



Project portfolio selection and scheduling under a fuzzy environment

Xiaoxiong Zhang^{1,2,5} · Keith W. Hipel^{2,3,4} · Yuejin Tan⁵

Received: 21 August 2018 / Accepted: 25 February 2019 / Published online: 7 March 2019
© Springer-Verlag GmbH Germany, part of Springer Nature 2019

Abstract

The problem of integrated project portfolio selection and scheduling (PPSS) is among the most important and highly pursued subjects in project management. In this study, a mathematical model and algorithm are designed specifically to assist decision makers decide which projects are to be chosen and when these projects are to be undertaken. More specifically, the PPSS problem is first formulated as a nonlinear multi-objective model with simultaneous consideration of benefit and risk factors. Due to the complexity and uncertainty involved in most real life situations, fuzzy numbers are incorporated into the model, which can provide decision makers with more flexibility. Then, an inverse modeling based multi-objective evolutionary algorithm using a Gaussian Process is presented to obtain the Pareto set. Finally, an illustrative example is used to demonstrate the high efficacy of the foregoing approach, which can provide decision makers with valuable insights into the PPSS process. The proposed algorithm is found to be more effective compared with two other popular algorithms.

Keywords Project portfolio selection and planning · Multi-objective optimization · Fuzzy numbers · Inverse modeling · Gaussian process

1 Introduction

Project portfolio selection and scheduling (PPSS) strives to choose appropriate projects from a given set of competing alternatives and make an overall development plan. Obviously, excellent project systems are regarded as critical to organizations as poorly selected projects are often dysfunctional and may even compromise other projects [1]. However, decision makers (DMs) often face a difficult tradeoff during a project management process. On the one hand, they cannot afford to develop an indiscriminately large number of projects subject to scarce resources, such as money,

manpower, and equipments. On the other hand, they need to develop the right types of projects to maximize their contributions. Therefore, there is a great demand for an overall plan for the selection and scheduling of multiple projects to form an effective and efficient portfolio solution.

The PPSS process, however, is difficult to carry out for DMs, as they may have to take into account extensive data as well as conflicting criteria. Another issue in the problem is associated with the uncertainty of the decision-making environment. In most real-life situations, it is often difficult or even impossible, for DMs to provide complete and precise information regarding project management problems due to complexity, time pressure or limited expertise. Fuzzy set theory was proposed to handle this, because it can provide DMs with more freedom to represent uncertainty and flexibility regarding information [2, 3]. Some authors have focused on PPSS problem under a fuzzy environment. In general, a great deal of attention in the literature has been devoted to the selection, ranking, or scheduling problem in various fields, under both crisp and fuzzy environments. The common approaches toward the problem can be classified into three categories: multi-criteria decision making (MCDM), mathematical programming models, and intelligent optimization algorithms. Please refer to Sect. 2 for a detailed literature review.

✉ Xiaoxiong Zhang
zxxandxx@163.com

¹ The Sixty-third Research Institute, National University of Defense Technology, Nanjing 210007, Jiangsu, China

² Department of Systems Design Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada

³ Centre for International Governance Innovation, Waterloo, ON N2L 6C2, Canada

⁴ Balsillie School of International Affairs, Waterloo, ON N2L 6C2, Canada

⁵ College of Systems Engineering, National University of Defense Technology, Changsha 410073, Hunan, China

Normally, the PPSS problem consists of selecting the appropriate projects from a given set of alternatives and scheduling their development time under various constraints. One of the main goals is to maximize the project benefits to the highest possible extent. Meanwhile, uncertainties may arise in the development of the selected projects. The risk of some projects' failure to provide desired benefits is another important factor that cannot be overlooked. In other words, various criteria need to be considered throughout the PPSS process. Thus, the PPSS problem could be treated as a multi-objective optimization problem which is NP-hard in nature. In the currently existing studies, two solutions are mostly applied in solving the foregoing problems. The first one focuses on transforming multi-objective problems into a single objective. The key limitation is that each optimization run finds only one single solution and knowledge of the problem is required to solve it. However, the PPSS problem may consist of contradictory objectives (like benefit and risk). Quite often, these objectives contradict each other. That is, the larger benefit often means a higher corresponding risk. An increment in one feature could be done at the cost of the other. Therefore, it is unreasonable to mix these objectives into one objective. The second approach uses population based multi-objective optimizing algorithms to search for a set of Pareto optimal solutions.

Although research has been carried out in the PPSS field, several issues remain to be elucidated. First, most existing studies focused only on the selection or scheduling of project portfolios. However, it makes more sense to take into account the selection and scheduling of project portfolios simultaneously. Next, benefit is mostly treated as the main objective in the evaluation of various solutions whereas risk is seldom considered, which may cause unreasonable or ineffective solutions to be recommended [1]. In addition, most existing studies used crisp values in the PPSS modeling and solving process, which is difficult or impossible in real life situations. Thus, incorporating uncertainties regarding information in the decision making process is more realistic and natural. Last, most algorithms proposed to tackle the PPSS problem often entail great computational costs, especially when the number of objectives is large [4].

To address all the aforesaid research void, a quantitative mathematical model and an inverse modeling based multi-objective evolutionary algorithm (IMMOEA) are proposed in this paper to solve the PPSS problem. First, the PPSS problem is formulated as an optimization problem with new objectives under fuzzy constraints. Then, a novel algorithm, named IMMOEA, is used and modified to solve the model in order to obtain the Pareto set. Comparable experiments are also executed to verify its effectiveness. To the best of our knowledge, this is the first time that such an algorithm has been designed specifically for solving the PPSS problem within a fuzzy environment.

The rest of the paper is organized as follows. In Sect. 2, a brief review of different methods developed for the project portfolio selection and/or scheduling problem is presented. Subsequently, the PPSS problem is briefly described and a quantitative model is designed having appropriate constraints and formulations. The proposed algorithm for solving the PPSS problems is then explained in detail in Sect. 4. In Sect. 5, an illustrative example is studied to verify the algorithm's feasibility and practicality. Comparative studies are carried out to demonstrate its usefulness and efficacy. Finally, conclusions and suggestions are made in Sect. 6.

2 Literature review

To date, a great deal of attention in the literature has been devoted to tackling the evaluation, selection or scheduling of project portfolios in various fields. In general, DMs need to evaluate project portfolios and execute the selection (scheduling) process based on certain criteria or try to find the most appropriate set of solutions that can better meet their special requirements. To sum up, all these methods used to tackle this area can be classified into three categories: (1) MCDM methods, (2) mathematical programming models, and (3) intelligent optimization algorithms. In the following part, this paper reviews and analyzes some popular and representative methods adopted in existing studies for each category mentioned above as explained in the following three paragraphs.

Within different MCDM methods, Analytic Hierarchy Process (AHP) and Analytic Network Process (ANP) are two of the most widely used approaches in the literature. For both methods, the relative importance for a given set of projects should be determined. Some other distance-based approaches, like TOPSIS, VIKOR or outranking methods, like ELECTRE, PROMETHEE, are also used to evaluate and select an appropriate set of projects. For example, Vetschera and de Almeida [5] studied the use of PROMETHEE outranking methods for portfolio selection problems. Likewise, Tavana et al. [6] combined Data Envelopment Analysis (DEA), the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS), and linear Integer Programming (IP) for the screening, ranking, and selecting of appropriate project portfolios in a fuzzy environment. Recently the use of hybrid methods combining two or more techniques has recently received more attention. Overall, numerous approaches can be applied to solve MCDM problems associated with project portfolio selection or scheduling [7, 8].

In the second category, researchers have proposed different mathematical programming models based on reality and solved the models to obtain appropriate solutions. Hassanzadeh et al. [9] developed a multi-objective integer

programming model with imprecise information both in objectives and constraints for use with Research and Development (R&D) project portfolio selection problems. Schaefer and Cruz-Reyes [10] proposed a mathematical model framework for portfolio optimization in which projects were allowed to be divided into different tasks and dependencies within tasks could be handled as well. Sefair et al. [11] developed a mean-semivariance project portfolio selection model and used the model in the oil and gas industry. Chen et al. [12] formulated a mixed integer linear programming model considering multiple concurrent projects with the objective of minimizing the total tardiness of all projects. In summary, researchers have been focusing on proposing various models to represent the real world situations and trying to solve the associated models to obtain appropriate solutions [13–16].

As for the third category, some intelligent optimization algorithms have been designed to tackle the complexity of the problem, especially as the number of projects grows. Yassine et al. [17] proposed two new genetic algorithm approaches for scheduling project activities under a multi-project, resource-constrained, and iterative environment. Xiong et al. [18] addressed the problem of R&D project selection associated with resource allocation and activity scheduling by using a cooperative coevolutionary multi-objective algorithm to obtain high-quality solutions. Kumar et al. [19] proposed three meta-heuristics for the solution of project portfolio selection and scheduling problem. The performance of the proposed algorithms has been found to be promising in terms of quality and convergence. Overall, the advantage of using intelligent optimization algorithms in the PPSS problem is their high efficiency in finding appropriate solutions.

Despite the considerable amount of research in the PPSS field, more comprehensive quantitative model under uncertainty and corresponding algorithms are needed to conduct selection and scheduling processes. Thus, an improved scientific model and algorithm for making reasonable decisions for the PPSS problem could prove to be very useful.

3 Problem formulation

In this section, the mathematical model about the PPSS problem is proposed. First, common notations used throughout this paper are listed. Then, the portfolio definition, objectives, and constraints are given with explanations in the following subsections.

