An Agent and multi-criteria methods based approach for Web service selection

1st Houcine BELOUAAR
Computer science department
LINFI Laboratory
Biskra University, Algeria
belouaar.houcine@gmail.com

2nd Okba KAZAR
Computer science department
LINFI Laboratory
Biskra University , Algeria
kazarokba@gmail.com

3rd Nadia Kabachi
Computer science department
ERIC Laboratory
Lyon1 University, France
nadia.kabachi@univ-lyon1.fr

***Abstract*—With the significant growth of the number of services on the Internet, the selection of a web service among several similar services has become a very delicate task for the web services client. In this paper, we propose an approach based on the MCDM methods for selecting the best services. Our approach is composed of two main steps, i) we use the AHP method to assign weights to each non-functional property that qualify a Web service. ii) We use these weights to rank the list of the similar services in order to select the best one. For ranking services we use the TOPSIS method. We implemented our approach as a distributed solution based on a set of Agents that interact together in order to obtain the best desired service as fast as possible. The use of Agent allows to improve the efficiency of the proposed approach. To demonstrate the applicability of our approach we have used a set of real services related to SMS sending. The obtained results shows the efficiency of our approach.**

***Index Terms*—Web Service Selection, QoS, AHP, TOPSIS, Service Ranking, Multi-Agent System.**

I. Introduction

Nowadays, Web services have become indispensable in almost every field. They constitute a category of web solutions used in industries, such as banking, shopping, and communications. They can be described simply as any service offered on the Web. For example, in a weather web application, a user fills the city name, the website sends the weather forecast for that city as response.

Web services are the most appropriate technologies for implementing the Service Oriented Architecture (SOA). SOA is an architectural style for building service-oriented solutions [1]. It is composed of three participants [2]: i) Service Provider which develops the services and makes them available on the Internet for users. ii) Service Registry: it is a centralized location in which the interfaces of the services are published by the providers. iii) Service Requester (sometime called service client) which searches services in the service registry and consumes them by sending requests to the service providers. Web services are based on standard technologies and protocols (based on XML) that allow applications to communicate remotely via the Internet, regardless of the platforms and languages on which they are based. They are widely used by companies, which allow them to expose a number of services and exchange information between them.

Due to the great success of web services the number of deployed web services in the web is growing fast. A simple request from a client can have hundreds (or thousands) of similar services as result. As consequence, the client has a serious issue for selecting the "best" service among the returned list of services. We mean by "best" service, the service that respond to the client requirements. In fact, the Quality of Service (QoS) is considered one of the most important metrics for distinguishing between services. In general, the returned services are not sorted based on QoS, but they sorted based on their registration date in UDDI registry [3]. In the literature, several solutions have been proposed to deal with the problem of selecting the best services. Some of these solutions are based on i) syntax/semantic based approaches [4] [6], ii) user profile based approaches [7], iii) genetic algorithm based approach [8] iv) QoS based approaches [5]. Generally, the QoS based approaches are most popular in the industry. The QoS can be defined as the description or measurement of the overall performance of a service by taking into consideration the non-functional properties such as: reliability, availability, response time,...etc [9]. A web service consists of a functional part that describes what the service must do and a non-functional part that describes how the service can do it.

Besides, selecting services based on QoS is like solving a multi-criteria problem by searching a service that meets several non functional properties. Multiple Criteria Decision Making (MCDM) is a family of multi-criteria decision analysis methods. We can cite for example, Elimination and Choice Translating algorithm (ELECTRE) [10], Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) [11], AHP [12],...etc.

In this paper, we propose an approach based on the MCDM methods for selecting the best services. Our approach is composed of two main steps: First, we assign weights to each non-functional property that qualify a Web service. We use in this step the AHP method. Second, we use these weights to rank the list of the similar services in order to select the best one. For ranking services we use the TOPSIS method. As the number of services in the Internet becomes very high, the process of weighting and ranking services takes long time. But, the user wants to get an answer to his request in minimal time. So, we propose to implement our approach as a distributed solution based on a set of Agents that interact together in order to obtain the best desired service as fast as possible.

Agents are software entities that interact with an environment and are likely to change and evolve based on external and internal stimuli, the latter being due to the proactive and deliberative capabilities of the Agents themselves [14]. We provide in this work a collaborative solution in which, each Agent is responsible of a unique task in our system. In this way, we accelerate the processing of the user request. In order to evaluate our approach we show its applicability on an example of selecting the best service among a set of real services related to SMS sending web services.

