Reducing the Calculations of Quality-Aware Web Services Composition Based on Parallel Skyline Service

Maryam Moradi

Department of Computer Engineering,
Yazd Branch, Islamic Azad University,
Yazd, Iran

Sima Emadi*

Department of Computer Engineering,
Yazd Branch, Islamic Azad University,
Yazd, Iran

Abstract—The perfect composition of atomic services to provide users with services through applying qualitative parameters is very important. As expected, web services with similar features lead to competition among the service providers. The key challenge to find an appropriate web service for composition occurs when multiple aspects of quality (such as response time, cost, etc.) are considered in the optimal composition of services. Skyline service provides the best service with consideration of qualitative parameters by using superior analysis. In this study, Skyline algorithm is used to find a set of best possible services compositions while taking into account qualitative parameters. The parallelism technique in this study, had significant impact on reducing response time and increasing the speed of service composition as well as reduction in composition calculations. The results of the analysis and evaluation of the proposed method shows optimum runtime and the best composition.

Keywords—service composition; parallel Skyline service; the dominant relationship; service quality

I. INTRODUCTION

In recent years, the application of web-based systems in institutions and government agencies is increasing. Introduction of web services is an effective approach in business structures to provide the required capabilities of service providers for services composition. Selection of the appropriate service on user's request is based on the service quality of services. Several different methods have been suggested to solve the problem of web services composition based on qualitative characteristics. These methods can be divided into two types of exact methods and approximate methods according to [2]. The first type is known as non-innovative methods which selects the best design from all available designs by examining and calculating the candidate's routes and thus provide a more precise answer. In the second type or innovative methods, contrary to the first type, an ideal design that is close to the best and most accurate answer will be chosen.

Due to the importance of optimal composition of web services in recent years, a lot of works have been done in the field of each method. By studying various types of innovative algorithm, one can concluded that many problems still exist to solve in web services composition based on qualitative characteristics. For example, each of these methods usually

have local optimality problem alone; or in basic genetic algorithm, crossover and mutation operation acts randomly and without any guidance, which leads to degeneration of the method. Therefore, efforts to improve efficiency such as using combined methods, operators like revolution operator or adding functions to improve were performed. These techniques are provided for better speed, faster convergence, and higher efficiency in large spaces. Based on the mentioned studies, there is no specific benchmark tool for evaluating the algorithm. Although some researchers used different simulation environments or different data to compare them with each other, the results show that different methods have different disadvantages and they do not have any specific standard. Skyline algorithm method and parallelism technique are used in this proposed method in order to provide the best composition with regard to the shortest response time in high scalability.

The paper is as follows; the second section expresses the concept of Skyline service. In the third section, related works in the field of service composition by Skyline service will be discussed. The proposed method is described in the fourth section. And the fifth section analysis and evaluation and the results of the proposed method are presented and in the final section, conclusions and recommendations for the future studies are presented.

II. SKYLINE SERVICE ALGORITHM

Skyline service includes a set of services that is not dominated by other services. In this regard, the concepts required are as follows:

A. Dominance Service and Skyline Service

In service composition, services that are better in all aspects of service quality compared to other services are dominance service.

For example, assume web service composition of travel assistance: This composition, for the user, provides travel assistance package. Examples of needed services include: travel planner, maps and weather forecasting. Weather conditions during travel are an important factor for weather service. A user enjoys a number of choices for each of these services. As noted, here limitations with similar capability arises which are discussed later. In Table 1, five maps

providers and four travel planner service providers are presented [2].

TABLE I. FIVE MAPS SERVICE PROVIDERS AND FOUR TRAVEL PLANNER SERVICE PROVIDERS [2]

Sid	Operation	Latency	Fee	Reputation
Map Providers				
A	Geo code	0.5	0.8	2
	Get map	1	0	2
B	Geo code	0.7	0.3	3
	Get map	2	0.5	3
C	Geo code	0.5	0.2	2
	Get map	1.5	0.9	2
D	Geo code	0.3	0.7	2
	Get map	1	0.4	2
E	Geo code	0.6	0.7	3
	Get map	0.8	0.5	3
Trip Planner Providers				
T	Search Trip	2	0	2
	Get Trip	2	0.8	2
G	Search Trip	1	0	3
	Get Trip	2	1	1
H	Search Trip	2	1	2
	Get Trip	3	1	2
I	Search Trip	3	1	3
	Get Trip	2	0	2

