Not All Clouds Are Transparent: Handling Unavailable Attributes in CSP Selection

P Navya, Sanjaya Kumar Panda, ***Senior Member, IEEE***, and Rashmi Ranjan Rout, ***Senior Member, IEEE***
Department of Computer Science and Engineering
National Institute of Technology Warangal, Warangal - 506004, Telangana, India
np23csr1r03@student.nitw.ac.in, sanjaya@nitw.ac.in, rashrr@nitw.ac.in

I. Introduction

Selecting a cloud service provider (CSP) is a complex task for enterprises, as each CSP offers a distinct set of quality of service (QoS) attributes with varying values [1]. These attributes include durability, response time, best practices, throughput, availability, compliance, latency, reliability, and successability [2]. Traditionally, researchers have employed a multi-attribute decision-making (MADM) algorithm to select a suitable CSP, operating under the assumption that complete QoS attribute values are available. However, certain QoS attribute values may be unavailable in many CSPs. This lack of values makes the selection process non-transparent for enterprises, hindering their ability to identify the most suitable CSP. Consequently, the unavailable values can be imputed using either simple or advanced techniques to facilitate the selection of a suitable CSP. Common imputation techniques include mean, minimum, maximum, regression, k-nearest neighbour, and rough set theory. After the imputation process, an MADM algorithm can then be applied to identify the most suitable CSP. Such MADM algorithms include the technique for order preference by similarity to ideal solution (TOPSIS), the best holistic adaptable ranking of attributes technique (BHARAT), and multi-objective optimization on the basis of ratio analysis (MOORA) [3], [4].

Tomar et al. [3] have developed a hybrid MADM algorithm for selecting a suitable CSP using the TOPSIS algorithm. They have performed a sensitivity analysis to ensure a robust and comprehensive assessment of CSP. Jong and Ahmed [4] have used the TOPSIS algorithm to identify an optimal solar energy site. The above works have selected a suitable CSP or solar energy sites by assuming all the QoS attribute values are available. Recently, our earlier work has considered unavailable QoS attribute values and applied three simple imputation techniques: mean, minimum, and maximum [2]. It reveals that TOPSIS, combined with the mean imputation technique (i.e., QTOPSIS), outperforms other techniques. This phenomenon motivates us to develop an MADM algorithm with an enhanced imputation technique to further improve the efficiency of cloud service selection. This paper presents the QoS-aware rough set TOPSIS (Q-RUST) algorithm, which integrates the rough set theory (RST) and the TOPSIS algorithm. Q-RUST imputes unavailable values with lower and upper approximations, as applicable, and applies the TOPSIS algorithm to determine the best CSP. The lower and upper approximations define the boundary region, representing the certain and possible values of the unavailable QoS attributes.

II. Design

Consider an incomplete decision matrix, as depicted in Eq. (1), which consists of m CSPs, n QoS attributes, and their corresponding weights W. Here, P_{ij} is the performance value of CSP_i, $1 \le i \le m$, on QoS attribute A_j, $1 \le j \le n$, and $\times$ denotes unavailable QoS attribute values with respect to CSPs. Each attribute A_j is classified as either beneficial (maximized) or non-beneficial (minimized). The problem is to select the most suitable CSP when some of the QoS attribute values for CSPs in the decision matrix are unavailable. Alternatively, the problem is to rank the CSPs based on their comparative performance.

$$\begin{array}{c} \\ \\ CSP_1 \\ CSP_2 \\ \vdots \\ CSP_m \end{array} \begin{array}{c} \begin{array}{cccc} A_1 & A_2 & \cdots & A_n \\ W_1 & W_2 & \cdots & W_n \end{array} \\ \left\{ \begin{array}{cccc} P_{11} & \times & \cdots & P_{1n} \\ P_{21} & P_{22} & \cdots & \times \\ \vdots & \vdots & \ldots & \vdots \\ P_{m1} & P_{m2} & \cdots & P_{mn} \end{array} \right\} \end{array} \tag{1}$$

To determine the most suitable CSP from the decision matrix in Eq. (1) when some values are unavailable, the Q-RUST algorithm is employed. Q-RUST is an MADM algorithm that ranks CSPs by comparing them to an ideal solution. It involves data normalization, handling unavailable values using RST, finding the maximum and minimum values, calculating the best and worst ideal solutions, determining the overall score, and ranking CSPs, which are described as follows.

(1) Data Normalization: It is performed by dividing the performance value, P_{ij}, by the square root of the sum of squares of the corresponding QoS attribute values across all the CSPs. Mathematically,

$$N_{ij} = \frac{P_{ij}}{\sqrt{\sum_{i=1}^{m} (P_{ij})^2}}, 1 \le i \le m, 1 \le j \le n \tag{2}$$

where N_{ij} is the normalized performance value of CSP_i on QoS attribute A_j.

