Web Service Composition Method Based on FAHP and TOPSIS

Zhanlin Yu, Longchang Zhang, Chengwen Zhang

Abstract
Since open and dynamic Internet environment increased the uncertainty of decision-making, the composition service selection based on quality of service (QoS) is still challenging. Besides, user’s weight expressed by fuzzy pair-wise comparison matrix is closer to user’s habits. To solve the difficulties above, Web service composition method based on FAHP and TOPSIS (WSCM_FT) is presented to support the hybrid QoS model expressed by real numbers, interval numbers, triangular fuzzy numbers and intuitionistic fuzzy numbers and user’s weight model expressed by fuzzy pair-wise comparison matrix. WSCM_FT includes five main steps: calculating user’s weight, converting hybrid QoS into intervals, weighting normalized decision-matrix, determining the positive and negative ideal solution, and calculating the close-degrees of candidates. Finally, some experiments are given using actual QoS data to demonstrate the benefits and effectiveness of our approach.

Keywords: FAHP, TOPSIS, Web Service, Service Composition, Quality Of Service (QoS)

1. Introduction
In distributed environment, Web services have become an important part of commercial application. However, most of atomic services can not satisfy the complex user requirements [1]. Therefore, service providers expect their services can be integrated into value-added composite services autonomously. With the gradual increasing in the number of service providers and service consumers, there will be a large number of Web services with the same function and different quality of service (QoS). According to user’s QoS requirements, it is one of important tasks [1] in service composition process to select atomic services or composition services from the Web services with same function and different quality of service (QoS). In Web service composition community, Zeng et al. [2] presented the issue about QoS-driven service selection, which could be solved by the method of multi-objective decision making [1, 3-6]. Longchang Zhang et al. [3] proposed a dynamic Web service composition algorithm which supports multi-period QoS information and hybrid QoS models expressed by real numbers (R), interval numbers (IR), triangular fuzzy numbers (TFNs) and intuitionistic fuzzy numbers (IFNs). Based on our previous works [3], the following problem is taken into account. Currently, the user weights are based on the assumption that user can express his weight with an explicit number or do not take into user’s weight account. This assumption can no longer satisfy with the requirement of users since it is difficult to provide the explicit and complete weight on QoS attributes some time for user, so the expressions of users’ weights are closer to users’ habit should be considered.

To solve difficulties above, this paper presents a user weight model expressed by fuzzy pair-wise comparison matrix, a novel Web service composition method based on FAHP and TOPSIS (WSCM_FT). WSCM_FT takes hybrid QoS expressed by real numbers, interval numbers, triangular fuzzy numbers and intuitionistic fuzzy numbers and user weight with fuzzy pair-wise comparison matrix into account synthetically and it includes five main steps. First, we introduce a method to calculate user weight for fuzzy pair-wise comparison matrix. Second, we introduce a data conversion method to convert hybrid QoS values into interval numbers. Thirdly, the way of calculating weighted normalized decision-matrix is discussed. Then, the method of
determining the positive-ideal and negative-ideal solution is introduced. Finally, we present a fineness function to evaluate and rank alternatives.

The remainder of this paper is organized as follows: Section II summarizes the related work in this field. Section III introduces user weight model and hybrid QoS model of Web service. WSCM_FT is presented in Section IV. A series of experiments is proposed in Section V to show the effectiveness and benefits of our approach. Finally, Section VI concludes the paper and outlines our future work.

