QoS-based Web Service Composition Applying an Improved Genetic Algorithm (IGA) Method

Pooya Shahrokh, Department of Computer Engineering, Zanjan Branch, Islamic Azad University, Zanjan, Iran
Faramarz Safi-Esfahani, Faculty of Computer Engineering, Najafabad Branch, Islamic Azad University, Najafabad, Iran

ABSTRACT

In recent years, it has been made possible to compose exiting services when a user’s request cannot be satisfied by a single web service. Web service composition is faced with several challenges among which is the rapid growth in the number of available web services leading to increased number of web services offering the same functionalities. The difference between similar services is Quality of Service (QoS) consisting of various non-functional factors such as execution time, availability, security, etc. As a result, multiple choices are possible in making a composition plan. Among numerous plans, selecting a composition plan that fulfills customer’s requirements has become an important and time-consuming problem. In this paper, the researchers propose a semi-heuristic genetic algorithm that is a combination of both a heuristic method and the genetic algorithm. This heuristic method changes chromosomes based on unsatisfied constraints. Research findings show that the proposed method can be applied to find a composition plan that satisfies user’s requirements more efficiently than other methods.

KEYWORDS
Genetic Algorithm, Heuristic Methods, QoS, Quality of Service, Web Service, Web Service Composition

1. INTRODUCTION

Most applications are sophisticated in that they cannot be performed using a single, and isolated web service. Therefore, Web users often need to compose different services to achieve a more complex task that cannot be fulfilled by an individual service. Service composition refers to the process of developing a composite service in turn (Wang L. J., 2012). In fact, web service composition creates new functionalities through aggregating different services that is a time consuming problem and is classified as NP-hard (Canfora, 2005) (Wang L. J., 2012). Web service composition based on qualitative criteria aims at finding the best combination so that the concentration of the desired qualitative aspects of the whole system is compliant. Besides, local and general optimization can be considered. In local optimality, the highest QoS is selected in the partner service for each task. This method works greedily and quickly, but the service is not guaranteed to be optimal. Though this method has a greedy function and a high speed, it does not guarantee the optimality of the whole composition. In general optimality that we look for services for each act, the juxtaposition of which ultimately causes the composite service QoS to have the highest value. If the number of candidate services and the number of processes increase, the investigation of individual composite plans will
be practically unfeasible. Hence, there is a need for an approach that is able to find an appropriate combination within a reasonable time.

In (Ko, 2008), constraint satisfaction based solution which combines simulated annealing approach with Tabu search has been proposed. Tabu search is used for generating neighbor plans and simulated annealing heuristic is applied for accepting or rejecting the neighbor plan. High probability of getting stuck in local optimum is thought to be this method’s problem because it is unable to work on more than one composition plan at the same time, while this probability in methods such as genetic algorithms is able to work on several composition plans.

The randomness characteristic of GA can help to escape from local optimums, but on the other hand, it can cause to get far away from optimum solution. Due to this characteristic, we add several heuristic methods to control the randomness of GA in such a way that mutation is carried based on the quality parameters of the issue so that the algorithm does not move away from the optimal space of the optimal combination without being trapped by the local optimality. The dynamic weighting is also adopted to determine the required quality parameters.

The researchers compare the execution time of the proposed algorithm with the pure genetic and PSO algorithms. The experiment results signify that this modification makes the algorithm to be more time-efficient.

The remainder of this paper is organized as follows. In section 2 and 3, the related studies and the QoS model of web service composition are presented. In section 4, the researcher proposes the improved GA-based composition algorithm. Section 5 provides the computational results of the algorithm, and finally conclusion and future work are given in section 6.

2. CONCEPTS

SOA has turned into one of the predominant paradigms in business to enable efficient and flexible business processes mostly implemented via web service. W3C defines a web service as “a software system designed to support interoperable machine-to-machine interaction over a network”. It is a XML-based, self-described software entity which can be advertised, located, and used across the internet using a set of standards such as SOAP (Zeng, 2004), WSDL, UDDI (Dustdar, 2005) (Syu, 2012).

Web service providers register their web services into UDDI registries. SOAP is a XML-based protocol specification for exchanging information between peers in a decentralized, distributed environment. It also provides a simple and lightweight mechanism for communication between web services. SOAP can form the foundation layer of a web services protocol stack. WSDL is employed to describe the interfaces of all web services regardless of the underlying technology. WSDL defines services as collections of network endpoints or ports. When a service provider wants to register a web service at a UDDI server (web service directory), it describes the web service by WSDL and puts it in the UDDI registry. When service requesters come to a UDDI server to look for a web service, the WSDL file of web services are presented to them (Dustdar, 2005).

Figure 1 shows the standard architecture of web services. The web service model consists of a service provider, service registry and service consumer entities. The service provider offers the web service registry, the service registry contains information about the service provider and the service consumer retrieves the information. To show the concept of service composition, consider a composition service that consists of N abstract Tasks:

\[ \text{task}_1, \text{task}_2, \ldots, \text{task}_n \]  

In order to perform each task, there are a lot of candidate web services with the same functionalities and different non-functionalities that are shown as follows: CS\(_1\), CS\(_2\), ..., C\(_n\). Consequently, there are CS\(_1\)(task\(_1\))*CS\(_2\)(task\(_2\))*CS\(_1\)(task\(_n\)) ways to combine the web services. There will be various combinations
An Empirical Investigation of Mobile Banking Adoption in Developing Countries
www.igi-global.com/article/an-empirical-investigation-of-mobile-banking-adoption-in-developing-countries/111077?camid=4v1a

Application Integration: Pilot Project to Implement a Financial Portfolio System in a Korean Bank
www.igi-global.com/chapter/application-integration-pilot-project-implement/18374?camid=4v1a