

# 融合遗传蚁群算法的 Web 服务组合研究<sup>①</sup>

曹腾飞, 符云清, 钟明洋

(重庆大学 软件学院, 重庆 400044)

**摘要:** 为了提高 Web 服务组合流程中服务选择技术的收敛性能, 提出了一种基于遗传算法与蚁群算法相融合的多目标优化策略, 用于解决基于 QoS 的 Web 服务组合问题。本文首先将 Web 服务组合的全局最优化问题转化为寻求一条 QoS 最优解的路径问题, 并通过改进遗传算法得到蚁群算法中初始路径的信息素分布, 再通过改进蚁群算法来求得最优解。仿真实验结果表明, 该改进算法能在较少的进化代数下得到最优路径, 提高了 Web 服务组合的快速全局搜索能力。

**关键词:** Web 服务组合; 蚁群算法; 遗传算法; QoS; 全局最优

## Based Web Service Composition with Genetic Algorithm and Ant Colony Optimization

CAO Teng-Fei, FU Yun-Qing, ZHONG Ming-Yang

(College of Software Engineering, Chongqing University, Chongqing 400044, China)

**Abstract:** To improve the convergence ability of service selection technology in process of Web service composition, the paper presents a multi-objective optimization strategy based on genetic algorithm and ant colony algorithm to solve global optimization problem in QoS-based Web service composition. In the paper, global optimization problem in Web service composition is presented as a QoS optimal routing problem. And then, an improved genetic algorithm is proposed to get pheromone distribution in initial route of ant colony algorithm. At last, an improved ant colony algorithm is presented to get the optimal solution. Simulation result suggests that the improved algorithms can get the optimal routing in less evolutionary generation than typical algorithms, and improve global search ability in Web Service composition.

**Key words:** web service composition; ant colony algorithm; genetic algorithm; QoS; global optimum

## 1 引言

随着 Web 服务技术的普及与发展, Web 服务已经成为下一代分布式处理系统的核心, 共享在网络上的 Web 服务越来越趋于稳定和易用, 但是单一的 Web 服务功能简单且有限, 难以满足某些实际应用的需求, 因此有必要对现有的单个的 Web 服务进行组合, 以构建功能更为强大的 Web 服务来支持各种应用需求。满足相同功能需求而具有不同 QoS 参数的 Web 服务实例存在多个, 如何从中选择满足各服务节点功能需求的具体服务, 构建一个可执行的服务链来完成用户的需求称为 Web 服务组合问题。

Web 服务组合问题属于 NP 难问题, 传统典型的解决主要采用穷举算法和贪婪算法。由于求最优解的穷举算法复杂度太高, 求局部最优解的贪婪算法不一定能解得全局最优解, 当前较流行的 Web 服务选择算法主要是采用基于启发式的选择算法, 如遗传算法、粒子群优化算法和蚁群优化算法等<sup>[1,2]</sup>。这些方法都存在如收敛性能较差以及搜索全局最优解的能力不强等问题。文章在现有研究的基础上, 考虑上述研究方法存在的不足, 提出了一种基于遗传算法与蚁群算法相融合的 Web 服务组合算法。

<sup>①</sup> 基金项目:国家自然科学基金(71102065)部分资助

收稿时间:2011-09-28;收到修改稿时间:2011-10-21

## 2 Web服务组合模型定义及问题转换

### 2.1 Web 服务组合模型的基本概念

抽象服务 (Abstract Service, 简称 AS). 抽象服务是构成服务组合流程模型的基本逻辑单元, 抽象服务仅包含功能描述和接口信息; Web 服务组合流程建模中, 建模人员利用服务结点按照特定的业务逻辑建立通用服务组合流程模型。

具体服务 (Concrete Service, 简称 CS). 在 Web 服务组合模型中, 它是一个具有特定功能的真实的服务, 它可以是一个原子服务也可以是一个组合服务, 它是通过一个 QoS 属性向量来标识这个 Web 服务性质的, QoS 属性向量见下文服务组合模型转换。

