

结合对策论与多目标 MILP 的 Web 服务组合调度问题求解

李 靖¹, 乔 蕊¹, 刘志中²

(1. 周口师范学院计算机科学与技术学院, 河南 周口 466001;

2. 哈尔滨工业大学计算机科学与技术学院, 哈尔滨 150001)

摘 要: 针对单目标优化问题, 结合对策论与多目标混合整数线性规划 (MILP), 提出一种 Web 服务组合 (WSC) 调度方法。根据 5 个基本的服务质量 (QoS) 属性定义 WSC 的 QoS 属性, 将 WSC 调度问题转换为求解 MILP 问题, 服务商之间的交互模拟为一个非合作、不完备的贝叶斯对策, 利用参与者的贝叶斯纳什均衡保证服务商对于组合任务的公平竞争, 从而达到最小化执行时间和成本, 同时最大化声誉、可靠性和可用性的目标。数值分析结果表明, 该方法具有较高的求解效率, 能避免效应函数中的多属性聚合现象, 适用于求解中小规模的 WSC 调度问题。

关键词: 对策论; 混合整数线性规划; Web 服务组合; 服务质量; 多目标优化

中文引用格式: 李 靖, 乔 蕊, 刘志中. 结合对策论与多目标 MILP 的 Web 服务组合调度问题求解 [J]. 计算机工程, 2016, 42(1): 11-17.

英文引用格式: Li Jing, Qiao Rui, Liu Zhizhong. Solution of Web Service Composition Scheduling Problem Combining with Game Theory and Multi-objective MILP [J]. Computer Engineering, 2016, 42(1): 11-17.

Solution of Web Service Composition Scheduling Problem Combining with Game Theory and Multi-objective MILP

LI Jing¹, QIAO Rui¹, LIU Zhizhong²

(1. School of Computer Science and Technology, Zhoukou Normal University, Zhoukou, Henan 466001, China;

2. School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China)

[Abstract] Aiming at single objective optimization problem, a Web Service Combination (WSC) scheduling method based on game theory and multi-objective Mixed Integer Linear Programming (MILP) is proposed. The Quality of Service (QoS) attributes of WSC are defined according to five basic QoS attributes. WSC scheduling problem is converted to the multi-objective optimization problem considering QoS. The interaction between service providers is simulated as a non-cooperative and incomplete Bayesian countermeasure. The competition strategy is guaranteed by the Bayesian Nash equilibrium of participants when service provider executes tasks. Thus, the execution time and execution cost is minimized, the reputation, reliability and availability are maximized at the same time. The numerical analysis result shows that the proposed method has a high solving efficiency, can avoid the aggregation problem of multiple attributes in utility function, and it is suitable for solving the small and medium size WSC scheduling problems.

[Key words] game theory; Mixed Integer Linear Programming (MILP); Web Service Composition (WSC); Quality of Service (QoS); multi-objective optimization

DOI: 10.3969/j.issn.1000-3428.2016.01.003

1 概述

Web 服务调度^[1]对业务全局处理性能具有重要影响, 通常需要考虑用户约束和参数设置, 从而保证客户满意度。目前, Web 服务组合 (Web Service

Composition, WSC) 调度已成为提高客户满意度的关键技术, 涉及大量服务选择、移动以及不同服务之间协同调度等动态行为^[2], 然而 WSC 中存在的多 QoS 属性聚合问题仍没有得到较好地解决。

本文提出一种基于对策论和多目标混合整数线

基金项目: 国家自然科学基金青年基金资助项目 (61300124); 河南省科技厅软科学研究计划基金资助项目 (132400411365); 河南省高等学校重点科研基金资助项目 (15B520046)。

作者简介: 李 靖 (1982 -), 女, 讲师、硕士, 主研方向为服务计算、Web 服务调度; 乔 蕊, 讲师、硕士; 刘志中, 讲师、博士后。

收稿日期: 2015-07-27 **修回日期:** 2015-08-23 **E-mail:** lijinghnzk@126.com

性优化的调度方法,对当前最优的 Web 服务组合技术进行改进,主要工作为:(1)将 Web 服务调度问题转变为求解多目标混合整数线性优化问题^[3],不仅将多目标(最小化执行时间和成本,同时最大化声誉、可信度和可用性)聚合为假设的效应函数,且提出一种目标规划方案。(2)将服务商之间的交互模拟为一个非合作、不完备的贝叶斯对策^[4]。

2 相关研究

服务组合是按照一定的业务逻辑对现有服务进行集成以解决 Web 目标调度问题。文献[5]在总结各种 WSC 方法的基础上,将其归纳为语义型和过程型 WSC 方法。语义型 WSC 为 Web 服务描述和请求描述添加能被计算机系统理解的语义,通过推理自动生成 WSC。过程型 WSC 使用流程定义工具和语言对 WSC 业务过程进行建模,在具体流程执行前根据服务质量(Quality of Service, QoS)属性选择具体服务来替代流程模型中的抽象服务,从而获得可执行的组合服务方案,本文主要研究过程型 WSC 方法。针对过程型 WSC 的调度,学者们提出了许多方法。例如,文献[6]提出一种针对 Web 服务调度问题的数学计算框架,将服务组合问题模拟为一个多目标混合整数线性规划问题,并通过整数规划解决该问题,然而服务组合问题采用线性或者非线性数学模型。因此,基于精确方法的解决方案不适用于带有大量 Web 服务组合的任务。

