Journal of Southeast University (English Edition) Vol. 24, No. 3, pp. 308 – 311 Sept. 2008 ISSN 1003—7985

Web services composition with QoS bound based on simulated annealing algorithm

Liu Qing Zhang Shilong Yang Rui Lian Xiangjian

(School of Information, Renmin University of China, Beijing 100872, China)

Abstract: In order to enable quality-aware web services selection in the process of service composition, this paper first describes the non-functional requirements of service consumers and the quality of elementary service or composite service as a quality vector, and then models the QoS (quality of service)-aware composition as a multiple criteria optimization problem in extending directed graph. A novel simulated annealing algorithm for QoS-aware web services composition is presented. A normalizing for composite service QoS values is made, and a secondary iterative optimization is used in the algorithm. Experimental results show that the simulated annealing algorithm can satisfy the multiple criteria and global QoS requirements of service consumers. The algorithm produces near optimum solution with much less computation cost.

Key words: web services; quality of service (QoS); service composition; simulated annealing

Web services are new software systems which can be identified by Internet according to XML based open standards (e. g., SOAP, WSDL, UDDI and WSFL). Service providers and users can describe and request services using such standards. The technology of web services provides an integration and interaction mechanism to the heterogeneous, autonomous and loosely coupled distributed applications. As a single web service simply to meet the needs of some practical applications, there is a need for composing a single existing web service, to generate more complex features and more powerful web services to support various applications. On the one hand, with the study of web services heating up and more and more web services being published, initial conditions for integrating services have formed. On the other hand, due to the impact of the software reuse, people are gradually paying attention to combinations of the existing services. For example, many enterprises need to integrate businesses and eliminate the large amount of information islands. At present there are some different concepts of combination for web services; a definition is given in Ref. [1]. A service composition combines services following a certain composition pattern to achieve a business goal, solve a scientific problem, or provide a new service function in general. Service compositions may themselves become services, making composition a recursive operation.

Due to the increase in the number of web services, many services have the same or similar functions but often with different qualities of service. Under the premise of the functional requirements, the service which meets the users' QoS bounds is an effective service. In the study of the combination of web services, choice of services considering QoS becomes an important issue[2]. The current service selection methods based on QoS of web services can be divided into two categories: the calculation[3–5] of QoS attributes and the semantic-based[6] approach. QoS attributes calculation method can achieve the global optimization of composite services with QoS bounds. Most of the existing literature obtained composite services on the basis of QoS attributes matrices with linear programming[3] or genetic algorithms[5]. It is an NP-complete problem that searches for a plan of the optimal composite service which meets a number of QoS attributes[5]. The genetic algorithm is the most common optimization method. Its encoding method and design of genetic operator influence the efficiency and effectiveness of problem solving. There is not a universal coding theory at present. Linear programming is inefficient when the problem is with difficulty formulated as a linear program. In this paper, the heuristic random search technology—the simulated annealing algorithm is used to solve service composition problems.

1 Web Services Composition Based on QoS

1.1 Definition for QoS of composite service

The QoS of a web service includes k attributes. The simulation experiments in this paper consider five basic attributes[3]: price, execution time, availability, reliability, and service reputation. Assume that a certain composite service named P will accomplish N tasks, the task t_i will be imposed on one particular type of service s_i, and s_i includes m services with the same function. Then the QoS of the composite service P can be defined:

1) Service price

$$Q_{\text{price}} = \sum_{i=1}^{N} q_{\text{price}}(s_{ij}, t_i) \qquad 1 \leqslant j \leqslant m$$

where $q_{\text{price}}(s_{ij}, t_i)$ represents the money that a service requester has to pay for invoking service s_{ij} in the service class s_i for accomplishing task t_i. The composite service price $Q_{\text{price}}(P)$ is a sum of every service s_{ij}'s price in P.

