Trusted Cloud Service Selection Algorithm Based on Lightweight Intuitionistic Fuzzy Numbers

YULI YANG, NANYUE YU, AND YONGLE CHEN, (Member, IEEE)
College of Information and Computer, Taiyuan University of Technology, Taiyuan 030024, China
Corresponding author: Yuli Yang (yangyuli@tyut.edu.cn)

This work was supported in part by the Natural Science Foundation of Shanxi Province under Grant 201901D211076, and in part by the National Key Research and Development Program of China under Grant 2018YFB0803402.

ABSTRACT
Trusted cloud service selection algorithms aim to select the cloud service with the highest satisfaction from many cloud services with the same or similar functions. However, many existing selection algorithms adopt the inaccurate QoS measurement method and subjective weights of QoS attributes, which make it difficult for cloud users to select the trusted cloud service. Therefore, this paper proposes a trusted cloud service selection algorithm based on lightweight intuitionistic fuzzy numbers. Firstly, based on the consideration of precise and imprecise QoS metrics, three kinds of QoS attributes are transformed into lightweight intuitionistic fuzzy numbers in a uniform way, and corresponding weight coefficients are expressed with the calculated standard deviation. Then, the trusted cloud service selection algorithm based on improved TOPSIS is proposed to guide cloud users to choose the cloud service with the highest satisfaction. The accuracy of the proposed algorithm was validated by two cases and the efficiencies of the proposed algorithm and the traditional cloud service ranking algorithm based on triangular fuzzy numbers were compared.

INDEX TERMS
Cloud service selection, SMI, intuitionistic fuzzy number, TOPSIS.

I. INTRODUCTION
In recent years, as one of the most promising computing technologies, cloud computing has grown enormously since it can deliver large-scale information technology (IT) resources over the Internet, and provide Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as Service (SaaS) with pay-as-you-go model [1]. In order to exploit overwhelming benefits of rapid elasticity and disaster recovery provided by cloud services, instead of buying and managing IT resources, many Cloud Users (CUs) have purchased or rented cloud services at competitive prices. Computing resources and storage resources are used as metered services, so that CUs can focus on their own business [2].

However, a large number of Cloud Service Providers (CSPs) exist in the increasingly competitive cloud service market. These services have the similar or same functions but different Quality of Service (QoS), so CUs have the difficulty in the selection of the CSPs with the highest satisfaction according to their own QoS requirements [3]. For example, when a user rents a PaaS, it is difficult to choose the trusted one among AWS Elastic Beanstalk, Heroku, Force.com, Google App Engine, etc.

Trusted cloud service selection refers to the evaluation of QoS attributes of CSPs according to CUs’ demands for selecting the cloud service with the highest satisfaction. It plays a vital role in promoting the further development of cloud services, and is widely concerned among the experts in the business community and academia. At present, trusted cloud service selection involves the following challenges: accurate measurements of different types of QoS [4], objective descriptions of CUs’ preference demands of different QoS attributes [5], and rapid ranking of the cloud services according to the demands of CUs [6].

To solve the above challenges, this paper focuses on the measurement of QoS, calculation of QoS weight coefficient and the ranking of cloud services, and proposes a trusted cloud service selection algorithm based on Lightweight Intuitionistic Fuzzy Numbers (LIFNs). The algorithm can guide CUs to select the most satisfying cloud service. The contributions in the study are summarized below.
Different types of QoS attributes are quantified with LIFNs in a uniform way. The method based on the calculated standard deviation is designed to express the CUs’ preference weights of different QoS attributes. The trusted cloud service selection algorithm based on the improved Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) is proposed.

The remainder of this paper is organized as follows. Section II introduces related studies on cloud service selection. Section III presents the preliminary knowledge of Intuitionistic Fuzzy Sets (IFSs) and TOPSIS. Section IV describes the trusted cloud service selection algorithm based on LIFNs. Section V discusses the accuracy and efficiency of the proposed algorithm through simulation experiments. Section VI gives the conclusions.

II. RELATED STUDIES
The calculation methods of QoS attributes and their weights, and the cloud service selection algorithms are introduced below.

A. CALCULATION METHODS OF QoS ATTRIBUTES AND THEIR WEIGHTS
To describe QoS attributes and CUs’ preferences effectively, a lot of methods have been designed by researchers. Taking into account the ambiguity and uncertainty of QoS attributes and CUs’ preferences, the fuzzy numbers were used to describe them for uniform handling [7]. In order to deal with linguistic variables effectively, each linguistic variable was transformed into a fuzzy set and triangular fuzzy numbers was used to evaluate the weights of different attributes [8]. As for QoS attributes described with crisp numbers, based on the consideration of the user’s objective and subjective preferences, the entropy method and the fuzzy AHP method were designed to calculate objective and subjective weights, respectively [9]. In digital hospitality industry, a new service quality model based on intuitionistic fuzzy number AHP was designed to determine the weights of different attributes [10]. To choose the trusted cloud service, fuzzy logic was used to describe different attributes and expert evaluation method was used to determine the weighting coefficients of different attributes [11]. As for QoS attributes described with exact values, in order to describe the changes in users’ preferences effectively, the MCDM method based on Markov chain and the best-worst method were proposed [12].

