

文章编号: 1674-9057(2017)04-0699-08

doi:10.3969/j.issn.1674-9057.2017.04.024

# 基于改进粒子群优化算法的 Web 服务组合

陆湘鹏, 叶恒舟

(桂林理工大学“嵌入式技术与智能信息处理”广西高校重点实验室, 广西 桂林 541006)

**摘要:** 针对现有的基于 PSO 的服务组合方法存在只能支持具体的工作流或仅包含顺序结构的工作流、仅使用迭代次数作为终止条件、未处理种群早熟或处理效果不是很理想等问题, 提出了一种基于改进的粒子群优化算法的 Web 服务组合方法。它可支持包含顺序、并发与选择模式的工作流; 向初始粒子中添加了少量通过局部优选策略生成的粒子; 调整了认知系数和社会系数的设置方案; 引入种群的多样性概念, 提出了一种早期多样性修补机制以避免早熟; 设定了一种提前终止迭代操作的条件。实验结果表明, 该算法在寻优效果与时间开销上具有一定优势。

**关键词:** 粒子群优化; Web 服务组合; 早熟收敛; 提前终止条件

**中图分类号:** TP393

**文献标志码:** A

## 0 引言

互联网技术的迅速发展和 Web 服务技术的逐步成熟, 促进了 Web 服务的应用与繁荣。Internet 上的单个 Web 服务的功能往往比较简单。为了满足日益复杂的用户需求, 将多个 Web 服务按照一定的规则和要求组合, 是一种有效途径。

Web 服务组合可以分为 3 阶段: 1) 确定描述事务逻辑的工作流; 2) 为工作流中的每个任务搜索候选服务库; 3) 生成满足用户需求的组合服务。本文仅考虑第 3 阶段的问题, 即在工作流及候选服务库已知的前提下, 为每个任务从大量候选服务中快速、动态地选择一个候选服务, 使得到的组合服务满足用户的功能与非功能需求, 这是一个 NP 难问题<sup>[1]</sup>。基于贪心策略的局部优选策略<sup>[2-3]</sup>具有较好的时空性能, 但不能保证全局约束。将该问题建模为一个已知模型, 如整数线性规划<sup>[4]</sup>、多维多选择背包问题<sup>[2]</sup>、混合整数线性规划<sup>[5]</sup>等, 再采用进化类算法求解是一种有效途

径。文献 [6-8] 等采用了遗传算法, 存在收敛速度慢、稳定性较差等不足。与遗传算法相比, 粒子群优化算法具有参数少、收敛快的优点, 在很多优化问题上表现出了良好的搜索能力。

一些学者开始应用粒子群优化算法处理基于 Quality of Service (QoS) 的 Web 服务组合问题: Ludwig 提出一种将 PSO 算法与带权二分图的最优匹配算法结合的方法<sup>[9]</sup>; 刘莉平等提出将粒子的飞行速度定义为 QoS 的值、位置的更新定义为将服务替换为与此 QoS 最接近的服务实例的方法<sup>[10]</sup>; 胡珀等提出的方法采用了变异策略、线性变化的学习因子、新的惯性权重策略<sup>[11]</sup>。以上 3 种算法的实验与基本的粒子群优化算法比较, 说服力欠佳; 后两种只测试了给定的服务组合业务模型, 没有针对随机的服务组合模型进行测试。Kang 等利用理想点的方法将具有多目标服务组合优化问题转化为单目标问题<sup>[12]</sup>, 但其测试的服务组合业务模型的工作流模式固定、任务个数偏少。温涛等提出的基于粒子圆周轨道和零惯性权重, 使

收稿日期: 2016-03-08

基金项目: 国家自然科学基金项目 (51365010); 广西自然科学基金项目 (2014GXNSFBA118269); 广西高校嵌入式技术与智能信息处理重点实验室开放基金项目 (2016-01-02)

作者简介: 陆湘鹏 (1992—), 男, 硕士, 计算机技术专业, sapper@live.com。

通讯作者: 叶恒舟, 博士研究生, 副教授, 18123408@qq.com。

引文格式: 陆湘鹏, 叶恒舟. 基于改进粒子群优化算法的 Web 服务组合[J]. 桂林理工大学学报, 2017, 37(4): 699-706.

