

# A QoS-based service composition optimization method

Xiaolong Wang

Company of postgraduate management  
The Academy of equipment  
Beijing, China  
E-mail: [379139010@qq.com](mailto:379139010@qq.com)

Peng Wang

The Academy of Equipment  
Beijing, China  
[johnsonwp@sina.com](mailto:johnsonwp@sina.com)

Liang Chen

Company of postgraduate management  
The Academy of equipment  
Beijing, China  
E-mail: [sky35022123@gmail.com](mailto:sky35022123@gmail.com)

Peng Zou

The key laboratory  
The Academy of equipment  
Beijing, China  
E-mail: [zou\\_peng@163.com](mailto:zou_peng@163.com)

Jun He

The key laboratory  
The Academy of equipment  
Beijing, China  
E-mail: [hejun0123@sina.com](mailto:hejun0123@sina.com)

**Abstract**—With the development of web services technology, service composition gradually call people's attention, and how to improve the efficiency of Web service composition is the focus of research. In this paper, in order to improve the QoS (i.e., Quality of Service) of composition service, first of all using Pareto dominant elite screening collection of services to reduce redundant services; And then using genetic algorithm to combinatorial optimization of elite service set, is presented based on the Pareto dominant combined with genetic algorithm optimization method; Finally, the experiment proves combinatorial optimization methods for improving the efficiency of services is effective.

**Keywords**— Quality of Service; service composition ; Pareto dominant; Genetic Algorithm

## I. INTRODUCTION

In recently years, rapid development of web services technology for industrial, commercial and the third industry has provided an effective means to business process. However, the massively distributed computing services have become increasingly complex and single web service cannot completely meet the needs of complex business. The emergence of service composition technology provides a new train of thought for web application of the enterprise.

Web service composition is defined by a series of related business functions with a web service business processes and according to certain logic composed to build new capabilities to meet business needs of service[1], so called Composition Service. The web services involved in the composition called Component Service. Component Service itself can be either a composition service, also can be atomic service.

With the number of the same functional web services rapid growth, which provide a space for component services selection for the composition service. Thus, for a particular business process, how to choose the appropriate component services enable composition service has the best quality of service, has become the industry and academia's focus [2]. Literature [3] proposed a service composition QoS optimization problem method for the use of technology based on multi-attribute decision making method of the simple weighted addition, when the problem size is large, the solving efficiency is very low. Literature [4] proposed a service composition global optimization method based on hybrid linear programming , its solving efficiency is directly related to the size of service composition and the size of the service composition's candidate set. In the literature [5], the expansion of the web service QoS, using genetic algorithm to achieve the global QoS constraints of composition service's optimization problem. In the literature [6], for web service composition presents a genetic algorithm, the method introduces Immune to genetic algorithm, effectively improve the service's quality and processing speed. In the direct use of Heuristic Algorithm (such as genetic algorithms, etc.) for solving, computational complexity increases with the increasing of web services exponentially.

In this paper, heuristic algorithm for solving combinatorial optimization problems with the build candidate solutions to increase the number of services exponential growth problems and take Pereto dominant component services for pretreatment to reduce redundant services, obtaining a collection of elite service, and then use genetic algorithms for combinatorial optimization elite collection of

---

I. Supported by the National High-Tech Research and Development Program of China (863 Program) under Grant No. 2012AA012902.

services, thereby improving the efficiency of service composition.

## II. QOS MODEL OF WEB SERVICE

Factors affect composition service's QoS : (I). Component service's QoS, because composition service is consisting of component services, so the QoS of component service is particularly important for composition service's QoS; (II). The structure of composition service, the structural relationship between services in composition service is not static, service and service relationships between the different structures will affect the performance of composition service. This paper takes a fixed structure as the basis for the topology of composition service QoS optimization.

