Journal of Theoretical and Applied Information Technology

www.jatit.org

# A FUZZY-BASED USER-CENTRIC APPROACH FOR SELECTING THE OPTIMAL COMPOSITION OF SERVICES

[1]MAHDI BAKHSHI, [2]ABBAS OLFAT, [3]GHASEM OLFAT, [4]FARHAD MARDUKHI
[1]Master Student, Department of Computer, Islamic Azad University of Najaf Abad, Iran
[2,3] Software Engineer, Persia Soft Co, Kermanshah, Iran
[4]PhD Student, Faculty of Computer, University of Isfahan, Iran

**ABSTRACT**

Service-Oriented Applications are being regarded as the main practical solution for distributed environments. In such systems, though each service is able to response the user request independently, it is essential to compose them for supplying a compound value-added service enable us to address the complex requests. Since, there may be a number of compositions to providing the requested service, it is so important to find one whose properties are close to user's desires. In this paper, a new approach for evaluating the service compositions is presented which attempts to obtain the user desires. This approach uses fuzzy logic to infer on the basis of quality measures ranked by user.
.

**Keywords:** *Web service, service composition, Quality of Service (QoS), user preferences, fuzzy logic.*

## 1. INTRODUCTION

Service composition has been a main problem in service based environment during recent years, and still is being concentrated by many researchers. Service composition means how several simple services get together to establish a new compound service with high value. Up to now, the diverse techniques have been presented based on different points of view for performing service composition [2,3,4,5]. There are different services which have the same functionality and can replace one another. These services are of course different with regard to quality factors such as response time, availability, throughput, security, reliability, execution cost and etc. Therefore web service composition problem leads to quality engineering problem, because these services should be chosen in such a way that the best QoS is prepared for the total composition. On the other hand the quality factors reflect the need of users and their satisfaction.

In this paper, we are going to find a way for selecting the optimal composition among feasible different compositions, according to quality properties of services by establishing a fuzzy system. Fuzzy system is a proper method to express the user desires at a way which is understandable for both human and machine. By moving toward the age of information, a hypothesis can formulate the human knowledge in the systematic form, and introduce an approximate description that is reliable and analyzable. This important subject is applicable by a fuzzy system [1].

The rest of this paper is organized as follows. Section 2 reviews the related works and then in section 3 the definitions of the services, their quality properties and declaration of problem are stated. In section 4, after showing the user preferences using quality driven fuzzy logic, we try to design a system for ranking the composite services according to user preferences based on fuzzy logic. In section 5, we consider the system implementation and at the end, in section 6, we evaluate the work and reason about.

## 2. COMPOSITION OF SERVICES

The constitutive unit of service oriented systems is service. These services can be combined and produce one service with added value. A composite service is an umbrella structure aggregating multiple other elementary and composite web services, which interact with each other according to a process model [3]. The composition of services caused by reaching to a predetermined objective, that don't become certain by elementary services. Two determinant factors in composition of services are qualitative and functional properties of services.

One of the service properties is its functional. Because of comprehensive using of services, some of them are designed in a manner that their functional properties can solve simple problems for atomic tasks. For solving more complex problems, the software developers select a suitable composition to solve the problem by one by one execution, at special sequence. This sequence is distinct on executive plan related to distinctive problem. The presented services for one composition should be functionality coordinated and able to be composed.

Other property of service is its quality. Quality of service takes into consideration by functional and nonfunctional requirements. For example, functional requirement in one service can be founded as suitable output, while the fast finding them is as nonfunctional requirements. The user gives score to comply with every requirement based on its criteria. For example, the user is ready to wait more for finding a more suitable output. In fact, quality criteria of each service can be expressed by functional attributes such as response time and cost, and nonfunctional attributes such as reliability and availability [6]. It is necessary to mention, to compare quality of services, QoS expresses quality as quantity. The selection of one optimal composite service is related to quality criteria of the services that are composed together. In fact, the selection of services should be in a manner that the composite service satisfy the user requirements.