服务候选集 (Service Candidate Set, 简称 C). 服务候选集是指由不同服务提供者提供的具有相同调用接口以及能够实现相同功能的一组服务, 同一服务群中服务具有相同的功能, 所不同的是各个服务的 QoS 属性。

### 2.2 Web 服务组合模型的问题转换

Web 服务动态组合问题就是在服务组合模型执行过程中, 从各个抽象服务 {AS<sub>1</sub>, AS<sub>2</sub>, ..., AS<sub>m-1</sub>, AS<sub>m</sub>} 对应的服务候选集 {C<sub>1</sub>, C<sub>2</sub>, ..., C<sub>m-1</sub>, C<sub>m</sub>} 中选择具体的 Web 服务组成一个可执行的服务链, 使得服务链在满足特定 QoS 约束的前提下, 多个目标 (QoS 参数) 达到最优化, 本文只讲述顺序型服务组合路径, 对于其他情形的路径, 如并行、条件和循环, 根据文献 [1] 的方法可递归成顺序型路径。Web 服务组合工作流程图 1 所示。

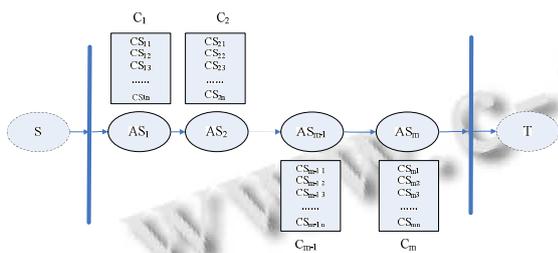


图 1 Web 服务组合工作流程图

该组合服务的问题模型可以进一步转化为一个有向无环图, 如图 2 所示。

在当前服务组合研究中, 由于 QoS 描述的必要性和重要性, 很多学者给出了不同的属性描述方式 [2]。由于篇幅限制, 本文采用了时间、价格和可靠性等几个最常用的参数作为服务组合模型中具体服务 CS<sub>i</sub> 的 QoS 属性向量 (T<sub>i</sub>, C<sub>i</sub>, R<sub>i</sub>)。其中, T<sub>i</sub> 表示服务从开始到

结束所花的时间 time; C<sub>i</sub> 表示服务提供商提供的相关服务的价格 cost; R<sub>i</sub> 表示服务的可靠性 reliability, 顺序型服务组合模型中 QoS 属性向量对应的计算函数如表 1 所示。

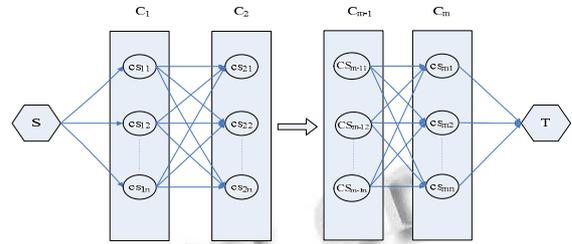


图 2 服务组合模型的问题转换

表 1 顺序型服务组合模型中 QoS 属性的计算函数

QoS 属性	时间 (T(t))	费用 (C(t))	可靠性 (R(t))
计算函数	$\sum_{i=1}^m T(t_i)$	$\sum_{i=1}^m C(t_i)$	$\prod_{i=1}^m R(t_i)$

## 3 服务组合中遗传算法与蚁群算法相融合

### 3.1 蚁群遗传算法基本思想

遗传算法具有快速的全局搜索能力, 具有可扩展性, 易于和其它算法相结合, 但遗传算法对于系统中的反馈信息利用不充分, 当求解到一定范围时会做大量的无效迭代; 蚁群算法由于具有正反馈机制, 通过信息素的更新规则使它具有很强的鲁棒性和全局搜索能力, 但同时由于其初始信息素的匮乏, 导致算法开始速度比较慢。