3.1 Notations

In this section, some common notations used in the model formulation are explained (Table 1).

3.2 Portfolio definition

In the PPSS problem, a DM or organization needs to select one or more projects from a set of I candidate projects and subsequently schedule the starting time of each selected project within a time of period of length T under various constraints. Thus, the associated decision variables, x_{it} , can be defined as

$$x_{it} = \begin{cases} 1 & \text{if project } i \text{ starts at time } t \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $i = 1, 2, \dots, I$ and $t = 1, 2, \dots, T$.

In this situation, any vector $x = \{x_{11}, x_{12}, \dots, x_{1T}; x_{21}, x_{22}, \dots, x_{2T}; \dots, x_{I1}, x_{I2}, \dots, x_{IT}\}$ with a length of $I \cdot T$ variables, denotes a portfolio. Note that the duration of each project d_i ($i = 1, 2, \dots, I$), where d_i can be a time unit such as months or years, is given. Hence, in this study, we focus on the starting period of each project if it is chosen.

For example, a DM in an armament company may have a budget to develop some new weapons to meet the capability requirements of military operations over the next 5 years. Ten possible candidate types of weapons are under consideration. However, only a small set of weapons (projects) can be chosen and developed due to budget constraints. In this case, there are altogether $2^{I \cdot T} = 2^{10 \times 5} = 2^{50}$ vectors or solutions. Then, the vector $x = \{0, 0, 1, 0, 0; 1, 0, 0, 0, 0; \dots, 0, 0, 0, 0, 0\}$ is an example of a profile. It denotes that the first weapon is chosen in the third year, the second weapon is selected in the first year, and the last weapon is not chosen. However, DMs need to evaluate different portfolios based on appropriate criteria under various constraints.

Note that each project can provide corresponding benefits during its duration. In addition, these projects have specific resource requirements in different periods and may possess risks that should be taken into account when they arise. Normally, higher benefits for a project mean more resources are needed and the risks are greater. In general, benefits, resource requirements, and risks are time-dependent.

3.3 Objectives

Different portfolios need to be evaluated by the DMs based on a set of criteria over T time periods. There are two main types of criteria: benefit type such as profits and market shares, for which one desires higher values, and cost type like risks which one wishes to minimize.

To begin with, consider the first type of criterion. Assume that there are Q kinds of benefits that DMs want to maximize simultaneously. Each benefit is represented by an objective function. For a specific time period, each benefit contains two main components. On the one hand, for each feasible portfolio, the individual contributions of the selected

Table 1 Notations used in this paper

<i>Indices</i>	
i	Index of projects ($i=1, 2, \dots, I$)
t	Index of time ($t=1, 2, \dots, T$)
k	Index of periods ($k=1, 2, \dots, T$)
q	Index of benefits ($q=1, 2, \dots, Q$)
j	Index of synergies ($j=1, 2, \dots, J$)
u	Index of resources ($u=1, 2, \dots, U$)
<i>Parameters</i>	
I	Number of projects
T	Length of periods associated with time
Q	Number of benefits (risks)
J	Number of possible synergies
U	Number of different types of resources
d_i	Duration of project i
$c_{q,i,k+1-t}$	Value of benefit q provided by project i in period k if selected at time t
a_{qjk}	Value of additional benefit q arising from synergy j in period k
g_{jk}	Possible value denoting whether synergy j is activated in period k
F_{qk}	Value of benefit q a portfolio can provide in period k
F_q	Value of benefit q a portfolio can provide throughout all periods
$r_{q,i,k+1-t}$	Probability of project i failing to provide benefit q in period k if selected at time t
R_q	Probability of a portfolio having at least one project that fails to provide benefit q throughout all periods
\tilde{R}_{uk}	Fuzzy minimum amount of resource u in period k
\tilde{R}_{uk}^S	Fuzzy maximum amount of resource u in period k
$r\tilde{s}_{u,i,k+1-t}$	Fuzzy amount of resource u needed by project i in period k if selected at time t
\tilde{h}_{ujk}	Fuzzy amount of resource u needed by synergy j in period k
α_u	Confidence level for resource u
<i>Sets</i>	
A_j	Set of all the related projects for synergy j to be activated
<i>Decision variable</i>	
x_{it}	Whether project i starts at time t

projects based on their implementation periods, should be considered. On the other hand, synergies may exist between certain projects, which will result in additional positive or negative effects [20]. Let $c_{q,i,k+1-t}$ be the value of benefit q provided by project i in period k . Note that in period k , the implementation time of that project is $k + 1 - t$ if it is selected (starts) at time t .

Based on the foregoing discussion, the time-dependent objective function associated with benefit q in period k is defined as

$$F_{qk}(x) = \sum_{i=1}^I \sum_{t=1}^k c_{q,i,k+1-t} \cdot x_{it} + \sum_{j=1}^J g_{jk}(x) a_{qjk} \tag{2}$$

$q = 1, 2, \dots, Q, k = 1, 2, \dots, T,$

where the double summation term on the left denotes the sum of the benefit value that each individual project can provide, and the summation on the right represents the sum of the benefit value arising from the interactions between

certain projects. Note that g_{jk} represents whether synergy j comes into effect in period k , and it takes value 1 if synergy j is activated and 0 otherwise. In other words, only when the portfolio contains all the projects that belong to the set $A_j(j = 1, 2, \dots, J)$ in period k ($k = 1, 2, \dots, T$), is $g_{jk}(x)$ instrumental.

Then, each benefit value q that a portfolio x can provide throughout entire periods should be aggregated as

$$F_q(x) = \sum_{k=1}^T \left(\sum_{i=1}^I \sum_{t=1}^k c_{q,i,k+1-t} \cdot x_{it} + \sum_{j=1}^J g_{jk}(x) a_{qjk} \right) \tag{3}$$

$$= \sum_{k=1}^T F_{qk}(x) \quad q = 1, 2, \dots, Q,$$

Now consider the second type of criterion, mainly associated with risk of failure. Throughout the entire process, there exist some probabilities that projects may fail to provide corresponding benefits due to various causes (such as

technology, cost, organization), which is also an important factor that needs to be considered. Herein, we assume that failure or success of a project does not affect other projects. This does not contradict interactions between projects in terms of providing benefits. For example, a reconnaissance plane and an attack plane, working together, can assist in identifying and destroying certain targets more easily. However, the former does not affect the attack ability of the latter, whereas the latter does not affect the detection ability of the former.

In this study, the term risk specifically stands for the probability of failing to provide certain benefits arising from internal factors (e.g. equipment aging or malfunction) throughout this paper. In addition, we assume that once a project fails to provide certain benefits, it can no longer provide benefits later on. In this case, if a portfolio works as planned and provides proper benefits in period k , then all the projects in this portfolio must work properly in the previous period $k-1$. Thus, we consider only the risk value of a given portfolio throughout the entire period T . In this reason, the total risk objective associated benefit q for a given portfolio x can be formulated as

$$R_q(x) = 1 - \prod_{i=1}^I \left(\prod_{t=1}^{d_i} (1 - r_{qit}) \right)^{\sum_{i=1}^T x_{it}} \quad q = 1, 2, \dots, Q, \tag{4}$$

where r_{qit} denotes the probability of project i failing to provide benefit q in implementation period t , d_i is the duration of project i . $\prod_{t=1}^{d_i} (1 - r_{qit})$ denotes the probability of project i working properly throughout its lifecycle. $\sum_{i=1}^T x_{it}$ represents the selection status of project i . It takes value 1 if project i is selected and 0 otherwise. $\prod_{i=1}^I \left(\prod_{t=1}^{d_i} (1 - r_{qit}) \right)^{\sum_{i=1}^T x_{it}}$ denotes the probability that all the selected projects work as planned and provide proper benefits throughout the entire period. This process can be treated as a series system with $\sum_{i=1}^I \sum_{t=1}^T x_{it}$ elements. Thus, the whole formula represents the probability of having at least one project failing to provide benefit q throughout the entire period.

Then, different portfolios can be evaluated based on their values associated with these two criteria. DMs always want to select appropriate portfolios with greater benefit values while having lower risk values at the same time. However, as mentioned above, the larger benefit often means a higher corresponding risk. That is, an increment in one feature could be done at the cost of the other. Therefore, it is unreasonable to mix these objectives into one objective.

3.4 Constraints

PPSS is a complicated process consisting of choosing the most appropriate projects and scheduling their starting

times. It often takes place under a limited development time and constrained resources throughout the entire decision-making process. In most real-life situations, some parameters cannot be fully known due to a lack of knowledge, time pressure or limited expertise in the problem domain, in which case there exists uncertainty. To handle this, special attention is paid to the uncertainty on parameters in the resource constraints in this study. Next, some preliminaries on fuzzy set theory are first given in the following subsections. Subsequently, different constraints in conformity with real-life situations are proposed.

3.4.1 Fuzzy resource constraints

To begin with, the definition of a triangular fuzzy number (TFN) is given as displayed in Fig. 1. A TFN \tilde{a} on \mathbf{R} can be expressed as a triplet $(a, \underline{a}, \bar{a})$ when its membership function $\mu_{\tilde{a}}(x) : \mathbf{R} \rightarrow [0, 1]$ is equal to

$$\mu_{\tilde{a}}(x) = \begin{cases} 0 & x \leq a - \underline{a}, \\ 1 - \frac{a-x}{\underline{a}}, & a - \underline{a} < x \leq a, \\ 1 - \frac{x-\bar{a}}{\bar{a}-a}, & a < x \leq a + \bar{a}, \\ 0 & x > a + \bar{a}, \end{cases} \tag{5}$$

where a is deemed as the middle value, and \underline{a} and \bar{a} are the left and right spreads from a , respectively.

