Journal of Ambient Intelligence and Humanized Computing (2023) 14:2915–2934
https://doi.org/10.1007/s12652-023-04532-w

ORIGINAL RESEARCH

Decision making framework for heterogeneous QoS information: an application to cloud service selection

Rohit Kumar Tiwari[1] · **Rakesh Kumar**[1] · **Gaurav Baranwal**[2] · **Rajkumar Buyya**[3]

Received: 18 October 2020 / Accepted: 10 January 2023 / Published online: 2 February 2023

Abstract
In recent times, appropriate decision-making in challenging and critical situations has been very well supported by multicriteria decision-making (MCDM) methods. The technique for order of preference by similarity to ideal solution (TOPSIS) is the most widely used MCDM method for solving decision problems. However, it restricts decision-makers to use only one type of Quality of Service (QoS) information, and it suffers from the rank reversal problem. Restriction to only one type of QoS makes the decision problems more challenging, as it restricts the decision-makers freedom. Further, the rank reversal problem makes the decision result unreliable. To address these issues of TOPSIS, we have proposed a reliable rank reversal robust modular TOPSIS (RMo-TOPSIS). RMo-TOPSIS allows crisp, interval, fuzzy, intuitionistic and neutrosophic fuzzy QoS metrics. It does not suffer from the rank reversal problem. Cloud computing provides computing services on-demand basis without involving maintenance by its users. The availability of many cloud service providers and their services makes cloud service selection a challenging problem. To validate RMo-TOPSIS, we select the cloud service selection consisting of different types of QoS metrics as an application. Experiments on cloud service selection show consistency and accuracy in results obtained by RMo-TOPSIS and its robustness against rank reversal.

Keywords Decision making · TOPSIS · MCDM · Heterogeneous QoS · Cloud service selection · Rank reversal

1 Introduction

With the increase in the size of the business organization, the executive body of the organizations has to take decisions in a complex environment to avoid the risk. They take the help of decision-making techniques or optimization techniques to solve the decision problems. Optimization techniques are used to find the optimal solution in many fields like electricity load and price forecasting (Saeedi et al. 2019; Abedinia et al. 2019; Gao et al. 2019a, b; Ghadimi et al. 2018; Khodaei et al. 2018), service selection, etc. MCDM is a widely used decision-making approach in many fields like automobile (Yousefi and Hadi-Vencheh 2010; Yang et al. 2020), defense (Wu and Mendel 2010), web service (Purohit 2020), cloud service selection (Hussain et al. 2020a, b; Hussain et al. 2020a, b; Garg et al. 2013), etc. to find the optimal alternative or service to carry out business. Recently, in the web service domain, Artificial Intelligence (AI) enabled methods, e.g. neural collaborative filtering-based approaches (Chowdhury et al. 2020; Regunathan et al. 2018; Xu et al. 2021; Gao et al. 2019a, b; Huang et al. 2021), have been used for QoS prediction and recommendation in web service selection. One of the major limitations of such collaborative filtering-based systems is that they require good quality of data in a good amount. However, in many domains such as cloud service selection, weapon selection, etc., a limited amount of data related to alternatives is available; therefore, collaborative filtering is not possible, i.e. AI-enabled methods cannot be used. So, MCDM plays a crucial role in decision-making in such domains.

MCDM is a widely used decision-making approach in many fields like automobile (Yousefi and Hadi-Vencheh 2010; Yang et al. 2020), defense (Wu and Mendel 2010), web service (Purohit 2020), cloud service selection (Hussain

✉ Rohit Kumar Tiwari
rohitkushinagar@gmail.com

[1] Department of Computer Science and Engineering, Madan Mohan Malaviya University of Technology, Gorakhpur 273010, India

[2] Department of Computer Science, BHU, Varanasi, India

[3] Cloud Computing and Distributed Systems Laboratory, School of Computing and Information Systems, University of Melbourne, Melbourne, Australia

et al. 2020a, b; Hussain et al. 2020a, b; Garg et al. 2013), etc. to find the optimal alternatives or service to carry out business. It is a collection of methods such as the technique for order of preference by similarity to ideal solution (TOPSIS) (Yoon and Hwang 1981), analytic network process (ANP) (Saaty 2006), analytic hierarchy process (AHP) (Saaty 1990), and elimination and choice translating reality (ELECTRE) (Roy 1990) to solve the decision problem in case of qualitative, quantitative and inter-dependent criteria. TOPSIS is one of the most common MCDM methods due to its simplicity and efficiency. It finds the best alternative based on its distance from the ideal and worst alternative. The alternative having maximum geometric distance from worst and minimum from ideal is the optimal alternative. However, the original TOPSIS method proposed by Hwang and Yoon works only for decision matrices with crisp values and causes the rank reversal problem in some scenarios (Wang and Luo 2009; García-Cascales and Lamata 2012). In the rank reversal phenomenon, the rank of alternative changes on the addition or removal of another alternative to the decision problem i.e. an alternative which is optimal before the addition or removal of non-optimal alternatives becomes non-optimal after their addition or removal. Since, the TOPSIS is the most widely used MCDM method for decision-makers and has been used in many fields like defense, automobile, service selection, etc. So, the use of TOPSIS may cause a huge loss to the organization's credibility and capital due to its rank reversal problem.

There are many generalizations of standard TOPSIS to address its limitation of operating on crisp decision matrix only. Yue (2011) proposed an extension of the TOPSIS method for a decision matrix based on interval numbers. Later Jahanshahloo et al. (2009) and Dymova et al. (2013) also proposed its generalization for interval numbers. Similarly, many generalizations of TOPSIS were proposed by many authors for fuzzy (Chen 2000; Krohling and Campanharo 2011; Lee et al. 2014), intuitionistic fuzzy (Boran et al. 2009), hesitant fuzzy (Xu and Zhang 2013), spherical fuzzy (Kutlu Gündoğdu, and Kahraman 2019) and neutrosophic fuzzy (Biswas et al. 2016; Abdel-Basset et al. 2019) environment. Wang et al. (2022) proposed a generalization of TOPSIS called three-way TOPSIS to effectively solve the decision-making problem. Keikha (2022) proposed a generalized hesitant fuzzy number based TOPSIS method to solve the decision problem modeled considering the uncertainty. Akram et al. (2021) proposed an extension of the TOPSIS to model the decision problem in a complex spherical fuzzy information scenario. Monika (2022) modeled the cloud service selection problem using TOPSIS with a 3-D spherical fuzzy set to remove vulnerability in the decision-making process of decision-makers. Sadabadi et al. (2022) proposed an improved fuzzy TOPSIS using triangular fuzzy numbers by removing the shortcomings of fuzzy TOPSIS. However, the above generalizations work only for a decision matrix with one type of data. However, it is not always the case that decision-makers will have a decision matrix with only one type of data due to diversity in their knowledge and information available to decide, e.g. in the case of a route selection problem, where the distance is fixed or quantifiable but traffic on the route is qualitative and depends on personal familiarity with it. Due to the availability of various types of information in different forms like crisp, fuzzy, interval, neutrosophic, etc. for a particular decision problem, it has become a challenge to make a decision with standard TOPSIS or its generalization of a specific type. Many authors have proposed the generalization of TOPSIS to deal with a heterogeneous decision matrix. Peng et al. (2012) proposed a TOSPIS based group decision-making technique for crisp, interval and linguistic types of information, where each decision matrix comprises a specific type of information and is transformed into the triangular fuzzy decision matrix. All transformed decision matrices were aggregated into a single matrix using ordered weighted averaging (OWA) operator and fuzzy TOPSIS was applied to find an optimal alternative. Wang and Wang (2008) extended the TOPSIS for numerical, interval, linguistic and fuzzy information. They used the transformation function to map a data type into another data type and used TOPSIS on a unified decision matrix to find the best alternative. Li et al. (2010) proposed a method similar to TOPSIS for decision making in heterogeneous QoS but it was complex and unable to rank the alternatives accurately (Lourenzutti and Krohling 2016). Lourenzutti and Krohling (2016) proposed an extension of TOPSIS called modular TOPSIS (Mo-TOPSIS) for decision matrix with crisp, interval, triangular fuzzy number and intuitionistic fuzzy set information without any transformation function.

There are extensions of other MCDM methods for heterogeneous QoS environments. An extension of TODIM by Fan et al. (2013) finds the best alternative on decision matrix with crisp, interval and triangular fuzzy number. They used the cumulative distribution function to transform three data types into a random variable and used the gain–loss concept of the classical TODIM method to find the dominance of an alternative over others to rank them. Liu et al. (2020) extended the ELECTRE method for decision matrix with heterogeneous QoS of type crisp, interval, TFN and IFS. They used transformation functions to map crisp, interval and TFN into intuitionistic fuzzy number and ranked the cloud services from a unified decision matrix using ELECTRE. Espinilla et al. (2013) proposed an MCDM method for heterogeneous QoS information using transformation functions to transform data type into a particular data type.

Although, the above studies can deal with heterogeneous QoS information, but most of them use transformation functions to make a unified decision matrix. The transformation of one data type to another causes some information loss and affects decision-making (Lourenzutti and Krohling 2016). The method Mo-TOPSIS proposed by Lourenzutti and Krohling (2016) does not use any transformation functions. However, Mo-TOPSIS suffers from rank reversal phenomenon, which may affect its use in decision problems that are very critical like defense, hazardous material route selection, etc. It cannot be also used in a decision involving heavy capital expenditure like automobile, manufacturing, cloud service selection, etc. To show the rank reversal in Mo-TOPSIS, we implemented it. We monitored the cases of rank reversal by dynamically changing the number of alternatives from the original decision matrix by removing some alternatives. Table 1 shows the decision matrix with heterogeneous QoS and their weight, where QoS_1 and QoS_2 are costly QoS parameters while others are benefit parameters. It also shows the closeness index obtained using Mo-TOPSIS and rank of alternatives. We can observe that alternative A_4 is the best alternative as per QoS weight and decision matrix. We removed alternative A_3 from the decision matrix of Table 1 and computed the closeness index of each alternative and ranked them to monitor the rank reversal. Table 2 shows the decision matrix, closeness index and each alternative's rank on the removal of A_3. We can observe that like TOPSIS, Mo-TOPSIS also suffers from rank reversal problem as A_4, which was ranked first is now ranked second after removal of non-optimal alternative A_3. Now, A_1 is ranked best even A_4 is present in the decision matrix.

The rank reversal phenomenon in Mo-TOPSIS is further investigated by performing a series of experiments. We considered the decision matrix with alternatives ranging from four to eight and monitored rank reversal. We randomly removed 1 to $n-2$ alternatives from the decision matrix with n alternatives. Initially, we considered a decision matrix with four alternatives i.e. $n=4$ and created 10 (${}^4C_1+{}^4C_2$) decision matrices by removing one and two alternatives from the original and found the number of cases causing rank reversal. The same process is carried out for the decision matrix with $n=4, 5, 6, 7$ *and* 8 alternatives. The result obtained is shown in Fig. 1. We can observe that the rank reversal cases increase with decision matrix size and depend on entries in the decision matrix.

Therefore, considering the rank reversal problem in Mo-TOPSIS for heterogeneous QoS without any data transformation functions and wide use of TOPSIS in research, an extension of Mo-TOPSIS called robust Mo-TOPSIS (RMo-TOPSIS) is developed that is robust to rank reversal phenomenon. The RMo-TOPSIS uses different processes of computing normalized decision matrix, positive ideal solution (PIS) and negative ideal solution (NIS) to avoid rank reversal. RMo-TOPSIS is robust and efficient than the existing Mo-TOPSIS and will help decision-makers to avoid any type of risk in decision. Contributions of this work are as follows:

1. A rank reversal robust RMo-TOPSIS method for heterogeneous QoS information without any data transformation functions is proposed and compared with state-of-the-art works.
2. A sensitivity analysis is performed to show the robustness of RMo-TOPSIS against the rank reversal phenomenon.
3. A case study of RMo-TOPSIS in cloud service selection as its application in decision making is given to demonstrate its effectiveness.