针对 Web 服务组合问题,研究人员提出了一系列元启发式算法,如遗传算法、模拟退火算法、禁忌搜索算法、贪婪随机自适应搜索过程(Greedy Randomized Adaptive Search Procedure, GRASP)算法和路径重连^[7]算法等。文献[8]将粒子群优化应用于多目标服务组合问题的近似帕累托最优规划,文献[9]利用 GRASP 解决由服务水平协议违法引起的最小化总成本问题,文献[10]结合 GRASP 和路径链接,解决了运行时的组合 Web 服务问题。文献[11]从业务流程出发,提出基于对策论的资源分配策略,改进了 Web 服务商与对策论之间的竞争关系。文献[12]通过联合 Web 服务商和顾客的形成联盟,提出一种基于核的解决方案,用于管理市场。文献[13]描述了一个在并行任务及其组合服务性能之间存在冲突的数学模型,一定程度上解决了单个目标的优化问题,然而该模型中存在多目标 QoS 属性聚合问题。文献[14]提出了一种基于目标递进的 Web 服务组合方法,采用前向链的搜索策略,建立基于条件的有向无环图,可有效利用输入、输出参数和前提、后置条件确定服务组合顺序,一定程度上解决了服务组合存在的问题,然而,该方法的执行效率仍

需进一步提高。文献[15]提出基于 QoS 关联的 Web 服务组合决策算法,考虑服务类在 QoS 属性上的关联关系,对 QoS 数据进行统一规格化和综合评估,提高了执行效率。文献[16]利用全局方法模拟并优化多目标 QoS 属性,采用效应函数聚集多目标,从而设计帕累托最优执行计划集。但在传统的 Web 调度方法中,多 QoS 属性聚合问题并未得到很好地解决。因此,本文结合对策论与多目标优化调度模型,提出一种 WSC 调度方法。

3 QoS 属性

假设将 Web 服务视为旅行规划系统,包含许多组合服务,如预订机票、旅游保险、宾馆预订、汽车或自行车租赁和路线规划等。在这些服务中,Web 服务可有序、并行执行。图 1 利用简单流程图说明了带有 Web 服务的旅行规划系统,其中,Web 服务 1 与 Web 服务 2 并行执行。用户将目的地、出发日期和返回日期输入到旅行规划系统中,系统根据这些信息确定可选择的航班和宾馆房间。

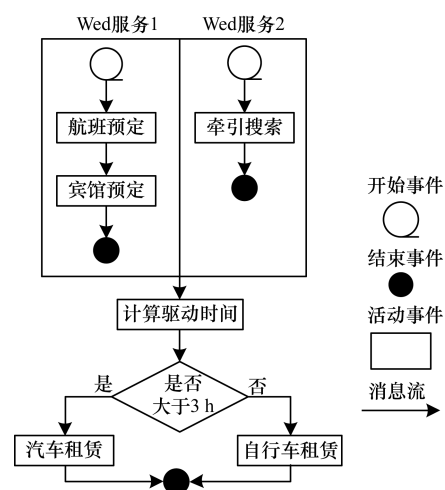


图 1 带有 Web 服务的旅行规划系统

由于业务表示法可以包括 IF-THEN-ELSE 结构,因此可能生成 2 个不同执行频率的任务图。一个执行路径包括并行活动,但是在 THEN 和 ELSE 分支机构中没有活动。

本文转换规则如下:(1)若原业务流程建模标记(Business Process Modeling Notation, BPMN)图包含网关决定,则执行预处理程序,即利用 Java 程序将 BPMN 图表转化为一个任务图,生成 2 个分支: THEN 分支和 ELSE 分支。(2)利用水平并行路径同步执行任务。(3)活动、开始和结束元素在任务图中表示节点,将图元素转化为一个数学优化模型。BPMN 中不涉及执行频率,网关产生 THEN 和 ELSE 元素时,生成的 0~1 之间的实数与该元素执行的相对频率有关,在实际实现中可从记录数据中获得执

行频率。关于基本服务, 文献[17]提出基于 QoS 属性的 5 个目标, 将其应用于组合服务中, 最终的优化目标为: 最小化执行时间和执行成本, 同时最大化声誉、可靠性和可用性。

3.1 基本 QoS 属性

文献[17]提出了基本 QoS 属性的 5 个目标, 具体描述如下:

(1) 执行成本

EC_j^i 表示执行成本, 即服务商 j 提供给服务用户 i 所获得的执行收入。

(2) 执行时间

ED_j^i 表示执行时间, 即衡量 Web 服务 i 执行请求与服务商 j 响应之间的时间延迟, 利用表达式 $ED_j^i = T_{\text{process}} + T_{\text{trans}}$ 计算执行时间, 其中, T_{process} 表示 Web 服务商提供的处理时间; 传输时间 T_{trans} 由先前执行的 Web 服务计算得到:

$$T_{\text{trans}} = \frac{\sum_{k=1}^{N_e} T_k}{N_e} \quad (1)$$

其中, T_k 是传输时间的第 k 次观测值; N_e 是观测总次数。

(3) 声誉

RP_j^i 表示声誉, 是用户满意度的度量, 且依赖于 Web 服务的先验知识。不同用户对同一服务有不同的满意度, 因此, 将声誉定义为 Web 服务排名的平均值:

$$RP_j^i = \frac{\sum_{k=1}^{N_r} G_k}{N_i} \quad (2)$$

其中, G_k 是用户 k 的服务排名; N_i 为 Web 服务的评估次数。

(4) 可靠性

R_j^i 表示可靠性, 即请求在期望时间内得到响应的概率。可靠性与软硬件结构有关, 依赖 Web 服务器及其与客户连接的质量。利用表达式 $R_j^i = N_e / K$ 计算可靠性, 其中, N_e 为期望时间内正确响应请求的 Web 数量; K 为理论评估次数。

(5) 可用性

AV_j^i 表示可用性, 即来自服务商 j 的一个 Web 服务 i 的可用性, 利用表达式 $AV_j^i = T_a / \alpha$ 计算可用性, 其中, T_a 为在最后 α s 内 Web 服务 i 可用的整体时间; α 通过 Web 服务管理员设置, 依赖应用程序进行变化。若 Web 服务的应用程序被频繁访问, 则需设置较小的 α 值; 若服务很少被访问, 则需要设置较大的 α 值。

3.2 组合 QoS 属性

业务流程图可以有多个执行路径, 其依赖于中间网关或并行处理。为了量化业务流程, 给出以下

建模元素: 执行路径 ep_l 包括并行任务序列 (开始和结束 2 个专用任务); l 是所有执行路径集; $freq_l$ 是执行路径 ep_l 的执行频率; 子路径 sp_m^l 不包括任何并行序列, 因为它是执行路径的子集; 执行路径 ep_l 的执行计划 epl^l 是一个有序对 (i, j) 的集合, 其中, i 表示 ep_l 中的任务; j 表示 Web 服务商。

(1) 执行时间

执行计划 epl^l 的执行时间是每个子路径 sp_m^l 的所有执行时间的总和, 子路径 sp_m^l 属于执行路径 ep_l , 执行时间为 $ED_l = \max_{sp_m^l \in ep_l} \sum_{i \in sp_m^l, (k, j) \in epl^l, k=i} ED_j^i$ 。

(2) 执行成本

在执行路径 ep_l 中, 执行计划 epl^l 的执行成本是该计划执行成本 EC_j^i 的总和, 表示为 $EC_l = \sum_{(i, j) \in epl^l} EC_j^i$ 。

(3) 声誉

在执行路径 ep_l 中, 执行计划 epl^l 的声誉是路径 ep_l 的个体声誉 RP_j^i 的权重平均和, 表示为 $RP_l = \frac{1}{N_s} \sum_{(i, j) \in epl^l} RP_j^i$, 其中, N_s 为样本元素数量。

(4) 可靠性

在执行路径 ep_l 中, 执行计划 epl^l 的可靠性表示为 $R_l = \prod_{(i, j) \in epl^l} R_j^i$ 。

(5) 可用性

在执行路径 ep_l 中, 执行计划 epl^l 的可用性为 $AV_l = \prod_{(i, j) \in epl^l} AV_j^i$ 。

4 Web 服务组合调度

4.1 多目标优化

将 Web 服务调度问题转换为多目标优化问题, 包括多目标元素和整数规划。二进制变量 y_j^i 表示服务商 j 是否执行 Web 服务 i ; 当 $y_j^i = 1$ 时, 表示执行; 当 $y_j^i = 0$ 时, 表示不执行。

4.1.1 执行时间和执行成本约束

设 A 为一个给定业务处理图的所有任务集, 假设存在 Web 服务商 WS 集, 可将其分配给任何任务 $i, i \in A$ 。对于每个任务 i , 仅分配一个服务商, Web 服务商必须遵守以下约束:

$$\sum_{j \in WS} y_j^i = 1, \forall i \in A \quad (3)$$

定义 ST_{il} 为执行路径 ep_l 中任务 i 的最早开始时间; TD_{il} 为执行路径 ep_l 中任务 i 的执行时间; $j \rightarrow k$ 为任务 k 继承任务 j ; MXT 为完成任何任务的最大时间。条件约束 (4) 表示: 在任何执行路径 ep_l 中, 任何任务 i 的执行时间等同于 Web 服务商执行时间的总和。约束条件 (5) 表示: 在任何执行路径 ep_l 中, 任务 k 在统一执行路径中紧接着任务 j , 任务 j 完成