2. Related Work

In the Web service composition research, Li Zhen et al. [1] proposed a new QoS ontology for describing the hybrid QoS data and the QoS aggregation method, but it cannot describe QoS information with intuitionistic fuzzy numbers. Intuitionistic fuzzy set has been proven to be highly useful to deal with uncertainty and vagueness [7, 8] and applied to QoS-based service selection [6]. Longchang Zhang [3] proposed a hybrid QoS model expressed by real numbers, interval numbers, triangular numbers and intuitionistic fuzzy numbers. Tran VX et al. [4] presented a more comprehensive and detailed QoS ontology (WS-QoSOnto) which involves QoS role, QoS description, QoS level and QoS group concepts, but did not give specific definition of QoS criteria and aggregation methods. A user-driven QoS model was proposed in Ref. [9] and The Analytic Hierarchy Process (AHP) and the Brown–Gibson (BG) methods were adapted to facilitate quality assessment, but this real-based QoS model could not describe the QoS attributes with strong uncertainty. Ping Wang presented a QoS model expressed by An intuitionistic fuzzy set [6]. Obviously all of the QoS attributes are described by intuitionistic fuzzy numbers is unreasonable and reduce the accuracy of decision-making algorithm. A QoS model based on random numbers was presented in Ref. [10]. However, it is difficult to find a random function described QoS change accurately in extremely uncertain environment and the establishment of random function needs lots of data samples. And it is unreasonable not to take into account the user's individual QoS requirements. In addition, all QoS models above did not consider the expression habit of user weight. Based on the hybrid QoS model expressed by real numbers, interval numbers, triangular numbers and intuitionistic fuzzy numbers in Ref. [3], this paper presents a user weight model expressed by fuzzy pair-wise comparison matrix, which is closer to user's expression habit.


However, the algorithms above cannot evaluate user weight expressed by fuzzy pair-wise comparison matrix and hybrid QoS models expressed by real numbers, interval numbers, triangular fuzzy numbers and intuitionistic fuzzy numbers. To solve difficulties above, WSCM_FT is proposed in the first time.
3. User-centric Web Service QoS Model

User-centric service selection must take user’s QoS weight and the QoS of Web service into account; therefore, we will introduce the user weight model expressed by fuzzy pair-wise comparison matrix and hybrid QoS model expressed by real numbers, interval numbers, triangular fuzzy numbers and intuitionistic fuzzy numbers in this section.

3.1. User weight model

As it is difficult to accurately express their QoS weight for users, fuzzy pair-wise comparison matrix closed to users’ habit is presented in this section. According to the express habit of users, five relationship symbols are defined to compare the relationship of two QoS attributes. Fig.1 describes the relationship symbol and an instance matrix of user weight. Price is moderately more important than availability, strongly more important than response time, very strongly more important than reliability, extremely more important than reputation in the first line of the instance matrix.

\[
\begin{align*}
\text{Equally important} & : Pr = Av = RT = Ra = Rp = \\
\text{Moderately more important} & : Pr > Av \geq RT \geq Ra \geq Rp \\
\text{Strongly more important} & : RT = Av \geq Pr \geq Ra \\
\text{Very strongly more important} & : Ra = RT \geq Pr \geq Av \\
\text{Extremely more important} & : Rp = Ra = RT = Pr
\end{align*}
\]

Figure.1 QoS attribute relationship symbols and instance matrix

3.2. Quality of service for Web services

In order to distinguish the pros and cons of candidate plans with the similar functionality and different quality of service for user, it is necessary to establish user weight model and QoS model. Different service providers and participants may use different QoS concepts for describing service quality information [3-6]. It is reasonable to use language to express reliability and reputation which are gotten from the feedback of users [1]. Ortiz G et al. [5] classified QoS criteria into three main categories: Execution Criteria, Service Provider Criteria, and Exception-related Criteria. To facilitate the discussion, the QoS model in this paper includes 5 criteria: price, availability, response time, reliability, reputation, etc. The definitions of five criteria refer to the definition in ref. [3]. Price is expressed by real numbers; availability is expressed by interval numbers; response time is expressed by interval numbers; reliability is expressed by triangular fuzzy numbers, the language set is defined as very high, high, medium, low, and very low and be interpreted as triangular fuzzy set ([8,10,10],[5,7,9],[3,5,6],[1,3,4],[0,0,2]); reputation is expressed by intuitionistic fuzzy numbers, the language set is defined as (very high (VH), high (H), medium (M), low (L), and very low (VL)) to describe reputation and be interpreted as intuitionistic fuzzy set (<0.9,0.1- π , π >, <0.7,0.3- π , π >, <0.5,0.5- π , π >, <0.3,0.7- π , π >, <0.1,0.9- π , π >), and we utilize \( L(\pi) \) to represent the score given by user. In the 5 criteria of QoS, price and response time are cost criteria and others are benefit criteria. In addition, upper and lower limits of interval numbers are non-negative real numbers.