B. CLOUD SERVICE SELECTION ALGORITHMS
Cloud service selection is a typical multi-criteria decision-making (MCDM) problem. Most cloud service selection algorithms can be roughly divided into two main categories: multi-objective optimization and multi-attribute optimization. According to the above classification method, the state-of-the-art cloud service selection methods have been reviewed and the differences between the study and previous studies are summarized below.

1) MULTI-OBJECTIVE OPTIMIZATION
Most previous studies treated cloud service selection as a multi-objective optimization problem and converted it into a multi-objective programming model. To optimize several conflicting criteria simultaneously in the selection process of cloud services, the improved artificial bee colony algorithm was used to deal with the complicated multi-objective optimization problem [13]. Considering the interdependencies among various performance measurements, Al-Faifi et al. proposed a hybrid multi-criteria decision method to guide consumers to select the best CSPs [14]. Based on the consideration of the accuracy and diversity of cloud service recommendation, the modified ranking prediction and recommendation algorithms were proposed to simultaneously improve ranking prediction and users’ satisfaction [15]. Considering various kinds of optimization objectives, a new hybrid multi-objective evolutionary algorithm was designed to optimize the Inter-Cloud service selection and composition [16]. To choose the best cloud service, the enhanced multi-objective particle swarm optimization algorithm and hierarchical clustering algorithm were used [17]. A particle swarm optimization based multi-objective algorithm was proposed to search for the feasible service with the lower cost [18].

2) MULTI-ATTRIBUTE OPTIMIZATION
The problem of the multi-attribute cloud service selection was deemed as a multi-attribute optimization problem and transformed into a single-objective function. Compliance-based multi-dimensional trust evaluation system was designed to evaluate the trustworthiness of cloud service providers from different perspectives [19]. In order to select optimal manufacturing services, Li et al. proposed a hybrid MCDM method utilizing the rough set theory to handle the vagueness and uncertainty of cloud manufacturing services effectively [20]. The cloud service model was considered as a business model and a novel two-way ranking method was proposed to rank both cloud service providers and users [21]. In order to deal with uncertain information, the fuzzy linear best-worst method was proposed to rank different CSPs [22]. A novel 2-tuple fuzzy linguistic multi-criteria group decision-making method was designed for e-commerce [23]. Based on the consideration of analytical hierarchical process and TOPSIS, a hybrid MCDM method was proposed to rank the cloud services in terms of the overall performance [24].

Existing methods have some defects, such as inaccurate metrics of QoS attributes and subjective weights of CUs’ preference, which may lead to inaccurate ranking results. Besides, the trusted cloud service selection algorithms mainly aim to recommend the cloud service with the highest evaluation value to CUs and may cause the problem of service overload. The proposed algorithm in the paper is designed
to quantify QoS attributes uniformly, calculate objective weights of CUs’ preference, and recommend the cloud service with the highest satisfaction to CUs.

III. PRELIMINARIES
In order to better understand the idea in this paper, the preliminaries of IFSs and TOPSIS are discussed below.

A. IFSs
Compared with traditional fuzzy sets, IFSs proposed by Atanassov considers the three aspects of membership degree, non-membership degree and hesitation degree [25]. Therefore, IFSs is an extension of traditional fuzzy sets. It is more flexible and practical in dealing with ambiguity and uncertainty and has been applied in many fields, such as decision-making, medical diagnosis, and market forecasting.

The related definitions of IFSs [26] are provided as follows.

Definitions: With $X$ as a given universe of discourse, IFSs $A$ on $X$ can be defined as $A = \{x, \mu_A(x), v_A(x)\}$ ($x \in X$), in which $\mu_A(x)$ and $v_A(x)$ indicate membership degree function and non-membership degree function; $0 \leq \mu_A(x) \leq 1$, $0 \leq v_A(x) \leq 1$ and $0 \leq \mu_A(x) + v_A(x) \leq 1$ are the restriction conditions; $\pi_A(x)$ is the third parameter indicating the degree of hesitation of $x$ to $A$. It can be calculated as $\pi_A(x) = 1 - \mu_A(x) - v_A(x)$ and satisfies $0 \leq \pi_A(x) \leq 1$, $x \in X$.

B. TOPSIS
TOPSIS method was proposed by Hwang and Yoon [27]. Its basic principle is to select the optimal solution based on the calculated distances of each alternative to positive ideal solutions (PIS) and negative ideal solutions (NIS). PIS is the most preferred alternative with the maximum attribute values, whereas NIS is the least preferred alternative with the minimum attribute values. TOPSIS gives the optimal candidate which has the shortest Euclidean distance away from PIS and the farthest Euclidean distance away from NIS. TOPSIS is one of the most efficient methods used to solve MCDM problems [28], [29].

IV. TRUSTED CLOUD SERVICE SELECTION ALGORITHM
Trusted Cloud Service Selection Algorithm based on LIFNs (TCSSAL) is designed to help CUs to select the trusted cloud service. In the algorithm, different QoS attributes are described with LIFNs in a unified way and TOPSIS is used to rank CSPs. Because TOPSIS could ensure the maximum profits of CUs and CSPs, avoid risks as much as possible, and help CUs to select the cloud service with the highest satisfaction according to their demands. The details of the proposed algorithm is illustrated below.

