Selecting the Optimal Web Service Composition Based on QoS

Shrouq Al-Shreef, Mahmoud H. M. Saheb
Department of Graduate Studies, Palestine Polytechnic University, Hebron, Palestine
shrouq.1511@gmail.com; alsaheb@ppu.edu

ABSTRACT

The fast spread of web services in our businesses and day-to-day lives has made QoS an essential aspect for both the service providers and consumers. The main problem is how the consumer obtains a high comprehensive quality composite service when there are a large number of web services available; the choice of the optimal path depends on the QoS for every atomic service. Our contribution is studying the influence of the reputation factor in the process of selecting the optimal path in the absence of one of four factors (Availability, Reliability, Response Time, and Price) and the possibility of covering for this absence. We have used the reputation factor when calculating the QoS by using artificial bee colony algorithm for selecting the optimal web service composition; then we analyzed the impact of reputation on the process of selecting web service composition in terms of the QoS and accuracy of the solution. Also, we studied the impact of the reputation factor in the case of the absence of one of the four factors through three experiments and a set of comparisons. The result was that the reputation factor could cover for factors such as availability, Response Time, and technical support. We used multiple linear regression and polynomial regression to show the prediction of the reputation factor using the four other factors. The result had higher confidence when we used multiple polynomial regression where the Residual Sum of Squares (RSS) was less than the multiple linear regression. In addition, we analyzed the association between reputation and the four other factors using ANOVA test; the result indicates that there is a significant association between reputation and (availability, response time, and price), but the association is not significant with the reliability.

Keywords—Web service, Quality of Service (QoS), Service-oriented architecture (SOA), Web Service Composition, Reputation.

1   Introduction

Web services have been gaining popularity since the introduction of Service-oriented architecture (SOA) which is one of the latest software architectures and cloud computing. SOA has been created primarily to meet business requirements and to remove the gap between software and businesses [1]. Web service uses a standard-based way to realize SOA. Web services are internet-based modular applications that use the Simple Object Access Protocol (SOAP) for communication and transfers data in XML through the internet [2].

Web Service composition aims at selecting and interconnecting web services provided by different partners according to a business process [3]. In a composite web service, every atomic service has a large

number of web service providers that provide similar functionality service with different non-functional property values. The Quality of Service (QoS) can be used as a criterion for service selection. QoS is considered a measure to differentiate between the services and their providers. From the consumers' perspective, knowing the QoS provided by the service provider plays a crucial role in choosing a particular web service over its alternatives. Through the test of QoS, we can rank the web services from the best to the worst in the service registry. The description of each service includes its functional and non-functional properties. Functional properties present the objectives of the service while the quality of service is a set of non-functional attributes such as response time, throughput, reliability, and availability [4]. Cloud service is becoming popular, and several leading IT enterprises including Google, IBM, Microsoft, and Amazon have started to offer cloud services to their customers [5].

Cloud service selection currently constitutes a major challenge attracting the research community to work on and investigate [6]. According to the customer type, the cloud services are divided into two categories: the enterprise cloud services for small and medium-sized enterprises (SMEs) and the cloud application for individual customers [7]. SOA is considered one of the newest software architectures available. SOA have different characteristics and specifications when compared with other software architectures, but the aim of SOA is to remove the gap between software and businesses and to achieve loose coupling among interacting software components through the use of simple, and well-defined interfaces [1]. SOA mainly includes three interactive entities; service provider, service consumer and service registry [8]. Figure (1) represents the web service model and the components of SOA.

2   Web Service Composition

Web service composition is considered a hot and active research area in SOA. In a lot of business to business applications, a single service is not enough to respond to the user's request, so services should be combined through services composition to achieve a specific user's request. Web services make it possible to achieve interoperability Business-to-Business (B2B) from interconnection services offered by multiple business partners based on business processes. This interconnection of web services to meet a specific business process is called Web Service composition [9].
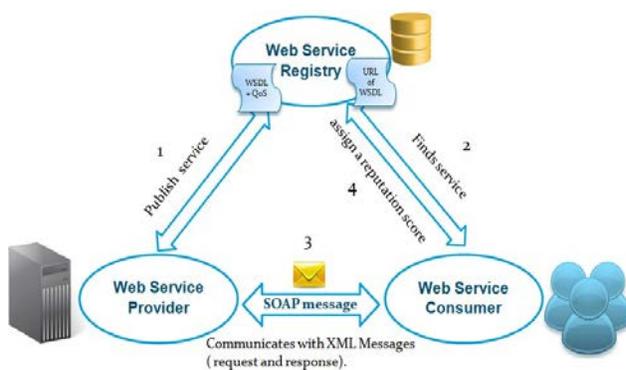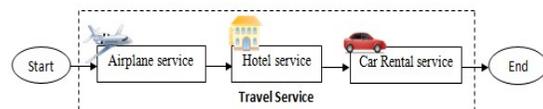

Fig. 1: Web Service Lifecycle   **Fig. 2: Travel Composite Service**

2.1 Example of composition

The composition of service contains a set of atomic services which are combined together on the basis of the rules of composition. The aim of this combination is to satisfy a specific demand of the user and that demand cannot be achieved by a single service. For example, if a user wants to travel to a foreign country (Turkey) so it is not enough for him to book the trip tickets, but there are many things that must be taken into account, he might reserve a hotel, rent a car, etc. Figure (2) explains the meaning of composite service through an example on travel composite service.
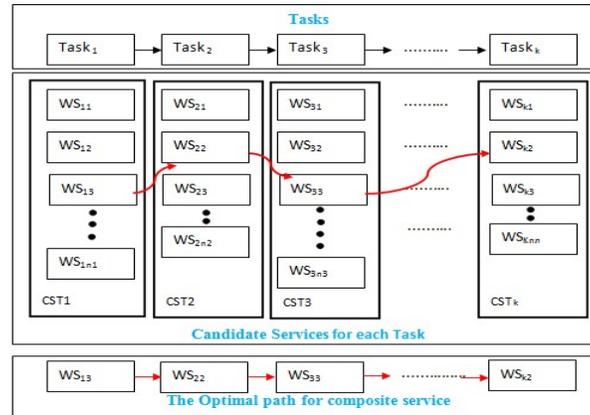

Fig. 3: Structure of Web Services Composition

2.2 Structure of Web Services Composition

In the previous example, in Figure (2), we assumed the composite service contains 3 tasks, and each task has 25 candidate services. Then there will be 25^3 possible web service selection solutions. Making the optimal web service selection decision from such a large number of possible solutions is computationally intractable. In general Figure (3) describes the nature of the structure of web services composition. As it is shown in Figure (3) there are a number of tasks (k). In each task there are a set of providers providing a similarity service ($p_1, p_2, p_3, \cdots\cdots, p_n$) and the providers are called candidate service ($cs_1, cs_2, cs_3, \cdots\cdots, cs_n$). The number of probability of choosing the optimal path is huge. The number of the probability of paths can be represented in the equation (1), where C_1 represents the number of candidate services in Task 1 while C_2 represents the number of candidate services in Task 1 and so on. k is the number of tasks while the C_k represents the number of candidate services in Task k.

$$\# \text{ of paths} = C_1 * C_2 * C_3 \cdots\cdots * C_k \qquad (1)$$

The general term of quality of service (QoS) is "the totality of characteristics of service that bear on its ability to satisfy stated and implied needs of the user of the service" [10]. Calculating the QoS for each atomic service in the composition dependents on five factors response time, cost, availability, reliability, and reputation.

2.3 Mathematical representation of composition service

The composition service has a set of services $S_i, \forall i \epsilon [1 \cdots\cdots n]$. n is the number of candidate services and have a set of tasks $T_j, \forall j \epsilon [1 \cdots\cdots k]$. k is the number of tasks. An undetermined number of tasks, k, can

be used to compose a service and an unlimited number of services, *n*, for each task *Tj* can be found, as it is shown in equation (2).

$$X_{i,j} = \begin{pmatrix} x_{11} & x_{21} & . & . & . & x_{k1} \\ x_{12} & x_{22} & . & . & . & x_{k2} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ x_{1n} & x_{2n} & . & . & . & x_{kn} \end{pmatrix}$$

(2)

If the service Si is allocated to task *Tj,* set 1 in the matrix, as it is shown in equation (3)

(3)

$$x_{i,j} = \begin{cases} 1, & if\ service\ i\ is\ allocated\ to\ task\ j \\ 0, & otherwise \end{cases}_{\forall i \epsilon [1 \cdots n], \forall j \epsilon [1 \cdots k]}$$

We can consider that composition web service is a set of atomic web services, and the sum of rows and that of columns in matrix X should be 1. In our example Travel composite service, we suppose the composite service contains 3 tasks, and each task has 10 candidate services n =10, k=3. The possible web service selection solution 10^3 = 1000. In the matrix, Y represents one of the possible combinations in which service:

- service S5 will execute task T1
- service S3 will execute task T2
- service S7 will execute task T3 - Task T3 will be executed by service S2.

$$Y_{i,j} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

In matrix Y, the solution of web service composition is (T1-S5, T2-S3, and T3-S7).

