基于扩展图规划的 Top-K 服务组合方法研究

徐 猛,崔立真,李庆忠
(山东大学计算机科学与技术学院,山东济南 250101)

摘 要: 自动服务组合是目前云计算中的关键技术与研究热点.为大规模用户提供多个满足个性化需求的组合服务是当前云环境下自动服务组合中急需解决的问题.提出了基于扩展图规划的 Top-K 服务组合方法,借助服务索引和增加图规划中的辅助节点,使得经过一次规划搜索即可找到 Top-K 个满足用户 QoS 要求的组合服务.实验表明,该方法能够有效提高服务组合的效率,并保证服务组合结果的正确性,更加适用于云计算环境下海量网络服务及大规模用户个性化需求的自动服务组合问题.

关键词: 服务组合; Top-K; 扩展图规划; 辅助节点
中图分类号: TP393 **文献标识码:** A **文章编号:** 0372-2112 (2012) 07-1404-06
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2012.07.019

An Extended Graph-Planning Based Top-K Service Composition Method

XU Meng, CUI Li-zhen, LI Qing-zhong
(*School of Computer Science and Technology, Shandong University, Jinan, Shandong* 250101, *China*)

Abstract: ASC (Automatic Service Composition) is the key issue in cloud computing. It is an urgent problem of ASC in cloud computing to provide multiple composition service which can satisfy the personalized requirements for large-scale users. An extended graph-planning based Top-K service composition method is proposed in this paper. By using service indexes and the auxiliary nodes in extended planning graph, Top-K composition service can be found through one search. The approach can improve the efficiency of service composition and ensure the correctness of the result. It applies to the ASC problem in cloud which has a large number of services and the users' requirements are personalized.

Key words: service composition; Top-K; extended graph-planning; auxiliary node

1 引言

云计算是近几年兴起的一种新的计算模式.云计算以互联网为基础,对外以服务的方式提供计算能力,具有动态性、可伸缩性等特点.云计算中用户的个性化需求更加明显,主要表现在功能需求难以枚举,功能需求的抽象程度、描述都有明显不同.如何协同云中可用的 Web 服务,进行自动服务组合,构建满足用户不断涌现的大规模个性化需求的组合服务,是云计算中的关键技术与研究热点,也是云计算当前面临的一大挑战.

当前,为网络用户提供 Top-K 个候选组合服务的需求十分迫切.一方面,云计算中海量 Web 服务之间蕴含着丰富的关联关系,如何建立海量 Web 服务的群体涌现行为与个性化用户需求之间的关联,主动为用户提供组合服务的多个推荐,需要能够给出一组满足用户需求的候选组合服务供用户选择.另一方面,由于服务的高度动态性及自治性,其状态随时可能发生变化,而有效性和可靠性又是云服务提供的关键特性,同时也需要能够找到多个满足用户需求的组合服务候选,在组合服务执行失效的情况下能够及时进行替换.

然而,目前的自动 Web 服务组合方法大都返回一个满足用户需求的组合服务[1~4].由于 Web 服务的数量巨大,如何能够在服务组合过程中减小搜索空间,提高服务组合效率,快速的找到符合用户需求的 Top-K 个组合服务成为自动服务组合的一大挑战.基于上述原因,本文提出一种基于扩展图规划的 Top-K 服务组合方法.该方法能够适用于具有大规模候选服务的服务组合问题,并且能够通过一次搜索即可找到其中 Top-K 个组合服务,提高了服务组合的效率.同时,基于该方法,我们设计开发了一个 Web 服务组合系统.

收稿日期:2010-12-15; 修回日期:2011-07-10
基金项目:国家自然科学基金(No.61003253);国家科技支撑计划(No.2009BAH44B02);山东省自然科学基金(No.ZR2010FQ010,No.ZR2010FM031);山东省科技发展计划(No.2009GG10001002);高等学校博士学科点专项科研基金(No.200804221031);山东大学自主创新基金(No.2009TS030)

2　问题描述

服务组合就是将当前网络上可用的 Web 服务,组合成为更大粒度的新的 Web 服务,从而满足用户的需求.为更好的描述该问题,下面给出一些基本定义.