用基于三角函数的非线性动态学习因子及种群早熟收敛预测与处理方法的算法<sup>[13]</sup>、张燕平等提出的针对前者的改进算法<sup>[14]</sup>，均只测试了顺序 workflow，没有测试带有并发、选择、循环等模式的常见 workflow。

针对上述问题，本文提出的改进 PSO 算法可支持包含顺序、并发、选择的复杂 workflow。对 PSO 的改进主要体现在：向初始粒子中添加了少量通过局部优选策略生成的粒子；调整了认知系数和社会系数的设置方案；针对 PSO 算法容易过早收敛于局部最优值的问题，提出了一种渐近式的种群多样性修补策略以避免早熟；在种群初始化、学习因子设计等方面也作了一些改进。实验结果表明，该算法在寻优效果与时间开销上具有一定优势。在服务选择时，现有的方法均以迭代次数作为终止条件，不同问题的最优迭代次数难以选择，本文提出的提前终止迭代操作方法，免除了选择一个固定的终止迭代次数的麻烦，比现有以迭代次数来终止迭代的方法所用的平均时间更少，具有一定的应用价值。

## 1 Web 服务组合问题

### 1.1 QoS 属性及其归一化

随着网络上 Web 服务数量的增加，存在着很多功能相同的服务，因而引入了 QoS 来表征服务的非功能特性。常见的 QoS 属性包括服务价格、执行时间、可靠性、吞吐率、信誉度等。这些属性可以分为两类：积极属性，如可靠性、信誉度等，其值越大，服务质量越好；消极属性，如执行时间、服务代价等，其值越小，服务质量越好。

Web 服务各属性的单位、取值范围、影响效果不同，若要综合评价 QoS 属性，需要作归一化处理。选择归一化公式：

$$Q'_{t,i,k} = \begin{cases} \frac{Q_k^{\max} - Q_{t,i,k}}{Q_k^{\max} - Q_k^{\min}}, & \text{当 } k \text{ 为消极属性时;} \\ \frac{Q_{t,i,k} - Q_k^{\min}}{Q_k^{\max} - Q_k^{\min}}, & \text{当 } k \text{ 为积极属性时。} \end{cases} \quad (1)$$

其中， $Q_{t,i,k}$ 、 $Q'_{t,i,k}$  分别表示第  $t$  个任务的第  $i$  个候选服务的第  $k$  个 QoS 属性归一前、后的值， $Q_k^{\max}$ 、 $Q_k^{\min}$  分别表示第  $k$  个 QoS 属性的最大值、最小值。

### 1.2 workflow 模式

workflow 用于描述组合服务的业务逻辑。在常见的工作流中，涉及顺序、并发、选择及循环等 4 种结构模式<sup>[15]</sup>，如图 1 所示。

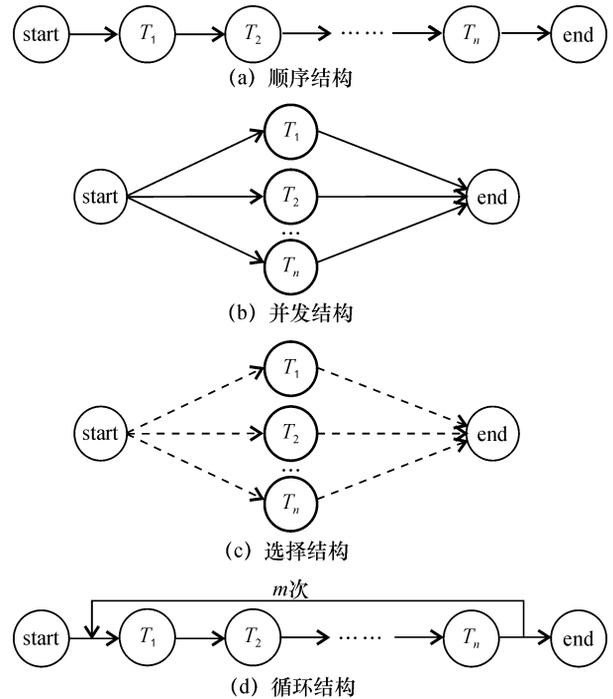


图 1 Web 服务组合的 4 种基本结构

Fig. 1 Basic structures of Web services composition

### 1.3 QoS 聚合函数

Web 服务组合后需要得到一个整体 QoS 值来作为评价服务组合方案好坏的标准和依据，这就需要聚合 QoS 属性。如表 1 所示，不同的 QoS 属性

表 1 聚合函数的定义

Table 1 Definition of aggregate functions

QoS 属性	顺序	并发	选择	循环
执行时间	$\sum_{i=1}^n Q_i$	$\max\{Q_1, \dots, Q_n\}$	$\min\{Q_1, \dots, Q_n\}$	$m \times \sum_{i=1}^n Q_i$
服务价格	$\sum_{i=1}^n Q_i$	$\sum_{i=1}^n Q_i$	$\min\{Q_1, \dots, Q_n\}$	$m \times \sum_{i=1}^n Q_i$
信誉度	$(\sum_{i=1}^n Q_i)/n$	$(\sum_{i=1}^n Q_i)/n$	$(\sum_{i=1}^n Q_i)/n$	$(\sum_{i=1}^n Q_i)/n$
可靠性	$\prod_{i=1}^n Q_i$	$\prod_{i=1}^n Q_i$	$\max\{Q_1, \dots, Q_n\}$	$(\prod_{i=1}^n Q_i)^m$

