

Research on the Qos Optimization of Web Service Composition Process

Xiangshan Li, Ziyi Su*

College of Computer Science and Information Technology, Northeast Normal University, Changchun
Email: zimai0410@sina.com

Received: Jan. 21st, 2014; revised: Feb. 16th, 2014; accepted: Feb. 27th, 2014

Copyright © 2014 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

Abstract

With the fast development of the cloud computing, the research on the Web Service has faced more challenges. Generally, traditional Web Service composition algorithms only take the consideration of the functional requirements, but care little about the non-functional requirements. This paper summarizes the research about the Web Services composition, most of which are based on the PSO algorithm, simulated annealing algorithm, skyline algorithm, etc. In this paper, we attempt to apply CS algorithm to the Web Service composition problem. Through experimenting on the services sets which are built by choosing similar services in the QWS dataset, comparing with the traditional method which uses PSO algorithm on the Web Service composition problem, the feasibility of the proposed method is verified.

Keywords

Qos; Web Service Composition Algorithm; Cs Algorithm

Web服务组合QoS最优化问题研究

李香善, 苏子义*

东北师范大学计算机科学与信息技术学院, 长春
Email: zimai0410@sina.com

收稿日期: 2014年1月21日; 修回日期: 2014年2月16日; 录用日期: 2014年2月27日

*通讯作者。

摘要

随着云计算的蓬勃发展, Web 服务的研究和应用也迎来了新的挑战。传统的服务组合一般只考虑到服务的功能性需求, 对服务的非功能性需求和总体服务质量(QoS)考虑较少。本文首先对近年来基于 QoS 的服务组合问题进行了归纳总结, 目前大多数研究都是基于粒子群算法、模拟退火算法、skyline 方法等。本文做出新的尝试, 提出将布谷鸟算法应用于服务组合问题, 通过使用从 QWS 数据集中挑选出的相似服务构建服务集进行实验, 与传统的应用粒子群算法的服务组合方法进行对比, 对提出的方法的可行性进行了验证。

关键词

QoS; Web 服务组合算法; 布谷鸟算法

1. 引言

随着云计算与 Web 服务的发展, 越来越多的学者对 Web 服务组合 QoS 问题进行了多方面的探索。Web 服务组合问题从一些功能相似或相同但 QoS 不同的服务中选择一个或多个服务组成 Web 工作流以满足最终的业务需求, 通常包括功能需求与 QoS 需求。QoS 表示的是一个服务的服务质量, 根据国际标准 ISO8402 的相关定义, 它代表的一些非功能性的属性, 目前研究中最常涉及到的 QoS 属性有: 服务响应时间、服务成功的概率、服务价格、服务可信度等。关于 QoS 的研究目前主要关注两方面:

1) 从 QoS 可信度方面。由于 Internet 和云计算技术的快速发展, 网络上的 Web 服务数量日益增多, 而这些 Web 服务所对应的 QoS 也相应增加, 在如此多的 Web 服务中, QoS 的可信度受到了不小的挑战。如何得到令用户满意的可信的 QoS 就成为了一个重要的研究问题。如文章[1]中提出了新的 web 服务可靠性预测方法。通过扩展 UDDI 模型, 并且增加新的角色, 通过对历史数据的分析来预测服务的质量。并且提出基于事件驱动的泊松适配抽样来收集 QoS 反馈, 降低预测服务器的负载。

2) 随着 Web 服务的增多, 网络上具有相同或相似功能的 Web 服务逐渐增多, 但往往都具有不相同的 QoS 值, 而单一的 Web 服务又不能提供令人满意的服务。从功能覆盖的角度, 采用单一 Web 服务实现过于复杂的业务逻辑, 不仅会使得单一 Web 服务功能过于复杂而难于维护, 更甚者在云计算的业务合作大背景下, 不采用业务组合而实现一个完善功能是不现实的, 目前的在线预订、在线支付等业务即是基于此而采用 Web 服务组合的方式来达到跨企业的业务组合。同时, Web 服务所提供的高内聚低耦合的软件系统架构方式也提高了功能和业务模块的利用率, 因此大部分情况下用户想要达成某种目标, 都需要通过组合多个 Web 服务而达到。因而目前越来越多的研究关注于如何进行 QoS 最优的服务组合。基于 QoS 最优的服务组合问题, 就是根据已知的组合流程选取具体服务, 使得整个流程的总体 QoS 达到最优。本文主要针对基于 QoS 最优的 Web 服务组合问题进行研究。

