' duration to cover some shortcomings in the existing multi-project models. The problem is represented in a two-stage model in which the objective function is to minimize the maximum total weighted tardiness of the projects.

The structure of the paper is as follows: Section 2 describes the related literature review. The definitions of the problem are presented in detail in Section 3. The proposed mathematical model and the two-stage approach are explained in detail in Section 4. Section 5 describes one simple numerical example with its results to clarify the proposed model. Computational experiments are explained in Section 6. Finally, the conclusion and further research are presented in Section 7.

2. Related works

The related works about the multi-project scheduling problem, the resource management policies, and the project scheduling problem under uncertainty are mentioned in this section briefly.

2.1. Multi-project scheduling problem

The RCMPSP comes from practical multi-project environments, in which a number of projects concurrently share limited resources in precedence or other constraints [12]. In fact, the single project management rarely occurs today, and companies usually manage more than one project simultaneously called "multi-project management". The importance of multi-project management has increased over the last decades and is still growing. In the middle of the last century, project and multi-project management gained momentum; the share of project work has increased since then, and the penetration of firms by corresponding management methods has not stopped at the beginning of this century [13]. The researchers concur that the literature of the project management problem is heavily biased towards the single project environment, while there are few studies on the multi-project problem [14].

The main reason for the insignificant fruition with regard to the topic of multi-project scheduling compared with the single project one comes from its high complexity, which is affected by many factors such as the huge solution space, intense contending for resources, various and conflicting objectives, the inter-project dependence and priority, the high level of uncertainty, and so on [12]. Therefore, many researchers have studied recently the multi-project problem to overcome this identified gap [15–18]. In addition, some heuristic priority rules and metaheuristics have been studied to solve the RCMPSP [19–22].

2.2. Resource sharing policy

In the literature of the multi-project problem, the primary topic is the allocation of common resources to simultaneous projects since the resource-based relations define the multi-project problem by joining the individual projects together. The characteristics of resource usage by the individual project in the multi-project environment are described in accordance with the resource management policy [4]. In the multi-project problem, there are several projects that are executed in parallel, and they use the common resource pool for one resource type, at least. There are several approaches to optimally allocating the resources to the activities of multiple projects, such as the resource sharing policy, the resource dedication policy, etc. [16,17,19]. Regarding the existence of different existing policies, in this paper, the most common one, i.e., resource sharing policy, is applied to determine how to allocate the common resources to projects.

2.3. Project scheduling under uncertainty

There are many studies in which the deterministic environment is considered for the project scheduling problem [15,16,18,19,23]. However, in the real world, uncertainty during the project execution exists. In order to consider uncertainty in problem modeling, different assumptions can be applied. In some researches, the costs of activities are considered uncertain [24,25] while studying the project scheduling problem. However, the most often objective function in the project scheduling problem is the optimization of the project duration [9]. Thus, the duration of activities with direct influence on the makespan of the project is studied as an uncertain parameter in the following studies.

2.3.1. Stochastic project scheduling problem

The stochastic RCPSP or Stochastic RCPSP (SRCPSP) is the optimization problem that is solved when the deterministic durations in RCPSP are replaced by stochastic variables. While the goal in the classic RCPSP is to find a schedule with a minimum schedule length or makespan, the goal in SRCPSP is to minimize the expected makespan [26]. For more information, please refer to many studies that apply the stochastic approach to uncertainty in the project scheduling problem [27–29]. The serious challenging point for stochastic RCPSP is that, according to the main characteristic of the project, i.e., uniqueness, there are difficulties accessing enough historical data to fit a probability distribution for an uncertain parameter. Therefore, applying the stochastic approach to the project scheduling problem is susceptible to limitations from the practical point of view.

2.3.2. Fuzzy project scheduling problem

The fuzzy project scheduling approach is based on the concept of fuzzy activity duration, produces fuzzy schedules, and requires the membership function of the uncertain activity duration [30]. In this approach, the duration of the activities is estimated by experts, and the project manager deals with imprecise and vague judgment. For more information about fuzzy RCPSP, please refer to [31–34]. Therefore, similar to the determination of the distribution function for activities' duration in the stochastic approach, there are some challenges for project managers to determine the membership function for fuzzy activity durations.

Thus, fitting distribution function with its parameters or defining fuzzy membership function for the activities' duration has challenges from a practical point of view. In other words, this can seriously limit the application of these two approaches to the project scheduling problem.

2.3.3. Robust project scheduling problem

The robust optimization approach can immunize the project scheduling problem against uncertainty. There are only three studies that apply this approach to the RCPSP with uncertain duration in the single project problem, as will be mentioned in the following. Chakrabortty et al. [35] studied the RCPSP in which the activity durations were represented by random variables with different probability distribution functions. They proposed a robust optimization-based approach that produced reasonably good solutions under any likely input data scenario. Their proposed approach guarantees the feasibility of solutions and produces high-quality solutions. Bruni et al. [7] proposed an adaptive robust optimization model to derive the resource allocation decisions that minimized the worst-case makespan under general polyhedral uncertainty sets, assuming that the activity durations were subject to interval uncertainty. Moreover, a general decomposition approach was proposed by them to solve the robust counterpart of the RCPSP, further tailored to address the uncertainty set with the protection factor. Artigues et al. [36] proposed models for project scheduling when there was considerable uncertainty in the activity durations. They developed and implemented a scenario-relaxation algorithm and a scenario-relaxation-based heuristic. The first algorithm produces optimal solutions, but requires excessive running times even for medium-sized instances; the second algorithm produces high-quality solutions for medium-sized instances and outperforms two benchmark heuristics.

The above-mentioned studies have been done in the area of the single project scheduling problem. According to the large-scale multi-project scheduling problem, the effect of uncertainty can be more destructive. In the multi-project scheduling problem, some projects are related to each other by the common resources, and the investment of time, cost, resources, etc. is done on a large scale. Therefore, the application of the robust optimization approach as a more conservative approach that can immunize the problem against uncertainty is totally necessary. To the best of our knowledge, there is no research on the application of the robust optimization approach in the multi-project scheduling area. In the present paper, the robust optimization approach is applied to the multi-project scheduling problem under the resource constraint and uncertain duration of activities. In this research, the resource sharing policy is considered. Each project has a determined due date. In addition, the importance weight of the projects is different. The aim is to obtain an optimized structure for all of the projects in such a way that the maximum total weighted tardiness of the projects will be minimum. In this study, the development of the existing models can be demonstrated in two ways according to Figure 1.

3. Problem statement

The RCMPSP with uncertain activity durations is studied in this paper. The considered multi-project problem contains defined projects, $G = 1, 2, ..., q$. All of the projects are shown by activity-on-node network, $Graph = (V, E)$, in which the nodes demonstrate the activities of projects and the arcs represent the precedence relations between activities, E. The set of activities for each project is indicated by $V = \{0, 1, ..., n + 1\}$. For each activity $i \in V$ of project g, there is a set $P_{ig} \subset R_+$ containing the possible values for the duration of activity i of project g (R_+ is the set of non-negative real numbers). Therefore, in the discrete set of $P_{ig} = \{p_{ig1}, p_{ig2}, p_{ig3}, ..., p_{ig|P_i|}\}$, the minimum and maximum durations for activity i of project g are $P_{ig}^{\min} \equiv \min_{P_{ig}} P_{igc}$ and $P_{ig}^{\max} \equiv \max_{P_{ig}} P_{igc}$, respectively. The durations of activities 0 and $n + 1$ are considered zero: $P_{0g} = P_{n+1,g} = \{0\}, \forall g$. It is noticeable that when $p_{ig} \in P_{ig}$, $p_g = (p_{0g}, p_{1g}, ..., p_{n+1,g})$ shows one possible scenario for the activities' duration of project g. When $|P_{ig}| = 1, \forall i \in V, \forall g \in G$, the problem converts to the deterministic RCPSP.

As mentioned before, the precedence relationship between activities is shown by the binary relation of $E \subset V \times V$. The activity i of project g can be started after all its predecessors are finished. The projects apply the resource sharing policy. It means that they utilize common resources from the resource pool. There are $b_{igk} \in N$ units of resource k required by activity $i \in V$ of project g during its execution. In each project, the required resources in any type for dummy activities of 0 and $n+1$ are zero: $b_{0gk} = b_{n+1,g,k} = 0, \forall g \in G, \forall k \in R$.

