

基于 QoS 的语义 Web 服务组合优化算法

王广正

(安徽工业大学 计算机学院, 安徽 马鞍山 243002)

摘 要: 在面向服务的环境下, 存在众多可满足服务请求者要求的服务组合方案, 如何从中选出最优或近似最优方案是 Web 服务组合优化要解决的问题. 设计出一个不必探索整个解空间的优化算法——SBG 算法, 为服务请求者提供最优选择方案. 实验结果表明, 这种方法可有效提高选择方案的求解效率.

关键词: 语义 Web 服务; 服务组合; 优化; SBG 算法

中图分类号: TP311

文献标识码: A

文章编号: 1000-7180(2011)07-0093-05

Optimal Algorithm for QoS-aware Semantic Web Service Composition

WANG Guang-zheng

(Department of Computer Science, Anhui University of Technology, Ma'anshan 243002, China)

Abstract: In service-oriented environments, there are a number of service composition schemes which can satisfy the needs of service requesters. Web service composition optimization is proposed to solve the problem of selecting the optimal scheme from the schemes. Therefore, the paper presents an optimized algorithm that finds the optimal solution for service requesters without having to explore the entire solution space. Experimental results show that the algorithm efficiency has been improved.

Key words: semantic web service; service composition; optimization; SBG algorithm

1 引言

Web 服务是一种运行在 Web 上的自包含、自描述、模块化的应用程序, 具有高度的互操作性、跨平台性和松耦合等特点. 用户可根据需要查找和调用 Web 服务, 并快速地设计、开发、部署和发布各种应用程序. 在 SOA 架构下, 可提供很多功能相似而服务质量(QoS)不同的单个 Web 服务. 单个 Web 服务往往不能满足用户的功能要求, 因此需进行 Web 服务组合. 与此同时, 语义网技术^[1]为 Web 服务提供了语义描述方法, 使得 Web 服务组合过程能够被智能主体自治地执行, 从而形成一种新的技术——语义 Web 服务. 关于语义 Web 服务组合的研究一直是焦点和难点问题^[2]. 若在组合时存在多个候选服务均可满足任务功能要求的情况, 则相应

可行 Web 服务组合方案总数就会以指数效应增长. 如何从中选择一个最优或近似最优的方案并将其返回给客户成为迫切需要解决的问题, 称为服务组合优化问题. 文中考虑的服务组合优化问题是找到可靠性最大且满足价格和响应时间约束的组合方案.

目前服务组合优化算法主要有两类: 整数规划^[3-5]和启发式方法^[6]. 用整数规划方法解决 Web 服务选择优化问题, 可为服务请求者提供最优选择方案. 用启发式方法也可为服务请求者提供最优或次优选择方案.

文中提出一种高效的解决问题的方法——SBG (Schrijver-Bound-Greed) 算法. 该方法以 Schrijver 的理论为基础^[7], 构造服务组合优化问题对应图, 然后利用限界和贪心思想求得图的最短路径, 即找到服务组合优化问题的最优解, 并通过大量满足均匀

收稿日期: 2010-11-08; 修回日期: 2010-12-05

基金项目: 国家自然科学基金项目(60473142); 安徽省教育厅自然科学研究项目(2006KJ238B); 安徽工业大学青年教师科研基金(QZ201015)

分布的随机数据对 SBG 算法进行实验以证明其有效性。

2 QoS 本体和服务组合问题描述

2.1 QoS 本体

(1) 价格 (Price)

价格(P)是指一个服务请求者访问服务所需的开销^[2,5,8]。通常服务的价格越高,服务所提供的功能越多,质量越高;所供服务功能相似条件下,价格越小越好,且通常服务请求者对完成某任务有一定预算,所以价格常作为服务组合优化中的约束条件,具有上界。

在所有服务组合的工作流结构中,包括环、并行活动和条件分支,组合服务的价格为各单个服务价格的线性组合,为和形式^[2,5,8]。

(2) 响应时间 (Response time)