A. METRICS FOR DIFFERENT TYPES OF QoS ATTRIBUTES
Service measurement index (SIM) is used to evaluate cloud services according to CUs’ demands [30]. QoS attributes contained in SIM can be divided into three types: the attributes described with the exact value, the interval value, and the language value. Supposing that there are $N$ CSPs with $M$ QoS attributes. The attribute evaluation matrix $A$ is expressed in Eq. (1), where $m_{ij}(i = 1, 2, \cdots, N; j = 1, 2, \cdots, M)$ represents the $j$-th attribute value of the $i$-th cloud service provider; $m_{i(N+1)j}$ represents the user’s demand value for the $j$-th attribute.

$$A = \begin{bmatrix} m_{11} & \cdots & m_{1M} \\ \vdots & \ddots & \vdots \\ m_{i(N+1)1} & \cdots & m_{i(N+1)M} \end{bmatrix}$$

In order to measure the QoS attributes of cloud services accurately and simply, the method based on LIFNs is designed to describe three types of QoS attributes uniformly. Each LIFNs only contains $\mu_A(x)$ and $v_A(x)$.

With LIFNs, the matrix $A_{(N+1) \times M}$ is converted into the matrix $B_{(N+1) \times M} = f_{ij} = (\mu_{ij}, v_{ij})(i = 1, 2, \cdots, N + 1; j = 1, 2, \cdots, M)$. The process of converting three kinds of QoS attributes into LIFNs is described as follows.

1) EXACT VALUE
$m_{ij}$ used to describe the exact value is normalized with Eq. (2).

$$rm_{ij} = \begin{cases} \frac{\text{max}_j - m_{ij}}{\text{max}_j - \text{min}_j} & \text{if } j \text{ is negative} \\ \frac{m_{ij} - \text{min}_j}{\text{max}_j - \text{min}_j} & \text{if } j \text{ is positive} \end{cases}$$

where $\text{max}_j$ and $\text{min}_j$ respectively represent the maximum and minimum values of the $j$-th attribute of all CSPs; $rm_{ij}(i = 1, 2, \cdots, N; j = 1, 2, \cdots, M)$ represents the normalized value of the $j$-th attribute of the $i$-th cloud service provider; $rm_{i(N+1)j}$ represents the normalized measure of the user’s demand for the $j$-th attribute.

2) INTERVAL VALUE
The interval value $[m_{ij}^L, m_{ij}^U]$ is normalized with Eq. (4), in which $m_{ij}^L$ and $m_{ij}^U$ are the lower and upper bounds of the interval, respectively. Then, the normalized interval value $[rm_{ij}^L, rm_{ij}^U]$ is converted into LIFNs with Eq. (5).

3) LANGUAGE VALUE
Language values describing the attributes, such as distrust or high trust, are converted into LIFNs according to
Table 1.

| Level | Language terms | Interval values | LIFNs |
|-------|----------------|-----------------|-------|
| 1     | distrust       | [0.0, 0.2]      | (0.0, 0.8) |
| 2     | low trust      | (0.2, 0.4)      | (0.2, 0.6) |
| 3     | neutral trust  | (0.4, 0.6)      | (0.4, 0.4) |
| 4     | high trust     | (0.6, 0.8)      | (0.6, 0.2) |
| 5     | absolute trust | (0.8, 1.0)      | (0.8, 0.0) |

C. CLOUD SERVICE RANKING BASED ON IMPROVED TOPSIS

According to CUs’ demands, the method based on the improved TOPSIS is used to rank different CSPs according to the following steps.

Step 1: The user’s weight matrix for different attributes is recorded as \( W = [W_1, W_2, \ldots, W_M] \). The lightweight intuitionistic fuzzy number matrix \( B \) is converted into the weighted lightweight intuitionistic fuzzy number matrix \( C \) with Eq. (9).

\[
C = B \otimes W
\]

where \( c_{ij} \) and \( cv_{ij} \) are denoted as the degrees of support and opposition, respectively. As for the attributes described with exact values, corresponding \( c_{ij} \) and \( cv_{ij} \) are calculated with Eq. (10). As for the attributes described with the interval values or language values, corresponding \( c_{ij} \) and \( cv_{ij} \) are calculated with Eq. (11).

\[
\begin{align*}
(c_{ij}, cv_{ij}) &= (W_j \cdot rm_{ij}, 1 - W_j \cdot rm_{ij}) \quad (10) \\
(c_{ij}, cv_{ij}) &= (W_j L \cdot rm^{L}_{ij}, W_j U \cdot (1 - rm^{U}_{ij})) \quad (11)
\end{align*}
\]

Step 2: The optimal value vector and the worst value vector are calculated with Eq. (12) and Eq. (13), in which \( C^+ (C^-) \) represents the optimal (worst) value vector of the weighted intuitionistic fuzzy matrix; \( c_{ij}^+(c_{ij}^-) \) represents the maximum (minimum) support to the \( j \)-th attribute; \( cv_{ij}^+(cv_{ij}^-) \) represents the minimum (maximum) opposition to the \( j \)-th attribute.

\[
\begin{align*}
C^+ &= [(c_{ij}^+, cv_{ij}^+), \ldots, (c_{ij}^+, cv_{ij}^+), \ldots, (c_{ij}^M, cv_{ij}^M)] \quad (12) \\
C^- &= [(c_{ij}^-, cv_{ij}^-), \ldots, (c_{ij}^-, cv_{ij}^-), \ldots, (c_{ij}^M, cv_{ij}^M)] \quad (13)
\end{align*}
\]