It should be noted that sometimes to request a web service, it requires to call a number of services. For example, to request maps provider, it requires two functions of geo code and get map. In addition, it is possible that dependency constraints exist between these two functions (like get map which is dependent to geo code). Therefore, these functions can be arranged in a sequence of dependency constraints; the production process of Skyline service composition by quality values of any services in Table 1 are described. According to Table 1, for calculation of quality for each service, records of the service must be combined. The results are shown in Table 2. Calculation of qualitative parameters is according to the equation (1):

Service name: ($\sum$ Operation, $\sum$ Latency, $\sum$ Fee, $\sum$ Reputation) [2]   *(1)*

TABLE II. SUMS OF EACH QUALITATIVE PARAMETER [2]

Sid	Latency	Fee	Reputation
Map Providers			
A	1.5	0.8	2
B	2.7	0.8	3
C	2	1.1	2
D	1.3	1.1	2
E	1.4	1.2	3
Trip Planner Providers			
F	4	0.8	2
G	3	1	2
H	5	2	2
I	5	1	3

By using Dominance Analysis, the best services from each group are chosen. As in each group (Map Provider: A, B, C, D, E) and (Trip Planner Provider: F, G, H, I), the lowest numerical value is selected as dominance or superior service. In the first group, A and D and in second group F and G, are selected as the dominance services. To understand better, the dominance service selection process for map provider is described. As

noted, the lowest value is selected as the best services to combine that have been selected at the first stage of existing services.

TABLE III. SELECTED NUMERICAL VALUES AT THE FIRST STAGE [2]

Sid	Latency	Fee	Reputation
Map Providers			
A	1.5	0.8	2
D	1.3	1.1	2
E	1.4	1.2	3

Then, according to the second column of qualitative parameters in Table 3, the best service is selected as the dominance service. Service selection process carried out in the second group is likewise.

TABLE IV. DOMINANCE SERVICES IN THE SECOND GROUP [2]

Sid	Latency	Fee	Reputation
Map Providers			
A	1.5	0.8	2
D	1.3	1.1	2

Selected services are the best ones among available services that ultimately will be combined. All possible compositions are shown in Table 5.

TABLE V. SKYLINE IMPLEMENTATION OF SERVICE COMPOSITION [2]

C-Sky	Latency	Fee	Reputation
C-Sky (A,F)	5.5	1.6	4
C-Sky (A,G)	4.5	1.8	4
C-Sky (D,F)	5.3	1.9	4
C-Sky (D,G)	4.3	2.1	4

B. Double Progressive Algorithm

The origin of this algorithm is progressive counting principle. In this principle. After making the root node which is the parent node, latter nodes are built upon it. The rule to create each node is that selected services available in composition are different only in one service with its child nodes. For example, the root node in Figure 1 is a1b1c1 its child nodes are (a1b2c1) (a1b1c2) (a2b1c1). The number of services composition produced of Skyline are calculated in form of $|A|*|B|*|C|=27$.

In this algorithm, all services of Skyline are stored in the memory. General space of services composition is proliferated based on the principle of progressive counting and using the lattice.
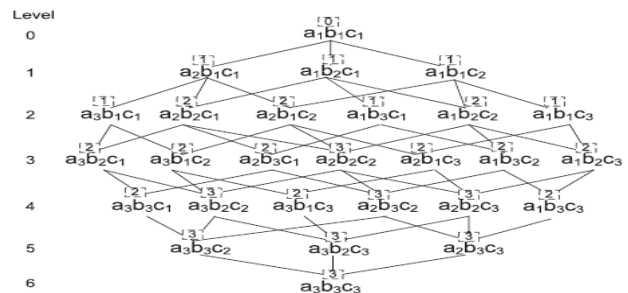

Fig. 1. Expansion lattice [3]

In Double Progressive, the algorithm sorts services composition in ascending order within an expansion lattice; as well as showing services composition of Skyline progressively [3].

It is noteworthy that in expansion lattice, a service composition that Ni (No. of composition) has higher ranking than its ancestor does not mean that Ni is dominated by its ancestor; Therefore, expansion lattice only determines the counting order between among and proves that Ni is counted after its ancestor. But for nodes that do not have parent-child relationship, such as: (a2, b1, c1) at level 1 and (a1, b3, c1) at level 2, a correct order must be ensured, since it is possible (a1, b3, c1) has less score than (a2, b1, c1); Hence, it should be considered earlier. To achieve the progressive counting, expansion lattice (T) with a heap (H) is used. Expansion lattice ensures that the parent node is considered before the child node, which is very desirable; because the score of a parent node cannot be greater than the score of child nodes. On the other hand, the heap determines the order of nodes without parent-child. The process of making compositions from the first level to the heap (H) begins with Composition Service Execution Plan (CSEP) 1. Each stage of the counting will have two sub-phases: 1. Extraction: a service composition with the lowest Ni rank, is extracted from the heap and compared with the existing Skyline. Finally, Ni will be entered into Skyline if it is not dominated and omitted [3].