后才执行任务 k 。约束条件(6)表示:每条执行路径的执行时间等于所有子路径 sp_m^i 执行时间的最大值。约束条件(7)表示执行路径的总执行时间不超过 MXT 。

$$\sum_{j \in WS} ED_j^i y_j^i = TD_{il}, \forall i \in A, \forall ep_l \in L \quad (4)$$

$$(TD_{jl} + ST_{jl}) \leq ST_{kl}, \forall j \rightarrow k, j, k \in A, \forall ep_l \in L \quad (5)$$

$$ED_i \geq \sum_{i \in sp_m^i} TD_{il}, \forall ep_l \in L \quad (6)$$

$$ED_i \leq MXT, \forall ep_l \in L \quad (7)$$

定义 MXC 为实施任何执行计划的可用预算之和, EC_i 为执行路径 ep_l 的执行成本。约束条件(8)表示:执行路径总成本为执行该路径中所有任务的执行成本之和。约束条件(9)表示:任何路径的总执行成本不超过 MXC 。

$$\sum_{i \in ep_l, j \in WS} EC_j^i y_j^i = EC_l, \forall ep_l \in L \quad (8)$$

$$EC_l \leq MXC, \forall ep_l \in L \quad (9)$$

4.1.2 声誉约束

定义 $MNRP$ 为最小期望, $|ep_l|$ 为 ep_l 任务数量, 执行路径 ep_l 的声誉 RP_l 存在如下约束条件:

$$RP_l = \sum_{i \in ep_l, j \in WS} RP_j^i y_j^i / |ep_l|, \forall ep_l \in L \quad (10)$$

$$RP_l \geq MNRP, \forall ep_l \in L \quad (11)$$

4.1.3 可用性和可靠性约束

定义 MNA 为最小期望可用性, AV_l 为执行路径 ep_l 的可靠性。可用性和可靠性的聚合函数为非线性, 其线性化通过 Neperian 对数实现。可用性 MNA 和可靠性 ep_l 存在如下约束条件:

$$AV_l \geq \ln(MNA), \forall ep_l \in L \quad (12)$$

$$AV_l = \sum_{i \in ep_l, j \in WS} \ln(AV_j^i) y_j^i, \forall ep_l \in L \quad (13)$$

$$R_l \geq \ln(MNRL), \forall ep_l \in L \quad (14)$$

$$R_l = \sum_{i \in ep_l, j \in WS} \ln(R_j^i) y_j^i, \forall ep_l \in L \quad (15)$$

Web 服务组合调度问题的性能目标为:

$$\min f_1 = \sum_{ep_l \in L} freq_l EC_l \quad (16)$$

$$\min f_2 = \sum_{ep_l \in L} freq_l ED_l \quad (17)$$

$$\min f_3 = \sum_{ep_l \in L} freq_l RP_l \quad (18)$$

$$\min f_4 = \sum_{ep_l \in L} freq_l AV_l \quad (19)$$

$$\min f_5 = \sum_{ep_l \in L} freq_l R_l \quad (20)$$

其中, $freq_l$ 表示执行路径 ep_l 的频率。

通过目标达到法^[12]解决先验分析中出现的多目标优化问题, 即给定一个目标函数集 f_1, f_2, \dots, f_m ,

在可行区域 X 内进行最小化, 转换为以下单目标优化问题:

$$\begin{aligned} & \text{minimum } \sigma, \text{ submit to } f_i(x) - \sigma w_i \leq \bar{f}_i \\ & i = 1, 2, \dots, m, x \in X \end{aligned} \quad (21)$$

其中, f_i 为对象目标; w_i 为与目标有关的权重; σ 表示目标值。普遍假设 $f_i = f_{i, \min}, i = 1, 2, \dots, m$, 可行域中每个目标等价于所有单个目标函数的最小值。假设 $w_j = f_j, j = 1, 2, \dots, m$, 通过求解式(21), 最小化所有的目标值。因为式(16)~式(20)的目标 $i = 3, 4, 5$ 必须最大化, 所有(21)改写为:

$$\begin{aligned} & \text{minimum } \sigma, \text{ submit to } f_i(x) - \sigma w_i \leq \bar{f}_i, i = 1, 2 \\ & f_i(x) + \sigma w_i \geq \bar{f}_i, i = 3, 4, 5, x \in X \end{aligned} \quad (22)$$

其中, 在最小化后的所有属性目标值中, 最大值使用 σ^\vee 表示; $\bar{f}_i = f_{i, \max}, i = 3, 4, 5$ 表示响应目标函数在可行域 X 的最大值。理想值 (\bar{f}_i, \bar{f}_i) 提供组合 Web 服务的性能范围, 权重向量决定了发展方向: 权重 w_i 越小, 目标函数 f_i 越重要。若 $\sigma^\vee > 0$, 表明至少有一个目标值不满足不等式(22), 若 $\sigma^\vee \leq 0$, 表明所有目标值都满足不等式(22)。

在经典混合整数线性规划中, Web 服务组合问题等价于一个多维多选择背包问题 (Multi-dimensional Multi-choice Knapsack Problem, MMKP)^[13]。由于 MMKP 的每个实例可作为一个典型的 Web 服务组合问题, 如 NP 困难问题, 因此目标规划将式(22)构建为一个带有额外限制条件的 MMKP。式(22)属于 NP 完全问题, 本文利用对策论解决该问题。

4.2 对策论

将 Web 服务调度问题描述为一个非合作、不完备贝叶斯对策。该对策的结构对所有服务商都公开, 但每个服务商仅知道自己的收益函数。通过参与者的贝叶斯纳什均衡确保服务商执行 WSC 任务的竞争策略。