This paper is organized as follows: Section II presents some related work. Section III describes our proposed approach. In the Sections IV and V we present how we applied the AHP and TOPSIS methods. Before concluding we present in Section VI the obtained experimentation results. Finally, Section VII presents a conclusion and future directions of this work.

II. Related Work

In [15], the authors proposed a web service QoS Manager module responsible for measuring QoS information for the collected Web services, and information is stored in the Web Service Storage.

In the work proposed in [13], the author has added a new component called OoSBF (Quality of Service Bootstrapping Framework) which considers the need for QoS qualification integration when publishing a new Service and before it is used.

QoS is an important factor for the evaluation of distributed paradigms, indeed several studies mention that the justification of QoS is done at the time of the selection of service or during execution. Several quality services are mentioned in this work, we can cite: Latency, Execution time, Response time, Throughput, Availability, Reliability, and Accessibility.

In the work proposed in [16], the author added to the UDDI register a new component called MEC, which deals with the evaluation of possible components in order to help the applicant of the service in question to choose the Web service to invoke. It confirms that, in order to carry out a better selection of web services, it was necessary to use the different non-functional criteria instead of the functional criteria since in general they are the same functionality and since this is not a single objective solution, the author proposed a multi-criteria evaluation.

In [17], we have proposed a web service selection approach based on the TOPSIS method. This approach has been tested by a first step experiment with four quality criteria and in the second part by six quality criteria: response time, reliability, documentation, security, precision and cost. The entries of this proposal being the decision matrix including the weights of each criterion. These weights, which play a very decisive role in the results obtained by TOPSIS, are captured by the system designer. In [17], we did not take into consideration the weights relating to the criteria, these weights are filled manually by the designer.

III. Proposed Approach

In this section, we present our approach which is depicted in the Figure 1. In this figure, we show the life cycle of a user request for a given Web service. The request is a keyword with a set of criteria. First, in our approach the provider publishes a set of Web services in the service registry (UDDI). It associates with each service description a set of QoS information such as availability, reliability, etc. (A.1 in the figure). The UDDI saves the QoS in a local data base (see A.2). Second, the `Requester Agent` receives requests from the service client. Hence, the `Requester Agent` sends the Web service requests with the desired QoS to `Broker Agent` (B.1 in the Figure). The latter, forwards these requests to the service registry in order to find all the services that answer to the request and the desired QoS (see B.2). Once the `Broker Agent` receives the list of similar services (B.3 and B.4 in the Figure), it interacts with `Weighting Agent` in order to calculate the weights that are related to the desired QoS criteria (see B.5). In this step, we use AHP method to calculate such weights. The obtained list of weighted services is sent to the `Ranking Agent` (see B.6). The `Ranking Agent` classifies this list of weighted services by assigning scores to each one. We use for this, the TOPSIS method. In this way, we can obtain the best service based on the assigned scores. Finally, the `Ranking Agent` sends the best service to the `Requester Agent` which is an answer to the service client request.

The two used methods (AHP and TOPSIS) in our approach are considered as the most popular methods for multi-criteria decision methods.

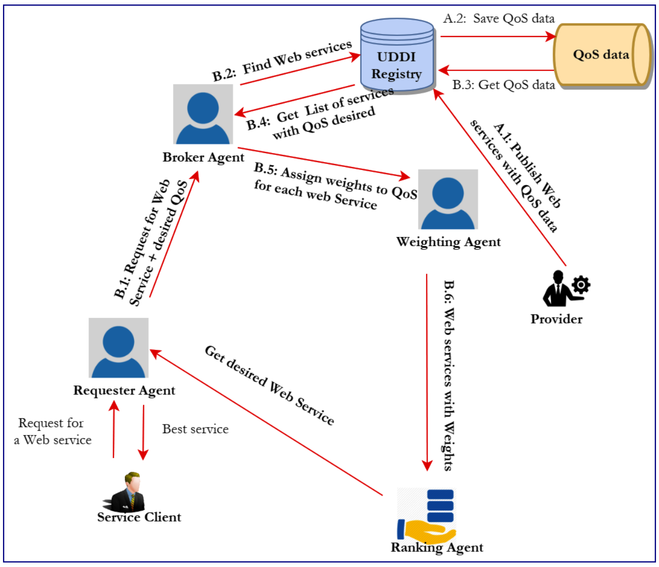

Fig. 1. Proposed approach

A. Description of Agents

This section describes each agent in our approach.