The rest of this paper proceeds as follows. Section 2 describes the various preliminary concept used to develop the RMo-TOPSIS while Sect. 3 explains the proposed

Table 1 Decision matrix with heterogeneous QoS and rank of alternatives using Mo-TOPSIS

Alternatives	QoS_1 (crisp)	QoS_2 (interval)	QoS_3 (crisp)	QoS_4 (TFN)	QoS_5 (IFS)	Closeness index	Rank
QoS weight	0.23	0.23	0.24	0.24	0.06		
A_1	126	[186, 295]	0.95	[0.5, 0.6, 0.7]	[0.65, 0.25]	0.5425	2
A_2	107	[182, 280]	0.85	[0.3, 0.5, 0.7]	[0.30, 0.65]	0.4543	3
A_3	140	[210, 330]	0.98	[0.4, 0.6, 0.9]	[0.50, 0.35]	0.4473	4
A_4	138	[197, 312]	0.93	[0.5, 0.7, 1.0]	[0.45, 0.50]	0.5543	1

Table 2 The rank of alternatives after removal of alternative A_3 using Mo-TOPSIS

Alternatives	QoS_1 (crisp)	QoS_2 (interval)	QoS_3 (crisp)	QoS_4 (TFN)	QoS_5 (IFS)	Closeness index	Rank
QoS weight	0.23	0.23	0.24	0.24	0.06		
A_1	126	[186, 295]	0.95	[0.5, 0.6, 0.7]	[0.65, 0.25]	0.5319	1
A_2	107	[182, 280]	0.85	[0.3, 0.5, 0.7]	[0.30, 0.65]	0.4423	3
A_4	138	[197, 312]	0.93	[0.5, 0.7, 1.0]	[0.45, 0.50]	0.5314	2

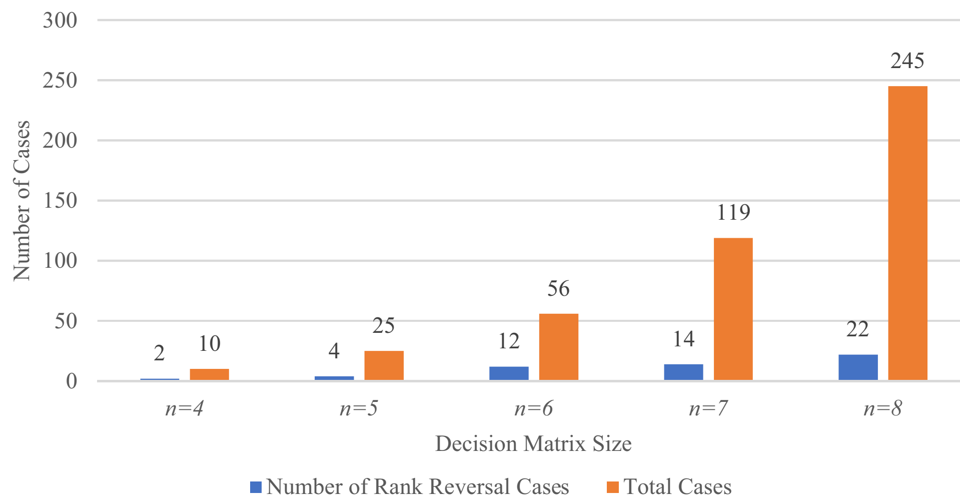

Fig. 1 Variation of rank reversal cases with decision matrix size

method for decision making with heterogeneous QoS information. Section 4 presents an application of RMo-TOPSIS in cloud service selection through a case study. It also shows the correctness and robustness of the proposed method through sensitivity analysis. Finally, the conclusion and future directions are discussed in Sect. 5.

2 Preliminaries

In this section, various concepts used for developing TOPSIS for the heterogeneous QoS are discussed.

2.1 Fuzzy set theory

In fuzzy set theory, each element of the set has some membership degree along with its value. It is the extension of classical set theory where the membership degree of the element represents the degree of its belongingness to the fuzzy set. It has been extensively used to handle uncertainty and vagueness in various problems. It helps decision-makers or experts to provide their judgment in linguistic terms. The various concepts and definitions used in the fuzzy set are explained below.

Definition 1 (Zadeh 1965) Let X is a universe of discourse, then a fuzzy set A is a collection of elements $x\epsilon X$ and membership function $\mu_A(x)$ characterizes each x. A fuzzy set A is denoted by-

$$A = \{x, \mu_A(x) | x\epsilon X\} \tag{1}$$

where

$\mu_A(x) : X \rightarrow [0, 1]$

Definition 2 A triangular fuzzy number (TFN) N is a triplet of real numbers represented as (α, β, γ) where α, β and γ represents the least, most and highest possible values. If $\alpha = \beta = \gamma$ then N is a crisp number. The membership function $\mu_N(x)$ for each x in the fuzzy set is computed using TFN N is defined as

$$\mu_N(x) = \begin{cases} \frac{x-\alpha}{\beta-\alpha} & if\ x \in [\alpha, \beta] \\ \frac{\gamma-x}{\gamma-\beta} & if\ x \in (\beta, \gamma] \\ 0 & otherwise \end{cases} \tag{2}$$

Definition 3 Let $N_1 = (\alpha_1, \beta_1, \gamma_1)$ and $N_2 = (\alpha_2, \beta_2, \gamma_2)$ are two TFNs, λ is a real number then the various arithmetic operation defined on N_1 and N_2 are-

$$N_1 + N_2 = (\alpha_1 + \alpha_2, \beta_1 + \beta_2, \gamma_1 + \gamma_2) \tag{3}$$

$$N_1 - N_2 = (\alpha_1 - \alpha_2, \beta_1 - \beta_2, \gamma_1 - \gamma_2) \tag{4}$$

$$N_1 * N_2 = (\alpha_1 * \alpha_2, \beta_1 * \beta_2, \gamma_1 * \gamma_2) \tag{5}$$

$$\lambda * N_1 = \left(\lambda\alpha_1, \lambda\beta_1, \lambda\gamma_1\right) \tag{6}$$

Example 1 Let A = (−3, 2, 4) and B = (1, 0, 6) are two TFN, then the various operations discussed above can be performed as-

$$A + B = (-3 + 1, 2 + 0, 4 + 6) = (-2, 2, 10)$$

$$A - B = (-3 - 1, 2 - 0, 4 - 6) = (-4, 2, -2)$$

$$A * B = (-3 * 1, 2 * 0, 4 * 6) = (-3, 0, 24)$$

If $\lambda = 0.5$ then

$$\lambda A = (0.5 * -3, 0.5 * 2, 0.5 * 4) = (-1.5, 1, 2)$$

Definition 4 (Dağdeviren 2009) The Euclidean distance between the set of TFNs $A = \{(\alpha_1^A, \beta_1^A, \gamma_1^A), (\alpha_2^A, \beta_2^A, \gamma_2^A) \dots (\alpha_n^A, \beta_n^A, \gamma_n^A)\}$ and $B = \{(\alpha_1^B, \beta_1^B, \gamma_1^B), (\alpha_2^B, \beta_2^B, \gamma_2^B) \dots (\alpha_n^B, \beta_n^B, \gamma_n^B)\}$ is computed using Eq. (7).

$$d(A,B) = \sqrt{\sum_{i=1}^{n} \frac{1}{3}\left[\left(\alpha_i^A - \alpha_i^B\right)^2 + \left(\beta_i^A - \beta_i^B\right)^2 + \left(\gamma_i^A - \gamma_i^B\right)^2\right]} \quad (7)$$

2.2 Intuitionistic fuzzy set

An intuitionistic fuzzy set (IFS) is an extension of classical fuzzy set theory. It is helpful in an environment where the decision-maker lacks knowledge. IFS provides flexibility to decision-makers to give their judgment in membership and non-membership degree both. The membership and non-membership degree represent the truth and falsity involve in the rating of the decision process. The various concepts and operations performed on IFS are discussed below.

Definition 5 (Atanassov 1986) Let X is a universe of discourse, then an IFS A is a collection of elements $x \epsilon X$ and each x is characterized by the membership function $\mu_A(x)$ and non-membership function $\nu_A(x)$. An IFS A is denoted by-

$$A = \{\langle x, \mu_A(x), \nu_A(x)\rangle | x \epsilon X\} \quad (8)$$

where

$$\mu_A(x) : X \to [0,1]$$

$$\nu_A(x) : X \to [0,1]$$

$$0 \le \mu_A(x) + \nu_A(x) \le 1$$

The hesitancy degree $\pi_A(x)$ for $x \epsilon A$ is defined as

$$\pi_A(x) = 1 - \mu_A(x) - \nu_A(x)$$

Definition 6 (Atanassov 1994) Let $A = \langle x, \mu_A(x), \nu_A(x)\rangle$ and $B = \langle x, \mu_B(x), \nu_B(x)\rangle$ are two IFS, λ is a real number then the various arithmetic operation defined on A and B are-

$$A + B = \langle x, \mu_A(x) + \mu_B(x) - \mu_A(x) * \mu_B(x), \nu_A(x) * \nu_B(x)\rangle \quad (9)$$

$$A.B = \langle x, \mu_A(x) * \mu_B(x), \nu_A(x) + \nu_B(x) - \nu_A(x) * \nu_B(x)\rangle \quad (10)$$

$$\lambda * A = \langle x, 1 - \left(1 - \mu_A(x)\right)^\lambda, (\nu_A(x))^\lambda\rangle \quad (11)$$

$$A^\lambda = \langle x, (\mu_A(x))^\lambda, 1 - \left(1 - \nu_A(x)\right)^\lambda\rangle \quad (12)$$

Definition 7 Let $A = \{\langle x_1, \mu_A(x_1), \nu_A(x_1)\rangle, \langle x_2, \mu_A(x_2), \nu_A(x_2)\rangle, \dots \langle x_n, \mu_A(x_n), \nu_A(x_n)\rangle\}$ and $B = \{\langle x_1, \mu_B(x_1), \nu_B(x_1)\rangle, \langle x_2, \mu_B(x_2), \nu_B(x_2)\rangle, \dots \langle x_n, \mu_B(x_n), \nu_B(x_n)\rangle\}$ be two IFS vectors of the length of n for $X = \{x_1, x_2 \dots, x_n\}$, then the Euclidean distance between A and B is defined as-

$$d(A,B) = \sqrt{\sum_{i=1}^{n} \frac{1}{2}\left[\left(\mu_A(x_i) - \mu_B(x_i)\right)^2 + \left(\nu_A(x_i) - \nu_B(x_i)\right)^2\right]} \quad (13)$$

2.3 Neutrosophic fuzzy set

Neutrosophic fuzzy set (NFS) is an extension of IFS where each element of NFS has an indeterminacy degree along with truth and falsity degree. It helps experts to express their opinion where the expert is not confident about his decision or not familiar with the decision problem. The various concepts and operations used in NFS theory are discussed below through various definitions.

Definition 8 Let X is a universe of discourse, then the NFS A is a collection of all $x \epsilon X$ where each x is characterized by truth $\mathcal{T}_A(x)$, indeterminacy $\mathcal{I}_A(x)$, and falsity $\mathcal{F}_A(x)$ membership function. The NFS A is denoted as-

$$A = \{\langle x, \mathcal{T}_A(x), \mathcal{I}_A(x), \mathcal{F}_A(x)\rangle | x \epsilon X\} \quad (14)$$

where

$$\mathcal{T}_A(x) : X \to]^-0, 1^+[$$

$$\mathcal{I}_A(x) : X \to]^-0, 1^+[$$

$$\mathcal{F}_A(x) : X \to]^-0, 1^+[$$

$$0^- \le \mathcal{T}_A(x) + \mathcal{I}_A(x) + \mathcal{F}_A(x) \le 3^+$$

$^-0 = 0 - \varepsilon, 1^+ = 1 + \varepsilon, \varepsilon$ is an infinitesimal number