3   Literature Review

There are many approaches that have been proposed for web service quality composition modeling. In [11] [12] the authors depend on the Web Service Description Language (WSDL) to define the functional properties and non-functional properties of the service; this approach has some problems such as the issue of run-time support is not addressed. In [13], the authors define QoS for web service by using XML schemas that both service consumers and providers apply to define the agreed QoS parameters; also this approach allows for the dynamic selection of web service depending on various QoS requirements. In [14] the authors proposed a predictive QoS model for workflows involving QoS properties. Many works describe QoS-aware service composition as a multidimensional, multi-objective, multi-choice knapsack problem (MMMKP); which takes many QoS criteria into consideration to obtain an optimal composite service that has high QoS value [15]. Some approaches focus on reducing the complexity of the composition using standard optimization algorithms [16] [17]. Some researchers propose QoS constraints such as minimum availability and reliability to restrict the composite services [18] [19]. There are many

approaches that are presented for the web services discovery process, and the main idea of these approaches is to ensure the correctness of the value that assigned by the consumer regarding the reputation of the service; also, some of these works are based on the functional properties, and the others are based on both the functional and nonfunctional properties (QoS). Al-Shargabi et al. 2014. Present a web service composition technique based on the user preferences such as price and availability; furthermore, they use a web service selection agent forced by users preferences, but the effect on the reputation in the selection approach did not be included [20]. Zhang et al. 2014. They have presented a tool or WS-QoS measurement, and this tool calculates the reputation based on the similarity values between the value offered by the service and the measured quality data value, but this algorithm is not updated for the trust and reputations, making trustworthiness information reflect the latest changes in service [21]. Nianhua et al. 2012. They have presented a reputation evaluation algorithm based on the similarity theory for the newly added web service; in addition, they use trust and similarities as weights for computing reputations from different recommenders [22]. Wang et al. 2011. Propose an approach for measuring reputation precisely, and they give a solution for the malicious rating of service users, and their approach, including two phases. In the first phase, they detect malicious feedback ratings using the Cumulative Sum Method. And in the second phase, they used the Pearson Correlation Coefficient to detect and reduce the effect of different user feedback [23]. Nepal et al. 2010. Present a fuzzy trust evaluation approach for web services, and they present a trust-based reputation management framework for web service selection [24]. Sathya et al. 2010 evaluated the various techniques that are used in the quality of service based service discovery approach; also, they defined a set of criteria for QoS discovery approach; in addition, they organized the approaches into three main categories, including functional-based service discovery approach, nonfunctional based service discovery approach, and user-based service discovery approach [25]. Josang et al. 2008. Used Bayesian reputation systems as a trust model for evaluating the quality of service in a single framework [26].

Most of the researchers assume that the QoS values are obtained from service providers but, may be unreliable, or the QoS values we need may be unknown. Therefore, many researchers have suggestions on how to predict the unknown QoS values [27][28][29]. There are two main types of approaches to predict QoS values for web services the first approach is one is neighborhood-based and the second approach called model-based approaches. Table (I) represents the summary of the main idea, advantage, and disadvantage of the two approaches. Qiu et al. 2013 have proposed a reputation-aware QoS value prediction approach based on CF and in the first step, the authors calculate the reputation of each user based on their contributed values and then takes advantage of reputation-based ranking to exclude the values contributed by unreliable users [30]. Tang et al. 2014 have proposed a hybrid trust-aware service recommendation method for a service-oriented environment with social networks by combining global trust and local trust evaluation [31]. Jianlong et al. 2016 present an effective QoS prediction approach, namely RMF, for predicting unknown web service QoS values [32]..

Table I: Approaches of QoS prediction			
Approach	Main Idea	Advantage	Disadvantage
Neighborhood-Based	Also named collaborative filtering (CF) approaches, which utilize the historical invocation information of similar neighbors to make a prediction.	Easy to understand and implement.	1- Bad prediction accuracy when the data density is very low. 2- Not suitable to be
Model-based	Matrix factorization (MF) is one of the most well-known mode-based, which is to exploit the latent factors that can	Accurate and scalable in many	Building efficient models.

4   Contributions and Methodology

Through studying the previous researches dealing with the same issue of selecting the optimal web service composition and using the bee algorithm to solve the problem, we have found that those researches depended on four main factors to measure the QoS which are response time, availability, cost, and reliability. In this study, we suggested studying the impact of reputation factor on selecting the optimal path and the possibility of reputation covering for the other four factors. We used the reputation factor when calculating the QoS through using artificial bee colony algorithm for selecting the optimal web service composition. Reputation factor is an aggregation of ratings for a service from consumers for a specific period. The main contribution of this study is to cover for the absence of one of the QoS factors by the reputation factor. We discuss our proposed model used to measure the impact of reputation factor on the selection process of composite web service, so we divide our proposed model into five steps, which must be consecutive as Figure (4).

4.1   Structure of Web Service

Initially, the service provider publishes its services and makes them available on the internet, through attaching a description of service and QoS which is in the format of WSDL file, and this file is stored in the web services registry. Then the service consumer tries to search his own desired service in the registry that suits his requirements in term of QoS. The consumer sends his request as input data which in turn is translated into a message in XML language. Then the consumer waits for the response message. The response occurs immediately after the search process, through receiving a message of URL WSDL file about target service. After obtaining the URL of the web service from the registry, the consumer makes a binding with the provider through SOAP message. At this stage, a consumer can use this service efficiently.
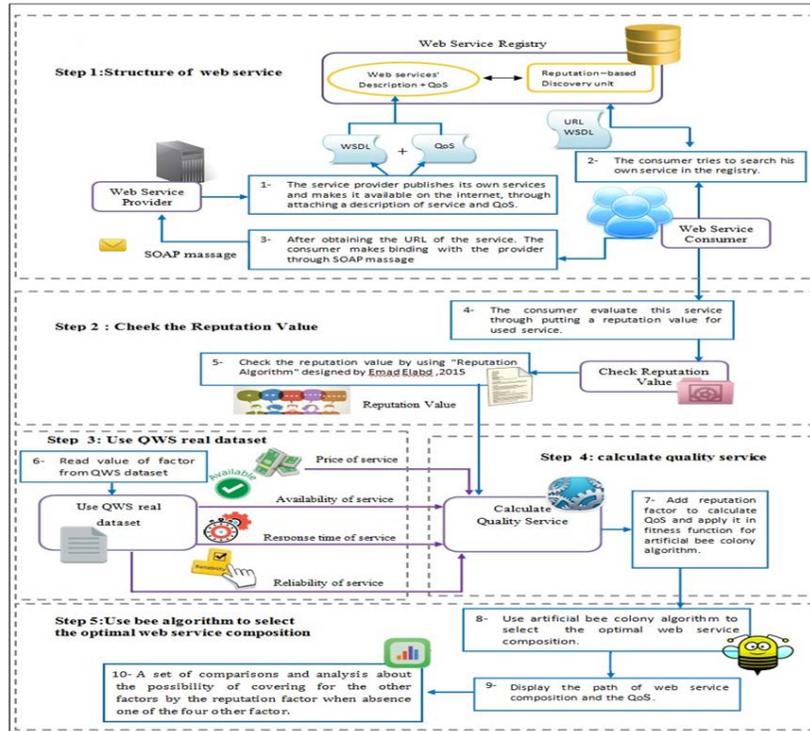
Fig. 4: The proposed model of measuring the impact of reputation factor on the selected web service composition process

4.2 Reputation Module

The service Reputation Module is responsible for collecting data from the service consumer, processing data, updating the Reputation Scores for a related service provider to ensure the integrity and objectivity of a web service reputation evaluation. We used the Reputation algorithm proposed by the El-Kafrawy et al. for updating the reputation of the web service based on the trust factors of the consumers and reputation threshold (RT) [33].

4.3 Aggregation Formulas for QoS Computation of Composite Service

The aggregation rules are different for different QoS properties and based on the composition's control flow. In general, there are four main control patterns: Sequential, Parallel, loop, and Selection. Each of them defines a separate aggregation rule. We have listed the aggregation rules for response time, price, availability, reliability, and reputation in Table (II) [14]. Figure (5) represents the patterns for Service Composition, and this is illustrated in detail as follows: Any execution path of web service composition is composed of four fundamental patterns [34]:

1. Sequential Figure (5.a): The computation formula for QoS is listed in Table (II.a).
2. Parallel Figure (5.b): Each Si is executed in parallel, the computation formula for QoS is listed in Table (II.b).
3. Loop Figure (5.c): Supposing that the circulation model is executed k times. The computation formula for QoS is listed in Table (II.c).
4. 4) Selection Figure (5.d): Supposing that the probability of each service Si being selected

$$P_i . \sum_{i=1}^{n} P_i \ = \ 1$$

The computation formula for QoS is listed in Table (II.d).