假设每个服务商 j 有多个动作, 以执行时间的 $|A|$ 维向量形式表示, 业务流程中每个任务具有相同的执行时间, C_j 为服务商 j 的有限动作集合, 一个典型动作 $ED_j \in C_j$ 表示为:

$$\begin{aligned} ED_j &= (ED_j^1, ED_j^2, \dots, ED_j^{|A|}) \\ ED_j^i &> 0, i = 1, 2, \dots, |A| \end{aligned} \quad (23)$$

服务商 j 对任务 i 提供的执行时间 ED_j^i 和执行成本 EC_j^i 的间接约束为:

$$\begin{cases} T_{\min ij} \leq ED_j^i \leq T_{\max ij} \\ P_{\min ij} \leq EC_j^i \leq P_{\max ij} \end{cases}$$

其中, $T_{\min ij}, T_{\max ij}$ 分别表示服务商 j 对任务 i 的最小执行时间和最大执行时间; $P_{\min ij}, P_{\max ij}$ 分别表示执行任务 i 需要付给服务商 j 的最低执行成本和最高执

行成本。

将服务商 j 执行任务 i 时的收益设为一个仿射变换, ED_j^i 的单调下降函数为:

$$EC_j^i = -\frac{P_{\max ij} - P_{\min ij}}{T_{\max ij} - T_{\min ij}} ED_j^i + P_{\min ij} \quad (24)$$

式(24)表明用户响应请求的时间越短, 需要支付的费用越多。包含 n 个服务商的策略表示为 $n|A|$ 维向量 $ED = (ED_1, ED_2, \dots, ED_n)$ 。给定任何策略文件, 调度问题的优化解决方案是确保将任务分配给服务商, $U_j(ED)$ 表示每个服务商 j 的效应函数:

$$U_j(ED) = \sum_{i \in A} EC_j^i y_j^i \quad (25)$$

为了给服务商的策略添加不确定元素, 为每个服务商 j 关联类型, 每个类型为不同参数 $P_{\min ij}$ 和 $P_{\max ij}$ 的实例描述。 θ_j 为通用类型, Θ_j 为服务商 j 类型的有限集合。每个服务商知道自己的类型, 即效应函数式(25), 但不知道其他服务商的关联类型。设联合概率分布类型为 $p(\theta_1, \theta_2, \dots, \theta_m)$, $\theta_1 \in \Theta_1, \theta_2 \in \Theta_2, \dots, \theta_n \in \Theta_n$, 所有服务商都知道该联合概率分布类型。

对一个不完备信息或贝叶斯策略, 假设关联类型是服务商提供的服务, 所以, 纯策略 $s_j: \Theta_j \rightarrow C_j$ 是从 Θ_j (类型) 到 C_j (活动) 的一个映射, 服务商 j 的所有纯策略集合使用 S_j 表示。

服务商 j 的混合策略 σ_j 是任何纯策略 s_j 自己本身集合的概率分布, 即:

$$\sum_{s_j \in S_j} \sigma(s_j) = 1, \sigma(s_j) \geq 0, \forall s_j \in S_j \quad (26)$$

设 Σ_j 为服务商 j 的所有混合策略有限集合, 使用 $\sigma_j(ED_j | \theta_j)$ 表示在混合策略 σ_j 下服务商 j 在类型为 θ_j 时存在活动 ED_j 的概率。

在组合策略 $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_n)$ 下, 服务商 j 的先验期望效应函数给定如下:

$$EU_j(\sigma) = \sum_{\theta \in \Theta} p(\theta) \sum_{ED \in S} \left(\prod_j \sigma_j(ED_j | \theta_j) \right) U_j(ED, \theta) \quad (27)$$

其中, $\theta = (\theta_1, \theta_2, \dots, \theta_n)$; $\Theta = \Theta_1 \times \Theta_2 \times \dots \times \Theta_n$; $S = S_1 \times S_2 \times \dots \times S_n$ 。

对于一个删除的组合策略 σ_{-j} , 服务商 j 的最优响应集合定义为:

$$BR_j(\sigma_{-j}) = \arg \max_{\sigma_j \in \Sigma_j} EU_j(\sigma_j, \sigma_{-j}) \quad (28)$$

服务商提供策略的贝叶斯纳什均衡是一个满足 $\sigma_j \in BR_j(\sigma_{-j})$ 的组合策略。根据贝叶斯纳什均衡定义, 每个服务商对其他服务商策略有一个最优响应, 贝叶斯纳什均衡总是存在于 Web 服务调度问题的对策论规划中。

5 实验结果与分析

本文利用数值实例分析算法性能, 许多参数(如执行成本和执行时间)需要通过同一 Java 程序随机生成, 利用 Java 程序将一个业务问题的执行时间转换成 GNU 线性编程工具(GNU Linear Programming Kit, GLPK)需要的数据格式^[5]。由于必须符合业务流程, 因此执行路径不能随机生成。所有测试均在配有双核 3 GHz, Windows XP 系统的 PC 机上实现。

图 2 为数值实例的业务流程, 共有 6 个任务 ($T_1 \sim T_6$)、3 条执行路径, 这些任务由 5 个服务商完成。随机生成个体服务属性: 执行时间为 [0.5 s, 8 s], 执行成本为 [1 元, 100 元], 可用性为 [0.95, 0.999 99], 声誉为 [0.8, 0.99], 可靠性为 [0.85, 0.95]。在业务流程中, Web 服务执行时间中也包括用户从等待直到开始服务的时间。