Definition 9 (Wang et al. 2010) Let X is a universe of discourse, then the single value neutrosophic set (SVNS) A is collection of all such $x \epsilon X$ where each x is characterized by truth $\mathcal{T}_A(x)$, indeterminacy $\mathcal{I}_A(x)$, and falsity $\mathcal{F}(x)$ membership function. The SVNS A is denoted as-

$$A = \left\{ \langle x, \mathcal{T}_A(x), \mathcal{I}_A(x), \mathcal{F}_A(x) \rangle \middle| x \epsilon X \right\} \tag{15}$$

where

$$\mathcal{T}_A(x) : X \to [0, 1]$$

$$\mathcal{I}_A(x) : X \to [0, 1]$$

$$\mathcal{F}_A(x) : X \to [0, 1]$$

$$0 \le \mathcal{T}_A(x) + \mathcal{I}_A(x) + \mathcal{F}_A(x) \le 3$$

Definition 10 (Ye 2015) Let $A = \langle x, \mathcal{T}_A(x), \mathcal{I}_A(x), \mathcal{F}_A(x) \rangle$ and $B = \langle x, \mathcal{T}_B(x), \mathcal{I}_B(x), \mathcal{F}_B(x) \rangle$ are two SVNS, α is a real number then the various arithmetic operation defined on A and B are-

$$A \oplus B = \langle x, \mathcal{T}_A(x) + \mathcal{T}_B(x) - \mathcal{T}_A(x)\mathcal{T}_B(x), \mathcal{I}_A(x)\mathcal{I}_B(x), \mathcal{F}_A(x)\mathcal{F}_B(x) \rangle \tag{16}$$

$$A \otimes B = \langle x, \mathcal{T}_A(x)\mathcal{T}_B(x), \mathcal{I}_A(x) + \mathcal{I}_B(x) - \mathcal{I}_A(x)\mathcal{I}_B(x), \mathcal{F}_A(x) + \mathcal{F}_B(x) - \mathcal{F}_A(x)\mathcal{F}_B(x) \rangle \tag{17}$$

$$\alpha A = \langle x, 1 - \left(1 - \mathcal{T}_A(x)\right)^{\alpha}, (\mathcal{I}_A(x))^{\alpha}, (\mathcal{F}_A(x))^{\alpha} \rangle \ for\ \alpha > 0 \tag{18}$$

$$\phi_2\left(A_i, B_i\right) = \max\left(\frac{2 + \mathcal{T}_A\left(x_i\right) - \mathcal{I}_A\left(x_i\right) - \mathcal{F}_A\left(x_i\right)}{3}, \frac{2 + \mathcal{T}_B\left(x_i\right) - \mathcal{I}_B\left(x_i\right) - \mathcal{F}_B\left(x_i\right)}{3} \right) - \min\left(\frac{2 + \mathcal{T}_A\left(x_i\right) - \mathcal{I}_A\left(x_i\right) - \mathcal{F}_A\left(x_i\right)}{3}, \frac{2 + \mathcal{T}_B\left(x_i\right) - \mathcal{I}_B\left(x_i\right) - \mathcal{F}_B\left(x_i\right)}{3} \right) \tag{22}$$

$$A^{\alpha} = \langle x, (\mathcal{T}_A(x))^{\alpha}, 1 - (1 - \mathcal{I}_A(x))^{\alpha}, 1 - (1 - \mathcal{F}_A(x))^{\alpha} \rangle \ for\ \alpha > 0 \tag{19}$$

Example 2 Let $A = \langle 0.8, 0.1, 0.2 \rangle$ and $B = \langle 0.6, 0.3, 0.4 \rangle$ are two neutrosophic set, then the various operations discussed above can be performed as-

$$A \oplus B = \langle 0.8 + 0.6 - 0.8 * 0.6, 0.1 * 0.3, 0.2 * 0.4 \rangle = \langle 0.92, 0.03, 0.08 \rangle$$

$$A \otimes B = \langle 0.8 * 0.6, 0.1 + 0.3 - 0.1 * 0.3, 0.2 + 0.4 - 0.2 * 0.4 \rangle = \langle 0.48, 0.37, 0.52 \rangle$$

If $\lambda = 0.5$ then

$$\lambda A = \langle 1 - (1 - 0.8)^{0.5}, (0.1)^{0.5}, (0.2)^{0.5} \rangle = \langle 0.86, 0.32, 0.45 \rangle$$

$$A^{\lambda} = \langle (0.8)^{0.5}, 1 - (1 - 0.1)^{0.5}, 1 - (1 - 0.2)^{0.5} \rangle = \langle 0.89, 0.05, 0.11 \rangle$$

Definition 11 (Huang 2016) Let $A\langle\langle x_1, \mathcal{T}_A\left(x_1\right), \mathcal{I}_A\left(x_1\right), \mathcal{F}_A\left(x_1\right) \rangle, \langle x_2, \mathcal{T}_A\left(x_2\right), \mathcal{I}_A\left(x_2\right), \mathcal{F}_A\left(x_2\right) \rangle, \dots \langle x_n, \mathcal{T}_A\left(x_n\right), \mathcal{I}_A\left(x_n\right), \mathcal{F}_A(x_n) \rangle\rangle$ and $B = \langle x_1, \langle \mathcal{T}_B\left(x_1\right), \mathcal{I}_B\left(x_1\right), \mathcal{F}_B(x_1) \rangle, \langle x_2, \mathcal{T}_B\left(x_2\right), \mathcal{I}_B\left(x_2\right), \mathcal{F}_B(x_2) \rangle, \dots \langle x_n, \mathcal{T}_B\left(x_n\right), I_B\left(x_n\right), \mathcal{F}_B(x_n) \rangle\rangle$ be two SVNS vector of length of n for $X = \{x_1, x_2 \dots, x_n\}$, then the Euclidean distance between A and B is defined as-

$$D(A, B) = \left[\sum_{i=1}^{n} \left(\sum_{j=1}^{4} \beta_j \phi_j\left(A_i, B_i\right) \right)^2 \right]^{1/2} \tag{20}$$

where $\beta_i \in [0, 1], \sum_{j=1}^{4} \beta_j = 1 \ and$

$$\phi_1\left(A_i, B_i\right) = \frac{\left|\mathcal{T}_A\left(x_i\right) - \mathcal{T}_B\left(x_i\right)\right|}{3} + \frac{\left|\mathcal{I}_A\left(x_i\right) - \mathcal{I}_B\left(x_i\right)\right|}{3} + \frac{\left|\mathcal{F}_A\left(x_i\right) - \mathcal{F}_B\left(x_i\right)\right|}{3} \tag{21}$$

$$\phi_3\left(A_i, B_i\right) = \frac{\left|\mathcal{T}_A\left(x_i\right) - \mathcal{T}_B\left(x_i\right) + \mathcal{I}_B\left(x_i\right) - \mathcal{I}_A\left(x_i\right)\right|}{2} \tag{23}$$

$$\phi_4\left(A_i, B_i\right) = \frac{\left|\mathcal{T}_A\left(x_i\right) - \mathcal{T}_B\left(x_i\right) + \mathcal{F}_B\left(x_i\right) - \mathcal{F}_A\left(x_i\right)\right|}{2} \tag{24}$$

Example 3 Let $A = \langle 0.8, 0.0, 0.4 \rangle$, $B = \langle 0.6, 0.0, 0.2 \rangle$ and $C = \langle 0.6, 0.0, 0.6 \rangle$ and are three single value neutrosophic set, Then the distance computed between neutrosophic set A, B and C considering $\lambda = 1 \ and \ \beta_1 = \beta_2 = \beta_3 = \beta_4 = 0.25$ is shown below

$$D(A, B) = 0.0583$$

$$D(A, C) = 0.1417$$

2.4 Interval number

Interval number is the simplest way to handle uncertainty in the decision process. It helps decision experts to rate a parameter from a range of values, where the parameter can have any value from the specified range. It is the extension of the concept of real numbers where each number is represented with a lower and upper bound. The various operations on the interval number that are being used in this study are discussed below.

Definition 12 (Tsaur 2011) A number $X = [x^L, x^U]$ is called interval number, if it can be defined on the real number line and $x^L < x^U$ *and* $x^L, x^U \epsilon R$ where x^L and x^U are the lower and upper bound respectively. The center and width of an interval number are defined as-

$$m(X) = \frac{x^L + x^U}{2} \tag{25}$$

$$w(X) = x^U - x^L \tag{26}$$

Definition 13 If $X = \{[x_1^L, x_1^U], [x_2^L, x_2^U] \ldots [x_n^L, x_n^U]\}$ and $Y = \{[y_1^L, y_1^U], [y_2^L, y_2^U] \ldots [y_n^L, y_n^U]\}$ are two interval number vectors of length *n*, then the Euclidean distance between X and *Y* is computed as-

$$d(X, Y) = \sqrt{\sum_{i=1}^{n} \frac{1}{2}\left[(x_i^L - y_i^L)^2 + (x_i^U - y_i^U)^2\right]} \tag{27}$$

2.5 TOPSIS

TOPSIS is the most widely used MCDM method to solve decision problems. It has been used in various fields like cloud service selection, web service selection, optimal route selection, supplier selection, etc. It ranks alternatives based on their geometric distance from an ideal and worst solution. It is widely used as it is better than VIKOR, AHP, and ANP in terms of efficiency and is simple to understand (Jahan et al. 2010; Mousavi-Nasab and Sotoudeh-Anvari 2017). The various steps of the original TOPSIS method used to rank alternatives are discussed below.

Step-1: Let there are m alternatives A_1 to A_m and n criteria C_1 to C_n involve in a decision problem to rank alternatives, then construct a decision matric of size $m \times n$ as follow.

$$DM = \begin{array}{c} \\ A_1 \\ A_2 \\ \vdots \\ A_m \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & \ldots & C_n \end{array} \\ \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \ldots & x_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m,1} & x_{m,2} & \cdots & x_{m,n} \end{bmatrix} \end{array} \tag{28}$$

where $x_{i,j}$ denotes the rating of alternative i for criterion j.

Step-2: Transform the decision matrix values on the same scale using Eq. (29) to find out the normalized decision matrix.

$$N = \begin{array}{c} \\ A_1 \\ A_2 \\ \vdots \\ A_m \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & \ldots & C_n \end{array} \\ \begin{bmatrix} n_{1,1} & n_{1,2} & \cdots & n_{1,n} \\ n_{2,1} & n_{2,2} & \ldots & n_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ n_{m,1} & n_{m,2} & \cdots & n_{m,n} \end{bmatrix} \end{array} \tag{29}$$

where

$$n_{i,j} = \frac{x_{i,j}}{\sqrt{\sum_{i=1}^{m} x_{i,j}^2}}$$

Step-3: Let $W = [w_1, w_2, \ldots, w_n]$ are the weight of criteria given by the expert as per their needs, the weighted normalized decision matrix is computed using Eq. (30) to incorporate decision-makers preference.

$$W = \begin{array}{c} \\ A_1 \\ A_2 \\ \vdots \\ A_m \end{array} \begin{array}{c} \begin{array}{cccc} C_1 & C_2 & \ldots & C_n \end{array} \\ \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,n} \\ w_{2,1} & w_{2,2} & \ldots & w_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{m,1} & w_{m,2} & \cdots & w_{m,n} \end{bmatrix} \end{array} \tag{30}$$

where

$$w_{i,j} = n_{i,j} * w_j$$

Step-4: A decision problem is always associated with cost and benefit criteria. The cost and benefit criteria are those criteria whose minimum and maximum value optimizes the cost of the optimal alternative. Let K_1 and K_2 are the set of cost and benefit criteria associated with the decision problem, then the positive ideal solution (PIS) and negative ideal solution (NIS) are computed using Eqs. (31) and (32) respectively. The PIS denotes the ideal alternative while NIS represents the worst alternative.

$$PIS = [Pos_1, Pos_2 \ldots Pos_n] \tag{31}$$

$$NIS = [Neg_1, Neg_2 \ldots Neg_n] \tag{32}$$

where

$$Pos_j = \begin{cases} \min_i(w_{i,j}) & if\ C_j \in K_1 \\ \max_i(w_{i,j}) & if C_j \in K_2 \end{cases}$$

$$Neg_j = \begin{cases} \max_i(w_{i,j}) & if\ C_j \in K_1 \\ \min_i(w_{i,j}) & ifC_j \in K_2 \end{cases}$$

Step-5: Compute the Euclidean distance of A_i from *PIS* and *NIS* using Eqs. (33) and (34) respectively.

$$D_i^+ = D(A_i, PIS) = \sqrt{\sum_{j=1}^{n} \left(w_{i,j} - Pos_j\right)^2} \tag{33}$$

$$D_i^- = D(A_i, NIS) = \sqrt{\sum_{j=1}^{n} \left(w_{i,j} - Neg_j\right)^2} \tag{34}$$

Step-6: Compute the closeness index CI_i of each alternative A_i using Eq. (35).

$$CI_i = \frac{D_i^-}{D_i^+ + D_i^-} \tag{35}$$

Step-7: The rank of alternative is computed using closeness index CI. The alternative having a maximum value of CI is the optimal alternative. The alternative with the minimum value of CI is the worst.