注： $Q_i$  为第  $i$  个候选服务的 QoS 属性值； $n$  为任务个数； $m$  为循环次数。

在不同的结构模式下具有不同的聚合规则<sup>[5,10]</sup>。

#### 1.4 Web 服务组合问题

设给定的工作流程中涉及任务  $t_1, t_2, \dots, t_n$ ，每个任务  $t_i (i = 1, 2, \dots, n)$  拥有候选服务集  $\{s_{i1}, s_{i2}, \dots, s_{im}\}$ 。Web 服务组合的目标就是为每个任务挑选一个候选服务，使得到的组合服务满足用户的偏好。

## 2 传统的粒子群算法

PSO 算法由 Eberhart 和 Kennedy 于 1995 年提出，是一种模拟鸟类捕食的进化类算法<sup>[16]</sup>。PSO 算法将鸟类个体看作搜索空间中具有重量和体积的粒子，在搜索空间中以一定的速度飞行，开始时粒子的位置、速度随机，通过迭代搜寻最优值。飞行速度由个体飞行经验和群体飞行经验动态调整，即每一次迭代时，粒子同时跟踪两个“极值”来更新自己的速度：个体最优解，即每个粒子自身所找到的最优解；群体最优解，即整个群体目前找到的最优解<sup>[17]</sup>。

设问题的搜索空间为  $D$  维，粒子总数为  $N$ ，粒子群中第  $i$  个粒子的位置表示为  $\mathbf{X}_i = (X_{i1}, X_{i2}, \dots, X_{id})$ ，第  $i$  个粒子的速度表示为  $\mathbf{V}_i = (V_{i1}, V_{i2}, \dots, V_{id})$ ，第  $i$  个粒子的最优位置表示为  $\mathbf{P}_i = (P_{i1}, P_{i2}, \dots, P_{id})$ ，整个粒子群的最优位置表示为  $\mathbf{G} = (G_1, G_2, \dots, G_d)$ ，第  $i$  个粒子第  $k+1$  代的速度和位置更新公式为

$$\begin{cases} \mathbf{V}_i^{k+1} = w \times \mathbf{V}_i^k + c_1 r_1 (\mathbf{P}_i^k - \mathbf{X}_i^k) + c_2 r_2 (\mathbf{G}^k - \mathbf{X}_i^k), \\ \mathbf{X}_i^{k+1} = \mathbf{X}_i^k + \mathbf{V}_i^{k+1}, \\ \|\mathbf{V}_i^{k+1}\| \leq \mathbf{V}_{\max} \end{cases} \quad (2)$$

其中： $0 < i \leq N, 0 < d \leq D, i \in \mathbf{Z}^+, d \in \mathbf{Z}^+; r_1, r_2$  为  $[0, 1]$  的服从均匀分布的随机数； $w$  是惯性权重； $w \times \mathbf{V}_i^k$  表示粒子先前速度所起的惯性作用，Shi 和 Eberhar 在 1998 年正式引入惯性权重的概念<sup>[18]</sup>，这部分代表前一代速度对下一代速度的影响<sup>[19]</sup>； $c_1, c_2$  为学习因子； $c_1$  为认知系数， $c_1 r_1 (\mathbf{P}_i^k - \mathbf{X}_i^k)$  是个体的“认知”部分，因为它仅考虑粒子自身的经验，表示粒子本身的思考； $c_2$  为社会系数， $c_2 r_2 (\mathbf{G}^k - \mathbf{X}_i^k)$  是个体的“社会”部分，表示粒子间的社会信息共享<sup>[20]</sup>； $\mathbf{V}_{\max}$  是代表最大速度的常数。

## 3 基于改进粒子群优化算法的 Web 服务组合

### 3.1 粒子编码策略

采用整数定长的编码方式，第  $i$  个粒子的位置  $\mathbf{X}_i = (X_{i1}, X_{i2}, \dots, X_{in})$  表示一种服务组合方式， $n$  为任务个数， $X_{ij}$  表示为第  $i$  个任务选择编号为  $j$  的候选服务。

### 3.2 局部优选预处理

通常粒子群优化算法的初始位置是随机生成的。考虑到具有较优 QoS 值的服务有较大的可能成为种群最优值选用的服务，在初始化粒子群时，加入了 5 个用局部优选法<sup>[2]</sup>选择的粒子，它们分别以综合 QoS 效用最高、执行时间最短、服务价格最低、信誉度最高、可靠性最高为优选目标。