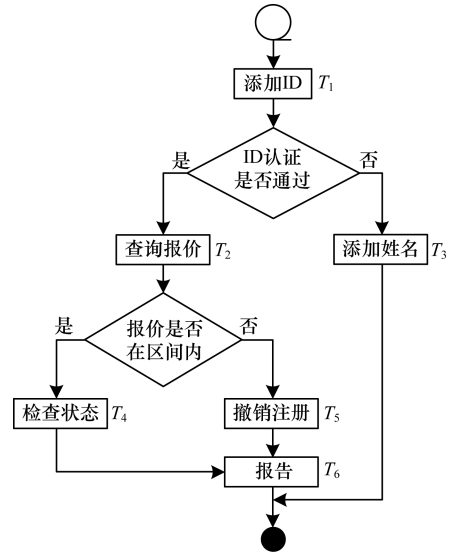


图 2 数值实例的业务流程

5.1 多目标线性优化分析

本文研究了调度问题的产生原因, 并给出了最优化 Web 服务组合调度问题的求解方案。表 1 为全局目标优化结果, 其中, 最大偏差表示优化后的 QoS 属性值相比目标 QoS 属性值的偏差, 以执行成本为例, $|356 - 266| / 266 \times 100\% = 33.83\%$ 。

表 1 全局目标优化结果

QoS 属性	属性值		最大偏差/%
	优化后的 QoS	目标 QoS	
执行成本/元	356	266	33.83
执行时间/s	30.36	23.74	27.89
声誉	0.895 6	0.920 8	2.70
可用性	0.794 1	0.816 9	2.79
可靠性	0.363 8	0.375 4	3.09

表2所示为每条执行路径的组合属性值,共有3条执行路径: $T_1-T_2-T_4-T_6$, $T_1-T_2-T_5-T_6$, T_1-T_3 。对式(16)~式(20)中的每个单目标进行初始化,得到理想的执行时间、执行成本、声誉、可用性和可靠性,这些信息确保了Web服务组合的性能范围。表3为在不同Web服务数(10,25,50)和任务数(25,5)下的测试结果。仿真参数与图2所示的数值实例中的参数相同,而每个Web服务的执行时间在0.01~10之间单调变化。包含25个任务的业务流程有3条执行路径,其中的1条路径有2条子路径,另外2条路径分别包含12条子路径。包含5个任务的业务流程有2条执行路径,每条路径都没有子路径。

从表3可以看出, σ^v 表明包含25个任务的业务流程的瓶颈项为执行成本,包含5个任务时的瓶颈项为执行时间。总体而言,在Web服务数与任务数的每个组合上,执行时间都在2s以内,在实际应用中是可以接受的。

表2 全局执行路径的组合属性值

QoS属性	$T_1-T_2-T_4-T_6$	$T_1-T_2-T_5-T_6$	T_1-T_3
执行成本/元	127	349	322
执行时间/s	11.00	29.60	30.58
声誉	0.95	0.89	0.92
可用性	0.90	0.75	0.80
可靠性	0.54	0.27	0.28

表3 不同Web服务数和任务数下的测试结果

Web服务数	任务数	执行成本/元	总执行时间/s	声誉	可用性	可靠性	σ^v
10	25	0.942 0	0.545 9	0.052 8	0.046 0	0.012 4	0.942 0
25	25	1.241 0	1.125 9	0.075 5	0.122 0	0.028 6	1.241 0
50	25	1.069 7	0.917 2	0.103 8	0.145 1	0.323 9	1.069 7
10	5	1.395 8	1.569 2	0.093 1	0.064 2	0.147 9	1.569 2
25	5	0.947 9	1.386 0	0.063 8	0.083 1	0.052 5	1.386 0

5.2 针对对策论的数值分析

针对对策论的数值分析,考虑图2描述的仅有2个服务商的业务流程,通过表4定义6个任务的执行时间区间 $[T_{min_{ij}}, T_{max_{ij}}]$ 。

表4 6个任务的执行时间区间 s

任务	服务商1	服务商2
1	[15,20]	[14,21]
2	[14,19]	[13,18]
3	[15,16]	[16,18]
4	[16,18]	[10,11]
5	[6,8]	[1,5]
6	[2,5]	[3,4]

表5为服务商类型的价格区间 $[P_{min_{ij}}, P_{max_{ij}}]$,每个服务商的第1个类型是 θ_j^1 ,比第2个类型 θ_j^2 价格低。设服务商类型的联合概率分布为:

$$\begin{aligned}
 p(\theta_1^1, \theta_2^1) &= 0.3, p(\theta_1^1, \theta_2^2) = 0.1 \\
 p(\theta_1^2, \theta_2^1) &= 0.2, p(\theta_1^2, \theta_2^2) = 0.4 \quad (29)
 \end{aligned}$$