为了克服两种算法各自的缺陷, 形成优势互补。首先利用遗传算法的随机搜索、快速性和全局收敛性产生相关问题的初始信息素分布; 然后利用蚁群的并行性、正反馈机制以及求解效率高等特征来对相关问题进行求解, 这种遗传算法与蚁群算法融合的算法称为蚁群遗传算法 [3]。而本文正是利用两者结合的优点, 将其引入到 Web 服务组合中寻求最优路径的问题中来。

### 3.2 Web 服务组合模型建立

把服务候选集中的具体服务表示为节点, 把两个服务候选集的两个具体服务的路径表示为连接两节点的边, 把需要选择的具体服务的 QoS 值作为该边的权, 那么组合服务问题就可以被抽象成为一个带 QoS 约束的有向图。根据文中 2.2 所讲, 我们用有向图 G=(V, C, A, QoS) 来表示组合服务问题, 其中 V={S, C<sub>1</sub>, C<sub>2</sub>,

$C_m, \dots, T\}$  为一系列服务候选集的集合。s 表示起始点, t 表示目标点,  $C_i (i=1, 2, \dots, m-1, m)$  表示 m 个服务候选集, 每一个候选集  $C_i$  都有一组具体服务  $\{cs_{i1}, cs_{i2}, \dots, cs_{i(n-1)}, cs_{in}\}$  可供流程选择。具体服务的节点  $A = \{(cs_{in}, cs_{jn}) | cs_{in} \in C_i, cs_{jn} \in C_j, i \neq j\}$  为一系列弧的集合,  $QoS_{in}(T_{in}, C_{in}, R_{in})$  与相应的具体服务相对应, 表示具体服务  $CS_{in}$  的服务质量评价指标。这样最终 Web 服务组合问题就是在带权有向图无环中, 寻求从起始点 S 到目标点 T 的一条 QoS 总和最优的路径问题<sup>[4]</sup>。

### 3.3 改进的遗传算法求蚁群初始信息素

**改进的种群编码** 采用路径编码方法, 以从起始节点到目标节点路径上的节点序列作为染色体的编码, 一条路径就是一条染色体, 因此每个染色体都代表了一个可行解, 对一个染色体而言, 其中的每个基因的值都代表了一组可选服务的选择结果<sup>[5]</sup>, 其编码形式如下图所示:

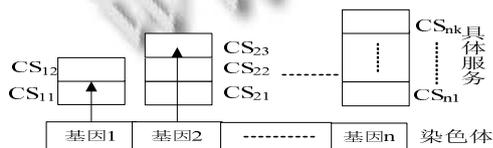


图3 染色体编码方式

**适应度函数** 设定染色体 g 的适应度函数为  $F(g) = \frac{w_r R(g)}{w_c C(g) + w_t T(g)}$ , 在上述  $F(g)$  的计算公式中,  $w_r$ ,

$w_c, w_t$  表示各种 QoS 属性指标的权重,  $\{w_r + w_c + w_t = 1\}$ ;  $R(g)$ 、 $C(g)$ 、 $T(g)$  分别表示染色体中组合服务的时间、价格以及可靠性的 QoS 属性分量值。

**初始种群生成** 以随机的方式生成多个服务候选集, 依次从每个服务候选集中随机选出具体的服务组成染色体, 形成规模为 N 的初始种群。

**选择算子** 采用轮盘赌选择方法和最佳个体保留策略。

**改进的单点交叉算子** 两条染色体随机选择一个公共的节点作为交叉点, 从起始节点到交叉点的节点顺序保持不变, 将交叉点与目标节点间的顺序相互交换。

**变异算子** 从染色体中随机选择一个节点(起始节点、目标节点除外)作为变异点, 从它对应的服务候选集中随机选择新的 Web 服务(变异基因)替换当前基因。

**遗传算法结束条件** 设遗传算法迭代次数为 N, 设置  $N_{\min} \leq N \leq N_{\max}$ ,  $N_{\min}$ 、 $N_{\max}$  分别为遗传算法最小和最大迭代次数, 其上下限由人工自行设定。在遗传算法的迭代过程中同时统计进化率, 其公式为:

$$\overline{F(g)} = (F_n(g) - F_{n-1}(g)) / \sum_{i=1}^n F_i(g), \text{ 在设定的迭代次数范围内,}$$