### 3.3 学习因子设计

学习因子  $c_1, c_2$  取值的变化反映了粒子自身认知能力和社会认知能力的变化。本文采用了 Ratnaweera 等提出的认知系数与社会系数随时间动态调整的策略<sup>[21]</sup>，即

$$\begin{cases} c_{1f} = (c_{1f} - c_{1c}) \times t/T + c_{1c}; \\ c_{2f} = (c_{2f} - c_{2c}) \times t/T + c_{2c}, \end{cases} \quad (3)$$

其中： $c_{1f} = 0.5, c_{1c} = 2.5, c_{2f} = 2.5, c_{2c} = 0.5, t$  为当前迭代的次数， $T$  为总迭代次数。

### 3.4 多样性修复策略

PSO 算法容易早熟，本文采用一种早期多样性修复策略以避免或减缓早熟现象。基本思想是：在每次迭代时检测种群多样性，并根据多样性值、当前迭代次数、粒子的当前值与最优值决定该粒子是否应该由新的随机粒子替代。

对于粒子  $\mathbf{X}_i$ ，在第  $t$  次迭代过程中会被新的随机粒子替代的概率  $P_i$  为

$$P_i = \begin{cases} 1 - \frac{t}{T}, & \text{当 } pd_{\text{cur}} \leq pd_{\text{min}} \text{ 时;} \\ \min\{1, \alpha[\log(f(\mathbf{G})) - \log(f(\mathbf{X}_i))]\} \times \\ \frac{pd_{\text{cur}} - pd_{\text{min}}}{pd_{\text{max}} - pd_{\text{min}}}, & \text{当 } pd_{\text{min}} < pd_{\text{cur}} < pd_{\text{max}} \text{ 时;} \\ 0, & \text{当 } pd_{\text{cur}} \geq pd_{\text{max}} \text{ 时。} \end{cases} \quad (4)$$

其中： $pd_{\text{max}}, pd_{\text{min}}$  为种群多样性的两个阈值； $t, T$  分别为当前迭代次数和总迭代次数； $\mathbf{G}$  为种群当前的最优粒子； $f(\mathbf{G}), f(\mathbf{X}_i)$  分别为  $\mathbf{G}$  和  $\mathbf{X}_i$  的适

应值;  $\alpha$  为多样性修复参数;  $pd_{cur}$  为当前种群的多样性。

$$pd_{cur} = \left( \sum_{i=1}^n \text{dis}(X_i, \mathbf{G}) \right) / N. \quad (5)$$

其中:  $N$  是种群的粒子个数;  $\text{dis}(X_i, \mathbf{G})$  是粒子  $X_i$  和  $\mathbf{G}$  之间的布尔距离, 由式(6) 确定。

$$\text{dis}(X_i, X_j) = \left( \sum_{d=1}^D ((X_{i,d} = X_{j,d})? 0:1) \right) / D. \quad (6)$$

$P_i$  的计算基于如下思维。在 PSO 迭代的前期, 种群多样性较大, 即  $pd_{cur} \geq pd_{max}$ , 此时  $P_i = 0$ , 即不作替换; 中期, 种群多样性适中, 即  $pd_{min} < pd_{cur} < pd_{max}$ , 此时  $P_i$  取决于种群最优值和该粒子的适应值, 以概率  $P_i$  进行粒子“重生”操作, 即用随机生成的新粒子替代粒子  $X_i$ ; 后期, 种群多样性较小, 即  $pd_{cur} \leq pd_{min}$ , 此时  $P_i$  的大小取决于当前迭代次数和总迭代次数, 以  $P_i$  进行“拯救”种群操作, 即除了保留一个种群最优粒子外, 其余粒子以  $(1 - t/T)$  的概率用随机生成的新粒子替换。

### 3.5 提前终止策略

PSO 大多采用迭代次数作为终止条件, 但合理的迭代次数不容易选择, 往往会出现已经收敛仍继续迭代的情形, 有必要制定较为合理的提前终止条件。

实验发现, 当多样性适中, 且第  $k$  代相对效用增量较小时, 种群往往已经收敛或接近于收敛。因此, 提出了一种提前终止迭代过程 (即第  $k$  次迭代时终止) 的约束条件:

$$\begin{cases} pd_{min} \leq pd_{cur} \leq pd_{max}; \\ df^k \leq \left(1 - \frac{t}{T}\right) \times dfr. \end{cases} \quad (7)$$

其中,  $dfr$  为实验确定的常数;  $df^k$  表示第  $k$  代相对效用增量。

$$df^k = \frac{\sum_{i=1}^n (H_i^k - H_i^{k-1}) + (g^k - g^{k-1})}{g^k}, \quad (8)$$

其中:  $H_i^{k-1}$ 、 $H_i^k$  分别为第  $k-1$  代和第  $k$  代时粒子  $X_i$  的历史最优值;  $g^{k-1}$ 、 $g^k$  分别为第  $k-1$  和第  $k$  代时粒子群的最优值。