### A. QoS properties of web service

As the basis of composition service, component services directly affect the overall QoS of composition service. Currently ITU published Web service's QoS properties nearly a hundred kinds. This paper selects the five most commonly used QoS properties: Availability, Duration, Response time, Reliability, Cost (properties described in Table 1)for the evaluation of the quality of web services. Thus, QoS of web service can be described through a five-dimensional vector , that is:  $QoS = (AV, DU, RT, RE, C)$ .

Table 1. Web service's QoS properties describe table

Name	description
AV	The availability of service while communicating
DU	The duration of service to work
RT	Response time of service request
RE	The reliability of when communicating services
C	Service's consumption of costs

Because different QoS properties in dimensions and order of magnitudes have a big difference, and some of the QoS properties represent pros and cons or even completely opposite. Therefore, we must classify and normalize the QoS properties. The five properties can be divided into two categories: positive and negative properties.  $RT$  ( response time) and  $C$  (service costs) is the smaller the value, which means that the higher quality of service, are negative properties, which can be normalized as follows:

$$V_{mj} = \begin{cases} (Q_j^{\max} - Q_{mj}) / (Q_j^{\max} - Q_j^{\min}), & Q_j^{\max} - Q_j^{\min} \neq 0 \\ 1, & Q_j^{\max} - Q_j^{\min} = 0 \end{cases} \quad (1)$$

$AV$  (Availability),  $DU$  (Duration) and  $RE$  (Reliability) is the bigger the value, which means that the higher quality of service, are positive properties. For the positive properties can be normalized as follows:

$$V_{mj} = \begin{cases} (Q_{mj} - Q_j^{\min}) / (Q_j^{\max} - Q_j^{\min}), & Q_j^{\max} - Q_j^{\min} \neq 0 \\ 1, & Q_j^{\max} - Q_j^{\min} = 0 \end{cases} \quad (2)$$

The equation:  $m = (1, 2, \dots, N)$  represents the number of services,  $j = (1, 2, \dots, 5)$  represents the number of properties,  $Q_j^{\max}, Q_j^{\min}$  represents the maximum and minimum value of the  $j$ -th property, and  $Q_{mj}$  represents the  $j$ -th property of service  $m$ .

Thus, service QoS can be normalized vector  $\vec{V} = [V_1, V_2, V_3, V_4, V_5]$ ,  $V_1, V_2, V_3, V_4, V_5$  five variables are five web services QoS properties of the normalized represent. In order to evaluate the overall quality of service pros and cons, can use equation 3 to calculate the integrated service QoS values  $Q$ , the higher the  $Q$  value is the better quality of service.

$$\begin{cases} Q = \sum_{i=1}^5 \omega_i V_i \\ \sum_{i=1}^5 \omega_i = 1, \text{ and } 0 \leq \omega_i \leq 1 \end{cases} \quad (3)$$

In the equation:  $\omega_i$  is the weight of the  $i$ -th index.

### B. QoS calculation model of composition services

In the service composition, web composition services can be provided by the component services in a sequential structure, concurrent structure, selection structure, loop structure composition of four basic models. According to the literature [7], the results showed that concurrent structure, selection structure, loop structure can be converted to sequential structure, sequence structure to perform this service composition model:

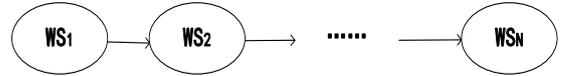


Figure 1 sequence structure of service composition

In Figure 1, the composition service QoS properties for calculating shown in Table 2:

Table 2. QoS models of sequence composition service

Properties	Equation
AV	$AV = \frac{1}{n} \sum_{j=1}^n AV_j$
DU	$DU = \sum_{j=1}^n DU_j$
RT	$RT = \sum_{j=1}^n RT_j$
RE	$RE = \prod_{j=1}^n RE_j$
C	$C = \sum_{j=1}^n C_j$

In table 2's equations,  $j$  represents the  $j$ -th component service,  $n$  represents the total number of component services. Finally, you can get composition service QoS properties normalized vector:  $\vec{V} = [V_1, V_2, V_3, V_4, V_5]$  and the comprehensive evaluation value  $Q$  can according to equations 1, 2 and 3 calculate.