3.3. QoS aggregation

In Web services composition, the general services composition flow model are composed of several task nodes, where each node represents a group of services. The composition service QoS can be calculated by atomic service QoS. The methods of composition service QoS
aggregation refer to the aggregation methods in ref. [3] and the QoS calculation formulas of sequence and parallel are listed in Table 1.

<table>
<thead>
<tr>
<th>QoS</th>
<th>Price</th>
<th>Availability</th>
<th>Response Time</th>
<th>Reliability</th>
<th>Reputation</th>
</tr>
</thead>
<tbody>
<tr>
<td>(sequence)</td>
<td>$P(w_c) = \sum_{i=1}^{n} P_i$, $A(w_c) = \prod_{i=1}^{n} A_i$, $T(w_c) = \sum_{i=1}^{n} T_i$</td>
<td>$R(w_c) = \frac{1}{n} \sum_{i=1}^{n} r_i$</td>
<td>$C(w_c) = \frac{1}{n} \sum_{i=1}^{n} c_i$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(parallel)</td>
<td>$P(w_c) = \sum_{i=1}^{n} P_i$, $A(w_c) = \frac{1}{n} \sum_{i=1}^{n} A_i$, $T(w_c) = \max_{i=1}^{n}(T_i)$</td>
<td>$R(w_c) = \frac{1}{n} \sum_{i=1}^{n} r_i$</td>
<td>$C(w_c) = \frac{1}{n} \sum_{i=1}^{n} c_i$</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 4. WSCM_FT

The hybrid QoS model supports real numbers, interval numbers, triangular fuzzy numbers and intuitionistic fuzzy numbers. And the user weight model expressed by fuzzy pair-wise comparison matrix is closer users’ habit. Unfortunately, there is not any appropriate approach for service composition with hybrid data types and fuzzy pair-wise comparison matrix. Therefore, WSCM_FT is proposed based on FAHP and TOPSIS in this section. WSCM_FT utilizes FAHP to calculate user weight and TOPSIS [17] to select optimal composition plan. The input parameters of WSCM_FT are fuzzy pair-wise comparison matrix and multi-attribute hybrid QoS decision matrix. We construct the hybrid QoS decision matrix of alternative composition plans firstly. Let $p = \{p_1, p_2, \cdots, p_n\}$ be a discrete set of candidate composition plans, $q = \{q_1, q_2, \cdots, q_n\}$ be a set of aggregated QoS attributes, and $D=(d_{ij})_{n \times n}$ be the hybrid decision matrix, where $d_{ij}$ is an attribute value of $q_j$ at $p_i$ ($i=1, 2, \cdots, m; j=1, 2, \cdots, n$). And then, let $Q=(q_{ij})_{n \times n}$ be the fuzzy pair-wise comparison matrix (Fig.1). The details of WSCM_FT are described as follows.

#### 4.1. Calculating user weight

Step1: Establish calculating matrix $\overline{Q}=(\overline{q}_{ij})_{n \times n}$ from $Q=(q_{ij})_{n \times n}$, $\overline{q}_{ij}$ is calculated as the conversion relationship in table 2.