The calculation process of \( (c_{ij}^+, cv_{ij}^+) \) and \( (c_{ij}^-, cv_{ij}^-) \) is given as follows. First of all, the distance matrix \( D = [d_{11}, \ldots, d_{1M}, \ldots, d_{N1}, \ldots, d_{NM}] \) of CUs’ demands is calculated. \( d_j \) represents the subtraction between the \( j \)-th attribute value of the \( j \)-th cloud service provider and the \( j \)-th attribute value of CUs’
demands and is calculated with Eq. (14).

\[ d_{ij} = c\mu_{ij} - c\mu_{i(N+1)j} \]  (14)

Then, as for positive and negative attributes, according to different values of \( d_{ij} \), the computing processes of \((c\mu_{ij}^+, c\nu_{ij})\) and \((c\mu_{ij}^-, c\nu_{ij})\) are divided into three scenarios. The calculation method for positive attributes is given in detail.

1) WHEN ALL \( d_{ij} \) ARE GREATER THAN ZERO

The minimum (maximum) value of \( d_{ij} \) is denoted as \( d_{i1}(d_{Nj}) \). The weighted intuitionistic fuzzy number of the \( l \)-th cloud service provider, \((c\mu_{lj}, c\nu_{lj})\), is assigned to \((c\mu_{lj}^+, c\nu_{lj}^-)\) and the weighted intuitionistic fuzzy number of the \( k \)-th cloud service provider, \((c\mu_{kj}, c\nu_{kj})\), is assigned to \((c\mu_{kj}^-, c\nu_{kj}^-)\).

2) WHEN ONE PART OF \( d_{ij} \) ARE GREATER THAN ZERO, AND THE OTHER PART ARE LESS THAN ZERO

The minimum value of \( d_{ij} \) greater than zero is denoted as \( d_i \) and the minimum value of \( d_{ij} \) less than zero is denoted as \( d_k \). The weighted intuitionistic fuzzy number of the \( l \)-th cloud service provider, \((c\mu_{lj}, c\nu_{lj})\), is assigned to \((c\mu_{lj}^+, c\nu_{lj}^-)\) and the weighted intuitionistic fuzzy number of the \( k \)-th cloud service provider, \((c\mu_{kj}, c\nu_{kj})\), is assigned to \((c\mu_{kj}^-, c\nu_{kj}^-)\).

3) WHEN ALL \( d_{ij} \) ARE LESS THAN ZERO

The minimum (maximum) value of \( d_{ij} \) is denoted as \( d_{i1}(d_{Nj}) \). \((c\mu_{ij}^+, c\nu_{ij}^-)\) is set to the weighted intuitionistic fuzzy number of the \( l \)-th cloud service provider, \((c\mu_{lj}, c\nu_{lj})\), is assigned to \((c\mu_{lj}^+, c\nu_{lj}^-)\) and the weighted intuitionistic fuzzy number of the \( k \)-th cloud service provider, \((c\mu_{kj}, c\nu_{kj})\), is assigned to \((c\mu_{kj}^-, c\nu_{kj}^-)\).

The calculation process of \((c\mu_{ij}^+, c\nu_{ij}^-)\) and \((c\mu_{ij}^-, c\nu_{ij}^-)\) for negative attributes is opposite to that of positive attributes, so it is omitted here for the sake of brevity.

Step 3: Eq. (15) and Eq. (16) are used to calculate the distance between different CSPs and the optimal (worst) solution. \( D_i^+(D_i^-) \) indicates the distance between the \( i \)-th cloud service provider and the optimal (worst) solution.

\[ D_i^+ = \sqrt{\frac{1}{2} \sum_{j=1}^{M} [(c\mu_{ij} - c\mu_{ij}^+)^2 + (c\nu_{ij} - c\nu_{ij}^-)^2]} \]  (15)

\[ D_i^- = \sqrt{\frac{1}{2} \sum_{j=1}^{M} [(c\mu_{ij} - c\mu_{ij}^-)^2 + (c\nu_{ij} - c\nu_{ij}^-)^2]} \]  (16)

Step 4: The relative closeness degrees of CSPs are calculated with Eq. (17).

\[ \rho_i = \frac{D_i^-}{D_i^+ + D_i^-} \]  (17)

where \( \rho_i \) represents the relative closeness degree of the \( i \)-th cloud service provider and is between 0 and 1. The bigger the value of \( \rho_i \) is, the better the overall performance of the cloud service provider is. Therefore, \( \rho_i \) is used to rank all CSPs.

V. SIMULATION EXPERIMENT

The simulation experiment was carried out using Matlab R2015a on a laptop with Intel Core i7-3632 Quad Mobile processor at 2.2 GHz and 12 GB of RAM in 64-bit Windows 10 operating system. The accuracy and efficiency of TCSSAL were validated by the following simulation experiments.

A. ACCURACY OF TCSSAL

1) CASE 1: EVALUATION OF SECURITY OF CSPs

The security attributes of CSPs obtained from three IaaS providers are used to validate the proposed algorithm. Table 2 shows 21 evaluation metrics, including five positive attributes and sixteen negative attributes [31]. With Goal-Question-Metric (GQM) paradigm, an evaluation methodology of cloud security services based on objective weighting of metrics was proposed in the previous study [31]. As for three IaaS providers in Table 2, with the previous method [31], the relative evaluation values of them were ranked as CSP\(_1 > \) CSP\(_2 > \) CSP\(_3\). CSP\(_3\) had the highest level of security compliance, and CSP\(_1\) had the lowest level of security compliance. The proposed algorithm was used to rank those CSPs as follows.