响应时间(Rt)^[2,5,8]是指从服务请求者的请求消息发出到最终服务执行结束所需的时间开销。响应时间的快慢与服务质量有密切联系,响应时间越小越好,通常服务请求者会提出一定上界。

对不同的工作流结构,组合服务的响应时间计算公式不同,是非线性组合。在并行结构下, N 个活动执行的时间是各活动时间的最大值, $R(z) = \max_{i=1, \dots, n} \{Rt_i\}$ ^[5,8];其余流程结构为和的形式^[5,8]。为使约束简化,需对并行结构的响应时间转换为如下形式:

$$R(z) = \max_{i=1, \dots, n} \{Rt_i\} \leq Rt_{\max}$$

$$\Rightarrow \begin{cases} Rt_1 \leq Rt_{\max} \\ Rt_2 \leq Rt_{\max} \\ \dots \\ Rt_n \leq Rt_{\max} \end{cases}$$

(3) 可靠性 (Reliability)

Web 服务可靠性一般用失效的百分比标识其值。它是对服务质量的一个综合评判^[2],是 Web 服务选择的一个非常重要的质量度量标准,可以之作为质量目标构建 Web 服务选择优化问题。

其中可靠性子特性主要有:

- 成熟性度量:平均失效间隔时间(MTBF);
- 容错性度量:避免失效(Failure avoidance)和抵御误操作(Incorrect operation avoidance);
- 易恢复性度量:平均恢复时间(Mean recovery time)、易修复性(Restorability)和可用性(Availability)。

可靠性可以用平均失效间隔时间、避免失效、抵

御误操作、平均恢复时间、易修复性和可用性等标示可靠性值大小。

在所有服务组合的工作流结构中,组合服务的可靠性均为单个服务的乘积形式^[8],为容易求解,乘积形式求对数后变为和形式,且对数不改变服务可靠性的特性,转换后较大的值仍代表较好的质量,所以目标函数中系数为可靠性的对数形式。

2.2 服务组合优化问题

Web 服务组合优化问题是从众多可行的服务组合方案中寻找一个满足约束且质量最好的方案^[6,8],是一种特殊的整数规划问题。本文中的约束是指价格和响应时间约束,服务质量以可靠性为标准。服务组合优化问题可形式化如下:

$$\max R = \sum_{i=1}^n \sum_{j=1}^{l_i} x_{ij} \log rel_{ij} \quad (1)$$

$$P(n) = \sum_{i=1}^n \sum_{j=1}^{l_i} x_{ij} P_{ij} \leq P_{\max} \quad (2)$$

$$Rt(n) = \sum_i \sum_j x_{ij} Rt_{ij} \leq Rt_{\max} \quad (3)$$

$$\sum_{j=1}^{l_i} x_{ij} = 1, i \in \{1, 2, \dots, n\} \quad (4)$$

$$x_{ij} \in \{0, 1\} \quad (5)$$

式(1)为目标函数。期望找到可靠性最大的组合方案:

- n 为任务个数;
- l_i 为满足每个任务功能需求的候选服务数,

$$L = \max_{i=1}^n (l_i);$$

- rel_{ij} 为第 i 个任务的第 j 个候选服务的可靠性。

式(2)为价格资源约束条件。由 2.1 节可知,各种工作流结构均为和形式。 P_{ij} 为访问第 i 个任务的第 j 个候选服务的所需价格。

式(3)为响应时间资源约束条件。所有工作流结构均可转换为和形式,但 i 和 j 的取值需根据具体工作流结构来定,故不限定 i 和 j 的起、终值;如存在并行结构且此结构包括 n 个活动,则需 n 个表达响应时间约束。 Rt_{ij} 为访问第 i 个任务的第 j 个候选服务的所需响应时间。