(2) Handling Unavailable Values using RST: To handle unavailable values, RST is employed in the decision matrix [5]. RST approximates unavailable values based on the similarity among attribute values through lower and upper

approximations. The cloud service selection (S) is defined as a two-tuple, $<C, A>$, where C is the set of CSPs and $A = CA \cup DA$, in which CA represents the condition attributes and DA represents the decision attribute. The indiscernibility (IND) relation for B is mathematically expressed as follows.

$$IND(B) = \{(x, y) \in C \times C \mid \forall a \in B,\ a(x) = a(y)\} \quad (3)$$

where B is any subset of the set C. The equivalence class of $x \in U$ is $[x]_B = \{y \in C \mid (x, y) \in IND(B)\}$. For each decision class $X \subseteq C$, the lower and upper approximations are mathematically defined as follows.

$$\underline{B}(X) = \{x \in C \mid [x]_B \subseteq X\} \quad (4)$$

$$\overline{B}(X) = \{x \in C \mid [x]_B \cap X \neq \emptyset\} \quad (5)$$

If the performance value is unavailable, RST estimates it based on the equivalence class to which the object belongs as follows. If the value lies in the lower approximation (certainly belongs to a QoS attribute value), the unavailable value of that QoS attribute replaces within $\underline{B}(X)$. If the value lies in the upper approximation (may belong to a QoS attribute value), the unavailable value is replaced by the most frequent value among the QoS attribute values in $\overline{B}(X)$. This RST-based imputation preserves the dependency relationships between condition and decision attributes, allowing for the realistic estimation of unavailable values.

(3) Finding the Maximum and Minimum Values: Q-RUST multiplies the weights by the normalized value and calculates the minimum and maximum value of each QoS attribute.

(4) Calculate the Best and Worst Ideal Solutions: The ideal best and worst solutions are calculated using Eq. (6) and Eq. (7). Here, the maximum (or minimum) value is used for beneficial attributes, and the minimum (or maximum) value is used for non-beneficial attributes to determine the ideal best (or worst) solution. Mathematically,

$$D_{ibest} = \sqrt{\sum_{j=1}^{n} (N_{ij} - max_j \text{ or } min_j)^2}, 1 \leq i \leq m \quad (6)$$

$$D_{iworst} = \sqrt{\sum_{j=1}^{n} (N_{ij} - min_j \text{ or } max_j)^2}, 1 \leq i \leq m \quad (7)$$

(5) The overall score for each CSP is determined as follows.

$$SI_i = \frac{D_{iworst}}{D_{ibest} + D_{iworst}} \quad (8)$$

(6) Finally, the Q-RUST algorithm ranks all CSPs, and the CSP with the highest overall score is selected as the most suitable CSP.

III. Evaluation

The simulations were conducted using the QoS for web services (QWS) dataset [6] to evaluate the performance of the Q-RUST algorithm in comparison with the QTOPSIS and the traditional TOPSIS algorithms. This dataset consists of 2507 CSPs and 8 attributes. The results of the QTOPSIS algorithm were taken from our earlier work [2] for comparison. Fig. 1 illustrates that the top three CSPs remain consistent across all three algorithms, demonstrating high correlation in ranking performance, with minor variations observed in lower-ranked CSPs. Further, the sensitivity analysis shown in Fig. 2 confirms that the Q-RUST algorithm exhibits strong robustness, maintaining stable CSP rankings across ten different scenarios.

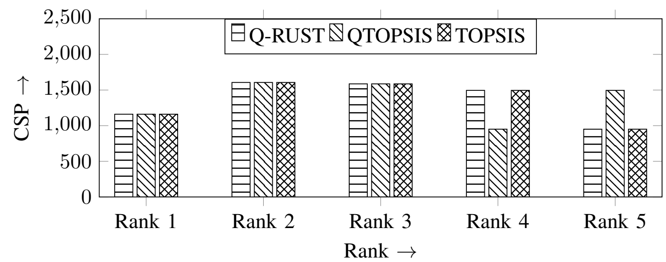

Fig. 1: Comparison of top five CSPs.

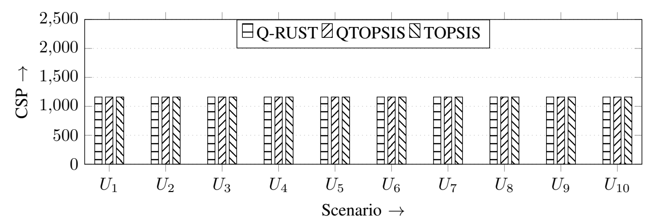

Fig. 2: Sensitivity analysis of the Q-RUST, QTOPSIS, and TOPSIS algorithms.

IV. Conclusion and Future Work

This paper has introduced the Q-RUST algorithm, which integrates RST and the TOPSIS algorithm to impute unavailable values and enhance the efficiency of CSP ranking. The proposed algorithm effectively handles incomplete QoS attribute values, providing stable and reliable ranking outcomes. This study can be further extended by integrating fuzzy and hybrid imputation strategies or applying the framework to other service selection domains, such as edge and fog computing environments.

References

[1] R. Garg, "Mcdm-based parametric selection of cloud deployment models for an academic organization," *IEEE Transactions on Cloud Computing*, vol. 10, no. 2, pp. 863–871, 2022.

[2] P. Navya, S. K. Panda, and R. R. Rout, "Quality of service-centric multi-attribute decision-making algorithms for handling unavailable performance data in cloud service selection," *Cluster Computing*, vol. 28, no. 9, p. 561, 2025.

[3] A. Tomar, R. R. Kumar, and I. Gupta, "Decision making for cloud service selection: a novel and hybrid mcdm approach," *Cluster computing*, vol. 26, no. 6, pp. 3869–3887, 2023.

[4] F. C. Jong and M. M. Ahmed, "Multi-criteria decision-making solutions for optimal solar energy sites identification: A systematic review and analysis," *IEEE Access*, 2024.

[5] Q. Zhang, Q. Xie, and G. Wang, "A survey on rough set theory and its applications," *CAAI Transactions on Intelligence Technology*, vol. 1, no. 4, pp. 323–333, 2016.

[6] E. Al-Masri and Q. H. Mahmoud, "Investigating web services on the world wide web," https://qwsdata.github.io/, pp. 795–804, 2008.