Table II: Aggregation formulas to compute the overall QoS of service compositions [14]

property	a- Sequential	b- Parallel	c- Loop	d- Selection
Availability	$\prod_{i=1}^{m} A(S_i)$	$\prod_{i=1}^{p} A(S_i)$	$A(S)$	$\prod_{i=1}^{n} P_i * A(S_i)$
Reliability	$\prod_{i=1}^{m} R(S_i)$	$\prod_{i=1}^{p} R(S_i)$	$R(S)^k$	$\prod_{i=1}^{n} P_i * R(S_i)$
Response	$\sum_{i=1}^{m} T(S_i)$	$Max(T(S_i)_{i\epsilon\{1\cdots p\}})$	$K * T(S)$	$\sum_{i=1}^{n} P_i * T(S_i)$
Time	$\sum_{i=1}^{m} P(S_i)$	$\sum_{i=1}^{p} P(S_i)$	$K * P(S)$	$\sum_{i=1}^{n} P_i * P(S_i)$
Price	$f_S(F(S_i)), i\epsilon\{1\cdots m\}$	$f_P(F(S_i)), i\epsilon\{1\cdots p\}$	$f_L(F(S_i)), i\epsilon\{1\cdots k\}$	$f_C(F(S_i)), i\epsilon\{1\cdots n\}$

Assuming that the other three non-sequential patterns can be converted to a sequential pattern, so in this work, we depend on the sequential pattern as the basis to research the issue of web service composition optimization. Different types of indexes have different dimensions. It is necessary to eliminate the incommensurability stemming from the different dimensions and different dimension unit [35]. Therefore, all indexes need to be normalized to a dimensionless interval according to a certain utility function (usually it is normalized to [0,1]). There are two phases in merging the multi-dimension resource constraints.

1) Scaling Phase: QoS properties were divided into negative and positive. Negative includes response time and price for service. The better service has less response time and less cost, so they are negative and normalized by equation (4).

$$V_{m,j} = \begin{cases} \dfrac{Q_j^{max} - Q_{m,j}}{Q_j^{max} - Q_j^{min}} & , Q_j^{max} - Q_j^{min} \neq 0 \\ 1 & , Q_j^{max} - Q_j^{min} = 0 \end{cases} \qquad (4)$$

(a) Sequential pattern   (b) Parallel pattern   (c) Loop pattern   (d) Selection pattern
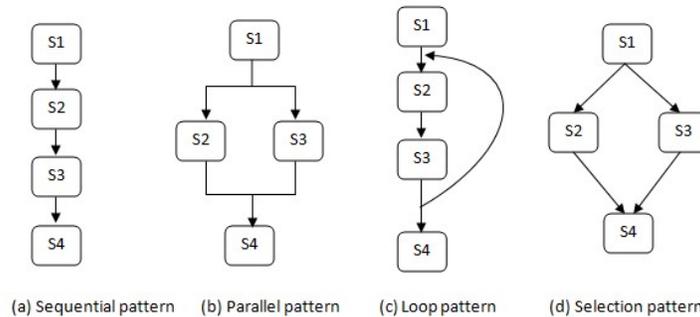
Fig. 5: Four fundamental patterns for Service Composition [34]

Positive includes availability, reliability, and reputation. The better service has higher availability, reliability, and reputation, so they are positive and normalized by Equation (5).

$$V_{m,j} = \begin{cases} \dfrac{Q_{m,j} - Q_j^{min}}{Q_j^{max} - Q_j^{min}} & , Q_j^{max} - Q_j^{min} \neq 0 \\ 1 & , Q_j^{max} - Q_j^{min} = 0 \end{cases} \qquad (5)$$

Where $m = (1,2,3,\cdots\cdots,N)$ represents the number of services, $j = (1,2,3,4,5)$ represents the number of properties ,Q^{min}_j,Q^{max}_j represents the minimum and maximum the jth property, and $Q_{m,j}$ represents the jth property of service m.

2) Weighting Phase: After scaling phase, weight determines based on the nature and type of the web service

$$Q = \begin{cases} \sum_{i=1}^{5} \omega_i V_i \\ \sum_{i=1}^{5} \omega_i = 1 \quad ,1 \geq \omega_i \geq 0 \end{cases} \tag{6}$$

where ω_i represents the weight of the jth property,Q represents the Quality of web service.

5   Experiments and Results

We conducted the experiments on a computer which has the following specification: Intel (R)-Core i5, 2520M CPU, 250 GHz, a 4GB RAM processor, Microsoft Windows 7 and an 8 Mbps Wi-Fi internet connection. Also, Java language is used to build the code of our experiments, and NetBeans IDE 8.0.2 to apply the experiments practically.

In this study, two algorithms are used. The first algorithm is "Update Service Reputation algorithm" and the second one is "Multi-Objective Bees Algorithm". Update Service Reputation algorithm is used for updating the reputation value for the service based on the value given by the consumer after using it. This algorithm was also used for updating the trust factor of the consumer. Multi-Objective Bees Algorithm was used to find the optimal composite web service based on QoS.

In our experiments, a dataset with real QoS from Cloud Armor is used. Cloud Armor is a research project at the University of Adelaide which aims at developing a scalable trust management system for cloud services [36]. In this project, the researchers gathered the consumers' feedback from a set of cloud computing providers such as Cloud Hosting Reviews, Best Cloud Computing Providers, and Cloud Storage Reviews and Ratings [36]. The dataset contains approximately 10,000 feedback by 7,000 consumers for 113 real-world cloud services, and the feedback are based on Quality of Service (QoS) attributes [36]. In addition, we added some missing information in the dataset that was obtained from websites that test the quality of service [37][38]. After that, a simulation program using Java language is built to evaluate each web service based on "trust result" attribute in the dataset. This attribute depends on the user's opinion of the evaluated service. It is also used to measure the impact of reputation factor on selecting composite web service. In addition, we used the program to create a set of comparisons to show the possibility of covering for the other factors by the reputation factor. We suppose that the composite service contains three tasks as a simple composite service. We divided the dataset into four sets, each having three tasks and each task has about nine candidate services. Each candidate service has four real QoS properties (Ava, Rel, RT, P).

5.1   Results of effecting reputation factor on other QoS factors

We set up an experiment to study the effect of reputation factor on other QoS factors (Ava, Rel, RT, P). More specifically, our experiment aims to check whether the reputation factor will cover the absence of one or more QoS factor or not. To this end, we conduct our experiment three times. The first setting is conducted using the data of the four factors: availability, reliability, response time, and price. In the second setting, we replaced the reliability factor with the technical support factor due to the lack of

sufficient information about the reliability factor in the dataset. The third setting is conducted on services that have full data information. In each experiment's setting, we suggested 7 cases for selecting the optimal composite path using bee algorithm. We applied bee algorithm into several cases to study the possibility of covering reputation factor during the absence of one of the other four factors. We made a comparison between the case in which the reputation factor is absent with a set of cases that have the reputation value but lack one of the other factors. The following cases were suggested:

- Case 1: in this case, we took the following factors (Ava, Rel, RT, P) to get an optimal solution without considering the reputation factor in QoS
- Case 2: in this case, we took the following factors (Rel, RT, P, Rep) to get an optimal solution without considering the availability factor in QoS.
- Case 3: in this case, we took the following factors (Ava, RT, P, Rep) to get an optimal solution without considering the reliability factor in QoS.
- Case 4: in this case, we took the following factors (Ava, Rel, P, Rep) to get an optimal solution without considering the response time factor in QoS.
- Case 5: in this case, we took the following factors (Ava, Rel, P, Rep) to get an optimal solution without considering the price factor in QoS.
- Case 6: in this case, we took the all five factors (Ava, Rel, RT, P, Rep) to get an optimal solution.
- Case 7: in this case, we only took the reputation factor to get an optimal solution.

We conducted the selection process 10 times to get an optimal path with the highest quality. In addition, this has been applied to all four sets and also for all suggested cases.

5.1.1   Results of experiment 1:

Table (III) contains all the results of the first experiment for all sets (Set1, Set2, Set3, and Set4). The column (QoS) in the table represents the quality of service value that was obtained by the selected path using the bee algorithm while the (accuracy of the solution) represents the proximity of the optimal solution. The matching ratio represents the result of comparing the path obtained from case 1 with the path obtained from each of the other suggested cases. Since the dataset contained null values, the percentage was calculated in the tasks in all the suggested case. The average of the null value for three tasks was calculated as the null value ratio. Figure (6) represents the differences of QoS for each set in all the cases. As shown in Figure (6), we note that the factor of availability is an important factor because in its absence the QoS was low and in contrast, the price is less important and as shown in the figure the absence of price factor has no significant impact on QoS. Figure (7) represents the accuracy of the solution path for each set in all the cases in the absence of one of the four factors. As shown in Figure (7), the case of "4 Factors without availability" has received less accuracy of solution for all sets but the case of "4 Factors without price" has received higher accuracy of solution for set 1, set 2 and set 4. We compared the path result of the first case with the results of the rest suggested cases which have mentioned in the previous section to analyze the match ratio in the solution of the path. For example, when taking the case when Availability factor was absent and comparing it to the case with no reputation, the result showed that the percent of matching was 33% and the similarity is in the service that achieved the task one (T1). In addition, when comparing the case of "4 factors without reputation" with the case of "4 factors without reliability" indicates that the proportion of similarity was 66% and the similarity is in the service that achieved the task one (T1) and task three (T3). The same results apply to the case of "4 factors without price". In contrast, there is no similarity when comparing the case of "4 factors without reputation" with the cases of "4 factors without response time" and "1 factor: the reputation". We explained the ratio of the

similarity through a set of figures. It was found that the orange color area which represents the absence of the factor of reliability, appeared in all four sets. It constituted 66%, 100%, 66%, 33% in Figures 8, 9, 10 and 11 respectively. This indicates that in the case of absence of the reliability factor the reputation factor can cover for it because there is a matching ratio, but there is some doubt about reliability factor because the dataset contained null values especially the reliability, so the experiment was repeated by replacing the reliability with technical support because it contained sufficient data. Because the similarity of the path is incomplete, we analyzed the percentage of the null values in the dataset. It was found that the coverage area had the highest null ratio. The null ratio for figures 12, 13, 14, and 15 was 25.80%, 34.10%, 32.40% and 24.40%. By looking at the figures that represent the matching ration the area in the case of absence of response time, it seems that there is little or no matching which indicate that there is a relationship between the proportion of matching and the proportion of null values.