### 2.1. Execution Plans

Today, different models and languages are used to describe the composition from services. State chart is one of the most common of these methods to express service compositions.

A simplified state chart which specifies a "Travel Planner" composite web service is depicted in Figure 1. In this composite service, a search for attractions is performed in parallel with a flight accommodation booking. When searching and booking operations are completed, the distance from the hotel to the accommodation is computed, and a car or a bike rental service is invoked [3].

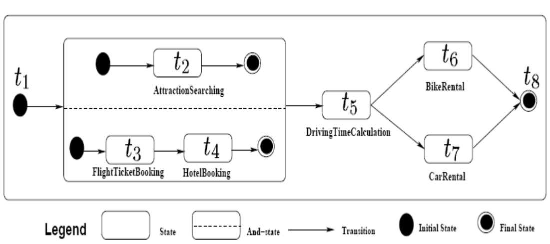



*Figure 1. State chart of a composite service "Travel Planner"*

In this state chart, operations and their sequence are observable. These operations and their sequences introduce an execution path for one composite service. For example, in this state chart, one execution path is $<t_2,t_3,t_4,t_5,t_6>$. For doing every task, there may be several candidate services that have the same functionalities but have different qualitative properties.

The selection of one service between the members of one community for performing every task, during the execution of one composite service, cause to form an execute plan for the performing of an execution path. For example, for a distinctive execution path an execution plan can be as $p=\{<t_2,s_{16}>,<t_3, s_{23}>,<t_4,s_{28}>,<t_5,s_{36}>,<t_6,s_{43}>\}$. It's necessary that because of being various execution paths and possibility of choosing various choices among services of a community, there are several execution plan to execute a composite service.

## 3. THE OPTIMAL COMPOSITE SERVICE

In the system space, many candidate services are existed by different providers for each operation, that it is difficult and almost impossible to select suitable service from them by user [2],[7]. Then with automatic composition of services, a computer system can select a suitable service from a lot of services for every task, with the least time and high attention. For each task on execution plan should select a service that its functional properties show its ability for doing this task, and according to user criteria, the obtained service also has the best quality until finally obtain an optimal composite service. Optimization can be considered in two forms of local and global optimization. In local optimization for each task a service with the highest QoS will be selected from the total candidate services. This method has high speed but has not any guaranties for all service optimization. In global optimization, these services are selected for each task that with putting them together, finally QoS of composite service have the highest possible value. The method that is stated in this paper, considers global optimization in a manner that a chosen composite service has had ability to solve the user's needs.

### 3.1. Quality of Composite Service

Quality of each service is specified according to quality criteria related to that service. According to four generic quality criteria (cost, response time, availability and reliability) for elementary services, quality vector of service S is defined as (1):

$$Q(s) = (q_{\cos t}(s), q_{Rt}(s), q_{Av}(s), q_{Re}(s)) \quad (1)$$

Methods for compute the quality criteria values are different. These values are used for computing QoS of composite services. There are some aggregation functions that are used to compute the QoS of each quality criterion's composite service. Table 1 provides some of these aggregation functions for an execution plan [10].

*Table 1. Aggregation function for computing quality values*

| QoS Attr. | Sequence | Switch | Flow | Loop |
|---|---|---|---|---|
| **Time (T)** | $\sum_{i=1}^{m} T(t_i)$ | $\sum_{i=1}^{n} p_i * T(t_i))$ | $Max\{T(t_i)_{i\in\{1..p\}}\}$ | $k * T(t)$ |
| **Cost (C)** | $\sum_{i=1}^{m} C(t_i)$ | $\sum_{i=1}^{n} p_i * C(t_i))$ | $\sum_{i=1}^{p} C(t_i)$ | $k * C(t)$ |
| **Availability (A)** | $\prod_{i=1}^{m} A(t_i)$ | $\sum_{i=1}^{n} p_i * A(t_i))$ | $\prod_{i=1}^{p} A(t_i)$ | $A(t)^k$ |
| **Reliability (R)** | $\prod_{i=1}^{m} R(t_i)$ | $\sum_{i=1}^{n} p_i * R(t_i))$ | $\prod_{i=1}^{p} R(t_i)$ | $R(t)^k$ |
| **Custom Attr. (F)** | $f_s(F(t_i))$ $i \in \{1 \ldots m\}$ | $f_B((p_i, F(t_i)))$ $i \in \{1 \ldots n\}$ | $f_F(F(t_i))$ $i \in \{1 \ldots p\}$ | $f_L(k, F(t))$ |