### III. PARETO DOMINANCE AND GENETIC ALGORITHM OPTIMIZATION FOR WEB SERVICE COMPOSITION

#### A. Pareto dominance

Classic heuristic algorithm for solving combinatorial optimization problems of web services, there is a feasible solution to increase the number of component services with exponential growth problem. Example: For a three components of the composition service, when the number of candidate services of each component increased 2-times, the composition services will increase feasible solution  $2 * 2 * 2$  times. Thus, by every component service candidate solutions screening, elite candidate solution set, the remaining services for combinatorial optimization greatly reduce the number of candidate solutions, which greatly improves the efficiency of optimization algorithm. This paper introduces Pareto dominance for screening candidate components service, and then using a combination of genetic algorithm for optimal solutions. Pareto dominant thoughts<sup>[8]</sup> is described as:

Assuming a maximization problem  $f_i(\vec{X})$ ,  $\vec{X} = (x_1, x_2, \dots, x_n)$  and  $\vec{Y} = (y_1, y_2, \dots, y_n)$  are  $f_i(\vec{X})$  solution space's any two solutions:

Definitions I: Iff  $\forall i = \{1, 2, \dots, n\}: f_i(\vec{X}) < f_i(\vec{Y})$ , when called  $Y$  dominant in  $X$ ;

Definitions II: Iff  $\forall i = \{1, 2, \dots, n\}: f_i(\vec{X}) \leq f_i(\vec{Y})$  and  $\exists i = \{1, 2, \dots, n\}: f_i(\vec{X}) < f_i(\vec{Y})$ , when called  $Y$  weakly dominant in  $X$ ;

Definitions III: Iff  $Y$  is not dominant in  $X$  and  $X$  is not dominant in  $Y$ , when called  $X$  and  $Y$  are non-discriminatory;

Definitions VI: Pareto set  $P = \{\vec{X} \in \Omega \mid \neg \exists \vec{Y} \in \Omega: f(\vec{Y}) < f(\vec{X})\}$

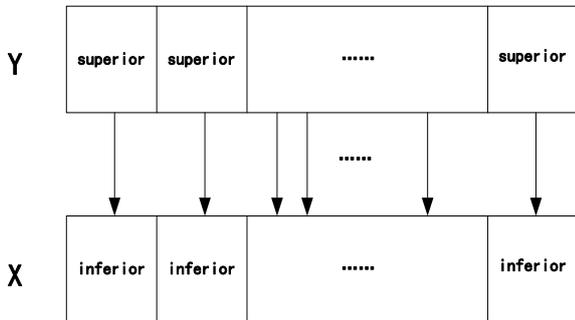


Figure 2. Pareto dominance schematic diagram

#### B. Web services based on Pareto dominance selecting

Assuming a service is provided by a combination of  $N$  component services, where the number of every component services is  $Num_i$ , for a specific component services, which may be described as follows QoS:

$$Q_{ij} = (AV_{ij}, DU_{ij}, RT_{ij}, RE_{ij}, C_{ij}) \quad (4)$$

$$(i = 1, \dots, N; j = 1, \dots, Max(Num_i))$$

$Q_{ij}$  represents the  $i$ -th component of the  $j$ -th service QoS property vector. If the  $i$ -th component services in the  $m$ -th service and the  $n$ -th service QoS as a function of each property satisfies the equation 5 Relationship, claimed that service  $m$  dominant in service  $n$ .

$$\begin{cases} f(AV_{im}) > f(AV_{in})(m, n \in [0, Num_i]) \\ f(DU_{im}) > f(DU_{in})(m, n \in [0, Num_i]) \\ f(RT_{im}) < f(RT_{in})(m, n \in [0, Num_i]) \\ f(RE_{im}) > f(RE_{in})(m, n \in [0, Num_i]) \\ f(C_{im}) < f(C_{in})(m, n \in [0, Num_i]) \end{cases} \quad (5)$$