Table 3: The analysis of the first experimental results in the absence of one of QoS factors

Experiment 1: all Factors used are Availability, Reliability, Price, and Response Time											
	Set 1						Set 2				
	4 Factors without Reputation (Ava, Rel, RT, P)						4 Factors without Reputation (Ava, Rel, RT, P)				
Comparison	QoS	Accuracy of the solution	Match ratio of the path	Match areas	Null value	Comparison	QoS	Accuracy of the solution	Match ratio of the path	Match areas	Null value
4 Factors without Availability (Rel, RT, P, Rep)	3.0087	0.752175	33%	T1	T1=19.4%, T2=10%, T3=25%	4 Factors without Availability (Rel, RT, P, Rep)	3.0699	0.767475	100%	T1,T2,T3	T1=30.5%, T2=19.4%, T3=12.5%
4 Factors without Reliability (Ava, RT, P, Rep)	2.9688	0.7422	66%	T1, T3	T1=16.6%, T2=12.5%, T3=16.6%	4 Factors without Reliability (Ava, RT, P, Rep)	3.0699	0.767475	100%	T1,T2,T3	T1=27.7%, T2=16.6%, T3=12.5%
4 Factors without Price (Ava, Rel, RT, Rep)	3.1823	0.795575	66%	T1, T3	T1=30.5%, T2=16%, T3=31.1%	4 Factors without Price (Ava, Rel, RT, Rep)	3.4877	0.871925	66%	T1, T3	T1=50%, T2=30.5%, T3=21.8%
4 factors without Response Time (Ava, Rel, P, Rep)	3.1389	0.784725	0%	No match	T1=22.2%, T2=17.5%, T3=30.5%	4 factors without Response Time (Ava, Rel, P, Rep)	3.0818	0.77045	33%	T2	T1=41.6%, T2=25%, T3=21.8%
All five Factors (Ava, Rel, RT, P, Rep)	3.6379	0.72758	100%	T1, T2, T3	T1=24.4%, T2=16%, T3=31.1%	All five Factors (Ava, Rel, RT, P, Rep)	3.9817	0.79634	100%	T1,T2,T3	T1=40%, T2=24.4%, T3=17.5%
1 Factor the Reputation (Rep)	1	1	33%	T3	T1=0%, T2=0%, T3=0%	1 Factor the Reputation (Rep)	1	1	0%	No match	T1=0%, T2=0%, T3=0%

6

	Set 3						Set 4					
	4 Factors without Reputation (Ava,Rel,RT,P)						4 Factors without Reputation (Ava, Rel, RT, P)					
Comparison	QoS	Accuracy of the solution	Match ratio of the path	Match areas	Null value	Comparison	QoS	Accuracy of the solution	Match ratio of the path	Match areas	Null value	Null value ration

4 Factors without Availability (Rel, RT, P, Rep)	2.8362	0.70905	33%	T1	T1=30.5%, T2=16.6%, T3=19.4%						
4 Factors without Reliability (Ava, RT, P, Rep)	3.1213	0.780325	66%	T1,T2	T1=22.2%, T2=16.6%, T3=22.2%	4 Factors without Availability (Rel, RT, P, Rep)	3.0247	0.756175	0%	No match	T1=25%, T2=18.7%, T3=29.5%
4 Factors without Price (Ava, Rel, RT, Rep)	2.9556	0.7389	100%	T1, T2, T3	T1=38.8%, T2=30.5%, T3=33.3%	4 Factors without Reliability (Ava, RT, P, Rep)	3.2041	0.801025	33%	T1	T1=27.7%, T2=14.5%, T3=25%
4 factors without Response Time (Ava, Rel, P, Rep)	3.0772	0.7693	0%	No match	T1=41.6%, T2=25%, T3=30%	4 Factors without Price (Ava, Rel, RT, Rep)	3.4179	0.854475	0%	No match	T1=45.4%, T2=25%, T3=38.6%
All five Factors (Ava, Rel, RT, P, Rep)	3.2041	0.64082	66%	T1, T3	T1=33.3%, T2=24.4%, T3=26.6%	4 factors without Response Time (Ava, Rel, P, Rep)	3.2291	0.807275	33%	T2	T1=27.2%, T2=18.7%, T3=27.2%
						All five Factors (Ava, Rel, RT, P, Rep)	3.6733	0.73466	33%	T2	T1=36.3%, T2=20%, T3=30%
1 Factor the Reputation (Rep)	1	1	0%	No match	T1=0%, T2=0%, T3=0%	1 Factor the Reputation (Rep)	1	1	0%	No match	T1=0%, T2=0%, T3=0%
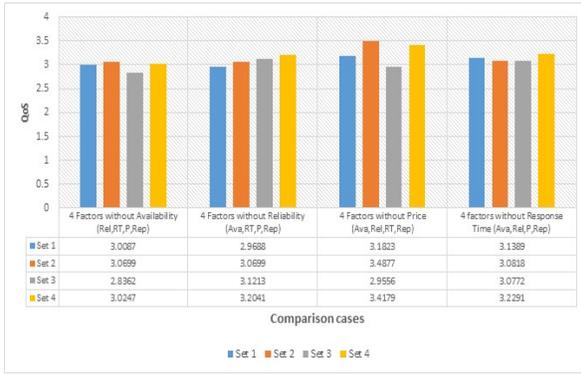
Fig. 6: Experiment 1. The differences in QoS for each set in all cases in the absence one of the four factors.
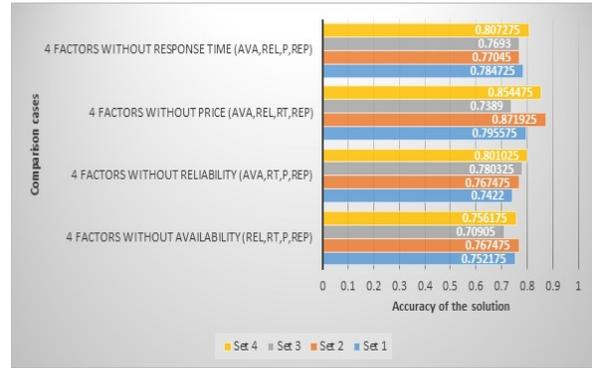
Fig. 7: Experiment 1. The accuracy of the solution path for each all set in all cases in the absence one of the four factors.

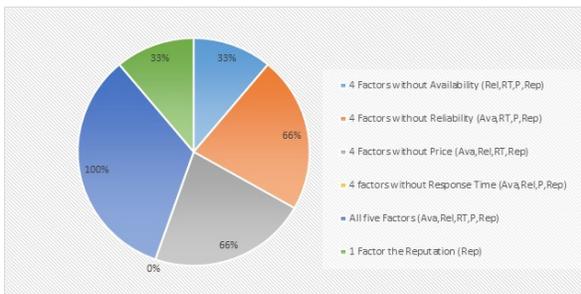**Fig. 8: Experiment 1 (Set 1). The Matching ratio of the paths in all cases with the path in case 1 "4 Factors without Reputation (Ava, Rel, RT, P)".**
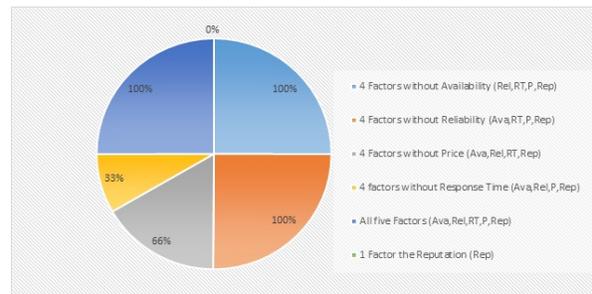