2. 研究背景

目前对于基于 QoS 的 Web 服务组合问题的研究大致可以分为两类。第一类是关注如何在每个抽象服务中选取一个具体服务, 通过组合来满足用户的功能性需求。目前大部分研究都是基于第一类的研究。因此在本节中将主要对第一类研究进行更细致的分类。第二类是如何在每个抽象服务中选取具体服务的集合来满足用户的功能性需求。文章[2]中作者提出, 处于对 Web 服务非功能性属性提升的考虑, 通过平衡遍历的方法来聚合具有相似的功能的服务, 即从服务群中选择多个具有相同功能的具体服务来共同完

成某一项任务，从而提升整个服务组合的性能。

2.1. 基于服务选择策略分析

基于服务选择策略，可以将 Web 服务组合问题分为基于全局最优的原则和基于局部最优的原则。目前大部分的服务组合算法都是基于局部最优的算法。局部最优算法即单独考虑每个候选服务的 QoS 信息，并进行加权和排序，并根据此加权和选择一个满足局部约束条件的加权和最大的服务。例如文章[3]中提出了一种用于服务组合的迭代选择算法，该算法聚合局部最优服务，按照服务组合逻辑执行顺序迭代，因而在服务组合执行前或者运行时不作任何修改。

基于全局最优的原则跳出了只考虑单个抽象服务的局限，而是考虑整个服务组合的钻则。可以满足多个目标函数，即满足多个功能的需求，更加符合用户的需求。但是一般这种算法的计算量都比较大，且复杂度高，随着服务数量的增加，整个服务组合的性能都会受到影响。例如文章[4]中，在基于 QoS 的 Web 服务组合问题中，提出一种采用路径模板编码机制的遗传算法来解决多路径全局优化问题，通过交叉变异从而扩大了种群的多样性，扩大搜索范围，并且用实验证明了该算法的优越性。或者有些研究致力于将全局 QoS 约束分解的方法。例如文章[5]中，提出一种全新的基于全局 QoS 约束分解的动态选择服务方法，将全局 QoS 约束自适应地分解为满足用户偏好的局部约束，然后再利用局部最优的方法获得最合适的组合服务。文章[6]也是将研究重心放在了将 QoS 感知的服务组合问题分解为两个相对简单的问题，即服务质量约束的分解问题和局部选取上面。在对 QoS 进行建模的过程中，根据不同的 workflow 结构和权重对 QoS 属性进行聚合，从而在服务组合过程中考虑到了多种 QoS 属性，并通过与其他的全局最优算法的比较，证明了该方法的优越性。

基于局部最优的原则，即对于单一的服务结点功能需求，按照服务的 QoS 参数信息进行加权排序，并以此选择加权和最大的那个服务执行组合服务结点的功能。由于单纯依靠局部最优的原则或者全局最优的原则，都无法满足现在用户的需求，因而现在很多研究都趋向于将这两种方法进行融合，从而达到更好的求解效果。

2.2. 基于问题模型建立方式分类

可以分为多目标转化为单目标的决策优化问题和多目标的决策优化问题。多目标转化为单目标的决策优化问题，是将多个目标函数线性规划为单目标函数。大部分对于该问题只能得出一个最优解，无法为用户提供选择的空间，当最优解无法执行的时候也会导致任务的失败。如[7]，是将服务组合问题转化为线性规划问题，对 Web 服务的 QoS 进行线性规划，形成一个新的效用函数，从而将服务组合问题进行形式化的描述，再通过对粒子群算法的遗传算法变异进行优化。也有些文章在只能得出一个最优解这一问题上做出了改进，例如文章[8]为将单目标的决策优化问题，将目标函数表示为一个关于 QoS 的线性函数，通过在服务执行中的选择——动态绑定，取得最优解集。文章[9]提出，将服务动态选择的全局优化问题转化为带 QoS 约束的多目标组合优化问题，通过智能优化算法求解，最后产生满足约束条件的优化服务组合流程集。