A set of activities $F \subset V$ is one "Forbidden Set" of a precedence relation A if it is an anti-chain of A and at least for one type of resource $k \in R$: $\sum_{i \in F} b_{ik} > b_k$. Therefore, these sets can give rise to resource conflicts during project execution. A subset-minimal forbidden set is called a "Minimal Forbidden Set" or mfs. The set of $mfss$ for precedence relation A is

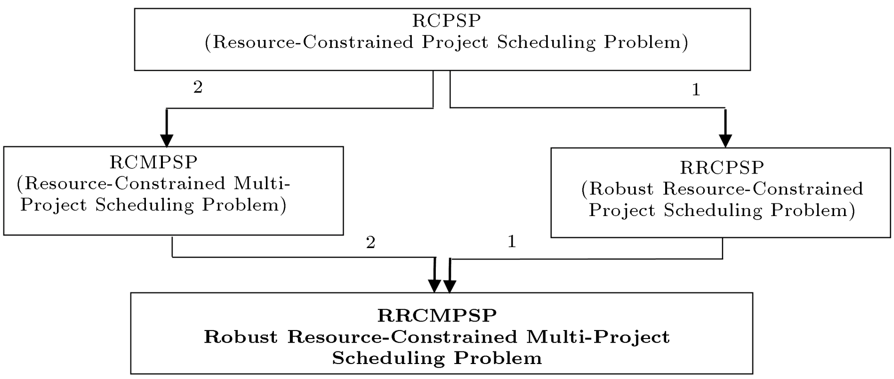

Figure 1. The model development.

written as $F(A)$ [36]. Any of the resource conflicts can be removed by adding extra precedence relationships to the primary precedence graph for postponing some activities such that the makespan can be determined by applying an Early Start policy (ES-policy) on an extended graph. Therefore, the extra precedence relationships $X \subseteq (V \times V) \backslash E$ should be found in such a way that the extended graph, $Graph'(V, (E \cup X))$, is acyclic and $F(T(E \cup X)) = \phi$ [7]. According to Balas [37], the set X containing pairs of activities that lead to one feasible ES-policy can be called a sufficient selection. After defining one selection and adding the extra precedence relationship X to the primary precedence graph E, the resource constraints can be ignored according to the precedence relationship in the EUX, and the makespan can be obtained by calculating the critical path problem on the extended graph, $Graph'(V, (E \cup X))$ [7].

The binary decision variable, $x_{igjg'}$, is introduced in this paper to show the precedence relationship between the activities. According to the characteristics of the multi-project scheduling problem, one activity and its predecessor activities are not essentially within the same project, and it is possible that one activity becomes the predecessor of another activity from a different project. Therefore, the precedence relationship between the two projects is introduced in this paper based on two reasons. The first reason is that, in many real-world multi-project scheduling problems, the precedence relationship exists between the activities of two projects. For example, consider two projects in an area with low population density: (1) construction of the residential complex and (2) installation of the town gas station. In this example, the high-pressure equipment installation activity in the second project is the predecessor of the installation and testing of the town gas system of the residential complex in the first project. In the cases with no precedence relationship between two projects, the special case may occur with $g = g'$ in the $x_{igjg'}$ notation.

The second reason is relevant to the applied calculation method. According to the minimal forbidden set, any of the resource conflicts can be removed by adding extra precedence relationships (X) to postpone some activities. Based on the resource sharing policy in the multi-project scheduling problem, the activities of different projects utilize common resources from the resource pool. Therefore, the extra precedence relationships (X) can be also created between two activities from different projects. Therefore, the variable $x_{igjg'}$ should present both of the projects between which the extra precedence relationship (X) exists.

In the $Graph'(V, (E \cup X))$, the start time of activity i of project g, $s_{i,g}(X,p)$, is the longest path from the scheduling time horizon 0 to activity i of project g. Thus, one should check the paths originating from the start activities of all projects (not only the start activity of the project g containing i) while calculating $s_{i,g}(X,p)$.

The resource flows between the activities are demonstrated by the transshipment networks [36], which can be called (resource) flow network. The number of resource types k transferred from the end of activity i of project g to the start of activity j of project g' is represented by flow $f(i,g,j,g',k) \equiv f_{igjg'k} \in \mathbb{N}$. It is notable that, for each resource type, a separate flow network will be created. The resource flow should satisfy the conservation constraints and, also, the lower and upper bounds on the flow for intermediate (not start or end) nodes [36].

There are several selections for the same schedule. Bruni et al. [7] presented a numerical example for two different selections in the project with 5 activities. They illustrated that when the activities' durations were deterministic, the project makespan would be the same for two different selections. However, according to the uncertainty condition, when the delays of the activities are also considered, different selections cause different makespan. This example shows the importance of proper resource allocation policy under uncertainty. They also stated that, in some cases, especially in the multi-project scheduling problem, the resources cannot be easily transferred between the activities; hence, the decisions about resource transfers should be made with greater care and sensitivity. As mentioned earlier, to the best of our knowledge, there has been no research on the investigation of this problem in the multi-project scheduling environment.

In this paper, the Robust Resource-Constrained Multi-Project Scheduling Problem (RRCMPSP) is studied as a two-stage robust optimization model. In this study, some projects are considered as a multi-project problem and should be scheduled while the durations of activities are not certain. For each project, a due date, DD_g, is determined by the global project manager as a deadline for finishing each of the projects and is notified to the local project managers. The aim is to minimize the deviation of each project makespan from its due date, while the required resources are in common and the activities' durations are uncertain. Of note, in the multi-project problem, the cost of deviating from the due date is not equal for different projects. Therefore, the degree of priority and importance of project g, demonstrated by w_g as its weight, should be considered in the calculation such that:

$$\sum_g w_g = 1.$$

The question is how to allocate and share the common resources between different activities so that the maximum weighted tardiness for all projects, shaping multi-project, can be minimized while the activities' durations are uncertain. Therefore, this study seeks one sufficient selection of all projects in which the maximum weighted difference between the projects' makespan and their due dates is minimized.

4. Mathematical modeling of the problem

The resource-constrained multi-project scheduling problem under uncertain durations of activities is formulated as a two-stage robust optimization model. In the following, the notations of indices, parameters, and variables used in the proposed models are represented.

4.1. The notations

A list of the notations applied in the proposed models is as follows:

Indices

- G The set of projects in the multi-project problem
- V The set of activity nodes
- R The set of renewable resources
- E The set of precedence relations between activities
- P The set of scenarios belonging to activities' durations

Parameters

- w_g The weight (priority degree) of project g
- DD_g The due date of project g
- $P_{i,g}^h$ The duration of activity i in project g under scenario h
- b_{igk} The required resource type k for performing activity i of project g
- b_k The capacity of resource type k
- $P_{i,g}^{\min}$ The minimum scenario value for the duration of activity i in project g
- $P_{i,g}^{\max}$ The maximum scenario value for duration of activity i in project g

Variables

- TTa^* The total weighted tardiness of projects
- Ta_g The tardiness of project g
- $S_{i,g}^h$ The start time of activity i of project g under scenario h
- $x_{igjg'}$ The decision variable with value one when activity i of project g is the predecessor of activity j of project g'; otherwise, it takes the value zero
- $f_{igjg'k}$ The number of resource units of type k transferred from the end of activity i of project g to the start of activity j of project g'
- a_{ig} The decision variable with value one if the duration of activity i of project g takes the maximum value, and it takes the value zero if the duration of activity i of project g takes the minimum value
- LP_g The longest path of project g in the multi-project network
- $\varphi_{g''igjg'}^{\min}\varphi_{g''igjg'}^{\max}$ The minimum and maximum flows belonging to project g'' transferred from activity i of project g to activity j of project g', respectively
- $S_{i,g}$ The start time of activity i belonging to project g

4.2. The first-stage model

The following is the mathematical formulation of the first stage model:

$$\min TTa^* = \sum_{g=1}^{G} w_g.Ta_g, \tag{1}$$

s.t.