定义 1　候选服务集 S　已注册到系统中,系统在服务组合阶段可以使用的服务称为候选服务,表示为 s_i.所有候选服务构成的集合称为候选服务集,表示为 $S=\{s_i \mid s_i$ 已注册在系统中$\}$.S 中的候选服务 s_i 可表示为三元组 $s_i=<p(s_i)_{in}, p(s_i)_{out}, QoS>$,其中 $p(s_i)_{in}$为 s_i 的输入参数,$p(s_i)_{out}$为 s_i 的输出参数,QoS 为 s_i 的 QoS 属性.

定义 2　组合请求 R　用户向系统发起的一次服务组合请求称为组合请求,表示为 R.$R=\{P_{in}, P_{out}\}$.其中 P_{in}为输入参数集合,P_{out}为输出参数集合.

定义 3　语义树 T　候选服务集 S 使用的参数之间的语义关系称为语义树,表示为 T.语义树 T 描述了概念之间的父子关系.因此,在考察一个服务是否满足要求时,不仅要考虑其输入、输出参数,同时要考虑概念之间的语义关系.由此,可将表示服务的三元组扩展为 $s_i=<p(s_i)_{in}', p(s_i)_{out}', QoS>$,其中 $p(s_i)_{in}'=p(s_i)_{in}\cup p(s_i)_{in}$中元素的子类,$p(s_i)_{out}'=p(s_i)_{out}\cup p(s_i)_{out}$中元素的父类.

定义 4　可行解(Solution)　对于组合请求 R,及组合服务 s,若组合服务 s 对于组合请求 R 的输入和输出参数都满足,即 $p(s)_{in}'\subseteq P_{in}$并且 $P_{out}\subseteq p(s)_{out}'$,则组合服务 s 能够满足组合请求,称组合服务 s 为组合请求 R 的一个可行解.对于可行解的定义需要注意,组合服务 s 可以是多个服务组合而成,也可以是单一的一个候选服务.

服务组合问题可以表示为给定三元组 < 候选服务集 S,语义树 T,组合请求 R >,求组合服务 CS(Composition Service),使得 CS 为对于 R 的可行解.服务组合系统的输入包括 4 个文件:(1)Web 服务描述文件(WSDL).该文件用于描述给定的服务集,给出了每一个服务的输入、输出参数.服务组合系统基于服务集进行服务组合.(2)Web 服务水平描述文件(WSLA).该文件用于描述服务集中各个服务的 QoS 属性,本文以响应时间为 QoS.(3)概念语义关系描述文件(OWL).该文件用于描述服务集中使用到的概念之间的语义关系.概念数量从几千到几万不等.(4)服务组合请求文件(Challenge).该文件使用 WSDL 格式,其中包含了要求的输入参数及输出参数.文件(1)、(2)描述了候选服务集 S,文件(3)描述了语义树 T.文件(4)描述了组合请求 R.

3　基于扩展图规划的 Top-K 服务组合系统

3.1　系统框架

服务组合系统的整体框架如图 1 所示,整个系统封主要有初始化(initialize)、服务组合(startQuery)和停止组合(stopComposition)三个操作.初始化(initialize)操作完成系统的初始化,服务组合(startQuery)操作向服务组合系统发起一次组合请求,停止组合(stopComposition)操作停止服务组合系统的运行.系统由解析器(Parser)、服务组合器(Composer)、结果生成模块(Solutions)、服务库(Services Repository)、语义树(Concept Forest)及索引(Indexes)组成.解析器主要负责解析 WSDL、WSLA、OWL 及 Challenge 等 4 个输入文件;服务组合器主要负责根据服务库、语义树及用户需求进行服务组合,生成扩展规划图;结果生成模块根据服务组合器生成的扩展规划图对可行解进行排序,生成最终的 Top-K 个可行解;服务库和语义树根据 WSDL、WSLA 及 OWL 文件建立,全部放在内存中;索引是为了提高对服务库及语义树的查询效率而建立的,具体结构在 3.2 节中介绍.

系统运行分为 2 个阶段,即初始化阶段和服务组合阶段.系统初始化阶段,Client 调用服务组合系统的初始化(initialize)操作,将 WSDL、WSLA、OWL 文件地址传送给服务组合系统(①);解析器读取并解析文件,建立服务库、语义树及其索引(②),系统初始化完毕.