C. Parent table

In progressive algorithm a node alone can be produced several times from the other parent nodes, where proliferation problem occurs because a node can have m parents, where m is number of Skylines' service, and similar child nodes can be produced when each of ancestors develop. As shown in Figure 1, number of every node shows its parent node number (index). For example, the nodes (a2, b2, c2) are placed three times in the heap, since it has three parents, and when each of them develop, (a2, b2, c2) will be produced and will be put into H. The nodes proliferation problem is associated with many computational problems, because many nodes are analyzed several times. This node can be used more than once in Skyline, causing a false Skyline [4].

The parent table is a suitable solution to solve the problem of nodes proliferation with minimal computations. Instead of considering all the ancestors, the parent table only stores the data of number of parents for a node. A basic rule is that a single node can only be placed in the heap, when all its parents are analyzed, and since the highest number of parents for a node is m, the parent table can at least be given a higher bit than number of parents in a node. This means that maximum of one bit more than the specified number of services be calculated. According to the expansion lattice, the following characteristics are followed:

Assuming the index of the S-Skyline starts with 1, the number of parents of each node is written above it.

The parent table, stores the number of parents in each node. In each comparison with another node, a figure of the number of parents will be decreased and the table will be updated with new values. Finally, each node reaches to zero, will be placed

into heap. The operation ensures that all child nodes are placed into heap before parent node [3].

One of the key features of the parent table is that a table should be stored in the heap which have the lowest possible size without a single node and the ancestors be symbiotic in the heap. Heap size is a determinant factor in the overall performance of the algorithm. It should be noted that other methods to avoid the proliferation of nodes can be created.

D. Button-Up Algorithm

DPA algorithm decides with two major functions:

(F1)Heap functions: verify that the child node entered into the heap before the parent node.

(F2)Skyline comparisons: choosing the best services through existing services and taking into account the dominant relationship.

BUA algorithm is a bottom to top computational framework and presents a linear comparison strategy for better performance and scalability.

This algorithm carries out optimization and cost calculations with positive traits inherited from DPA. The algorithm's strategy is to use linear compositions while doing comparisons to select the best composition. A linear composition is to compare the results of two nodes with the next node, and finally achieving the best possible composition [4]. Figure 2 displays schema of a linear composition.
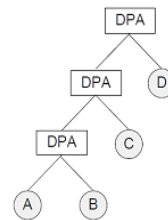

Fig. 2. linear composition [3]

III. RELATED STUDIES

Today, service composition is discussed as a key challenge in software systems. Using the appropriate method to combine services, provides favorable results than other existing approaches. It should be noted that by study and research in the service-oriented fields and service composition, new and optimal methods in this case are presented.

In this section, different methods of service composition are studied and presented and the best method is chosen among the existing methods.

Wu et al. used Dynamic Skyline Composition Algorithm for combining web services based on service quality, where the compositions of web-based services are performed dynamically. With the emergence of new services, old services will be removed and quality of service also changes. The advantages of this approach is to identify and select the best web services through a set of services based on service quality, and also using linear compositions to reduce the number of selected web services from the set of available services. The disadvantage of this study was lack of evaluating the algorithm

on a composition of web services in real and unreal data sets [5].

Benouaret et al. presented fuzzy dominance method for calculating Skyline service based on quality of service, in which key challenges including increasing number of web services and qualitative aspects have been considered. The users can enter their desired quality of service. The advantages of this new concept are α-dominance of Skyline service for composition of web-based services in high dimension that calculation in its composition has decreased. This method is also able to select the best services through quality services is undesirable. The disadvantages of this method, lack of awareness of sufficient information for the selected feature among Web services [6].

Benouaret et al. used fuzzy logic and Top-k algorithm to services' preferences and their compositions based on user requests. In this method, user preference is modeled based on fuzzy method, and the RDF is used to determine the relationship between the web services. The advantages of this study are improvements on diversity of web services compositions while maintaining service composition with the highest score. The disadvantages of this method is limits in fuzzy method in composition of services to match the user requests through using set of comparative methods [7].