Fig. 9: Experiment 1 (Set 2). The Matching ratio of the paths in all cases with the path in case 1 "4 Factors without Reputation (Ava, Rel, RT, P)".
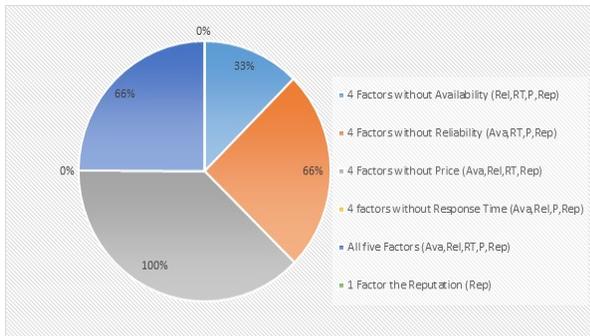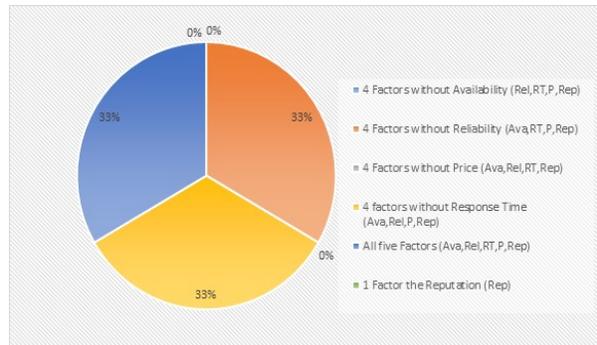
Fig. 10: Experiment 1 (Set 3). The Matching ratio of the paths in all cases with the path in case 1 "4 Factors without Reputation (Ava, Rel, RT, P)".

Fig. 11: Experiment 1 (Set 4). The Matching ratio of the path in all case 1 "4 Factors without Reputation (Ava, Rel, RT, P)".
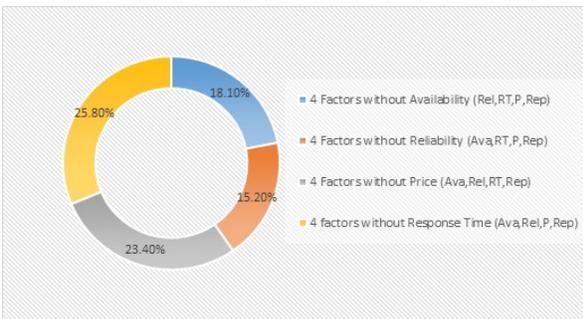

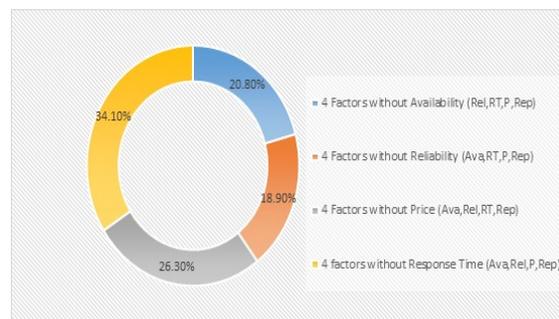**Fig. 12: Experiment 1 (Set 1). The null ratio in all cases.**

Fig. 13: Experiment 1 (Set 2). The null ratio in all cases.

Table 4: The analysis of the second experiment results in the absence of one of QoS factors

Experiment 2: all Factors used are Availability, Technical Support, Price, and Response Time

Set 1				
Comparison	4 Factors without Reputation (Ava, TS, RT, P)			
	Qos	Accuracy of the solution	Match ratio of the path	Match areas
4 Factors without Availability (TS, RT, P, Rep)	2.8861	0.721525	33%	T3
4 Factors without Technical Support (Ava, RT, P, Rep)	2.8146	0.70365	100%	T1, T2, T3
4 Factors without Price (Ava, TS, RT, Rep)	3.4175	0.854375	33%	T3
4factorswithoutResponse Time (Ava, TS, P, Rep)	3.0735	0.768375	33%	T3
All five Factors (Ava, TS, RT, P, Rep)	3.4412	0.68824	33%	T3
1 Factor the Reputation (Rep)	1	1	33%	T3

Set 2				
Comparison	4 Factors without Reputation (Ava, TS, RT, P)			
	QoS	Accuracy of the solution	Match ratio of the path	Match areas
4 Factors without Availability (TS, RT, P, Rep)	2.6429	0.660725	66%	T1, T2
4 Factors without Technical Support (Ava, RT, P, Rep)	3.0699	0.767475	100%	T1, T2, T3
4 Factors without Price (Ava, TS, RT, Rep)	3.2174	0.80435	0%	No match
4factorswithoutResponse Time (Ava, TS, P, Rep)	3.0035	0.750875	0%	No match
All five Factors (Ava, TS, RT, P, Rep)	3.429	0.6858	100%	T1, T2, T3
1 Factor the Reputation (Rep)	1	1	0%	No match
Set 3				
Comparison	4 Factors without Reputation (Ava, TS, RT, P)			
	QoS	Accuracy of the solution	Match ratio of the path	Match areas
4 Factors without Availability (TS, RT, P, Rep)	2.7017	0.675425	66%	T2, T3
4 Factors without Technical Support (Ava, RT, P, Rep)	3.1213	0.780325	33%	T1
4 Factors without Price (Ava, TS, RT, Rep)	3.3869	0.846725	66%	T2, T3
4 factors without Response Time (Ava, TS, P, Rep)	2.9847	0.746175	0%	No match
All five Factors (Ava, TS, RT, P, Rep)	3.585	0.717	100%	T1, T2, T3
1 Factor the Reputation (Rep)	1	1	0%	No match

Set 4				
Comparison	4 Factors without Reputation (Ava, TS, RT, P)			
	QoS	Accuracy of the solution	Match ratio of the path	Match areas
4 Factors without Availability (TS, RT, P, Rep)	3.0251	0.756275	66%	T2, T3
4 Factors without Technical Support (Ava, RT, P, Rep)	3.2041	0.801025	100%	T1, T2, T3
4 Factors without Price (Ava, TS, RT, Rep)	3.8291	0.957275	100%	T1, T2, T3
4 factors without Response Time (Ava, TS, P, Rep)	3.1688	0.7922	33%	T2
All five Factors (Ava, TS, RT, P, Rep)	4.0069	0.80138	100%	T1, T2, T3
1 Factor the Reputation (Rep)	1	1	0%	No match

6.1.1   Results of experiment 2:

 Due to the lack of adequate information and especially the reliability data that was collected in the first experiment from the websites that test the quality of service, we have replaced reliability factor with

another factor, which is technical support due to the availability of its information in the original dataset. The aim of this experiment is to study the possibility of covering for the reputation factor in the absence of technical support factor, so we repeated the experiment using the four factors (Ava, TS, RT, and P). Table (IV) represents a summary of all the results of the second experiment for all sets (Set1, Set2, Set3, and Set4). Figure (16) represent the differences of QoS in experiment 2 for each set in all cases. Figure (17) represent the accuracy of the solution path in experiment 2 for each set in all cases in the absence of one of the four factors. As shown in the figures, the proportion of similarity was generally high, and especially in the case of absence of the technical support factor as it was 100% in Figure (18), Figure (19), and Figure (21) and 33% in Figure (20). The result indicates that the reputation factor can cover the technical support factor.

6.1.2   Results of experiment 3:

Due to the fact that technical support is not considered as one of the four main factors of QoS, we repeated the experiment by using 43 services that had full data about Availability, reliability, response time and price. We distributed the full services on the three tasks where each task had 14 candidate service. Table (V) represents the results of the third experiment in terms of similarity ratio with the path resulting from the case of the absence of reputation. Figure (22) represents the differences in QoS and shows that the availability and reliability factors have less QoS and this indicates their importance. Figure (23) illustrates the highest matching ratio in the absence of the availability factor.

6.1.3   Discussion the results of the experiments:

We summarized the results of the three experiments in Table (VI) that presents the possibility of the reputation factor covering for the absence of one of the other factors. The coverage ration in Table (VI) is the percentage of times that there was a possibility that the reputation factor will cover the other factors, For example, it was 92% for availability because it could cover 3 out of 4 cases in Ex1 and all cases in Ex2 and Ex3. Based on the results of our experiments, it was suggested that the value of the coverage ratio is accepted if it was higher the 50%. We conclude that the reputation factor can replace some factors such as availability, response time and technical support as shown in Figure (25).