3 RMo-TOPSIS

The Mo-TOPSIS proposed by Lourenzutti and Krohling (2016) uses QoS weight to compute the distance of each alternative from PIS and NIS to accommodate customer priority, instead of computing weighted decision matrix using Eq. (30). Its correctness can be verified by expanding Eq. (33) used in traditional TOPSIS to compute the separation measure.

$$D_i^+ = \sqrt{\sum_{j=1}^{n} \left(w_{i,j} - Pos_j\right)^2} = \sqrt{\sum_{j=1}^{n} \left(n_{i,j} * w_j - n_{i,j}^+ * w_j\right)^2}$$
$$= \sqrt{\sum_{j=1}^{n} w_j^2 \left(n_{i,j} - n_{i,j}^+\right)^2} \tag{36}$$

That means, we can directly use QoS weight at the time of computation of separation measure. The proposed Mo-TOPSIS works properly for decision matrix with the heterogeneous type of QoS information, but it is not robust to rank reversal (Wang and Luo 2009; García-Cascales and Lamata 2012) in some scenarios as standard TOPSIS also suffers from it. Mo-TOPSIS suffers from rank reversal due to its normalization procedure and process of computing PIS and NIS. Due to issues in normalization methods, the normalized

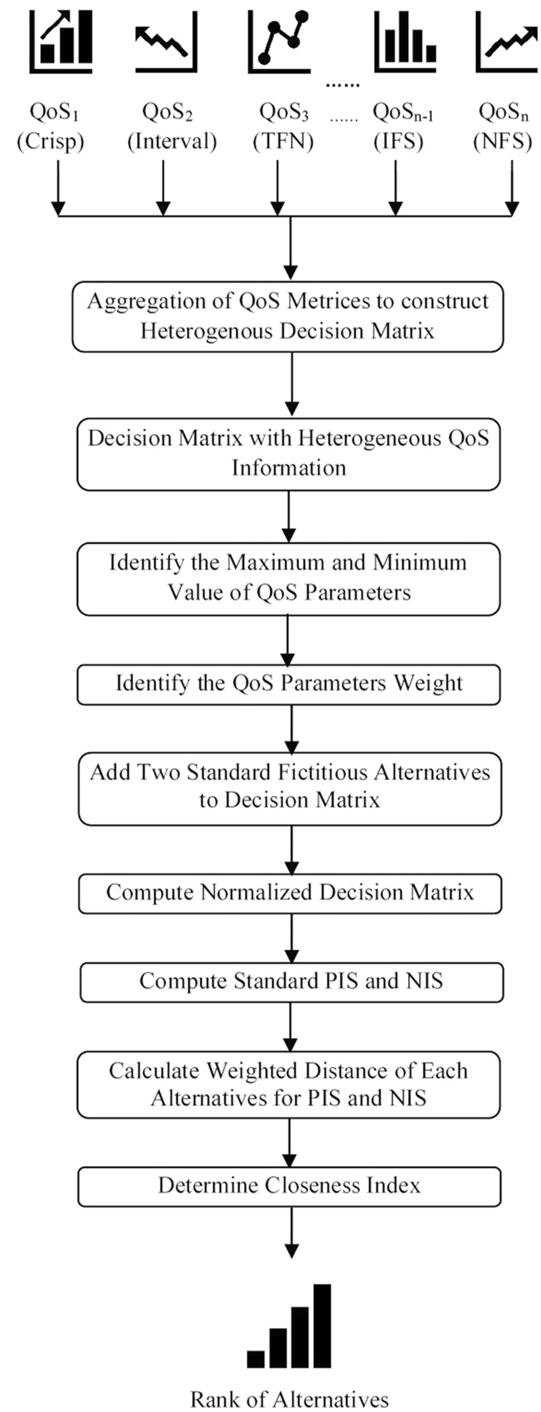

Fig. 2 Flowchart of RMo-TOPSIS

decision matrix changes on the addition or elimination of an alternative, which in turn changes the PIS and NIS. It affects the ranking of alternatives due to different values of separation measure of each alternative from ideal and worst alternatives.

The proposed extension of Mo-TOPSIS called RMo-TOPSIS uses two standard fictitious alternatives to fix the normalized value of each QoS of an alternative, which does not change with a change in the number of alternatives. The addition of two fictitious alternatives makes the normalization method consistent by fixing the denominator of it. It also uses standard PIS and NIS, which do not depend on the decision matrix. The RMo-TOPSIS has been designed to work for a decision matrix having crisp, interval, TFN, IFS and neutrosophic fuzzy type of QoS attributes. However, it can be extended for QoS information of other types using the same concept. The flowchart of RMo-TOPSIS is shown in Fig. 2. The various steps of RMo-TOPSIS are discussed below.

Step 1: Identify the QoS parameters

The proposed approach initially collects the set Q of necessary and desirable QoS parameter that is to be fulfilled by the set of alternatives A, where $Q = \{QoS_1, QoS_2, \dots, QoS_n\}$ and $A = \{A_1, A_2, \dots, A_m\}$.

Step 2: Construct the decision matrix

The decision-maker or expert constructs the decision matrix of size $m \times n$ for TOPSIS based on their experience. Each entry of the decision matrix represents the QoS parameters rating of the decision-maker. The decision matrix is represented in Eq. (37) where $r_{i,j}$ represents the rating of the QoS parameter QoS_j for alternative A_i. R_i can be a vector having crisp, interval number, TFN, IFS or NFS type information.

$$D = \begin{bmatrix} R_1 & R_2 & \cdots & R_m \end{bmatrix}^T \tag{37}$$

where

$$R_i = \begin{bmatrix} r_{i,1} & r_{i,2} & \dots & r_{i,n} \end{bmatrix}$$

Step 3: Identify the maximum and minimum value of QoS parameters

The maximum and minimum values of QoS parameters for each type are identified. Let the minimum and maximum value of each QoS parameter is defined using Eq. (38).

For example, if QoS_j is of crisp type and has $\left[r_{min_j}, r_{max_j}\right]$ as $[1, 5]$, then each alternative can have QoS_j value between 1 and 5. Any other value is not allowed. Similarly, if QoS_j is of interval type and has $\left[\left[r^L_{min_j}, r^U_{min_j}\right], \left[r^L_{max_j}, r^U_{max_j}\right]\right]$ as $[[0, 0], [5, 5]]$, then each alternative can have QoS_j value as $[x, y]$ where $0 \le x, y \le 5$. The minimum and maximum values of IFS type QoS parameter are [0, 1] and [1, 0] as IFS is minimum when the degree of membership value μ is 0 and non-membership value υ is 1. Similarly, it is maximum when the membership value μ is 1 and non-membership value υ is 0. The NFS is minimum when the degree of truth, indeterminacy and falsity are 0, 1 and 1. Similarly, it is maximum when the degree of truth, indeterminacy and falsity are 1, 0 and 0. So, any alternative can have a QoS value between the above range for IFS and NFS type.

Step 4: Identify the QoS parameters weight

The decision-maker provides the weight of all QoS parameters available in set Q as $W = \{w_1, w_2, \dots, w_n\}$. The weight is assigned by the decision-maker based on the experience and preference of each QoS. The weight assigned to each QoS is a real number such that $\sum w_i = 1$.

Step 5: Add two fictitious alternatives

Two fictitious alternatives F_1 and F_2 are added to the decision matrix, which is computed using Eqs. (39) and (40). These fictitious alternatives help to fix the normalized value of the QoS parameter in case of addition or removal of an alternative, which changes in Mo-TOPSIS and causes a rank reversal in it.

$$F_{1,j} = \begin{cases} r_{min_j} & \text{if } QoS_j \text{ is crisp type} \\ \left[r^L_{min_j}, r^U_{min_j}\right] & \text{if } QoS_j \text{ is interval type} \\ \left[r^\alpha_{min_j}, r^\beta_{min_j}, r^\gamma_{min_j}\right] & \text{if } QoS_j \text{ is TFN type} \\ [0, 1] & \text{if } QoS_j \text{ is IFS type} \\ [0, 1, 1] & \text{if } QoS_j \text{ is NFS type} \end{cases} \quad \text{for } j = 1 \text{ to } n \tag{39}$$

$$MinMax(QoS_j) = \begin{cases} \left[r_{min_j}, r_{max_j}\right] & \text{if } QoS_j \text{ is crisp type} \\ \left[\left[r^L_{min_j}, r^U_{min_j}\right], \left[r^L_{max_j}, r^U_{max_j}\right]\right] & \text{if } QoS_j \text{ is interval type} \\ \left[\left[r^\alpha_{min_j}, r^\beta_{min_j}, r^\gamma_{min_j}\right], \left[r^\alpha_{max_j}, r^\beta_{max_j}, r^\gamma_{max_j}\right]\right] & \text{if } QoS_j \text{ is TFN type} \\ [[0, 1], [1, 0]] & \text{if } QoS_j \text{ is IFS type} \\ [[0, 1, 1], [1, 0, 0]] & \text{if } QoS_j \text{ is NFS type} \end{cases} \tag{38}$$

$$F_{2,j} = \begin{cases} r_{max_j} & \text{if } QoS_j \text{ is crisp type} \\ \left[r^L_{max_j}, r^U_{max_j}\right] & \text{if } QoS_j \text{ is interval type} \\ \left[r^\alpha_{max_j}, r^\beta_{max_j}, r^\gamma_{max_j}\right] & \text{if } QoS_j \text{ is TFN type} \\ [1,0] & \text{if } QoS_j \text{ is IFS type} \\ [1,0,0] & \text{if } QoS_j \text{ is NFS type} \end{cases} \quad for\ j = 1\ to\ n \tag{40}$$

The modified decision matrix obtained after adding F_1 and F_2 are shown in Eq. (41) as-

$$D = \left[R_1\ R_2\ \cdots\ R_m\ , F_1, F_2\right]^T \tag{41}$$

Step 6: Compute the normalized decision matrix

The normalization of the decision matrix scales all the values of it on the same scale. The decision matrix consists of different types of elements like crisp, internal number, fuzzy number, etc. are normalized to make them on the unit scale. The normalized decision matrix is computed using Eq. (42).

$$N = \left[N_1\ N_2\ \cdots\ N_m\ , N_{m+1}, N_{m+2}\right]^T \tag{42}$$

where

$$N_i = \left[n_{i,1}\ n_{i,2}\ \ldots\ n_{i,n}\ \right]$$

$$n_{i,j} = \begin{cases} \frac{r_{i,j}}{r_{max_j}} & \text{if } QoS_j \text{ is crisp type} \\ \left[\frac{r^L_{i,j}}{r^U_{max_j}}, \frac{r^U_{i,j}}{r^U_{max_j}}\right] & \text{if } QoS_j \text{ is interval type} \\ \left[\frac{r^\alpha_{i,j}}{r^\gamma_{max_j}}, \frac{r^\beta_{i,j}}{r^\gamma_{max_j}}, \frac{r^\gamma_{i,j}}{r^\gamma_{max_j}}\right] & \text{if } QoS_j \text{ is TFN type} \\ r_{i,j} & \text{otherwise} \end{cases} \tag{43}$$

There is no need to normalize the IFS and NFS type QoS parameters as they are already in the unit scale. The above normalization method will not cause a change in normalized QoS value on addition or removal of alternatives as the denominator is fixed and its value is the maximum value of QoS i.e. any alternative cannot provide QoS value better than it. For example, if the QoS value varies between 1 to 100 then all alternatives will provide QoS in between 1–100. So, it does not depend on the addition and removal of alternatives.

Step 7: Compute the PIS and NIS

Let J_1 and J_2 are the set of benefit and cost QoS parameters such that $J_1 \subseteq Q, J_2 \subseteq Q\ and\ J_1 \cup J_2 = Q$. The *PIS* and *NIS* are computed using Eqs. (44) and (45) respectively and their computation depends on the type of data in the normalized decision matrix.

$$PIS = \left\{ \left(s^+_j, j \epsilon J_1\right), \left(s^-_j, j \epsilon J_2\right) \right\} for\ j = 1\ to\ n \tag{44}$$

$$NIS = \left\{ \left(s^-_j, j \epsilon J_1\right), \left(s^+_j, j \epsilon J_2\right) \right\} for\ j = 1\ to\ n \tag{45}$$

$$s^+_j = \begin{cases} n^+_j & \text{if } QoS_j \text{ is crisp type} \\ \left[n^+_{Lj}, n^+_{Uj}\right] & \text{if } QoS_j \text{ is interval type} \\ \left[n^+_{\alpha j}, n^+_{\beta j}, n^+_{\gamma j}\right] & \text{if } QoS_j \text{ is TFN type} \\ \left[n^+_{\mu j}, n^+_{\upsilon j}\right] & \text{if } QoS_j \text{ is IFS type} \\ \left[n^+_{\mathcal{T}j}, n^+_{\mathcal{I}j}, n^+_{\mathcal{F}j}\right] & \text{if } QoS_j \text{ is NFS type} \end{cases}$$

$$s^-_j = \begin{cases} n^-_j & \text{if } QoS_j \text{ is crisp type} \\ \left[n^-_{Lj}, n^-_{Uj}\right] & \text{if } QoS_j \text{ is interval type} \\ \left[n^-_{\alpha j}, n^-_{\beta j}, n^-_{\gamma j}\right] & \text{if } QoS_j \text{ is TFN type} \\ \left[n^-_{\mu j}, n^-_{\upsilon j}\right] & \text{if } QoS_j \text{ is IFS type} \\ \left[n^-_{\mathcal{T}j}, n^-_{\mathcal{I}j}, n^-_{\mathcal{F}j}\right] & \text{if } QoS_j \text{ is NFS type} \end{cases}$$