<table>
<thead>
<tr>
<th>Triangular fuzzy number</th>
<th>Relationship</th>
<th>Description</th>
<th>Conversion relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1, 3]</td>
<td>$\approx$</td>
<td>Equally important</td>
<td>$a_{ij}=[u_j, m_j, l_j]$, $a_i=1/a_{ij}=[1/u_j, 1/m_j, 1/l_j]$, $a_0=1$</td>
</tr>
<tr>
<td>[3, 4]</td>
<td>$&gt;$</td>
<td>Moderately more important</td>
<td></td>
</tr>
<tr>
<td>[4, 5]</td>
<td>$\cong$</td>
<td>Strongly more important</td>
<td></td>
</tr>
<tr>
<td>[6, 7]</td>
<td>$\div$</td>
<td>Very strongly more important</td>
<td></td>
</tr>
<tr>
<td>[8, 9, 10]</td>
<td>$\succ$</td>
<td>Extremely more important</td>
<td></td>
</tr>
<tr>
<td>[1, 2, 3][3, 4, 5][5, 6, 7][7, 8]</td>
<td>$\approx$</td>
<td>Intermediate values</td>
<td>$\overline{q}_{ij}$ can be get from upper triangular.</td>
</tr>
</tbody>
</table>

From, we can get the following features:

1) For $\forall i, j \in N$, $a_{ij}=[u_j, m_j, l_j]$, $a_{ij}=1/a_{ij}=[1/u_j, 1/m_j, 1/l_j]$, $a_0=1$; 2) The lower triangular of $\overline{Q}$ can be obtained from the upper triangular.

Step 2: Defuzzification

For each triangular fuzzy numbers, its center of gravity (geometric center) is usually a good representation of a fuzzy number [18]. The triangular fuzzy numbers can be defuzzified based on the following formula.
\[ R = \frac{\int_{-\infty}^{\infty} x \mu(x) \, dx}{\int_{-\infty}^{\infty} \mu(x) \, dx} = \frac{\int_{-\infty}^{\infty} x \mu(x) \, dx + \int_{-\infty}^{\infty} x \mu(x) \, dx}{\int_{-\infty}^{\infty} \mu(x) \, dx + \int_{-\infty}^{\infty} \mu(x) \, dx} \] , where \( \mu(x) \) denotes the grade of membership of \( R \) and \( S \) denotes the relative interval of integration. After the defuzzification process, \( \vec{Q} = (\vec{q}_i)_{m \times n} \) is converted to \( \hat{Q} = (\hat{q}_i)_{m \times n} \) in which there are only real numbers.

Definition 1: if \( \forall i, j, k \in N, \ a_{ij} \times a_{jk} = a_{ik} \), then \( \hat{Q} \) strictly related to the consistency of the pair-wise comparison judgments.

According to the characteristics 2, the construction of user weight only considers the upper triangular or lower triangular of pair-wise comparison matrix.

Step3: Calculating weight vector

The count of QoS attributes is greater than 3 [4, 5], so the order of matrix determined by the count of QoS attributes is greater than 3. In order to reduce the complexity of calculation, we make use of the method in Ref. [7]. User's weight vector is calculated as follows from matrix \( \hat{Q} \).

Let \( \omega = (\omega_1, \omega_2, \ldots, \omega_n) \) be user's weight, for \( \forall i \in N, \ \omega_i = \omega_i^* \sum_{i=1}^{n} \omega_i^* \), while \( \omega_i^* = \sqrt{\prod_{j=1}^{n} a_{ij}} \). (1)

Let \( \lambda_{\text{max}} \) be the largest eigenvalue of \( \hat{Q} \), then \( \lambda_{\text{max}} = \sum_{i=1}^{n} \omega_i s_i \), while \( s_i = \sum_{j=1}^{n} a_{ij} \). (2)

Step3: Consistency verification

The errors any two attributes relation may lead to user weight value have large deviation between calculated value and actual value, so the unreasonable result in service selection will be obtain. We use the verification method of Saaty [7] to verify the consistency of matrix \( \hat{Q} \).