Step 1: The security attribute matrix \( A \) was established according to Table 2.

\[
A = \begin{bmatrix}
100 & 89 & \cdots & 5 \\
80 & 72 & \cdots & 4 \\
70 & 60 & \cdots & 10
\end{bmatrix}
\]

Step 2: The normalized matrix \( A' \) was generated according to \( A \).

\[
A' = \begin{bmatrix}
1 & 1 & \cdots & 0.83 \\
0.33 & 0.41 & \cdots & 1 \\
0 & 0 & \cdots & 0
\end{bmatrix}
\]

Step 3: The normalized matrix \( A' \) was converted into the lightweight intuitionistic fuzzy number matrix \( B \) with Eqs. (3)-(5).

\[
B = \begin{bmatrix}
(1, 0) & (1, 0) & \cdots & (0.83, 0.17) \\
(0.33, 0.67) & (0.41, 0.59) & \cdots & (1, 0) \\
(0, 1) & (0, 1) & \cdots & (0, 1)
\end{bmatrix}
\]

Step 4: The weight coefficient of different attributes \( W \) was calculated with Eqs. (6)-(8).

\[
W = [0.047, 0.046, \cdots, 0.050]
\]

Step 5: The weighted lightweight intuitionistic fuzzy number matrix \( C \) was calculated with Eqs. (9)-(11).

\[
C = \begin{bmatrix}
(0.047, 0.953) & (0.05, 0.95) & \cdots & (0.042, 0.958) \\
(0.016, 0.984) & (0.02, 0.98) & \cdots & (0.05, 0.95) \\
(0, 1) & (0, 1) & \cdots & (0, 1)
\end{bmatrix}
\]

Step 6: The optimal (worst) value vector \( C^+(C^-) \) was calculated with Eqs. (12)-(14).

\[
C^+ = ((0.047, 0.953), (0.046, 0.954), \cdots, (0.0495, 0.9505))
\]
TABLE 2. Evaluation metrics of service availability for three different CSPs.

| Evaluation metrics                                                                 | CSP₁   | CSP₂   | CSP₃   |
|-----------------------------------------------------------------------------------|--------|--------|--------|
| Applications deployed with implementation of security incident response plan      | 100    | 80     | 70     |
| applications deployed with configuration of SIEM incident reporting               | 89     | 72     | 60     |
| systems deployed with implementation of regular testing procedures                | 90     | 80     | 66     |
| FRC and DDoS attack detection rate(%)                                             | 99     | 97     | 95     |
| system deployed with implementation of redundancies                               | 79     | 74     | 80     |
| Mean-time of incident discovery(h)                                                | 1      | 1.2    | 1.3    |
| Mean-time of incident recovery(h)                                                 | 0.9    | 1      | 2      |
| Recover time of failed tasks(s)                                                    | 120    | 80     | 300    |
| Cloud failure rate(%)                                                             | 2      | 1      | 3      |
| FRC and DDoS attack detection time(s)                                              | 82     | 62     | 44     |
| Time to identify attack source(s)                                                  | 100    | 120    | 160    |
| Detection False Positives: percent of legitimate traffic identified as attacks      | 1      | 3      | 5      |
| Detection False Negatives: percent of attacks not detected                         | 5      | 6      | 10     |
| Percent of attacks incorrectly classified                                         | 2      | 4      | 5      |
| Mean-cost of incident recovery($x1000$)                                           | 9      | 12     | 20     |
| Mean-cost of loss caused by incident($1000$)                                       | 3      | 5      | 7      |
| Application offline-time due to incidents(h)                                       | 2      | 3      | 5      |
| Financial cost of fault tolerance($x10000$)                                        | 3      | 4      | 5      |
| Data replication financial cost($x10000$)                                         | 40     | 30     | 40     |
| Processing overhead of attack detection mechanisms(s)                             | 20     | 30     | 25     |
| Application response time under attack(s)                                          | 5      | 4      | 10     |

\[ C^- = ((0, 1), (0, 1), \ldots, (0, 1)) \]

**Step 7:** The distance between different CSPs and the optimal (worst) solution and the solution was calculated with Eqs. (15)-(16).

\[
(D_1^+, D_2^+, D_3^+) = (0.076, 0.095, 0.211) \\
(D_1^-, D_2^-, D_3^-) = (0.193, 0.154, 0.023)
\]

**Step 8:** The relative closeness degree of CSPs was calculated with Eq. (17). The calculation result was as follows.

\[
(\rho_1, \rho_2, \rho_3) = (0.72, 0.62, 0.09)
\]

which would represent that three CSPs were ranked as \( \rho_1 > \rho_2 > \rho_3 \).

