

一种基于 QoS 剪枝的 Top-k 自动服务组合方法

杨汝涛^{1,2}, 张绍谦^{1,2}, 窦万春^{1,2}

(1. 南京大学计算机软件新技术国家重点实验室, 江苏南京 210093;

2. 南京大学计算机科学与技术系, 江苏南京 210093)

摘 要: 在大规模的 Web 服务中自动搜索满足功能性需求并且 QoS 前 k 优的服务组合是一个具有挑战性的问题. 为解决此问题, 本文方法首先对服务组合进行建模, 然后通过前向服务过滤算法缩小求解空间, 最后利用贪婪算法进行 QoS 剪枝以加快 Top-k 服务组合的后向搜索. 本文给出了该方法的系统结构设计, 实验结果验证了此方法的可行性和高效性.

关键词: Web 服务; 服务组合; 服务质量 (QoS); 本体语义; 剪枝

中图分类号: TP393 **文献标识码:** A **文章编号:** 0372-2112 (2012) 07-1489-03

电子学报 URL: <http://www.ejournal.org.cn> **DOI:** 10.3969/j.issn.0372-2112.2012.07.033

A QoS Pruning-Based Top-k Automatic Service Composition Method

YANG Ru-tao^{1,2}, ZHANG Shao-qian^{1,2}, DOU Wan-chun^{1,2}

(1. State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, Jiangsu 210093, China;

2. Department of Computer Science and Technology, Nanjing University, Nanjing, Jiangsu 210093, China)

Abstract: Generally, in the situation of vast amount of Web services, it is often a challenge to automatically searching for Top-k QoS-optimal service composition plans. To address the challenge, service composition modeling is proposed firstly. Then, a forward service filtering algorithm is employed for reducing solution spaces. Lastly, a greedy-based pruning algorithm is designed for backward searching for Top-k QoS-optimal solutions efficiently. In addition, a platform is designed for implementing the proposed method, and its experiment results validate the effectiveness and efficiency of the method.

Key words: web service; service composition; quality of service (QoS); ontological semantics; pruning

1 引言

在面向服务的体系结构(SOA)中, Web 服务是通过 URI 来标识的支持网络交互的软件应用, 其接口和绑定可以被定义、描述和发现^[1]. 面向服务的计算本质上是以协同为中心, 因此服务组合是 Web 服务应用中最为关键的技术之一. 当前, 对服务组合的研究大概可归于两类: 一类是工作流组合, 首先将服务需求规划出虚拟的任务流程, 然后为每个子任务选择服务完成实际的服务组合^[2,3]; 另一类是自动服务组合, 通过语义匹配服务之间的接口, 智能规划组合方案^[4,5]. 本文将重点关注本体语义驱动的自动服务组合问题.

服务质量(QoS)最早由 Menasce^[6]引入 Web 服务中, 用于描述其提供承诺的服务功能时体现出来的非功能属性, 如响应时间、吞吐量、安全性等等. 本文的 QoS 将

以响应时间为例, 择优选择 Top-k 服务组合, 其他 QoS 属性以及多 QoS 聚合同样适用于此方法. 在工作流服务组合研究中, Zeng 等^[2]给出了 Web 服务组合的 QoS 模型, 并在此模型基础上提出了全局最优服务选择方法. Alrifai 等^[3]提出将全局约束分解为局部约束, 然后利用局部服务选择来获取近似最优服务组合的方法. 在工作流组合中, 任务的调用逻辑先于服务选择, 而在具体的组合服务执行过程中, 语义匹配是一大问题. 在自动服务组合方面, 语义 Web 成为研究热点, W3C 制定了网络本体语言(OWL), 斯坦福等大学提出了 OWL-S. Gao 等^[7]基于参数匹配采用动态规划的算法求解服务组合中最优服务选择问题, 首次考虑了接口约束问题, 但仅限于两个 Web 服务之间的接口. 本文在服务组合参数匹配中, 考虑了一对多, 多对一以及多对多的服务依赖关系. 大规模服务数量和本体语义数量将考验自动服务组合

的搜索效率.Jiang 等^[5]在 WSC'09^[4]上设计出 QoS 最优的自动服务组合方法.不同于上方法,本文的基于 QoS 剪枝的自动服务组合方法将求出服务质量前 k 优的服务组合.

2 服务组合建模

CWSC2011^[8]根据当前的 Web 服务发展状况,设定了一个实验可行的服务组合应用环境.本章在其驱动下,对服务组合进行建模.