### 3.6 粒子适应值的计算

粒子适应值可用组合服务的综合服务质量度量<sup>[10]</sup>。若用  $f_1(x)$ 、 $f_2(x)$ 、 $f_3(x)$ 、 $f_4(x)$  分别代表组合服务  $x$  的执行时间、服务价格、信誉度与可靠性的聚合值, 则粒子适应值  $f(x)$  为

$$f(x) = \sum_{i=1}^4 [b_i \times f_i(x)], \quad (9)$$

其中:  $b_i$  为各 QoS 属性的权值, 本文取  $b_i = 0.25$ 。

### 3.7 算法流程

基于改进 PSO 的 Web 服务组合算法可描述为输入: workflow 及候选服务, 与 PSO 相关的参

数

输出: 种群最优值  $g$

step1. 将每个任务的候选服务按 QoS 从小到大排序。

step2. 初始化第 0 代粒子群, 计算  $X_i$  的适应度  $f(X_i)$ 、更新最优值。

step3. 对每一代种群进行如下操作:

① 根据式(2)更新粒子的位置和速度;

② 根据式(9)计算  $X_i$  的适应度  $f(X_i)$ , 更新最优值;

③ 根据式(5)计算当前种群多样性  $pd_{cur}$ , 根据式(4)应用多样性修复策略;

④ 根据式(8)计算相对效用增量  $df^k$ , 若满足式(7)转到 step4; 否则, 转到 step3。

step4. 输出种群最优值  $g$ 。

该算法的时间消耗主要集中在 step3, 其时间开销由实际迭代次数确定。由于实际迭代次数不大于  $T$  (预设的迭代次数), 算法的最坏时间复杂度为  $O(T)$ 。与常规 PSO 相比, 该算法增加了计算种群多样性和相对效用增量时的开销, 这两个量每次迭代需要计算一次, 其最坏时间复杂度也为  $O(T)$ 。算法的空间复杂度与常规 PSO 一致, 为  $O(N)$ ,  $N$  为粒子总数。

## 4 实验仿真与分析

### 4.1 实验环境及细节说明

本文实验是在 PC 机上实现的, 主要配置为: Intel(R) Core(TM) i7-4790K、CPU 4.0 GHz, 8 GB 内存, 64 位 Windows 10 TH2 操作系统, 系统及开发环境均运行在固态硬盘上。编程语言为 Java 1.8。在测试时的工作流是随机生成的, 包含顺序、选择和并发 3 种模式, 其中并发和选择模式的数量各占模式总数的 1/4, 同次测试中 3 种算法的工作流、初始位置、QoS 属性相同。选取了具有代表性的执行时间、服务价格、信誉度、可靠性 4 个属性, 分配给每个候选 Web 服务的 QoS 值在一定范围内

(表 2)随机生成。实验时,将本文的 PSO(记为 ZPSO)、刘莉平等<sup>[10]</sup>的 PSO(记为 XPSO)、胡珀等<sup>[11]</sup>的 PSO(记为 YPSO)进行对比,结果为 50 次运算后的平均值。考虑到 QoS 属性取值的随机性,每次测试得到的最优效用值是不一样的,因此,以 XPSO 得到的最优解为参照度量效用值。

表 2 QoS 属性的取值范围  
Table 2 Ranges of QoS attributes

QoS 属性	执行时间	服务价格	信誉度	可靠性
取值范围	(0,60]	(0,100]	(2,10]	(0.1,1]

### 4.2 参数取值

图 2、图 3 是在 100 个粒子、100 个任务、150 个候选服务,不同种群多样性阈值上、下限条件时,本文算法(ZPSO)的种群最优值和时间开销结果。可以看出,种群多样性阈值对效用与时间开销均有较为明显的影响。比较而言,阈值上限取 20、下限取 10 较为合适,其后的实验均取此值。

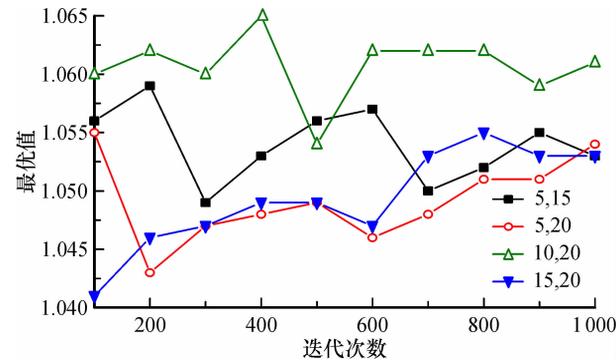


图 2 种群多样性阈值对种群最优值的影响  
Fig. 2 Performance of optimal value with varied threshold values