而多目标的决策优化问题由于自身的特点，多数情况下直接就可以生成含有多个解的最优解集，用户可以根据自己的需要选择更适合自己的解。如文章[10]中提出运用归档式多目标模拟退火算法，设计出一种新的 QoS 全局最优 Web 服务选择算法，同时优化多个目标函数，以生成一个 Pareto 最优解集，由决策者及服务使用者根据实际需求和偏好在 Pareto 集中选择一个最优解。这种方法也更加符合人们对于服务组合问题的需求，也不会在使用时对各属性的权重分配问题上造成过多的负担。但是基于这方面的考虑也为服务组合问题的研究增加了难度。

2.3. 基于服务选择方法的执行期分类

分为在执行前进行的服务选择和在执行中进行的服务选择。前者不能适应不可预知的服务环境的变化, 后者的计算量大难以控制整个流程的最优性。目前大部分的研究都在执行前进行服务选择, 在执行中进行的服务还很少。动态 Web 服务组合问题的研究, 就是出于对这方面的考虑。目前很多研究算法也致力于将两种方法融合。例如在文章[8]中, 作者就致力于结合这两种方法, 通过服务执行前的服务选择和服务执行中的动态服务选择提高 Web 服务执行效率和服务成功率, 并通过实例分析证明了该方法的可行性。

2.4. 基于 QoS 的 Web 服务选择采用的优化算法分类

基于优化算法, 可以分为穷举法、贪婪算法和基于生物学的优化算法。由于基于 QoS 的 Web 服务选择问题已经被证明为是一个 NP 难的问题。使用穷举法的时间消耗过大, 贪婪算法的效果不甚理想。所以大多数研究者在解决该问题时都采用了优化方法, 从而更快更好地得到令用户满意的解。在优化方法的选择上有着不同的策略。由于智能优化方面研究的快速发展, 在 Web 服务选择问题上, 可以采用遗传算法、粒子群算法、蚁群算法、模拟退火算法等比较常见的生物优化方法。例如文章[7]、[9]和[11]都是采用粒子群算法对服务组合问题进行求解。粒子群算法属于目前较为流行的在解决基于 QoS 的服务组合优化问题时采用的方法。文章[7]是在粒子群优化算法的基础上, 对粒子群进行了遗传变异从而得到更好的优化效果。文章[11]中提出执行计划选择的分支界限算法(BB4EPS)来解决 QoS 感知的服务组合问题。对一些概念和意见提出理论上的模型, 利用效用函数作为目标函数。实际上许多研究者在使用粒子群算法对 QoS 感知的服务组合问题进行优化时, 都是交叉使用了一些遗传算法, 这样使得求解的效率得到了显著的提高。其他研究者也在使用其他的生物学优化算法来对该问题进行求解的领域中做出了研究。例如文章[10]使用模拟退火算法来对该问题进行求解, 并且得到了另人满意的结果。作者使用归档式的多目标模拟退火算法, 通过比较不同参数情况下的 CPU 开销和算法产生的全局 QoS 属性目标函数值, 证明了该方法的可行性。文献[12]中研究了蚁群算法在服务组合优化系统中的应用, 在基本蚁群算法的基础上提出了 MDPACO 算法, 该算法能适应服务组合优化过程中 Web 服务的动态性、不稳定性以及多种 QoS 属性限制等问题。

目前, 不论是使用粒子群算法还是蚁群算法, 在一定程度上都可以解决服务组合问题。但由于这些算法过程都比较复杂, 并且涉及的参数比较多, 为算法的实现增加了难度, 服务组合的结果受参数选择的影响比较大。基于此本文提出了将布谷鸟算法应用于服务组合问题, 并且为了算法效率的效率, 在随机取值的过程中, 加入了服从正太分布的随机取值的方法对原始布谷鸟算法进行改进。本文通过使用从 QWS 数据集中挑选出的相似服务构建服务集, 从而对提出的方法的可行性进行了验证。结果证明, 布谷鸟算法在解决服务组合问题上可以有很好的可行性, 同时, 对参数取值方法改进后的布谷鸟算法具有更好的性能。