式(4)为配置约束条件,以保证每个任务只选择一个候选服务。

式(5)为变量取值约束条件。 x_{ij} 为第 i 个任务的第 j 个候选服务的决策变量,确定选或不选第 i 个任务的第 j 个候选服务,取值为 0 或 1。

上述问题可采用穷尽搜索法^[9](时间复杂度为 $O(L^n)$) 或单纯形法^[10](时间复杂度为 $O(2^{n-1})$) 来解决,时间复杂度较高。

3 服务组合优化问题求解

文中提出一种高效的不需比较所有组合方案的 SBG 算法。利用 Schrijver 的理论,在构造服务组合优化问题对应的图中,根据边界去掉所有不可能的组合方案,在所有可行方案中根据贪心思想找到图的最长路径,即可找到组合优化问题最优解。

3.1 一种有效的组合优化算法——SBG

整数规划问题可表示为^[7,11]: $\text{Max}\{c^T x \mid Ax \leq b, x \in \{0,1\} \text{—向量}\}$, 其中 $A = (a_{ij}) \in Z^{m \times n}, b = (b_1, b_2, \dots, b_m) \in Z^m, c^T = (c_1, c_2, \dots, c_n)$ 。对服务组

合优化问题,具体参数为: $A = \begin{bmatrix} A_1 \\ A_2 \\ A_3 \end{bmatrix}$, 其中 $A_1 =$

$(P_{ij}), A_2 = (R_{ij}), A_3 = (\{-1, 1\}); b = (P_{\max}, R_{\max}, 1, 1, \dots, 1, -1, -1, \dots, -1); c^T = (\log \text{rel}_{ij})$ 。

整数规划对应的图 G 的信息如下^[7,11]:

- 图的顶点: $V = \{0, \dots, n\} \times \{-U, \dots, +U\}^m$, 其中 $U = (n+1)(mS)^m, S = \max\{|a_{ij}|, |b_i|, i = 1, \dots, m; j = 1, \dots, n\}$

- 图的边: $((i, u'), (j, u'')) \in E \Leftrightarrow j = i + 1, u'' = u' + kA_i, k \in \{0, 1\}$

- 边的长度:

$$l((i, u'), (j, u'')) = \begin{cases} -c_i, & \text{当 } u'' = u' + A_i \\ 0, & \text{当 } u'' = u' \end{cases}$$

其中,长度为 0 的称为水平情况,否则为倾斜情况。

整数规划解和图的路径间关系^[11]如下:如 x 是整数规划的一个可行解,当且仅当图 G 中存在一条从 $(0, 0)$ 到 (n, b') 的路径,且 $b' \leq b$ 。整数规划最优解即图 G 中从 $(0, 0)$ 到 (n, b') 的最短路径,且 $b' \leq b$ 。

服务组合优化问题和图的关系如下:因服务组合优化问题是特殊的整数规划问题,所以服务组合优化问题可转换为图 G 从 $(0, 0)$ 到 (n, b') 的最短路径问题,且 $b' \leq b$ 。

根据文献^[7,11]的结论可知,对服务组合优化问题,可构造其对应的图,顶点边界为 $0 \leq v_x \leq n, 0 \leq v_y(\text{资源}) \leq b(\text{资源}), v_y$ 中不存在大于 2 且小于任务个数的值,并得到其可行解。

具体的算法描述如下。

算法 SBG

输入:每个任务的候选服务对应的可靠性、价格和响应时间。

输出:最优服务组合方案。

① 初始化:

$vx=0$; %图 G 开始点横轴值(x)

$vy=zeros(1,m)$; %图 G 开始点纵轴值(y)

$dist=0$; %开始点到当前点的最长路径距离

$path=0$; %开始点到当前点的最长路径对应解二进制之和,如 110,则记录 6

$node=[vx \quad vy \quad dist \quad path]$; %node 存储当前点、到当前点%距离、路径值信息

② 构造图:

在构造图时利用上界除去不可行解,利用贪心思想记录开始点到当前的距离和路径值信息

for $i=1$ to $L1+L2+\dots+Ln$

$temp=[]$; %临时存储第 i 列结点信息

for $h=1$ to n_row % n_row :node 的行数

取出 node 中第 h 个结点(当前点)的纵轴值 vy 、距离 $distpre$ 和路径值信息 $pathpre$

%水平情况

if $temp$ 为空 或 $temp$ 中没有当前点

%把当前点插入到 $temp$ 中

$temp=[temp; i \quad vy \quad distpre \quad pathpre]$

else % 有此点,

if $distpre <$ 原有点距离 %比较哪个距离大

%留下小的(贪心思想)