$$Ta_g \geq S_{n+1,g}^h - DD_g, \qquad \forall g \in G, h = 1, ..., |P|, \tag{2}$$

$$S_{j,g'}^h \geq S_{i,g}^h + P_{i,g}^h - M(1 - x_{igjg'}),$$
$$\forall (i,j) \in V \times V, \ \ \forall g, g' \in G \times G, \quad i \neq j \ \text{ or }$$
$$g \neq g', \ \ h = 1, ..., |P|, \tag{3}$$

$$\sum_{g'} \sum_{\substack{i \in V, \\ i \neq 0}} \sum_{g} f_{0gig'k} = b_k, \qquad \forall k \in R, \tag{4}$$

$$\sum_{g} \sum_{\substack{j \in V, \\ j \neq n+1}} \sum_{g'} f_{jgn+1g'k} = b_k, \qquad \forall k \in R, \tag{5}$$

$$\sum_{g' \in G} \sum_{\substack{j \in V, j \neq n+1 \\ (j \neq i \ or \ g \neq g')}} f_{jg'igk} = b_{igk},$$
$$\forall i \in V \backslash \{0, n+1\}, \quad \forall k \in R, \quad \forall g \in G, \tag{6}$$

$$\sum_{g' \in G} \sum_{\substack{j \in V, j \neq 0 \\ (j \neq i \ or \ g \neq g')}} f_{igjg'k} = b_{igk},$$
$$\forall i \in V \backslash \{0, n+1\}, \quad \forall k \in R, \quad \forall g \in G, \tag{7}$$

$$f_{igjg'k} \leq \min\{b_{igk}, b_{jg'k}\}.x_{igjg'},$$
$$\forall (i,j) \in V \times V, \quad \forall (g,g') \in G \times G, \quad \forall k \in R,$$
$$i,j \neq 0, n+1, \tag{8}$$

$$x_{igjg'} = 1, \qquad \forall (i,g,j,g') \in E, \tag{9}$$

$$S_{0g} = 0, \qquad \forall g \in G, \tag{10}$$

$$Ta_g \geq 0, \qquad \forall g \in G, \tag{11}$$

$$S^h_{i,g} \geq 0, \qquad \forall i \in V, \ \forall g \in G, \quad h = 1, ..., |P|, \tag{12}$$

$$f_{igjg'k} \geq 0, \qquad \forall (i,j) \in V \times V,$$
$$\forall (g,g') \in G \times G, \ \forall k \in R, \tag{13}$$

$$x_{igjg'} \in \{0,1\}, \quad \forall (i,j) \in V \times V, \quad \forall g, g' \in G \times G. \tag{14}$$

The minimization of the total weighted tardiness of the projects is displayed in Eq. (1) as the objective function. The tardiness of each project is the difference between the project's makespan and its determined due date and is obtained by Constraint (2). Constraint (3) demonstrates the precedence relationships between the activities, where M is a big number. Therefore, based on this constraint, the successor activity, j, cannot start earlier than the finish time of its predecessors under each scenario. The sum of resource flows (type k) sent from dummy start nodes 0 is equal to the available capacity of resource (type k), as mentioned in Eq. (4). In addition, based on Eq. (5), the sum of resource flows type k sent from the activities of all projects to the dummy finish nodes $n+1$ of projects is equal to the available capacity of resource type k.

The sum of incoming resource flows (type k) from other activities to activity i of project g is equal to the required resource type k for performing the activity i of project g, which is described in Eq. (6). Similarly, Eq. (7) ensures that the sum of resource flows (type k) exiting from activity i of project g to other activities is equal to the required resource type k for executing the activity i of project g. Constraint (8) ensures that the resource flow (type k) transferred from the activity i of project g to the activity j of project g' is quite equal to the minimum value of $\{b_{igk}, b_{jg'k}\}$. In addition, this equation prevents resource transferring between two activities, where there is no precedence relationship between them.

According to Eq. (9), the binary variable, x, is equal to 1 for the two activities with precedence relationship between them. The start time of (dummy) activities 0 for all projects is zero (the start point of the scheduling horizon), as demonstrated in Eq. (10). Based on Constraint (11), the tardiness of projects cannot be negative. Constraints (12) and (13) introduce the nonnegative decision variables of the start time of activities and the resource flow between the activities, respectively. At last, the binary variable, x, is presented in Eq. (14).

In this stage, the best structure $E \cup X$ is obtained for the existing scenarios regarding the precedence relationships and resource requirements. This structure is the output of the first-stage model, which is required as an input for the second-stage model. In fact, this structure is achieved while the total weighted tardiness of projects as an objective function is minimized.

4.3. The second-stage model

In this section the mathematical formulation for the second stage model is presented:

$$\max TTa^* = \sum_{g=1}^{G} w_g.Ta_g, \tag{15}$$

s.t.

$$Ta_g \leq (LP_g - DD_g), \qquad \forall g \in G, \tag{16}$$

$$LP_{g''} = \sum_{(i,g,j,g') \in EUX} (p^{\min}_{ig}.\varphi^{\min}_{g''igjg'} + p^{\max}_{ig}.\varphi^{\max}_{g''igjg'}),$$
$$\forall g'' \in G, \tag{17}$$

$$\sum_{(i,g,j,g') \in EUX} \varphi^{\max}_{g''igjg'} \leq a_{ig},$$
$$\forall i \in V \setminus \{0, n+1\}, \ \ \forall g \in G, \ \ \forall g'' \in G, \tag{18}$$

$$\sum_{(i,g,j,g') \in EUX} \varphi^{\min}_{g''igjg'} \leq 1 - a_{ig},$$
$$\forall i \in V \setminus \{0, n+1\}, \ \ \forall g \in G, \ \ \forall g'' \in G, \tag{19}$$

$$\sum_{(i,g,j,g') \in EUX} (\varphi^{\min}_{g''igjg'} + \varphi^{\max}_{g''igjg'}) = 1,$$
$$\text{for} \ \ i = 0, \ \ \forall g'' \in G, \tag{20}$$

$$\sum_{(i,g,j,g') \in EUX} (\varphi^{\min}_{g''igjg'} + \varphi^{\max}_{g''igjg'}) = 1,$$
$$\forall g'' \in G, \ \ j = n+1, \ \ g' = g'', \tag{21}$$

$$\varphi^{\min}_{g''igjg'} = 0, \qquad \forall i \in V, \ \ \forall g, g'' \in G \times G,$$
$$j = n+1, \ \ g' \neq g'', \tag{22}$$

$$\varphi^{\max}_{g''igjg'} = 0, \qquad \forall i \in V, \ \ \forall g, g'' \in G \times G,$$
$$j = n+1, \ \ g' \neq g'', \tag{23}$$

$$\sum_{(i,g,j,g')\in EUX} \varphi^{\min}_{g''igjg'} + \varphi^{\max}_{g''igjg'}$$

$$= \sum_{(j,g',i,g)\in EUX} \varphi^{\min}_{g''jg'ig} + \varphi^{\max}_{g''jg'ig},$$

$$\forall i \in V \setminus \{0, n+1\}, \quad \forall g \in G, \quad \forall g'' \in G, \quad (24)$$

$$Ta_g \geq 0, \qquad \forall g \in G, \quad (25)$$

$$\varphi^{\min}_{g''igjg'} \geq 0, \quad \forall (i,g,j,g') \in EUX, \quad \forall g'' \in G, \quad (26)$$

$$\varphi^{\max}_{g''igjg'} \geq 0, \quad \forall (i,g,j,g') \in EUX, \quad \forall g'' \in G, \quad (27)$$

$$x_{igjg'} \in \{0,1\}, \quad \forall (i,j) \in V \times V, \quad \forall g, g' \in G \times G, \quad (28)$$

$$x_{igjg'} = 1, \qquad \forall (i,g,j,g') \in E, \quad (29)$$

$$a_{ig} \in \{0,1\}, \qquad \forall i \in V, \quad \forall g \in G, \quad (30)$$

$$a_{0g} = a_{n+1g} = 0, \qquad \forall g \in G, \quad (31)$$

$$S_{0g} = 0, \qquad \forall g \in G. \quad (32)$$

In the second-stage model, the worst-case scenario should be found in such a way that the total weighted tardiness of the projects is maximized, as represented in Eq. (15). Eq. (16) shows how to obtain the projects' tardiness. In this equation, the finish time of each project is obtained by the longest path (LP_g) method in the overall network of the projects, as demonstrated in Eq. (17). In the single project problem, the longest path can be obtained by $\sum_{(i,j)\in EUX} (p_i.\varphi_{ij})$, where p_i is the duration of activity i and φ_{ij} is the flow transferred from activity i to activity j. The multiplication of p_i and φ_{ij} leads to the nonlinearity of this formula. The binary variable, a_i, is introduced to linearize the formula and is converted it to $\sum_{(i,j)\in EUX} (p_i^{\min}.\varphi_{ij}^{\min} + p_i^{\max}.\varphi_{ij}^{\max})$, in which $p_i^{\min}$ and $p_i^{\max}$ are the minimum and maximum values of the duration belonging to activity i, respectively. For detailed information about calculating the longest path of "single project" and how to linearize it, please refer to Artigues et al. [36].