3. 基于布谷鸟算法的服务组合算法

布谷鸟算法是由 Xin-She Yang 在 2009 年在文章[13]中提出的一种全新的启发式优化算法。本文采用布谷鸟算法作为搜索算法。大量实验已经证明布谷鸟算法在解决多目标函数方面的表现要优于其他现有的算法。这是由于布谷鸟算法相较于粒子群算法和遗传算法来说, 需要调整的参数比较少。除了巢的数量 n 以外, 只涉及到一个变量 pa , 而通过证明, 参数 pa 对收敛速度不产生太大影响, 因而不需要针对每个特定的问题都调整参数的值。而且相较于其他的元启发式优化算法, 布谷鸟算法具有较好的通用性。

为了便于问题讨论, 下面首先介绍服务组合的模式和 QoS 计算相关的基本概念。

3.1. 服务组合的模式和 QoS 的计算

简单来说, 服务组合流程可以理解为基于 Web 服务的工作流, 基于 WFMC (Workflow Management Coalition, 工作流管理联盟)定义的 4 种基本模型: 顺序模型、并行模型、选择模型和循环模型。服务组合的 QoS 的计算方法已经有了一些研究[14]-[16], 在研究中可以根据用户的实际需要而选择相应的 QoS 属性, 本文为了能够将问题简单明了的阐述, 只考虑其中 4 种 QoS 参数进行研究, 包括响应时间 T、执行价格 C、服务可用性 A 和服务可靠性 R 进行研究, 实际应用中可根据需要对 QoS 参数的选择进行调整。

服务组合的 QoS 可以根据被选择出来的服务的 QoS 的组合运算来表示。而此时的 QoS 的计算方法就要基于服务组合的模式来决定。顺序模型即为被选择的服务顺序的依次都被执行, 而并行模型为并行的处理被选择的服务, 而选择模型中服务只以一定的概率被选择执行, 循环模型即在服务组合中存在循环, 从而被选择的服务将被循环执行 k 次。

在 4 种基本服务组合模型下, 上述 4 种 QoS 参数的计算规则如下表 1 所示。

3.2. 布谷鸟算法的概述

布谷鸟算法是受到生物行为的启发, 模拟布谷鸟这种生物的孵蛋行为和一些鸟类和果蝇的 Levy 飞行行为而成。一些特定的布谷鸟种群在繁育下一代的过程中, 是将它们的蛋放在其他宿主的鸟巢中, 与此同时它们会将其他鸟类的蛋抛出鸟巢, 从而增加自己的蛋被孵化的概率。当宿主发现自身的巢中的蛋, 并非自己所下的蛋时, 会选择将这个蛋抛出巢或者是直接抛弃这个巢, 重新筑造一个新的巢。

在 Yang 提出的原始布谷鸟算法中有如下三种理想假设:

- 1) 每只布谷鸟一次只产一个蛋, 并将这一个蛋随机的放在一个巢中。
- 2) 宿主巢中的高质量的布谷鸟蛋将被孵化成为下一代布谷鸟。
- 3) 布谷鸟可以选择的宿主鸟巢的数量是一定的, 并且布谷鸟蛋被发现的概率是一个固定值 pa。

根据如上假设, 布谷鸟寻找鸟巢的位置更新公式如下:

$$x_{iy}^{n+1} = x_{iy}^n + \alpha \times Levy(\lambda) \tag{1.1}$$

公式(1.1)中, x_{iy}^n 、 x_{iy}^{n+1} 分别表示第 i 个鸟巢在第 n 代和第 n + 1 代时, 第 j 维的位置。Levy(λ) 为莱维飞行的跳跃路径, 这是一种随机搜索的路径。为了防止有时的跳跃路径过长从而影响搜索的结果, 因而设定了一个路径长短的调节量 α , 对于不同的情况, 可以选择不同的 α 值。