Table 5: The analysis of the third experiment results in the absence of one of QoS factors

Comparison	4 Factors without Reputation (Ava, Rel, RT, P)			
	QoS	Accuracy of the solution	Match ratio of the path	Match areas
4 Factors without Availability (Rel, RT, P, Rep)	2.4399	0.609975	66%	T1, T3
4 Factors without reliability (Ava, RT, P, Rep)	2.4399	0.609975	0%	No match
4 FactorswithoutPrice (Ava, Rel, RT, Rep)	2.8295	0.707375	0%	No match
4 factors without Response Time (Ava, Rel, P, Rep)	2.9046	0.72615	33%	T2
All five Factors (Ava, Rel, RT, P, Rep)	2.9429	0.58858	33%	T2
1 Factor the Reputation (Rep)	1	1	0%	No match

6.2 Results of predicting reputation factor based on the other factors

We used the services that have full data for all factors to predict the reputation depending on the other factors; so we divided the full data into two groups, the first group contained 32 rows of services. It was used as training data to obtain an equation using interpolation, but in the second group, we used it as testing data for the 10 rows that were not used in the training data. In addition, we applied this data to the equation given by interpolation to predict reputation factors. Then conducted a comparison between the actual value and the expected value for the reputation, and we calculated the difference between them.
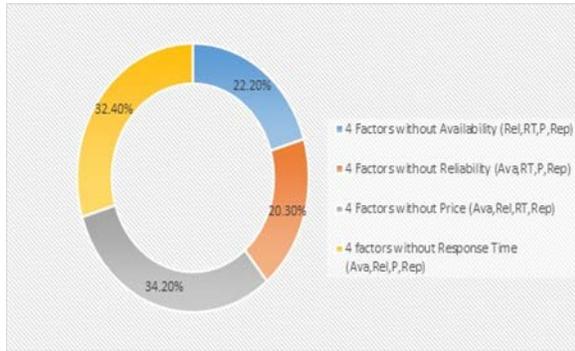
Fig. 14: Experiment 1 (Set 3). The null ratio in all cases
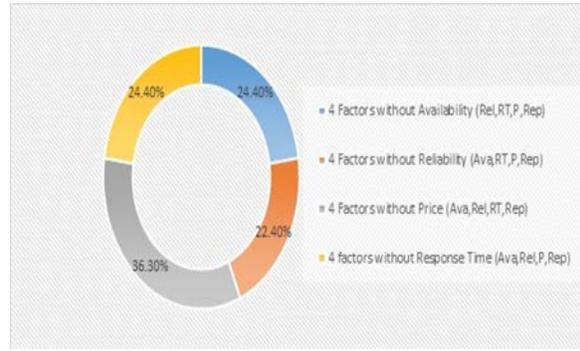
Fig. 15: Experiment 1 (Set 4). The null ratio in all cases.
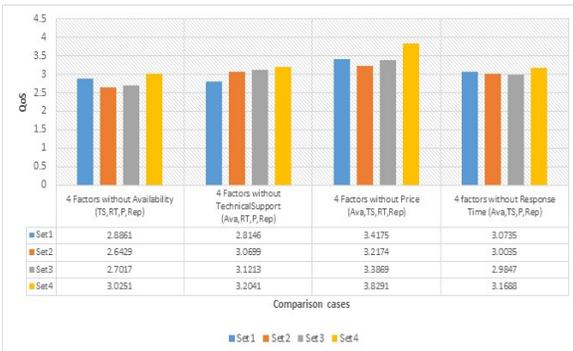
Fig. 16: Experiment 2. The differences in QoS for each set in all cases in the absence one of the four factors.
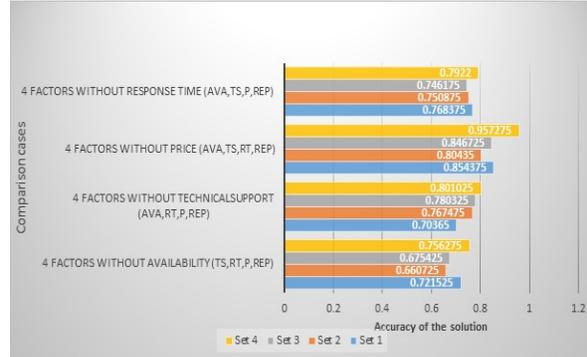
Fig. 17: Experiment 2. The accuracy of the solution path for each set in all cases in the absence one of the four factors.
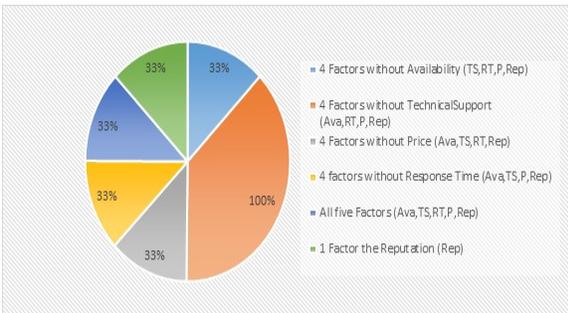

Fig. 18: Experiment 2 (Set 1). The Matching ratio of the paths in all cases with the path in case 1 "4 Factors without Reputation (Ava, TS, RT, P)".
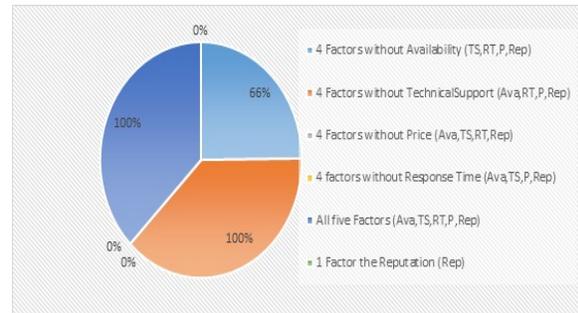

Fig. 19: Experiment 2 (Set 2). The Matching ratio of the paths in all cases with the path in case 1 "4 Factors without Reputation (Ava, TS, RT, P)".
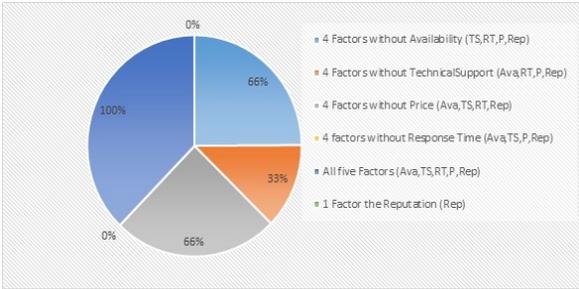
Fig. 20: Experiment 2 (Set 3). The Matching ratio of the paths in all paths in all cases with the path in case 1 "4 Factors without Reputation (Ava, TS, RT, P)".
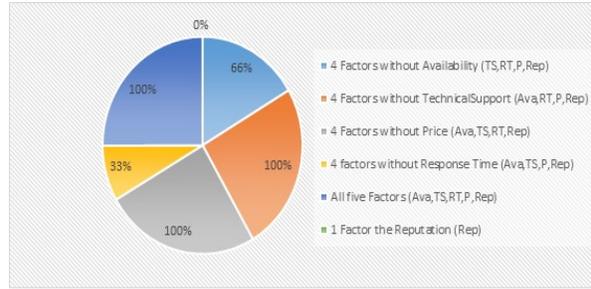

Fig. 21: Experiment 2 (Set 4). The Matching ratio of the cases with the path in case 1 "4 Factors without Reputation (Ava, TS, RT, P)".
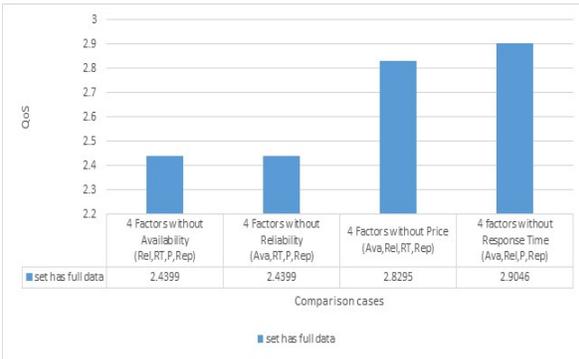

Fig. 22: Experiment 3. The differences in QoS for each set in all cases in the absence one of the four factors.
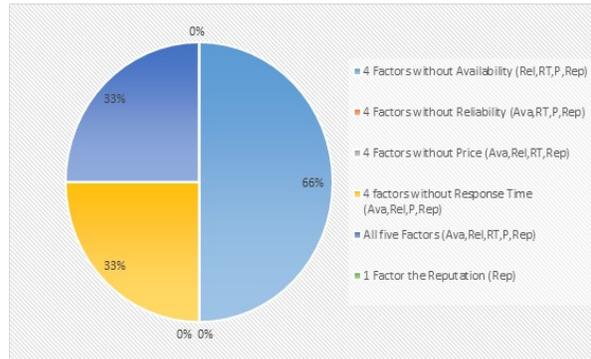

Fig. 23: Experiment 3. The Matching ratio of the paths in all cases with the path in case 1 "4 Factors without Reputation (Ava, Rel, RT, P)".
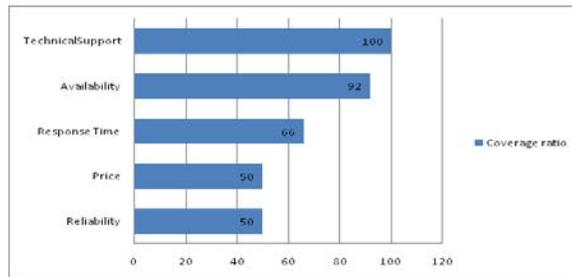