系统初始化完成以后,可以接受 Client 的服务组合请求,Client 调用服务组合系统的服务组合(startQuery)操作,将服务组合请求文件传送给服务组合系统(③);服务组合器将使用解析器解析文件(④),根据查询请求及服务库、语义树和索引,建立扩展规划图(⑤);结果生成模块根据扩展规划图对组合结果进行筛选、排序,将符合请求的组合结果通过调用 Client 的 callback 接口,将组合结果返回给 Client(⑥).

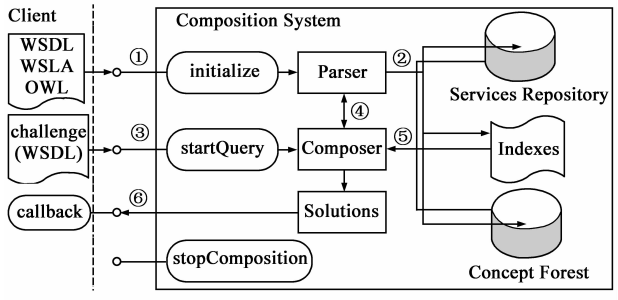

图1　服务组合系统架构

3.2　索引

由于系统需要处理的服务数及概念数从几千到几万不等,为了能够快速高效的找到符合需求的组合服务,系统需要建立索引,来加快对服务及概念的查找速度,提高整体效率.

3.2.1 概念索引

概念索引根据 OWL 文件中描述的语义关系建立各概念间的父子关系.其形式化描述如下:

Concept_Index::=List of Inst_Node;
Inst_Node::=<Inst_ID,Class_Node>;
Inst_ID::=the hash value of instance;
Class_Node::=<Class_Name,Super_Class_Pointer>;
Super_Class_Pointer::=the pointer to the super class of the class;

概念索引的结构如图 2 所示.概念索引基于哈希表实现.其中 Insti 表示 OWL 文件中的实例,Coni 表示 OWL 文件中的类.每一实例指向其所实现的类.各类又分别指向其直接父类.由此,根据 Instance 查找其对应的类时,所需时间复杂度为 O(1).根据类查找其直接父类的时间复杂度也为 O(1).

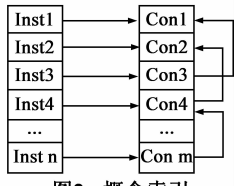

图2 概念索引

3.2.2 服务索引

服务索引根据 WSDL 文件中的服务描述建立服务和其输入输出参数之间的关系,以便于快速检索服务.服务索引的形式化描述如下:

Service_Index::=<List of Part_Node,List of Service_Node>;
Part_Node::=<Part_ID,Service_List_In,Service_List_Out>;
Service_List_In::=List of S_Node;
Service_List_Out::=List of S_Node;
S_Node::=<Service_Name,S_pointer>;
S_Pointer::=the pointer to the S_Node;
Service_Node::=<Service_ID,Part_List_In,Part_List_Out>;
Part_List_In::=List of input partname;
Part_List_Out::=List of output partname;

服务索引的结构如图 3 所示.服务索引基于哈希表实现,其中 pi 表示服务的 partname,指向以 pi 为第一个输入和输出参数的服务链表,其中存储了服务名.si 表示服务,指向 si 的完整的输入和输出参数列表.

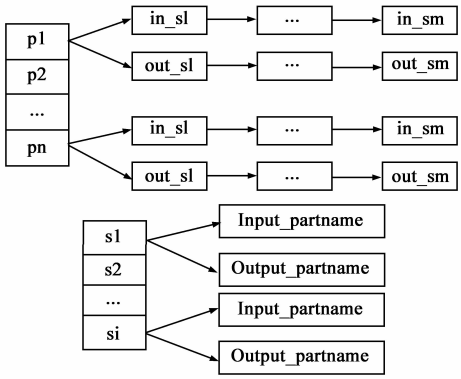

图3 服务索引

3.3 基于扩展图规划的 Top-K 服务组合算法

本节详细介绍基于扩展图规划的 Top-K 服务组合算法.为了更清晰的介绍该算法,首先给出算法中使用的相关定义.

定义 5 可用参数列表 可用参数列表 = $R.P_{in}$ ∪ $R.P_{in}$的所有祖先∪规划图中已有服务的输出参数∪规划图中已有服务的输出参数的所有祖先.可用参数列表包含了所有的可用参数及其祖先.