The result was consistent with the previous evaluation [31]. According to the evaluation metrics in Table 2, among the top five positive attributes, \( CSP_1 \) had four maximum values. Among the sixteen negative attributes, \( CSP_3 \) had ten maximum values. So it was clear that \( CSP_1 \) had the highest security, whereas \( CSP_3 \) had the lowest security. Thus, the accuracy of the proposed algorithm was verified, and the TCSSAL in the paper could help CUs to select the most satisfying CPSs according to their security requirements.
2) CASE 2: EVALUATION OF CSPs CONSIDERING CUs’ DEMANDS

QoS attributes of CSPs were obtained from five cloud IaaS providers: Amazon EC2 (CSP1), Windows azure (CSP2), Rackspace (CSP3), private cloud utilizing OpenStack (CSP4) and private cloud utilizing eucalyptus (CSP5). The evaluation metrics [32] shown in Table 3 involve three levels (second, first, and top) and CUs’ demands. The cloud service selection algorithm was converted to the lightweight intuitionistic fuzzy number matrix $B$ with Eqs. (3)-(5).

$$B = \begin{bmatrix}
(0.5, 0.5) & (0.83, 0.17) & (0.79, 0.21) \\
(1.0, 0.0) & (0.67, 0.33) & (1.0, 0.0) \\
(0.38, 0.62) & (0.83, 0.17) & (0.14, 0.86) \\
(1.0, 0.0) & (1.0, 0.0) & (0.06, 0.94) \\
(0.5, 0.5) & (0.67, 0.33) & (0.0, 1.0) \\
(0.0, 1.0) & (0.0, 1.0) & (0.06, 0.94)
\end{bmatrix}$$

was converted to the lightweight intuitionistic fuzzy number matrix $B$ with Eqs. (3)-(5).

$$C = \begin{bmatrix}
(0.16, 0.84) & (0.25, 0.75) & (0.29, 0.71) \\
(0.33, 0.67) & (0.20, 0.80) & (0.37, 0.63) \\
(0.12, 0.88) & (0.25, 0.75) & (0.05, 0.95) \\
(0.33, 0.67) & (0.30, 0.70) & (0.02, 0.98) \\
(0.16, 0.84) & (0.20, 0.80) & (0.0, 1.0) \\
(0.0, 1.0) & (0.0, 1.0) & (0.02, 0.98)
\end{bmatrix}$$

Step 4: The optimal value vector and the worst value vector were calculated with Eqs. (12)-(14).

$$C^+ = ((0.12, 0.88), (0.20, 0.80), (0.02, 0.98))$$

$$C^- = ((0.33, 0.67), (0.30, 0.70), (0.37, 0.63))$$

Step 5: The relative closeness degree of CSPs was calculated with Eq. (17).

$$(\rho_1, \rho_2, \rho_3, \rho_4, \rho_5) = (0.41, 0.20, 0.87, 0.60, 0.35)$$

The remaining calculation steps for “second-level” and all “first-level” attributes are similar to the calculation process of the “capacity” attribute. Some of their calculation results were shown in Tables 4 and 5, respectively.

Similarly, as for all top-level attributes, the calculation method was similar to that of “capacity” attribute. The relative closeness degree of five CSPs was

$$(\rho_1, \rho_2, \rho_3, \rho_4, \rho_5) = (0.54, 0.29, 0.59, 0.65, 0.60)$$

which indicated that five CSPs were ranked as

$$CSP_4 > CSP_5 > CSP_3 > CSP_1 > CSP_2$$

CSP4 was the most suitable cloud service provider.

The differences in the above two ranking results are interpreted as follows. In the previous study [32], in the calculation of the minimum distance, the normalization was not performed, thus enlarging the roles of some factors in decision-making. However, in this paper, different QoS attributes were normalized, and the measurement method based on LIFNs was designed to quantify them. So, the method presented in this work could be more quantitative and help CUs to choose the cloud service with the highest satisfaction.

B. EFFICIENCY OF TCSSAL

Due to the limited availability of the data for cloud service selection, the data generated randomly were used to simulate the process. Execution time of the proposed algorithm TCSSAL was compared with the algorithm of Cloud Service Selection utilizing AHP and TOPSIS (CSSAT) [33]. In CSSAT, the triangular fuzzy number was used to describe the experiences of experts and AHP was utilized to calculate the weights of QoS attributes. The method of fuzzy TOPSIS was used to rank CSPs.

In the simulation experiment, the above two algorithms were used to sort different numbers of CSPs. The experiment with a given number of CSPs was repeated 10 times and the average value was calculated as experimental time. As the number of CSPs increased, the time overhead of the two algorithms was shown in Figure 1. With the increase in the number of CSPs, the time cost required by TCSSAL was less than that of CSSAT. The reasons were given as follows. In CSSAT, AHP was used to calculate the weight coefficients of different attributes and the method based on the triangular fuzzy number was used to rank different CSPs. In TCSSAL, the calculated standard deviation was used to calculate the weight coefficient and the methods based on LIFNs was used to sort CSPs. Compared with CSSAT, the computational overhead of both methods in TCSSAL was less than that of CSSAT, so its time cost required was shorter than that of CSSAT.
It should be noted that for the cloud service selection involving many QoS attributes described with accurate values, TCSSAL could ensure the accuracy of the cloud service selection, and require less time cost than CSSAT. However, when the cloud service selection involved a lot of QoS attributes described with inaccurate values, the accuracy of CSSAT was higher than TCSSAL because the triangular fuzzy numbers used in CSSAT could describe the uncertainty of inaccurate attributes more accurately.