As depicted at table 1, for a Sequence construct of tasks $\{t_1 \ldots t_m\}$, the Time and Cost functions are additive while Availability and Reliability are multiplicative. The Switch construct of Cases $1,\ldots,n$, with probabilities $p_1,\ldots,p_n$ such that $\sum_{i=1}^{i=n} p_i = 1$, and tasks $\{t_1,\ldots, t_n\}$ respectively, is always evaluated as a sum of the attribute value of each task, times the probability of the Case to which it belongs. The aggregation functions for the Flow construct, are essentially the same as those for the Sequence construct, except for the Time attribute where this is the maximum time of the parallel tasks $\{t_1,\ldots,t_p\}$[8]. Finally, a Loop construct with k iterations of task t is equivalent to a Sequence construct of k copies of t [10]. Of course, this table includes a lot of quality attributes. As mentioned in the last line, other features are definable by user.

So, using above aggregation functions, the quality vector of a composite service's execution Plan p is defined as in (2):

$$Q(p) = (Q_{\cos t}(p), Q_{Rt}(p), Q_{Av}(p), Q_{Re}(p)) \quad (2)$$

The related quality vectors to feasible execution plans can be saved in the system and can be used for selecting the optimal execution plan.

### 3.2. Selecting the Optimal Execution Plan

The basic idea of global planning is the same as query optimization in database management systems. Several execution plans are identified and the optimal plan is selected. We assume that for each task $t_j$, there is a set of candidate web services $S_j$ that are available to which task $t_j$ can be assigned. Based on the available web services, by selecting a web service for each task through an execution path, the global planner will generate a set of execution plans P is defined as in (3):

$$P = \{p_1, p_2, \ldots\ldots\ldots .., p_n\} \quad (3)$$

*n* is the number of execution plans. After a set of execution plans is generated, the system needs to select an optimal execution plan. When selecting the execution plan, instead of computing the quality vector of a particular web service, each execution plan's global service quality vector needs to be computed. Once the quality vector for each execution plan is derived, the matrix Q, that each row represents an execution plan's quality vector is obtained as in (4):

$$Q = \begin{pmatrix} Q_{1,1} & Q_{1,2} & Q_{1,3} & Q_{1,4} \\ Q_{2,1} & Q_{2,2} & Q_{2,3} & Q_{2,4} \\ . & . & . & . \\ . & . & . & . \\ Q_{n,1} & Q_{n,2} & Q_{n,3} & Q_{n,4} \end{pmatrix} \quad (4)$$

There are some techniques for selecting the optimal execution plan, but the purpose of this paper is designing one fuzzy system that scores execution plans, while the execution plan with higher score is selected as an optimal execution plan.

## 4. FUZZY-BASED APPROACH ACCORDING TO USER PREFERENCES

User's need to use considered services with different quality properties cause them to have a determinative role in the process of service composition. For example, the cost criterion may be the first grade importance for a user, but his need can be provide with a medium response time, and for other users these preferences are vice versa. The main problem is to provide an approach that presents a way for selecting the optimal composition of services according to user preferences and quality criteria of services and their aggregation values based on a specific composite service. Our work is an approach that relies on the concept of domain ontology for description of services by specifying valid vocabulary and adding semantic concepts for description of services. These vague semantic descriptions located in the form of fuzzy rules and create a criterion to measurement of composite services, and then determine and measure the importance of each rule according to user's clear point of views. In fact, we provide an approach for giving score to composite services by entering the user's point of views in the process of fuzzy inference.