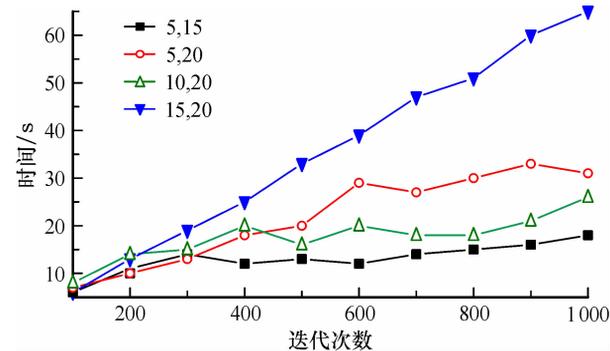


图 3 种群多样性阈值对时间开销的影响  
Fig. 3 Performance of time with varied threshold value

图 4、图 5 对比了在粒子个数为 100、最大迭代次数为 500 时,不同任务数和候选服务数情况下参数  $dfr$  的取值对本文算法(ZPSO)的迭代次数、种群最优值的影响。可以发现,随着  $dfr$  的增大,实际迭代次数趋于减少,且在一定范围内影响明显;而所获得的最优值随着  $dfr$  取值的增加而减小;问题的规模对  $dfr$  也存在一定的影响。在后续的实验,中,  $dfr$  取为 150。

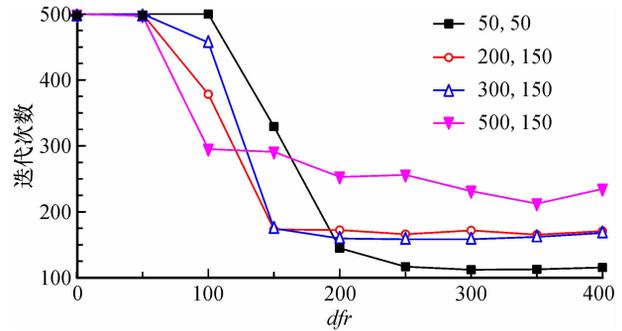


图 4 不同任务数、候选服务数条件下  $dfr$  取值对迭代次数的影响

Fig. 4 Performance of ZPSO iterations with varied  $dfr$  value on condition of different tasks and service candidates

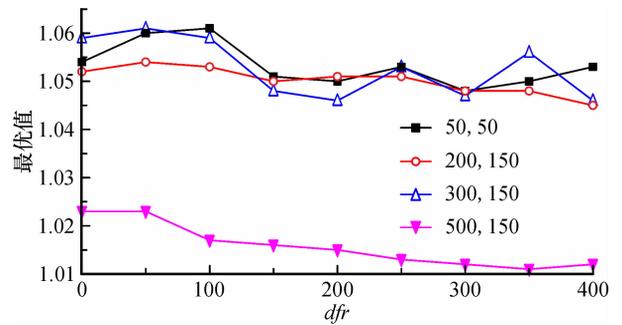


图 5 不同任务数、候选服务数条件下  $dfr$  取值对最优值的影响

Fig. 5 Performance of ZPSO optimal value with varied  $dfr$  value on condition of different tasks and service candidates

### 4.3 结果分析

图 6、图 7 对比了迭代次数对 3 种 PSO 效果的影响。本文的方法在最优值与时间开销上均有优势。特别是在迭代次数设置较大时,提前终止条件作用明显。

图 8、图 9 测试了迭代次数 300 时,任务数对 3 种算法的影响。图 8 表明,本文的算法在任务数增加时会减弱寻优效果,但影响要弱于 YPSO。时间开销与 YPSO 接近,远优于 XPSO。