Levy 分布函数的概率密度函数为:

$$Levy \sim u = t^{-\lambda}; 1 < \lambda < 3 \tag{1.2}$$

Table 1. Calculation rules of four kinds of QoS parameters

表 1. 4 种 QoS 参数的计算规则

控制结构	顺序	并行	选择	循环
响应时间 T	$\sum_{i=1}^n T(S_i)$	$\max\{T(S_1), T(S_2), \dots, T(S_n)\}$	$\sum_{i=1}^n P_{oi} T(S_i)$	$K * T(S)$
执行价格 C	$\sum_{i=1}^n C(S_i)$	$\sum_{i=1}^n C(S_i)$	$\sum_{i=1}^n P_{oi} C(S_i)$	$K * C(S)$
服务可用性 A	$\prod_{i=1}^n A(S_i)$	$\prod_{i=1}^n A(S_i)$	$\sum_{i=1}^n P_{oi} A(S_i)$	$A(S)^k$
服务可靠性 R	$\prod_{i=1}^n R(S_i)$	$\prod_{i=1}^n R(S_i)$	$\sum_{i=1}^n P_{oi} R(S_i)$	$R(S)^k$

公式(1.2)是一个带重尾的概率分布, λ 为一个幂次系数, 虽然公式(1.2)能够描述随机游走的过程, 但是很难用编程语言进行数学语言描述, 因而大部分关于布谷鸟的研究都使用 Mantegna 提出的模拟莱维飞行路径公式(1.3)。

模拟 Levy 飞行跳跃路径公式:

$$s = \frac{u}{|v|^{\frac{1}{\beta}}} \quad (1.3)$$

公式(1.3)中参数 β 同公式(1.2)中 λ 的关系为 $\lambda = 1 + \beta$, β 的取值范围为 $0 < \beta < 2$, 在布谷鸟算法中取 β 的值为 1.5。

基于布谷鸟算法的服务组合算法的步骤如下:

Step 1: 用 3.3 节中的建模方法将多目标的服务组合问题转化为单目标问题;

Step 2: 初始化 n 个鸟巢, 即 n 个服务组合方案;

Step 3: 将各个巢中的解解码为服务组合方案中的解, 判断该服务组合是否满足约束条件(公式(2.4)); 若满足, 则计算目标函数的值作为该解的适应度(公式(2.3));

Step 4: 根据位置更新公式(公式(1.1))生成 n 个新的解, 并保留原有的最优解;

Step 5: 根据被发现的概率 pa , 将被发现的解生成新的解;

Step 6: 计算每个解对应的适应度(公式(2.3)); 并找出最小适应度所对应的解。更新最优解集。

Step 7: 判断是否满足终止条件(迭代次数), 不满足, 则返回 Step 4。

3.3. 对问题建模

本节中将基于上节所述 4 种 QoS 参数进行研究, 其中服务响应时间 T 和执行价格 C 作为目标函数的决定因素, 服务可用性 A 和服务可靠性 R 作为约束条件, 服务组合问题就可以形式化描述为如下形式:

$$\text{Min } F(P) = \{[T(P), C(P)]\} \quad (2.1)$$

$$\text{s.t.} \begin{cases} A(P) \geq A_0 \\ R(P) \geq R_0 \end{cases} \quad (2.2)$$

其中, $T(P)$, $C(P)$, $A(P)$, $R(P)$ 为对应的服务组合中的 QoS 计算公式。式(2.1)为目标函数, 即将 $T(P)$ 和 $C(P)$ 同时极小化。

本文以理想点的方法, 将多目标的问题转化为单目标的问题, 取所有服务中服务响应时间 T 和执行价格 C 的最小值作为理想点 T^* 和 C^* 。本文的目标函数就可以表示为如下形式:

$$\text{MinFitness}(P) = \sqrt{[T(P) - T^*]^2 + [C(P) - C^*]^2} \quad (2.3)$$

$$\text{s.t.} \begin{cases} A(P) \geq A_0 \\ R(P) \geq R_0 \end{cases} \quad (2.4)$$