### 4.1. Definition of Variables and Membership Functions of the System

In many application domains, the transition between the memberships of an individual from one set to another is smooth. Consider, For example, height of a human. Small children grow, but when do they stop to be small. Such kinds of knowledge can be encoded using techniques from fuzzy logic [15].

Vague knowledge, i.e. rules based on fuzzy logic, are also important from the perspective of evaluating values of attributes that have very complex dependencies with other attribute values. The vague membership functions can be modeled in the form of some sets by fuzzy logic. On the other hand, in simplest form, a domain ontology would specifies the valid vocabulary of describing (naming) functional and nonfunctional properties that are allowed to occur in service descriptions, but we need a domain ontology that can help in defining categories through linguistic variables. For example, the response time could be described with the terms fast, normal, slow, very slow [14].

With complete knowledge of linguistic variables, we can define the membership functions. In our work, we use the triangular and trapezoidal shapes for defining membership functions. Fig. 2 shows the membership function of response time fuzzy variable.

According to expressed quality criteria, we define linguistic variables in the form of fuzzy sets based on domain ontology to describe web services, as defined in Fig. 3. The reason of using triangular shapes for defining input variables and defining variable terms as a symmetrical form is permanent change at input membership functions and the distinction between different quality vectors, but their definition is possible as Fig. 2.

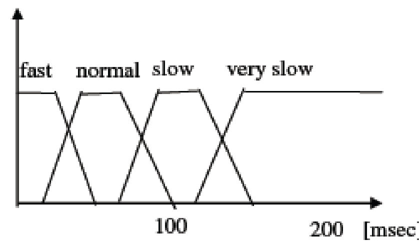
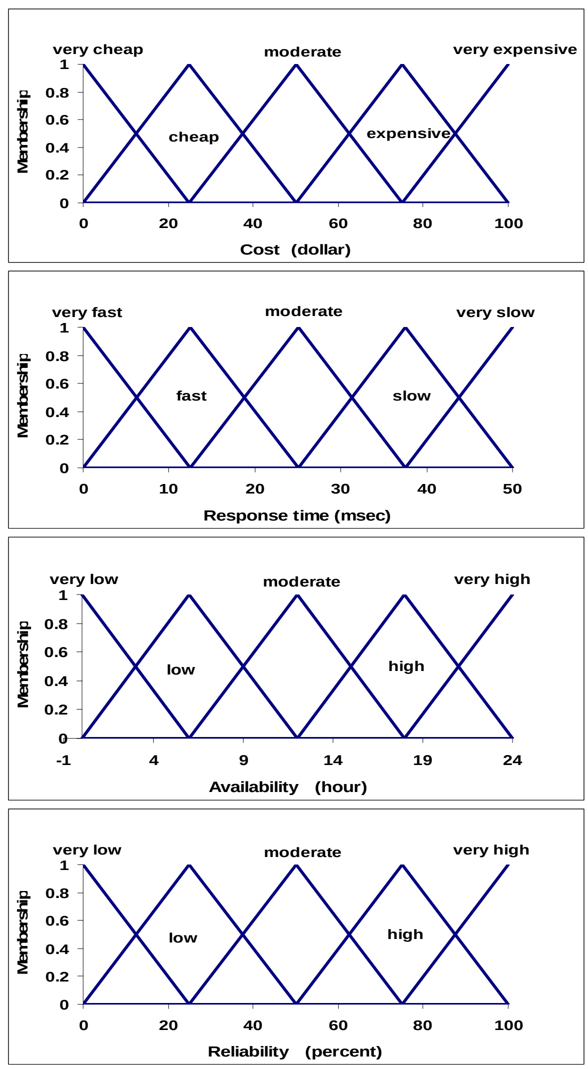



*Figure 2. membership functions for fuzzy terms defining the response time of a service[14]*