Based on the forgoing definition, fuzzy numbers are used in this study to model uncertainty. Hence, the constraint associated with each resource spent on all the selected projects in each period is given as

$$R\tilde{I}_{uk} \leq \sum_{i=1}^I \sum_{t=1}^k r\tilde{s}_{u,i,k+1-t} \cdot x_{it} \leq R\tilde{S}_{uk} \tag{6}$$

$u = \{1, 2, \dots, U\}, k = \{1, 2, \dots, T\},$

where $r\tilde{s}_{u,i,k+1-t}$ means the fuzzy amount of resource u needed by project i in period k if it is selected at time t ,

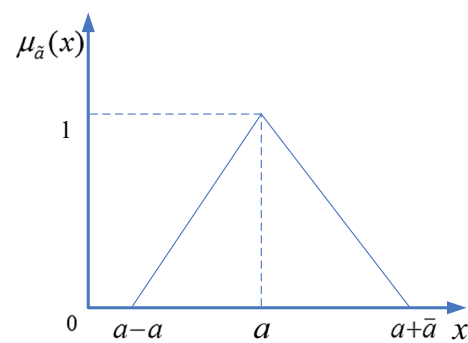


Fig. 1 A triangular fuzzy number

also denoted as a triplet $(rs_{u,i,k+1-t}, \underline{rs}_{u,i,k+1-t}, \overline{rs}_{u,i,k+1-t})$. $\tilde{R}I_{uk}$ and $\tilde{R}S_{uk}$ are the fuzzy minimum and maximum amount of resources that DMs can provide in period k for resource u , also expressed as $(RI_{uk}, \underline{RI}_{uk}, \overline{RI}_{uk})$ and $(RS_{uk}, \underline{RS}_{uk}, \overline{RS}_{uk})$, respectively. In general, the amount of resource in each period should not vary a lot.

Moreover, synergies between certain projects can also consume some resources. In this situation, extra resources are needed in order to come into effect and provide corresponding benefits. Using the same logic as was applied in the first objective function, the resource constraint should be modified as

$$\tilde{R}I_{uk} \leq \sum_{i=1}^I \sum_{t=1}^k r\tilde{s}_{u,i,k+1-t} \cdot x_{it} + \sum_{j=1}^J g_{jk}(x) \tilde{h}_{ujk} \leq \tilde{R}S_{uk}, \quad (7)$$

$$u = \{1, 2, \dots, U\}, \quad k = \{1, 2, \dots, T\},$$

where the double summation term denotes the sum of all the selected projects' individual resource requirements associated with resource u in period k , and the summation on the right part represents the amount of resource u needed by all the activated synergies in period k . In addition, given that resources may not be fully utilized at the end of each period, we assume that the remaining resources would not be used again at the next time point.

To tackle the TFNs used in the above constraint, the comparison method [21] is used, for which main idea is to enable a comparison between two fuzzy numbers using α -cuts without creating extra elements. Specifically, given a TFN, $\tilde{a} = (a, \underline{a}, \bar{a})$, set a confidence level $\alpha \in [0, 1]$. Then, the triangular fuzzy intervals under α -cut, as shown in Fig. 2, can be expressed in mathematical form as

$$\tilde{a}_\alpha = \{x | \mu_{\tilde{a}}(x) \geq \alpha\} = [a_\alpha^L, a_\alpha^R] = [a - (1 - \alpha)\underline{a}, a + (1 - \alpha)\bar{a}] \quad (8)$$

Given two TFNs $\tilde{a} = (a, \underline{a}, \bar{a})$ and $\tilde{b} = (b, \underline{b}, \bar{b})$, $\tilde{a} \leq \tilde{b}$ at the confidence level $\alpha \in [0, 1]$, if the following two conditions are satisfied:

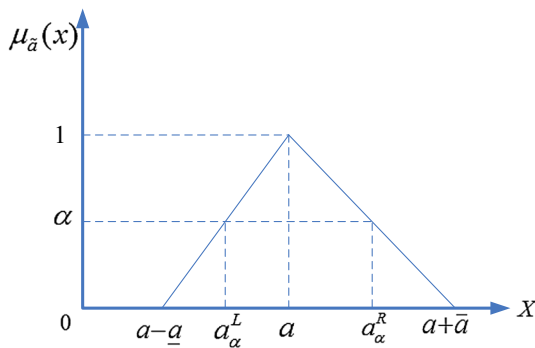


Fig. 2 Triangular fuzzy intervals under α -cut

$$a + (1 - \beta)\underline{a} \leq b + (1 - \beta)\underline{b} \quad (9)$$

$$a - (1 - \beta)\bar{a} \leq b - (1 - \beta)\bar{b} \quad (10)$$

for all $\beta \in [\alpha, 1]$.

Thus, given a confidence level $\alpha_u \in [0, 1]$ for resource u , the resource constraint (7), as mentioned above, can be transformed into the following two parts, shown as

$$\begin{aligned} RI_{uk} + (1 - \beta_u)\overline{RI}_{uk} &\leq \sum_{i=1}^I \sum_{t=1}^k (rs_{u,i,k+1-t} + (1 - \beta_u)\overline{rs}_{u,i,k+1-t}) \cdot x_{it} \\ &+ \sum_{j=1}^J g_{jk}(x) \cdot (h_{ujk} + (1 - \beta_u)\bar{h}_{ujk}) \\ &\leq RS_{uk} + (1 - \beta_u)\overline{RS}_{uk} \quad u = \{1, 2, \dots, U\}, k = \{1, 2, \dots, T\}, \end{aligned} \quad (11)$$

$$\begin{aligned} RI_{uk} - (1 - \beta_u)\underline{RI}_{uk} &\leq \sum_{i=1}^I \sum_{t=1}^k (rs_{u,i,k+1-t} - (1 - \beta_u)\underline{rs}_{u,i,k+1-t}) \cdot x_{it} \\ &+ \sum_{j=1}^J g_{jk}(x) \cdot (h_{ujk} - (1 - \beta_u)\underline{h}_{ujk}) \\ &\leq RS_{uk} - (1 - \beta_u)\underline{RS}_{uk} \quad u = \{1, 2, \dots, U\}, k = \{1, 2, \dots, T\}, \end{aligned} \quad (12)$$

where $\beta_u \in [\alpha_u, 1]$, constraint (11) is obtained based on constraint (9) and constraint (12) is obtained based on constraint (10), both of which denote the constraint associated with resource u in period k .

3.4.2 Other constraints

(1) Constraints on synergies between projects

As mentioned above, g_{jk} represents whether synergy j comes into effect in period k . It takes value 1 only when the portfolio x contains all the projects simultaneously belonging to the set A_j in period k so that synergy j can be activated. Based on above, $g_{jk}(x)$ can be represented in mathematical form as

$$g_{jk}(x) = \prod_{i \in A_j} \sum_{t=\max(1, k-d_i+1)}^k x_{it} \quad j = 1, 2, \dots, J, k = 1, 2, \dots, T, \quad (13)$$

where d_i denotes the duration of the i th project as before, $\sum_{t=\max(1, k-d_i+1)}^k x_{it}$ takes value 1 if project i exists in a portfolio in period k , $k - d_i + 1$ is the earliest starting time for project i so that it is still executed in period k . In case that $k - d_i + 1$ is smaller than 1, which is meaningless, t takes the maximum of 1 and $k - d_i + 1$. In other words, this constraint ensures that the function $g_{jk}(x)$ takes value 1 only when the portfolio x verifies the synergy j .

- (2) Constraints on the simultaneous execution of projects
 Sometimes, DMs may impose restrictions on the number of projects that can be carried out at the same time in a certain period k . In this case, lower and upper bounds are defined as $\underline{n}(k)$ and $\bar{n}(k)$.

$$\underline{n}(k) \leq \sum_{i=1}^I \sum_{t=\max(1, k-d_i+1)}^k x_{it} \leq \bar{n}(k), \quad k = 1, 2, \dots, T, \tag{14}$$

where $\sum_{t=\max(1, k-d_i+1)}^k x_{it}$ takes value 1 if project i is executed in period k , and 0 otherwise.

- (3) Constraints about uniqueness
 Each project can at most be executed only once. However, DMs may demand that certain projects must be executed. For a particular project, this restriction can be implemented by setting its lower bound NL_i as 1. Otherwise, NL_i equals 0.

$$NL_i \leq \sum_{t=1}^T x_{it} \leq 1 \quad i \in \{1, 2, \dots, I\} \tag{15}$$

Notice that the above constraint on the lower bound permits them to have a value of 1 when certain projects must be implemented.

- (4) Constraints about the starting period
 This constraint ensures that the starting periods of certain projects are within the given intervals if chosen by using the following inequality:

$$v_i \sum_{t=1}^T x_{it} \leq \sum_{t=1}^T t \cdot x_{it} \leq \omega_i \quad i \in \{1, 2, \dots, I\}, \tag{16}$$

where v_i and ω_i are the lower and upper bounds associated with the starting time for project i .

- (5) Constraints about precedence
 Like other project scheduling problems, in some cases there may exist some precedent constraints among certain projects. For example, suppose that project i must be executed prior to project m . Then, it can be expressed in a mathematical form as

$$\sum_{t=1}^T t \cdot x_{it} \leq \sum_{t=1}^T t \cdot x_{mt} \quad 1 \leq i, m \leq I \tag{17}$$

In summary, this PPSS model can be formulated with the objectives given in Eqs. (3) and (4), subject to the constraints provided in constraints (11)–(17) inclusive.

