

# 普适计算中面向最优质量的服务复用方法

田鹏伟, 方存好, 钟 鸣

(清华大学计算机科学与技术系信息科学与技术国家实验室, 北京 100084)

**摘 要:** 针对普适计算模式下的服务复用及服务质量问题, 提出一种面向最优质量的服务复用方法。通过建立服务功能倒排索引完成已有服务的功能匹配, 采用多属性决策理论进行服务质量的量化和评价, 并利用分枝定界法搜索最优质量的服务组合, 生成满足用户需求的服务。实验结果验证了该方法在提高生成服务质量方面的有效性。

**关键词:** 普适计算; 服务质量; 服务复用

## Optimal Quality-oriented Service Reuse Method in Pervasive Computing

TIAN Peng-wei, FANG Cun-hao, ZHONG Ming

(National Laboratory of Information Science and Technology, Department of Computer Science and Technology,  
Tsinghua University, Beijing 100084)

**【Abstract】** For the service reuse problem in pervasive computing, an optimal quality-oriented approach is proposed. The approach achieves service functionality match based on an inverted index, evaluates service quality with the multiple attribute decision making theory, and employs branch and bound algorithm for optimal composition plan generation. Experimental results demonstrate the effectiveness of the proposed approach in improving the quality of generated services.

**【Key words】** pervasive computing; Quality of Service(QoS); service reuse

### 1 概述

普适计算是一种以人为中心的计算模式<sup>[1]</sup>。在普适计算环境下, 复用其中部署的各种设备提供的服务, 为用户提供按需服务是一个重要问题。不同设备提供的服务质量并不相同, 于是如何生成质量最优的按需服务通常为用户所关心。

在最优质量的服务生成方面, 已有大量的研究工作在开展, 但基本方法比较有限<sup>[2-3]</sup>, 广泛采用的思路是认为需求中的每个基本功能对应一个实现该功能的服务集合, 为每个功能选择一个服务, 使得整体服务质量达到最优。主要实现方法是整型规划<sup>[2]</sup>。

已有方法假设了需求中的每个基本功能由单一的服务来实现<sup>[2]</sup>, 该假设存在较大局限性。在普适计算环境中, 除了各设备提供的服务, 已经生成的按需服务也是一类重要的服务资源。这类服务以组合方案的形式定义, 通常覆盖多个基本功能, 在现有方法中这类服务是难以被复用的。

为了复用已生成的按需服务中的功能相似部分, 减少生成按需服务组合方案的复杂性, 本文从每个可复用服务实现的功能需求部分出发, 利用分枝定界法<sup>[4]</sup>在可复用服务及相应的复用操作中动态选择, 使得生成的服务达到质量最优。

### 2 服务功能和功能需求的表示

每个服务都有对应的应用领域, 相应的描述如图1所示。

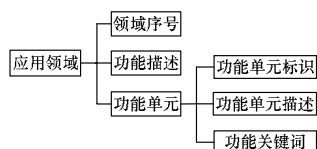


图1 应用领域的定义

功能单元定义了领域中最基本的功能操作, 每个功能单元通过一组关键词描述, 同时有全局唯一的标识。

普适计算环境中的服务通常可以分为2类:

(1)基本服务: 单一设备提供的服务, 一个基本服务可能提供多个操作。

(2)组合服务: 通过调用基本服务完成其功能的已有服务, 由相应的服务组合方案来定义。

每个基本服务及相应操作的功能由其对应的功能单元描述; 每个组合服务的功能, 由其组合方案中包含的各基本服务的功能及服务间的组合方式决定。基本服务的组合方式由3种控制结构来建模<sup>[2]</sup>: 顺序, 选择和并行结构。从而一个组合服务的功能可以表示为功能单元的组合, 用巴科斯范式文法定义为

$$F ::= \varepsilon \mid f \mid F \oplus F \mid F \otimes F \mid F \parallel_c F \quad (1)$$

其中,  $\varepsilon$  表示一个空的功能单元;  $f$  表示只包含单一功能单元的组合;  $\oplus$ 、 $\otimes$  和  $\parallel_c$  分别是顺序、并行和选择结构操作符。