表6表示包含2个服务商的5个动作的组合QoS属性值,表示为 $ED_j^1, ED_j^2, ED_j^3, ED_j^4, ED_j^5$ 。

表5 6个任务的价格区间 元

任务	服务商类型			
	θ_1^1	θ_1^2	θ_2^1	θ_2^2
1	[1.50,2.50]	[15.0,25.0]	[1.30,1.50]	[13.0,15.0]
2	[0.20,2.00]	[2.00,2.00]	[1.70,1.80]	[17.0,18.0]
3	[0.85,1.25]	[8.5,12.5]	[0.75,1.22]	[7.50,12.2]
4	[0.02,1.00]	[2.00,10.0]	[0.25,0.75]	[2.50,7.50]
5	[0.33,0.66]	[3.30,6.60]	[0.05,0.25]	[0.50,2.50]
6	[0.50,1.50]	[5.00,15.0]	[0.51,1.51]	[5.10,15.1]

表6 服务商1和服务商2的组合QoS属性值

单一调度策略	组合QoS属性值
ED_1^1	(20.0,19.0,16.0,18.0,8.0,2.0)
ED_1^2	(17.5,19.0,15.5,18.0,8.0,2.0)
ED_1^3	(20.0,16.5,15.5,18.0,8.0,5.0)
ED_1^4	(15.0,16.5,15.5,18.0,6.0,5.0)
ED_1^5	(15.0,14.0,15.0,16.0,6.0,2.0)
ED_2^1	(14.0,18.0,16.0,11.0,1.0,4.0)
ED_2^2	(14.0,18.0,16.0,11.0,1.0,4.0)
ED_2^3	(21.0,15.5,16.0,11.0,1.0,4.0)
ED_2^4	(17.5,15.5,17.0,10.0,3.0,4.0)
ED_2^5	(21.0,18.0,17.0,10.5,5.0,4.0)

服务商j的单一调度策略标记为 $ED_j^{\alpha_j}ED_j^{\beta_j}$ ($\alpha_j, \beta_j = 1, 2, 3, 4, 5$),表示服务商j类型分别是 θ_j^1, θ_j^2 时,策略为 $ED_j^{\alpha_j}ED_j^{\beta_j}$ 。实例中包含2个服务商,每个服务商有5个策略,因此,共有 $C_5^2 = 25$ 种组合策略。给定类型p的先验联合概率(见式(29)),在策略 $(ED_1^{\alpha_1}ED_1^{\beta_1}, ED_2^{\alpha_2}ED_2^{\beta_2})$ 下服务商j的期望效用函数通过式(30)给定:

$$\begin{aligned}
 u_j(ED_1^{\alpha_1}ED_1^{\beta_1}, ED_2^{\alpha_2}ED_2^{\beta_2}) &= p(\theta_1^1, \theta_2^1) U_j(ED_1^{\alpha_1}, ED_2^{\beta_2}, \theta_1^1, \theta_2^1) \\
 &+ p(\theta_1^1, \theta_2^2) U_j(ED_1^{\alpha_1}, ED_2^{\beta_2}, \theta_1^1, \theta_2^2) \\
 &+ p(\theta_1^2, \theta_2^1) U_j(ED_1^{\alpha_1}, ED_2^{\beta_2}, \theta_1^2, \theta_2^1) \\
 &+ p(\theta_1^2, \theta_2^2) U_j(ED_1^{\alpha_1}, ED_2^{\beta_2}, \theta_1^2, \theta_2^2) \quad (30)
 \end{aligned}$$

表7为 $U_1(ED_1, \theta_1^1), U_2(ED_2, \theta_2^1)$ 的数值表示,例如, $U_1(ED_1^1, \theta_1^1), U_2(ED_2^1, \theta_2^1)$ 表示为0.43,1.91。

表 7 效应函数 $U_1(ED_1, \theta_1^i), U_2(ED_2, \theta_2^i)$ 的数值表示

策略	ED_1^1	ED_2^1	ED_3^1	ED_4^1	ED_5^1
ED_1^1	0.43, 1.91	1.93, 0.73	0.43, 1.71	0.43, 1.78	0.43, 1.65
ED_2^1	0.50, 1.91	0.14, 2.58	0.50, 1.71	0.14, 2.12	0.14, 1.99
ED_3^1	2.91, 0.07	2.55, 0.42	1.41, 1.37	1.05, 1.79	1.05, 1.66
ED_4^1	1.41, 1.57	1.05, 1.92	1.41, 1.37	1.05, 1.79	1.05, 1.66
ED_5^1	0.00, 3.53	0.00, 3.86	0.00, 3.36	0.00, 3.52	0.00, 3.27

贝叶斯对策包含 25×25 个正则对策的贝叶斯纳什均衡, 利用 <http://www.gametheorysociety.org/resources.html> 网站获取组合策略的 5 个纳什均衡。表 8 为服务商 1 的均衡组合策略, 括号中的数字表示实施该策略的概率。

表 8 纳什均衡的服务商组合调度策略

纳什均衡	服务商 1 的组合调度策略
1	$ED_1^3 ED_3^1$ (0.22) 和 $ED_1^4 ED_1^3$ (0.78)
2	$ED_1^4 ED_3^1$ (0.78) 和 $ED_1^3 ED_1^4$ (0.22)
3	$ED_1^3 ED_1^3$ (0.22) 和 $ED_1^4 ED_1^4$ (0.78)
4	$ED_1^3 ED_1^3$ (0.09), ED_1^4 (0.13) 和 $ED_1^4 ED_1^4$ (0.78)
5	$ED_1^4 ED_1^3$ (0.09), $ED_1^3 ED_1^4$ (0.22) 和 $ED_1^4 ED_1^4$ (0.69)