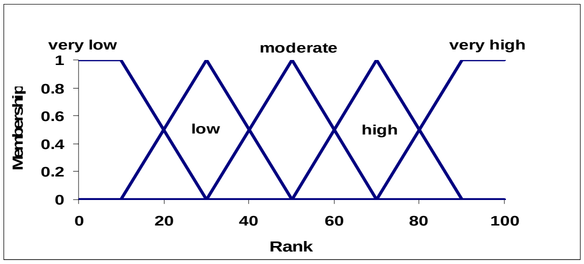



*Figure 3. membership functions for defining linguistic variables of the system*

To define membership functions in this approach, it is important to use equal terms for definition of system's linguistic variables. It is considerable, because of logical relationship between the input and output variable terms in the fuzzy rules formation of the system.

## 4.2. Modeling User Preferences Based Weighting the Rules

We view preferences as the information that describes the constraints on the properties of an individual in order to be accepted for further consideration. We specify different levels of acceptance with fuzzy membership functions.

We model user preferences with fuzzy IF-THEN rules. Fuzzy IF-THEN rules allow to evaluate good approximations of desired attribute values in the very effective way [16],[17]. The IF part consists of membership function of various properties of an individual, and the THEN part is one of the membership functions of a special concept called Rank. Intuitively, a fuzzy rule describes which composition of attribute values a user is willing to accept to which degree, where attribute values and degree of acceptance are fuzzy sets, i.e. vague. An example of fuzzy rule can be:

IF Cost=Cheep and Res.Time=Fast THEN Rank=High

This approach with assumption existence of fuzzy rules that can be criteria for ranking quality vectors related to feasible execution plans, gives more weight to the rules that are more important from user's point of view.

The confidence factor (CF) of every rule which is a number between 0 and 1, can express the confidence value and the importance of the rule to obtain the final result. Equation (5) expresses the effect of this factor in computing the result [18].

$$\text{Membership}_{\text{con,i}} = \text{Membership}_{\text{premise,i}} \times \text{CF}_{\text{i}} \quad (5)$$

This equation shows that the membership function of conclusion part in each rule *i*, is a coefficient from membership function of premise part and the confidence factor, that is related to that rule.

We can provide the preliminary of fuzzy system with complete understanding and knowledge from the quality criteria and the defining input and output linguistic variables with equal terms. After that, we obtain some category of rules by creating fuzzy rules equal to number of terms that are used for defining linguistic variables for every input variable. For the expression of rules, we obtain some categories of rules, by creation one logical mapping between input variable terms in premise part and output variable in conclusion part for every category, that the effect of each rule at ranking should be distinct by the user. This work is done by catching the importance grade of each input quality criterion and located it as a confidence factor related to one category of rules. Therefore we define the importance grade as a number from 0 to 100 and by conversion of distance is used as confidence factor.

For introducing fuzzy rules, we must create a logical mapping according to this point that low or high value of variable is considerable for user. The fuzzy rules for cost variable that low value of this variable is considerable for user can be expressed as follow:

$CF_{cost}$ *IF Cost=very cheap THEN Rank=very high*

$CF_{cost}$ *IF Cost=cheap THEN Rank=high*

$CF_{cost}$ *IF Cost=moderate THEN Rank=moderate*

$CF_{cost}$ *IF Cost=expensive THEN Rank=low*

$CF_{cost}$ *IF Cost=very expensive THEN Rank=very low*

While, we express the fuzzy rules for availability variable that high value of this variable is considerable for user as follow:

$CF_{av}$ *IF Availability=very high THEN Rank=very high*

$CF_{av}$ *IF Availability=high THEN Rank=high*

$CF_{av}$ *IF Availability=moderate THEN Rank=moderate*

$CF_{av}$ *IF Availability=low THEN Rank=low*

*$CF_{av}$ IF Availability=very low THEN Rank=very low*

As pointed out above, we express (N=20) fuzzy rules for the fuzzy system that are according to the numbers of system linguistic variables and variable terms, the number of these rules are variable. These rules are criterion for evaluating different composite services.

