

基于图的 Web 服务组合的研究*

王一飞, 吴素芹, 王 榕

(盐城工学院 信息工程学院, 江苏 盐城 224051)

摘要: 从 Web 服务质量入手, 将 QoS 参数属性作为图的权重, 提出了基于图的 Web 服务组合建模, 并介绍了用不同的算法分别在模型上寻找解决方案。

关键词: Web 服务; 服务组合; 服务选择

中图分类号: TP393

文献标识码: A

Research of Web services composition based on graph

WANG Yi Fei, WU Su Qin, WANG Rong

(College of Information Engineering, Yancheng Institute of Technology, Yancheng 224051, China)

Abstract: This paper start from the QoS, give a Web service composition based on the graph, and find the solution of composition on the model using different algorithm.

Key words: Web services; services composition; services selection

近年来, 基于 XML 的 Web 服务技术迅速发展, 为互联网应用提供了一种共享数据的有效手段。Web 服务的高效执行方式, Web 服务与其他成熟技术的有机结合以及 Web 服务的组合是解决现实应用问题的重要技术。

Web 服务是一种自包含、自描述、模块化的程序, 它吸收了分布式计算、Grid 计算和 XML 等各种技术的优点, 解决了异构分布式计算以及代码与数据重用等问题, 具有高度的互操作性、跨平台性和松耦合性, 引起了世界范围内学术界和工业界的极大兴趣^[1-2]。

为满足 Web 服务的技术需求, W3C 等国际标准组织制定了一系列 Web 服务规范, 如统一描述发现集成 UDDI (Universal Description, Discovery and Integration)、简单对象访问协议 SOAP (Simple Object Access Protocol)、Web 服务描述语言 WSDL (Web Service Description Language) 等, 都是用于描述、发布、发现和调用 Web 服务的, 这些规范共同构成了 Web 服务的技术体系^[3]。

Web 服务所执行的功能可以从简单的请求到复杂的商业过程中的任何事, 然而单个 Web 服务的功能有限, 难以满足实际应用中的多种多样的需求, 因此, 为了更加充分地利用共享的 Web 服务, 有必要将共享的

Web 服务组合起来, 提供功能更为强大的服务。Web 服务组合是个非常复杂的问题^[4-7], 它涉及到 Web 服务的描述、Web 服务的发现^[8]、Web 服务的选择^[9]、Web 服务的匹配、Web 服务的调度^[10-11]、服务质量 (QoS)^[12-13]等一系列问题。

1 基于图的 Web 服务组合模型

组合服务是通过一系列数据流和控制流联系在一起的基本服务, 研究 Web 服务组合问题, 也就是研究如何找到一组需要调用的 Web 服务, 以及以何种方式进行调用这些 Web 服务。图论中的点可以代表基本服务, 边可以代表 Web 服务的调用, 再把 QoS 属性参数^[12-13]也加入图中, 这样就可以用 1 个带有权值的有向图来表示服务的组合问题, 可以根据图论中的算法和服务的属性参数, 在图中寻找最优的服务组合方案。

把组合 Web 服务问题看成在 1 个带有权值的有向图 $G=(V, E)$ 中寻找最优路径的问题, 其中 V 代表 Web 上服务的集合, E 代表 Web 上两个服务之间的连接的集合。另外定义 n 为图中节点的数量, e 代表图中边的数量, 也就是 $n=|V|, e=|E|$ 。

每个 Web 服务被表示为图中的一个节点 (node), Web 服务的 QoS 参数 (如响应时间、费用、可靠性、可用性等) 可表示为节点的权值。然而有关图的算法通常把

* 基金项目: 国家自然科学基金项目 (60673175); 盐城工学院科技项目 (XKY2007069)

网络与通信 Network and Communication

权值定义在边上,而不是在节点上,所以本研究把原来在节点上属性的权值(响应时间 T , 费用 C , 可靠性 R , 可用性 A 等)转移到边上。

基于图的 Web 服务组合模型构造如下:

- (1) 图中的每一个节点代表一个 Web 服务。
- (2) 在图开始处加一个开始节点, 在结束时加一个结束节点。
- (3) 同一列的节点代表是同一个服务社区的服务。
- (4) 图中边代表该边的前驱节点所代表的 Web 服务可以调用该边后继节点所代表的 Web 服务。
- (5) 边上的权值代表该边后继节点所代表的 Web 服务的 QoS 的综合参数。

图 1 是构造的 Web 服务组合模型图, 它有 4 个服务社区: 第 1 个服务社区有 4 个候选服务, 第 2 个服务社区有 2 个候选服务, 第 3 个服务社区有 4 个候选服务, 第四个服务社区有 3 个候选服务, 边上的值是边后继节点所代表服务的 QoS 的综合参数。

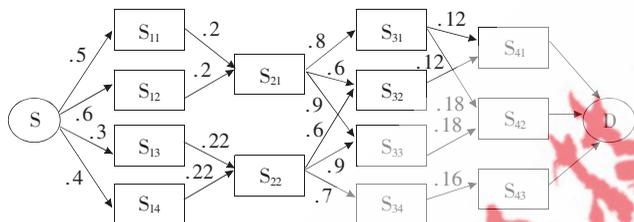


图 1 基于图的 Web 服务组合模型

图 1 中符号的含义:

- (1) S_i : 代表一个服务社区, 它是一个自治的系统, 包含着多个功能相同或相似但非功能(例如价格、响应时间等)不同的单个 Web 服务。
- (2) S_{ij} : 是一个基本 Web 服务, 它在 Web 服务社区 S_i 中, 其中 j 是社区为了标识 Web 服务的标记。
- (3) S : 起始节点, 代表组合服务的开始。
- (4) D : 目标节点, 代表组合服务的结束。

2 基于图的 Web 服务组合算法

要从图中找出 S 到 D 的路径, 并能够同时满足多种 QoS 约束的可行路径。本文主要考虑多重 ($k \geq 2$) 路径约束(也称为多约束)的情况。由于多约束的 QoS 是 NP 完全问题, 为此, 研究人员设计了很多启发式算法。然而这些算法往往具有很大的局限性: (1) 计算复杂度过高, 无法应用到实际环境中; (2) 算法性能较差, 不能找到实际存在的可行路径; (3) 算法只是针对某些特殊情况而设计, 不具有普遍性。

本文则将多种 QoS 度量转化为单一的权值, 然后用几种算法对整个图进行计算最短路径。使用加权公平队列, 费用、响应时间、可用性和可靠性都可以转化为函数中的参数, 不再彼此独立。这样, 原来多约束的 NP 完全问题就可以简化为多项式的复杂度。

在此介绍几种算法求解满足约束条件的方案。

(1) 扩散法。该算法可以找出所有可能执行组合的路径, 其思想是从源点开始, 通过逐个询问图中的其他节点, 逐步逼近并达到目标节点。首先使用广度优先的方法, 在多条路径中寻找可行路径, 在询问抵达目标节点之前, 不能断定可行路径。为了避免回路和扩散重复信息, 节点需要记录大量探测数据。

扩散算法在节点比较少的时候可以用来寻找满足约束条件的最优解决方案, 但当节点较多时, 它的时间复杂度会呈指数级上升, 所以在大规模问题时不为人们所接受。

(2) Dijkstra 算法。在经典的图论问题中, 可以使用 Dijkstra 算法计算图中任意 2 个节点的最短路径^[4]。本研究把该算法的思想应用在服务组合过程中, 求解从 S 到 D 的路径, 并且使服务质量函数最小。具体求解步骤如下:

① 首先进行初始化, 用 $d[v]$ 表示从服务开始到调用到服务 v 的质量函数值, 由于刚开始, 所以除了服务开始节点 s 外, 所有的节点 $d[v]$ 都设为无穷大, $d[s]=0$; $p[v]$ 表示调用服务 v 的服务节点, 开始时都没有调用, 所以 $p[v]$ 都设置为未定义。

② 设置 1 个集合 S , 表示已经找到的符合要求的服务节点, 一开始设置为空集合; 设置 1 个集合 Q , 表示所有的服务节点, 当算法找到 1 个符合要求的节点时, 则把该节点从 Q 中删除, 加入到 S 集合中。

③ 当集合 Q 不空时, 从集合中找出参数最小的服务节点 u , 并且把它加到集合 S 中, 对于每个 u 服务可以调用的服务, 如果 $d[v]$ 大于 $d[u]$ 与 $w[u, v]$ 的总和, 则把 $d[u]$ 与 $w[u, v]$ 的总和赋值给 $d[v]$, 同时把 $p[v]$ 设置为 u 。

④ 回到步骤 3, 直到 Q 集合为空时, 算法结束。

具体算法如图 2 所示, 在这个算法中, 边上的权值不能是负值, 如果存在负值的话, 在选取最小值时, 就会出错。

```

1 function Dijkstra (G, w, s) //G=Graph, w=weight, s=source
2   for each vertex v in V[G] // Initializations
3     d[v] := infinity
4     previous[v] := undefined
5   d[s] := 0
6   S := empty set
7   Q := set of all vertices
8   while Q is not an empty set
9     u := Extract_Min(Q)
10    S := S union {u}
11    for each edge (u, v) outgoing from u
12      if d[v] > d[u] + w(u, v) // Relax (u, v)
13        d[v] := d[u] + w(u, v)
14        previous[v] := u

```

图 2 Dijkstra 算法伪代码

Dijkstra 算法从图中节点入手进行寻找最优解决方案, 另外还可以从图中的边入手开始寻找。

网络与通信 Network and Communication

(3) Bellman-Ford 算法。是求解单源点的最短路径问题的一种算法^[4]。单源点的最短路径问题是指：给定一个加权有向图 G 和源点 s ，对于图 G 中的任意一点 v ，求从 s 到 v 的最短路径。