在 5 个贝叶斯纳什均衡中, 服务商 2 的组合策略是 $ED_2^2 ED_2^4$, 概率为 0.24, 且 $ED_1^4 ED_2^4$ 概率为 0.76。在所有均衡中, 服务商 1 和服务商 2 的期望效应值分别为 5.16 和 8.40。至此, 贝叶斯纳什均衡问题得以解决。服务商 1 能实施任何平衡策略, 服务商 2 能实施单一策略。本文方法符合 Web 服务竞争行为关系, 但对可能的动作和类型进行策略组合的计算量较大。贝叶斯策略主要为: Web 服务商提供服务价格, 拍卖者通过合适的机制决定赢家, 实现组合竞争机制, 可应用于实际的 Web 服务调度中。

6 结束语

本文研究了 Web 服务环境中与业务流程有关的调度问题, 提出对策论结合多目标混合整数线性规划的优化方法。相比现有的 Web 服务组合方法, 本文方法有 2 个创新点: (1) 针对 QoS 属性的 Web 服务组合问题, 对每个单目标进行优化, 同时避免了假设效应函数中的多属性聚合问题。(2) 通过使用贝叶斯模拟服务商间的竞争关系, 有效计算策略的贝叶斯纳什均衡, 保证了服务商竞争组合服务任务的公平性。分析结果表明, 本文方法能有效求解基于非合作与合作策略的 Web 服务组合调度问题, 但其仅将基于竞争的机制与 Web 服务组合数学优化模型进行结合, 下一步将在此基础上模拟 Web 服务商的竞争关系, 进一步提高 WSC 求解效率。

参考文献

- [1] 黄龙涛, 邓水光, 戴康, 等. 基于 MapReduce 的并行 Web 服务自动组合 [J]. 电子学报, 2012, 40(7): 1397-1403.
- [2] 常亮, 刘进, 古天龙, 等. 基于动态描述逻辑的语义 Web 服务组合 [J]. 计算机学报, 2013, 36(12): 2468-2478.
- [3] Marler R T, Arora J S. Survey of Multi-objective Optimization Methods for Engineering [J]. Structural & Multidisciplinary Optimization, 2004, 26(6): 369-395.
- [4] 范小芹, 蒋昌俊, 方贤文, 等. 基于离散微粒群算法的动态 Web 服务选择 [J]. 计算机研究与发展, 2010, 29(1): 147-156.
- [5] 倪晚成, 刘连臣, 吴澄. Web 服务组合方法综述 [J]. 计算机工程, 2008, 34(4): 85-87.
- [6] 李金忠, 夏洁武, 唐卫东, 等. 基于 QoS 的 Web 服务选择算法综述 [J]. 计算机应用研究, 2010, 30(10): 3622-3627.
- [7] Gendreau M, Potvin J Y. Handbook of Metaheuristics [M]. Berlin, German: Springer, 2010.
- [8] Wang Lijuan, Shen Jun, Luo Junzhou, et al. A Survey on Bioinspired Algorithms for Web Service Composition [C] // Proceedings of the 16th International Conference on Computer Supported Cooperative Work in Design. Washington D. C., USA: IEEE Press, 2012: 569-574.
- [9] Matos R S, Maciel P R M, Silva R M A. QoS-driven Optimization of Composite Web Services: An Approach Based on GRASP and Analytical Models [J]. International Journal of Web & Grid Services, 2013, 9(3): 304-321.
- [10] Parejo J A, Segura S, Fernandez P, et al. QoS-aware Web Services Composition Using GRASP with Path Relinking [J]. Expert Systems with Applications, 2014, 41(9): 4211-4223.
- [11] 董军, 詹聪明, 舒静. 网络路由基于对策论的资源分配策略 [J]. 电子学报, 2010, 38(2): 143-147.
- [12] 陈国彬, 张广泉. 基于线性规划 QoS 感知的 Web 服务组合模型 [J]. 控制工程, 2013, 20(6): 1131-1134.
- [13] LI Haifeng. QoS-aware Web Services Composition: A Cooperate Optimization Approach [J]. Key Engineering Materials, 2011, 46(7): 69-74.
- [14] 王成良, 冯欣. 基于目标递进的 Web 服务组合方法 [J]. 计算机工程, 2011, 37(6): 52-54.
- [15] Zheng Huiyuan, Zhao Weiliang, Yang Jian, et al. QoS Analysis for Web Service Compositions with Complex Structures [J]. IEEE Transactions on Services Computing, 2013, 6(3): 373-386.
- [16] 张莉, 张斌, 黄利萍, 等. 基于服务调用特征模式的个性化 Web 服务 QoS 预测方法 [J]. 计算机研究与发展, 2013, 50(5): 1066-1075.
- [17] Zeng Liangzhao, Benatallah B, Ngu A H H, et al. QoS-aware Middleware for Web Services Composition [J]. IEEE Transactions on Software Engineering, 2004, 30(5): 311-327.

编辑 陆燕菲