International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2017 IJSRCSEIT | Volume 2 | Issue 4 | ISSN : 2456-3307

Service Composition to Enhance the QOS of Web Services

BadhriSupraja, Dr. B. Lalitha
Department of Computer Science and Engineering,Jntua, Ananthapur, India

ABSTRACT

QoS-based service selection becomes a usually accepted procedure to support fast and dynamic net service composition.An adaptation technique is employed for choosing services supported the hardness of QoS constraints. The fundamental plan is to sample services that represent a selected quality-value vary. The quality-value vary of candidate services is split into smaller sub-ranges within which representative services ar sampled and evaluated. At now, the dimensions of the QoS sub-ranges is decided adaptably supported the hardness of the QoS constraints. In this, we'll notice the edge worth to keep up an affordable level of optimality so as to extend the success rate of service composition. during this we address this downside and propose an answer that mixes global optimisation with local selection techniques to learn from the benefits of each globals. The projected resolution consists of 2 steps: Initially, we use Mixed number Programming(MIP) to search out the best decomposition of global QoS constraints into local constraints. Second, we use distributed local selection to search out the most effective net services that satisfy these local constraints.
Keywords: WebServices, QoS, Optimization, Service Composition.

I. INTRODUCTION

The service-oriented computing paradigm and its realization through standardized internet service technologies offer a promising answer for the seamless integration of business applications to make new added services. With the growing range dispute internet services that offer an equivalent practicality. However, differ in quality parameters, i.e,..the composition drawback in deciding the services on the selection of element services with regards to useful and non-functional necessities. The QoS optimisation in our model is allotted by a group of distributed service brokers. the thought is to decompose QoS global constraints into a group of local constraints that may function a conservative upper/lower bounds, such the satisfaction of local constraints by a neighborhood service broker guarantees the satisfaction of the global constraints. By combining global optimisation with local selection our approach in an exceeding position should be ready to expeditiously solve the selection drawback in a distributed manner. Experimental evaluations show that our approach is ready to succeed in close-to-optimal results abundant quicker than existing 'pure' global optimisation approaches.

II. METHODS AND MATERIAL

A. QOS Computation of Composite Services

The QoS evaluation of a composite service is determined by the QoS values of its part services in addition because the composition model used (e.g. sequential, parallel, conditional and/or loops). In this paper, we have focuses on to specialise in the consecutive composition model. Different models could also be reduced or reworked to the consecutive model. The QoS vector for a composite service candidateelementis outlined as represents the estimated value of the ithQoS attribute of candidate element and may be aggregative from the expected QoS values of its part services. In our model we use to contemplate 3 sorts of QoS aggregation functions: 1) summation, 2) multiplication and 3) minimum relation. Table 1 shows samples of these aggregation functions.

$$Q_{CS} = \{q'_1(CS), \ldots, q'_r(CS)\}. \quad q'_i(CS)$$

Aggregation type	Examples	Function
Summation	Response time	$q'(CS) = \sum_{j=1}^{n} q(s_j)$
	Price	
	Reputation	$q'(CS) = 1/n \sum_{j=1}^{n} q(s_j)$
Multiplication	Availability	$q'(CS) = \prod_{j=1}^{n} q(s_j)$
	Reliability	
Minimum	Throughput	$q'(CS) = \min_{j=1}^{n} q(s_j)$

Table 1: Examples of QoS aggregation functions

B. Global QOS Constraints

Global QoS constraints represent user's end-to-end QoS needs. These may be expressed in terms of higher (and/or lower) bounds for the aggregate values of the different QoS criteria. As mentioned earlier, we solely think about negative QoS criteria. Thus our model have bound constraints.