服务的功能需求包括2个方面: 用户需要的功能单元以及执行的业务流程。可以用式(1)中定义的操作符来定义业务流程, 从而功能需求也表示为功能单元的组合。

### 3 服务质量属性的定义和求解

#### 3.1 服务质量属性的选取

目前, 已有很多服务质量属性被提出<sup>[2]</sup>, 如响应时间、

**基金项目:** 国家自然科学基金资助项目(90604027)

**作者简介:** 田鹏伟(1981-), 男, 博士研究生, 主研方向: 主动服务, 服务复用; 方存好, 助理研究员; 钟 鸣, 博士研究生

**收稿日期:** 2010-02-20 **E-mail:** tpw04@mails.tsinghua.edu.cn

价格、执行成功率、用户满意度等。本文基于两方面的考虑对普适计算环境中的服务质量属性进行选择。

(1)服务质量的客观性：服务的质量属性是其在执行过程中客观存在的,应该尽量避免主观的不稳定因素造成的误差。因此,本文主要选择客观的质量属性,暂时不考虑用户满意度、服务评价等需要用户主观去测量的标准。

(2)普适计算环境的可控性：普适计算环境中提供服务的设备是可管理和可维护的,服务传输网络的质量通常也可以保证。在这种情况下,只要调用方式和参数正确,服务能够正确执行和返回。因此,服务可靠性方面的属性,如执行成功率、可用性等,通常可以忽略。

综合上述两方面因素,本文考虑2类服务质量属性：服务的响应时间,价格。这2类属性均为成本型属性,其值越大,代表的质量越差。

### 3.2 基本服务的质量属性

给定一个基本服务  $s$  和其提供的功能操作  $op$ , 质量属性主要包括：

(1)响应时间  $q_{ic}(s, op)$ ：响应时间是指从服务调用请求发出到结果返回这一过程所需的时间。

(2)价格  $q_{pr}(s, op)$ ： $q_{pr}(s, op)$  表示调用服务  $s$  中的操作  $op$  时需要支付的费用。

### 3.3 组合服务的质量属性

组合方案  $cp$  定义的组合服务的质量属性由  $cp$  中的基本服务决定：

(1)响应时间  $q_{ic}(cp)$ ：基于文献[2]中采用的关键路径法对组合方案进行分析,求得组合服务响应时间。

(2)价格  $q_{pr}(cp)$ ：组合服务的价格是其组合过程中所调用的所有基本服务的价格的总和。

响应时间和价格2个质量属性构成了基本服务和相应操作以及组合服务的质量属性向量。

## 4 面向最优质量的服务复用

### 4.1 服务的索引和匹配

为了进行高效的服务功能匹配和复用,对已有的服务建立基于功能的倒排索引。本文基于功能单元建立服务索引,其中以功能单元的标识为索引键,包含该功能单元的服务信息作为索引项。采用传统的二叉搜索树<sup>[4]</sup>作为索引结构,索引的构建、匹配可以结合相应算法<sup>[4]</sup>来完成。

### 4.2 服务质量的评价

服务的质量由各个质量属性共同决定,可以利用多属性决策理论对其进行评价<sup>[2,5]</sup>。具体地说,就是采用简单加权法<sup>[5]</sup>完成服务质量的评价。

给定一个包含  $n$  个候选服务的集合： $S=\{s_i | 0 \leq i < n\}$ , 所有服务的质量属性向量构成  $S$  的质量矩阵：

$$Q(S)=[q_{ij} | q_i = q(s_i); 0 \leq i < n; 0 \leq j \leq 1] \quad (2)$$

其中,  $q(s_i) = \langle q_{ic}(s_i), q_{pr}(s_i) \rangle$ 。不同质量属性值的度量单位不同,首先要对各属性标准化<sup>[2,5]</sup>,采用线形尺度变换法<sup>[5]</sup>完成成本型属性的标准化：

$$q'_{ij} = \frac{\min_{0 \leq k < n} (q_{kj})}{q_{ij}}, 0 \leq j \leq 1 \quad (3)$$

其中,  $\min_{0 \leq k < n} (q_{kj})$  表示  $Q(S)$  中第  $j$  列的最小值。

将服务质量矩阵  $Q(S)$  中每一个元素  $q_{ij}$  转化为其标准化值  $q'_{ij}$ , 形成标准化的服务质量矩阵  $Q'(S)=[q'_{ij} | 0 \leq i < n; 0 \leq j \leq 1]$ 。基于  $Q'(S)$ , 服务集合  $S$  中的任一服务  $s_i$  的服务质量可以量化

为其服务质量分数<sup>[2,5]</sup>：

$$QScore(s_i) = \sum_{j=0}^1 (q'_{ij} \times W_j) \quad (4)$$

其中,  $W_j$  表示第  $j$  列对应的质量属性的权重,满足  $W_j \in [0,1]$  且  $\sum_{j=0}^1 W_j = 1$ , 由用户根据其偏好来设定,系统默认各质量属性权重相同(均为 0.5)。基于式(4)计算出的质量分数,在服务候选集合  $S$  中,质量分数越高,则服务质量越好。