In the multi-project scheduling problem, the EUX is the overall structure of all projects including the primary precedence relationships between activities (E) and the extra precedence relationships caused by resource constraint (X). Therefore, in the studied problem, the projects are interrelated with each other in this structure. Thus, for obtaining the longest paths of the projects, a flow per project should be sent from 0 activities to other activities in the overall structure, as demonstrated by $\varphi_{g''igjg'}$. It is worth mentioning that the first index (g'') in the decision variable $\varphi_{g''igjg'}$ shows the project for which we want to calculate the longest path.

In order to linearize the longest path formula, Constraints (18) and (19) are created in which the binary variable, a_{ig}, takes the value 0 when the duration value of activity i of project g is minimum and, thus, $\varphi^{\max}_{g''igjg'} = 0$. On the other hand, a_{ig} takes value 1, showing that the duration value of activity i of project g is maximum and, thus, $\varphi^{\min}_{g''igjg'} = 0$.

As mentioned before, the predecessor of one activity can be the activity within the same project or from the other projects. Therefore, the longest path of one project does not necessarily originate from the activity 0 of that project, and it can also start from the 0 activity of other projects. According to Eq. (20), the summation of flows by calculating the longest path of the project g'', originated from start nodes 0 of all projects, to the overall structure should be equal to 1. Besides, Eq. (21) implies that the flow calculating the longest path of project g'' should end in the node $n+1$ of project g''. Eqs. (22) and (23) ensure that the flow calculating the longest path of one project cannot enter the end node $n+1$ of other projects. For each flow, the conservation law should be satisfied, i.e., the sum of flows entering the activity i of project g should be equal to the sum of flows exiting from the activity i of project g. This law is presented in Eq. (24).

Constraint (25) introduces the nonnegative variable of the projects' tardiness. The flows related to the longest path calculations are represented in Eqs. (26) and (27). The binary variable, x, is defined in Eq. (28), while it should take value 1 for the activities with precedence relationship between them, as stated in Eq. (29). The binary variable, a, is described in Eq. (30). Eq. (31) represents that, for all start nodes 0, the variable a takes value 0. Finally, the start time of the projects is set at time 0, as shown in Eq. (32).

4.4. The two-stage exact approach

The scenario relaxation algorithm is an iterative optimization algorithm that generates optimal robust decisions with respect to the deviation and relative robust objectives. The key insight of the scenario relaxation algorithm is that, for a problem with a large number of possible scenarios, only a small subset of scenarios actually has to be explicitly examined when searching for the deviations from the optimal (or relative) robust solution. For more information about the scenario relaxation algorithm, please refer to [38].

In this paper, the objective function is to minimize the maximum total weighted tardiness of the multi-project problem under uncertain activities' durations. A two-stage model is presented for the RRCMPSP in Sections 4.1 and 4.2. Based on the mentioned modeling, the set of decision variables can be divided into two groups. The first-group variables are those related to the sufficient selection decisions $X \in \chi$. The second group variables are related to the calculation of the longest paths in the structure obtained by the first-stage model. According to the Benders alphabet, the first-stage model corresponds to the "Master problem", and the second-stage model is similar in spirit to the "Sub problem".

According to Artigues et al. [36], a duration scenario p is extreme if $p_i = p_i^{\min}$ or $p_i = p_i^{\max}$ for all $i \in \nu$. They also proved that there is always an extreme duration scenario for which the maximum absolute regret of an ES-policy X is reached. Therefore, in the worst case for the studied problem in this paper, the number of algorithm iterations can be $|P| = 2^v$, where v is the number of activities belonging to all projects.

In this approach, the scenarios are gradually added to the problem structure in the sequential iterations. First, one scenario of activity durations is considered (any arbitrary number of scenarios can be considered), and the first-stage model is solved. The aim is to obtain the structure $E \cup X$, for which the total weighted differences between the projects' makespan and their due dates are minimized. In other words, considering the existing scenario, we search for an optimized $E \cup X$ with minimum total weighted tardiness of the projects. In the next step, the second-stage model is the worst-case scenario for the obtained structure of the first-stage model such that the objective function (total weighted tardiness of the projects) will be maximized. Then, the mentioned scenario should be added to the scenario set of the first-stage model. This algorithm continues until the objective functions of the both stages become equal. In other words, the algorithm terminates when the minimum weighted tardiness of the optimized structure for the existing scenarios is equal to the maximum weighted tardiness of the worst-case scenario for the assigned structure.

The steps of the applied approach are described in the following, where iter is the counter of algorithm iterations:

Step 1 (preliminary). The set $\hat{P}_1$ containing only one scenario p^1 for the duration of all activities of the projects is considered. In addition, $iter = 1$, $LB = 0$, and $UB = +\infty$ are assumed;

Step 2 (first-stage model). Models (1)–(14) are solved in order to obtain $LB = TTa^*(\hat{P}_{iter})$. In addition, the corresponding ES-policy, X_{iter}, is obtained;

Step 3 (second-stage model). Models (15)–(32) are solved and the maximum $TTa^{\max}(X_{iter})$ for X_{iter} is obtained. The corresponding worst-case scenario, p^{iter+1}, is obtained. In addition, the $UB = TTa^{\max}(X_{iter})$ is considered;

Step 4 (optimality investigation). If $LB = UB$, then stop the algorithm. If $LB \neq UB$, then $iter = iter + 1$, $\hat{P}_{iter} = \hat{P}_{iter-1} \cup \{p^{iter}\}$ and the algorithm should continue from Step 2.

5. Numerical example

In this section, one simple example is presented to illustrate the application of the mentioned approach to the multi-project problem. Consider a multi-project problem that consists of three projects. Each project has only four activities (the start activities and end activities are dummies), as shown in Figure 2. There is only one renewable resource with ($b_1 = 7$). The required resource for performing each activity, the possible durations of activities, the determined due date of projects, and the importance weight of the projects are all represented in Table 1. Both of the first-stage and second-stage models are coded in GAMS v24.1.2 and solved by the "CPLEX" solver.

The EUX1 is obtained after solving the first-stage model in the first iteration. According to this structure, the total weighted tardiness of the projects according to the first scenario will be minimized. In the first scenario, the durations of all activities are considered at their minimum values (Table 2). To avoid untidiness caused by too many arcs, the representation of the whole EUXs is neglected in each iteration. The longest paths of the projects according to the first scenario are calculated and shown in Figure 3(a).

For the given EUX1 from the first-stage model,

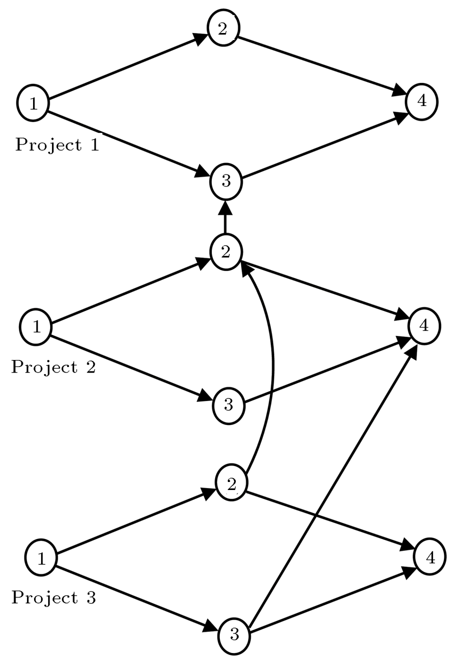

Figure 2. Multi-project network.