Utility Function:

In order to judge the multi-dimensional quality of a given internet service a utility operate is employed. The operater maps the standard vector Qs into one real values, to alter sorting and ranking of service candidates. during this paper we use a Multiple Attribute higher cognitive process approach for the utility function: i.e. the Simple Additive Weighting (SAW) technique [22]. The utility computation involves scaling the QoS attributes values to permit a homogenous measure of the multi-dimensional service qualities freelance of their units and ranges. within the scaling method every QoS attribute values is reworked into a values between 0 and 1, by examining it with the minimum and most feasiable values in keeping with the on the market QoS data of service candidates. For a composite service CS = {S1, . . . ,Sn}, the aggregated QoS values are compared with minimum and most feasible aggregate values. The minimum (or maximum) doable aggregate values may be simply calculable by aggregating the minimum (or maximum) value of every service category. Formally, the minimum and most aggregate values of the k-thQoS attribute of service are computed as follows:

$$Qmin'(k) = \sum_{j=1}^{n} Qmin(j,k)$$

$$Qmax'(k) = \sum_{j=1}^{n} Qmax(j,k)$$

$$Qmin(j,k) = \min_{\forall s_{ji} \in S_j} q_k(s_{ji})$$

$$Qmax(j,k) = \max_{\forall s_{ji} \in S_j} q_k(s_{ji})$$

whereQmin(j, k) is the minimum valueand Qmax(j, k) is the maximum value that can be expected for service class Sj according to the available information about service candidates of this class.

Now the utility part of the internet service s ϵ Sjis computed as

$$U(s) = \sum_{k=1}^{r} \frac{Qmax(j,k) - q_k(s)}{Qmax(j,k) - Qmin(j,k)} \cdot w_k$$

and the overall utility of a composite service is computed as

$$U'(CS) = \sum_{k=1}^{r} \frac{Qmax'(k) - q'_k(CS)}{Qmax'(k) - Qmin'(k)} \cdot w_k$$

C. Problem Statement

The drawback of finding the most effective service composition without enumerating all attainable mixtures is taken into account as associate improvement problem, during which the utility values needs to be maximized whereas satisfying all global constraints.

D. QOS-Aware Service Composition

The use of Mixed whole number Programming [18] to unravel the QoS-aware service composition drawback has been recently projected by many researchers [24, 25, 4,5, 6]. Binary decision variables are unit employed in the model to represent the service candidates. A service candidate sijis chosen within the optimum composition if its corresponding variable xij is about to one within the resolution of the model and discarded otherwise. Another disadvantage of this approach is that it needs the QoS information of obtainable net services be foreign from the service broker into the MIP model of the service composition, that raises high communication. To deal with these limitations, Wedivide the QoS-aware service composition drawback into 2 sub-problems which will be solvedadditional expeditiously in 2 succeeding phases. Figure 1 offers an summary on our approach. within the 1st phase, the service composition decomposes every global QoS constraints into local constraints on the other component services level and sends these constraints to the concerned service brokers. These constraints additionally embrace users preferences, that are unit non-level in terms of values of the QoS attributes. Within the second phase, every service broker performs local selection to seek out the most effective composite services that satisfy these local constraints. The 2 phases of our approach are unit describe within the next sub sections in additional details.

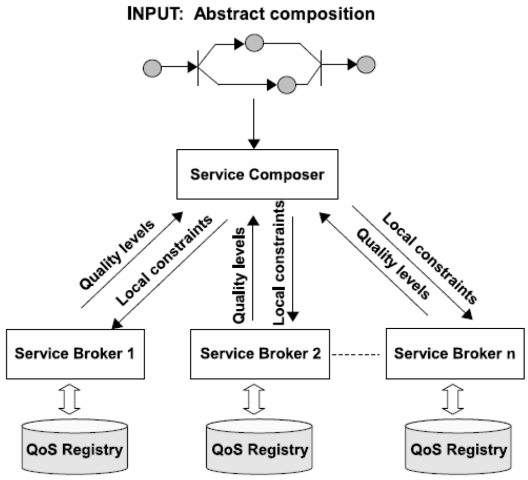
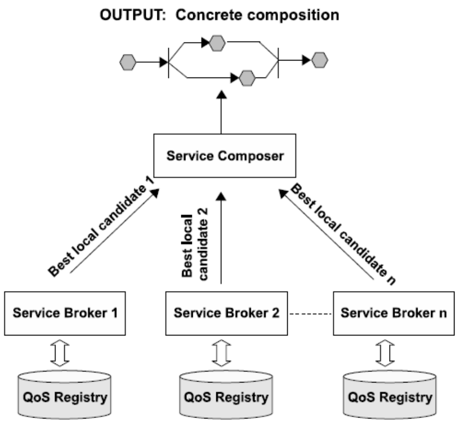


