

Performance Evaluation of Composed Web Services

Garima
Department of Computer
Science and Engineering,
KLSGIT, Belagavi
Karnataka, India

Kavita D. H.
Professor
Department of Computer
Science and Engineering,
KLSGIT, Belagavi
Karnataka, India

Aarti Shahapurkar
Professor
Department of Computer
Science and Engineering,
KLSGIT, Belagavi
Karnataka, India

ABSTRACT

Web Services technologies are established to deliver elasticity, inter-connection of dissimilar applications and have become popular. When user demand is not fulfilled, a composition is implemented to compose the existing services. It comforts to link over World Wide Web. Web-Service-Composition (WSC) is one of the flexible ways of integration of application and resource sharing. It aims to establish several services that handle the user necessities. It allows developers to create applications by focusing on the service oriented part. When lots of web services are available with similar characteristics and user is unable to decide which services are suitable, then we need to perform the composition of services. Performing WSC depends upon QoS parameter. Composing is a difficult problem and should guarantee quality parameters. Different approaches can be used and these must deal with different issues like performance. In our proposed work, we present the composed services with different performance parameters. In order to resolve the composed services, combining of different approaches were used.

Keywords

Quality-of-Service, Web Service Selection, Optimal solution, Heuristic.

1. INTRODUCTION

Web Services is the most skillful machinery. All the online communications and transactions are happening because of these web services. Web Services plays a major role in background since in professional community resources Web Services improve the professional process by means of different Web services. Earlier DCOM-CORBA-RMI was deployed to function. Web Service is an application component and these components are used to select the good web service based on user request. Web-Service-Composition (WSC) aims to organize numerous services to satisfy customer requirements. It has flexible ways of integration of applications and resource sharing. With the internet connectivity a user can access the available services in home instead of going to actual place like booking flight [1], hotel etc. But when there is a more a demand composition comes. These days, various web services are there which have similar functional characteristic. The characteristics of Web Services deals with the functional requirements that consist of its input-output, that invoke the execution and the nonfunctional requirements like cost, reliability etc. Its important benefits are that it allows greater and easier combination among system and applications. So if any communication fails to answer the query then there should be possibility of combination. In most of the cases, single web services cannot fully satisfy a customer needs and a service composition is needed to meet the requirements. During, service composition both requirements is important that is functional or non-functional. A Web Service can have several applications with the same functionality but may differ in terms of Quality-of-

Service parameters. Thus, it is the selection of web service application that delivers the best overall performance while meeting the business needs. Earlier only functional requirements of service composition is considered but today main focused is nonfunctional characteristics in service composition. Web services are the main technology which is itself responsible for evolution of IT-based services. It delivers a mechanism that combine several services together to handle business process. Automated service composition is valued in many fields, typical of e-commerce. There is increase in evolution of web services, business process with complex requirements in real world and also automatic composition requires flexibility. In Web-Service-Composition evaluating of performance parameters for a certain request is done by considering the values of quality parameters which is given by service provider. In reality, the values of Quality-of-Service parameter differ generally at request period which has several issues like network load, customers opening the similar services concurrently etc. Many algorithms are used to solve Composed Web Services problems. They are evolutionary algorithm, ant-colony, genetic algorithm etc. Every methodology focuses on a certain limitations established on the consumer's desires.

This paper introduces an approach that will evaluate the performance parameters for selecting the best services among the composed services.

2. LITERATURE REVIEW

Quality-of-Services of service composition has a main part to get a good web services. Several methods are there that are used for selecting nearly best Web-Service-Composition and also guarantees service execution performance.

In [1], the authors proposed a stochastic problem. The problem of WSC is to have effective, efficient composed systems that deliver guaranteed performance, while meeting business needs. They have considered random variables which represents Quality-of-Service parameters, which make composition or execution more convincing and also present different analysis methods that guarantee's with probabilistic part on service execution performance.

In [2], the authors used an Automatic-Composition of semantics web services. An infrastructure needs to be explained that permits the customer and the application to determine, organize, constitute and synthesis services automatically. They have used a method that determine and composed the automatic services that are based on description of web services.

In [5], the authors proposed Markov-HTN planning method that improves elasticity of automated Web-Services Composition. AI planning methods is used to achieve the flexibility. HTN-planning can be adopted to deal with the OWL-S Web-Service-Composition problems. They have used

combination of two models that are Markov decision process and HTN-planning model.

In [6], the authors proposed a Binary tree-based parallelization which explains Web-Service-Composition. They explain how the partition of composition problem into several number of parallel threads is done, which is presented by a framework. They also showed some approaches that can be benefited from Advanced Pruning Heuristics used in different algorithms.