2.1 原子服务建模

定义 1 原子服务 $S = \{Name, Input, Output, QoS\}$.

其中 Name 和 QoS 分别为服务的名称和服务质量; Input 和 Output 分别为服务的输入和输出参数集,每个参数为本体语义中的一个实例(Thing).

2.2 语义建模

本体语义描述 OWL 文件包括两类实体:概念(Class)和实例(Thing);以及两类逻辑关系:subClassOf 和 type 分别描述概念 A 是概念 B 的子概念,实例 T 的类型是概念 C.

定义 2 概念 $Class = \{Name, superClass\}$,其中 Class 是 superClass 的子概念;实例 $Thing = \{Name, Class\}$.其中子概念关系具有自反性和传递性.

定义 3 参数 p_1 匹配参数 p_2 (记做 $p_1 \rightarrow p_2$),当且仅当 p_1 的类型是 p_2 的类型的子概念.

定义 4 参数集合 P 匹配服务 S (记做 $P \rightarrow S$),当且仅当 $\forall p_i \in S.Input, \exists p_j \in P, p_j \rightarrow p_i$.

根据定义 4,给定可用参数集合 $P, P \rightarrow S$ 则服务 S 可被调用;而可用参数集合可以由一个服务提供,也可以由多个服务提供.因此服务依赖存在一对多,多对一和多对多的关系.

2.3 组合服务建模

定义 5 给定服务请求(输入 IR、输出 OR),组合服务是一组可串行化调用的一组原子服务集 $(w_1, w_2, \dots, w_n), n \geq 1$,使得:

$$w_i.Input \subseteq \bigcup_{0 < j < i} w_j.Output \cup IR; i = 1, \dots, n$$

$$OR \subseteq \bigcup_{0 < i \leq n} w_i.Output$$

本文设计的服务组合系统基于定义 5,为用户的服务请求寻找 QoS^[2]前 k 优的组合服务方案返回给用户.

3 自动服务组合系统的总体结构设计

图 1 是整个服务组合系统的结构图,其中实线箭头代表了操作流,虚线箭头代表了数据流.系统包括初始化模块和服务组合模块.初始化模块负责服务和概念数据的文件解析和预处理,为服务组合模块提供数据

基础.服务组合模块则实现了基于 QoS 剪枝的 Top-k 自动服务组合搜索.

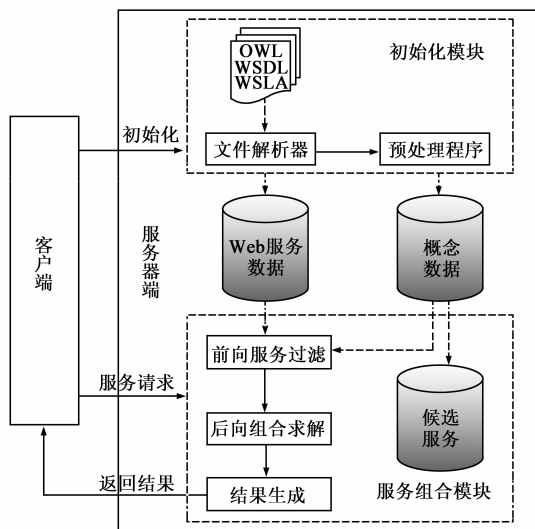


图1 服务组合系统结构图

4 基于 QoS 剪枝的 Top-k 自动服务组合

4.1 预处理

为提高实时服务组合搜索的效率,预处理在以下三个方面进行.

- (1)根据定义 2 把服务的参数从实例映射到概念.
- (2)根据服务表建立反向参数表.
- (3)挖掘服务依赖中前驱服务集 PRS 并保证查全率.

4.2 前向服务过滤

前向服务过滤完成两项任务.一是求取候选服务集,缩小服务组合求解空间,二是为保证后向组合搜索贪婪算法的正确性,求得每个服务从输入出发最小可达响应时间 frontQoS.

4.3 后向组合求解算法

该算法基于组合服务建模进行设计,详见算法 1.服务请求驱动的后向搜索保证了服务无冗余性.预处理中前驱服务的求取保证了组合搜索的查全率.基于 QoS 的剪枝利用了贪心思想,而前向服务过滤中 frontQoS 的求取保证了 Top-k 的正确性.剪枝的依据是当前未完成组合方案继续生成下去的最优 solutionQoS 可通过下述方法求出:

(1)对每个未满足服务 S_m 求从 END 出发的最长响应时间 endQoS.