Table 1. The multi-project required data.

	i	g	b_{ig1}	p_{ig}	Due date	Project weight
Project 1	1	1	0	$\{0\}$	7	0.3
	2	1	4	$\{4,5,6\}$		
	3	1	3	$\{2,4\}$		
	4	1	0	$\{0\}$		
Project 2	1	2	0	$\{0\}$	4	0.4
	2	2	3	$\{3,4\}$		
	3	2	5	$\{1,2,3\}$		
	4	2	0	$\{0\}$		
Project 3	1	3	0	$\{0\}$	4	0.3
	2	3	2	$\{2,3,5\}$		
	3	3	4	$\{4,5,6\}$		
	4	3	0	$\{0\}$		

Table 2. First scenario values for activities' durations.

	Project		
Activity	**1**	**2**	**3**
1	0	0	0
2	4	3	2
3	2	1	4
4	0	0	0

the second-stage model should be solved. The maximum total weighted tardiness for EUX1 is determined by finding the worst-case scenario, as presented in Table 3.

Figure 3(b) shows the longest paths of the projects according to the worst-case scenario (the demonstration of the longest paths based on the first scenario is ignored) resulting from the second-stage model in the first iteration.

The first-stage model should be solved regarding two scenarios for the activities' durations in the second iteration. The total weighted tardiness of the projects should be minimized with regard to these two scenarios. Therefore, the optimized EUX2 structure is obtained, which will be the input for the second-stage model. The longest paths of the projects only for the second scenario are depicted in Figure 4(a).

After that, the second stage model is solved while the objective is to maximize the total weighted tardiness of projects. In fact, for the given EUX2, the worst-case scenario should be achieved, which is demonstrated in Table 4. The longest paths of the

Table 3. The worst-case scenario obtained by the second-stage model in the first iteration.

	Project		
Activity	**1**	**2**	**3**
1	0	0	0
2	6	3	2
3	2	3	6
4	0	0	0

Table 4. The worst-case scenario obtained by the second-stage model in the second iteration.

	Project		
Activity	**1**	**2**	**3**
1	0	0	0
2	6	4	5
3	2	3	6
4	0	0	0

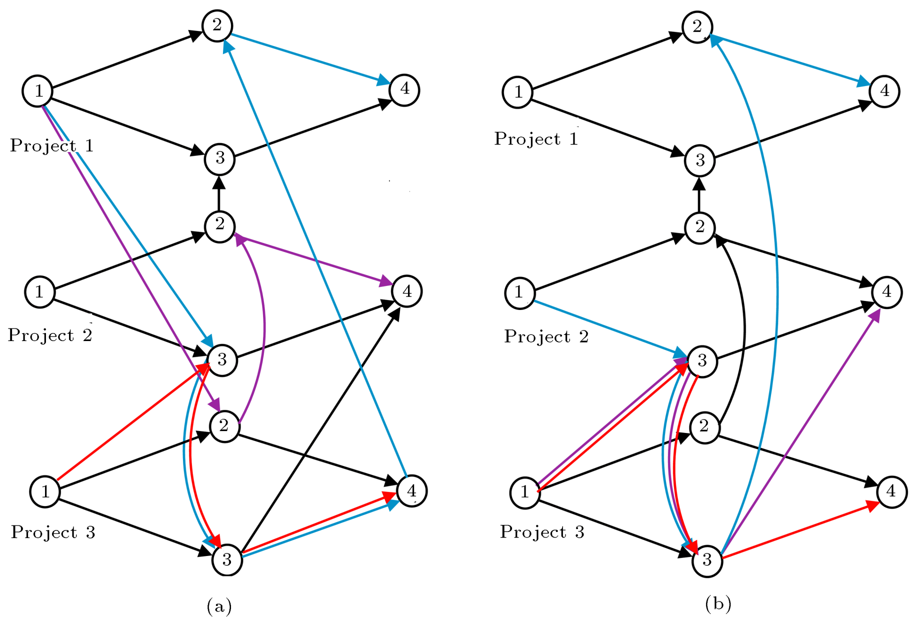

Figure 3. (a) First-iteration/level 1. Obj: 1.3, Ta_1: 2, Ta_2: 1, Ta_3: 1. (b) First-iteration/level 2. Obj: 5.9, Ta_1: 8, Ta_2: 5, Ta_3: 5, $a_{ig}=1$ for $\{2.1, 3.2, 3.3\}$.

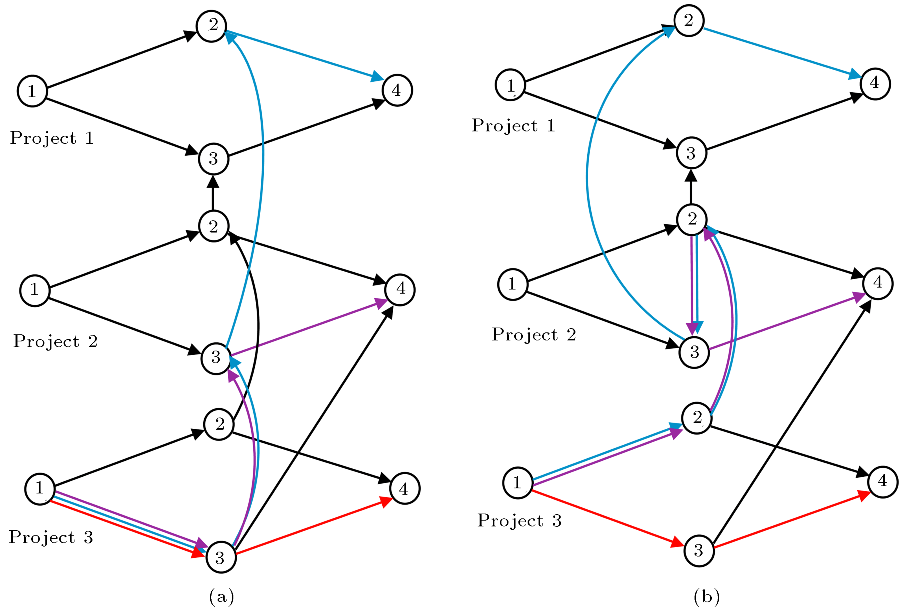

Figure 4. (a) Second-iteration/level 1 (second scenario): Obj: 5, Ta_1: 8, Ta_2: 5, Ta_3: 2. (b) Second-iteration/level 2: Obj: 7.1, Ta_1: 11, Ta_2: 8, Ta_3: 2, $a_{ig} = 1$ for $\{2.1, 2.2, 2.3, 3.2, 3.3\}$.

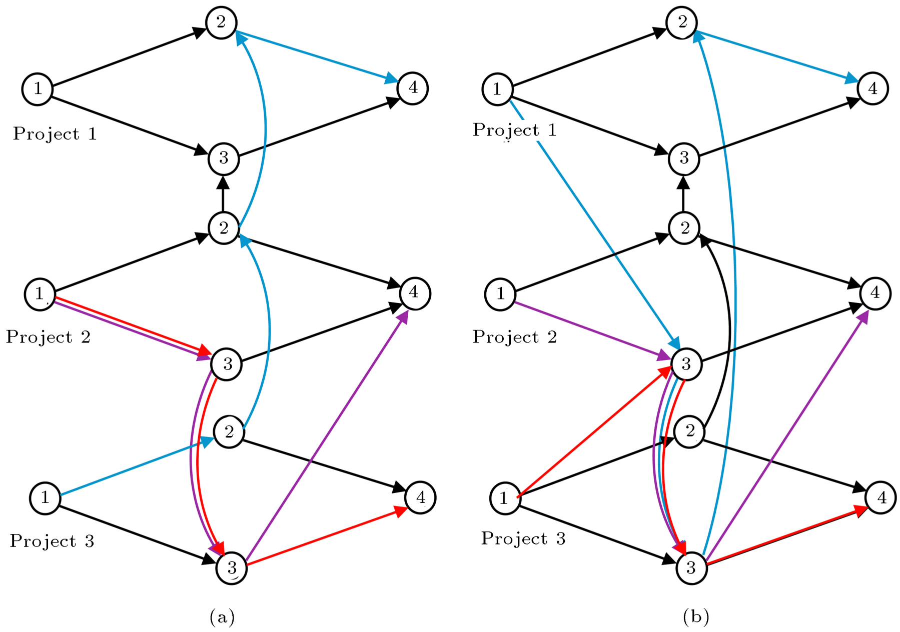

Figure 5. (a) Third-iteration/level 1 (third scenario): Obj: 5.9, Ta_1: 8, Ta_2: 5, Ta_3: 5. (b) Third-iteration/level 2: Obj: 5.9, Ta_1: 8, Ta_2: 5, Ta_3: 5, $a_{ig} = 1$ for $\{2.1, 3.2, 3.3\}$.

projects based on only the worst-case scenario are shown in Figure 4(b).