若连续三次进化率都小于最小进化率时, 则停止遗传算法迭代过程, 进入蚁群算法。

### 3.4 遗传算法与蚁群算法相融合

当遗传算法按照规则执行结束后, 对于迭代结束后生成的 N 条从起始节点 S 到目标节点 T 的路径, 选择其中适应值最高的前 10% 条路径, 设它们的集合为 V。对于 V 中的每个解  $v_j$ , 按照下式将其转换为蚁群算法部分信息素初值:  $\tau_G = K_j \cdot F(g)$ 。其中:  $k_j$  是一个常数;  $F(g)$  是路径  $p(i, j)$  适应度函数值,  $0 \leq j \leq n$ 。当 V 中有多条路径经过  $p(i, j)$  时, 则对  $\tau_G$  进行叠加。

### 3.5 改进的蚁群算法求最优解

#### 初始信息素设置

通过遗传算法得到了一定的路径信息素之后, 为了防止算法过早收敛, 陷入局部最优解, 本文采用最大-最小蚂蚁系统(MMAS)中的方法。综合这两部分, 信息素的初值  $\tau_s$  设置为:  $\tau_s = \tau_c + \tau_G$  其中:  $\tau_c$  是一个常数, 相当于 MMAS 算法中的  $\tau_{\max}$ ,  $\tau_G$  是遗传算法求解出的最优解所转换的信息素值。

#### 信息素更新规则

$\tau_{ij}(t)$  表示在 t 时刻路径  $p(i, j)$  上的遗留的信息素, 为了避免残留信息素过多引起的残留信息淹没启发式信息的问题, 使蚂蚁能准确感知路径信息, 我们规定在每只蚂蚁走完所有的服务节点后, 对  $p(i, j)$  路径上的信息素进行全局更新:

$$\tau_{ij}(t+n) = \rho \cdot \tau_{ij}(t) + \Delta \tau_{ij}(t) \quad (1)$$

$$\Delta \tau_{ij}(t) = \sum_{k=1}^m \Delta \tau_{ij}^k(t) \quad (2)$$

式中  $\Delta \tau_{ij}(t)$  表示本次循环中路径  $p(i, j)$  的信息素增量;  $\Delta \tau_{ij}^k$  为第 k 只蚂蚁在本次循环中途经路径  $p(i, j)$  上信息素的增量。

#### 信息素的限制规则

$$\tau_{ij}(t+n) = \begin{cases} \tau_{\min} & \tau_{ij}(t) < \tau_{\min} \\ \tau_{ij}(t) & \tau_{\min} \leq \tau_{ij}(t) \leq \tau_{\max} \\ \tau_{\max} & \tau_{ij}(t) > \tau_{\max} \end{cases} \quad (3)$$

为了提高全局搜索能力，文中采用信息素的限制规则，如公式(3)所示；即在蚁群算法运算时期，通过采用最大和最小信息素阈值来防止因信息素过高而陷入局部最优或者因其过低而丧失路径选择的几率。

自适应的值改变法

$$\rho(t) = \begin{cases} 0.95\rho(t-1) & \text{如果 } 0.95\rho(t-1) > \rho_{\min} \\ \rho_{\min} & \text{否则} \end{cases} \quad (4)$$

$\rho \in (0, 1)$ 表示信息素的残留因子系数，文中采用自适应  $\rho$  值改变方法保证了在一定的搜索速度下有效的提高了混合算法的全局搜索能力，其中  $\rho_{\min}$  为设定  $\rho$  的最小值。

蚁周 (ant-cycle) 模型计算方法

由于信息素的增量  $\Delta\tau_{ij}^k$  与该服务的 QoS 成正比关系，因此文中采用蚁周 (ant-cycle)模型计算，进行全局更新，即

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{f}{L_k} & \text{如果蚂蚁 } k \text{ 在本次循环中经过路径 } P(i,j) \\ 0 & \text{否则} \end{cases} \quad (5)$$