4 An inverse modeling-based multi-objective evolutionary algorithm

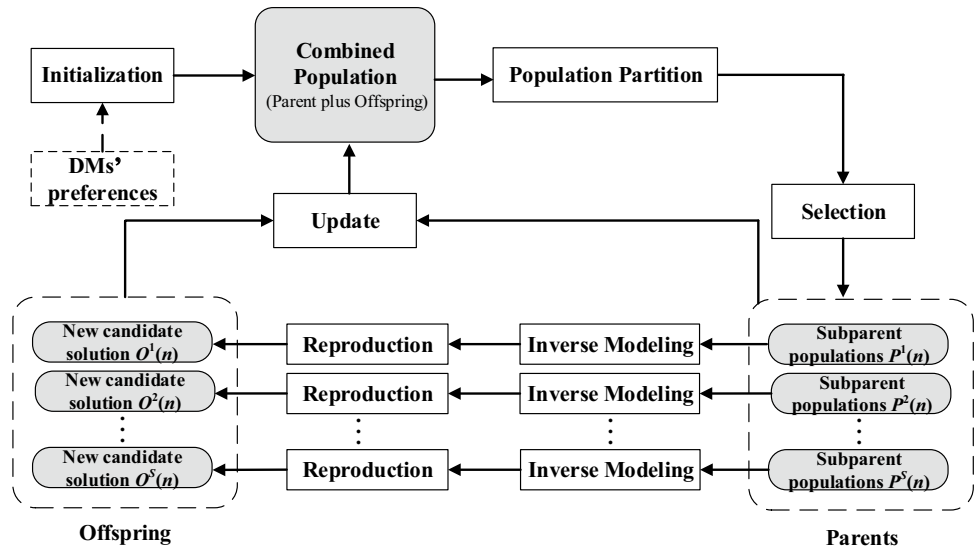
The classic PPSS problem is very complex. Several factors regarding benefit, risk, and resource constraints need to be considered simultaneously. Furthermore, additional constraints associated with time, the interrelationships between different projects, make the problem more complicated. Traditional mathematical programming methods are inefficient or limited in the application. In contrast, multi-objective evolutionary algorithms (MOEAs) have proven to be highly efficient in solving this kind of problem. However, unlike most MOEAs that store non-dominated solutions during the search, the IMMOEA proposed by Cheng et al. [22] uses Gaussian process based inverse modeling to create new solutions, which has exhibited robust search performance on a variety of multi-objective optimization test problems. Accordingly, the existing IMMOEA is used and modified in this paper to handle the new PPSS problem.

4.1 Overview of the proposed IMMOEA

Model-based algorithms have been demonstrated to be very effective in approximating the Pareto front for multi-objective problems [23]. A new model based algorithm with regard to IMMOEA is introduced in this section to obtain the Pareto solutions. Its main idea lies in mapping non-dominated solutions from the objective space (OS) to the decision space (DS) based on some constructed inverse models [22]. To the best of the authors' knowledge, there is no prior study using IMMOEA to solve the multi-objective PPSS problem under uncertainty.

Figure 3 displays an overview of the proposed IMMOEA, for which the pseudo code is given in Algorithm 1 [22] for the PPSS problem. To begin with, an initial population is generated, which is then partitioned into different groups based on some reference vectors (item 4 in Algorithm 1). Note that DMs' preferences can be incorporated in the initialization process by setting certain constraints on the values used to encode various projects. The population is then evaluated by estimating each solution's objectives and constraints. Herein, objectives refer to the benefits and risks of different PPSS solutions, whereas constraints refer to the extent to which different constraint limits are exceeded. As traditional IMMOEA focuses on solving optimization problems without considering the constraints. Thus, the penalty function is used to address various constraints associated with the PPSS problem. More specifically, the penalty function f is defined as

Fig. 3 Overview of IMMOEA based on [22]



$$\begin{aligned}
 f = & \sum_{u=1}^U \sum_{k=1}^T \max \left(0, R\tilde{I}_{uk} - \sum_{i=1}^I \sum_{t=1}^k r\tilde{s}_{u,i,k+1-t} \cdot x_{it} - \sum_{j=1}^J g_{jk}(x)\tilde{h}_{ujk} \right) \\
 & + \sum_{u=1}^U \sum_{k=1}^T \max \left(0, \sum_{i=1}^I \sum_{t=1}^k r\tilde{s}_{u,i,k+1-t} \cdot x_{it} + \sum_{j=1}^J g_{jk}(x)\tilde{h}_{ujk} - R\tilde{S}_{uk} \right) \\
 & + \sum_{k=1}^T \max \left(0, \bar{n}(k) - \sum_{i=1}^I \sum_{t=\max(1, k-d_i+1)}^k x_{it} \right) + \sum_{k=1}^T \max \left(0, \sum_{i=1}^I \sum_{t=\max(1, k-d_i+1)}^k x_{it} - \bar{n}(k) \right) \\
 & + \sum_{i=1}^I \max \left(0, NL_i - \sum_{t=1}^T x_{it} \right) + \sum_{i=1}^I \max \left(0, \sum_{t=1}^T x_{it} - 1 \right) \\
 & + \sum_{i=1}^I \max \left(0, v_i \sum_{t=1}^T x_{it} - \sum_{t=1}^T t \cdot x_{it} \right) + \sum_{i=1}^I \max \left(0, \sum_{t=1}^T t \cdot x_{it} - \omega_i \right) \\
 & + \sum_{i=1}^I \sum_{m=1}^I \max \left(0, \sum_{t=1}^T t \cdot x_{it} - \sum_{t=1}^T t \cdot x_{mt} \right),
 \end{aligned} \tag{18}$$

where each part in Eq. (18) denotes the extent to which the corresponding limit is exceeded in terms of the minimum resource constraint, the maximum resource constraint, the simultaneous execution of projects, the uniqueness of projects, the starting periods, and the precedence of projects, respectively. Taking the first term $\sum_{u=1}^U \sum_{k=1}^T \max(0, R\tilde{I}_{uk} - \sum_{i=1}^I \sum_{t=1}^k r\tilde{s}_{u,i,k+1-t} \cdot x_{it} - \sum_{j=1}^J g_{jk}(x)\tilde{h}_{ujk})$ as an example, for each resource u in each period k , the resource used in total cannot be lower than the fuzzy minimum amount of resource $R\tilde{I}_{uk}$, otherwise, the corresponding penalty is $R\tilde{I}_{uk} - \sum_{i=1}^I \sum_{t=1}^k r\tilde{s}_{u,i,k+1-t} \cdot x_{it} - \sum_{j=1}^J g_{jk}(x)\tilde{h}_{ujk}$. Conversely, the corresponding penalty function is 0, meaning that no

violation of the constraint is triggered. The same logic applies to the other constraints. Then, the penalty value is incorporated into the objective value of each solution. The greater (smaller) is the new benefit (risk) value, the better is the individual (solution).

Then, a selection procedure is carried out within each group to make up the parents of each group (item 5 in Algorithm 1). Next, a Gaussian process (GP) based inverse model is designed with regard to the generation of offspring in each group. Finally, the offspring and their parents are put together to form a new combined population. Some details of IMMOEA are provided as in Sects. 4.2–4.4 inclusive.

Algorithm 1: Pseudo code of IMMOEA for the PPSS problem

- 1: **Input:** population size Pop , maximum number of generations $MaxGen$, number of reference vectors S , random group size L ;
- 2: **Initialization:** $n = 0$, initialize population $P(0)$ and estimate each individual's corresponding benefit, risk, and penalty, generate the S uniform reference vectors;
- 3: **while** $n < MaxGen$ **do**
- 4: **Population Partition:** partition current population $P(n)$ by comparing each individual's relative position with the S predefined reference vectors;
- 5: **Selection:** Perform selection process in each group to generate their subparent population $P^1(n), \dots, P^S(n)$;
- 6: **for** $s = 1, 2, \dots, S$, **do**
- 7: **Inverse Modeling:** apply random grouping technique for each subpopulation $P^s(n)$ to determine the inverse models; Moreover, a GP is trained for each inverse model;
- 8: **Reproduction:** for each subpopulation $P^s(n)$, sample the objective space and then map them back to the decision space to generate new offspring $O^s(n)$ using the inverse models; Execute mutation process if it is not the final generation;
- 9: **end for**
- 10: **Update:** a new combined population is generated by incorporating the offspring and their current parents for the next generation;
- 11: $n = n + 1$;
- 12: **end while**

4.2 Partitioning the combined population

As illustrated in Algorithm 1, the combined population is first divided into several partitions. To begin with, S uniform reference vectors \vec{v}_s are generated for a problem having m objectives [24]. More precisely, a set of S uniformly distributed points $\vec{w}_s = (w_s^1, w_s^2, \dots, w_s^m), s = 1, 2, \dots, S$ are first generated, where each of the weight vectors w_s^i selects a value from $\left\{ \frac{0}{H}, \frac{1}{H}, \dots, \frac{H}{H} \right\}$, $\sum_{i=1}^m w_s^i = 1$, and H is a parameter controlling the number of vectors. After that, the uniformly distributed reference vector \vec{v}_s can be obtained by mapping \vec{w}_s from the unit hyperplane to a unit hypersphere as

$$\vec{v}_s = \frac{\vec{w}_s}{\|\vec{w}_s\|} \quad s = 1, 2, \dots, S \tag{19}$$