2) Execution time

$$Q_{\text{du}} = \sum_{i=1}^{N} q_{\text{du}}(s_{ij}, t_i) \qquad 1 \leqslant j \leqslant m$$

where $q_{\text{du}}(s_{ij}, t_i)$ represents the time that service s_{ij} costs in the service class s_i for finishing task t_i. The composite service execution time $Q_{\text{du}}(P)$ is a sum of time of every service

---

Received 2008-04-15.

Biography: Liu Qing (1967—), female, associate professor, qliu@ruc.edu.cn.

Foundation items: The National Natural Science Foundation of China (No. 60773217), Free Exploration Project (985 Project of Renmin University of China) (No. 21361231).

Citation: Liu Qing, Zhang Shilong, Yang Rui, et al. Web services composition with QoS bound based on simulated annealing algorithm[J]. Journal of Southeast University (English Edition), 2008, 24(3): 308 – 311.

s_{ij}'s execution time in P.

3) Reputation

$$Q_{\rm rep} = \frac{1}{N}\sum_{i=1}^{N} q_{\rm rep}(s_{ij}, t_i) \quad 1 \leqslant j \leqslant m$$

where $q_{\rm rep}(s_{ij}, t_i)$ is a measure of trustworthiness for accomplishing task t_i by service s_{ij} in service class s_i, which can be obtained from the users' experience of service s_{ij}. $Q_{\rm rep}(P)$ is the average of the reputation of all the services in the composite service P.

4) Reliability

$$Q_{\rm rel} = \frac{1}{N}\sum_{i=1}^{N} q_{\rm rel}(s_{ij}, t_i) \quad 1 \leqslant j \leqslant m$$

where $q_{\rm rel}(s_{ij}, t_i)$ is the probability that a request for accomplishing task t_i is correctly responded to by service s_{ij} within a maximum expected time frame. $Q_{\rm rel}(P)$ is the average of the reliability of all the services in the services composition P.

5) Availability

$$Q_{\rm av} = \frac{1}{N}\sum_{i=1}^{N} q_{\rm av}(s_{ij}, t_i) \quad 1 \leqslant j \leqslant m$$

where $q_{\rm av}(s_{ij}, t_i)$ is the probability that the service s_{ij} for accomplishing task t_i is accessible. $Q_{\rm av}(P)$ is the average of the availability of all the services in the services composition P. The QoS vector for service s_{ij} that completes task t_i is

$$\boldsymbol{q}(s_{ij}) = \{q_{\rm price}(s_{ij}, t_i), q_{\rm du}(s_{ij}, t_i), q_{\rm rep}(s_{ij}, t_i), q_{\rm rel}(s_{ij}, t_i), q_{\rm av}(s_{ij}, t_i)\}$$

1.2 Description of services composition

In this paper, the directed graph is used as the structure of the services composition.

Definition 1 Let $G = (V, E, C, v_s, v_g)$ be a connected, directed graph with a given weight, where $V = \{v_1, \ldots, v_n\}$ is the vertices collection of the graph G, and every vertex v_i is related to a QoS vector $\boldsymbol{q}(s_{ij})$ of service s_{ij}. The QoS vector is called the vertex weight; E is the collection of directed edges of the graph G. If $(v_i, v_j) \in E$, then $v_i \in V$, $v_j \in V$; $C = (c_1, \ldots, c_k)$ is the k global quality constraints which is brought out by users for the composite service; v_s is the start node that users bring up; v_g is the end node that users bring up.

Definition 2 Multi-constrained path: given the directed graph $G = (V, E, C, v_s, v_g)$, a path P from the start node to the end node is called a multi-constrained path.

1.3 Multi-constrained services composition

Based on the above definition, this paper addresses the problem that searching for an optimal service composition which meets multi-QoS attributes bound and users' functional requirements into a new problem: that of searching for an optimal multi-constrained path in a directed graph.