where $n^+_j = \max\left\{n_{i,j} | i = 1, \ldots m+2\right\}$, $n^+_{Lj} = \max\left\{n^L_{i,j} | i = 1, \ldots m+2\right\}$, $n^+_{Uj} = \max\left\{n^U_{i,j} | i = 1, \ldots m+2\right\}$, $n^+_{\alpha j} = \max\left\{n^\alpha_{i,j} | i = 1, \ldots m+2\right\}$, $n^+_{\beta j} = \max\left\{n^\beta_{i,j} | i = 1, \ldots m+2\right\}$, $n^+_{\gamma j} = \max\left\{n^\gamma_{i,j} | i = 1, \ldots m+2\right\}$, $n^+_{\mu j} = \max\left\{n^\mu_{i,j} | i = 1, \ldots m+2\right\}$, $n^+_{\upsilon j} = \max\left\{n^\upsilon_{i,j} | i = 1, \ldots m+2\right\}$, $n^+_{\mathcal{T}j} = \max\left\{n^{\mathcal{T}}_{i,j} | i = 1, \ldots m+2\right\}$, $n^+_{\mathcal{I}j} = \max\left\{n^{\mathcal{I}}_{i,j} | i = 1, \ldots m+2\right\}$, $n^+_{\mathcal{F}j} = \max\left\{n^{\mathcal{F}}_{i,j} | i = 1, \ldots m+2\right\}$. Similarly, $n^-_j = \min\left\{n_{i,j} | i = 1, \ldots m+2\right\}$, $n^-_{Lj} = \min\left\{n^L_{i,j} | i = 1, \ldots m+2\right\}$, $n^-_{Uj} = \min\left\{n^U_{i,j} | i = 1, \ldots m+2\right\}$, $n^-_{\alpha j} = \min\left\{n^\alpha_{i,j} | i = 1, \ldots m+2\right\}$, $n^-_{\beta j} = \min\left\{n^\beta_{i,j} | i = 1, \ldots m+2\right\}$, $n^-_{\gamma j} = \min\left\{n^\gamma_{i,j} | i = 1, \ldots m+2\right\}$, $n^-_{\mu j} = \min\left\{n^\mu_{i,j} | i = 1, \ldots m+2\right\}$, $n^-_{\upsilon j} = \min\left\{n^\upsilon_{i,j} | i = 1, \ldots m+2\right\}$, $n^-_{\mathcal{T}j} = \min\left\{n^{\mathcal{T}}_{i,j} | i = 1, \ldots m+2\right\}$, $n^-_{\mathcal{I}j} = \min\left\{n^{\mathcal{I}}_{i,j} | i = 1, \ldots m+2\right\}$, $n^-_{\mathcal{F}j} = \min\left\{n^{\mathcal{F}}_{i,j} | i = 1, \ldots m+2\right\}$

Step 8: Compute weighted euclidean distance from PIS and NIS

The weighted distance of each alternative A_i from *PIS* and *NIS* excluding the fictitious alternatives F_1 and F_2 is computed using Eqs. (46) and (47) respectively.

$$WD^+_i = \sqrt{\sum_{j=1}^{n} \left[w_j * d(s^+_j, n_{i,j})\right]^2} \ for\ i = 1\ to\ m \tag{46}$$

$$WD_i^- = \sqrt{\sum_{j=1}^{n} [w_j * d(s_j^-, n_{i,j})]^2} \; for \; i = 1 \; to \; m \qquad (47)$$

$$CI_i = \frac{WD_i^-}{WD_i^+ + WD_i^-} \qquad (48)$$

where

$$d\left(s_j^+, n_{i,j}\right) = \begin{cases} \sqrt{(s_j^+ - n_{i,j})^2} & if\ QoS_j\ is\ crisp\ type \\ \sqrt{\frac{1}{2}\left[(s_{Lj}^+ - n_{i,j}^L)^2 + (s_{Uj}^+ - n_{i,j}^U)^2\right]} & if\ QoS_j\ is\ interval\ type \\ \sqrt{\frac{1}{3}\left[(s_{\alpha j}^+ - n_{i,j}^\alpha)^2 + (s_{\beta j}^+ - n_{i,j}^\beta)^2 + (s_{\gamma j}^+ - n_{i,j}^\gamma)^2\right]} & if\ QoS_j\ is\ TFN\ type \\ \sqrt{\frac{1}{2}\left[\left(s_{\mu j}^+ - n_{i,j}^\mu\right)^2 + \left(s_{\upsilon j}^+ - n_{i,j}^\upsilon\right)^2\right]} & if\ QoS_j\ is\ IFS\ type \\ \sqrt{\left(\sum_{k=1}^4 \beta_k \phi_k\left(s_j^+, n_{i,j}\right)\right)^2} & if\ QoS_j\ is\ NFS\ type \end{cases}$$

$$d\left(s_j^-, n_{i,j}\right) = \begin{cases} \sqrt{(s_j^- - n_{i,j})^2} & if\ QoS_j\ is\ crisp\ type \\ \sqrt{\frac{1}{2}\left[(s_{Lj}^- - n_{i,j}^L)^2 + (s_{Uj}^- - n_{i,j}^U)^2\right]} & if\ QoS_j\ is\ interval\ type \\ \sqrt{\frac{1}{3}\left[(s_{\alpha j}^- - n_{i,j}^\alpha)^2 + (s_{\beta j}^- - n_{i,j}^\beta)^2 + (s_{\gamma j}^- - n_{i,j}^\gamma)^2\right]} & if\ QoS_j\ is\ TFN\ type \\ \sqrt{\frac{1}{2}\left[\left(s_{\mu j}^- - n_{i,j}^\mu\right)^2 + \left(s_{\upsilon j}^- - n_{i,j}^\upsilon\right)^2\right]} & if\ QoS_j\ is\ IFS\ type \\ \sqrt{\left(\sum_{k=1}^4 \beta_k \phi_k\left(s_j^-, n_{i,j}\right)\right)^2} & if\ QoS_j\ is\ NFS\ type \end{cases}$$

$\phi_1, \phi_2 \phi_3$ *and* ϕ_4 are defined in Eqs. (21)–(24) and $\beta_i \in [0, 1], \sum_{j=1}^4 \beta_j = 1$

Step 9: Compute the closeness index and rank the cloud services

The closeness index of each alternative A_i is computed using Eq. (48). The alternative with the highest value of the closeness index is considered as best while the alternative with the lowest value is considered as the worst alternative.

4 Performance evaluation

In this section, a case study is performed to show the application of the proposed RMo-TOPSIS in the field of cloud service selection. The service selection in the cloud environment is one of the critical problems due to the availability of many cloud services providing the same level of QoS. The cloud users have to select the right service as cloud services are hired for a long time and users do not churn from one service provider to another. So, the

Table 3 Details of QoS parameters

QoS parameter	Data type	QoS type	QoS weight	Minimum QoS value	Maximum QoS value
Cost	Crisp	Cost	0.23	100	200
Response time	Interval	Cost	0.23	[150, 160]	[350, 360]
Availability	Crisp	Benefit	0.20	0.5	1
Reliability	TFN	Benefit	0.14	(0.1, 0.2, 0.3)	(0.9, 0.95, 1.0)
Reputation	IFS	Benefit	0.06	‹0, 1›	‹1, 0›
Security	NFS	Benefit	0.14	‹0, 1, 1›	‹1, 0, 0›

Table 4 Decision matrix with heterogeneous QoS information for cloud service selection

Cloud services	Cost	Response time	Availability	Reliability	Reputation	Security
CS_1	126	[186, 285]	0.95	(0.5, 0.6, 0.7)	‹0.65, 0.25›	‹0.65, 0.35, 0.30›
CS_2	133	[180, 282]	0.88	(0.6, 0.8, 0.9)	‹0.35, 0.60›	‹0.80, 0.20, 0.15›
CS_3	110	[201, 291]	0.91	(0.6, 0.7, 0.9)	‹0.85, 0.10›	‹0.35, 0.65, 0.60›
CS_4	148	[197, 300]	0.87	(0.4, 0.6, 0.7)	‹0.80, 0.10›	‹0.90, 0.10, 0.05›
CS_5	107	[182, 280]	0.85	(0.3, 0.5, 0.7)	‹0.30, 0.65›	‹0.45, 0.55, 0.60›
CS_6	122	[195, 310]	0.96	(0.3, 0.4, 0.6)	‹0.55, 0.35›	‹0.50, 0.50, 0.50›
CS_7	140	[210, 330]	0.98	(0.4, 0.6, 0.9)	‹0.50, 0.35›	‹0.85, 0.15, 0.20›
CS_8	138	[197, 312]	0.93	(0.5, 0.7, 1.0)	‹0.45, 0.50›	‹0.75, 0.20, 0.20›
F_1	100	[150, 160]	0.50	(0.1, 0.2, 0.3)	‹0, 1›	‹0, 1, 1›
F_2	200	[350, 360]	1.00	(0.9, 0.95, 1.0)	‹1, 0›	‹1, 0, 0›

Table 5 Normalized decision matrix for cloud service selection using RMo-TOPSIS

Cloud services	Cost	Response time	Availability	Reliability	Reputation	Security
CS_1	0.630	[0.52, 0.79]	0.95	(0.5, 0.6, 0.7)	‹0.65, 0.25›	‹0.65, 0.35, 0.30›
CS_2	0.665	[0.5, 0.78]	0.88	(0.6, 0.8, 0.9)	‹0.35, 0.60›	‹0.80, 0.20, 0.15›
CS_3	0.550	[0.56, 0.81]	0.91	(0.6, 0.7, 0.9)	‹0.85, 0.10›	‹0.35, 0.65, 0.60›
CS_4	0.740	[0.55, 0.83]	0.87	(0.4, 0.6, 0.7)	‹0.80, 0.10›	‹0.90, 0.10, 0.05›
CS_5	0.535	[0.51, 0.78]	0.85	(0.3, 0.5, 0.7)	‹0.30, 0.65›	‹0.45, 0.55, 0.60›
CS_6	0.610	[0.54, 0.86]	0.96	(0.3, 0.4, 0.6)	‹0.55, 0.35›	‹0.50, 0.50, 0.50›
CS_7	0.700	[0.58, 0.92]	0.98	(0.4, 0.6, 0.9)	‹0.50, 0.35›	‹0.85, 0.15, 0.20›
CS_8	0.690	[0.55, 0.87]	0.93	(0.5, 0.7, 1.0)	‹0.45, 0.50›	‹0.75, 0.20, 0.20›
F_1	0.500	[0.42, 0.44]	0.50	(0.1, 0.2, 0.3)	‹0.0, 1.0›	‹0.0, 1.0, 1.0›
F_2	1.000	[0.97, 1.00]	1.00	(0.9, 0.95, 1.0)	‹1.0, 0.0›	‹1.0, 0.0, 0.0›

Table 6 PIS and NIS computed from the normalized decision matrix

	Cost	Response Time	Availability	Reliability	Reputation	Security
PIS	0.500	[0.42, 0.44]	0.50	(0.1, 0.2, 0.3)	‹0.0, 1.0›	‹0.0, 1.0, 1.0›
NIS	1.000	[0.97, 1.00]	1.00	(0.9, 0.95, 1.0)	‹1.0, 0.0›	‹1.0, 0.0, 0.0›

proper selection of cloud services avoids capital loss and failures.

The proposed method is also validated by comparing it with MCDM based methods used for ranking alternatives with QoS of heterogeneous type. The robustness against rank reversal is demonstrated through sensitivity analysis comprising a series of experiments.

4.1 Cloud service selection using RMo-TOPSIS

In this case study, an e-commerce company wants to deploy its application on a server and there are eight infrastructure as a service (IaaS) providers CS_1 *to* CS_8, which offers the hosting facility. The e-commerce company wants to know the best IaaS provider based on QoS parameters cost, availability, response time, reliability, reputation and security. Table 3 shows the QoS parameters, their data type, its type i.e. cost or benefit and weight provided by the e-commerce company for each QoS to find the best service. Cost is crisp as it is provided by the cloud service provider. Availability is crisp as it is claimed in the service level agreement signed by the cloud service provider. Response time is interval as it depends on the load of the cloud machine, network, etc. and

Table 7 Weighted distance from PIS, NIS, closeness index and rank obtained using RMo-TOPSIS

Cloud services	Weighted distance from PIS	Weighted distance from NIS	Closeness index	Rank
CS_1	0.1923	0.2823	0.5949	2
CS_2	0.1909	0.2843	0.5983	1
CS_3	0.1958	0.2794	0.5880	3
CS_4	0.2078	0.2718	0.5667	6
CS_5	0.2083	0.2715	0.5659	7
CS_6	0.2126	0.2694	0.5589	8
CS_7	0.2047	0.2779	0.5759	5
CS_8	0.2023	0.2792	0.5799	4

has different response times for the same request (Liu et al. 2020). Reliability is a qualitative parameter and depends on expert perception (Liu et al. 2020). So, it is considered as TFN and will provide flexibility to experts for rating cloud services with partial knowledge. Reputation is perceived by cloud users and is a subjective matter, so IFS is used to represent it. Security is also qualitative QoS and easy to express in a fuzzy environment. Therefore, NFS is used to represent it. The QoS parameters cost and response time are costly QoS while rest are benefit QoS i.e. larger the better. Table 3. also shows the minimum and maximum value of each QoS parameter that cloud services can provide.