式中， $L_k$  为蚂蚁  $k$  经过本次循环的路径的长度， $f$  为所要选择的原子服务的 QoS 衡量指标，其值可以据公式(6)、(7)计算。

$$f = 1 / \sqrt{T_i^2 + C_i^2 + 1/R_i^2} \quad (6)$$

$$L_i = \sum_{i=1}^n \sqrt{(T_{i+1}-T_i)^2 + (C_{i+1}-C_i)^2 + 1/(R_{i+1}-R_i)^2} \quad (7)$$

(6)式中： $T_i, C_i, R_i$  为路径中某个具体服务的 QoS 属性向量的权值；(7)式中  $L_i$  表示蚂蚁在服务候选集  $C_{i+1}$  中所选择的具体服务与服务候选集  $C_i$  所选择的具体服务之间的距离。

轮盘赌原理的路径选择策略

在  $t$  时刻位于节点  $i$  的第  $k$  只蚂蚁选择节点  $j$  为下一个目标的概率为：

$$P_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{j \in allowed} [\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta} \cdot j \in allowed & \\ 0 & \text{否则} \end{cases} \quad (8)$$

式中  $\eta_{ij}$  表示由节点  $i$  转移到节点  $j$  的启发信息，本文中选择具体服务的概率与该具体服务的 QoS 成正比关系，因此可取  $\eta_{ij} = kf$  其中  $k$  为常数， $f$  值由公式(6)

计算得出，参数  $\alpha$  和  $\beta$  分别用来控制信息素浓度和启发信息的相对重要程度。allowed 是第  $k$  只蚂蚁下一步可以选择的服务候选集。

通过路径选择策略计算出节点  $i$  的可选服务的概率后，使用轮盘赌选择方法来确定下一个候选节点  $j$ ，这样可以一定程度上延缓算法陷入局部最优。

3.6 服务组合中混合遗传蚁群算法的算法流程

针对上述所讲内容，改进的算法流程如下：

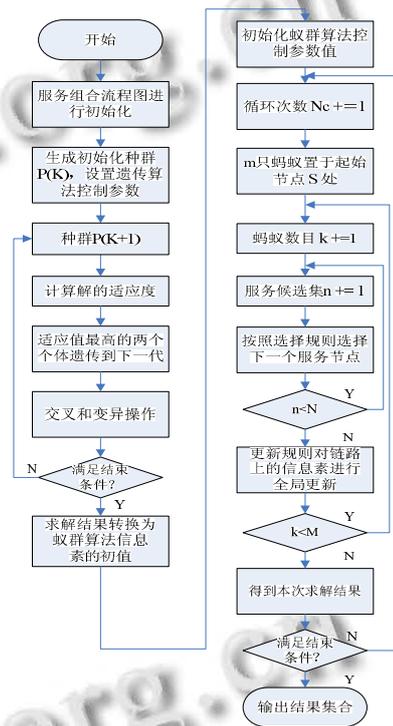


图 4 服务组合模型中混合遗传蚁群算法流程

4 仿真实验与结果分析

为了验证本文方法的可行性和有效性，本文将改进算法与经典蚁群算法在服务组合问题中进行了求解和对比实验，实验参数设定见表 3。限于篇幅，本文只列出了一组假定的服务候选集基础数据，如表 2 所示，其中含有 20 个基本服务， $T, C, R$  为服务的 QoS 属性分量， $T$  表示时间（范围在 5-30 之间）、 $C$  表示价格（范围在 40-90 之间）、 $R$  表示可靠性（范围在 1.0-2.0 之间），其中  $T, C, R$  均为随机生成的，其他 14 组基础数据集合也是用相同的方法随机生成。模拟实验过程中，适当调节参数均可得到比较理想的结果。由实验数据收敛曲线图 5 可以看出改进的遗传蚁群算法在寻求组合服务的最优解的过程中能在较少的进化代数