In many applications, the solution to an optimal multi-constrained path in a directed graph is to use the weighted combination of the linear function $q = w_1 q_1 + w_2 q_2 + \ldots + w_k q_k$ (q_i is the i-th QoS attribute of the service, $1 \leqslant i \leqslant k$, w_i is the weight of q_i), so that different QoS values are integrated into a single one and then using the Dijkstra algorithm to obtain the shortest path of the directed graph in polynomial time. But the Dijkstra algorithm does not consider the restriction of the shortest path, so it cannot meet the users' requirements. For example, the service price is less than c_1; the execution time is less than c_2, and the reliability is higher than c_4.

Since the problem that searching a multi-constrained ($k \geqslant 1$) shortest path is an NP-complete problem, when the number of nodes increases, the exhaustive search is impossible. In this paper, we use an approximate algorithm—simulated annealing algorithm that uses polynomial time to obtain a near optimum solution.

2 Simulated Annealing Algorithm for Service Composition

The simulated annealing algorithm (SA)[7] is a random searching algorithm based on a Monte Carlo iterative strategy brought up by Kirkpatrick in 1983. The method is inspired by experimental observations on crystallization from a melt. At high temperatures, the atoms in the melt are free to move around the sample. As the temperature is reduced, the atoms tend to crystallize into a solid. If the sample is annealed, i. e., cooled slowly, then the sample stands a better chance of forming a perfect crystal, which is the global minimum energy configuration of the system. Defects cost energy, and samples with defects correspond to local minima of the energy. Since there are an enormous number of possible configurations with defects, the energy landscape of the cooled solid is very complicated with numerous hills and valleys. Annealing allows the system to explore the landscape and settle down into one of the lower valleys. The SA is less dependent on the problem information, so it is widely used in many applications.

The SA for multi-QoS attributes constrained service composition is that:

1) Data preprocessing: Each QoS attribute of composite service P is normalized by scaling its values so that they fall within a small specified range 0 to 1.0.

2) Randomly choose $n_{\rm path}$ different service paths from the graph ($n_{\rm path} \leqslant n \times n$, n is the total number of the nodes in the graph) and use the SA for each path.

3) $T \leftarrow T_{\max}$: According to the functional combination required by users, randomly choose one service s_{ij} from the candidate service class s_i, until all the needed services have been chosen. According to the definition in section 1.1, we get the values for this path P; the path which cannot meet the need is deleted. E. g. the price of path P is higher than that which the users can afford, the time is beyond the limit time that users require, and get a new path P.

4) Randomly choose a new path P' from the neighborhood of path P. The neighborhood of path P is the path with $f(T)$ randomly changed nodes, $f(T) = \frac{T}{T_{\max}} \times$ (the total number of the nodes in path P).

if eval(P) is larger than eval(P')

then $P \leftarrow P'$

else choose P' with probability of $\exp\left(-\frac{\Delta eval}{T}\right)$

repeat this step for KT times.

5) Set $T \leftarrow rT$

if $T \geqslant T_{min}$

then do 4)

else do 3)

6) Secondary iterative optimization: Design function $g(Q_i(P)) = Q_i(P) - 0.1Q_i(P)Q_i(P)$, where $Q_i(P)$ is the value of the QoS of the current best solution; assume $g(Q_i(P))$ as constraint, go to 1), and search for the optimal solution. If the new solution is close to (abs (eval (P) − eval(P')) < exp(−6)), the best one for bestcount times or no solution can be found in count consecutive times, then end (In the experiment, bestcount = 8, count = 3).