In [8], the authors used an effective QoS-Aware Service Composition with a Probabilistic Selection Policy of services. They have adjusted the problem of repetitions of services execution in long-term. The Probabilistic Selection Policy allows of choosing several services with the same workflow and also the modeled this problem with the Linear Programming.

In [10], the authors proposed effective WSC with Quality-of-Service constraints by using Hybrid methods. They proposed a mix solution that mixes the Overall-optimization with Local-selection methods and also the solution that consist distributed local selection.

In [13], the authors proposed an optimized Quality-of-Service and removal of redundancy by using Web-Service-Composition approach. It works by finding the candidate solution with the best quality of service while satisfying its functional goals. Removing redundancy in composition process is very useful.

In [15], the authors proposed a Scalable and Approximate mechanism for WSC. They explain the flow of scalable but approximate Quality-of-Service-aware composition mechanism which handles the complexity, with the quality parameters.

3. PROBLEM STATEMENT

Web Services are getting additional popularity in environment. By using internet connectivity individual person can access the available services from anywhere without going to actual residence. But when employer, worker and customer demand more services then the necessity of composed services came. These days, countless Web Services are present in internet and several Web Services which have related functional representative. Web-Service-Composition is the task of merging services together. It is performed to achieve the user needs. Several algorithms are used to perform WSC. The paper introduces algorithms that will select the best services area among the composed plan that solve the interrelated reaction to the customer queries and also analyzes QoS-parameters.

4. DETAILED METHODOLOGY

In our proposed work, we have started by using Amazon Datasets and then evaluated by using proposed approaches. Here, we consider a set of web services and for each set of I/O. Then for each web services a set of QoS parameters. The formal approaches that are used to evaluate are:

4.1 Suboptimal

It is the strategy that helps to reduce the searching space. Suboptimal solution use A* approach. The main part is to have optimal solution path that satisfies the constraints with time by providing the dependent subgraph, also a query.

Firstly, demonstrate levels with nodes of dependent subgraph i.e. starting to level(0) upto level (max+1).It begins by traversing in backward direction that starts from last node. Consider N, which is the collection of nodes in the dependent subgraph and a state is a group of nodes. It has initial state that is Ne and goal state that is Ns. Its cost function F(s) over each state s is $F(s) = G(s) + H(s)$. G(s) means the cost beginning from initial to current state and H(s) is the heuristic function which means how far the existing state from the objective state. Also, optimal strategy used stochastic constraints. Given a dependency subgraph G(N,E) where N indicates vertices and E as edges. Each edge of the dependent subgraph is noted with an Input image. To begin with include two dummy nodes Ns and Ne of the subgraph. The query inputs are the outputs of Ns.

4.2 Scalable Composition technique

It produces the result in much lesser time. This mechanism generates an optimal solution in a reasonable time limit also called heuristic technique. First partition the Dependency Subgraph into multiple layers according to the level of each. A layer includes collection of nodes having similar levels. Add one dummy node (if necessary) in every layer to ensure that every edges are among the successive layers. If a node in one layer takes an input that is not created from its immediate previous layer, we insert dummy node in its intermediate layers and pass the input through the dummy nodes. One layer is now connected to its immediate predecessor-successor. The I/O of a dummy node is same. The Quality-of-Service parameter of the dummy node are set to their best value, e.g., the response time of the dummy node is set to 0, the reliability and the availability are set to 1 and the invocation cost of the dummy node is set to 0. The invocation cost for a service which is not dummy is 1.

5. PROPOSED MODEL

The main part of selecting the services is depends on QoS-parameters which can be done by evaluating its performance metrics using Sub-Optimal and Heuristic approaches. It can be implemented in java or C sharp language on Ubuntu Linux system or in Microsoft visual studio.

The figure 1 explains an architecture overview. It consists of three main parts. The first part is Dependent Graph generator, which takes the majority of the data from the data base and constructs Dependent Graph. The next part is the Sub Graph generator. When the queries originate at execution time, the Sub Graph generator generates the subgraph from the dependent graph. Finally, the third part executes the dissimilar algorithm that is Optimal Solution and Scalable heuristic method that satisfies the client request with quality parameters. The algorithms were evaluated against Amazon Web Service Datasets. Network was selected by Amazon website. It is based on customer requirements that get the item from the Amazon website. If a product i is recently purchased with j, then its graph includes a directed edge from i to j. Later, it simulates to evaluate Quality-of-Service parameters like response time, turnaround time. Heuristic and Sub Optimal Algorithm is used to evaluate performance metrics. In end, it will show its calculated timing of each parameter. Plotting of graphs will be done in excel sheet which consist of the values of parameters with both the methods. Later, comparison is done by plotting the graphs of each parameter.