In the third iteration, the first-stage model is solved regarding three scenarios. The longest paths of projects only for the third scenario are shown in Figure 5(a).

By obtaining the EUX3, the second-stage model can be solved. The obtained worst-case scenario and the longest paths of the projects are depicted in Table 5 and Figure 5(b), respectively.

By comparing the objective functions of the first-stage and second-stage models in the third iteration, it is realized that the algorithm should stop when both of the objective functions have the same value, i.e. 5.9 in this simple example. The results of each iteration are presented in Table 6 in brief.

After three iterations, the optimized value of the objective function of this example has been obtained. This value is found by the best structure according to the resource-constraint and precedence relationships in

Table 5. The worst-case scenario obtained by the second-stage model in the third iteration.

Activity	Project 1	2	3
1	0	0	0
2	6	3	2
3	2	3	6
4	0	0	0

which the total weighted tardiness of the projects is minimum. With regard to the existing uncertainty in the activities' durations, the obtained result is robust. In fact, it ensures that if any scenarios occur for the activities' durations (in this example, we have $(3^4) \times (2^2) = 324$ possible scenarios occurrence), the total weighted tardiness of the projects will not be greater than 5.9. This is exactly the characteristic of the robust optimization method that immunizes the problem against uncertainty and keeps the results feasible and almost optimal.

6. Computational experiments

Both of the first-stage and second-stage models are coded in GAMS v24.1.2 and solved with the CPLEX solver. The experiments were run on a personal computer with an Intel(R) Xeon(R) CPU E7-8890 v4@ 2.20 GHz 2.19 GHz (2 processors) and 42 GB RAM under Windows 10 operation system.

6.1. The test problems

In this paper, in order to generate the test problems, the software RanGen [39] is applied to deterministic RCPSP. To adapt the test problems to RRCMPSP, the required additional data are considered. In addition, the number of activities can be chosen. In this research, the number of activities, $n = 30$, is considered for each project in the multi-project problem. The application of this software provides us instances with different values of the parameters related to the structure of the projects. The considered parameters include order strength, resource factor, and resource constrainedness, which are explained briefly in the following:

- **Order Strength (OS):** The number of precedence relations is divided by the theoretical maximum number of precedence relations in the network. The minimum value of OS is 0 (in the parallel network), and the maximum value of OS is 1 (in the serial network case). Therefore, it can take values from 0 to 1. In this research, OS can be chosen from two values $\{0.4, 0.7\}$;
- **Resource Factor (RF):** How many different resources used on average by the activities are determined by this factor. The minimum value of RF is 0 (no resource requirements for executing the activities), and the maximum value of RF is 1 (when all the activities require all kinds of resources). Therefore, it can take values from 0 to 1. In this research, RF chooses a value from the set $\{0.25, 0.5, 0.75\}$;
- **Resource Constrainedness (RC) (per resource type):** This factor can be obtained through Eq. (33) [40]:

$$RC_k = \frac{DMND_k}{R_k}, \quad \text{for all } k \in R, \tag{33}$$

where R_k is the capacity of resource type k, and $DMND_k$ is the average quantity of resource type k demanded when required by an activity and can be calculated by Eq. (34):

$$DMND_k = \frac{\sum_N r_{ijk}}{\sum_N \left\{ \begin{array}{l} 1 \; if \; r_{ijk} \succ 0 \\ 0 \; if \; r_{ijk} = 0 \end{array} \right\}}, \text{ for all } k \in R, \tag{34}$$

where r_{ijk} is the per-period requirement of resource type k by activity j of project i, and N is the set of all activities to be scheduled. In this research, RC chooses a value from the set $\{0.3, 0.6\}$. For each combination of OS, RF, and RC, five instances of the RRCMPSP are

Table 6. The summary of the results obtained from iterations.

		First iteration Projects 1	2	3	Second iteration Projects 1	2	3	Third iteration Projects 1	2	3
First stage model	Projects tardiness	2	1	1	8	5	2	8	5	5
	The total weighted tardiness		1.3			5			5.9	
Second stage model	Projects tardiness	8	5	5	11	8	2	8	5	5
	The total weighted tardiness		5.9			7.1			5.9	
Optimality check			×			×			√	

considered. Each multi-project problem is assumed to be containing three projects. Thus, $2(OS) \times 3(RF) \times 2(RC) \times 5$(per combination examples) $\times$ 3(number of project in each multi project problem) = 180 instances are needed to be randomly generated by *RanGen*.

6.2. Computational experiments

There are 12 classes of problems with respect to different values of factors: OS, RF, and RC. The average execution time per iteration for both of the first-stage and second-stage models is calculated. In addition, the average number of iterations is recorded. Table 7 represents the computational results of the test problems.

Figures 6 to 8 illustrate the sensitivity analysis of the obtained results for different levels of factors RC, RF, and OS, respectively. As is shown by these figures, the behavior of the solution approach is strongly related to the instances and their characteristics.

There are 6 classes of problems based on the different values of OS and RF, in which the effect of factor RC should be examined. As shown in Figure 6, the most effective factor is RC, which strongly impacts on the performance of the applied approach. It is notable that the linear histogram is fitted just for showing the effect of the factors' value on the results schematically. According to these 6 experiments, the computational time grows rapidly according to the higher value of RC. In other words, when the value of RC increases, the instances become harder to solve and the approach needs requires time for execution. Of note, this impact is mainly observed on the performance of the first-stage model, in which extra precedence relationships (X) should be obtained (according to the resource constraint). Therefore, RC,

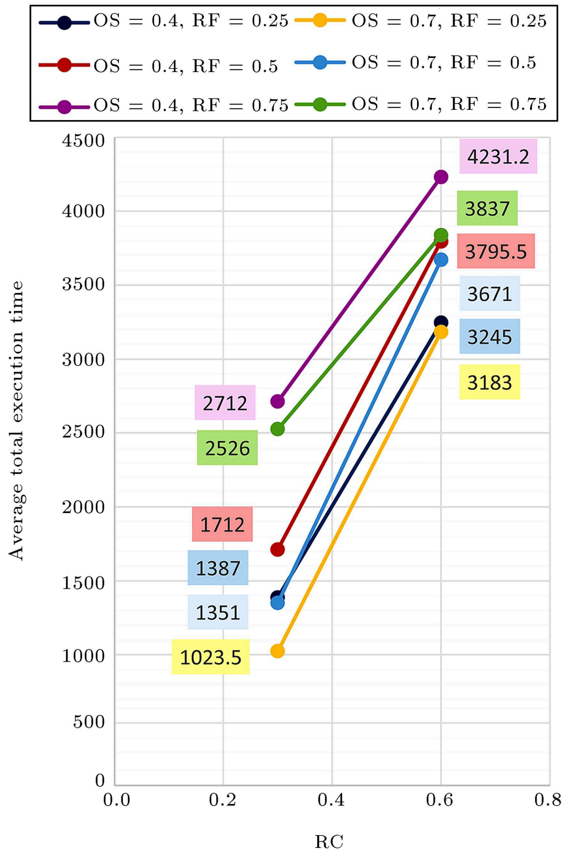

Figure 6. The sensitivity analysis of Resource Constrainedness (RC).