Bellman-Ford 算法如图 3 所示，它和 Dijkstra 算法很相似，但 Dijkstra 算法中的权值不能为负，因为如果有负值，则出现负循环时，就找不到最优路径，Bellman-Ford 算法比它多了检查的步骤（行 10-12），这样如果权值为负值，会返回错误。Bellman-Ford 算法可以在伪多项式时间内完成。

```

1 Bellman-Ford ( $G, w, s$ ) //  $G$ =Graph,  $w$ =weight,  $s$ =source
2 Determine Single Source ( $G, s$ )
3 Set Distance( $s$ )=0; Predecessor( $s$ )=nil;
4 For each vertex  $v$  in  $G$  other than  $s$ ,
5   set Distance( $v$ )=infinity, Predecessor( $v$ )=nil;
6 for  $i < -1$  to  $|V(G)| - 1$  do //  $|V(G)|$  Number of vertices in the graph
7   for each edge  $(u, v)$  in  $E[G]$  do
8     if Distance( $v$ ) > Distance( $u$ ) +  $w(u, v)$  then
9       set Distance( $v$ ) = Distance( $u$ ) +  $w(u, v)$ , Predecessor( $v$ ) =  $u$ ;
10  for each edge  $(u, r)$  in  $E[G]$  do
11    if Distance( $r$ ) > Distance( $u$ ) +  $w(u, r)$ ;
12    return false; // This means that the graph contains a cycle of negative weight
13 // and the shortest paths are not well defined
14 return true; // Lengths of shortest paths are in Distance array

```

图 3 Bellman-Ford 算法伪代码

Web 服务能够较好地解决异构应用之间、松散耦合环境下的互操作、集成和协作问题，成为国内外软件技术研究的重要方向。Web 服务的组合是正在兴起的新技术，它将彻底改变提供电子商务和客户软件应用的方式，是国内外在信息集成、软件工程等领域的关注的焦点，也是 Web 服务技术的主要发展方向之一。

本文主要关注 Web 服务的组合，将 QoS 参数属性作为图的权值，提出了基于图的 Web 服务组合建模，使用扩散算法、Dijkstra 算法、Bellman-Ford 等算法在模型中寻找组合的路径，并将各个算法进行比较，为 Web 服务的组合做好了理论准备。

Web 服务是处在动态的互联网上，在过去几年里 Web 上提供的 Web 服务数量急剧增多，通过人工在巨大的 Web 服务仓库中发现人们需要的服务是不现实的，所以服务的自动发现是服务调度的前提，我们会继续对服务的自动发现、服务的组合模型、服务的自动组合做进一步研究。

参考文献

[1] 岳昆, 王晓玲, 周傲英. Web 服务核心支撑技术: 研究综述[J]. 软件学报, 2004, 15(3): 428-442.

[2] 李曼, 王大治, 杜小勇, 等. 基于领域本体的 Web 服务动态组合[J]. 计算机学报, 2005, 28(4): 644-650.

[3] DAVIES N J, FENSEL D, RICHARDSON M. The future of Web services [J]. BT Technology Journal, 2004, 22 (1): 118-130.

[4] MILANOVIĆ N, MIROSLAW M. Current solutions for Web service composition [J]. IEEE Internet Computing, 2004, 26 (5): 51-59.

[5] MEDHAHED B, BOUGUETTAYA A, ELMAGARMID A K. Composing Web services on the semantic Web [J]. The VLDB, 2003, 12(4): 333-351.

[6] STEFAN T, RANIA K, THOMAS M. Composition of coordinated Web services [C]. IFIP International Federation for Information Processing, 2004: 294-310.

[7] BOUALEM B, MARLON D, QUAN Z. Declarative composition and peer-to-peer provisioning of dynamic Web services [C]. In the Proceedings of the 18th International Conference on Data Engineering, 2002.

[8] 张智, 李瑞轩. 基于对等网的 Web 服务发布和发现机制研究 [J]. 计算机工程与设计, 2006, 27(16): 2949-2951.

[9] MASSIMO M, ALESSANDRA M, ALESSANDRO P. Declarative policies for Web service selection [C]. In the Proceedings of the Sixth IEEE International Workshop on Policies for Distributed Systems and Networks, 2005.

[10] 谷清范, 吴介一, 张斌兵. 网格调度机制研究综述[J]. 计算机应用研究, 2006, 23(5): 1-4.

[11] 官荷卿, 张文博, 魏俊, 等. 一种应用敏感的 Web 服务请求调度策略[J]. 计算机学报, 2006, 29(7): 1189-1198.

[12] 单志广, 林闯, 肖人毅, 等. Web QoS 控制研究综述[J]. 计算机学报, 2003, 27(2): 145-156.

[13] ZENG Lang Zhao, BENATALLAH B, DUMAS M. Quality driven Web services composition [J]. IEEE Transaction on Software Engineering, 2004, 30(5): 311-327.

(收稿日期: 2009-08-10)

作者简介:

王一飞, 男, 1982 年生, 讲师, 东南大学硕士, 主要研究方向: Web 服务组合、工作流。

吴素芹, 女, 1979 年生, 讲师, 主要研究方向: 数据库应用系统开发、计算机图形设计。

王榕, 女, 1982 年生, 讲师, 主要研究方向: 邮件过滤。