**TABLE 3. Evaluation metrics of QoS attributes for five CSPs.**

| Top level | First level | Second level | CSP1 | CSP2 | CSP3 | CSP4 | CSP5 | User |
|-----------|-------------|--------------|------|------|------|------|------|------|
|           |             |              | 4    | 8    | 4    | 5    | 5    | 4    |
| accountability |             |              | CPU  | 9.6  | 12.8 | 8.8  | 12.8 | 9.6  | 6.4  |
| agility    |             |              | Memory | 15   | 14   | 15   | 16   | 14   | 10   |
| capacity   |             |              | Disk | 1690 | 2040 | 630  | 500  | 400  | 500  |
| elasticity |             |              | Time | 80-120 | 520-780 | 20-200 | 120-180 | 120-180 | 60-120 |
| availability |             |              |      | 99.95% | 99.99% | 100% | 99.99% | 99.9% | 99.99% |
| assurance  |             |              | Upload time | 13.6 | 15   | 21   | 12.1 | 10.5 |
| Serviceability |       |              | CPU   | 17.9 | 16   | 23   | 8    | 6    |
| serviceability |           |              | Memory | 7    | 12   | 5    | 18   | 14   |
| serviceability |           |              | Free support | 0   | 1    | 1    | 1    | 1    |
| Cost       |             |              | VM Cost | 0.68 | 0.96 | 0.96 | 0.68 | 0.68 | <1   |
|            |             |              | Data inbound | 300  | 300  | 240  | 300  | 300  | 100  |
|            |             |              | Data outbound | 330  | 450  | 540  | 330  | 330  | 200  |
|            |             |              | storage | 1200 | 1500 | 1500 | 1200 | 1200 | 1000 |
| performance |             |              | Time range | 80-120 | 520-780 | 20-200 | 120-180 | 120-180 | 60-120 |
|            |             |              | Average time | 100  | 600  | 30   | 100  | 100  | 100  |
| security   |             |              | level (0-10) | 4    | 8    | 4    | 5    | 5    | 4    |

**TABLE 4. Second level.**

| Top level | First level | CSP1 | CSP2 | CSP3 | CSP4 | CSP5 |
|-----------|-------------|------|------|------|------|------|
| agility   |             | 0.406 | 0.197 | 0.869 | 0.604 | 0.354 |
| capacity  |             | 0    | 1    | 1    | 1    | 1    |
| elasticity|             | 0    | 0    | 1    | 0    | 0    |
| assurance |             | 0.513 | 0.566 | 0.427 | 0.584 | 0.523 |
| Serviceability |   | 1    | 1    | 1    | 1    | 1    |
| serviceability |   | 0    | 1    | 1    | 1    | 1    |

**TABLE 5. First level.**

| Top level | CSP1 | CSP2 | CSP3 | CSP4 | CSP5 |
|-----------|------|------|------|------|------|
| agility   | 0.187 | 0    | 0.534 | 0.331 | 0.143 |
| assurance | 0.307 | 0.928 | 0.534 | 1    | 0.764 |
| Cost      | 0.174 | 1    | 0.957 | 0.174 | 0    |

**FIGURE 1. The time overhead of TCSSAL and CSSAT.**

VI. CONCLUSION

This paper proposes a trusted cloud service selection algorithm to choose the most appropriate cloud service for CUs. In order to evaluate cloud services more accurately, LIFNs is used to measure different types of QoS attributes in a uniform way and the calculated standard deviation is designed to describe CUs’ preferences for different QoS attributes. In addition, TOPSIS theory is used to rank cloud services. In the experiments, the accuracy of the proposed algorithm was validated in two cases and its efficiency was compared with that of the conventional cloud service selection method.

The cloud service selection involves a lot of uncertainty evaluation criteria and requires the wide knowledge base. Therefore, it is necessary to collect experts’ decision-making experiences and design a hybrid method of weight calculation based on the consideration of CUs’ preferences and expert experiences. It is our main research work in the future. Besides, the dynamic measurement of CUs’ preferences will be integrated into the proposed algorithm.

REFERENCES

[1] S. Ghazouani and Y. Slimani, “A survey on cloud service description,” J. Netw. Comput. Appl., vol. 91, pp. 61–74, Aug. 2017.
[2] V. Hayyolalam and A. A. Pourhaji Kazem, “A systematic literature review on QoS-aware service composition and selection in cloud environment,” J. Netw. Comput. Appl., vol. 110, pp. 52–74, May 2018.
[3] F. A. M. Ibrahim and E. E. Hemayed, “Trusted cloud computing architectures for infrastructure as a service: Survey and systematic literature review,” Comput. Secur., vol. 82, pp. 196–226, May 2019.
[4] H. Ma, Z. Hu, K. Li, and H. Zhu, “Variation-aware cloud service selection via collaborative QoS prediction,” IEEE Trans. Services Comput., early access, Jan. 28, 2019, doi: 10.1109/TSC.2019.2895784.
O. Gireshna, N. Somu, and M. R. G. Raman, “WNN-EDAS: A wavelet neural network based multi-criteria decision-making approach for cloud service selection,” in *Computational Intelligence in Pattern Recognition*, Singapore: Springer, 2020, pp. 853–865.

M. S. Rahman, I. Khalil, A. Alabdulatif, and X. Yi, “Privacy preserving service selection using fully homomorphic encryption scheme on untrusted cloud service platform,” *Knowl.-Based Syst.*, vol. 180, pp. 104–115, Sep. 2019.

I. Patiniotakis, S. Rizou, and Y. Verginadis, “Managing imprecise criteria in cloud service ranking with a fuzzy multi-criteria decision making method,” in *Proc. Eur. Conf. Service-Oriented Cloud Comput.*, Berlin, Germany: Springer, 2013, pp. 34–48.