- `Requester Agent`: is responsible of receiving requests from service client and forward them to `Broker`

Agent. It contains two modules: i) An interface module which allows to other Agents to interact with this Agent. ii) Communication module: it is a module responsible of the treatment the messages exchanged between the current Agent and the rest of the Agents.

- Broker Agent: It deals with the client requests for searching the desired services in the service registry. The returned services are sent to the Weighting Agent. It contains a communication module; it saves the similar web services in a local database.
- Weighting Agent: It has a processing module which is responsible of calculating the weight of the QoS criteria for each service that is received from the Broker Agent. To do so, it uses the AHP method.
- Ranking Agent: It allows to rank the web services that are received from the Weighting Agent. After that, it provides the best service according the client requirements. This Agent uses the TOPSIS method for the service ranking.

B. Functional description of the approach

Here, we explain how the agents interact together in order to give an answer (best service) to the service client. As explained above the service provider has to publish a set of services in the service registry. The client sends to the system a request containing functional and non functional criteria. The Requester Agent receives the request and start communicates with the other agents following the sequence diagram presented in the Figure 2.

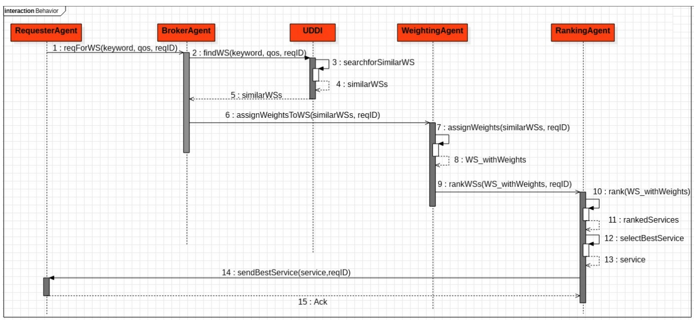

Fig. 2. Functional description of the approach

In this figure, the Requester Agent sends an asynchronous message to the Broker Agent to find services with a desired QoS. The latter, sends a synchronous message to the UDDI. As a result Broker Agent receives a list of similar services which is forwarded to the Weighting Agent as an asynchronous message. This Agent weights the services and sends an asynchronous message to the Ranking Agent to classify them based on the calculated scores. The best service is sent as an asynchronous message to the Requester Agent.

IV. Weighting of QoS Criteria using AHP

In this section, we present how we have applied the AHP method to designate weights related to quality criteria. First, the quality of service is defined in our work as a combination of several criteria which may be quantitative such as response time, availability, reliability, etc. or qualitative such as usability, accessibility, etc. These parameters can then be considered as criteria of choice between several discovered web services that respect the client requirements. In this work, we have selected four criteria:

- **Reliability:** is the ability of a service to perform the required functions under specified conditions for a given period of time [21].
- **Throughput:** is the number of completed service requests over a time period [22].
- **Response time:** is the time duration between a service user sending a request and receiving a response [23].
- **Cost:** is the price to be spent to execute a web service, this cost may be provided by the service provider [5].

The AHP uses a set of numerical rates, where each rate is related to a given preference (see Table I).

TABLE I
Preference Scale [18]

Verbal Judgment of Preference	**Numerical Rating**
Extremely preferred	9
Very strongly to extremely preferred	8
Very strongly preferred	7
Strongly to very strongly preferred	6
Strongly preferred	5
Moderately to strongly preferred	4
Moderately preferred	3
Equally to moderately preferred	2
Equally preferred	1

The following three-step procedure provides a good approximation of the synthesized priorities [18]:

- **Step 1:** Sum the values in each column of the pairwise comparison matrix.
- **Step 2:** Divide each element in the pairwise matrix by its column total.
- **Step 3:** Compute the average of the elements in each row of the normalized matrix.

For example, Table II and Table III show how the AHP method is applied to obtain weight for the QoS criteria: Cost, Response time (Rt), Reliability (Rel), and Throughput (Th).