In [8] ranking and clustering of web services is provided by using criteria of dominance relationship. In this research, retrieval, selection and composition of web services are done according to their increase and matching on the basis of criterion matching and using Skyline method among different qualitative parameters. The disadvantage of this method is lack of optimization at the time of service composition. In addition, reduced weight of service quality, results in decreased accuracy of retrieval services and thereby causes loss of important information. The advantage of this method is ranking suitable services on request which defines dominance relationships between services. The clustering provides prevailing interface between matched parameters. The proposed algorithms have been effective on the user's actual requests on true and false data.

Yu et al. presented the calculated Skyline service for invalid qualitative data, in which service composition in which the performances of services compositions are tested according to fluctuations in the dynamic environment. In this study, qualities of real services are provided by inherently invalid services. The optimization approach of available services may not change the quality of services over time. Therefore, multiple and sometimes conflicting qualitative criteria for the selection of each service requires using the weight characteristics. The disadvantage of this method is false understanding of weight characteristics which leads to loss of services desired by users. The advantages of this method are creating a new concept called p-dominant Skyline service that is suitable for weight characteristics and quantitative parameters. In addition, use of R-tree structure due to pruning dual structure, is effective in optimal composition web services in Skyline algorithm [9].

Alrifai et al. used Skyline service to reduce the number of candidate services composition. And by using K-means clustering algorithm, they have produced clusters of trees. This algorithm receives inputs of Skyline and returns the structure of the binary tree and defines the root. Indeed it can be said that it creates dominance relationship between the web services. And puts them in order based on the characteristics of services quality as these services belong to Skyline. The advantage of this method is removing invalid services and limitations of complex calculations and long processing time as they result in decreased efficiency [10].

IV. THE PROPOSED APPROACH

As mentioned in the previous section, Skyline was introduced as an appropriate method by having advantages including high scalability, reduced process time of combining and providing the best composition. In previous studies, each methods providing the best composition of web services had problems. Parallel Skyline service algorithm, by using parallelism, increases the speed of combining huge volume of services. The steps in proposed method are as follows:

- First the user enters his request in form of input service and enters the desired output and service quality. Requested service format is as follows:

Request: (Input/s, Output/s, QoS)

In this step, preprocessing is performed on incoming request. The pre-processing is operations such as search and retrieval of required service from data base and designing an expansion graph in parallel.

Construction of expansion graph in sequential order is described in the second part. In this paper construction of each graphs and compounds in each level are parallel unlike [11]. In other words, compounds of each level due to being independent, can be done simultaneously, so the desired graph is generated faster. And by increasing the number of services involved in the composition, computing time and graph construction will have no significant increase. As Figure 3 shows, in every line of the graph, the possible composition are created in the context of parent-child rules, and the final graph is created again by using parent-child.
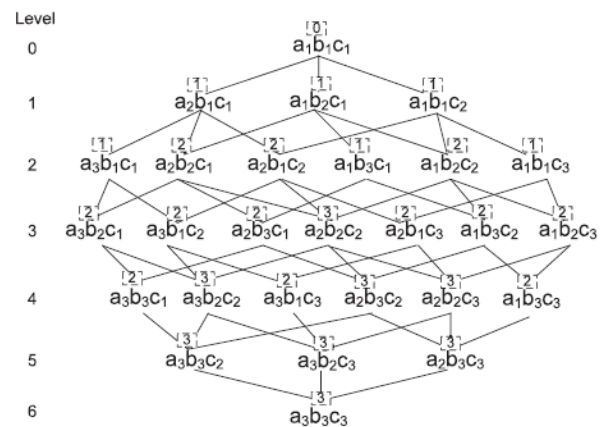

Fig. 3. Expansion Graph [3]

A. *Selecting the best service composition in each expansion graph by using a parallel linear combination*

According to Figure 4, parallel linear combination means comparing obtained combination of any comparison at all levels. For example, as shown in Figure 4, first two service composition of A, B through DPA algorithm are compared, and then the result is compared with a higher level of service composition which is C [3].
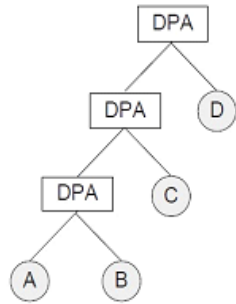

Fig. 4.   linear combination [3]

After the parallel linear combination in each expansion graph, the best combination of the graph will be determined, which are defined as the Skyline Services.