Figure 1: phase 1:Decompostition of global QoS constraints



Phase 2: Local selection of best candidates

Decomposition of Global QoS Constraints

To ensure the fulfillment of global QoS constraints in an exceeding service composition drawback while not enumerating all attainable combos of element internet service, we decompose every QoS global constraint c0 into a collection of n local constraints c1, . . . , cn (n is that the variety of abstract service categories within the composition request). The local constraints function a conservative higher bounds, such the satisfaction of local constraints guarantees the satisfaction of global constraints. the global constraint on total execution worth is then mapped to the value levels of service categories. we use Mixed number Program (MIP) [18] finding techniques to search out the most effective mapping of global constraints to local quality levels. Not as the MIP model in [24, 25, 5, 6], our MIP model has abundant less variety of variables (i.e. the standard levels rather than actual service candidates) and might be, therefore, solvedeasily.

Determining Quality Levels

Quality levels area unit initialized for every service category s_j by dividing the worth ranges of every QoS attribute q_k into a collection of the separate quality values as represented in figure. Thepair of values assign each quality level q^z_{jk} a values p^z_{jk}between 0 and 1, that estimates the advantage of victimization this quality level as a neighborhood constraint. These values is decided as follows. First, we work out $h(q^z_{jk})$, i.e. the amount of candidate services that might qualify if this level was used as local constraint. Second, we calculate the utility value of every service candidate within the service category victimization the utility operate and confirm(4) $u(q^z_{jk})$, i.e. the best utility value that may be obtained by considering these qualified services.

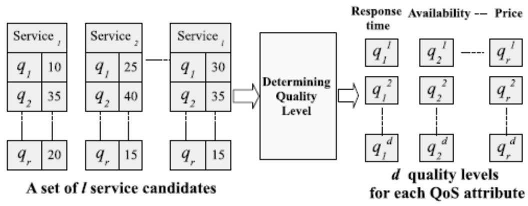


Figure 2: Quality Level Selection

Formulating the MIP Model

We use MIP model to search out the most effective decomposition of QoS constraints into local constraints. Therefore, we use a binary call variable $x^z_{jk}$for each local quality level q^z_{jk} such that $x^z_{jk} = 1$ if q^z_{jk} is selected as a local constraint for the QoS attribute q_k at the service class S_j , and $x^z_{jk} = 0$ otherwise. Note that the whole variety of variables within the model equals to n .m .d, i.e. it's freelance of the amount of service candidates. If the amount of quality levelssatisfiesd <= l we will make sure that the scale of our MIP model is smaller than the scale of the previous MIP model[24, 25, 5, 6]

Local Selection

After moulding global QoS constraints into local ones, the second phase of our approach is to perform local selection for every service category repeatdly. Upon the receipt of local constraints and user' preferences from the service composition, every service broker performs the local selection and returns the most effective internet service candidate to the service composition. The received local constraints area unit used as higher bounds for the QoS values of element services. Internet services that violate these higher bounds area unit skipped from the selection. An inventory of qualified services is made and sorted by their utility values.

III. RESULTS AND DISCUSSION

Performance Study

The aim of this analysis is to validate our hypothesis that our approach achieves close-to-optimal results with a way lower computation time compared to "pure" global optimization approach as planned by [15, 25, 6]. within the following use the label "hybrid" to consult with our resolution and also the label "global" to consult with the "pure" global improvement approach.

Performance Analysis

The measurability of QoS-based service composition systems is full of the time complexity of the applied formula. In our approach, we use mixed Integer programming to resolve a part of the matter, namely, the decomposition of the global QoS constraints into local ones. the particular selection of services, however, is finished victimization distributed local selection strategy, that is extremely economical and climbable. The local utility computation for service candidates features a linear complexity with relevance of the amount of service candidates, i.e. O(l). As service brokers will perform the local selection in parallel, the overall time complexity of this step isn't full of the amount of service categories, hence, the complexity of the second step remains O(l). The time complexity of our approach is dominated by the time complexity of the constraint decomposition half. the amount of call variables in our MIP model is n . m .d, wherever n is that the variety of service categories, m is that the variety of global QoS constraints and d is that the variety of quality levels. Consequently, the time complexity of our approach is freelance on the amount of accessible net services, that makes it additional climbable than existing solutions that admit "pure" global improvement.