Table 7. The computational results.

	Parameters			Time			
	OS	RF	RC	First-stage average computational time per iteration	Second-stage average computational time per iteration	Average total computational time per iteration	Average number of iterations
1	0.4	0.25	0.3	23 min, 5 s	2 s	23 min, 7 s	3.2
2			0.6	54 min, 2 s	3 s	54 min, 5 s	5.6
3		0.5	0.3	28 min, 30 s	2 s	28 min, 32 s	4
4			0.6	1 h, 3 min, 13 s	2.5 s	1 h, 3 min, 15.5 s	4.8
5		0.75	0.3	45 min, 10 s	2 s	45 min, 12 s	4.2
6			0.6	1 h, 10 min, 28 s	3.2 s	1 h, 10 min, 31.2 s	6
7	0.7	0.25	0.3	17 min, 1.5 s	2 s	17 min, 3.5 s	3.2
8			0.6	53 min	3 s	53 min, 3 s	2.2
9		0.5	0.3	22 min, 28.3 s	2.7 s	22 min, 31 s	2
10			0.6	1 h, 1 min, 8 s	3 s	1 h, 1 min, 11 s	5.4
11		0.75	0.3	42 min, 2.5 s	3.5 s	42 min, 6 s	3
12			0.6	1 h, 3 min, 54.4 s	2.6 s	1 h, 3 min, 57 s	6.2

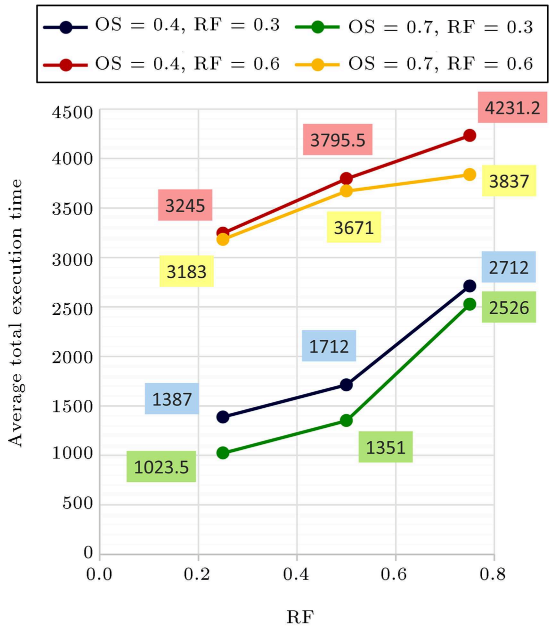

Figure 7. The sensitivity analysis of Resource Factor (RF).

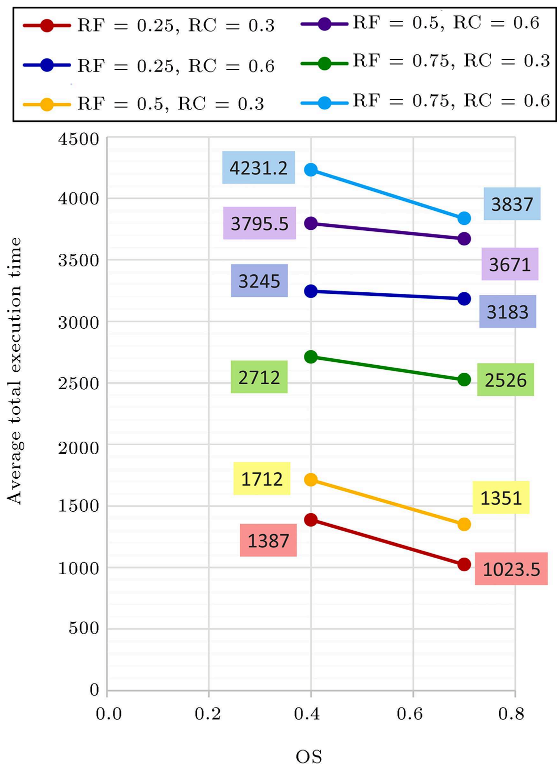

Figure 8. The sensitivity analysis of Order Strength (OS).

which is mainly related to the resource constraint, has strong influence on the performance of the first-stage model, and the second-stage model is not influenced significantly by this factor.

The next effective factor is RF that is also related to the resource constraint and has an observable influence on the first-stage model. When RF increases, the first-stage model becomes harder to solve and, consequently, consumes more time. Therefore, RF is the second effective factor that influences the obtained results. As demonstrated by Figure 7, a sensible increase in the computational time occurs by increasing the RF.

The OS factor has the least influence on the computational time of the obtained results, as depicted in Figure 8. In most cases, there is a decrease in the computational time when the value of factor OS changes (in some cases, there are not significant changes). With respect to the definition of OS factor, when OS increases, the structure of the projects moves from the parallel structure to the serial structure. Therefore, the problem becomes easier according to the resource constraints, and it is expected that the running time of the algorithm decreases significantly. However, why does it not happen? The reason is related to the extra precedence relationship X that is added to the E set to remove the resource conflict. In other words, the E set (based on the OS factor) is not the only parameter that affects the physical structure of the project network, and EUX is the final structure of the network.

7. Conclusion and further research

The Robust Resource-Constrained Multi-Project Scheduling Problem (RRCMPSP) was studied in this paper, in which the objective function was to minimize the maximum total weighted tardiness of the projects. The durations of the activities belonging to the projects were uncertain and defined with discrete values called scenarios. The resource sharing policy was applied in this study for resource allocation in the multi-project problem. Moreover, there was a deadline for each project determined by the global project manager, and each project had its own weight of importance, which dictates which project should receive more consideration. To ensure exact results, a scenario-relaxation algorithm was applied and implemented for the proposed robust multi-project scheduling problem. Then, the computational results were discussed. It was found obviously that the factor RC had more influence on the behavior of the solution approach, which is an important factor, especially in the multi-project problems.

Some extensions of this research as a future study might be of interest. While the limitation of this study is that the presented exact solution method is not able to solve large-size problems in a reasonable amount of time, developing the heuristic and meta-heuristic algorithms is suggested to solve the large-sized RRCMPSP. As another extension, considering uncertainty in the resources availability and its effect in managing the multi-project problems would be of interest. In addition, some constraints can be added to this model such as the multi-mode activities, nonrenewable resources, multi-skill resources, etc. while the other objective functions such as minimum cost or maximum quality are considered.

References

1. Ma, W., Che, Y., Huang, H., and Ke, H. "Resource-constrained project scheduling problem with uncertain durations and renewable resources", *International Journal of Machine Learning and Cybernetics*, **7**(4), pp. 613–621 (2016).

2. Hindelang, T.J. and Muth, J.F. "A dynamic programming algorithm for decision CPM networks", *Operations Research*, **27**(2), pp. 225–241 (1979).

3. Kruger, D. and Scholl, A. "A heuristic solution framework for the resource constrained (multi-)project scheduling problem with sequence-dependent transfer times", *European Journal of Operational Research*, **197**, pp. 492–508 (2009).

4. Besikci, U., Bilge, U., and Ulusoy, G. "Resource dedication problem in a multi-project environment", *Flexible Services and Manufacturing Journal*, **25**, pp. 206–229 (2013).

5. Payne, J.H. "Management of multiple simultaneous projects: A state-of-the-art review", *International Journal of Project Management,* **13**(3), pp. 163–168 (1995).

6. Aritua, B., Smith, N.J., and Bower, D. "Construction client multi-projects - A complex adaptive systems perspective", *International Journal of Project Management*, **27**, pp. 72–79 (2009).

7. Bruni, M.E., Puglia Pugliese, L., Di Beraldi, P., and Guerriero, F. "An adjustable robust optimization model for the resource-constrained project scheduling problem with uncertain activity durations", *Omega*, **71**, pp. 66–84 (2017).

8. Bertsimas, D. and Sim, M. "The price of robustness", *Operations Research*, **52**(1), pp. 35–53 (2004).

9. Herroelen, W. and Leus, R. "Project scheduling under uncertainty: Survey and research potentials", *European Journal of Operational Research*, **165**, pp. 289–306 (2005).