(2) $\forall S_m \in RS, \forall S_n \in OS, S_m \in S_n.PRS$

$$solutionQoS = \max \{ S_m.frontQoS + S_n.endQoS \}.$$

其中 OS 为未满足的服务集,RS 为最小前驱服务集.

算法 1 后向 Top-k 组合求解算法

Input: service request R, Class, Services, FFS, CS, SQ, topk

Output: ordered Top-k service composition Comp

1: PriorityQueue < Solution > Comp, UnComp

2: initialize(unComp, OR)

3: while Comp.size < topk

4: undoneSolution ← UnComp.remove()

5: if undoneSolution.isBingo(IR) //定义 5

6: undoneSolution.produceDAG()

7: Comp.add(undoneSolution)

8: else

9: undoneSolution.backward(
 topk - Comp.size, maxQos)

10: update UnComp

11: update maxQos be the k-th Qos of Solutions

12: end if

13: end while

5 实验评估

本方法在一台配置 2.80GHz 双核 CPU、2G 内存、JDK1.6 和 Windows XP 的单机上进行实验评估.实验采用的数据集为 CWSC2011 提供的 11 组测试数据集,可通过链接^[8]下载.实验结果经 CWSC2011 提供的 ResultChecker 程序^[8]验证了 Qos Top-k 组合结果的正确性.表 1 将从时间效率方面对本文方法进行评估(实验中 Top-k 取 10).

表 1 服务组合方法的时间性能

概念_服务数	初始化预处理时间(ms)	前向服务过滤时间(ms)	服务组合响应时间(ms)
4000_4000	305	18	250
6000_4000	210	153	1703
8000_4000	164	125	297
9000_6000	389	725	6969
10000_10000	449	297	1015
12000_6000	342	79	266
12000_8000	411	82	281
15000_10000	413	94	156
16000_8000	527	250	7360
18000_12000	541	110	406
24000_12000	605	110	468

表 1 从总体看初始化时间随着概念数和服务数的增加而上升.实验结果证明了过滤算法的高效性.总体服务组合的响应时间大多在 1s 以内,数据分析得出服务组合的响应时间跟组合方案中的可用服务、参数个数,以及组合的复杂程度有关.

6 总结

本文提出了一种基于 QoS 剪枝 Top-k 的自动服务组合方法,其实现系统能够根据用户提供的服务请求,自动在大量服务集中搜索满足需求的服务组合.该方

法采用本体描述来解决服务接口参数间的自动匹配问题,并利用 QoS 剪枝加快前 k 优服务组合的搜索.实验验证了方法的正确性和高效性.

参考文献

[1] G Alonso, F Casati, et al. Web Services: Concepts, Architectures, and Applications [M]. New York: Springer, 2004.

[2] L Zeng, B Benatallah, et al. QoS-aware middleware for web services composition [J]. IEEE Tran Software Engineering, 2004, 30(5): 311 – 327.

[3] M Alrifai, T Risse. Combining global optimization with local selection for efficient QoS-aware service composition [A]. Proc WWW’09 [C]. Madrid: ACM Press, 2009. 881 – 890.

[4] Web Service Challenge 2009 [OL]. <http://ws-challenge.georgetown.edu/wsc09/>.

[5] Wei Jiang, Charles Zhang, et al. QSynth: A tool for QoS-aware automatic service composition [A]. Proc ICWS’10 [C]. Florida: ACM and IEEE Computer, 2010. 42 – 49.

[6] D Menasce. QoS issues in web services [J]. IEEE Internet Compute, 2002, 6(6): 72 – 75.

[7] Y Gao, B Zhang, et al. Optimal selection of web services for composition based on interface-matching and weighted multi-stage graph [A]. Proc PDCAT’05 [C]. Dalian: IEEE, 2005. 336 – 370.

[8] China Web Service Cup (CWSC2011) [OL]. <http://debs.ict.ac.cn/cwsc2011>.

作者简介



杨汝涛 男, 1987 年出生于山东阳谷. 南京大学硕士研究生, 研究方向为服务计算.
E-mail: yrutao@gmail.com



窦万春 男, 1971 年生于江苏徐州. 教授、博士生导师、中国计算机学会“协同计算”专家委员会委员, 中国电子学会“云计算”专家委员会委员. 主要研究方向为服务计算、云计算、协同计算、工作流技术等.
E-mail: douwc@nju.edu.cn