Evaluation Methodology

The Dataset In our study, wediscovered with QoSdataset,which consists variety of track records. we contemplate the dataset that we've taken and realize the Qos Attributes, Table two lists the QoS attributes this dataset and provides a brief description of each attribute.In our dataset we've thought of some attributes like response time,availability,throughput and with this we can search the data related to those attribute values.

QoS Attribute	Description	Units Of Measurement
Response Time	Time taken to send a request and receive a response	millisecond
Availability	Number of successful invocations/total invocations	percent
Throughput	Total number of invocations for a given period of time	invocations / second
Likelihood of success	Number of response/number of request messages	percent
Reliability	Ratio of the number of error messages to total messages	percent
Compliance	To which extent a WSDL document follows the WSDL spec.	percent
Best Practices	To which extent a web service follows the Web Services Interoperability (WS-I) Basic Profile	percent
Latency	Time the server takes to process a given request	millisecond
Documentation	Measure of documentation (i.e. description tags) in WSDL	percent

Table 2: Parameters used in Quality Web Services

Experimental Results

(I)By givinga particular server in the search box we will get the graph based on the server data which we have searched.Here in x axsis we considered how many servers we selected and how many servers are lasting.In y axis total number of servers add up we will consider.

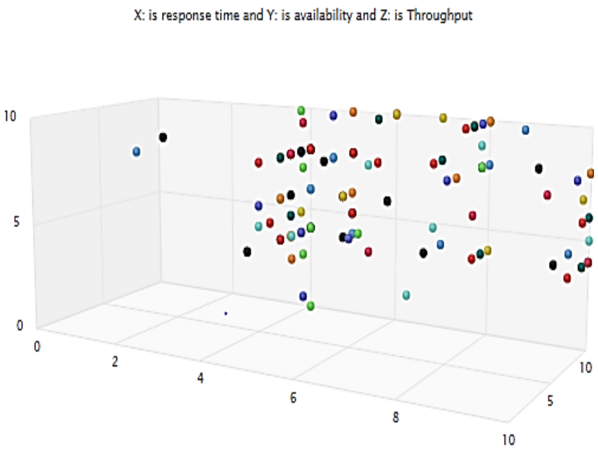
Graph for compute server

(II)Next we will see the threshold search.

Here by giving all the data like servername, response, availability, through put values we get the graph based on our search constraint. Here in x axsis we considered our constraint values and how many values matched our constraint.Iny axis number count we will consider.

Graph for Threshold search

(III)Now we will see the three dimensional graph of our project, that isbased on response time, availability and throughput.Here in x-axis contains response time,y-axix contains availability and z-axsis contains throughput.



Three dimensional graph
X-axis: refers to response time
Y-axis: refers to availability and
z-axis: refers to throughput.

IV.CONCLUSION

In this paper we finded the threshold value to maintain a reasonable level of optimality so as to extend the success rate of service composition. during this we also find the analysis ofoptimality and success rate, which are mainly affect by the number of candidate service instances and the hard-ness of the global constraints. By using the correlation information between a sub-range and QoS constraints, representative services may be selected more accurately. This reduces the number of failures of the service composition and increases the opti-mality with spending a less amount of composition time.

V. REFERENCES

[1]. M. M. Akbar, E. G. Manning, G. C. Shoja, and S. Khan. Heuristic solutions for the multiple-selection multi-dimension knapsack problem. In Proceedings of the Global Conference on Computational Science-Part II, pages 659-668, London, UK, 2001. Springer-Verlag.

[2]. E. Al-Masri and Q. H. Mahmoud. The qws dataset. Web page. http: //www.uoguelph.ca/~qmahmoud/qws/indexhtml/.

[3]. E. Al-Masri and Q. H. Mahmoud. Qos-based discovery and ranking of web services. In Proceedings of the IEEE Global Conference on Computer Communications and Networks, 2007.

