Computer Communications 34 (2011) 1137–1144

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

A novel heuristic algorithm for QoS-aware end-to-end service composition

Yuan-sheng Luo [a,*], Yong Qi [a,*], Di Hou [a], Lin-feng Shen [a], Ying Chen [b], Xiao Zhong [b]

[a] School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China
[b] IBM Research – China, Beijing 100193, China

ARTICLE INFO

Article history:
Available online 2 March 2010

Keywords:
Service composition
End-to-end
Multi-constraints optimal path
Meta-heuristic
Cross entropy

ABSTRACT

Many works have been carried out to find the efficient algorithms for QoS-aware service composition in recent years. Nevertheless, on one hand, some of these works only consider the local QoS attributes in Web services composition; on the other hand, some ideas derived from QoS selection algorithms for network routing are directly applied in service composition without any adaption. A service composition model with end-to-end QoS constraints has been presented in this paper. An improved heuristics HCE based on the observation of characteristic of end-to-end service composition is proposed as a novel solution. Simulation results reveal the better performance of proposed heuristic compared to the other two heuristics, HMCOP and generic CE algorithm.

1. Introduction

Research on Service Oriented Architecture (SOA) and Service Oriented Computing (SOC) brings a promising technique to create value-added business applications composed by dynamically selected individual services. This technology is so called service composition. Nevertheless, the terminology "service composition" not only is used in SOA and SOC in application level, but also appears in autonomic communication system [1] at network level. There are two differences between their functionality attributes. Firstly, the functionalities of services at application level are more abundant than those of services at network level. The services at network level are commonly the communication services, while the services at application level are designed for all kinds of business domain. Secondly, the application level services are loosely-coupled in comparison with the network level services. For example, the autonomous systems at network level share their information and cooperate with each other by Border Gateway Protocol (BGP), while the Web services at application level provide functionality to users independently and communicate with each other by end-to-end message exchange using network protocols. Due to the differences between the functionality attributes of services at the two levels, the non-functionality attributes, which commonly refers to quality of service (QoS), are also different. Without special emphasis, we focus on service composition with end-to-end QoS constraints in this paper, especially the QoS-aware Web service composition, which is not only a problem for local QoS policy guarantee of composite service, but also a optimization procedure for network QoS delivering.

QoS-aware Web service composition (QWSC) has drawn much attention in recent years. Some works have been carried out on service selection algorithms for composing services with multiple QoS constraints to find an optimal solution, which is a NP-complete problem [2–6]. Zeng et al. suggested solving this problem by Linear Integer Programming (LIP) [2]. Although LIP is an optimal algorithm, its computation time tends to grow exponentially with the size of the problem instance, thus it is limited to use the LIP algorithm in real scenarios, especially in time-critical scenario. And the network constraint is omitted in this paper. Several researchers put forward heuristic algorithms to find a near-optimal solution. Berbner et al. [4] presented an algorithm using the result of linear programming relaxation of LIP as the heuristic hint. The network constraint also is omitted in this paper. Yu et al. [3,6] presented a heuristic algorithm, which was developed by [7] and uses the concept named 'aggregate resource', to find a near-optimal solution for QWSC. They also consider the network constraints of Web service and propose another heuristic algorithm deriving from [8] named HMCOP to solve the service composition with end-to-end constraints. When the network attribute is considered in QWSC, the computation model becomes Multi-Constraints Optimal Path (MCOP) problem, which cannot be solved by LIP anymore. The MCOP has been studied for QoS guarantee in packet network [8] and QoS-aware service composition in autonomous network communication [1]. As aforementioned, there are some differences between the service composition at network level and the Web service composition. There is a type of heuristic algorithms called meta-heuristics which are adaptive to most of scenarios of combinatorial optimization problem [9], such as the evolutionary computation, tabu search, simulated annealing, ant colony optimization and cross-entropy algorithm. It is widely studied to utilize these meta-heuristics for QWSC

* Corresponding authors. Tel./fax: +86 029 82668971.
E-mail addresses: luodyx@msn.com (Y.-s. Luo), qiy@mail.xjtu.edu.cn (Y. Qi).

0140-3664/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2010.02.028

[5,10,11]. Nevertheless, most of these works just treated the QWSC as the common combinatorial optimization problem and there is less work adopting special optimization strategy by observing the different characteristics of service composition with end-to-end QoS constraints.

In this paper, we firstly introduce the decentralized Web service composition system architecture; according to the system architecture, the computation model and QoS model of QWSC is presented. After that, two efficient heuristics used in QWSC, HMCOP and CE algorithms are studied and on the basis of these two algorithms, an improvement strategy is proposed, which is intuitively derived from the observation of the characteristic of QWSC. The simulation experiments will be conducted to compare these three algorithms and the results will reveal the better performance of proposed strategy than original ones.