TABLE II
Comparaison matrix

	Cost	Rt	Rel	Th
Cost	1	0,3333	0,2	0,1429
Rt	3	1	0,5	0,1667
Rel	5	2	1	0,25
Th	7	6	4	1
Sum	16	9,3333	5,7	1,5595

TABLE III
WEIGHTS CRITERIA

					Weights
Cost	0,063	0,036	0,035	0,092	0,056
Rt	0,188	0,107	0,088	0,107	0,122
Rel	0,313	0,214	0,175	0,16	0,216
Th	0,438	0,643	0,702	0,641	0,606
Checksum	0,563	0,357	0,298	0,359	1

V. WEB SERVICES RANKING WITH TOPSIS METHOD

The TOPSIS is a method whose purpose is to be able to classify in order of choice a certain number of alternatives on the basis of a set of favorable and unfavorable criteria. It is part of the techniques used in the field of multi-criteria decision-making. It was developed by Hwang and Yoon in 1981 as an alternative to the ELECTRE method and can be considered as one of its most widely accepted variants.

The ranking of alternatives in TOPSIS is based on "similarity to the ideal solution", which avoids having the same resemblance to the positive ideal and the negative ideal [19] [20]. The different steps of the TOPSIS method are described in [17].

VI. EXPERIMENT

In this section we show how we apply our approach for on a real example. We show how to select the best service among a set of real services related to SMS sending web services. The used similar services are as follows: SMSWS, SMS, SendMessages, SendSMS, and SendSMSWorld. Each Web service is characterized by four criteria: reliability(Rel), throughput(Th), Cost and response time(Rt). The reliability and throughput are favorable criteria and the two other criteria are not favorable.

The assigned values for the web service criteria are used to create a decision matrix. For our example the obtained matrix is presented as follows:

$$\begin{array}{l} \\ SMSWS \\ SMS \\ SendMessages \\ SendSMS \\ SendSMSWorld \end{array} \begin{array}{c} \begin{array}{cccc} Rel & Th & Cost & Rt \end{array} \\ \begin{pmatrix} 79,3 & 6,3 & 11 & 751 \\ 81 & 5,2 & 2 & 113,8 \\ 53,6 & 4,5 & 9 & 291,07 \\ 67 & 6,8 & 3 & 1308 \\ 64,3 & 5,3 & 4 & 3103 \end{pmatrix} \end{array}$$

The application of the AHP method gives for each criterion the weights that are depicted in Table IV:

TABLE IV
PREFERENCE SCALE

Criteria	Cost	response time	Reliability	Throughput
Weight	0,056	0,122	0,216	0,606

These weights are the inputs for the `Ranking Agent` that uses the TOPSIS method. The TOPSIS method start by creating a decision matrix (presented above) which contains the Web services QoS values. This matrix is normalized as in Table V.

TABLE V
NORMALIZED MATRIX

Services/ Criteria	Rel	Th	Cost	Rt
SMSWS	0,508	0,496	0,724	0,217
SmsWS	0,519	0,409	0,132	0,033
SendMessage	0,344	0,354	0,592	0,084
SendSMS	0,429	0,535	0,197	0,378
SendSMSWord	0,412	0,417	0,263	0,896

The normalized matrix is used with previous weights in order to build the weighted normalized matrix. The obtained Matrix is depicted in Table VI.

TABLE VI
WEIGHTED NORMALIZED MATRIX

Alternative	Rel	Th	Cost	Rt
SMSWS	0,110	0,301	0,041	0,026
SmSWS	0,112	0,248	0,007	0,004
SendMessage	0,074	0,215	0,033	0,010
SendSMS	0,093	0,324	0,011	0,046
SendSMSWord	0,089	0,253	0,015	0,109

By applying the final step in our approach, we obtain the scores for each Web service (see Table VII). The best service is the one with the highest score. So in our example, the SendSMS is the best recommended service to the client.

TABLE VII
WEB SERVICE RANKING

Alternative	Score	order of choice
SMSWs	0,72737	2
SmsWs	0,61391	3
SendMessage	0,45466	4
SendSMS	0,73874	**1**
SendSMSWord	0,27189	5

VII. CONCLUSION

In this work, we proposed a approach based on selecting the best web service. Since the criteria weights have a large influence on the choice of web services to be provided to the user, we have used the AHP method to evaluate the weights instead of grabbing them manually by the designer. We have also used the TOPSIS method as a MCDM method to solve the problem Web service ranking. The use of Agents in the web services selection allows us to improve the efficiency of our approach.