### 4.3 面向最优质量的服务复用过程

在服务复用过程中,在所有可复用服务及相应的复用操作(构成搜索分枝)中,引入分枝定界法<sup>[4]</sup>进行最优质量服务搜索,保证最终生成的服务质量最优,如下所示,分为2个阶段：服务匹配(第1行~第10行)和复用(第11行~第35行)。

**输入** 服务的功能需求  $FSR$  基于功能单元的服务索引  $FUIndex$ ；分枝定界法中搜索分枝数  $T$

**输出** 具有最优质量的按需服务

伪代码描述：

/\*可复用服务的匹配\*/

1 SerTab  $\leftarrow$  null ;/\*匹配结果哈希表初始化为空\*/

2 for  $i=0$  to  $|FSR|-1$  do

3 Siset  $\leftarrow$  FSR[i] 在  $FUIndex$  匹配的服务集合;

4 for  $j=0$  to  $|Siset|-1$  do

5 SerTab.[Siset[j].SI]  $\leftarrow$  FSR[i];

6 end for

7 end for

8 SQRes  $\leftarrow$  {<SI,FUS>}, FUS=SerTab[SI] &  $|FUS|>1$ ;

9 将 SQRes 的结果按  $|FUS|$  从大到小排序;

10 FURes  $\leftarrow$  {<SI,FU>}, FU=SerTab[SI] 且  $|FU|=1$ ;

/\*面向最优质量的服务复用\*/

11 remain  $\leftarrow$  T; /\*记录未搜索完的分枝数\*/

12 CP[0],CP[1],...,CP[T-1]  $\leftarrow$  FSR; /\*初始化为 FSR \*/

13 QS[0],QS[1],...,QS[T-1]  $\leftarrow$  0; /\*初始质量分数为 0\*/

14 R[0],R[1],...,R[T-1]  $\leftarrow$  0; /\*SQRes 中服务未被复用\*/

15 while remain  $\neq$  0 do

16 Cur  $\leftarrow$  {m|QS[m] = max(QS[i]), CP[m]  $\neq$  null};

17 if CP[Cur] 中只包含服务/操作 then

18 返回 CP[Cur];

19 else if R[Cur] == 0 then /\*组合服务的复用\*/

20 利用 SQRes[Cur].SI 的可复用部分,在 CP[Cur] 中实现功能单元 SQRes[Cur].FUS;

21 QS[Cur]  $\leftarrow$  QScore(CP[Cur]); /\*更新质量分数\*/

22 R[Cur]  $\leftarrow$  1;

23 else /\*基本服务的复用\*/

24 fu  $\leftarrow$  CP[Cur] 中的任一功能单元;

25 SI  $\leftarrow$  fu 在 FURes 对应的基本服务;

26 if SI == null then

27 remain--;

28 CP[Cur]  $\leftarrow$  null;

29 else

30 用 SI 在 CP[Cur] 实现功能单元 fu;

31 QS[Cur]  $\leftarrow$  QScore(CP[Cur]);

32 end if

33 end if

34 end while

35 返回 null;

在可复用服务的匹配阶段,利用服务功能需求  $FSR$  中的每一个功能单元在建立的服务索引  $FUIndex$  中进行服务匹配,将匹配结果按照服务归类保存到哈希表 SerTab 中(第4行~第6行)。

在面向最优质量的服务复用阶段，基于匹配到的前  $T$  个可复用的组合服务构成搜索空间。每次选择质量分数最高的分枝实施服务复用(第 19 行~第 31 行)。当质量分数最高的搜索分枝所对应的组合方案已经完整时，则得到质量最优的组合方案，将其返回(第 18 行)。返回的服务组合方案通过执行引擎执行，即可生成按需的服务。

## 5 测试和分析

在局域网环境中，对本文提出的面向最优质量的服务复用方法进行了实现和部署。其中，服务索引/匹配部分部署在一台 Intel Pentium 2.8 GHz CPU、1 GB 内存的 PC 上；服务复用部分运行在一台 Intel Core II 2 GHz CPU、1 GB 内存的笔记本电脑上。在测试中，以科学计算作为应用领域来模拟一个普适计算环境：以 10 个基本运算作为功能单元，生成 300 个服务作为设备提供的基本服务，进而生成 100 个已有的组合服务。