2. System model

2.1. Decentralized system topology

In [22], we assume the customers just communicate with a broker middleware which delegate the customers to communicate and interact with all services. In this paper, a more general situation is considered.

As depicted in Fig. 1, customers interact with one of the broker middlewares which spread at different locations in the Web cloud. In each broker there are a pack of services advertised or registered. Each broker can communicate with each other by end-to-end message exchange. A formal model of service composition plan is defined as follows:

Definition 1. A Web service composition plan is a sequence of $n+2$ tasks $(T_0 T_1 \ldots T_{n+1})$. A task T_i has input parameters collection $T_i \cdot I$ and output parameters collection $T_i \cdot O$. I represents input parameters of customer and Q represents customer's requests. Where:

(1) $T_0 \cdot I \equiv T_0 \cdot O \equiv I;\; T_{n+1} \cdot I \equiv T_{n+1} \cdot O \supseteq Q$;
(2) For any $T_i (i = 1, 2, \ldots, n+1), T_i \cdot I \subseteq \bigcup_{j=0}^{i-1} T_j \cdot O$.

The Web service composition plan can be worked out by customers, domain experts or automated planner. Each task is correlated with an abstract service. An abstract service consists of the Web service-like operations a particular application domain typically offers [19]. Each abstract service is correlated with a collection consists of concrete services with overlapping functionality but different QoS attributes, as shown in Fig. 2. In the rest of this paper this model is taken as the default context.

2.2. Computation model

The multi-dimension multi-choice 0–1 knapsack problem (MMKP) can be reduced into the Web service composition problem with multiple QoS constraints. The computation model can be described as following linear integer programming formula:

$$max \quad \sum_{i=1}^{n} \sum_{j=1}^{l_i} F_{ij} x_{ij}$$
$$\text{subject to} \quad \begin{cases} \sum_{i=1}^{n} \sum_{j=1}^{l_i} r_{ij} x_{ij} \leqslant R_k, & k = 1, \ldots, m \\ \sum_{j=1}^{l_i} x_j = 1, & i = 1, \ldots, n \\ x_{ij} \in \{0, 1\} \end{cases} \tag{1}$$

where F_{ij} and r_{ij} are non-negative numbers. x_{ij} is the decision variable and it is correlated with $service_{ij}$. F_{ij} is profit or utility value concrete $service_{ij}$ can provide and it is called as objective function in LIP. r_{kij} is the kth dimension resource value of $service_{ij}$, R_k is the total constraint of kth dimension resource value. There are n abstract classes, l_i concrete services in each $class_i$ and m dimension re-

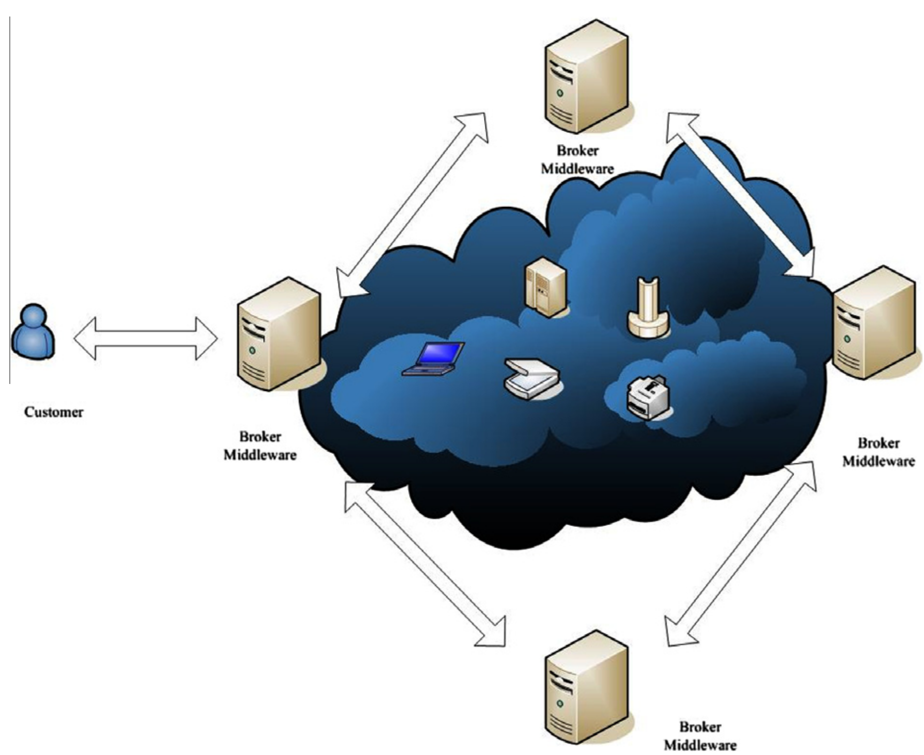

Fig. 1. System architecture.