The decision matrix obtained based on the rating provided by cloud experts is shown in Table 4. It consists of a level or values of QoS provided by each cloud service. The RMo-TOPSIS is applied to the decision matrix to find the best cloud service as per e-commerce company requirements. The next step of RMo-TOPSIS is to compute the two standard fictitious alternatives using Eqs. (39) and (40) i.e. cloud services F_1 and F_2 shown in Table 4. These two fictitious cloud services contribute to avoiding rank reversal in RMo-TOPSIS. The QoS values of the decision matrix are normalized on the same scale using Eq. (43) in the next step and shown in Table 5. Table 6 shows the PIS and NIS computed using Eqs. (44) and (45) on the normalized decision matrix. The weighted distance of each cloud service from PIS and NIS is computed using Eqs. (46) and (47). Table 7 shows the weighted distance of each cloud service along with the closeness index and rank. The distance between the NFS data type is computed using each beta value of 0.25 throughout the paper. The closeness index is computed using Eq. (48) and cloud services are ranked based on the closeness index. The cloud service having a maximum value of closeness index i.e. CS_2 is ranked as the best cloud service as per QoS requirement. It provides the best QoS for response time and reliability while third-best in security, fifth in cost, sixth in availability and seventh in reputation. Overall, CS_2 is the best among all cloud services and can be verified from the decision matrix.

4.2 Validation of result

The ranking result of the proposed method is compared with other methods to validate it. The rank of cloud services computed by RMo-TOPSIS on a decision matrix is compared with Mo-TOPSIS (Lourenzutti and Krohling 2016) and extended TODIM (Fan et al. 2013). It is also compared with generalizations of TOPSIS for a specific type of information to validate it further. Since Mo-TOPSIS does not support NFS, so we considered a decision matrix with QoS type crisp, interval, TFN, and IFS. The decision matrix along with QoS wight, which is used to compare rank obtained by Mo-TOPSIS and RMo-TOPSIS is shown in Table 8. It also shows the closeness index and ranks obtained using

Table 8 Rank obtained using Mo-TOPSIS and RMo-TOPSIS

Cloud services	Cost (0.23)	Response time (0.23)	Availability (0.24)	Reliability (0.24)	Reputation (0.06)	Mo-TOPSIS		RMo-TOPSIS	
						Closeness index	Rank	Closeness index	Rank
CS_1	126	[186, 285]	0.95	(0.5, 0.6, 0.7)	‹0.65, 0.25›	0.5356	4	0.5864	3
CS_2	133	[180, 282]	0.88	(0.6, 0.8, 0.9)	‹0.35, 0.60›	0.6293	2	0.5966	2
CS_3	110	[201, 291]	0.91	(0.6, 0.7, 0.9)	‹0.85, 0.10›	0.6539	1	0.6222	1
CS_4	148	[197, 300]	0.87	(0.4, 0.6, 0.7)	‹0.80, 0.10›	0.4237	7	0.5330	8
CS_5	107	[182, 280]	0.85	(0.3, 0.5, 0.7)	‹0.30, 0.65›	0.4589	6	0.5568	6
CS_6	122	[195, 310]	0.96	(0.3, 0.4, 0.6)	‹0.55, 0.35›	0.3728	8	0.5443	7
CS_7	140	[210, 330]	0.98	(0.4, 0.6, 0.9)	‹0.50, 0.35›	0.4645	5	0.5618	5
CS_8	138	[197, 312]	0.93	(0.5, 0.7, 1.0)	‹0.45, 0.50›	0.5575	3	0.5789	4

Table 9 The rank of cloud services using Mo-TOPSIS, RMo-TOPSIS and extended TODIM

Cloud services	QoS_1 (0.40)	QoS_2 (0.22)	QoS_3 (0.38)	Mo-TOPSIS		RMo-TOPSIS		Extended TODIM	
				Closeness index	Rank	Closeness index	Rank	Dominance degree	Rank
CS_1	645	[3, 4]	(3, 4, 5)	0.3234	5	0.4355	5	0.0000	5
CS_2	595	[2, 5]	(2, 4, 6)	0.3640	4	0.4684	4	0.3333	4
CS_3	700	[1, 3]	(4, 5, 6)	0.4882	3	0.4865	3	0.4404	3
CS_4	615	[2, 3]	(4, 6, 8)	0.6186	2	0.5571	2	0.9305	2
CS_5	670	[1, 3]	(6, 7, 8)	0.6685	1	0.5860	1	1.0000	1

Table 10 The ranking result of cloud services using G-TOPSIS and RMo-TOPSIS

Cloud service	QoS_1 (0.3483)	QoS_2 (0.1072)	QoS_3 (0.3483)	QoS_4 (0.1356)	QoS_5 (0.2464)	G-TOPSIS		RMo-TOPSIS	
						Closeness index	Rank	Closeness index	Rank
Amazon	0.29	92.89	126.66	110.33	104	0.7534	2	0.5724	2
Google	0.48	96	140.89	133	70.91	0.7438	4	0.5598	4
Century Link	0.9	100.15	81.21	78.66	70.09	0.6965	6	0.4545	6
HP	0.42	119.63	111.95	100.5	131.81	0.7458	3	0.5670	3
Microsoft Azure	0.45	29.07	152.96	42.47	90.83	0.7643	1	0.5894	1
Rackspace	0.42	74.53	6.19	130.84	90.37	0.5989	10	0.3976	9
City-Cloud	1.05	81.22	58.42	68.45	78.15	0.6745	7	0.4269	7
Linode	0.56	43.02	17.34	141.23	51.71	0.6029	9	0.3788	10
GoGrid	0.34	64.64	75.89	174.12	100.14	0.7231	5	0.5160	5
Softlayer	0.48	130.68	23.2	122.62	89.74	0.6216	8	0.4080	8
Minimum QoS	0	10	0	20	20				
Maximum QoS	2	150	200	200	150				

both methods. The maximum and minimum values of QoS attributes and their type are the same as shown in Table 3. The QoS weight used is shown in Table 8 along with the QoS name. It can be observed that Mo-TOPSIS and RMo-TOPSIS rank cloud services in a similar way as only the process of normalization is different and the rest is the same.

In the next experiment, the result validation of RMo-TOPSIS is shown by comparing the rank of cloud services obtained using Mo-TOPSIS (Lourenzutti and Krohling 2016) and extended TODIM (Fan et al. 2013). The decision matrix used for the experiment is shown in Table 9. Since, extended TODIM supports only decision matrix with QoS of type crisp, interval and TFN, so there are only three types of data. The QoS_3 is the benefit QoS while others are costly QoS. The QoS weight used to compute the rank is shown in Table 9 along with the QoS name. The minimum and maximum value of QoS_1, QoS_2 and QoS_3 used for RMo-TOPSIS are [500, 800], [[0, 1], [4, 5]] and [(1, 2, 3), [7, 8, 9]]. The value of theta used for calculating dominance degree in extended TODIM is 5. Theta signifies the loss aversion degree in the decision matrix. A better value of theta means a lesser loss aversion degree. The closeness index and ranked obtained for cloud services on the given decision matrix using Mo-TOPSIS and RMo-TOPSIS is stated in Table 9. Table 9 also shows the dominance degree and rank of cloud services using extended TODIM. It can be observed that all three methods rank the cloud services in the same manner, which validates the ranking result of the proposed method.

In another experiment, we validate the ranking result of the proposed method with rank reversal robust G-TOPSIS (Tiwari and Kumar 2021) for crisp type QoS. The decision matrix along with QoS weight is shown in Table 10. The QoS_1 and QoS_2 are costly parameters while others are

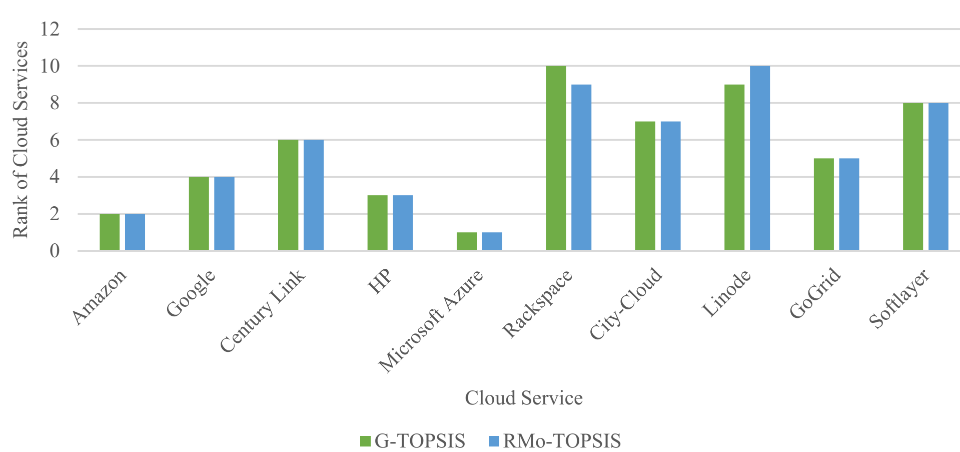

Fig. 3 Cloud services ranking using RMo-TOPSIS and G-TOPSIS

Table 11 Decision matrix and QoS weight for cloud services ranking using fuzzy TOPSIS and RMo-TOPSIS

Cloud services	QoS_1 (0.1195)	QoS_2 (0.1048)	QoS_3 (0.0748)	QoS_4 (0.0890)	QoS_5 (0.1345)	QoS_6 (0.1191)	QoS_7 (0.0898)	QoS_8 (0.1046)	QoS_9 (0.0744)	QoS_{10} (0.0894)
Amazon EC2	(0, 0.2, 0.4)	(0, 0, 0.2)	(0, 0.2, 0.4)	(0, 0.2, 0.4)	(0, 0.2, 0.4)	(0, 0.2, 0.4)	(0.4, 0.6, 0.8)	(0.4, 0.6, 0.8)	(0, 0.2, 0.4)	(0, 0.2, 0.4)
Digital ocean	(0, 0.2, 0.4)	(0.2, 0.4, 0.6)	(0, 0, 0.2)	(0, 0, 0.2)	(0, 0.2, 0.4)	(0, 0.2, 0.4)	(0.4, 0.6, 0.8)	(0.4, 0.6, 0.8)	(0, 0.2, 0.4)	(0, 0.2, 0.4)
Google	(0, 0.2, 0.4)	(0.4, 0.6, 0.8)	(0, 0.2, 0.4)	(0, 0.2, 0.4)	(0, 0.2, 0.4)	(0, 0.2, 0.4)	(0.4, 0.6, 0.8)	(0.4, 0.6, 0.8)	(0, 0, 0.2)	(0, 0, 0.2)
Microsoft azure	(0, 0.2, 0.4)	(0.2, 0.4, 0.6)	(0, 0.2, 0.4)	(0, 0.2, 0.4)	(0, 0, 0.2)	(0, 0, 0.2)	(0.2, 0.4, 0.6)	(0.2, 0.4, 0.6)	(0, 0.2, 0.4)	(0, 0.2, 0.4)
Rackspace	(0.2, 0.4, 0.6)	(0.2, 0.4, 0.6)	(0.2, 0.4, 0.6)	(0.2, 0.4, 0.6)	(0, 0.2, 0.4)	(0, 0.2, 0.4)	(0.4, 0.6, 0.8)	(0.4, 0.6, 0.8)	(0.4, 0.6, 0.8)	(0, 0.2, 0.4)
Softlayer	(0, 0.2, 0.4)	(0, 0, 0.2)	(0, 0.2, 0.4)	(0.2, 0.4, 0.6)	(0, 0.2, 0.4)	(0, 0.2, 0.4)	(0.4, 0.6, 0.8)	(0.4, 0.6, 0.8)	(0.2, 0.4, 0.6)	(0.2, 0.4, 0.6)

beneficial. The minimum and maximum values used for RMo-TOPSIS are also shown in Table 10. The rank and closeness index of cloud services computed using both methods are shown in Table 10. Figure 3 shows the visual comparison of rank obtained with both methods. It can be observed that both methods rank cloud services in a similar manner as the aggregation process is analogous with a slight difference in the computation method of some steps.