定义 6 可用服务 若服务的所有输入参数均在可用参数列表中,则称这一服务为可用服务.

定义 7 图规划算法 图规划算法将规划问题建模为一个五元组 $<P,P_0,C,A,T>$,其中 P 表示所有的状态空间,P_0 表示初始状态集,C 表示目标状态集,A 表示可操作的动作集,$T=P*A*P$ 定义了每个动作的前置条件 pre[a]和效果 sub[b],a、b∈A,并利用规划图[5]进行求解.

图规划算法求解服务组合问题就是要找到一个动作序列(组合服务),使得该问题可以从初始状态(输入参数集合 $R.P_{in}$)应用这个动作序列到达目标状态(输出参数集合 $R.P_{out}$).使用图规划算法求解服务组合问题时,首先根据输入参数集合 $R.P_{in}$找到所有可用服务并加入到规划图中,然后将新添加服务的输出参数及其祖先作为可用参数,进行新一轮可用服务的查找,直到可用参数满足输出参数集合 $R.P_{out}$的要求,或者无法添加新的服务为止,至此所求得的图称为规划图.通过反向搜索规划图,最终得到服务组合问题的可行解.

传统的图规划方法在进行服务组合时,在规划图建立完成之后,需要进行反向搜索,确定最终的组合服务.这一方法在求解单一组合服务时效率较高,但若需求解 Top-K 个组合服务,由于每一个组合服务都需要在最后进行反向搜索,效率大大降低.因此,本文提出扩展的图规划算法,在图规划方法的基础上,加入辅助节点,在搜索结束时直接得到所有可行解及其 QoS,无需在最终图上进行反向搜索.

定义 8 组合服务的 DAG 图 组合服务的 DAG 图为二元组 $<V,E>$,其中 $V=\{s|s\in$组合服务$\}$,$E=\{<s_i,s_j>|s_i$的一个或多个输出参数为 s_j 一个或多个的输入参数$\}$.

定义 9 组合服务的 QoS 根据不同 QoS 属性的计算方法及组合服务的 DAG 结构,计算得出的整个组合服务的 QoS 值称为组合服务的 QoS.本文仅考虑执行时间这一 QoS 属性,因此,组合服务的 QoS 即为组合服务中以执行时间为权重的最长路径.

定义 10 辅助节点 辅助节点的结构为 <QoS,AvailablePartName,DAG,Solution>,其中 QoS 表示组合服务执行到此节点时的 QoS 指标,AvailablePartName 表示组合服务执行到此节点时的可用参数列表,DAG 存储

到此节点时组合服务的 DAG 图,Solution 表示是否为可行解.

定义 11　扩展规划图　将规划图的命题层删除,并在每一个动作节点之后加入一个辅助节点所形成的图称为扩展规划图.

基于扩展图规划的 Top-K 服务组合算法的核心思想是通过利用扩展规划图,在搜索形成整个相关 Web 服务拓扑图的过程中,同时计算整个组合服务的 QoS 及 DAG 图,在搜索完成后,得出所有可行解及其 QoS,然后通过对 QoS 进行排序,直接得到 Top-K 个组合服务,避免进行二次搜索,从而节省执行时间.

基于扩展图规划的 Top-K 服务组合的具体算法如算法 1 所示.其中 1-3 行为初始化,对最初辅助节点、全局可用参数列表及搜索结束标志进行初始化;6-8 行判断当前是否找到了可行解,若找到可行解则将辅助节点进行合并,并标记为可行解;10-14 行判断是否有可用服务,若找到可用服务,则将可用服务加入到规划图中,并添加辅助节点,将新的可用参数加入到下一轮查找的可用参数列表中;15 行将当前可用参数列表赋值为下一轮的可用参数列表.