Let the consistency index (CI) is \( C.I. = (\lambda_{\text{max}} - n) / (n - 1) \). The final consistency ratio (CR), usage of which to judge whether the evaluations are sufficiently consistent, is calculated as the ratio of the CI and the random index (RI), as indicated

\[ C.R. = C.I. / R.I. \]  

We can get \( \lambda_{\text{max}} \leq 5.45 \) and \( R.I. = 1.12 \), from consistency index value table [7] while \( n = 5 \). The number 0.1 is the accepted upper limit for CR. If the final consistency ratio exceeds this value, the evaluation procedure has to be repeated to improve consistency. The measurement of consistency can be used to evaluate the consistency of fuzzy pair-wise comparison matrix.

4.2. Converting hybrid QoS into intervals

In this section, we introduce some methods to convert real numbers, interval numbers, triangular fuzzy numbers and intuition fuzzy numbers into interval numbers.

Converting R into IR: let \( r \in R \), then \( \hat{r} = [r, r] \). (4)

Converting TFNs into IR: Hepu Deng [18] summarized the methods of defuzzification which covert triangular fuzzy numbers into real numbers. However, it is more reasonable to convert triangular fuzzy numbers into interval numbers. For example, the value is 1 by fuzzy integral when the language phrase is low \([0, 0, 2]\). However, the value should change between 0 and 1 for the actual situation. When the value is 0, the probability of TFNs accord with low is 1. And
when the value is 1, the probability of TFNs accord with low is 0.5. Therefore, we convert the triangular fuzzy numbers into interval numbers. Let $\tilde{t} = [a'_l, a''_l, a'_u] \in \text{TFNs}$, then
$$\tilde{t} = \left[ a'_l + a''_l / 2, a''_l + a'_u / 2 \right].$$

Converting IFNs into IR: Atanassov found a method of converting IFNs into fuzzy number [8] named Atanassov operator. Atanassov operator $p \in [0,1]$ is defined, and then $\tilde{A} = u_A(x) + p \cdot \pi_A(x)$ is a fuzzy numbers, where $A = \{u_A(x), v_A(x), \pi_A(x)\}$ and we can get
$$\tilde{A} = [u_A(x), u_A(x) + \pi_A(x)].$$

Then, we can get the interval decision matrix $\tilde{D} = (\tilde{d}_{ij})_{m \times n}$ from $D = (d_{ij})_{m \times n}$ by Eq. (4), Eq. (5), and Eq. (6).

4.3. Weighting normalized decision-matrix

In order to eliminate the impact of different physical dimension, we utilize vector normalization to normalize the interval decision matrix. Let $\tilde{D} = (\tilde{d}_{ij})_{m \times n}$ be the normalized decision-making matrix from $D$. The formula of normalizing interval as follows [20]:
$$\begin{align*}
\tilde{d}^+_i &= \frac{d^+_i}{\sqrt{\sum_{j=1}^{m} (d^+_j)^2 + (d^-_j)^2}} \quad (i = 1, \ldots, m) \\
\tilde{d}^-_j &= \frac{d^-_j}{\sqrt{\sum_{i=1}^{n} (d^+_i)^2 + (d^-_i)^2}} \quad (j = 1, \ldots, n)
\end{align*}$$

After normalizing decision matrix, it is necessary to give the weighted normalized decision matrix based on user’s weight. Let $\omega = (\omega_1, \omega_2, \ldots, \omega_n)$ be user’s weight vector from matrix $Q$ by Eq. (1) and Eq. (2). Then we can get the weighted normalized decision matrix $\tilde{D} = (\tilde{d}_{ij})_{m \times n}$, where $\tilde{d}_{ij} = \omega_j \times \tilde{d}_{ij}$. By Interval multiplication [20], we have $\tilde{d}_{ij} = \omega_j \times \tilde{d}_{ij} = \left[ \omega_j \times \tilde{d}^+_i, \omega_j \times \tilde{d}^-_j \right]$. Then, we can transform the number into the number with the same dimension.