$temp=[temp; i \quad vy \quad distpre \quad pathpre]$

end

end

%倾斜情况

if 响应时间和价格满足约束且配置约束中不存在大于 2 且小于任务个数的值

if $temp$ 中没有当前点

$temp=[temp; i \quad vy+A(1,i) \quad distpre+C(i) \quad pathpre+power(2,i)]$;

else %有此点, 比较哪个距离小

if $distpre+C(i) <$ 原有点对应距离

%留下小的(贪心思想)

$temp=[temp; i \quad vy+A(1,i) \quad distpre+C(i) \quad pathpre+power(2,i)]$;

end

end

end

$node = temp$

end

end

③ 找到 $temp$ 中对应最短路径值 $dmax$ 和路径解 $pMax$

④ 找到 $pMax$ 对应最短路径解,即最优组合方案

由上述描述可知,SBG 算法首先去除了不可行解,然后在所有可行解中找到最优解.因图的长度均为非负数,故用贪心和限界找到最短路径,即服务组合优化问题最优解.

3.2 案例分析

案例 1 网上转账组合服务包括两个原子服务:一个是查询余额服务,另一个是转账服务.第一任务查询余额服务有 2 个候选服务,第二任务转账服务有 3 个候选服务,每个服务的可靠性对数、价格和响应时间具体如下,根据式(1),可表示为:

$$\max R = -0.139x_{11} - 0.428x_{12} - 0.212x_{21} - 0.538x_{22} - 0.105x_{23}$$

s. t.

$$\begin{cases} 6x_{11} + 5x_{12} + 3x_{21} + 2x_{22} + 4x_{23} \leq 12 = P_{\max} \\ 10x_{11} + 13x_{12} + 30x_{21} + 19x_{22} + 33x_{23} \leq 40 = R_{\max} \\ x_{11} + x_{12} = 1 \\ x_{21} + x_{22} + x_{23} = 1 \\ x_{ij} \in \{0,1\} \end{cases}$$

从所有组合方案中选一种最优组合方案.约束可简化表示为 $Ax \leq b$ 的形式,则此时

$$A = \begin{bmatrix} 6 & 5 & 3 & 2 & 4 \\ 10 & 13 & 30 & 19 & 33 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 \\ -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & -1 \end{bmatrix}$$

$$b = \begin{bmatrix} 12 \\ 40 \\ 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}$$

步骤 1:初始化, $\text{node} = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$.

步骤 2:构造图:

当 $i=1$ 时:水平情况下生成结点 $[1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]$,不含此点,插入临时结点集 temp 中;

倾斜情况下生成结点 $[1 \ 6 \ 10 \ 1 \ 0 \ -1 \ 0 \ 0.319 \ 1]$,其中 $\text{vy} = [6 \ 10 \ 1 \ 0 \ -1 \ 0]$;判断是否超过界限:两个资源约束没超过界限 $\text{vy}(1,1:2) = [6 \ 10] \leq [12 \ 40]$,且保证了配置约束 $\text{vy}(1,3:6) = [1 \ 0 \ -1 \ 0]$ 中不含大于 2 且等于任务个数 2 的数;判断 temp 中是否含有此点,不含此点,插入 temp 中.故 $\text{node} = [1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0; 1 \ 6 \ 10 \ 1 \ 0 \ -1 \ 0 \ 0.319 \ 1]$.