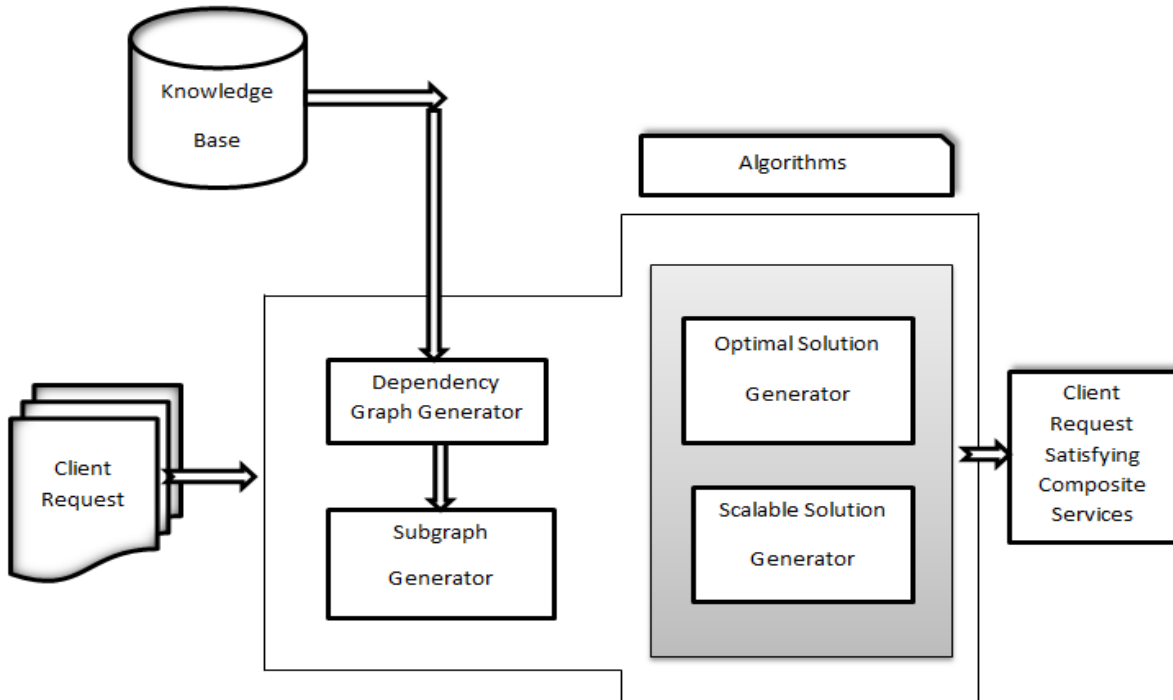


Figure 1: An Architecture Overview

6. RESULTS AND DISCUSSION

Validation of algorithm has done against the Amazon Web Services Datasets. The proposed algorithm considered the two qualities of service parameters i.e. Response time and Turnaround time that displays the comparison between the proposed methods. It will compare the values of parameters with the proposed methods. By applying the approaches on datasets we randomly chose the services and randomly assigned the values of response and turnaround time.

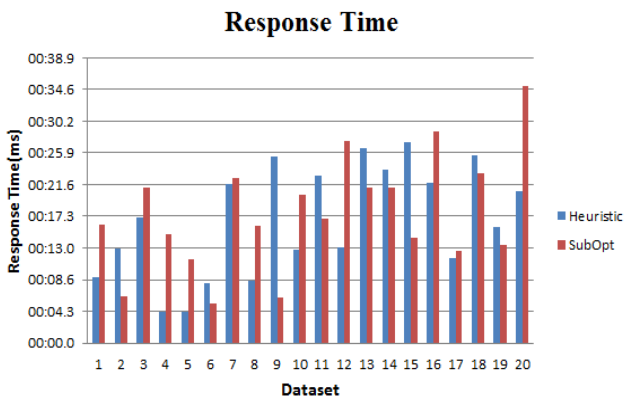


Figure 2: Comparison of Response Time

The figure 2 displays its final experimental results which indicate the response time calculated by both the methods. Response Time is the time which is taken from the customer and produced its first response. Response time is measured in ms unit. As it can be seen from the figure 2, in most of the cases heuristic gives optimal solutions, which means heuristic response better than the suboptimal method in most of the cases.