4.4. Determining the positive and negative ideal solution

In order to compare the pros and cons of the candidate plans in weighted normalized decision-making matrix $\tilde{D}$, we define the positive ideal solution and the negative ideal solution in $\tilde{D}$.

**Definition 2** Positive ideal solution ($\tilde{s}^+$) and negative ideal solution ($\tilde{s}^-$).

Positive ideal solution: $\tilde{s}^+ = (\tilde{s}^+_1, \tilde{s}^+_2, \ldots, \tilde{s}^+_n)$:
$$\begin{align*}
\tilde{s}^+_j &= \left[ \max(\tilde{d}^+_j), \max(\tilde{d}^-_j) \right] \quad (j = 1, \ldots, n) \\
\tilde{s}^-_j &= \left[ \min(\tilde{d}^+_j), \min(\tilde{d}^-_j) \right] \quad (j = 1, \ldots, n)
\end{align*}$$
Negative ideal solution: 
\( \tilde{s}^- = (\tilde{s}_{\text{min}}, \tilde{s}_{\text{min}}, \ldots, \tilde{s}_{\text{min}}) \)  
\( \tilde{s}^+ = (\tilde{s}_{\text{max}}, \tilde{s}_{\text{max}}, \ldots, \tilde{s}_{\text{max}}) \)  

\[
\hat{s}_{ji} = \begin{cases} 
\min_{\text{Benefit}}(\hat{d}_{ji}^b), & \text{Characteristic=Benefit} \\
\max_{\text{Cost}}(\hat{d}_{ji}^c), & \text{Characteristic=Cost} 
\end{cases} 
\]

Obviously, \( \tilde{s}^+ \) denotes the ideal plan and \( \tilde{s}^- \) denotes the imaginary worst plan.

4.5. Calculating the close-degrees of candidates

We utilize the common method-Euclidean distance to measure the interval numbers’ distance.

Let \( d(\hat{d}, \tilde{s}^+) \) and \( d(\hat{d}, \tilde{s}^-) \) be the distance from ith plan to positive and negative ideal solutions, then

\[
d(\hat{d}, \tilde{s}^+) = \frac{\sum_{j=1}^{n} \left( \left( \hat{d}_{ji}^b - \tilde{s}_{ji}^{b+} \right)^2 + \left( \hat{d}_{ji}^c - \tilde{s}_{ji}^{c+} \right)^2 \right) }{2n} \\
d(\hat{d}, \tilde{s}^-) = \frac{\sum_{j=1}^{n} \left( \left( \hat{d}_{ji}^b - \tilde{s}_{ji}^{b-} \right)^2 + \left( \hat{d}_{ji}^c - \tilde{s}_{ji}^{c-} \right)^2 \right) }{2n}
\]

(10).

Here, we introduce a close-degree (fineness) function to evaluate ith plan (shown in Eq. (11)).

\[
f(\hat{d}, \tilde{s}^+, \tilde{s}^-) = d(\hat{d}, \tilde{s}^+)/d(\hat{d}, \tilde{s}^-) + d(\hat{d}, \tilde{s}^+) (11),
\]

where the function takes QoS data of the ith plan and two ideal solutions as its inputs, and \( d() \) is the distance function of interval numbers, and the calculation method refers to Eq.(10). \( f(\hat{d}, \tilde{s}^+, \tilde{s}^-) \) calculates the ith plan’s close-degree. Obviously, the close-degree is bigger the candidate plan is better. Then we can sort alternatives via their close-degrees and obtain the final optimal decision making result by formula \( \max(f(\hat{d}, \tilde{s}^+, \tilde{s}^-)) \).

5. Experimental Evaluation

There are three experiments in this section. First, we compare the WSCM_FT and UMC [1] algorithm not considering user weight model. Then we compare the WSCM_FT and UMC algorithm considering user weight model. Finally, we analyze the time complexity of WSCM_FT.