The ranking result of the RMo-TOPSIS is also validated with the generalization of TOPSIS for the fuzzy environment. It is compared with the framework proposed by Kumar et al. (2017) to rank cloud services under a fuzzy environment. We considered the same decision matrix and QoS weight used by Kumar et al. (2017) shown in Table 11. The QoS_1 to QoS_4 are cost parameters while others are beneficial. The minimum and maximum value of QoS considered for Mo-TOPSIS is (0, 0, 0) and (1, 1, 1) for all. The rank of cloud services obtained on the decision matrix in Table 11 using fuzzy TOPSIS and RMo-TOPSIS is shown in Fig. 4. It can be observed that both methods rank cloud services in a similar manner, which validates our proposed method.

4.3 Sensitivity analysis

The sensitivity analysis is performed to monitor the rank reversal phenomenon in RMo-TOPSIS. The first two cases are carried out using deletion of cloud services from existing decision matrix while other with addition. The QoS weight, their data type, minimum and maximum value along with type i.e. cost and benefit used in all cases are the same as Table 3. In the first case, one cloud service is removed at a time from the decision matrix shown in Table 4 and RMo-TOPSIS is used to compute the rank of cloud services. Initially, we removed CS_1 from the decision matrix and computed the rank of the remaining cloud services. In the successive experiment, we removed CS_2, $CS_{3...}$ CS_8 from the decision matrix one at a time. Table 12 shows the closeness index computed in each experiment. It can be inferred that the closeness index does not depend on the removal of a cloud service and remains the same throughout all experiments. The rank of cloud services obtained in each experiment is shown in Fig. 5. It can be said that the order of rank of cloud services remains consistent on the removal of cloud services from the decision matrix. For example, it can be observed in experiment two that the rank of CS_1 changes to first and similarly the rank of other cloud services is also reduced by one as best cloud service CS_2 is removed, but overall their relative ranking order remains the same.

We performed a second case of sensitivity analysis by removing more than one cloud service from the existing decision matrix to monitor rank reversal. The parameters and decision matrix used are the same as Tables 3 and 4, respectively. Table 13 shows the closeness index of cloud

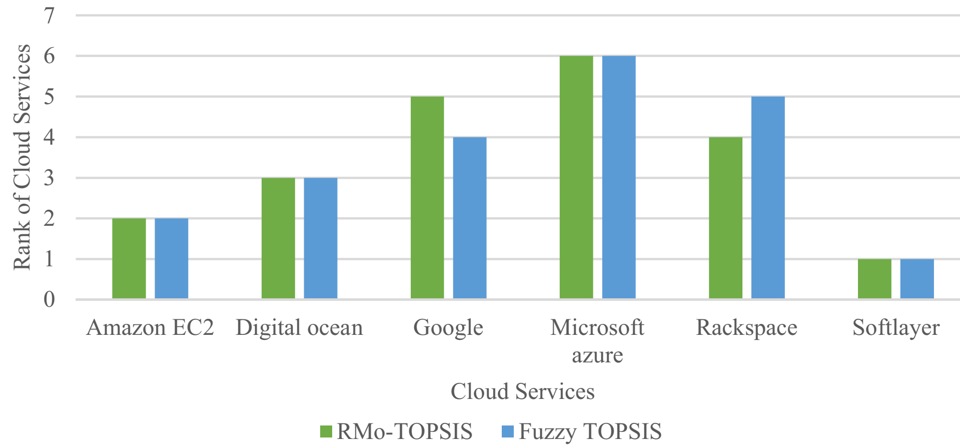

Fig. 4 The rank of cloud services using fuzzy RMo-TOPSIS and TOPSIS

Table 12 Closeness index of cloud services on removal

Cloud services	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5	Exp. 6	Exp. 7	Exp. 8
CS_1	–	0.5949	0.5949	0.5949	0.5949	0.5949	0.5949	0.5949
CS_2	0.5983	–	0.5983	0.5983	0.5983	0.5983	0.5983	0.5983
CS_3	0.5880	0.5880	–	0.5880	0.5880	0.5880	0.5880	0.5880
CS_4	0.5667	0.5667	0.5667	–	0.5667	0.5667	0.5667	0.5667
CS_5	0.5659	0.5659	0.5659	0.5659	–	0.5659	0.5659	0.5659
CS_6	0.5589	0.5589	0.5589	0.5589	0.5589	–	0.5589	0.5589
CS_7	0.5759	0.5759	0.5759	0.5759	0.5759	0.5759	–	0.5759
CS_8	0.5799	0.5799	0.5799	0.5799	0.5799	0.5799	0.5799	–

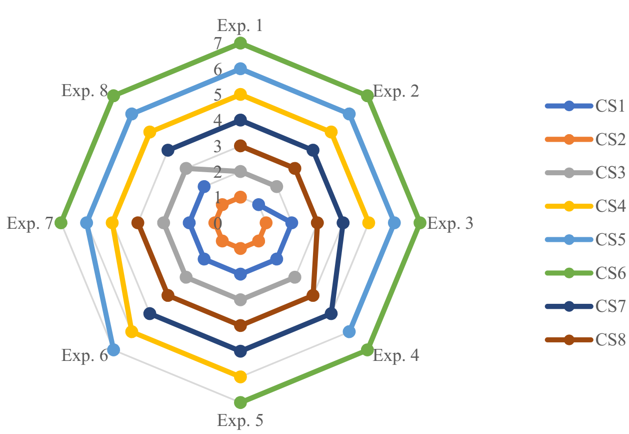

Fig. 5 The rank of cloud services on removal

Table 13 Closeness index of cloud services on the removal of multiple services

Cloud services	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5
CS_1	0.5949	–	–	–	0.5949
CS_2	0.5983	0.5983	0.5983		–
CS_3	0.5880	–	–	0.5880	–
CS_4	0.5667	–	0.5667	–	–
CS_5	–	–	0.5659	0.5659	–
CS_6	0.5589	0.5589	0.5589	0.5589	–
CS_7	–	0.5759	–	–	0.5759
CS_8	0.5799	0.5799	0.5799	–	–

service in the different experiments and we can observe that the closeness index remains constant and does not depend on the removal of more than one cloud service. So, we can say that RMo-TOPSIS is robust to the rank reversal in case of removal of more than one cloud service at a time.

We performed the third case of sensitivity analysis by adding one cloud service in an existing decision matrix to monitor robustness against rank reversal on the addition of service. The parameters and decision matrix used are the same as Tables 3 and 4 respectively. Initially, only three cloud services CS_1, $CS_{2,}$ and CS_3 are considered in the decision matrix, and rank is computed using RMo-TOPSIS. In the successive experiment CS_4, $CS_{5...}$ CS_8 is added one at a time in the decision matrix to compute the rank. Table 14 shows the closeness index of cloud service in various experiments. The closeness index of cloud services remains the same and does not change with the addition of service throughout all experiments. The rank of cloud services is visually shown in Fig. 6. It can be guaranteed from Fig. 6 that the rank of cloud service only changes on the addition of an optimal service but their rank order relative to ranking on

Table 14 Closeness index of cloud services on addition

Cloud services	Exp. 1	Exp. 2	Exp. 3	Exp. 4	Exp. 5	Exp. 6
CS_1	0.5949	0.5949	0.5949	0.5949	0.5949	0.5949
CS_2	0.5983	0.5983	0.5983	0.5983	0.5983	0.5983
CS_3	0.5880	0.5880	0.5880	0.5880	0.5880	0.5880
CS_4	–	0.5667	0.5667	0.5667	0.5667	0.5667
CS_5	–	–	0.5659	0.5659	0.5659	0.5659
CS_6	–	–	–	0.5589	0.5589	0.5589
CS_7	–	–	–	–	0.5759	0.5759
CS_8	–	–	–	–	–	0.5799

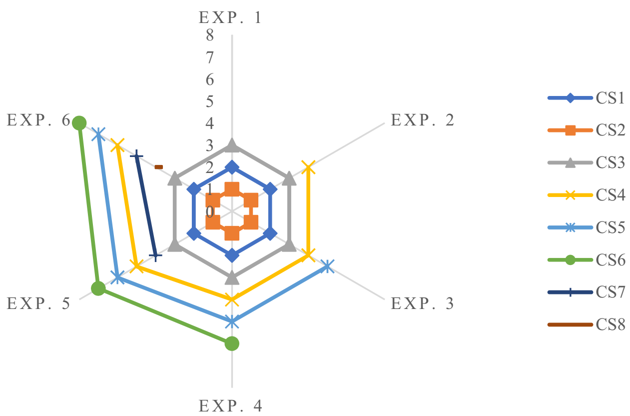

Fig. 6 The rank of cloud services on removal

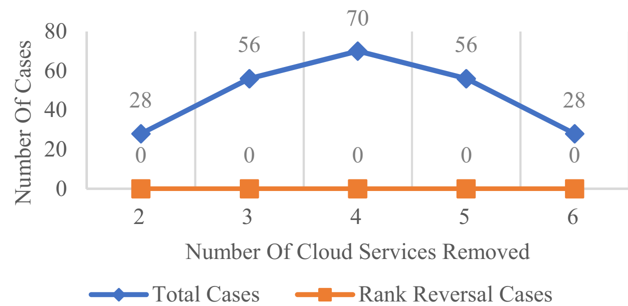

Fig. 7 Number of rank reversal cases with random removal of cloud services

Table 15 Closeness index of cloud services on the addition of multiple services

Cloud services	Exp. 1	Exp. 2	Exp. 3	Exp. 4
CS_1	0.5949	0.5949	0.5949	0.5949
CS_2	0.5983	0.5983	0.5983	0.5983
CS_3	0.5880	0.5880	0.5880	0.5880
CS_4	–	–	0.5667	0.5667
CS_5	0.5659	–	0.5659	0.5659
CS_6	–	0.5589	–	0.5589
CS_7	0.5759	0.5759	0.5759	0.5759
CS_8	–	0.5799	0.5799	0.5799

the original decision matrix remains the same, which proves RMo-TOPSIS is robust to rank reversal.

We performed the fourth case of sensitivity analysis by adding more than one cloud service in the existing decision matrix to monitor the rank reversal. We initially considered the first three cloud services in the decision matrix and randomly added more than one service in different experiments. Table 15 shows the closeness index of different cloud services in different experiments. It can be observed from Table 15 that the closeness index of each cloud service remains constant in all experiments and does not depend on the random addition of more than one cloud service, which indicates that RMo-TOPSIS is robust to rank reversal with the addition of more than one cloud services.

In another case, we randomly removed two to six cloud services from the decision matrix shown in Table 4 and monitored the rank reversal in RMo-TOPSIS. The parameters considered are the same as in Table 3 and obtained results are shown in Fig. 7. There are 28 ($^8C_2$) possible decision matrices on randomly removing two cloud services from the decision matrix shown in Table 4. Similarly, there will be ^8C_n possible decision matrices on removing n cloud services. In Fig. 7, the total possible number of decision matrices generated and cases with the rank reversal in RMo-TOPSIS on removing n cloud services are shown. It can be observed that there are zero cases with the rank reversal in each experiment, which proves our claim of robustness from the rank reversal in RMo-TOPSIS.

In the next experiment, we considered the decision matrix shown in Table 8 and monitored rank reversal in Mo-TOPSIS and RMo-TOPSIS. We considered the same parameters for the experiment as mentioned in the case study section for the decision matrix of Table 8. We randomly removed the one to six cloud services from the decision matrix and found out the number of removals causing a rank reversal in each method. Table 16 shows the result of the experiment. It can be observed that RMo-TOPSIS does not have a rank reversal in any case, while Mo-TOPSIS has the rank reversal in each case except for the removal of a single cloud service case. So, RMo-TOPSIS is robust to rank reversal.

Table 16 Rank reversal cases in different methods on random removal of services from decision matric in Table 4

Cloud services removed method	1		2		3		4		5		6	
	Total cases	Rank reversal	Total cases	Rank reversal	Total cases	Rank reversal	Total cases	Rank reversal	Total cases	Rank reversal	Total cases	Rank rever-sal
Mo-TOPSIS	8	0	28	2	56	7	70	6	56	5	28	2
RMo-TOPSIS	8	0	28	0	56	0	70	0	56	0	28	0

4.4 Comparison and discussion

In this section, we have compared the proposed RMo-TOPSIS with state of the art MCDM methods available in the literature. We have compared the RMo-TOPSIS with Mo-TOPSIS, G-TOPSIS, fuzzy-TOPSIS, and extended TODIM. It has been already discussed that RMo-TOPSIS ranks alternatives similar to all the above methods, while it supersedes them in handling rank reversal and QoS data types. Table 17 compares RMo-TOPSIS with the above methods in terms of handling data type and rank reversal. RMo-TOPSIS is capable of dealing with a decision matrix having crisp, interval, TFN, IFS, and NFS, as shown in its cloud service selection application in the case study. It is also free from rank reversal, as shown in the sensitivity analysis section. Mo-TOPSIS can only handle QoS data types of crisp, interval, TFN, and IFS, while it suffers from rank reversal, as shown in the introduction section. G-TOPSIS extends the original TOPSIS, which is robust in rank reversal and works only with the crisp data type. Fuzzy TOPSIS is another extension of TOPSIS in the fuzzy environment, which ranks alternatives based on QoS of TFN types. It is also robust to rank reversal. Extended TODIM is another MCDM that ranks alternatives based on QoS data types crisp, interval, and fuzzy, and it is also robust to rank reversal. It can be observed from the above discussion that RMo-TOPSIS outperforms all the above methods in terms of handling different QoS data types and robustness against rank reversal. So, RMo-TOPSIS will be a good choice for decision-makers to make a consistent decision in the presence of heterogeneous QoS information.