步骤 3:同步骤 2,可得出 $i=2,3,4,5$ 时的结点

集,如图 1 所示.图 1 显示了所有可行的路径,其中 $i=5$ 时:

$$\text{node} = \begin{bmatrix} 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 4 & 33 & 0 & 1 & 0 & -1 & 0.105 & 1 \\ 5 & 2 & 19 & 0 & 1 & 0 & -1 & 0.538 & 2 \\ 5 & 3 & 30 & 0 & 1 & 0 & -1 & 0.212 & 4 \\ 5 & 5 & 13 & 1 & 0 & -1 & 0 & 0.428 & 8 \\ 5 & 7 & 32 & 1 & 1 & -1 & -1 & 0.966 & 10 \\ 5 & 6 & 10 & 1 & 0 & -1 & 0 & 0.319 & 16 \\ 5 & 8 & 29 & 1 & 1 & -1 & -1 & 0.857 & 18 \\ 5 & 9 & 40 & 1 & 1 & -1 & -1 & 0.531 & 20 \end{bmatrix}$$

步骤 4:找到 temp 中满足每个任务均选择一服务($\text{node}(:,4:5) == [1 \ 1]$)且路径最短的,满足条件的 $\text{dmax} = 0.5310$ 和路径解 $\text{pMax} = 20$.

步骤 5:根据 pMax ,得最优方案为 $1 \ 0 \ 1 \ 0 \ 0$,即选择第一任务的第一候选服务和第二任务的第一候选服务组合的方案可靠性最大.经验证所选组合方案为最优方案.

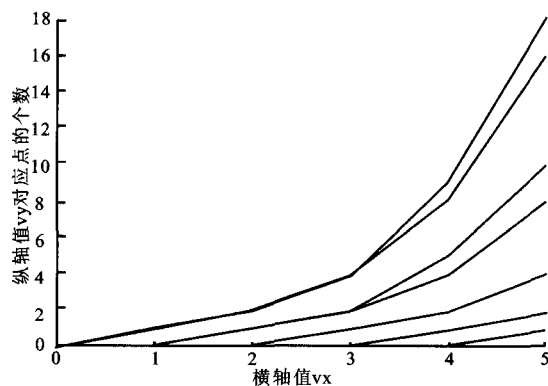


图 1 案例 1 服务组合优化对应图

图 G 的顶点: $V = \{0, \dots, n\} \times \{-U, \dots, +U\}^m$, 对图 G 中顶点 (X, Y) , $X \in \{0, \dots, n\}$, $Y \in \{-U, \dots, +U\}^m$. 图 1 横轴值 vx 表示顶点的横坐标 X , 因 Y 为多维向量, 故将 Y 转换为一维数据, 用图 1 的纵轴表示. 转换规则: 对每个横轴值, Y 是第几个生成, 则记为几, 故纵轴值 vy 表示对应点的生成次序, 即点的个数.

由案例 1 可知, 通过限界可去除大量无效结点, 使问题规模减少, SBG 算法时间复杂度降低, 且 SBG 算法为最优算法.

4 仿真实验

本节通过仿真实验的方法验证 SBG 算法的运行效率和优越性, 其中任务个数、候选服务个数和不满足需求率是决定算法性能和优越性的 3 个关键因素, 本仿真已在 2.4GHz, 2.0GB RAM 的计算机上使用

MATLAB7 实现,分别测试了上述 3 个因素对 SBG 算法性能影响程度,同时进行了优度实验。

4.1 任务数改变

为了分析组合任务数增长对 SBG 算法效率的影响,本组仿真设置了 10 组任务数,任务数从 5 增至 50;每组任务数产生 80 组测试用例,测试用例的数据均匀分布于区间(0,100);每个任务产生 30 个候选服务。设不满足需求率为 20%,即每组用例中共有 20%的组合方案不能满足客户需求。图 2 给出了该组实验的结果。