图 4 给出了一个简单的实例.其中圆形表示 Web 服务,其中依次表示 Web 服务名称及其响应时间,矩形表示辅助节点.在开始和最后建立两个虚拟 Web 服务:Start 和 End. Start 的输入、输出均为组合请求的输入,End 的输出为组合请求的输出.组合请求的输入为 A、B、C,输出为 G、I. Start 为虚拟 Web 服务,因此其辅助节点的 QoS 指标(本文 QoS 指标仅考虑响应时间)为 0,可用参数列表为 A、B、C,由于不满足输出请求,因此不是可行解.根据当前可用参数列表,可以满足 WS1 及 WS2 的输入,因此将 WS1 及 WS2 加入扩展规划图,并分别计算它们的辅助节点,结果分别为 < 5, A、B、C、D, DAG, FALSE > 和 < 7, A、B、C、E、F, DAG, FALSE > .在调用 WS3 和 WS4 时,需要合并 WS1 和 WS2 的输出,因此,在计算 WS3 和 WS4 的辅助节点时,将 WS1 和 WS2 的辅助节点合并,并分别加入 WS3 和 WS4,得到结果分别为 < 18, A、B、C、D、E、F、G, DAG, FALSE > 和 < 15, A、B、C、D、E、F、H、I, DAG, FALSE > .由于两个辅助节点的合并包括了请求输出的 G、I,因此,将两个辅助节点合并,得到一个可行解.

算法 1　基于扩展图规划的 Top-K 服务组合算法

输入:组合请求 R,候选服务集 S,语义树 T
输出:可行解队列 Q
1. 初始化最初辅助节点 tn1, tn1. QoS = 0, tn1. AvailablePartName = $R.P_{in}$, tn1. Solution = false tn1. DAG = Φ,
2. 初始化全局可用参数列表, AvailablePartName = $R.P_{in} \cup R.P_{in}$ 的所有父类
3. 初始化搜索结束标志, end_find = false
4. While(not end_find)
5. end_find = true
6. While($R.P_{out} \subseteq$ AvailablePartName)
7. 将可满足 $R.P_{out}$的辅助节点合并为一个辅助节点,标记为可行解,并将其加入可行解队列 Q
8. End while
9. While($\exists s_i \in S$ $P(s_i)in \subseteq$ AvailablePartName)
10. end_find = false
11. 将 s_i 加入到服务组合图中
12. 加入辅助节点 tni,计算辅助节点的 tni. QoS、tni. AvailablePartName 及 tni. DAG
13. AvailablePartName_next = AvailablePartName_next $\cup$ $P(s_i)_{out}$及其所有父类
14. End while
15. AvailablePartName = AvailablePartName_next
16. End while
17. Return Q

4　实验及结果分析

为了验证服务组合系统的效率及可伸缩性,我们使用 5 组不同的候选服务测试数据集进行了实验.实验环境为 Intel® Core™ 2 Duo 2.33GHz, RAM1GB, 硬盘 320GB.测试结果如表 1 所示.由表 1 可以看出,对于前 4 个测试集,服务组合系统的响应时间都在 1s 左右,并且随着服务数的增加,服务组合系统的响应时间没有明显增加,概念数与服务数之比也没有影响服务组合系统的响应时间.对于测试集 5,服务组合系统的响应时间快速增加到 18s 以上.通过大量实验,我们得出如下结论,服务组合的时间不仅与服务数或概念数相关,还与候选服务集中服务之间的关系相关.若将所有候选服务作为顶点,服务之间的关系作为边构成一个图,那么这个图的边的密集程度与服务组合的时间具有极强的正相关性.测试集 5