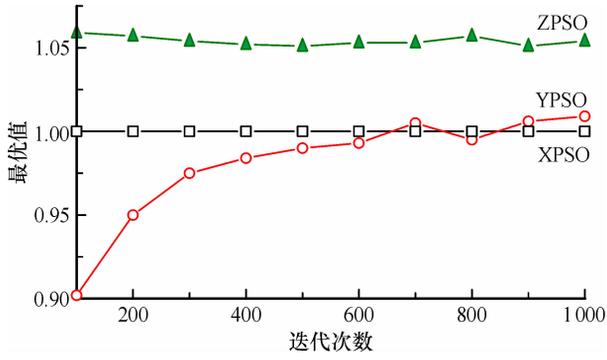


图6 迭代次数对种群最优值的影响

Fig. 6 Performance of optimal value with varied iterations

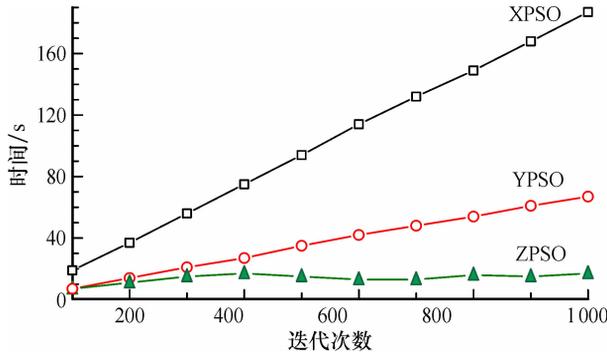


图7 迭代次数对执行时间的影响

Fig. 7 Performance of time with varied iterations

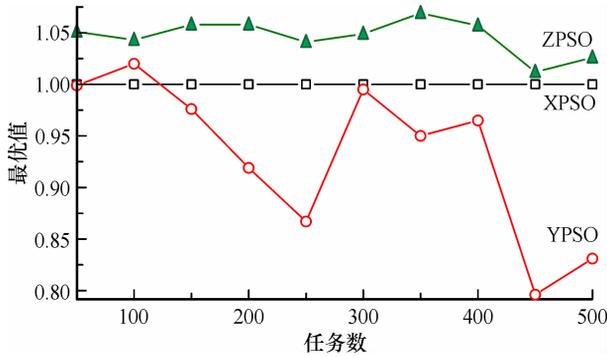


图8 任务数对最优值的影响

Fig. 8 Performance of optimal value with varied task number

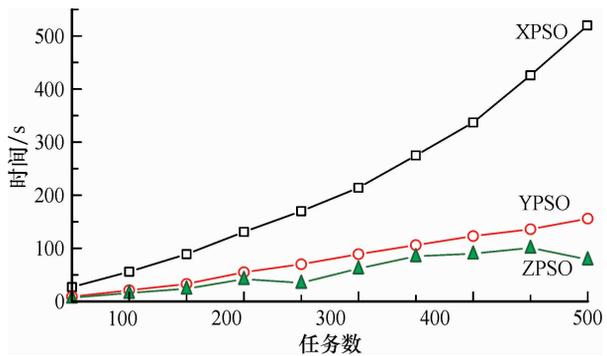


图9 任务数对执行时间的影响

Fig. 9 Performance of time with varied task number

图10、图11反映了候选服务数对算法性能的影响,其影响的效果、程度与任务数的影响基本一致。

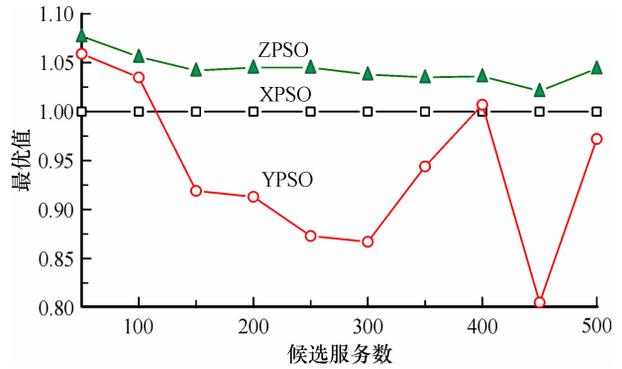


图10 候选服务数对最优值的影响

Fig. 10 Performance of optimal value with varied candidates number

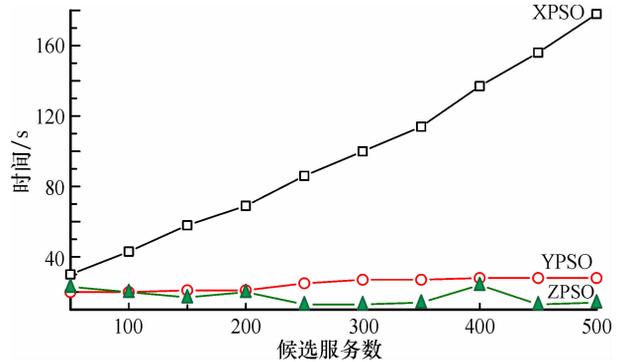


图11 候选服务数对执行时间的影响

Fig. 11 Performance of time with varied candidates number

图12、图13测试了初始粒子个数对最优值的影响。可以看出:随着粒子个数的增加,寻优效果缓慢增加,时间开销接近线性增加水平。对比可见,本文算法的寻优效果一直占优,时间开销也比较好的。