Once the reference vectors have been obtained, the total combined population is then partitioned into S subpopulations. This step is done by comparing their relative positions in the OS between each individual solution with the predefined reference vectors mentioned above [22]. More specifically, consider an arbitrary solution x_i as an example, which is assigned to reference vector \vec{v}_s under the condition that the angle between its position with that of \vec{v}_s is the minimum among all the reference vectors by calculating the sine function value of their angles using

$$s_i = \operatorname{argmin}_{s=1, \dots, S} \left(\frac{\vec{x}_i}{\|\vec{x}_i\|} \times \vec{v}_s \right) \tag{20}$$

4.3 Inverse modeling

Let X denote a set of the decision vectors (i.e., PPSS solutions as defined in Eq. (1)) and Y denote a set of objective objectives associated with benefits and risks, both of which represent the current parent population. To generate the offspring X^0 , a group of generated objective vectors Y^0 in the OS is mapped back to the DS using $P(X|Y)$ associated with the Bayes' Theorem [22]. Herein, $P(X|Y)$ is the conditional probability distribution, which can be estimated by a probabilistic inverse model mapping Y back to X . In fact, $P(Y|X)$ is the priori knowledge, i.e., the objectives functions mapping X to Y .

$$P(X^0) = \frac{P(X|Y)P(Y^0)}{P(Y|X)}, \tag{21}$$

In general, the probabilistic inverse model $P(X|Y)$ is decomposed into $m \times n$ univariate models $P(x_i|f_j)$. Note that m equals $2 \cdot Q$ (associated with Q benefits and Q risks) and n equals $I \cdot T$ (length of the decision variable) as mentioned in Sect. 3. However, it is very hard or impossible to estimate the inverse model for a problem with m objectives and n variables, especially those associated with large-scale optimization problems. In this reason, random grouping technology [25] is used for the building of multiple inverse models. Each univariate model takes only one single objective and one decision variable into account. More specifically, a total of m groups of inverse models are built for a problem having m objectives, where the j th objective f_j would be treated as the input variable in group j , and L variables are randomly picked as well. As can be seen, the total number of inverse models now decreases from $m \times n$ to $m \times L$, where L is much smaller than n .

Assume that N_j pairs of data are needed for training L GP models. Then, $T_{j,i}$ is denoted as the specific training data set used for training $P(x_i|f_j)$ in the j th group as

$$T_{j,i} = [f_j, x_i], \tag{22}$$

where $f_j = [f_j^1, \dots, f_j^{N_j}]^T$ and $x_i = [x_i^1, \dots, x_i^{N_i}]^T$, f_j^h is the j th objective value of the h th individual, and x_i^h is the i th decision variable of the h th individual.

The inverse mapping $P(x_i|f_j)$ can be assumed to be a latent function $g(\cdot)$, represented by a number of arbitrary function variables $g = \{g_1, g_2, \dots, g_{N_j}\}$, which follows a joint Gaussian distribution [26]:

$$P(g|f_j) = N(\bar{g}, C), \tag{23}$$

where $N(\bar{g}, C)$ represents the multivariate Gaussian distribution, \bar{g} denotes the mean vector, and C is the covariance matrix. $P(g|f_j)$ stands for the conditional probability associated with the training data f_j . In order to decrease the

computational burden of training a GP model, the mean function $\mu(f_j)$ which calculates the mean value of all the training data, is usually set to 0 by means of subtracting certain offset from $T_{j,i}$. Thus, Eq. (23) can be transformed to

$$P(g|f_j) = N(0, C) \tag{24}$$

A linear covariance function is used in this study for saving computational cost as

$$C(f_j^p, f_j^q) = (f_j^p)^T f_j^q \tag{25}$$

Assume that $x_i = g(f_j) + \varepsilon$ is affected by white noise $\varepsilon \sim N(0, (\sigma_n)^2 Z)$. Then, a proper noise model can be expressed as

$$P(x_i|g) = N(g, (\sigma_n)^2 Z), \tag{26}$$

where Z is an identity matrix. Thus, the following equality associated with the marginal likelihood is obtained:

$$P(x_i|f_j) = \int P(x_i|g)P(g|f_j)dg = N(0, C + (\sigma_n)^2 Z). \tag{27}$$

Based on (27), given a test input $f_{j,*}$, a Bayesian inference can be utilized to achieve the corresponding output $x_{i,*}$. The following equation is used to obtain the mean and variance, correspondingly.

$$\begin{aligned} \mu_{j,i} &= C_*^T (C + (\sigma_n)^2 Z)^{-1} x_i \\ (\sigma_{j,i})^2 &= C_{**} - C_*^T (C + (\sigma_n)^2 Z)^{-1} C_* \end{aligned} \tag{28}$$

where $C_{**} = [c(f_j^1, f_j^1), \dots, c(f_j^{N_t}, f_j^{N_t})]$ is composed of all the covariance values between each pairs of elements among f_j , and $C_* = [c(f_{j,*}^1, f_j^1), \dots, c(f_{j,*}^{N_t}, f_j^{N_t})]$ consists of the covariance values between each pair of elements associated with the test input $f_{j,*}$ and that of the training data f_j . In this way, the inverse model $P(x_i|f_j)$ is then transformed into a set of normal distributions as shown in Eq. (28).

In this study, the following equation is used to sample the input points $f_{j,*}$ as

$$f_{j,*} = \left\{ \frac{[f_j^{\min} - 0.5\xi_j, f_j^{\max} + 0.5\xi_j]}{N_s} \right\}, \tag{29}$$

where N_s is the sample size, which is usually set to N_t , f_j^{\max} and f_j^{\min} are the maximum and minimum values, respectively, associated with objective j in f_j , and $\xi_j = f_j^{\max} - f_j^{\min}$.

4.4 Reproduction

Once the inverse models have been obtained, the test input points $f_{j,*}$ can be mapped to the DS $x_{i,*}$ by means of adding

the mean values $\mu_{j,i} = (\mu_{j,i}^1, \dots, \mu_{j,i}^{N_s})$ with white noise $z \sim (N(0, (\sigma_{j,i}^1})^2), \dots, N(0, (\sigma_{j,i}^{N_s})^2))$ using

$$x_{i,*} = \mu_{j,i}(f_{j,*}) + z_{j,i}, \tag{30}$$

where $\mu_{j,i}^1, \dots, \mu_{j,i}^{N_s}$ and $(\sigma_{j,i}^1})^2, \dots, (\sigma_{j,i}^{N_s})^2$ are the outputs associated with a trained GP model in Eq. (28). After that, replace the current values $x_i = (x_i^1, \dots, x_i^{N_s})$ of each subpopulation with the new generated solutions $x_{i,*} = (x_{i,*}^1, \dots, x_{i,*}^{N_s})$. Then, a new combined population is formed by integrating the offspring individuals produced by each subpopulation with their parents. A new round is initiated until the termination condition is satisfied. For more details, please refer to [22, 27].

5 Illustrative example

Assume that an investor has a budget to renovate some of the projects over the next five years. However, due to limited resources (mainly associated with money, in thousands of monetary units), only a set of projects can be selected. Two objectives are taken into account as maximizing benefits (in thousands of monetary units) and minimizing risks under various constraints. The main values of the projects associated with benefits that each project can provide and the risks that each project may possess in failing to provide benefits in different periods are given in Tables 2 and 3, respectively. In general, the risk values increase as time goes by. Next, portfolios are evaluated based on the two objectives under the conditions of satisfying various constraints.

Regarding the first objective for each portfolio, the sum of all the benefit values that each individual project can provide is

$$\sum_{k=1}^5 \sum_{i=1}^{10} \sum_{t=1}^k c_{1,i,k+1-t} \cdot x_{it}, \tag{31}$$

Table 2 Benefit values of projects

Year of implementation	1	2	3	4	5
c_{11t}	0.6	0.8			
c_{12t}	0.7				
c_{13t}	0.5	0.6	0.7	0.5	0.3
c_{14t}	0.6	0.8	0.6		
c_{15t}	0.45	0.7			
c_{16t}	0.8	0.4			
c_{17t}	0.7				
c_{18t}	0.6	0.7	0.7	0.5	
c_{19t}	0.45	0.8	0.7	0.8	0.6
$c_{1,10,t}$	0.6	0.75	0.8		

Table 3 Risk values of projects

Year of imple- mentation	1	2	3	4	5
r_{11t}	0.15	0.20			
r_{12t}	0.10				
r_{13t}	0.05	0.08	0.10	0.12	0.15
r_{14t}	0.08	0.12	0.10		
r_{15t}	0.05	0.10			
r_{16t}	0.10	0.11			
r_{17t}	0.10				
r_{18t}	0.08	0.08	0.10	0.10	
r_{19t}	0.02	0.08	0.12	0.12	0.15
$r_{1,10,t}$	0.05	0.10	0.13		

Table 4 Fuzzy resource coefficients for projects

Year of imple- mentation	1	2	3	4	5
$r\tilde{s}_{11t}$	$2\tilde{5}$	$3\tilde{2}$			
$r\tilde{s}_{12t}$	$4\tilde{2}$				
$r\tilde{s}_{13t}$	$2\tilde{4}$	$3\tilde{7}$	$4\tilde{0}$	$3\tilde{5}$	$4\tilde{2}$
$r\tilde{s}_{14t}$	$4\tilde{0}$	$4\tilde{5}$	$4\tilde{8}$		
$r\tilde{s}_{15t}$	$1\tilde{5}$	$1\tilde{8}$			
$r\tilde{s}_{16t}$	$3\tilde{0}$	$2\tilde{4}$			
$r\tilde{s}_{17t}$	$4\tilde{5}$				
$r\tilde{s}_{18t}$	$5\tilde{0}$	$4\tilde{3}$	$5\tilde{2}$	$2\tilde{0}$	
$r\tilde{s}_{19t}$	$1\tilde{8}$	$2\tilde{3}$	$3\tilde{4}$	$4\tilde{5}$	$4\tilde{0}$
$r\tilde{s}_{1,10,t}$	$3\tilde{2}$	$3\tilde{6}$	$4\tilde{6}$		
\tilde{h}_{11k}	$1\tilde{5}$	$1\tilde{5}$	$1\tilde{5}$	$1\tilde{5}$	$1\tilde{5}$
$R\tilde{I}_{1,k}$	$\tilde{0}$	$\tilde{0}$	$\tilde{0}$	$\tilde{0}$	$\tilde{0}$
$R\tilde{S}_{1,k}$	$1\tilde{00}$	$1\tilde{20}$	$1\tilde{30}$	$1\tilde{15}$	$9\tilde{5}$

where the values for $c_{1,i,k+1-t}$ are given as tabulated in Table 2.