在质量属性方面，对于功能表示中包含  $n$  个功能单元的基本服务，将其响应时间和价格分别赋值为区间  $[n, 60n]$  和  $[n, 200n]$  中的一个随机数。基于此求得每个组合服务的质量属性。

### 5.1 质量分数

首先对文中提出的服务复用方法所生成服务的质量分数进行测试。将该方法和基本的服务复用方法进行比较，后者只要生成了完整的服务组合方案则立即返回，不考虑其质量。

在测试过程中，从已有服务的功能表示中提取包含 20、60 和 100 个功能单元的组合各 5 个作为功能需求，测试结果如图 2 所示。可以看出，在 3 类需求下，文中提出的方法一直能取得优于基本服务复用方法的服务质量，当需求中的功能单元比较少时，优势更为明显。

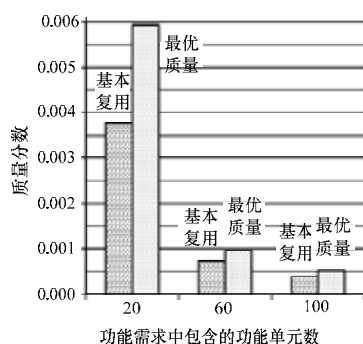


图2 质量分数测试结果

### 5.2 时间性能

进一步的，对本文提出的服务复用方法的时间性能进行

测试，同样与基本的服务复用方法比较。测试中从已有的服务功能表示中提取包含 5、10、20、30、40 个功能单元的组合各 5 个作为服务功能需求。测试结果如图 3 所示。可以看到，基本复用方法的时间性能较好。当需求中包含的功能单元多于 40 个时，平均需要 50 s 以上的时间来生成近似质量最优的服务组合方案。实际上，在现有研究工作中，最优质量的服务生成一直是复杂性较高的过程<sup>[2]</sup>。当需求中的功能单元较少时，本文的方法具有可接受的时间性能。

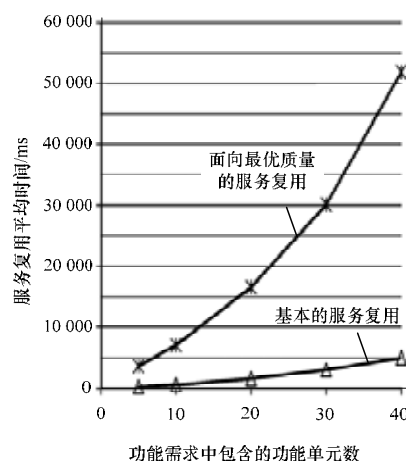


图3 时间性能测试结果

## 6 结束语

针对普适计算环境中的服务复用问题，本文提出了一种面向最优质量的服务复用方法。该方法基于分枝定界法完成最优质量的服务搜索和服务方案的生成。实验结果验证了该方法在提高服务质量方面的有效性。

## 参考文献

- [1] Weiser M. The Computer for the 21st Century[J]. Scientific American, 1991, 265(3): 94-104.
- [2] Zeng Liangzhao, Boualem B, Anne H H, et al. QoS-aware Middleware for Web Services Composition[J]. IEEE Transactions on Software Engineering, 2004, 30(5): 311-327.
- [3] 郭得科, 任彦, 陈洪辉, 等. 一种基于 QoS 约束的 Web 服务选择和排序模型[J]. 上海交通大学学报, 2007, 41(6): 870-875.
- [4] 严蔚敏, 吴伟民. 数据结构[M]. 北京: 清华大学出版社, 1997.
- [5] 刘树林. 多属性决策理论方法与应用研究[D]. 北京: 北京航空航天大学, 1997.

编辑 顾逸斐

(上接第 12 页)

## 参考文献

- [1] Abrial J R. B 方法[M]. 裴宗燕, 译. 北京: 电子工业出版社, 2004.
- [2] ISO/IEC. ISO/IEC 20926-03 Software Engineering——IFPUG 4.1 Unadjusted FSM Method[S]. 2003.
- [3] 李帆, 林立新. 软件工程项目管理——功能点分析方法与实践[M]. 北京: 清华大学出版社, 2005.

- [4] Rossi P, Fernandez G. Definition and Validation of Design Metrics for Distributed Applications[C]//Proceedings of the 9th International Software Metrics Symposium. Sydney, Australia: [s. n.], 2003.
- [5] Lai R, Huang Sun-Jen. A Model for Estimating the Size of a Formal Communication Protocol Specification and Its Implementation[J]. IEEE Transactions on Software Engineering, 2003, 29(1): 46-62.

编辑 索书志