内得出最优解，该算法克服了传统蚁群遗传算法的收敛性能的不足，提高了求解组合问题的效率。

表 2 服务组合实验基础数据

服 务	服务候选集合 1									
	1	2	3	4	5	6	7	8	9	10
T	9	9	17	18	20	22	9	5	21	14
C	58	67	73	49	54	80	73	77	65	43
R	1.2	1.8	1.3	1.5	1.6	1.3	1.4	1.9	1.1	1.3
	11	12	13	14	15	16	17	18	19	20
T	8	12	16	25	23	18	9	6	7	17
C	41	53	48	67	75	64	88	47	80	77
R	1.4	1.2	1.5	1.7	1.3	1.3	1.1	1.5	1.1	1.3

表 3 实验参数设定

参数名称	参数取值
实验次数	70
进化代数	50
种群规模	40
交叉概率	0.7
变异概率	0.05
信息启发因子 $\alpha$	1
期望启发因子 $\beta$	2
信息素挥发系数 $\rho$	0.95
蚂蚁数目	30
候选集合	15
候选 Web 服务	20
QoS 属性权重	$w_t=0.3, w_c=0.4, w_r=0.3$

### 5 结语

本文尝试使用遗传算法和蚁群算法融合来求解组合服务问题，遗传算法和蚁群算法作为新兴的模拟进

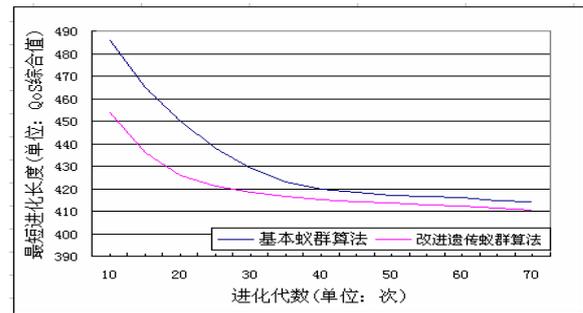


图 5 基本蚁群算法与改进混合算法的收敛曲线

化算法有它们各自的优点，在空间复杂度上与传统的算法相比具有较大的优越性，把它们结合起来运用在组合服务中，使求解过程中尽量避免陷入局部最优同时提高了组合服务流程的执行效率，具有一定的理论参考价值和实际意义，我们的下一步工作是进一步完善该在组合服务中的实际应用，来达到用户对组合服务的需求。

### 参考文献

- 1 Fang Qiqing, et al. A Global QoS Optimizing Web Services Selection Algorithm based on MOACO for Dynamic Web ServiceComposition. International Forum on Information Technology and Applications, 2009,10(11):38-42.
- 2 彭晓明,何炎祥,朱兵舰.蚁群算法在 Web 服务组合中的应用.计算机工程,2009,35(10):182-186.
- 3 熊志辉,李思昆,陈吉华.遗传算法与蚁群算法动态融合的软硬件划分.软件学报,2005,16(4):503-512.
- 4 刘书雷,等.一种服务聚合中 QoS 全局最优服务动态选择算法.软件学报,2007,18(3):646-656.
- 5 张晓光,等.一种基于遗传算法 QoS 敏感的 Web 服务组合法.山东大学学报(理学版),2007,42(9):1-6.

(上接第 69 页)

用,2004,(27):34-37.

- 16 罗天虎.基于 Multi-Agent 的虚拟组织知识管理研究.情报杂志,2007,(2):59-61.
- 17 龚伟同.Google—一个浑序组织的实验.商务周刊,2008,(5):86-89.
- 18 赵一等.VISA 的组织模式与物流混序联盟的构建.上海企

业,2005,(12):51-53.

- 19 金吾伦.混序组织及其应用.学习时报,2005,(10):1-3.
- 20 王小燕.混序组织—未来组织.中外管理,2003,(4):22-23.
- 21 樊继光.Visa 业务运营的三大核心及其对我国的借鉴.商场现代化,2008,(7):66-67.
- 22 王艳华.混序组织的边界. IT 经理世界,2008,(14):96-98.