In addition, in order to demonstrate the feasibility of our approach we have used a real example of Web services have a set of QoS values. The results shows that our approach can select the best and closest candidate services.

In a future work we plan to deal with the following aspects:

- improve the selection process by introducing other quality of service measures;
- improve the efficiency of the proposed approach by using the power of cloud computing technology.
- integrate the fuzzy logic in the web service selection. Where, the client request includes not only the discrete values of QoS but, also linguistic terms such as: excellent, good, and medium services.

REFERENCES

[1] Kayastha, Ronak R., and Jwalant Baria. "A survey on web service selection and ranking methods." International Journal of Computer Applications 117.16 (2015).

[2] Negi, Neerja, and Satish Chandra. "Web service selection on the basis of QoS parameter." 2014 Seventh International Conference on Contemporary Computing (IC3). IEEE, 2014.

[3] Chakhar, Salem, et al. "Multicriteria evaluation-based conceptual framework for composite Web service selection." Lamsade, University Paris Dauphine, France, Research Report (2011).

[4] Guo, Guangjun, et al. "A Method for Semantic Web Service Selection Based on QoS Ontology." JCP 6.2 (2011): 377-386.

[5] Liu, Yutu, Anne H. Ngu, and Liang Z. Zeng. "QoS computation and policing in dynamic web service selection." Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters. ACM, 2004.

[6] Purohit, Lalit, and Sandeep Kumar. "Web Service Selection using Semantic Matching." Proceedings of the International Conference on Advances in Information Communication Technology & Computing. ACM, 2016.

[7] Shi, Chunqi, Donghui Lin, and Toru Ishida. "User-centered qos computation for web service selection." Web Services (ICWS), 2012 IEEE 19th International Conference on. IEEE, 2012.

[8] Zhu, Wenlong, et al. "An Approach to Web Services Selection for Multiple Users." IEEE Access 5 (2017): 15093-15104.

[9] Ran, Shuping. "A model for web services discovery with QoS." ACM Sigecom exchanges 4.1 (2003): 1-10.

[10] Yu, Xiaohan, et al. "ELECTRE methods in prioritized MCDM environment." Information Sciences 424 (2018): 301-316.

[11] Hwang, C.L., Yoon, K.P. "Multiple attributes decision making methods and applications". Berlin: Springer-Verlag; 1981.

[12] Saaty, Thomas L. "Analytic hierarchy process." Encyclopedia of operations research and management science. Springer, Boston, MA, 2013. 52-64.

[13] Aljazzaf, Zainab. "Bootstrapping quality of web services." Journal of King Saud University-Computer and Information Sciences 27.3 (2015): 323-333.

[14] Frcon, Louis, and Okba Kazar. Manuel d'intelligence artificielle. PPUR Presses polytechniques, 2009.

[15] Al-Masri, Eyhab, and Qusay H. Mahmoud. "Discovering the best web service: A neural network-based solution." Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on. IEEE, 2009.

[16] Garca-Cascales, M. Socorro, and M. Teresa Lamata. "On rank reversal and TOPSIS method." Mathematical and Computer Modelling 56.5-6 (2012): 123-132.

[17] Belouaar, Houcine, Okba Kazar, and Khaled Rezeg. "Web service selection based on TOPSIS algorithm." Mathematics and Information Technology (ICMIT), 2017 International Conference on. IEEE, 2017.

[18] Saaty, Thomas L. "The analytic hierarchy process: Decision making in complex environments." Quantitative Assessment in Arms Control. Springer, Boston, MA, 1984. 285-308.

[19] Wang, Hongbing, et al. "Web services: problems and future directions." Web Semantics: Science, Services and Agents on the World Wide Web 1.3 (2004): 309-320.

[20] Madi, Elissa Nadia, Jonathan M. Garibaldi, and Christian Wagner. "A comparison between two types of Fuzzy TOPSIS Method." Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on. IEEE, 2015.

[21] Kuyoro Shade, O., et al. "Quality of service (Qos) issues in web services." IJCSNS International Journal of Computer Science and Network Security 12.1 (2012): 94-97.

[22] Singh, Aditya Pratap. "QoS ISSUES IN WEB SERVICES." From The Desk of HOD 6.2 (2015): 27.

[23] Zheng, Zibin, and Michael R. Lyu. "Qos evaluation of web services." QoS Management of Web Services. Springer, Berlin, Heidelberg, 2013. 19-39.