Fig. 24: The ratio covering the reputation factor for the other factors

Table 6: Covering the reputation factor for the other actors

Experiments	EXP-1				EXP-2				EXP-3	Summary
Set \ Factor	Set1	Set2	Set3	Set4	Set1	Set2	Set3	Set4	full Set	Coverage ratio
Availability	yes	yes	yes	no	yes	yes	yes	yes	yes	92%
Reliability	yes	yes	yes	yes	-	-	-	-	no	50%
Response Time	no	yes	no	yes	yes	no	no	yes	yes	66%
Price	yes	yes	yes	no	yes	no	yes	yes	no	50%
Technical Support	-	-	-	-	yes	yes	yes	yes	-	100%

Table 8: The results of multiple linear regression using training data

Experiments	Factors	Result	RSS	R^2
Ex1: multiple linear regression using all factors	X1=Ava, X2=Rel, X3=RT, X4=P, y=Rep	$Rep = -8.857560574 * 10^{-2}x_1 - 1.016724442 * 10^{-1}x_2 + 5.721720885 * 10^{-2}x_3 + 6.402648477 * 10^{-1}x_4 + 2.327258041$	8.5907	$5.266868602 * 10^{-1}$
Ex2: multiple linear regression without reliability factor	X1=Ava, X3=RT, X4=P, Rep	$Rep = -1.057107003 * 10^{-1}x1 + 1.995351246 * 10^{-2}x3 + 6.353639218 * 10^{-1}x4 + 2.147311612$	8.8469	$5.125741727 * 10^{-1}$

As shown in Table (VII) the Residual Sum of Squares (RSS) for Ex1 is more accurate than Ex2, and the reason is that the RSS in the first experiment is lower and this means it is fit based on the used data. The second type of interpolation is multiple polynomial regression and Table (VIII) represents the results of the multiple polynomial regression for two experiments. As shown in Table (VIII) the results were more accurate when taking reliability factor into account. Figure (27) represents a comparison between the experiments that used multiple linear regression and the experiments that used multiple polynomial regression. The result showed higher confidence when we used multiple polynomial regression, where the Residual Sum of Squares was less than the multiple linear regression. So the multiple polynomial regression using all factors is the most accurate because it has the least of the residual sum of squares that equals 6.0027.

6.3 Results of association with reputation using ANOVA test

We conducted ANOVA test to show if there are any statistical differences between the reputation factor and the other four factors. The association between reputation and the four other factors is presented in Table (IX) and in Figure (30). We can conclude that there is a significant association between reputation and (Ava, RT, and P), but it is not significant for reliability. It is also noted in Figure (28) that the f-ration of the reliability factor is the least compared with the other factors. Figure (29) represents the P-value for each factor, and as shown in the figure the P-value for reliability is 0.25123 which is higher than 0.05 and this means there is no an association between reputation factor and reliability factor, which may be explained by the fact that the data of reliability is not real because we obtained some of the reliability data from the website to solve the null problem in the used dataset (Cloud Armo).
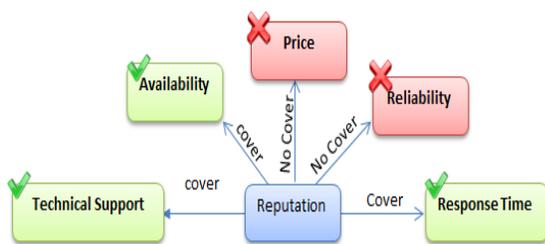

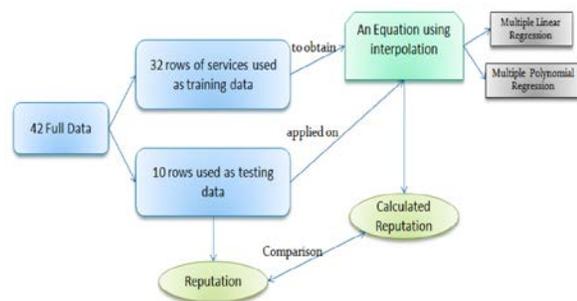**Fig. 25: The Covering reputation factor for the other factors**

Fig. 26: The proposed model of prediction of reputation factor based on the other factors
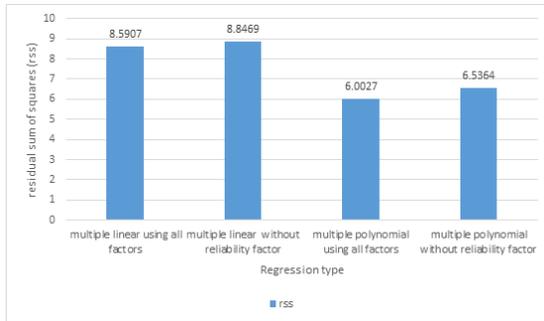
Fig. 27: The residual sum of squares (ss) for multiple linear regression and multiple polynomial regression
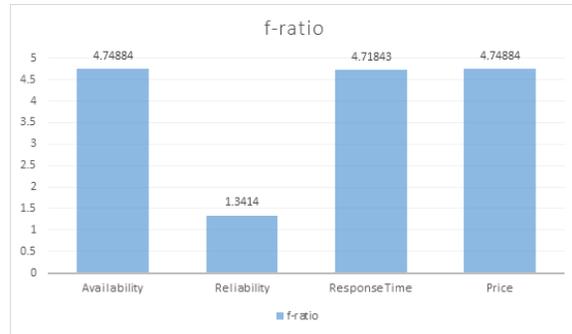

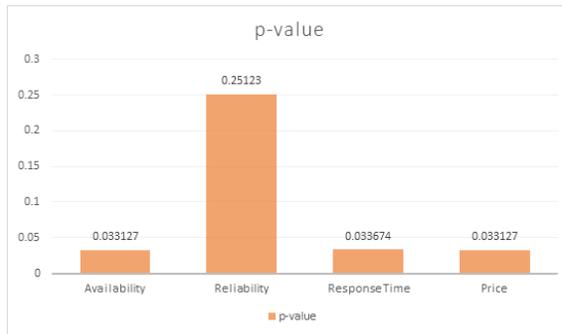Fig. 28: The value of f-ration for each factor
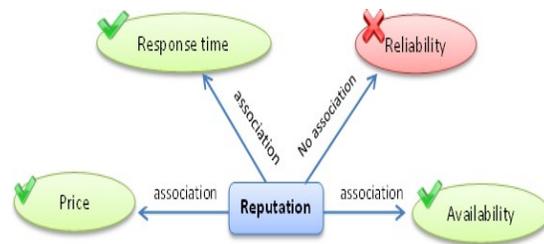

Fig. 29: The P-value for each factor

Fig. 30: The association factors with the reputation

Table 8: The results of multiple polynomial regression using training data

Experimens	Factors	Result	RSS	R^2
Ex1: multiple poly-nomial regression using all factors	X1=Ava, X2= Rel, X3=RT, X4=P, y=Rep	$Rep = 5.613808849 * 10^{-1}x_1^2 - 6.225172731 * 10^{-2}x_1x_2 - 7.40709063 * 10^{-2}x_1x_3 - 1.628093372 * 10^{-1}x_1x_4 + 9.3892666 * 10^{-3}x_2^2 - 5.835160577 * 10^{-2}x_2x_3 + 1.563231025 * 10^{-1}x_2x_4 + 9.162062868 * 10^{-2}x_3^2 - 0.41956877x_3x_4 - 0.227524679x_4^2 - 3.283535905x_1 + 1.516604158 * 10^{-2}x_2 + 1.527622797x_3 + 4.145155768x_4 - 2.490171441$	6.0027	$6.693 * 10^{-1}$
Ex2: multiple poly-nomial regression without reliability factor	X1=Ava, X3=RT, X4=P, y=Rep	$Rep = 0.684198347x_1^2 - 4.474802049 * 10^{-1}x_1x_3 - 7.548422255 * 10^{-1}x_1x_4 + 5.641417989 * 10^{-2}x_3^2 + 4.324633658 * 10^{-1}x_2x_4 - 4.624788122 * 10^{-2}x_4^2 - 0.62544914x_1 - 4.202647049 * 10^{-1}x_3 + 2.353277729x_4 + 1.883757923 * 10^{-1}$	6.5364	$6.398 * 10^{-1}$

Table 9: The summary of results of association with reputation using ANOVA test

Factors	f-ratio (F-Statistics)	p-value	significant at p <0.05
Availability	4.74884	0.033127 (less than 0.05)	significant
Reliability	1.3414	0.25123 (greater than 0.05)	not significant
Response Time	4.71843	0.033674 (less than 0.05)	significant
Price	4.74884	0.033127 (less than 0.05)	significant

7   Conclusion

Web services have received great attention because they support enterprises and business-to-business application, but when there is a large number of web services available on the repository service, it is not easy to find an execution path for composite service. In this study, we studied the possibility of using the reputation factor to cover for the other factors in the absence of one or more of the QoS factors. We used the Cloud Armor dataset. The dataset contains approximately 10,000 real feedbacks by 7,000 consumers for 113 real-world cloud service, and the feedbacks are based on QoS attributes. We supposed that the composite service contains three tasks, and we divided the dataset into four sets and each set has three tasks, and each task has about 9 candidate services that have four real OoS properties (Ava, Rel, RT, P). We built a simulation program in Java language to evaluate reputation for the web services based on trust result in the dataset, and we used the reputation algorithm proposed by the El-Kafrawy for updating reputation of the web services based on the trust factors of the consumers and reputation threshold. Then we obtained a reputation value for all web services located in the dataset.