B. *Selecting the best combination from existing Skyline services*

After determining any suitable combination of the two graphs, the results of each graph are compared with each other. Finally, the best combination will be presented to the user.

V.   SIMULATION RESULTS OF THE PROPOSED MODEL

For analysis and evaluation of parallel Skyline service, first implementation of proposed approach provided by Skyline algorithms were evaluated. And then the desired results with the same data on the algorithms provided by Skyline, Top-K algorithm and the proposed parallel Skyline algorithm were evaluated.

Here, the results of the proposed algorithm for the problem of finding the best composition in web services are described.

A. *Runtime*

In analyzing the results, it is observed that the addition of web services of data set, increases the runtime.

Due to the fact that the proposed model, the best service composition has been provided through parallel Skyline, thus runtime in equal number of web services declined compared to the Top-K algorithm and improvements occurred in the algorithm's runtime. Then in Table 6 runtime charts for Top-K and Parallel Skyline and Skyline algorithms are specified.

TABLE VI.   RUNTIME (IN MILLISECONDS) IN IMPLEMENTATION OF PARALLEL-SKYLINE AND TOP-K AND SKYLINE METHODS WITH EQUAL NUMBER OF WEB SERVICES

Number of Services	Skyline	Top-K	Parallel-Skyline
10000	3.4	2.12	1.35
20000	15.3	6.12	2.17
30000	37.5	13.27	4.41
40000	81.35	22.65	8.65

In Fig. 5 these three methods in low scale and equal number of services are compared and shown.
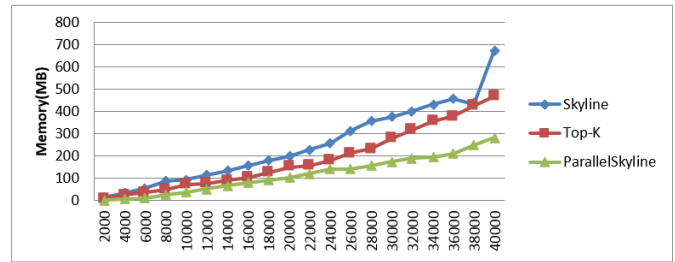

Fig. 5.   comparison of runtime in low scale

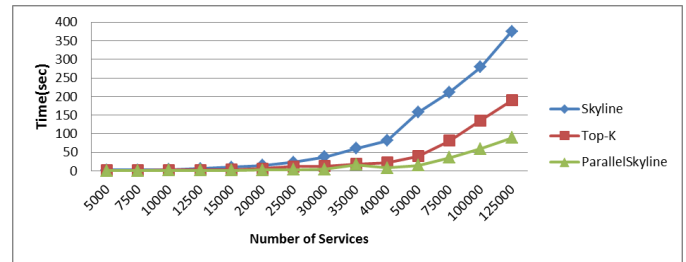These three methods are compared with each other in high scale and higher number of services.

Fig. 6.   comparison of runtime in high scale

As can be seen, Skyline method due to using parallel technique, has better composition speed than Top-K method.

B. *Memory Usage*

Another issue which can be analyzed in results is memory usage by proposed algorithm of Parallel-BUA.

Whereas Parallel-BUA method candidate services for composition has decreased; therefore, memory usage is less than Top-K and Skyline algorithms, and improvement occurs in memory usage. In Table 7 charts for the algorithms of Top-K and Parallel Skyline and Skyline are specified.

TABLE VII.   COMPARISON OF MEMORY USAGE IN TOP-K AND PARALLEL SKYLINE AND SKYLINE METHODS WITH EQUAL WEB SERVICES IN LOW SCALE (MB)

Number of Services	Skyline	Top-K	Parallel Skyline
10000	95.45	73.32	37
20000	178.95	150	101
30000	375	278.5	175
40000	675	469	282

In Fig. 7 these three methods in low scale and equal number of services are compared and shown.
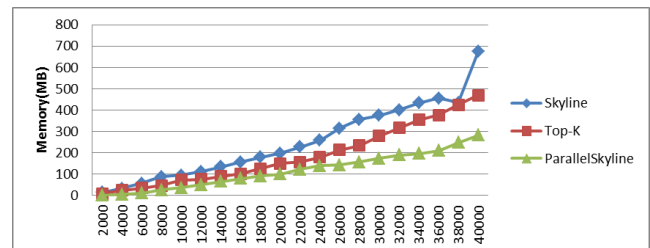