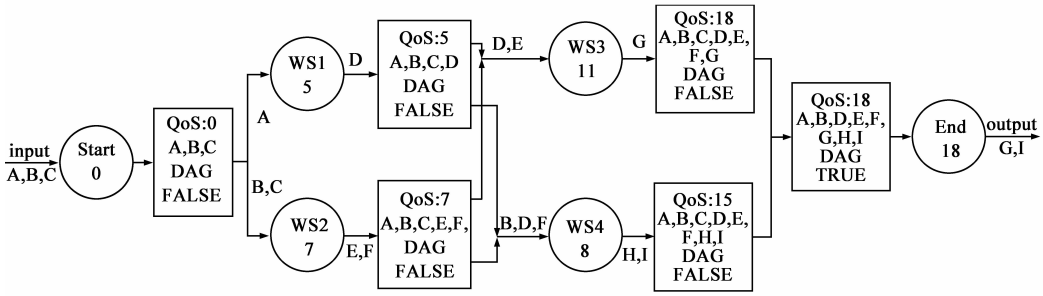

图4　基于扩展规划图的服务组合实例

就是一个例子,由于测试集5中服务之间的关系非常复杂,因此导致服务组合时间的显著增加.对于5个测试集的组合结果,我们使用了首届全国WEB服务竞赛组委会提供的ResultChecker程序进行验证所返回的组合结果是否有效和无冗余.有效是指服务组合结果中每个服务都可以被触发,且该组合结果使查询请求满足;该服务组合不存在输入输出边缺乏其一的参数;该服务组合不存在一条边上的前后两个服务结点无参数匹配的情况.无冗余是指服务组合结果中不存在这样的服务:移除它之后,服务组合结果仍然使查询请求满足.由表1可见,组合结果的正确率全部为100%,显示出服务组合系统有着良好的健壮性和可伸缩性.

表1 测试集及组合时间

测试集	服务数	概念数	时间(ms)	正确率
测试集1	8000	4000	312	100%
测试集2	10000	6000	500	100%
测试集3	4000	40000	1109	100%
测试集4	15000	100000	828	100%
测试集5	40000	20000	18625	100%

5 相关工作

目前在自动服务组合方面已有很多成果.文献[6]提出了一个服务组合的半自动化方法.其主要思想是构造一个服务依赖图(SDG),运用自底而上的REV * 搜索算法找到一个子图做为解决方案.文献[7]提出了一种使用倒表做为服务索引的快速服务组合模型,并且指出了如何处理数据之间的语义关系.文献[8]提出了用于优化多QoS的全局规划服务组合方法,通过线性规划方法,该方法可以处理多条执行路径.这一方法主要针对运行时的动态服务选择.文献[9]把自动服务组合看作经典的人工智能规划问题,然而该方法只考虑了单调增加的QoS指标.此外,在文献[10]中,作者将服务组合问题建模为一个多维度、多选择的0-1背包问题(MMKP)问题或多重约束的最优路径(MCOP)的问题,提出了高效的启发式服务组合算法.文献[1]提出了一种高效的剪枝算法,用于进行QoS感知的服务组合.文献[11]提出了一种基于动态描述逻辑的Web服务自动组合框架.在该框架中,Web服务自动组合被划分为逻辑层和实现层两部分,于是服务的自动组合问题在逻辑上归结为一个动作规划问题,在实现上归结为一个根据动作选择具体服务的服务选择问题.文献[12]从软件体系结构角度,基于软件体系结构描述语言XYZ/ADL和精化检验/模型检测方法,提出了一种Web服务组合的描述与验证方法.文献[13]提出了一种改进的离散粒子群算法,该算法能够在降低服务选择时间的同时,提高服务选择的质量.

然而,上述方法都无法快速有效的找到服务组合问题中的Top-K个可行解.因此,本文基于文献[14]提出了基于扩展图规划的Top-K服务组合算法,能够在大规模候选服务集上快速得到满足用户需求的Top-K个可行解.

6 总结

为了能够在大规模候选服务集中快速找到符合用户需求的Top-K个组合服务,本文提出了基于扩展图规划的Top-K服务组合方法,该方法通过在图规划中加入辅助节点,能够在服务组合过程中减小搜索空间,避免多次反向搜索,提高服务组合效率.通过实验,证明本文提出的基于扩展图规划的Top-K服务组合方法具有较好的性能,且可伸缩性较强,能够适用于具有大规模候选服务集的服务组合需求.

未来计划在以下方面进行进一步研究:首先,计划对本文使用的索引结构进行优化,以提高求解效率.另外,研究如何能够使该方法求解满足多QoS需求的服务组合问题.

参考文献

[1] Zhenqiu Huang, Wei Jiang, Songlin Hu, Zhiyong Liu. Effective pruning algorithm for qoS-aware service composition[A]. Proceedings of the 2009 IEEE Conference on Commerce and Enterprise Computing[C]. Washington DC, USA: IEEE Computer Society, 2009. 519 - 522.

[2] Yixin Yan, Bin Xu, Zhifeng Gu, Sen Luo. A qoS-driven approach for semantic service composition[A]. Proceedings of the 2009 IEEE Conference on Commerce and Enterprise Computing[C]. Washington DC, USA: IEEE Computer Society, 2009. 523 - 526.