We used Java program to conduct the experiment three times to study the effect of using the reputation factor in case of absence of one of the other factors and the possibility of covering this absence. The first experiment was conducted using the data of four factors (Ava, Rel, RT, P) based on the dataset, in the second experiment, we replaced the reliability factor with the technical support factor due to the lack of sufficient information on the reliability factor in the dataset. The third experiment was conducted on the services that have full data information. In each experiment, we suggested 7 cases for selecting the optimal composite path using bee algorithm, and we conducted a set of comparisons to show the possibility of covering reputation factor the absence of other factors.

The results proved that the reputation factor could cover some factors such as availability, response time and technical support. In addition, we represented a prediction of reputation factor based on the other factors using the multiple linear regression and polynomial regression, and the result had higher confidence when we used multiple polynomial regression, where the RSS was less than the multiple linear regression. We used ANOVA test to study the association of reputation with the other factors, and the results show a significant association between reputation and (Availability, Response Time, and Price), but it is not significant for reliability.

REFERENCES

[1]   Thomas Erl. Service-oriented architecture: concepts, technology, and design. Pearson Education India, 2005.

[2]   Jeff Offutt and Wuzhi Xu. Generating test cases for web services using data perturbation. ACM SIGSOFT Software Engineering Notes, 29(5):1–10, 2004.

[3]   Liangzhao Zeng, Boualem Benatallah, Anne HH Ngu, Marlon Dumas, Jayant Kalagnanam, and Henry Chang.QoS-aware middleware for web services composition. IEEE Transactions on software engineering, 30(5):311– 327, 2004.

[4]   Shuping Ran. A model for web services discovery with QoS. ACM Sigecom exchanges, 4(1):1–10, 2003.

[5]   Harry Katzan Jr. The education value of cloud computing. Contemporary Issues in Education Research, 3(7):37, 2010.

[6]   Omar Ali, Jeffrey Soar, Jianming Yong, et al. An investigation of the main factors to be considered in cloud computing adoption in Australian regional local councils. Journal of Contemporary Issues in Business and Government, 21(1):72, 2015.

[7]   Shuai Ding, Zeyuan Wang, Desheng Wu, and David L Olson. Utilizing customer satisfaction in ranking prediction for personalized cloud service selection. Decision Support Systems, 93:1–10, 2017.

[8]   Mohamad Ibrahim Ladan. Web services metrics: A survey and a classification. Journal of Communication and Computer, 9(7):824–829, 2012.

[9]   Andreea-Paula Jipa Abstract Eldorina-Andreea Alergus. Web service composition. 2008.

[10]   Recommendation ITU-T E.800. Overall network operation, telephone service, service operation, and human factors. Page 1, 2007.

[11]   Erik Christensen, Francisco Curbera, Greg Meredith, Sanjiva Weerawarana, et al. Web services description language (WSDL) 1.1, 2001.

[12]   Dimitris Gouscos, Manolis Kalikakis, and Panagiotis Georgiadis. An approach to modeling web service QoS and provision price. In Web Information Systems Engineering Workshops, 2003. Proceedings. Fourth International Conference on, pages 121–130. IEEE, 2003.

[13]   Min Tian, Andreas Gramm, Tomasz Naumowicz, Hartmut Ritter, and JS Freie. A concept for QoS integration in web services. In Web Information Systems Engineering Workshops, 2003. Proceedings. Fourth International Conference on, pages 149–155. IEEE, 2003.

[14]   Jorge Cardoso, Amit Sheth, John Miller, Jonathan Arnold, and Krys Kochut. Quality of service for workflows and web service processes. Web Semantics: Science, Services and Agents on the World Wide Web, 1(3): 281–308, 2004.

[15]   Michael C Jaeger, Gero Muhl, and Sebastian Golze. QoS- aware composition of web services: a look at selection algorithms. In Web Services, 2005. ICWS 2005. Proceedings. 2005 IEEE International Conference on. IEEE, 2005.

[16]   Changlin Wan, Carsten Ullrich, Limin Chen, Rui Huang, Jiewen Luo, and Zhongzhi Shi. On solving QoS-aware service selection problem with service composition. In Grid and Cooperative Computing, 2008. GCC'08. Seventh International Conference on, pages 467–474. IEEE, 2008.

[17]   Hiroshi Wada, Paskorn Champrasert, Junichi Suzuki, and Katsuya Oba. Multiobjective optimization of SLA-aware service composition. In Services-Part I, 2008. IEEE Congress on, pages 368–375. IEEE, 2008.

[18]   Rui Zhang, Chin Choy Chai, and Ying-Chang Liang. Joint beamforming and power control for multiantenna relay broadcast channel with QoS constraints. IEEE Transactions on Signal Processing, 57(2):726–737, 2009.

[19]   Zhenqiu Huang, Wei Jiang, Songlin Hu, and Zhiyong Liu. Effective pruning algorithm for QoS-aware service composition. In Commerce and Enterprise Computing, 2009. CEC'09. IEEE Conference on, pages 519–522. IEEE, 2009.

[20]   Bassam Al-Shargabi, Omar Sabri, and Alia Sabri. Toward a user preferences agent-based web service composition. International Journal of Information Technology & Computer Science (IJITCS), 14(2):1–5, 2014.

[21]   Haiteng Zhang, Zhiqing Shao, Hong Zheng, and Jie Zhai. Web service reputation evaluation based on QoS measurement. The Scientific World Journal, 2014, 2014.

[22]   Nianhua Yang, Xin Chen, and Huiqun Yu. A reputation evaluation technique for web services. International Journal of Security and Its Applications, 6(2):329–334, 2012.

[23]   Shangguang Wang, Zibin Zheng, Qibo Sun, Hua Zou, and Fangchun Yang. Evaluating feedback ratings for measuring reputation of web services. In Services Computing (SCC), 2011 IEEE International Conference on, pages 192–199. IEEE, 2011.

[24]   Surya Nepal, Wanita Sherchan, Jonathon Hunklinger, and Athman Bouguettaya. A fuzzy trust management framework for service web. In ICWS, pages 321–328, 2010.

[25]   M Sathya, M Swarnamugi, P Dhavachelvan, and G Sureshkumar. Evaluation of QoS based web-service selection techniques for service composition. International Journal of Software Engineering, 1(5):73–90, 2010.

[26]   Audun Jøsang, Touhid Bhuiyan, Yue Xu, and Clive Cox. Combining trust and reputation management for web-based services. In International Conference on Trust, Privacy and Security in Digital Business, pages 90–99. Springer, 2008.

[27]   Andriy Mnih and Ruslan R Salakhutdinov. Probabilistic matrix factorization. In Advances in neural information processing systems, pages 1257–1264, 2008.

[28]   Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. Computer, 42(8), 2009.

[29]   Zibin Zheng and Michael R Lyu. Personalized reliability prediction of web services. ACM Transactions on Software Engineering and Methodology (TOSEM), 22(2):12, 2013.

[30]   Weiwei Qiu, Zibin Zheng, Xinyu Wang, Xiaohu Yang, and Michael R Lyu. Reputation-aware QoS value prediction of web services. In Services Computing (SCC), 2013 IEEE International Conference on, pages 41–48. IEEE, 2013.

[31]   Mingdong Tang, Yu Xu, Jianxun Liu, Zibin Zheng, and Xiaoqing Liu. Combining global and local trust for service recommendation. In Web Services (ICWS), 2014 IEEE International Conference on, pages 305–312. IEEE, 2014.

[32]   Jianlong Xu, Zibin Zheng, and Michael R Lyu. Web service personalized quality of service prediction via reputation-based matrix factorization. IEEE transactions on reliability, 65(1):28–37, 2016.

[33]   Passent El-Kafrawy, Emad Elabd, and Hanaa Fathi. A trustworthy reputation approach for web service discovery. Procedia Computer Science, 65:572–581, 2015.

[34]   Hussein Al-Helal and Rose Gamble. Introducing replace-ability into web service composition. IEEE Transactions on Services Computing, 7(2):198–209, 2014.

[35]   Yiwen Zhang, Guangming Cui, Shu Zhao, and Jie Tang. IFOA4WSC: a quick and effective algorithm for QoS-aware service composition. International Journal of Web and Grid Services, 12(1):81–108, 2016.

[36]   Cloud Armor Dataset, school of computer science, University of Adelaide. https://cs.adelaide.edu.au/~cloudarmor/ds.html. Accessed 18 Oct. 2016.

[37]   Jon Jacobi, Online backup, PCWorld. https://www.pcworld.com/article/3211435/data-center-cloud/best-online-backup.html. Accessed 15 May. 2018.

[38]   Web   Hosting   Reviews   this   site   is   powered   by   host   gator. http://www.webhostingreviews.com/customize.htm. Accessed 15 April. 2017.