Turnaround Time

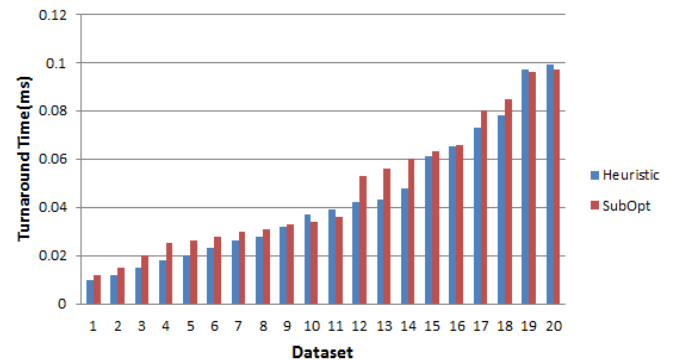


Figure 3: Comparison of Turnaround Time

The figure 3 shows the final experimental results which indicate the turnaround time. Turnaround Time is the time which is taken to perform a specific query and then return its complete result. Turnaround time is measured in ms unit. From the figure 3, in most of the cases heuristic gives optimal solutions, which means it can return the result in lesser time as compared to other method.

7. CONCLUSION

In the paper, evaluation of performance of WSC is presented by using Heuristic-based algorithm and Sub-Optimal algorithm. The Quality-of-Service is prepared by the non-functional parameters for the WSC. The result of the experiments on amazon datasets has proved that proposed approaches keeps a very good QoS-Performance. Datasets include the information of customer items which is presented in a form of edges and nodes. Evaluation of performance is done using nonfunctional requirements. Moreover, the full validation of algorithm has done against the Amazon Web Service Datasets.

8. FUTURE WORK

In future work experimentation of proposed algorithm with the large no. of Web Services can be done. It then evaluates the other strategies that would find the best Composed Services. Also progressed effectiveness of the algorithm and the client inclination must be formulated. Further it can be improved by removing all the redundant services. Different techniques used are like Logistic-Relapse and maximal likelihood estimation on approximating the probabilities distribution from real services logs. Intelligent approaches can take for determining quality levels with clustering algorithms. This approach improves scalability.

9. ACKNOWLEDGMENTS

This Paper is supported by the department of Computer Science and Engineering, KLSGIT, Belagavi. We would like to express our deepest appreciation to the Department as well as to the university.

10. REFERENCES

- [1] Soumi Chattopadhyay, Ansuman Banerjee, "QSCAS: QoS aware web service Composition Algorithms with Stochastic parameters" in 2016 IEEE International Conference on Web Services.
- [2] S. Kona, A. Bansal, and G. Gupta, "Automatic composition of semantic web services," in ICWS 2007.
- [3] A Michael C. Jaeger., "Wsc-08: continuing the web services challenge," in 10th IEEE Conference on E-Commerce and E-Services. IEEE, 2008.
- [4] W. Wiesemann et al., "A stochastic programming approach for qos aware service composition," in CCGRID 2008.
- [5] K Chen "Markov-htn planning approach to enhance flexibility of automatic web service composition," in ICWS 2009.
- [6] P. Hennig and W.-T. Balke, "Highly scalable web service composition using binary tree-based parallelization," in ICWS 2010.
- [7] P. Rodriguez-Mier et al., "Automatic web service composition with a heuristic-based search algorithm," in ICWS 2011.
- [8] Adrian Klein¹, Fuyuki Ishikawa², and Shinichi Honiden, "Efficient QoS aware Service Composition with a Probabilistic Service Selection Policy".
- [9] A. Klein, F. Ishikawa, and S. Honiden, "Efficient heuristic approach with improved time complexity for qos-aware service composition," in ICWS 2011.
- [10] Thomas Rise et al., "A hybrid approach for efficient web service composition with end-to-end qos constraints," 2012.
- [11] W. Jiang et al., "Continuous query for qos-aware automatic service composition," in ICWS 2012, June 2012.
- [12] B. B. Nandi et al., "Stochastic VM multiplexing for datacenter consolidation," in SCC, 2012.
- [13] Y.-M. Xia and Y.-B. Yang, "Web service composition integrating qos optimization and redundancy removal," in ICWS 2013.
- [14] C. Sepulveda and Jesus B., "Qos aware descriptions for restful service composition: security domain," WWW, 2015.
- [15] Ansuman Banerjee et al., "A scalable and approximate mechanism for web service composition," in ICWS 2015.