其中公式(2.3)为目标函数, 公式(2.4)为约束条件, 可以根据用户的需要自行设置 A_0 和 C_0 的值。

3.4. 实验与分析

为了验证本文所提出的将布谷鸟算法应用于服务组合问题中的有效性, 即在有效的时间内可以得到令用户满意的 QoS 值的服务组合方案。本文对此方法进行了大量的实验, 实验的环境为 windows7 64 位

操作系统, CORE i3 处理器, 开发环境为 Eclipse 和 Java。

本文的实验数据主要采用知名数据集 QWS, 该数据集是专门用于 Web 服务研究的数据集, 共收集了 5000 个 Web 服务。这些服务是通过 Web 服务爬虫引擎(WSCE)搜集到的, 大部分 Web 服务都是真实的服务。每个 Web 服务都包括 9 个 QoS 属性。本文选择其中的 4 个属性做研究, 分别为服务响应时间 T 、执行价格 C 、服务可用性 A 和服务可靠性 R 。

实验的服务组合中设置了 5 个服务组, 其中每个服务组中包含 50 个服务。将循环的次数分别设置为 500、800、1000, 对得出的适应度值进行比较。由于本算法是基于随机选取的优化算法, 因而在每次运行的结果上都会略有不同, 故在本文中对每种循环次数都进行了 5 次实验。表 2 为在设置了不同循环次数的情况下, 布谷鸟算法在该服务组合问题上得出的适应度值。通过与在服务组合问题中广泛应用的粒子群算法进行对比(表 3), 证明本文提出的将布谷鸟算法应用于服务组合问题的方法的有效性。

从数据可以看出, 随着循环次数的增加, 不论是从几次运算的平均值还是最小值, 应用布谷鸟算法得到的服务组合的适应度都成下降趋势, 而适应度即前文提到的目标函数值, 该值越小即适应度越好。而将应用布谷鸟算法的服务组合方法与应用粒子群算法的服务组合方法相比较可以看出, 在循环次数相同的情况下, 应用布谷鸟算法的服务组合方法的适应度均值和适应度最小值都比应用粒子群算法的服务组合方法好。

图 1 为在不同迭代次数下的 CPU 的运算时间, 单位为秒, 从图中可以看出, 即使在迭代 1000 次的情况下, CPU 的运算时间也不到 3 s, 因此可以说明该算法可以在有效的时间内求出相对满足需求的解。

Table 2. Fitness value of the service composition based on CS algorithm in different cycles
表 2. 基于布谷鸟算法的服务组合在不同的循环次数下适应度值

试验编号 \ 循环次数	500	800	1000
1	213.3017	193.1497	155.6876
2	199.5025	212.3254	229.3321
3	236.4934	191.5401	204.0522
4	206.9085	238.3521	199.6604
5	173.3771	170.5971	197.7614
适应度均值	205.91664	201.19288	197.29874
适应度最小值	173.3771	170.5971	155.6876

Table 3. Fitness value of the service composition based on PSO algorithm in different cycles
表 3. 基于粒子群算法的服务组合在不同循环次数下的适应度值

试验编号 \ 循环次数	500	800	1000
1	313.9575	251.2015	242.1235
2	322.9982	279.2582	221.345
3	248.2536	247.1325	210.765
4	252.1008	301.9651	250.1625
5	374.5171	235.1621	230.762
适应度均值	302.36544	262.94388	231.0316
适应度最小值	248.2536	235.1621	210.765

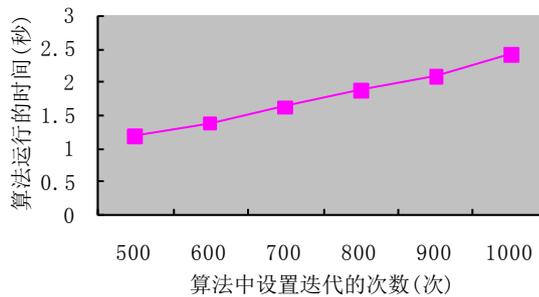


Figure 1. The CPU time of the improved algorithm and original algorithm

图1. 采用改进算法与原始算法的CPU运行时间(单位为秒)