[4]. B. Lalitha Optimizing Service Selection in Combinatorial Option by Resolving Non-linear constraints.

[5]. D. Ardagna and B. Pernici. Global and local qos, constraints guarantee in web service selection. In Proceedings of the IEEE Global Conference on Web Services, pages 805-806, Washington, DC, USA, 2005. IEEE Computer Society.

[6]. D. Ardagna and B. Pernici. Adaptive service composition in flexible processes. IEEE Transactions on Software Engineering, 33(6):369-384, 2007.

[7]. C. Aurrecoechea, A. T. Campbell, and L. Hauw. A survey of qos architectures. Multimedia Systems, 6(3):138-151, 1998.

[8]. B. Benatallah, Q. Z. Sheng, A. H. H. Ngu, and M. Dumas. Declarative composition and peer-to-peer provisioning of dynamic web services. In Proceedings of the Global Conference on Data Engineering, pages 297-308, Washington, DC, USA, 2002. IEEE Computer Society.

[9]. A. S. Bilgin and M. P. Singh. A daml-based repository for qos-aware semantic web service selection. In Proceedings of the IEEE Global Conference on Web Services, pages 368-375, Washington, DC, USA, 2004. IEEE Computer Society.

[10]. J. Cardoso, J. Miller, A. Sheth, and J. Arnold. Quality of service for workflows and web service processes. Journal of Web Semantics, 1:281-308, 2004.

[11]. F. Casati and M.-C. Shan. Dynamic and adaptive composition of e-services. Information Systems, 26(3):143-163, 2001.

[12]. Y. Cui and K. Nahrstedt. Supporting qos for ubiquitous multimedia service delivery. In Proceedings of the ACM Global Conference on Multimedia, pages 461-462, 2001.

[13]. M. Gillmann, G. Weikum, and W. Wonner. Workflow management with service quality guarantees. In Proceedings of the SIGMOD Conference, pages 228-239, 2002.

[14]. F. Li, F. Yang, K. Shuang, and S. Su. Q-peer: A decentralized qos registry architecture for web services. In Proceedings of the Global Conference on Services Computing, pages 145-156, 2007.

[15]. Y. Liu, A. H. H. Ngu, and L. Zeng. Qos computation and policing in dynamic web service selection. In Proceedings of the Global Global Wide Web Conference, pages 66-73, 2004.

[16]. I. Maros. Computational Techniques of the Simplex Method. Springer, 2003.

[17]. K. E. Michel Berkelaar and P. Notebaert. Open source (mixed-integer) linear programming system. Sourceforge. http://lpsolve.sourceforge.net/.

[18]. G. L. Nemhauser and L. A. Wolsey. Integer and Combinatorial Optimization. Wiley-Interscience, New York, NY, USA, 1988.

[19]. OASIS. Web services business process execution language, April 2007. http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf.

[20]. D. Pisinger. Algorithms for Knapsack Problems. PhD thesis, University of Copenhagen, Dept. of Computer Science, February 1995.

[21]. M. Wagner and W. Kellerer. Web services selection for distributed composition of multimedia content. In Proceedings of the ACM Global Conference on Multimedia, pages 104-107, New York, NY, USA, 2004. ACM.

[22]. K. . P. Yoon and C.-L. Hwang. Multiple Attribute Decision Making: An Introduction (Quantitative Applications in the Social Sciences). Sage Publications, 1995.

[23]. T. Yu, Y. Zhang, and K.-J. Lin. Efficient algorithms for web services selection with end-to-end qos constraints. ACM Transactions on the Web, 1(1), 2007.

[24]. L. Zeng, B. Benatallah, M. Dumas, J. Kalagnanam, and Q. Z. Sheng. Quality driven web services composition. In Proceedings of the Global Global Wide Web Conference, pages 411-421, 2003.

[25]. L. Zeng, B. Benatallah, A. H. H. Ngu, M. Dumas, J. Kalagnanam, and H. Chang. Qos-aware middleware for web services composition. IEEE Transactions on Software Engineering, 30(5):311-327, 2004.