Moreover, additional benefit (0.7, in this case) can be expected arising from the interactions between Projects 4 and 6. Therefore, the benefit value throughout the entire period should be aggregated as

$$Max F(x) = \sum_{k=1}^5 \sum_{i=1}^{10} \sum_{t=1}^k c_{1,i,k+1-t} \cdot x_{it} + \sum_{k=1}^5 0.7g_{1k}(x), \quad (32)$$

where $g_{1k}(x)$ equals 1 if the fourth and sixth projects exist simultaneously in period k . Otherwise, it takes a value of 0, where $A_1 = \{4, 6\}$, and $g_{1k} = \prod_{i \in A_1} \sum_{t=\max(1, k-d_i+1)}^k x_{it}$, $k = 1, 2, \dots, 5$.

Regarding the second objective for each portfolio, the probability of failure associated with projects should be expected to increase as time passes. The risk value corresponding to the second objective is formulated as

$$Min R(x) = 1 - \prod_{i=1}^{10} \left(\prod_{t=1}^{d_i} (1 - r_{1it}) \right)^{\sum_{i=1}^5 x_{it}}, \quad (33)$$

where the probabilities for each project to become invalid in different periods r_{1it} are given as tabulated in Table 3.

As for the resources, the fuzzy estimated amount of resources required for each project in different periods $r\tilde{s}_{1it}$ are given in Table 4. The fuzzy amount of resource for the synergy \tilde{h}_{11k} , as well as the fuzzy minimum and maximum amount of resources available in different periods are also given.

As mentioned above, the remaining resource at the end of each year will not be re-used in the next year. Based on constraint (7), the set of constraints on resources in each year can be obtained as

$$\begin{aligned}
 k = 1 : & \tilde{0} \leq 2\tilde{5}x_{11} + 4\tilde{2}x_{21} + 2\tilde{4}x_{31} + 4\tilde{0}x_{41} + 1\tilde{5}x_{51} + 3\tilde{0}x_{61} + 4\tilde{5}x_{71} + 5\tilde{0}x_{81} + 1\tilde{8}x_{91} + 3\tilde{2}x_{10,1} + 1\tilde{5}g_{11} \leq 1\tilde{00} \\
 k = 2 : & \tilde{0} \leq 3\tilde{2}x_{11} + 2\tilde{5}x_{12} + 4\tilde{2}x_{22} + 3\tilde{7}x_{31} + 2\tilde{4}x_{32} + 4\tilde{5}x_{41} + 4\tilde{0}x_{42} + 1\tilde{8}x_{51} + 1\tilde{5}x_{52} + 2\tilde{4}x_{61} + 3\tilde{0}x_{62} \\
 & + 4\tilde{5}x_{72} + 4\tilde{3}x_{81} + 5\tilde{0}x_{82} + 2\tilde{3}x_{91} + 1\tilde{8}x_{92} + 3\tilde{6}x_{10,1} + 3\tilde{2}x_{10,2} + 1\tilde{5}g_{12} \leq 1\tilde{20} \\
 k = 3 : & \tilde{0} \leq 3\tilde{2}x_{12} + 2\tilde{5}x_{13} + 4\tilde{2}x_{23} + 4\tilde{0}x_{31} + 3\tilde{7}x_{32} + 2\tilde{4}x_{33} + 4\tilde{8}x_{41} + 4\tilde{5}x_{42} + 4\tilde{0}x_{43} + 1\tilde{8}x_{52} + 1\tilde{5}x_{53} \\
 & + 1\tilde{8}x_{52} + 1\tilde{5}x_{53} + 2\tilde{4}x_{62} + 3\tilde{0}x_{63} + 4\tilde{5}x_{73} + 5\tilde{2}x_{81} + 4\tilde{3}x_{82} + 5\tilde{0}x_{83} + 3\tilde{4}x_{91} + 2\tilde{3}x_{92} + 1\tilde{8}x_{93} \\
 & + 4\tilde{6}x_{10,1} + 3\tilde{6}x_{10,2} + 3\tilde{2}x_{10,3} + 1\tilde{5}g_{13} \leq 1\tilde{30} \\
 k = 4 : & \tilde{0} \leq 3\tilde{2}x_{13} + 2\tilde{5}x_{14} + 4\tilde{2}x_{24} + 3\tilde{5}x_{31} + 4\tilde{0}x_{32} + 3\tilde{7}x_{33} + 2\tilde{4}x_{34} + 4\tilde{8}x_{42} + 4\tilde{5}x_{43} + 4\tilde{0}x_{44} + 1\tilde{8}x_{53} \\
 & + 1\tilde{5}x_{54} + 2\tilde{4}x_{63} + 3\tilde{0}x_{64} + 4\tilde{5}x_{74} + 2\tilde{0}x_{81} + 5\tilde{2}x_{82} + 4\tilde{3}x_{83} + 5\tilde{0}x_{84} + 4\tilde{5}x_{91} + 3\tilde{4}x_{92} + 2\tilde{3}x_{93} \\
 & + 1\tilde{8}x_{94} + 4\tilde{6}x_{10,2} + 3\tilde{6}x_{10,3} + 3\tilde{2}x_{10,4} + 1\tilde{5}g_{14} \leq 1\tilde{15} \\
 k = 5 : & \tilde{0} \leq 3\tilde{2}x_{14} + 2\tilde{5}x_{15} + 4\tilde{2}x_{25} + 4\tilde{2}x_{31} + 3\tilde{5}x_{32} + 4\tilde{0}x_{33} + 3\tilde{7}x_{34} + 2\tilde{4}x_{35} + 4\tilde{8}x_{43} + 4\tilde{5}x_{44} + 4\tilde{0}x_{45} \\
 & + 1\tilde{8}x_{54} + 1\tilde{5}x_{55} + 2\tilde{4}x_{64} + 3\tilde{0}x_{65} + 4\tilde{5}x_{75} + 2\tilde{0}x_{82} + 5\tilde{2}x_{83} + 4\tilde{3}x_{84} + 5\tilde{0}x_{85} + 4\tilde{0}x_{91} + 4\tilde{5}x_{92} \\
 & + 3\tilde{4}x_{93} + 2\tilde{3}x_{94} + 1\tilde{8}x_{95} + 4\tilde{6}x_{10,3} + 3\tilde{6}x_{10,4} + 3\tilde{2}x_{10,5} + 1\tilde{5}g_{15} \leq 9\tilde{5}
 \end{aligned} \quad (34)$$

where $g_{11} = x_{41} \cdot x_{61}, g_{12} = \left(\sum_{t=1}^2 x_{4t}\right) \cdot \left(\sum_{t=1}^2 x_{6t}\right), g_{13} = \left(\sum_{t=1}^3 x_{4t}\right) \cdot \left(\sum_{t=2}^3 x_{6t}\right), g_{14} = \left(\sum_{t=2}^4 x_{4t}\right) \cdot \left(\sum_{t=3}^4 x_{6t}\right)$ and $g_{15} = \left(\sum_{t=3}^5 x_{4t}\right) \cdot \left(\sum_{t=4}^5 x_{6t}\right)$, denoting whether the synergy takes place in the corresponding period. In general, the left side of (34) represents the fuzzy minimum amount of resources available in each period, whereas the right side limits the fuzzy maximum amount of resources available in each period. This set of constraints ensures that the resources used for a portfolio x in each period are within the given intervals.

Fuzzy constraints have to be transferred to crisp values for comparison. As mentioned above, a parameter $\alpha = 0.9$ is given, which can also be deemed as a confidence level. Suppose that the resources required for each project can increase or decrease both by 25% of their middle values, whereas the resource available in each year can increase or decrease by up to 20% and 30%, respectively. Thus, the above constraint on resources (34) should be transferred to two parts based on (11) and (12) for comparison. Next, some additional constraints are given by DMs by taking some real life situations and individual preferences into consideration.