[3] 李鑫,程渤,杨国纬,刘启和.一种基于事件的Web服务组合方法[J].软件学报,2009,20(12):3101-3116.
Li Xin, Cheng Bo, Yang Guowei, Liu Qihe. Method of web services composition based on events[J]. Journal of Software, 2009, 20(12): 3101 - 3116. (in Chinese)

[4] 范小芹,蒋昌俊,王俊丽,庞善臣.随机QoS感知的可靠Web服务组合[J].软件学报,2009,20(3):546-556.
Fan Xiaoqin, Jiang Changjun, Wang Junli, Pang Shanchen. Random-QoS-aware reliable web service composition[J]. Journal of Software, 2009, 20(3): 546 - 556. (in Chinese)

[5] Blum A, Furst M. Fast planning through planning graph analysis [J]. Artificial Intelligence, 1997, 90: 281 - 300.

[6] Q A Liang, S YW Su. And/or graph and search algorithm for discovering composite web services[J]. International Journal of Web Services Research, 2005, 2(4): 48 - 67.

[7] Zhifeng Gu, Bin Xu, Juanzi Li. Inheritance-aware document-

driven service composition[A]. Proceedings of CEC/EEE'07 [C]. Tokyo, Japan: IEEE Computer Society, 2007. 513 – 516.

[8] Liangzhao Zeng, B Benatallah, AHH Ngu, M Dumas, J Kalagnanam, H Chang. Qos-aware middleware for web services composition[J]. IEEE Transactions on Software Engineering, 2004, 30(5): 311 – 327.

[9] Mahsa Naseri, Ahmad Towhidi. Qos-aware automatic composition of web services using ai planners[A]. Proceedings of the Second International Conference on Internet and Web Applications and Services[C]. Washington DC, USA: IEEE Computer Society, 2007. 29 – 29.

[10] Tao Yu, Yue Zhang, Kwei-Jay Lin. Efficient algorithms for web services selection with end-to-end qos constraints[J]. ACM Trans Web, 2007, 1(1): 6.

[11] 万长林,韩旭,牛温佳,王文杰,史忠植.基于动态描述逻辑的服务组合及质量模型[J].电子学报,2010,38(8):1923 – 1928.
Wan Changlin, Han Xu, Niu Wenjia, Wang Wenjie, Shi Zhongzhi. Dynamic description logic based web service composition and QoS model[J]. Acta Electronica Sinica, 2010, 38(8): 1923 – 1928. (in Chinese)

[12] 张广泉,戎玫,朱雪阳,何亚丽,石慧娟.基于 XYZ/ADL 的Web 服务组合描述与验证[J].电子学报,2011,39(3A):86 – 93.
Zhang Guangquan, Rong Mei, Zhu Xueyang, He Yali, Shi Huijuan. Specification and verification of web service composition based on XYZ/ADL[J]. Acta Electronica Sinica, 2011, 39(3A): 86 – 93. (in Chinese)

[13] 王文彬,孙其博,赵新超,杨放春.基于非均衡变异离散粒子群算法的 QoS 全局最优 Web 服务选择方法[J].电子学报,2010,38(12):2774 – 2779.
Wang Wenbin, Sun Qibo, Zhao Xinchao, Yang Fangchun. Web services selection approach with QoS global optimal based on discrete particle swarm optimization with non-uniform mutation algorithm[J]. Acta Electronica Sinica, 2010, 38(12): 2774 – 2779. (in Chinese)

[14] Xianrong Zheng, Yuhong Yan. An efficient syntactic web service composition algorithm based on the planning graph model [A]. Proceedings of the 2008 IEEE International Conference on Web Services[C]. Washington DC, USA: IEEE Computer Society, 2008. 691 – 699.

作者简介

徐　猛　男,1978 年 10 月出生于青海格尔木.博士研究生,研究方向为软件与数据工程.
E-mail: xumeng_wf@163.com

崔立真　男,1976 年生,博士,副教授,主要研究方向为软件与数据工程、服务计算.
E-mail: clz@sdu.edu.cn

李庆忠(通信作者)　男,1965 年生,博士,教授,博士生导师,研究领域是大规模网络数据管理、数据集成等.
E-mail: lqz@sdu.edu.cn