5.1. Comparison of WSCM_FT and UMC not considering user weight model

In the algorithm of hybrid QoS model, UMC [1] supports real numbers, interval numbers and triangular fuzzy numbers, so we compare WSCM_FT and UMC. Through actual QoS data (the composition plans shown in Table 3), we will demonstrate that WSCM_FT is better than UMC in this section.
UMC does not consider the reputation (let value of reputation be 0) because it only supports real numbers, interval numbers, and triangular fuzzy numbers. Let \( \omega = (0.36, 0.21, 0.23, 0.05, 0.15) \) be user’s weight vector.

To show the benefits of our approach, we introduce a criterion to compute the difference of synthetically qualities of two plans. Let \( p_i \) be the optimal plan selected by WSCA_AT and \( p_j \) be the optimal plan selected by UMC, then \( \text{dif}(p_i, p_j) = f(p_i, s^-_i, s^+_i) - f(p_j, s^-_j, s^+_j) \) (13); if \( \text{dif}(p_i, p_j) > 0 \) then \( p_i \) is better than \( p_j \), else \( p_i \) is worse than \( p_j \).

Fig.2 describes the results of two algorithms. The optimal composition plan is 9 gotten by WSCM_FT; and the optimal plan is 4 obtained by UMC. We can get \( 9 - 4 = 0.768 - 0.701 = 0.067 \) by Eq. (13). Therefore, WSCM_FT is better than UMC.

5.2. Comparison of WSCM_FT and UMC considering user weight model

In this experiment, we compare WSCM_FT and UMC considering user model. Service composition candidates set is shown in Table 3. Let the value of reputation of candidates be 0 because UMC does not support intuitionistic fuzzy numbers. User weight model is shown in Fig.1.

We can get user weight vector \( \omega = (0.51, 0.2638, 0.1296, 0.0636, 0.0329) \) from instance matrix in Fig.1 by Eq. (1), \( \lambda_{\max} = 5.243 < 5.45 \) by Eq.(2), \( C.R. = 0.0542 < 0.1 \) by Eq.(3). UMC does not support user weight described by pair-wise comparison matrix, so let \( \omega = (0.2, 0.2, 0.2, 0.2, 0.2) \) be user weight vector of UMC.

Fig.3 describes the results of two algorithms. The optimal composition plan is 4 gotten by WSCM_FT; and the optimal plan is 12 obtained by UMC. We can get \( 4 - 12 = 0.8056 - 0.6653 = 0.1403 \) by Eq. (13). Therefore, WSCM_FT is better than UMC.
5.3. Time complexity of WSCM_FT

We analyze the time complexity of WSCM_AT on one PC with Intel Core 2 2.0 GHz CPU, 2 GB memory. The fuzzy pair-wise comparison matrix instance is shown in Fig.1, the number of composition plans changes from 10 to 50. We get the average time of 100 times execution time of WSCM_FT. The result of experiment demonstrates WSCM_FT has linear or polynomial time complexity in Fig.4, so it is an effective and fast algorithm.

![Figure 4. Time complexity of WSCM_FT](image)

6. Conclusion and Future Work

There is not any appropriate approach for service composition with fuzzy pair-wise comparison matrix. Based on previous studies, this paper presents a user weight model expressed by pair-wise comparison matrix which is closer to user’s habit. Furthermore, this paper proposes WSCM_FT to evaluate hybrid and calculate user weight vector from fuzzy pair-wise comparison matrix. Finally, the experimental results show the advantages and effectiveness of the WSCM_FT.

In the future, our plan is to improve WSCM_AT performance because lots of atomic services and tasks of composition service can generate a lot of composition plans will affect the user experience. In addition, the user’s context information will be taken into account in service composition process.

7. Acknowledgment

The authors are very grateful to the anonymous referees for their insightful and constructive comments and suggestions that have led to an improved version of this paper. This work is supported by the National Natural Science Foundation No. 60672121; Humanity and Social Science Youth foundation of Ministry of Education of China No. 12YJC8700302009T045.

8. References