DMs determined that at least two projects, whereas five projects at most, can be executed simultaneously in the second and third years. Thus, the following two inequalities are obtained using constraint (14):

$$2 \leq \sum_{t=1}^2 x_{1t} + \sum_{t=2}^2 x_{2t} + \sum_{t=1}^2 x_{3t} + \sum_{t=1}^2 x_{4t} + \sum_{t=1}^2 x_{5t} + \sum_{t=1}^2 x_{6t} + \sum_{t=2}^2 x_{7t} + \sum_{t=1}^2 x_{8t} + \sum_{t=1}^2 x_{9t} + \sum_{t=1}^2 x_{10,t} \leq 5 \tag{35}$$

$$2 \leq \sum_{t=2}^3 x_{1t} + \sum_{t=3}^3 x_{2t} + \sum_{t=1}^3 x_{3t} + \sum_{t=1}^3 x_{4t} + \sum_{t=2}^3 x_{5t} + \sum_{t=2}^3 x_{6t} + \sum_{t=3}^3 x_{7t} + \sum_{t=1}^3 x_{8t} + \sum_{t=1}^3 x_{9t} + \sum_{t=1}^3 x_{10,t} \leq 5 \tag{36}$$

DMs found that Project 6 must be selected and scheduled, whereas there are no mandatory constraints on other projects.

$$0 \leq \sum_{t=1}^5 x_{it} \leq 1 \quad i = 1, 2, 3, 4, 5, 7, 8, 9, 10 \tag{37}$$

$$1 \leq \sum_{t=1}^5 x_{6t} \leq 1 \Rightarrow \sum_{t=1}^5 x_{6t} = 1 \tag{38}$$

Some limitations on the starting periods of each project are given if selected:

$$\begin{aligned} \sum_{t=1}^5 t \cdot x_{1t} \leq 3, & 3 \sum_{t=1}^5 x_{3t} \leq \sum_{t=1}^5 t \cdot x_{3t} \leq 4, \sum_{t=1}^5 t \cdot x_{5t} \leq 3 \\ 2 \sum_{t=1}^5 x_{6t} \leq \sum_{t=1}^5 t \cdot x_{6t} \leq 3, & \sum_{t=1}^5 x_{8t} \\ \leq \sum_{t=1}^5 t \cdot x_{8t} \leq 4, & 3 \sum_{t=1}^5 x_{9t} \leq \sum_{t=1}^5 t \cdot x_{9t} \leq 5 \end{aligned} \tag{39}$$

As Project 3 can provide Project 8 with some additional support, the former one should be scheduled earlier before project 8 if chosen. Thus, a new constraint about precedence is given as

$$\sum_{t=1}^5 t \cdot x_{3t} \leq \sum_{t=1}^5 t \cdot x_{8t} \tag{40}$$

The bi-objective optimization model is now built with the objectives contained in (32) and (33), subject to the constraints (34)–(40). Then, IMMOCA is used to solve it using Matlab Version R2016a. Additionally, two other traditional multi-objective optimization algorithms, namely a multi-objective evolutionary algorithm based on decomposition (MOEA/D) [28] and non-dominated sorting genetic algorithm II (NSGA-II) [29], are also independently used to solve the problem under the same constraints for comparison purposes. MOEA/D decomposes a multi-objective optimization problem into a number of scalar optimization sub-problems and optimizes them simultaneously, whereas NSGA-II uses a fast non-dominated sorting procedure and an elitist-preserving approach to create a diverse Pareto-optimal front.

Three metrics, namely convergence (γ), diversity (Δ) and Generational Distance (GD), are calculated to measure the results of the aforementioned three different algorithms. More specifically, convergence (γ) is used to evaluate the accuracy of approximating the obtained results to the global Pareto-optimal front. If γ equals 0, it means that the obtained Pareto set is the actual Pareto set. Diversity (Δ) is used to evaluate the diversity among the obtained non-dominated solutions using Eq. (41), where n is the number of solutions, h_i is the distance between two adjacent solutions, \bar{h} is the average value of h_i , h_f and h_1 are distances between extreme solutions and boundary solutions, respectively. According to this metric, the smaller the diversity, the better is the diverse set of the obtained solution.

$$\Delta = \frac{h_f + h_1 + \sum_{i=1}^{n-1} |h_i - \bar{h}|}{h_f + h_1 + (n - 1)\bar{h}}. \tag{41}$$

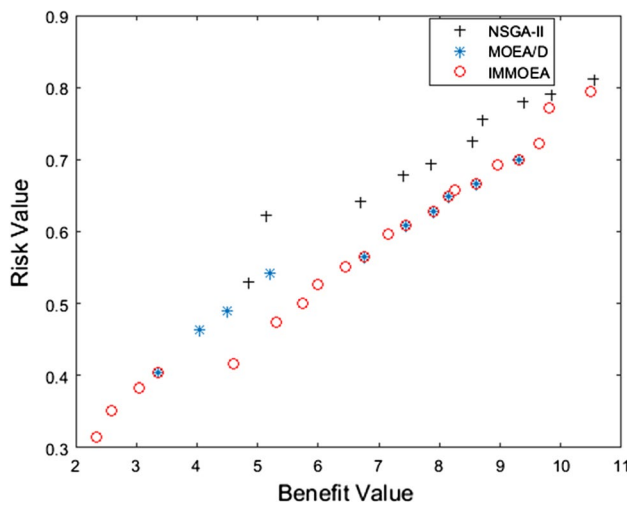


Fig. 4 The best results of three algorithms

GD is used to describe the distance between the obtained Pareto set and the true Pareto set using (42), where $dist_i$ is the shortest distance between the i th Pareto set and the true Pareto set. This one is used to evaluate not only convergence, but also diversity. In the following equation, a lower value is better.

$$GD = \frac{\sqrt{\sum_{i=1}^n dist_i^2}}{n} \tag{42}$$

There are two major objectives in this example, listed as benefit and risk. The former one illustrates the sum of all the overall benefit values that a portfolio solution can provide in a given period, and the latter one represents the probability

of having at least one project failing to provide its expected benefit throughout the entire period.

In what follows, set the population size at $Pop = 100$, the total number of generations at $MaxGen = 500$, and the random group size $L = 3$. Then, each algorithm is independently executed 10 times and its best results are recorded as shown in Fig. 4.

As shown in Fig. 4, the bottom-left points of IMMOEA, MOEA/D, and NSGA-II are (2.35, 0.32), (3.35, 0.40), and (4.85, 0.53), respectively. The top-right points of these three algorithms are (10.50, 0.79), (9.30, 0.70), and (10.55, 0.81), respectively. It is obvious that the Pareto set obtained by IMMOEA has the longest distribution curve among the three, followed by MOEA/D and NSGA-II. Hence, the Pareto set obtained by IMMOEA contains more portfolio solutions.

In most situations, the exact Pareto set is not known, and consequently, it is necessary to propose a reference set for the calculation of the above three metrics. To this aim, the above three algorithms are independently executed 10 times, and the best Pop non-dominated solutions are taken as the reference set. Moreover, $Pop * 10$ solutions are obtained for each algorithm. Then, the best, worst, mean, and standard deviation (std) among these solutions for each algorithm are recorded, as tabulated in Table 5. For all of the three metrics, a lower value is better.

As can be seen, the worst occurs with NSGA-II for all different measures for the three metrics compared with that of IMMOEA and MOEA/D, though the best value in convergence is not the worst. IMMOEA has the smallest mean value of the convergence metric, which means that IMMOEA has the best convergence. As for the diversity metric, IMMOEA also outperforms the others, followed by MOEA/D and NSGA-II. The best, worst, and mean values of GD for these measures having the smallest values are also recorded for IMMOEA, which in a way indicates the good convergence and diversity of IMMOEA. Moreover, Table 6 displays the average time for each round of these different algorithms. Note that the efficiency of IMMOEA is more superior to the other algorithms. Therefore, IMMOEA stands out from the other two algorithms regarding convergence and diversity, which means that it can produce more robust solutions in a shorter time.

After obtaining the Pareto set by IMMOEA, we continue the process to obtain a compromise solution using the TOPSIS approach [30], for which the criteria are benefit and risk.

Table 5 Statistical comparisons among three different algorithms

Metrics	Measures	Performances		
		IMMOEA	MOEA/D	NSGA-II
γ (convergence)	Best	0.0101	0.0140	0.0079
	Worst	0.1378	0.1318	0.2772
	Mean	0.0523	0.0555	0.1471
	Std	0.0342	0.0451	0.0973
Δ (diversity)	Best	0.0410	1.3551	1.6742
	Worst	0.0955	1.5918	1.8403
	Mean	0.0738	1.4982	1.7779
	Std	0.0184	0.0694	0.0470
GD (generational distance)	Best	0.0035	0.0056	0.0084
	Worst	0.0186	0.0283	0.0474
	Mean	0.0102	0.0161	0.0250
	Std	0.0039	0.0087	0.0151

Table 6 Average computation time (t) among the three different algorithms (in seconds)

Algorithms	IMMOEA	MOEA/D	NSGA-II
t/s	10.44	26.05	45.44

Table 7 Details of the compromise solution

Project	Selected	Year				
		First	Second	Third	Fourth	Fifth
1	Yes	✓				
2	Yes				✓	
3	No					
4	Yes			✓		
5	Yes	✓				
6	Yes			✓		
7	No					
8	Yes		✓			
9	No					
10	No					

Table 8 Statistical data at different confidence levels

α	Benefit	Risk	SolutionNum
1	10.30	0.77	84
0.9	10.35	0.78	80
0.8	11.25	0.83	77
0.7	11.00	0.79	77
0.6	11.70	0.84	71
0.5	12.40	0.86	65
0.4	11.25	0.83	63
0.3	11.25	0.83	64
0.2	11.70	0.84	56
0.1	12.45	0.87	54
0	13.85	0.92	49

The main idea of this MCDM approach lies in selecting the most appropriate alternative that is the nearest to the positive ideal solution, while farthest from the negative ideal one as well. Set the relative importance of these two criteria as 0.6 and 0.4, for the benefit criterion and risk criterion, respectively. Note that various algorithms and methodologies, especially preferences based multi-objective optimization, have been designed in recent decades to obtain a representative subset of the Pareto front or even certain compromise solutions [31].