### 4.3. Design of system

Figure 4 shows an aspect of the system. The system has several components which are described at below.

The complete knowledge of quality criteria and the definition of linguistic variables and membership functions have a determinative role in fuzzification process of composite services. The fuzzy rules that expressed based on the logical mapping between terms of linguistic variables, are the criterion for evaluating of different composite services. But, these rules are completely neutral against previous approaches of selecting suitable composition. Therefore, the user's role for preferring the rules express his needs increase. As observed in the figure, the received user preferences are based on the importance grade that is given to each quality criterion. Then by changing distance, this numbers stated as confidence factors or weight of each category of rules.

The plan generation unit produces all feasible plans based on workflow and presents services for doing tasks. These plans can be limited by user constraints. For example, at composite service of travel planner user can determine the maximum cost that he can pay for hotel or car rent and so, infeasible execution plans will be omitted.

On the other hand, aggregation functions for computing QoS of execution plans formed quality gathering unit which create quality vector of each execution plan.

Finally, the ranking of execution plans unit is that gives score to each of quality vectors by preferred fuzzy rules and fuzzy inference engine and the specified defuzzification method. The execution plan with the highest score is the indicator of the optimal composition of services.

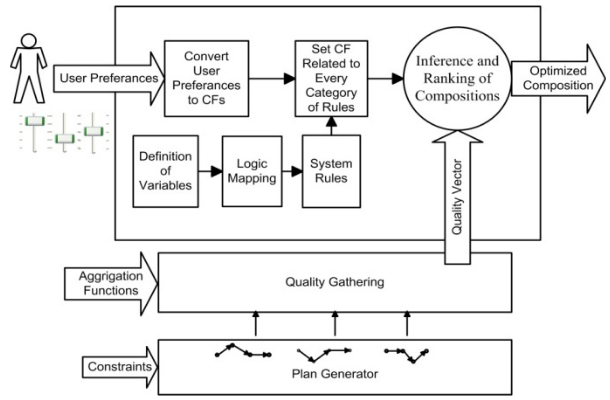



*Figure 4. The general view of designed system*

## 5. IMPLEMENTATION

We implemented this approach for designing fuzzy system by MATLAB application. We use text file for saving the quality vectors.

Fig. 5 Shows a quality vector and computed score in the system and also value of Membership functions interference in each fuzzy rule and final score that is computed. Confidence factor for all rules is 1.

This approach selects the best composite service from the user's point of view by computation the score of each execution plan's quality vector. If we want to see whether the implemented system has functionality correspond to the user, we review Fig. 1. We suppose that for tasks $t_6$, $t_7$ there are two candidate services and for others there is one candidate service with quality criteria values introduced in Fig. 6. Also we suppose that, the switch probability for tasks $t_6$, $t_7$ is equal to 0.5.

We call the execution plans of services $S_6$, $S_7$, $S_8$, $S_9$ respectively $P_1$, $P_2$, $P_3$, $P_4$. We compute the cost and response time attributes, by using of expressed aggregation functions. Table II. shows quality values and also the computation ranks for each of execution plans in two cases.

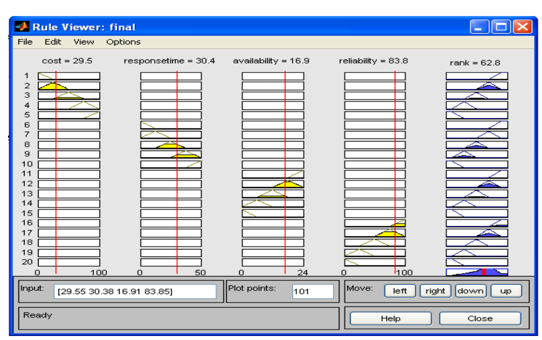



*Figure 5. A view of implemented system*

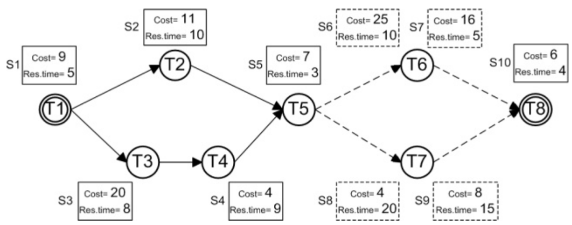



*Figure 6. The state chart of composite service and the values of quality criteria*