In the above algorithm, T_{max} is the initial temperature, K is the iterative times for each value of T, r is the cooling rate, T_{min} is the solidification temperature, and eval(P) is the value of QoS bound for path P. The algorithm uses a linear weighted function as evaluation:

$$\text{eval}(P) = \sum W_i Q_i(P)$$

where $Q_i(P)$ $(1 \leqslant i \leqslant 5)$ represents the five QoS attributes in the path P, i. e., $Q_{price}(P)$, $Q_{du}(P)$, $Q_{rep}(P)$, $Q_{rel}(P)$, $Q_{av}(P)$; c_i is the overall QoS constraint for the composite service; W_i is the weight of $Q_i(P)$.

$$\Delta\text{eval} = \text{eval}(P) - \text{eval}(P')$$

3 Simulations and Analysis

We conduct a series of experiments to study the effectiveness of the SA on QoS bounded service composition. This paper randomly generates some services. Each service class includes 100 candidate services, and each candidate service has five QoS attributes. The simulation program sets five QoS parameters that range as follows: service price (10, 30), execution time (20, 40), reliability (1, 5), reputation (1,5), and availability (1,5), service reliability, reputation and availability in decreasing trend. That is 1 represents the best and 5 represents the worst.

The experiment uses the composite service as shown in Fig. 1. In the figure, A to F represent six services which implement different functions. $A \rightarrow B$ represents that A is executed before B. Service B, C, D and E are of a relationship between two elections.

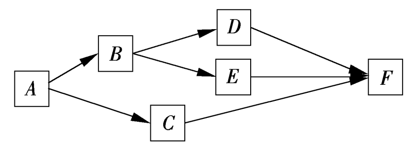

Fig. 1 Service composition P

The experiments are conducted on a computer with a CORE-DUO1 8 GHz CPU and 512 MB RAM, the developing environment is Eclipse 3. 2, Apache AXIS, and Tomcat 5. Simulation program sets the initial temperature $T_{max} = 10$, reducing 5% of the original temperature each time, and the solid temperature $T_{min} = 0.01$. It obtains the approximate optimal solution after 9 282 ms. Tab. 1 shows some QoS values of P. Fig. 2 shows a variety of QoS attributes in the composition process. They give us the confidence that our SA method can be used effectively to obtain good results which can satisfy multi criteria.

Tab. 1 The QoS data of an approximate optimal solution

Price	Duration	Reputation	Reliability	Availability
48.88	114.77	3.16	1.36	2.11
85.30	101.06	1.95	1.9	2.02
63.17	125.49	1.85	1.85	2.12
56.93	99.98	1.61	2.36	1.92
45.67	73.09	2.27	1.75	1.98
48.11	88.91	1.95	1.87	1.68
35.76	84.0	2.25	1.83	1.76
46.52	90.28	1.67	1.64	1.85
52.71	105.7	1.04	1.58	1.87
47.65	102.0	1.93	1.39	1.26