Details of the compromise solution are shown in Table 7. The symbol of “✓” indicates the starting time of its corresponding project. As shown in Table 7, six out of ten projects are chosen to be developed throughout the given period. More specifically, Projects 1 and 5 are selected to be developed in the first year, whereas Project 8 is developed in the second year, Projects 4 and 6 in the third year, and Project 2 in the fourth year. Taking the duration of each project into consideration, the numbers of projects existing in each year are 2, 3, 3, 4, and 2, respectively. Note that Projects 4 and 6 occur simultaneously in the third and fourth years. Thus, additional benefit can be raised from the interactions between them.

As mentioned above, fuzzy numbers are used in this model, and the confidence level α is utilized to transform them into precise values. It is logical to think that different efficient frontiers can be obtained at different confidence levels given by DMs. Thus, we continue the experiment using IMMOEA at different confidence levels for comparison purposes, while the other parameters used in the algorithm remain the same. The algorithm is re-executed 10 times at each value of α to get a 100×10 Pareto sets. However, some of the solutions are the same, the number of unique Pareto solutions under different circumstances need to be identified. After that, TOPSIS approach is used to get the compromise solutions under each circumstance, whose values associated with the two objectives are listed in Table 8. The solution number of each Pareto set varies each other. The statistical results are shown in Table 8.

It is apparent that when α equals 1, the fuzzy resource constraints are transformed into precise values. As α decreases, the number of total unique solutions under different confidence levels decreases as well. The variation range of the compromise solutions at different confidence levels is [10.30, 13.85] associated with the benefit value, and [0.77, 0.92] for the risk value. Actually, the compromise solutions obtained at different confidence levels are quite similar. For example, the corresponding compromise solutions are the same when α equals 0.8, 0.4, and 0.3. The same situation occurs when α is equal to 0.6 and 0.2. However, the unique solution number obtained when α equals 0.2 is lower than that of 0.6. In fact, as α decreases, the fuzzy resource constraints become more and more restrictive. Thus, the unique solution numbers should decrease as well. Note that the efficient Pareto solutions obtained at a higher confidence level are also feasible for any lower confidence level. Therefore, some of the solutions may be a part of different Pareto frontiers even under different confidence levels.

6 Conclusions

PPSS is an important strategic problem having significant impacts on the running efficiency of organizations. In this paper, a mathematical PPSS model is constructed as a bi-objective optimization problem under a fuzzy environment. As an extension to existing studies, the model focuses on maximizing various solutions' overall benefits and lowering the risk values simultaneously. Specific resource limitations and DMs' preferences are taken into account. To address the uncertainties regarding information, fuzzy numbers are also incorporated into the model, enabling DMs with more flexibility.

A hybrid algorithm called IMMOEA is used and modified to obtain the Pareto set, which is the first time that such an algorithm has been used to solve the PPSS problem. Three algorithms are employed in an example and different metrics are used simultaneously to compare their efficacy. The results show that IMMOEA outperforms MOEA/D and NSGA-II with respect to diversity, spread, and convergence. NSGA-II is the worst compared with the other two algorithms used in this study. Finally, details regarding the corresponding compromise solution are determined. Certain conclusions and suggestions are proposed based on the calculations. Hence, this research can provide project managers with additional insights into this PPSS procedure.

In future studies, the authors will consider projects with multiple activity relationships, various constraints, and even dynamic events occurring during the project execution time. These challenges will certainly contribute to the expansion of this methodology for tackling a broader range of problems.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China under Grants 71690233, 71501182, and 71571185. The authors would like to thank the editor and three anonymous reviewers for their constructive comments that helped us to improve the quality of this paper.

References

- Hall NG, Long DZ, Qi J, Sim M (2015) Managing under-performance risk in project portfolio selection. *Oper Res* 63(3):660–675
- Starkey A, Hagrais H, Shakya S, Owusu G, Mohamed A, Alghazawi D (2016) A cloud computing based many objective type-2 fuzzy logic system for mobile field workforce area optimization. *Memet Comput* 8(4):269–286
- Zhang X, Ge B, Jiang J, Tan Y (2016) Consensus building in group decision making based on multiplicative consistency with incomplete reciprocal preference relations. *Knowl Based Syst* 106:96–104
- Kitayama S, Yamazaki K (2012) Compromise point incorporating trade-off ratio in multi-objective optimization. *Appl Soft Comput* 12(8):1959–1964
- Vetschera R, De Almeida AT (2012) A PROMETHEE-based approach to portfolio selection problems. *Comput Oper Res* 39(5):1010–1020
- Tavana M, Keramatpour M, Santos-Arteaga FJ, Ghorbaniane E (2015) A fuzzy hybrid project portfolio selection method using data envelopment analysis, TOPSIS and integer programming. *Expert Syst Appl* 42(22):8432–8444
- Morton A, Keisler JM, Salo A (2016) Multicriteria portfolio decision analysis for project selection. *Multiple criteria decision analysis*. Springer, New York, NY, pp 1269–1298
- Jha P, Bali S, Kumar UD, Pham H (2014) Fuzzy optimization approach to component selection of fault-tolerant software system. *Memet Comput* 6(1):49–59
- Hassanzadeh F, Nemati H, Sun M (2014) Robust optimization for interactive multiobjective programming with imprecise information applied to R&D project portfolio selection. *Eur J Oper Res* 238(1):41–53
- Schaeffer S, Cruz-Reyes L (2016) Static R&D project portfolio selection in public organizations. *Decis Support Syst* 84:53–63
- Sefair JA, Méndez CY, Babat O, Medaglia AL, Zuluaga LF (2017) Linear solution schemes for Mean-SemiVariance Project portfolio selection problems: an application in the oil and gas industry. *Omega* 68:39–48
- Chen W, Lei L, Wang Z, Teng M, Liu J (2018) Coordinating supplier selection and project scheduling in resource-constrained construction supply chains. *Int J Prod Res* 56(19):6512–6526
- Rostami S, Creemers S, Leus R (2018) New strategies for stochastic resource-constrained project scheduling. *J Sched* 21(3):349–365
- Wang X, Ning Y (2018) Uncertain chance-constrained programming model for project scheduling problem. *J Oper Res Soc* 69(3):384–391
- Leyman P, Vanhoucke M (2017) Capital-and resource-constrained project scheduling with net present value optimization. *Eur J Oper Res* 256(3):757–776
- Esparcia-Alcázar AI, Almenar F, Vos TE, Rueda U (2018) Using genetic programming to evolve action selection rules in traversal-based automated software testing: results obtained with the TESTAR tool. *Memet Comput* 10(3):257–265
- Yassine AA, Mostafa O, Browning TR (2017) Scheduling multiple, resource-constrained, iterative, product development projects with genetic algorithms. *Comput Ind Eng* 107:39–56
- Xiong J, Leus R, Yang Z, Abbas HA (2016) Evolutionary multi-objective resource allocation and scheduling in the Chinese navigation satellite system project. *Eur J Oper Res* 251(2):662–675
- Kumar M, Mittal M, Soni G, Joshi D (2018) A hybrid TLBO-TS algorithm for integrated selection and scheduling of projects. *Comput Ind Eng* 119:121–130
- Perez F, Gomez T (2016) Multiobjective project portfolio selection with fuzzy constraints. *Ann Oper Res* 245:7–29
- Tanaka H (1984) A formulation of fuzzy linear programming problem based on comparison of fuzzy numbers. *Control cybern* 3:185–194
- Cheng R, Jin Y, Narukawa K, Sendhoff B (2015) A multiobjective evolutionary algorithm using Gaussian process-based inverse modeling. *IEEE Trans Evol Comput* 19(6):838–856
- Karshenas H, Santana R, Bielza C, Larranaga P (2014) Multi-objective estimation of distribution algorithm based on joint modeling of objectives and variables. *IEEE Trans Evol Comput* 18(4):519–542
- Liu H-L, Gu F, Zhang Q (2014) Decomposition of a multiobjective optimization problem into a number of simple multiobjective subproblems. *IEEE Trans Evol Comput* 18(3):450–455
- Yang Z, Tang K, Yao X (2008) Large scale evolutionary optimization using cooperative coevolution. *Inf Sci* 178(15):2985–2999

26. Campigotto P, Passerini A, Battiti R (2014) Active learning of Pareto fronts. *IEEE Trans Neural Netw Learn Syst* 25(3):506–519
27. Cheng R, Jin Y, Olhofer M, Sendhoff B (2016) A reference vector guided evolutionary algorithm for many-objective optimization. *IEEE Trans Evol Comput* 20(5):773–791
28. Zhang Q, Li H (2007) MOEA/D: A multiobjective evolutionary algorithm based on decomposition. *IEEE Trans Evol Comput* 11(6):712–731
29. Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 6(2):182–197
30. Rudnik K, Kacprzak D (2017) Fuzzy TOPSIS method with ordered fuzzy numbers for flow control in a manufacturing system. *Appl Soft Comput* 52:1020–1041
31. Wang H, Olhofer M, Jin Y (2017) A mini-review on preference modeling and articulation in multi-objective optimization: current status and challenges. *Complex Intell Syst* 3(4):233–245

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.