5 Conclusion and future work

In this paper, a novel MCDM based RMo-TOPSIS method is presented to evaluate alternatives accurately and consistently. This method is the generalization of TOPSIS, which can process a decision matrix with a different type of QoS like crisp, interval, TFN, IFS and NFS. RMo-TOPSIS has several advantages in comparison to existing methods. First, it does not transform the heterogeneous decision matrix into homogenous and avoids loss of information in the transformation process, which in turn makes it a more consistent and reliable method. Second, it is the first rank reversal free generalization of TOPSIS for heterogenous decision matrix. The elimination of rank reversal further improves

Table 17 Comparison of RMo-TOPSIS and Mo-TOPSIS

Method	Crisp	Interval	TFN	IFS	NFS	Rank reversal robustness
Proposed RMo-TOPSIS	✓	✓	✓	✓	✓	✓
Mo-TOPSIS (Lourenzutti and Krohling 2016)	✓	✓	✓	✓	x	x
G-TOPSIS (Tiwari et al. 2021)	✓	x	x	x	x	✓
Fuzzy TOPSIS (Kumar et al. 2017)	x	x	✓	x	x	✓
Extended TODIM (Fan et al. 2013)	✓	✓	✓	x	x	✓

the consistent and robust ranking of alternatives. The proposed method uses two standard fictitious alternatives, PIS and NIS to remove rank reversal from Mo-TOPSIS. Third, it is the first method that can process heterogeneous decision matrix with NFS type QoS.

An application of RMo-TOPSIS is presented in cloud service selection to show its practicability and efficacy in the decision-making process. Its ranking result is validated by comparing its ranking with existing MCDM methods for heterogeneous decision matrix and variations of TOPSIS for homogeneous QoS. It was found that it ranks cloud services consistently and is nearly similar to other methods. A sensitivity analysis consisting of a series of experiments is executed to show its rank reversal robustness and found that it is robust to rank reversal. Overall, the proposed method provides a consistent way to evaluate cloud services. In the future, RMo-TOPSIS can be integrated with other data types to ranks alternatives.

Acknowledgements We are thankful to the TEQIP-III project of the Ministry of Human Resource Development, Government of India for financially supporting this study.

Data Availability Statement The authors confrm that the data supporting the findings of this study are available within the article.

References

Abdel-Basset M, Manogaran G, Gamal A, Smarandache F (2019) A group decision making framework based on neutrosophic TOPSIS approach for smart medical device selection. J Med Syst 43(3):38

Abedinia O, Zareinejad M, Doranehgard MH, Fathi G, Ghadimi N (2019) Optimal offering and bidding strategies of renewable energy based large consumer using a novel hybrid robust-stochastic approach. J Clean Prod 215:878–888

Akram M, Kahraman C, Zahid K (2021) Extension of TOPSIS model to the decision-making under complex spherical fuzzy information. Soft Comput 25(16):10771–10795

Atanassov KT (1986) Intuitionistic fuzzy sets. Fuzzy Sets Syst 20:87–96

Atanassov KT (1994) New operations defined over the intuitionistic fuzzy sets. Fuzzy Sets Syst 61(2):137–142

Biswas P, Pramanik S, Giri BC (2016) TOPSIS method for multi-attribute group decision-making under single-valued neutrosophic environment. Neural Comput Appl 27(3):727–737

Boran FE, Genç S, Kurt M, Akay D (2009) A multi-criteria intuitionistic fuzzy group decision making for supplier selection with TOPSIS method. Expert Syst Appl 36(8):11363–11368

Chen CT (2000) Extensions of the TOPSIS for group decision-making under fuzzy environment. Fuzzy Sets Syst 114(1):1–9

Chowdhury RR, Chattopadhyay S, Adak C (2020) CAHPHF: context-aware hierarchical QoS prediction with hybrid filtering. IEEE Transactions on Services Computing

Dağdeviren MY (2009) Weapon selection using the AHP and TOPSIS methods under fuzzy environment. Expert Syst Appl 36(4):8143–8151

Dymova L, Sevastjanov P, Tikhonenko A (2013) A direct interval extension of TOPSIS method. Expert Syst Appl 40(12):4841–4847

Espinilla M, de Andrés R, Martínez FJ, Martínez L (2013) A 360-degree performance appraisal model dealing with heterogeneous information and dependent criteria. Inf Sci 222:459–471

Fan ZP, Zhang X, Chen FD, Liu Y (2013) Extended TODIM method for hybrid multiple attribute decision making problems. Knowl-Based Syst 42:40–48

Gao H, Xu Y, Yin Y, Zhang W, Li R, Wang X (2019a) Context-aware QoS prediction with neural collaborative filtering for Internet-of-Things services. IEEE Internet Things J 7(5):4532–4542

Gao W, Darvishan A, Toghani M, Mohammadi M, Abedinia O, Ghadimi N (2019b) Different states of multi-block based forecast engine for price and load prediction. Int J Electr Power Energy Syst 104:423–435

García-Cascales MS, Lamata MT (2012) On rank reversal and TOPSIS method. Math Comput Model 56(5–6):123–132

Garg SK, Versteeg S, Buyya R (2013) A framework for ranking of cloud computing services. Future Gener Comput Syst 29(4):1012–1023

Ghadimi N, Akbarimajd A, Shayeghi H, Abedinia O (2018) Two stage forecast engine with feature selection technique and improved meta-heuristic algorithm for electricity load forecasting. Energy 161:130–142

Huang HL (2016) New distance measure of single-valued neutrosophic sets and its application. Int J Intell Syst 31(10):1021–1032

Huang T, Zhao R, Bi L, Zhang D, Lu C (2021) Neural embedding singular value decomposition for collaborative filtering. IEEE Transactions on Neural Networks and Learning Systems

Hussain A, Chun J, Khan M (2020a) A novel customer-centric Methodology for Optimal Service Selection (MOSS) in a cloud environment. Future Gener Comput Syst 105:562–580

Hussain A, Chun J, Khan M (2020b) A novel framework towards viable cloud service selection as a service (cssaas) under a fuzzy environment. Future Gener Comput Syst 104:74–91

Jahan A, Ismail MY, Sapuan SM, Mustapha F (2010) Material screening and choosing methods—a review. Mater Des 31(2):696–705

Jahanshahloo GR, Lotfi FH, Davoodi AR (2009) Extension of TOPSIS for decision-making problems with interval data: interval efficiency. Math Comput Model 49(5–6):1137–1142

Keikha A (2022) Generalized hesitant fuzzy numbers and their application in solving MADM problems based on TOPSIS method. Soft Comput 26(10):4673–4683

Khodaei H, Hajiali M, Darvishan A, Sepehr M, Ghadimi N (2018) Fuzzy-based heat and power hub models for cost-emission operation of an industrial consumer using compromise programming. Appl Therm Eng 137:395–405

Krohling RA, Campanharo VC (2011) Fuzzy TOPSIS for group decision making: a case study for accidents with oil spill in the sea. Expert Syst Appl 38(4):4190–4197

Kumar RR, Mishra S, Kumar C (2017) Prioritizing the solution of cloud service selection using integrated MCDM methods under fuzzy environment. J Supercomput 73(11):4652–4682

KutluGündoğdu F, Kahraman C (2019) Spherical fuzzy sets and spherical fuzzy TOPSIS method. J Intell Fuzzy Syst 36(1):337–352

Lee G, Jun KS, Chung ES (2014) Robust spatial flood vulnerability assessment for Han River using fuzzy TOPSIS with α-cut level set. Expert Syst Appl 41(2):644–654

Li DF, Huang ZG, Chen GH (2010) A systematic approach to heterogeneous multiattribute group decision making. Comput Ind Eng 59(4):561–572

Liu M, Shao Y, Yu C, Yu J (2020) A heterogeneous QoS-based cloud service selection approach using entropy weight and GRA-ELECTRE III. Mathematical Problems in Engineering, 2020

Lourenzutti R, Krohling RA (2016) A generalized TOPSIS method for group decision making with heterogeneous information in a dynamic environment. Inf Sci 330:1–18

Monika SOP (2022) A framework for evaluating cloud computing services using AHP and TOPSIS approaches with interval valued spherical fuzzy sets. Clust Comput 25(6):4383–4396

Mousavi-Nasab SH, Sotoudeh-Anvari A (2017) A comprehensive MCDM-based approach using TOPSIS, COPRAS and DEA as an auxiliary tool for material selection problems. Mater Des 121:237–253

Peng DH, Gao CY, Wu LX (2012) TOPSIS-based multi-criteria group decision making under heterogeneous information setting. In: Advanced materials research, vol 378. Trans Tech Publications Ltd., pp 525–530

Purohit LA (2020) A study on evolutionary computing based web service selection techniques. Artif Intell Rev 54(2):1117–1170

Regunathan R, Murugaiyan A, Lavanya K (2018) Neural based QoS aware mobile cloud service and its application to preeminent service selection using back propagation. Procedia Comput Sci 132:1113–1122

Roy B (1990) The outranking approach and the foundations of ELECTRE methods. Multiple criteria decision aid. Springer, Berlin, Heidelberg, pp 155–183

Saaty TL (1990) How to make a decision: the analytic hierarchy process. Eur J Oper Res 48(1):9–26

Saaty TL (2006) Decision making with the analytic network process. Springer Science+ Business Media, LLC, Berlin

Sadabadi SA, Hadi-Vencheh A, Jamshidi A, Jalali M (2022) An improved fuzzy TOPSIS method with a new ranking index. Int J Inf Technol Decis Mak 21(02):615–641

Saeedi M, Moradi M, Hosseini M, Emamifar A, Ghadimi N (2019) Robust optimization based optimal chiller loading under cooling demand uncertainty. Appl Therm Eng 148:1081–1091

Tiwari RK, Kumar R (2021) G-TOPSIS: a cloud service selection framework using Gaussian TOPSIS for rank reversal problem. J Supercomput 77(1):523–562

Tsaur RC (2011) Decision risk analysis for an interval TOPSIS method. Appl Math Comput 218(8):4295–4304

Wang YM, Luo Y (2009) On rank reversal in decision analysis. Math Comput Model 49(5–6):1221–1229

Wang JG, Wang RQ (2008) Hybrid random multi-criteria decision-making approach with incomplete certain information. In: Chinese control and decision conference. IEEE, Yantai, Shandong, pp 1444–1448

Wang H, Smarandache F, Zhang YQ, Sunderraman R (2010) Single valued neutrosophic sets. Multispace Multistruct 4:410–413

Wang Y, Liu P, Yao Y (2022) BMW-TOPSIS: a generalized TOPSIS model based on three-way decision. Information Sciences

Wu D, Mendel JM (2010) Computing with words for hierarchical decision making applied to evaluating a weapon system. IEEE Trans Fuzzy Syst 18(3):441–460

Xu Z, Zhang X (2013) Hesitant fuzzy multi-attribute decision making based on TOPSIS with incomplete weight information. Knowl-Based Syst 52:53–64

Xu J, Xiao L, Li Y, Huang M, Zhuang Z, Weng TH, Liang W (2021) NFMF: neural fusion matrix factorisation for QoS prediction in service selection. Connect Sci 33(3):753–768

Yang M, Zhu H, Guo K (2020) Research on manufacturing service combination optimization based on neural network and multi-attribute decision making. Neural Comput Appl 32(16):1691–1700

Ye J (2015) Trapezoidal neutrosophic set and its application to multiple attribute decision-making. Neural Comput Appl 26(5):1157–1166

Yoon K, Hwang CL (1981) TOPSIS (technique for order preference by similarity to ideal solution)—a multiple attribute decision making : multiple attribute decision making—methods and applications, a state-of-the-at survey. Springer Verlag, Berlin

Yousefi A, Hadi-Vencheh A (2010) An integrated group decision making model and its evaluation by DEA for automobile industry. Expert Syst Appl 37(12):8543–8556

Yue Z (2011) An extended TOPSIS for determining weights of decision makers with interval numbers. Knowl-Based Syst 24(1):146–153

Zadeh LA (1965) Fuzzy sets. Inf Control 8(3):338–353