*Table 2. Value of Quality attributes and rank of execution plans*

| Ex.Plan | $Q_{cost}$ (p) | $Q_{res.t}$ (p) | $Rank_1$ | $Rank_2$ |
|---|---|---|---|---|
| $p_1$ | 69.5 | 34 | 34.6 | 35.4 |
| $p_2$ | 65 | 31.5 | 38.3 | 39.3 |
| $p_3$ | 59 | 39 | 39.8 | 30.6 |
| $p_4$ | 61 | 36.5 | 39.3 | 33.5 |

In the first case, if the user considers the importance grade 80 for cost criterion, 20 for response time and 0 for other criteria, the computation rank for execution plan $P_3$ is more by the implemented system. This privilege orders to select this execution plan.

But, in second case, if user considers the importance grade 20 for cost criterion, 80 for response time and 0 for other criteria, the computation ranks leads to select $P_2$ execution plan.

With assumption that exist quality vectors for one operation, we want from some users that with determining importance grade of each quality criterion, select the best execution plan from his point of view and then compare result with execution plan that selected by implemented program. Comparison shows that the provided result from the system is corresponded to user preferences and even in the cases that user is not able to select, system can do it.

Now, we set confidence factor of each quality criterion equal to 1 and draw charts related to changes of system variables. Fig. 7 shows the changes of cost variable and results of these changes on rank of composite services. The output level from the changes of input variables, show the logical changes on rank of composite services.

Then, with decreasing confidence factor related to this criterion and fixing other criteria, we can observe that, the chart gradient and width of ranking scores in the each case of confidence factor's reduction, is lessen. This subject is true for other criteria and is a reason for correct functionality of system. Fig. 8 shows difference of maximum and minimum ranking scores belong to composite services against changes of confidence factor related to one quality criterion and fixing the other criteria in 1.

As observed here, the user's point of view has direct effect on computed score for a composite service. In fact, user can select a suitable composition through his point of view.

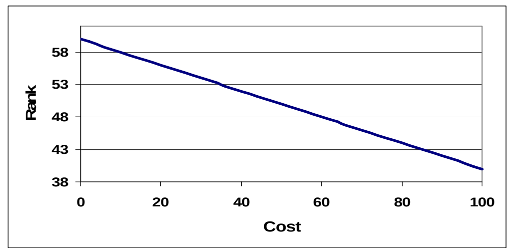



*Figure 7. Effect of cost variable changing on rank variable*

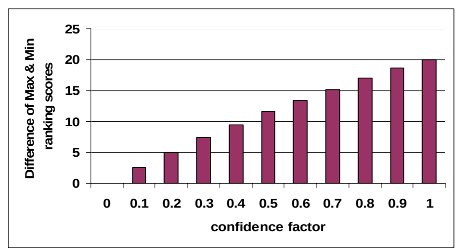



*Figure 8. Effect of confidence factor changing on width of ranking scores*

## 6. CONCLUSIONS

Since, sometimes, the user's requirements are not provided by individual services, we must achieve to noteworthy service by composition of some services. The presence of several suppliers leads to creation of services which have different quality properties. The various needs of the users for using services with different quality properties cause that the user's point of view has had a determinative role in the process of service composition.

In this paper, we presented a technique for selecting the optimal composition of services based on fuzzification of quality criteria of services and introducing fuzzy rules which are a criterion for ranking different composite services. For modeling user preferences we used the fuzzy rules with logical mapping between terms of input and output variables. In this technique, after catching the importance grade of each quality criterion from the user, the composition with the highest score is selected based on aggregated qualitative information of each feasible execution plan.