Based on Equation 5 definition, the use of Pareto dominance, you can search for the elite collection of services by comparing candidate services, to some extent reduce the number of candidate services, thereby improving the efficiency of solving combinatorial optimization. Selection process diagram is as follows:

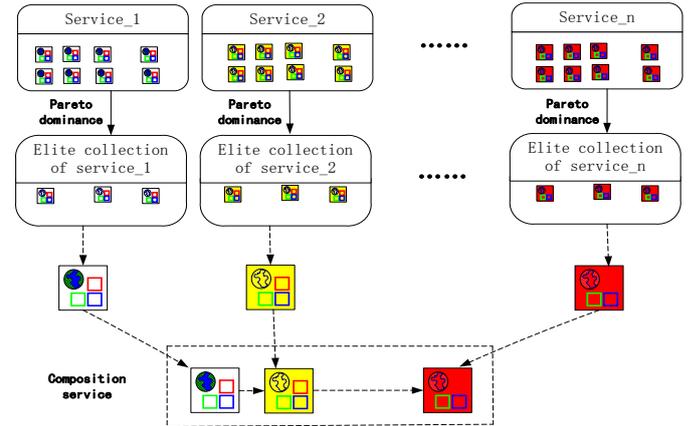


Figure 3. Pareto dominance search schematic diagram

#### C. Web services model for solving combinatorial optimization problems

Genetic algorithm is a classical heuristic algorithm proposed by J. Holland in 1975<sup>[10]</sup>. Genetic algorithm has superior global search capability, has been widely used in various optimization problems successfully. Therefore, this paper uses genetic algorithms to solve combinatorial optimization problems of web services.

The resulting set of services for the elite uses genetic algorithm to optimize QoS. Algorithm idea is as follows:

Assuming that services are provided by a composition of  $n$  component services, each of component services of the elite service number is  $m$ . The encoding is expressed as:

$X_{ij} (i = 1, 2, \dots, n; j = 1, 2, \dots, m)$ .  $i$  represents the  $i$ -th component service,  $X_{ij}$  represents the  $i$ -th component of the  $j$ -th candidate service's service number. Depending on the application scenario, the number of encoded bits with the the corresponding component services, the number of population size dynamically determined based on the problem size. Each chromosome coding randomly generated, determine the scope of  $[1, 2, \dots, m]$ . Fitness function associated with the target function is to determine whether the population genetic algorithm to generate the primary basis to meet the requirements [8, 9].

According to the QoS model of composition service built in Selection A of Part II. QoS-based Web service composition optimization problem can be described as: From the basic services the library to select the appropriate service instance in order to form the most comprehensive evaluation value  $Q$  satisfy users' functional requirements of a composition service.

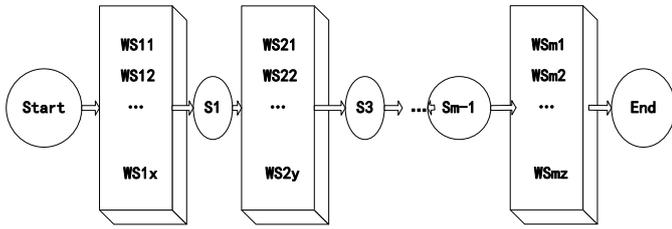


Figure 4. Process of Web service composition optimizing

As shown in Figure 4,  $S_i$  is middle matching node, block represents library including the same function atom services. An arrow represents an atomic service instance  $WS_{ij} (AV, DU, RT, RE, C)$ . Modeling service composing as a finite-dimensional vector  $\bar{X} = [x_1, x_2, \dots, x_m]$  though numbered each service in the libraries. For example, the composition services  $WS_{13} \rightarrow WS_{21} \rightarrow WS_{48} \rightarrow WS_{57}$  and  $WS_{19} \rightarrow WS_{34} \rightarrow WS_{45} \rightarrow WS_{52}$  can be described as  $\bar{X} = [3, 1, 0, 8, 7]$  and  $\bar{X} = [9, 0, 4, 5, 2]$ . Each bit value of  $\bar{X}$  represents the number of atomic service instance. Taking comprehensive evaluation value  $Q$  and composition service model  $\bar{X}$  as fitness and solution space to build the function model of web service composition optimizing. As shown follows:

$$\left\{ \begin{array}{l} fit(f(X)) = Q(X) = \sum_{i=1}^5 \omega_i V_i(X) \\ st. x_i \geq 0, \quad i = 1, \dots, m \\ V_1(X) > v_1 \\ V_2(X) > v_2 \\ V_3(X) > v_3 \\ V_4(X) > v_4 \\ V_5(X) > v_5 \end{array} \right. \quad (6)$$

Wherein:  $V_i(X)$  is the normalization vector of composition services' QoS properties  $(AV, DU, RT, RE, C)$ , which is determined by  $\bar{X}$ ;  $v_1, v_2, v_3, v_4, v_5$  are lowest

constraint offered by customers;  $m$  is the dimension of  $\bar{X}$ ; constraint of  $x_i$  is determined by the total number of instances in the  $i$ -th libraries.

For the specific fitness value is calculated as follows:

$$\left\{ \begin{array}{l} fit(X) = 0, (V_i(X) \leq v_i, i = 1, 2, 3, 4, 5) \\ fit(X) = Q(X) = \sum_{i=1}^5 \omega_i V_i(X), \text{ else} \end{array} \right. \quad (7)$$

This paper optimization criterion is the best individual in each generation fitness value  $GenMax(fit)$ , compare with the current maximum fitness value  $Max(fit)$ . If the continuous evolution after  $N$  generations, the best individual's fitness value is not significantly improved, which satisfies the equation 8. Iteration is terminated, otherwise, continue iterating until it reaches the maximum number of iterations.

$$\frac{GenMax(fit) - Max(fit)}{Max(fit)} \times 100\% < 0.01\% \quad (8)$$

#### IV. SIMULATION AND ANALYSIS

This paper through simulation experiments to verify the validity of the method. Experimental environment is Intel T7500 2.2G processor, 2048MB RAM and Windows 7 operating systems, programming languages is MATLAB. Service QoS property values in a certain range of values using randomly generated. Setting compositions in the simulation nodes matching services is 5, 10, 15, that is, 5, 10, 15 atomic services libraries, each library contains 100 candidate services.

In the genetic algorithm, the maximum iteration number of generation is  $T_{max} = 1000$ , crossover rate is 0.9, mutation rate of 0.1, population size defaults to 100. To verify the improved algorithm advantages in performance, while using standard genetic algorithm as comparative experiment, the simulation results shown:

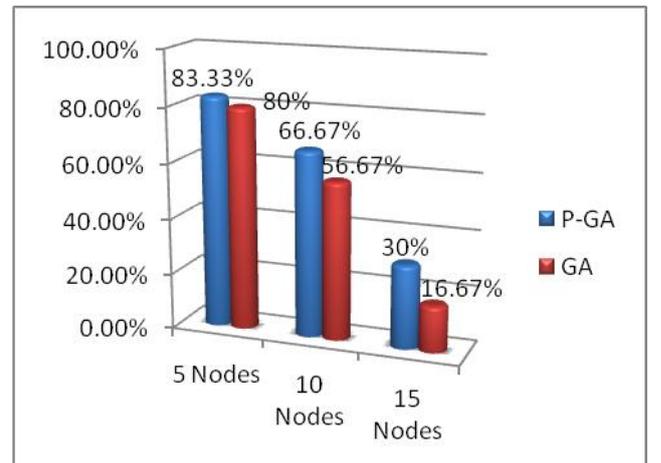


Figure 5. Comparison of optimizing success rate

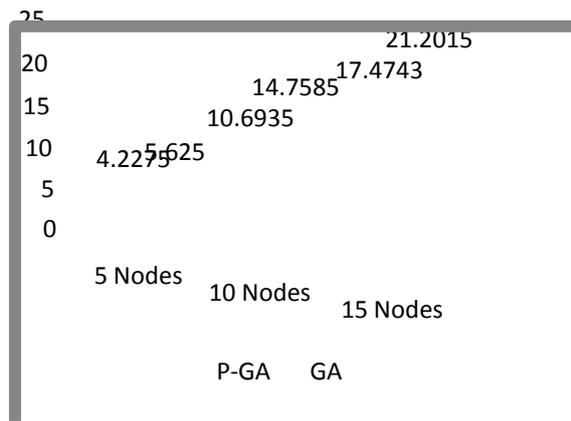


Figure 6. Comparison of optimization time

Figures 5 and 6 show that improved genetic algorithm compared to standard genetic algorithm significantly increased in the success rate and the composition time was also significantly reduced, proved that the effectiveness of the improved algorithm in the web service composition optimization problem solving.,

## V. CONCLUSION

This paper focuses on web services composition optimization problem of QoS properties. First, the component services use the Pareto dominance approach to select better service of the higher QoS values, get the elite collection of services for the next service combinatorial optimization, reduce a lot of redundant choice, reduce the amount of computation to improve the speed and efficiency of composition, better meet customer real-time needs. Then the services use genetic algorithms to optimize the composition of QoS properties. Finally, experimental verification, compared with the traditional genetic algorithm both success rate and run-time to prove the effectiveness of the algorithm.

## REFERENCES

- [1] ZHOU Qinglei, CHEN Mingzhao, ZHANG Bing. Multi-objective artificial bee colony algorithm applied in QoS-aware service composition optimization. *Application Research of Computers*, Vol. 29 No. 10 Oct. 2012, pp. 3625-3628.
- [2] LOU Yuansheng, TAO Zhenhong. Web service composition algorithm for QoS global optimization. *Engineering and Applications*, Vol. 47, No.8, 2011, pp. 207-210.
- [3] Zeng Liang-zhao, Benatallah B, Ngu A H H, et al. QoS-aware middleware for web service composition. *IEEE Transactions on Software Engineering*, Vol. 30, No. 5, 2004, pp. 311-327.
- [4] ZENG Liangzhao, Boualem Bernatallah, Marlon Dumas, Jayant Kalagnanam, Quan Z. Sheng. Quality Driven Web Services Composition. *Proceeding WWW'03 Proceedings of the 12<sup>th</sup> international conference on World Wide Web*. 2003. pp.411-421
- [5] SONG Xianfeng, LIU Junzhi. Optimization of GIS Web service Chaining Based on QoS. *Journal of University of Electronic Science and Technology of China*, Vol. 39, No. 2, 2010, pp. 298-301.
- [6] CHEN Liang, SUN Min. Web Services Composition Method

Based on Immune Genetic Algorithm. *Computer Engineering*, Vol. 36, No.10, 2010, pp. 226-227.

- [7] Cardoso J, Sheth A, Miller J, et al. Quality of service for workflow and web service processes. *Web Semantics: Science, Services and Agents on the World Wide Web*, Vol. 1, No. 3, 2004, pp. 281-308.
- [8] ZHENG Lijun. Based on genetic algorithm for multi-objective optimization and decision-making method. *National University of Defense Technology*. 2003, pp. 13-15.
- [9] XU Quanyuan, LU Ying. Pareto Optimal Solution Based on Multi-Objective Optimization Genetic Algorithms. *Computer CD Software and Applications*, Vol. 15, 2011, pp. 135-136.
- [10] WANG Lu. Research on Genetic Algorithms for Multi-Objective Optimization Algorithms. *Wuhan University of Technology*, March 2006, pp. 5-12.
- [11] DING Jianli, CHEN Zengqiang, YUAN Zhuzhi. On the Combination of Genetic Algorithm and Ant Algorithm. *JOURNAL OF COMPUTER RESEARCH AND DEVELOPMENT*, Vol. 40, No. 9, July 2003, pp. 1351-1354.