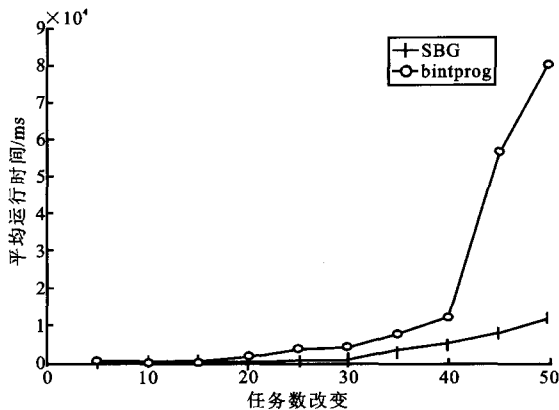


图2 候选服务个数为30的算法效率比较

该组实验表明,在候选服务数不变,任务数递增的情形下,相对已有的 Matlab 提供的求最优解方法(bintprog),SBG 算法具有良好的运行效率。

同效率实验一起,同时进行了优度实验,与 bintprog 所得最优值进行比较,结果显示最优值相同,即优度全部为 100%,因优度全部为 100%。

4.2 候选服务数改变

在该组实验中,任务数固定为 50,任务的候选服务个数从 5 增至 45,不满足需求率为 20%。不满足需求率由 20%上升至 80%。如图 3 所示。

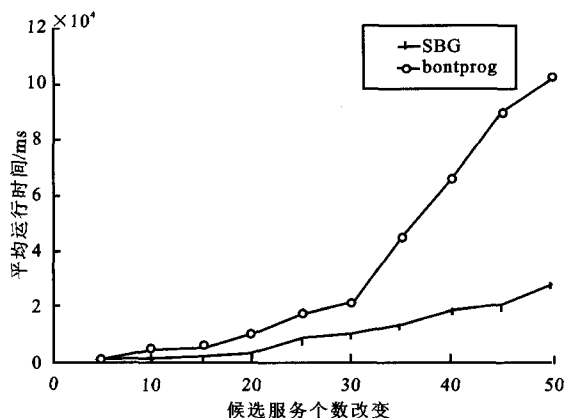


图3 任务个数为50的算法效率比较

该组实验表明,在任务数不变,候选服务数递增

的情形下,相对已有的 Matlab 提供的求最优解方法(bintprog),SBG 算法具有良好的运行效率。

同效率实验一起,也与 bintprog 所得最优值进行比较,结果显示两种方法所得最优值相同,即优度全部为 100%,SBG 算法所得为最优算法。

4.3 不满足需求率改变

在该组实验中,任务数固定为 30,任务的候选服务数固定为 20,不满足需求率由 20%上升至 80%。如图 4 所示。

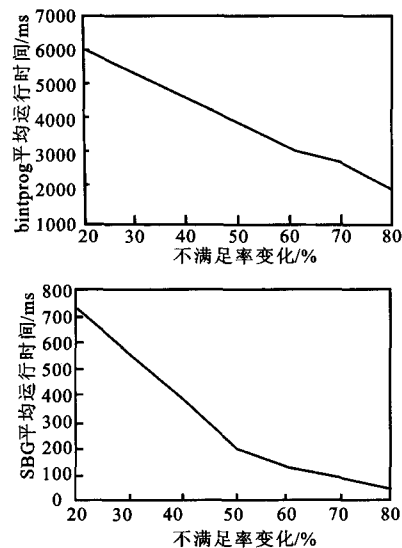


图4 不满足率变化时算法效率比较

该组实验表明,在任务数和候选服务数不变的情形下,随着不满足率增加,满足需求的可行方案减少,bintprog 方法和 SBG 方法所需时间均降低,但 SBG 算法时间降低速度更快,具有更好的运行效率。

同效率实验一起,也与 bintprog 所得最优值进行比较,结果显示两种方法所得最优值相同,即优度全部为 100%,故 SBG 算法所得为最优算法。

5 结束语

针对已有 Web 服务选择优化方法存在效率低下或优度不高的问题,文中提出了一种有效的不必探索整个解空间的 SBG 优化算法,为服务请求者选择出最优组合方案。实验结果表明,该算法提高了问题的求解效率。

参考文献:

- [1] Berners Lee T, Hendler J, Lassila O. The semantic web: a new form of web content that is meaningful to computers will unleash a revolution of new possibilities [J]. Sci Am, 2001, 284(5):34-43.