Several advantages can be stated for this technique. This technique emphasizes on accordance to the user preferences and quality properties of composite service. The user clearly states his preferences for selecting the composition of services. Therefore, additional to high care in expression of preferences, for modeling the different user preferences, there is no need to restate the rules according to these different preferences. Also, this technique is extensible against increasing of the quality criteria.

Finally, in order to optimize the selection of the composition of services, it may be useful to use one technique such as genetic algorithms, which the fitness of execution plans are computed according to our approach, that is described in this work.

**REFRENCES:**


[1] Li-Xin.Wang, "Course in fuzzy systems and control," *Prentice Hall* , ISBN:978-0135408827, 448 pages, Jun 1997.

[2] I.B.Arpinar, B.Aleman-meza, R.Zhang, A.Maduko, "Ontology-Driven Web Services Composition Platform," s.l. *IEEE International Conference on E-Commerce Technology (CEC'04),* 2004.

[3] L.Zeng, B.Benatallah, M.Dumas, J.Kalagnanam, Q.Z.Sheng, "Quality driven web services composition." *In 12th International conference of WWW* (pp. 411–421), 2003.

[4] J.Rao and X.Su, "A Survey of Automated Web Service Composition Methods," s.l. *ICAPS,* 2005.

[5] B.Srivastava and J.Koehler, "Web Service Composition - Current Solutions and Open Problems," s.l. *ICAPS*, 2004.

[6] C.Zhou, L-T.Chia, B-S.Lee, "Semantics in service discovery and QoS measurement," *IT Professional*, 7(2), 29–34. 2005.

[7] J.L.Ambite, G.Barish, C.A.Knoblock, M.Muslea, "Getting from Here to There: Interactive Planning and Agent Execution for Optimizing Travel," s.l. *Proceedings of the Fourteenth Conference on Innovative Applications of Artificial Intelligence* (IAAI-2002), 2002.

[8] J.Cardoso, "Quality of service and semantic composition of workflows," *PhD thesis, Univ. of Georgia,* 2002.

[9] J.Cardoso, A.Sheth, J.Miller, J.Arnold, and K.Kochut, "Quality of service for workows and web service processes," *Web Semantics: Science, services and Agents on the World Wide Web*, 1(3):281-308, April 2004.

[10] G.Canfora, M.D.Penta, R.Esposito, and M.Villani, "An approach for QoS-aware service composition based on genetic algorithms," *In Proceedings of the Genetic and Evolutionary Computation Conference (GECCO),* Washington, DC, June 2005.

[11] H.C-L and K.Yoon,"Multiple Criteria Decision Making." *Lecture Notes in Economics and Mathematical Systems*, Springer-Verlag, 1981.

[12] I.E.Grossmann, "Review of nonlinear mixed-integer and disjunctive programming techniques," *Optimization and Engineering,* 3(3):227-252, September 2002.

[13] M.C.Jaeger, G.Rojec-Goldmann, G.M¨uhl, "QoS Aggregation in Web Service Compositions," *In: Proceedings of IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE-05),* China, IEEE Press (2005). 181–185. 2005.

[14] I.Sora, D.Todinca, C.Avram, "Translating user preferences into fuzzy rules for the automatic selection of services," *5th International Symposium on Applied Computational Intelligence and Informatics.* Timisoara, Romania, May 2009.

[15] S.Agarwal and S.Lamparter, "User preference based automated selection of web service composition." *in Proceedings of the ICSOC Workshop on Dynamic Web Servicesses*, 2005.

[16] L.A.Zadeh,"fuzzy sets", *Information and control*,8, 338-353,1965.

[17] R.R.Yager, S.Ovchinnikov, R.M.Tong, H.T.Nguyen, eds, "Fuzzy sets and applications," selected Papers by L. A.Zadeh, *John Wiley and Sons*, New York, 1987.

[18] D.Nauck and R.Kruse, "How the learning of rule weights affects the interpretability of fuzzy systems," *Proceeding of 7th IEEE International Conference on Fuzzy Systems*, pp. 1235-1240, May 1998.