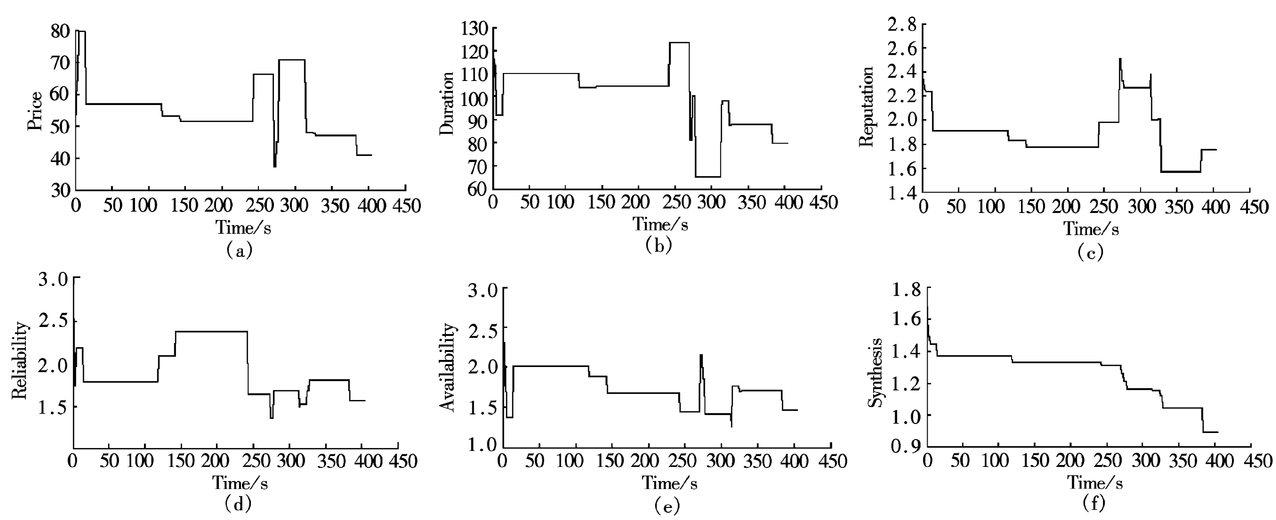

Fig. 2 QoS attribute in composition process

4 Conclusion

The QoS-aware web services composition is one of the key issues in the study of web services. The composite services, being dynamically selected and integrated, which meet the functional and non-functional attributes raised by users, have real practical value. This paper uses vectors to describe QoS attributes of services, and addresses the problem of searching for a optimal service composition which meets multi-QoS attributes bound and users' functional requirements into a problem of searching for an optimal multi-constrained path in a directed graph, and the designed SA to solve the problem. The results show that compared with the linear programming method and the Dijkstra algorithm, the proposed method can obtain a multi-constraint approximate optimal solution and has a lower time complexity. Our ongoing research includes the web services composition model description with context information and combination algorithms which cover exception handling.

References

[1] Curbera F, Roller R, Mukhi N, et al. The next step in web services[J]. *Communication of the ACM*, 2003, **46**(10): 29 − 34.

[2] Menasce Daniel A. Composing web services: a QoS view [J]. *IEEE Internet Computing*, 2004, **8**(6): 88 − 90.

[3] Zeng Liangzhao, Benatallah B, Dumas M, et al. Quality driven web services composition[C]//*Proceedings of the* 12*th International Conference on World Wide Web*. Budapest, Hungary, 2003: 411 − 421.

[4] Liu Yutu, Ngu Anne H H, Zeng Liangzhao. QoS computation and policing in dynamic web service selection[C]//*Proceedings of the* 13*th International Conference on World Wide Web*. New York, USA, 2004: 66 − 73.

[5] Canfora G, Penta M D, Esposito R, et al. An approach for QoS-aware service composition based on genetic algorithms [C]//*Proceedings of the* 2005 *Conference on Genetic and Evolutionary Computation*. Washington, DC, USA, 2005: 1069 − 1075.

[6] Tian M, Gramm A, Ritter H, et al. Efficient selection and monitoring of QoS-aware web services with the WS-QoS framework[C]//*Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence*. Beijing, China, 2004: 152 − 158.

[7] Kirkpatrick S, Gelatt C D, Vecchi M P. Optimization by simulated annealing[J]. *Science*, 1983, **220**(4598): 671 − 680.

基于模拟退火算法的 QoS 约束 web 服务组合

刘　青　张世龙　杨　锐　连祥鉴

(中国人民大学信息学院,北京 100872)

摘要:针对 web 服务组合中选择服务需感知服务的 QoS 属性问题,采用向量表示法描述原子服务及组合服务的 QoS 属性和用户提出的多项全局约束,把寻求满足多项非功能属性约束的最优服务组合问题转化为在有向图中搜索最优多约束路径问题,采用有向图对组合服务建模. 设计了多 QoS 属性约束的服务组合模拟退火算法,进行组合服务 QoS 属性的归一化处理和二次寻优. 实验结果表明该方法可求得满足各项 QoS 约束的可行解,模拟过程显示该算法以多项式时间复杂度选出近似最优解.

关键词:web 服务;服务质量;服务组合;模拟退火

中图分类号:TP311