Fig. 7.   Comparison of memory usage in Top-K and Parallel Skyline and Skyline methods in low scale

These three methods are compared with each other in high scale and higher number of services.
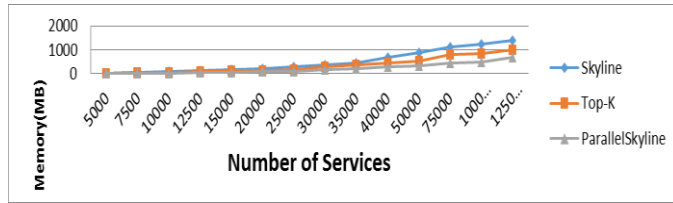

Fig. 8.   Comparison of memory usage in high scale

As you can see, Skyline method due to use of parallelism techniques and the use of more powerful processor, use less memory compared to the Top-K method.

In addition, in parallel Skyline method compared to Skyline service, number of services and sub-services is selected dynamically and it is done according to user requests. Then, by a Top-K filter is applied on user requests, and user creates services based on the quality and through Parallel Skyline the best composition is shown to the user. This method consumes less memory than the BUA and Top-K.

VI.   CONCLUSION AND FUTURE WORKS

The right composition of atomic services to provide services to users of is a challenge in web services. In this regard, choosing suitable atomic service required applying the right qualitative parameters is very important. Many methods have been suggested to solve the problem web services composition in regard to quality characteristics. As expected, web services with similar features lead to competition among the service providers. The key challenge to find an appropriate web service for composition occurs when multiple aspects of quality (such as response time, cost, etc.) are considered in the optimal composition of services. In this study various methods of service composition are discussed and compared. It was shown that previous methods had limitations in terms of computation and complexity. By using parallel Skyline service, selection speed and diversity has increased in a large scale space.

Suggested future works:

- Use parallel Skyline service to choose reliable services in social networks

- Use parallel Skyline service to predict quality of fuzzy services to solve the problem of inconsistency in fuzzy services.

- Use parallel Skyline service to solve inconsistent services

REFERENCES

[1]   Nuri S. and Tafakor F. (2013). Selecting optimal location services in service composition by using location service quality, mapping and geographic information Journal, p. 1-12

[2]   Wang, L., Shen, J. And Yong, J. (2012). A Survey on Bio-inspired Algorithms for Web Service Composition", Proceedings of the IEEE 16th International Conference on Computer Supported Cooperative Work in Design, pp. 569-574.

[3]   Yu, Q. And Bouguettaya, A. (2013). Efficient Service Skyline Computation for Composite Service Selection, IEEE Transactions on Knowledge & Data Engineering,vol.25, pp. 776-789.

[4]   Yu, Q. And Bouguettaya, A. (2010). Foundations for Efficient Web Service Selection", Springer Dordrecht Heidelberg London New York.

[5]   Wu, J., Chen, L. And Liang, T. (2014). Selecting Dynamic Skyline Services for QoS-based Service Composition, Applied Mathematics & Information Sciences an International Journal, vol. 8, PP. 2579-2588.

[6]   Benouaret, K., Benslimane, D. And HadjAli, A. (2011). On the Use of Fuzzy Dominance for Computing Service Skyline Based on QoS, IEEE Conference Web Services, PP. 540-547.

[7]   Benouaret, K., Benslimane, D., Hadjali, A. And Barhamgi, M. (2011). Top-k Web Service Compositions Using Fuzzy Dominance Relationship, International Conference on Services Computing, PP. 144-151.

[8]   Skoutas, D., Sacharidis, D., Simitsis, A. and Sellis, T. (2010). Ranking and Clustering Web Services Using Multicriteria Dominance Relationships, IEEE Tranactin on Services Computing,Vol.3, pp.163-177.

[9]   Yu, Q. and Bouguettaya, A. (2010). Computing Service Skyline from Uncertain QoWs", IEEE Tranaction on Services Computing, vol. 3, pp. 16-29.

[10] Alrifai, M., Skoutas, D. and Risse, T. (2010). Selecting Skyline Services for Qos-Based Web Service Composition, International World Wide Web Conference Committee, PP. 11-20.

[11] Chen, L., Cui, B. and Lu, H. (2011). Constrained Skyline Query Processing against Distributed Data Sites, IEEE Transactions on Knowledge and Data Engineering ,vol. 23,PP. 204-217.