4. 总结

本文首先总结了目前对于 Web 服务组合及 QoS 优化问题的相关研究, 针对不同的抽象服务选择策略进行分类比较。在此基础上, 本文提出了将新的元启发式算法——布谷鸟算法应用到 Web 服务组合问题中, 这在 Web 服务组合问题中是一种新的尝试, 由于布谷鸟算法所涉及到的参数较少, 因而算法容易实现。通过在 QWS 数据集上对本文提出的方法进行实验, 并且与目前在服务组合问题上应用广泛的粒子群算法进行了对比, 得到的适应度结果都能证明, 本文提出的方法要优于粒子群算法, 这也证明了本文提出的方法的有效性。

基金项目

中央高校基本科研业务费专项资金资助(12QNJJ025)(supported by “the Fundamental Research Funds for the Central Universities (12QNJJ025)”)。

参考文献 (References)

- [1] Tao, Q., Chang, H.-Y., Gu, C.-Q. and Yi, Y. (2012) A Novel Prediction Approach for Trustworthy QoS of Web Services. *Expert Systems with Applications*, **39**, 3676-3681.
- [2] 欧伟杰, 曾承, 曾青, 彭智勇, 王珍珍, 刘波, 马景燕 (2013) QoS 感知的高效抽象服务选择. *小型微型计算机系统*, **1**, 1-8.
- [3] 苏森, 李飞, 杨放春 (2008) 分布式环境中服务组合的迭代选择算法. *中国科学(E 辑: 信息科学)*, **10**, 1717-1732.
- [4] 冯建周, 孔令富 (2013) 基于 QoS 的 Web 服务组合中多路径全局优化方法的研究. *小型微型计算机系统*, **7**, 1493-1497.
- [5] 王尚广, 孙其博, 杨放春 (2011) 基于全局 QoS 约束分解的 Web 服务动态选择. *软件学报*, **7**, 1426-1439.
- [6] Mardukhi, F., NematBakhsh, N., Zamanifar, K. and Barati, A. (2013) QoS decomposition for service composition using genetic algorithm. *Applied Soft Computing*, **13**, 3409-3421.
- [7] 陈锦鹏, 万里 (2013) 基于离散粒子群算法全局 QoS 最优的 Web 服务选择. *网络安全技术与应用*, **1**, 50-53.
- [8] 梅俊, 程耕国, 鲍考明 (2012) 基于 QoS 的动态 Web 组合服务选择方法. *工业控制计算机*, **12**, 70-72.
- [9] 康国胜, 刘建勋, 唐明董, 徐宇 (2013) QoS 全局最优动态 Web 服务选择算法. *小型微型计算机系统*, **1**, 73-76.
- [10] 李金忠, 夏洁武, 刘昌鑫, 曾劲涛, 李满华 (2013) 一种新的 QoS 全局最优 Web 服务选择算法. *微电子学与计算机*, **3**, 97-101.
- [11] Liu, M., Wang, M.R., Shen, W.M., Luo, N. and Yan, J.W. (2012) A quality of service (QoS)-aware execution plan selection approach for a service composition process. *Future Generation Computer Systems*, **28**, 1080-1089.
- [12] 夏亚梅, 程渤, 陈俊亮, 孟祥武, 刘栋 (2012) 基于改进蚁群算法的服务组合优化. *计算机学报*, **2**, 270-281.
- [13] Yang, X.S. and Deb, S. (2009) Cuckoo search via Levy flights. *Proceedings of World Congress on Nature & Biologically Inspired Computing*, Coimbatore, 9-11 December 2009, 210-214.

- [14] Zeng, L.Z., Benatallah, B., Dumas, M., et al. (2003) Quality driven Web Service composition. *Proceedings of the International World Wide Web Conference*, Budapest, 20-24 May 2003, 411-421.
- [15] Alrifai, M. and Risse, T. (2009) Combining global optimization with local selection for efficient QoS aware service composition. ACM, New York, 881-890.
- [16] Dong, Y.Y., Ni, H., Deng, H.J., et al. (2011) Service selection strategy offering global optimal quality of service. *Journal of Chinese Computer System*, **32**, 455-459.