International Journal of Web Services Research
Volume 16 • Issue 2 • April-June 2019

Parameter Tuning for S-ABCPK

An Improved Service Composition Algorithm Considering Priori Knowledge

Ruilin Liu, School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China
Zhongjie Wang, School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China
Xiaofei Xu, School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China

ABSTRACT

QoS-aware service composition problem has been drawn great attention in recent years. As an NP-hard problem, high time complexity is inevitable if global optimization algorithms (such as integer programming) are adopted. Researchers applied various evolutionary algorithms to decrease the time complexity by looking for a near-optimum solution. However, each evolutionary algorithm has two or more parameters, the values of which are to be assigned by algorithm designers and likely have impacts on the optimization results (primarily time complexity and optimality). The authors' experiments show that there are some dependencies between the features of a service composition problem, the values of an evolutionary algorithm's parameters, and the optimization results. In this article, the authors propose an improved algorithm called Service-Oriented Artificial Bee Colony algorithm considering Priori Knowledge (S-ABC_{PK}) to solve service composition problem and focus on the S-ABC_{PK}'s parameter turning issue. The objective is to identify the potential dependency for designers of a service composition algorithm easily setting up the values of S-ABC_{PK} parameters to obtain a preferable composition solution without many times of tedious attempts. Eight features of the service composition problem and the priori knowledge, five S-ABC_{PK} parameters and two metrics of the final solution are identified. Based on a large volume of experiment data, S-ABC_{PK} parameter tuning for a given service composition problem is conducted using C4.5 algorithm and the dependency between problem features and S-ABC_{PK} parameters are established using the neural network method. An experiment on a validation dataset shows the feasibility of the approach.

KEYWORDS

Artificial Bee Colony (ABC) Algorithm, Parameter Tuning, Priori Knowledge, QoS-Aware Service Composition

1. INTRODUCTION

In the service era, people's lives rely more and more on the services delivered via the Internet. An evident trend is that user requirements are increasingly complicated, and no single service could completely fulfill a coarse-grained requirement (Liu and Wang, 2012). It is necessary to compose the functionalities of multiple web services. This is called "service composition" (Zeng, Benatallah et al. 2003; Rao and Su, 2005; Dustdar and Schreiner, 2005). A typical service composition problem is to consider a set of end-to-end QoS constraints (local and global) raised by customers and look for an optimal composite solution to satisfy these QoS constraints or called QoS-aware service composition problem (Zeng, Benatallah et al. 2004; Alrifai and Risse, 2009).

A key issue in solving QoS-aware service composition problem is the time complexity. This is because the number of candidate Web services is likely to be large (Canfora, Di Penta et al. 2005, July), and the selection of the best services from candidates to compose a solution that meets the QoS

DOI: 10.4018/IJWSR.2019040105

constraints is time-consuming and has been proven to be NP-hard (Mabrouk, Beauche et al. 2009). Researchers have proposed various algorithms to decrease the time complexity, such as skyline-based approach (Alrifai, Skoutas et al. 2010), approximation-based approaches (Yu, Zhang et al. 2007), fuzzy logic approaches (de Gyves Avila and Djemame, 2013), cuckoo search approaches (Boussalia and Chaoui, 2014), evolutionary algorithm-based approaches, etc. Among them, evolutionary algorithm shows better performance and has been widely applied. There are many frequently used evolutionary algorithms, such as genetic algorithm (Canfora, Di Penta et al. 2005, June), particle swarm optimization (Chen and Wang, 2007; Zhang, 2014). Among them, Artificial Bee Colony (ABC) algorithm is the simplest but a relatively powerful one (Karaboga and Basturk, 2008; Wang, Wang et al. 2013; Liu, Wang et al. 2014). It has only three control parameters, besides the procedure of ABC is simple to understand and implement (Karaboga and Akay, 2009; Akay and Karaboga, 2009). The summary and comparison among the several evolutionary algorithms are shown in Table 1. Researchers have applied ABC into service composition problems and have confirmed its superiority compared with other evolutionary optimization algorithms (Wang, Wang et al. 2013; Xu and Liu, 2014).

However, existing ABC researches have not fully addressed a key issue, namely, parameter tuning. More specifically, how to assign appropriate value to the control parameters? In the past experiments, we have found there is a certain dependency between the parameters' values and the final optimization results. Still, current ABC algorithm designers have to rely on their own experiences to manually and casually assign values to the parameters.

Akay and Karaboga studied the ABC parameter tuning problem (Akay and Karaboga, 2009) and have got some positive conclusions, e.g.:

- As ABC uses and exploitative process to efficiently converge minima and an explorative process to ensure sufficient diversity in the population, it does not need a large colony size (or called SN) to solve optimization problems with high dimensions.
- The second parameter Limit dictates the occurrence of scout bees that are responsible for ensuring the diversity of the population. For relatively small-sized populations, Limit should not be low to avoid the situation that there are many randomly-generated solutions in the present populations. However, since large populations have sufficient diversity, Limit does not stand out in these situations.

Table 1. A summary of evolutionary algorithms

Algorithm	Searching carrier	Solution carrier	Optimization method	Control parameters
Particle swarm optimization	particles	particle position	using self-experiences and swarm experiences to amend each particle move	SN, w, C_1, C_2, V_{max}, V_{max}, MCN
Ant colony optimization	ants	path of ants	using historical paths of the swarm to guide individual ants' path seeking	SN, α, β, ρ, η, MCN
Simulated annealing	molecules	molecule position	time-varying probability of accepting inferior new positions	SN, T_0, λ, β, MCN
Genetic algorithm	individuals	Chromosome coding	selection, crossing, and mutation of individuals	SN, α, β, MCN
Artificial bee colony algorithm	foraging bees	food source position	exploration and exploitation to food sources	SN, $Limit$, MCN

* *SN* is the size of searching carriers, *MCN* stands for the termination condition. The meaning of other parameters can be found in related books.