C.-C. Lo, D.-Y. Chen, C.-F. Tsai, and K.-M. Chao, “Service selection based on fuzzy TOPSIS method,” in *Proc. IEEE 24th Int. Conf. Adv. Inf. Netw. Appl. Workshops*, 2010, pp. 367–372.

A. Ouadah, A. Hadjali, F. Nader, and K. Benouaret, “SEFAP: An efficient approach for ranking skyline Web services,” *J. Ambient Intell. Hum. Comput.*, vol. 10, no. 2, pp. 709–725, Feb. 2019.

G. Bulyukozkan, O. Feyzioglu, and C. A. Havle, “Intuitionistic fuzzy AHP based strategic analysis of service quality in digital hospitality industry,” *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 1687–1692, 2019.

R. Srivastava and A. K. Daniel, “Efficient model of cloud trustworthiness in cloud service ranking with a fuzzy multi-criteria decision making method,” in *Proc. Eur. Conf. Service-Oriented Cloud Comput.*, Berlin, Germany: Springer, 2013, pp. 34–48.

S. Ding, C. Xia, C. Wang, D. Wu, and Y. Zhang, “Multi-objective optimization based ranking prediction for cloud service recommendation,” *Decis. Support Syst.*, vol. 101, pp. 106–114, Sep. 2017.

L. Liu, S. Gu, and D. Fu, “A new multi-objective evolutionary algorithm for inter-cloud service composition,” in *Proc. TIIIS*, 2018, vol. 12, no. 1, pp. 1–20.

D. Varapravath Kunhambu and R. V. Siva Balan, “Efficient multi-objective particle swarm optimisation based ranking system for cloud service selection,” *IET Commun.*, vol. 13, no. 3, pp. 297–304, Feb. 2019.

C. Zhang, J. Ning, J. Wu, and B. Zhang, “A multi-objective optimization method for service composition problem with sharing property,” *Swarm Evol. Comput.*, vol. 49, pp. 266–276, Sep. 2019.

S. Ding, C. Xia, C. Wang, D. Wu, and Y. Zhang, “Multi-objective optimization based ranking prediction for cloud service recommendation,” *Decis. Support Syst.*, vol. 101, pp. 106–114, Sep. 2017.

L. Liu, S. Gu, and D. Fu, “A new multi-objective evolutionary algorithm for inter-cloud service composition,” in *Proc. TIIIS*, 2018, vol. 12, no. 1, pp. 1–20.

C. L. Hwang and K. Yoon, “Methods for multiple attribute decision making,” in *Multiple Attribute Decision Making*, Berlin, Germany: Springer, 1981, p. 191.

P. Hajek and W. Froelich, “Integrating TOPSIS with interval-valued intuitionistic fuzzy cognitive maps for effective group decision making,” *Inf. Sci.*, vol. 485, pp. 394–412, Jun. 2019.

M. M. Salih, B. B. Zaidan, A. A. Zaidan, and M. A. Ahmed, “Survey on fuzzy TOPSIS state-of-the-art between 2007 and 2017,” *Comput. Oper. Res.*, vol. 104, pp. 207–227, Apr. 2019.

Carnegie Mellon. *The Cloud Service Measurement Initiative Consortium, Service Measurement Index (SMI)*, Accessed: Jul. 5, 2019. [Online]. Available: http://www.cloudcommons.com/about-smi/

T. Halabi and M. Bellaiche, “Towards quantification and evaluation of security of cloud service providers,” *J. Inf. Secur. Appl.*, vol. 33, pp. 55–65, Apr. 2017.

N. Ouaz, and K. Kirthivasan, “A computational model for ranking cloud service providers using hypergraph based techniques,” *Future Gener. Comput. Syst.*, vol. 68, pp. 14–30, Mar. 2017.

R. R. Kumar, S. Mishra, and C. Kumar, “Prioritizing the solution of cloud service selection using integrated MCDM methods under fuzzy environment,” *J. Supercomput.*, vol. 73, no. 11, pp. 4652–4682, Nov. 2017.

**YULI YANG** was born in Yicheng, Shanxi, China, in 1979. She received the M.S. degree in computer science and technology from Guangxi Normal University, China, in 2007, and the Ph.D. degree in computer science and technology from the Taiyuan University of Technology, Taiyuan, China, in 2015. She is currently a Lecturer with the College of Computer Science and Technology, Taiyuan University of Technology. Her research interests are related with computer network security, cloud computing, and trust management.

**NANYUE YU** was born in Yongji, Shanxi, China, in 1995. She received the B.S. degree in computer science and technology from the Taiyuan University of Technology, China. Her research interest is cloud computing.

**YONGLING CHEN** (Member, IEEE) was born in Weifang, Shandong, China, in 1983. He received the B.S. degree in computer science from Jilin University, in 2007, and the M.S. degree in computer science from the Institute of Software, Chinese Academy of Science, in 2009, and the Ph.D. degree in computer science from the University of Chinese Academy of Sciences, in 2013. From 2013 to 2015, he was an Assistant Professor with the College of Computer Science and Technology, Taiyuan University of Technology, Taiyuan, China. Since 2015, he has been an Associate Professor. His research interests are wireless sensor networks, indoor positioning, and the IoT security.