(下转第 101 页)

其通信复杂度和计算复杂度也都是 $O(1)$ 。

6 结束语

文中提出了一种能够计算两相交圆精确面积的安全协议,利用该协议,参与者能够安全地计算出两个相交圆的面积,并且保证各方的私有数据不会泄漏,分析证实了该协议的安全性和有效性。未来的工作是为其他计算几何问题设计出安全有效的多方计算协议。

参考文献:

- [1] Yao C. Protocols for secure computations[C]// Proc 23rd Annual IEEE Symposium on Foundations of Computer Science. Los Alamitos: IEEE Computer Society Press, 1982:160—164.
- [2] Goldreich. Foundations of cryptography: Volume II, basic applications[D]. Cambridge: Cambridge University Press, 2004.
- [3] Atallah M J, Du W L. Secure multi-party computational geometry[C]// Proc of 7th International Workshop on Algorithms and Data Structures (WADS). Berlin: Springer, 2001:165—179.
- [4] 荆巍巍. 安全多方计算中若干基础协议及应用的研究[D]. 安徽:中国科学技术大学,2008.
- [5] Goethals B, Laur S, Lipmaa H, et al. On private scalar product computation for privacy-preserving data mining[C]// proceedings of the 7th Annual International Conference in Information Security and Cryptology. Seoul, Korea: Springer, 2004:104—120.
- [6] Paillier P. Public key cryptosystems based on composite degree residuosity classes[C]// Proceedings of the 17th international conference on theory and application of cryptographic techniques. Berlin, Heidelberg: Springer-Verlag, 1999:223—238.
- [7] 徐维江. 网络计算中的私有数据保护问题及其应用研究[D]. 安徽:中国科学技术大学,2008.
- [8] Zhu Youwen, Huang Liusheng, et al. Three new approaches to privacy-preserving add to multiply protocol and its application[C]// Second International workshop on Knowledge Discovery and Data Mining, Moscow: IEEE, 2009:554—558.
- [9] Youwen Zhu, Liusheng Huang et al. Relation of PPA-TMP and Scalar Product Protocol and Their Applications [C]// Proceedings of the IEEE Symposium on Computers and Communications. Washington, DC, USA: IEEE Computer Society, 2010:184—189.

作者简介:

赵书峰 男,(1967—),副教授.研究方向为信息安全.
黄刘生 男,(1957—),博士生导师,教授.研究方向为包括信息安全、分布式计算以及无线传感器网络.

(上接第97页)

- [2] Yang FC, Su S, Li Z. Hybrid QoS-aware semantic web service composition strategies[J]. Science in China Series E, Information Science, 2008, 38(10): 1697—1716.
- [3] Zeng L Z, Benatallah B, Ngu A H H, et al. QoS-aware middleware for web services composition [J]. IEEE Trans Software Eng, 2004, 30(5): 311—327.
- [4] Dai Y, Yang L, Zhang B, et al. QoS for composite web services and optimizing [J]. Journal of Computers, 2006, 29(07): 1167—1178.
- [5] Menascé D A, Emiliano Casalicchio, Vinod Dubey, et al. On optimal service selection in Service Oriented Architectures[J]. Performance Evaluation, 2010, 67(8): 659—675.
- [6] Alrifai M, Risse T. Combining global optimization with local selection for efficient QoS-aware service composition. WWW 2009:881—890.
- [7] Schrijver A. Theory of linear and integer programming [J]. John Wiley & Sons, 1986.
- [8] Huang A F M, Lan C W, Yang S J H. An optimal QoS-based Web service selection scheme [J]. Inf Sci, 2009, 179(19): 3309—3322.
- [9] Wolsey L A. Integer Programming, John Wiley [R]. New York: 1998.
- [10] Maros I. Computational Techniques of the Simplex Method[M]. NY: Springer, 2003.
- [11] Huang S. Inverse problems of some NP-complete problems[J]. Algorithmic Applications in Management (AAIM 2005), LNCS 3521:422—426.

作者简介:

王广正 男,(1980—),硕士,讲师.研究方向为语义 Web 服务.