10. Li, Z. and Floudas, C.A. "Robust counterpart optimization: Uncertainty sets, formulations and probabilistic guarantees", Paper presented at the proceedings of the 6th conference on foundations of computer-aided process operations, Savannah (Georgia) (2012). http://focapo.cheme.cmu.edu/2012/proceedings/data/papers/030.pdf

11. Yamashita, D.S., Armentano, V.A., and Laguna, M. "Robust optimization models for project scheduling with resource availability cost", *Journal of Scheduling*, **10**, pp. 67–76 (2007).

12. Zheng, Z., Shumin, L., Ze, G., and Yueni, Z. "Resource constraint multi project scheduling with priority and uncertain activity durations", *International Journal of Computational Intelligence Systems*, **6**(3), pp. 530–547 (2013).

13. Kerzner, H., *Project Management: A Systems Approach to Planning, Scheduling, and Controlling*, Ed. 11th. pp. 1, 7, 42, 66, Hoboken: Wiley (2013).

14. Blismass, N., Sher, W.D., Thorpe, A., and Baldwin, A.N. "Factors influencing delivery within construction clients' multi-project environments", *Engineering, Construction and Architectural Management,* **11**(2), pp. 113–125 (2004a).

15. Wei-xin, W., Xu, W., Xian-long, G., and Lei, D. "Multi-objective optimization model for multi-project scheduling on critical chain", *Advances in Engineering Software*, **68**, pp. 33–39 (2014).

16. Wauters, T., Kinable, J., Smet, P., Vancroonenburg, W., Berghe, G., and Verstichel, J. "The multi-mode resource-constrained multi-project scheduling problem", *Journal of Scheduling*, **19**(3), pp. 271–283 (2016).

17. Besikci, U., Bilge, U., and Ulusoy, G. "Multi-mode resource constrained multi-project scheduling and resource portfolio problem", *European Journal of Operational Research*, **240**, pp. 22–31 (2015).

18. Zheng, Z., Shumin, L., Ze, G., Yueni, Z., and Zhang, X. "A critical chains-based distributed multi-project scheduling approach", *Neurocomputing*, **143**, pp. 282–293 (2014).

19. Singh, A. "Resource constrained multi-project scheduling with priority rules & analytic hierarchy process", *Procedia Engineering*, **69**, pp. 725–734 (2014).

20. Vázquez, E.P., Calvo, M.P., and Ordóñez, P.M. "Learning process on priority rules to solve the RCMPSP", *Journal of Intelligent Manufacturing*, **26**(1), pp. 123–138 (2015).

21. Wang, Y., He, Z., Kerkhove, L., and Vanhoucke, M. "On the performance of priority rules for the stochastic resource constrained multi-project scheduling problem", *Computers & Industrial Engineering*, **114**, pp. 223–234 (2017).

22. Uysal, F. "Hybrid mate-heuristic algorithms for the resource constrained multi-project scheduling problem", PhD Thesis, Graduate School of Natural and Applied Sciences of Middle East Technical University (2014).

23. Nabipoor Afruzi, E., Roghanian, E., Najafi, A.A., and Mazinani, M. "A multi-mode resource-constrained discrete time-cost tradeoff problem solving using an adjusted fuzzy dominance genetic algorithm", *Scientia Iranica*, **20**(3), pp. 931–944 (2014).
24. Hazir, O., Erel, E., and Gnalay, Y. "Robust optimization models for the discrete time/cost trade-off problem", *International Journal of Production Economics*, **130**(1), pp. 87–95 (2011).
25. Moghani Ghahremanlouie, S. and Fathi Hafashjani, K. "A novel robust model for discrete time-cost trade off problem", *International Journal of Industrial Engineering & Production Management*, **25**(1), pp. 1–14 (2014).
26. Rostami, S., Creemers, S., and Leus, R. "A new policy class for the stochastic RCPSP", *International Conference on Project Management and Scheduling (PMS)*, At Valencia, Spain, p. 15 (2016).
27. Tao, S., Wu, C., Sheng, Z., and Wang, X. "Stochastic project scheduling with hierarchical alternatives", *Applied Mathematical Modelling*, **58**, pp. 181–202 (2018).
28. Chiang, A.J. and Jeang, A. "Stochastic project management via computer simulation and optimisation method", *International Journal of Systems Science: Operations & Logistics*, **2**(4), pp. 211–230 (2015).
29. Ning, M., He, Z., Jia, T., and Wang, N. "Metaheuristics for multi-mode cash flow balanced project scheduling with stochastic duration of activities", *Automation in Construction*, **81**, pp. 224–233 (2017).
30. Chtourou, H. and Haouari, M. "A two-stage-priority-rule-based algorithm for robust resource-constrained project scheduling", *Computers and Industrial Engineering*, **55**(1), pp. 183–194 (2008).
31. Khalilzadeh, M., Shakeri, H., Gholami, H., and Amini, L. "A heuristic algorithm for project scheduling with fuzzy parameters", *Procedia Computer Science*, **121**, pp. 63–71 (2017).
32. Knyazeva, M., Bozhenyuk, B., and Rozenberg, I. "Resource-constrained project scheduling approach under fuzzy conditions", *Procedia Computer Science*, **77**, pp. 56–64 (2015).
33. Dixit, V., Srivastava, R.K., and Chaudhuri, A. "Procurement scheduling for complex projects with fuzzy activity durations and lead times", *Computers & Industrial Engineering*, **76**, pp. 401–414 (2014).
34. Huang, M., Yuan, J., and Xiao, J. "An adapted firefly algorithm for product development project scheduling with fuzzy activity duration", *Mathematical Problems in Engineering*, **2015**, Article ID 973291, 11 pages (2015).
35. Chakrabortty, R.K., Sarker, R.A., and Essam, D.L. "Resource constrained project scheduling with uncertain activity durations", *Computers & Industrial Engineering*, **112**, pp. 537–550 (2017).
36. Artigues, C., Leus, R., and Nobibon, F.T. "Robust optimization for resource-constrained project scheduling with uncertain activity durations", *Flexible Services and Manufacturing Journal*, **25**, pp. 175–205 (2013).
37. Balas E. "Project scheduling with resource constraints", In E.M.L. Beale, Ed., *Applications of Mathematical Programming Techniques*, pp. 187–200, The English Universities Press Ltd (1971).
38. Assavapokee, T., Realff, M.J., Ammons, J.C., and Hong, I-H. "Scenario relaxation algorithm for finite scenario-based min-max regret and min-max relative regret robust optimization", *Computers & Operations Research*, **35**, pp. 2093–2102 (2008).
39. Demeulemeester, E., Vanhoucke, M., and Herroelen, W. "A random network generator for activity-on-the-node networks", *Journal of Scheduling*, **6**, pp. 13–34 (2003).
40. Patterson, J.H. "Project scheduling: the effects of problem structure on heuristic performance", *Naval Research Logistic*, **23**(1), pp. 95–123 (1976).

Biographies

Elham Nabipoor Afruzi is currently a PhD Candidate in Industrial Engineering in K.N. Toosi University of Technology (KNTU), Tehran, Iran. She obtained BS and MS degrees in Industrial Engineering both from KNTU in 2010 and 2012, respectively. Her research interests include project scheduling and management, uncertainty modeling, and exact and meta-heuristic algorithms.

Abdollah Aghaie is a Professor of Industrial Engineering at K.N. Toosi University of Technology in Tehran, Iran. He received his BSc from Sharif University of Technology in Tehran, Iran, MSc from New South Wales University in Sydney in Australia, and PhD from Loughborough University in the U.K. His main research interests are in modeling and simulation, queuing system, quality management and control, knowledge management, and stochastic process.

Amir Abbas Najafi is currently an Associate Professor of Industrial Engineering at K.N. Toosi University of Technology. He received his BS degree in Industrial Engineering from Isfahan University of Technology Isfahan, Iran in 1996 and his MS and PhD degrees in Industrial Engineering from Sharif University of Technology Tehran, Iran in 1998 and 2005, respectively. His research interests include project scheduling and management, portfolio selection models, and applied operations research.