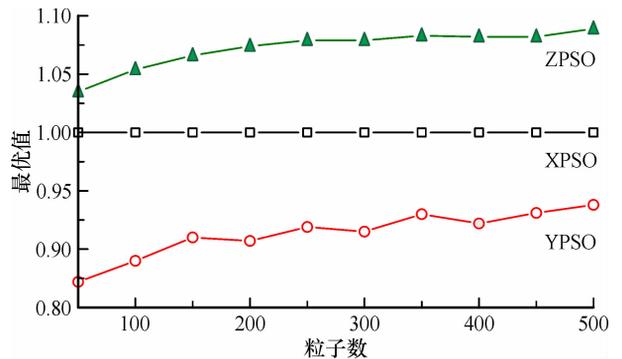


图12 粒子数对最优值的影响

Fig. 12 Performance of optimal value with varied particle numbers

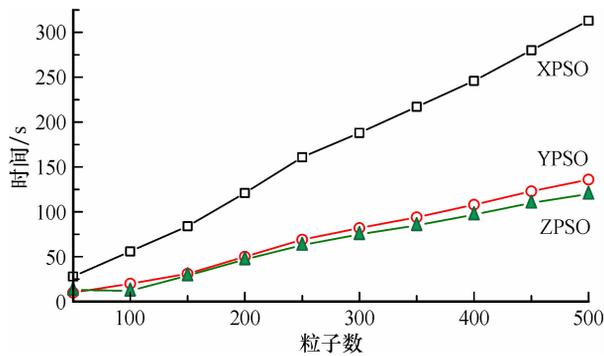


图 13 粒子数对执行时间的影响

Fig. 13 Performance of time with varied particle numbers

以上实验结果表明, 当选择合适的多样性阈值和  $dfr$  时, 本文提出的算法在时间开销与寻优效果上都有一定优势。但这些参数值与具体问题相关, 如何自适应调节这些参数需要进一步探索。

## 5 结束语

本文提出的基于粒子群优化算法的服务组合算法, 在种群初始化、认识与社会系数调整、种群多样性修补、提前终止条件选择等方面对 PSO 算法进行相应的改进。仿真实验验证了改进算法的优越性。

## 参考文献:

- [1] 刘彬, 张仁津. 基于 QoS 多目标优化的 Web 服务组合方法 [J]. 计算机工程与设计, 2012, 33 (3): 885-889.
- [2] Luo Y S, Qi Y, Hou D, et al. A novel heuristic algorithm for QoS-aware end-to-end service composition [J]. Computer Communications, 2011, 34 (9): 1137-1144.
- [3] 黎辛晓, 叶恒舟. 基于贪婪策略的局部优化服务组合方法 [J]. 桂林理工大学学报, 2012, 32 (2): 271-275.
- [4] Paganelli F, Ambra T, Parlanti D, et al. A semantic-driven integer programming approach for QoS-aware dynamic service composition [C] //Proceedings of Federation of Telecommunications Engineers of the European Community. Palermo, Italy: IEEE, 2011: 1-6.
- [5] Gabrel V, Manouvrier M, Murat C. Web services composition: complexity and models [J]. Discrete Applied Mathematics, 2015, 196 (2): 100-114.
- [6] Gupta I K, Kumar J, Rai P. Optimization to quality-of-service-driven web service composition using modified genetic algorithm [C] //Proceedings of Computer, Communication and Control, 2015 International Conference. Indore, India: IEEE, 2015: 1-6.
- [7] Li S, Yan Z, Jian L R, et al. Study on auto parts suppliers composition selection based on adaptive genetic algorithm

- [C] //Proceedings of 2015 IEEE International Conference on Grey Systems and Intelligent Services. Leicester, Britain: IEEE, 2015: 521-527.
- [8] Cao B Q, Liu J X, Liu X Q, et al. CHC-TSCM: a trustworthy service composition method based on an improved CHC genetic algorithm [J]. China Communications, 2013, 10 (12): 77-91.
- [9] Ludwig S A. Applying particle swarm optimization to quality-of-service-driven Web service composition [C] //Proceedings of 2012 IEEE 26th International Conference on Advanced Information Networking and Applications. Fukuoka, Japan: IEEE, 2012: 613-620.
- [10] 刘莉平, 陈志刚, 刘爱心. 基于粒子群算法的 Web 服务组合研究 [J]. 计算机工程, 2008, 34 (5): 104-106, 112.
- [11] 胡珀, 娄渊胜. 改进粒子群优化算法在服务组合中的应用 [J]. 计算机工程, 2011, 37 (17): 130-132.
- [12] Kang G S, Liu J X, Tang M D, et al. An effective dynamic Web service selection strategy with global optimal QoS based on particle swarm optimization algorithm [C] //Proceedings of the Parallel and Distributed Processing Symposium Workshops & PhD Forum, 2012 IEEE 26th International. Shanghai, China: IEEE, 2012: 2280-2285.
- [13] 温涛, 盛国军, 郭权, 等. 基于改进粒子群算法的 Web 服务组合 [J]. 计算机学报, 2013, 36 (5): 1031-1046.
- [14] 张燕平, 荆紫慧, 张以文, 等. 基于离散粒子群算法的动态 Web 服务组合 [J]. 计算机科学, 2015, 42 (6): 71-75.
- [15] 徐涛, 王新环. 基于多目标粒子群优化算法的 Web 服务组合 [J]. 计算机工程与设计, 2010, 31 (18): 4076-4081.
- [16] Eberhart R, Kennedy J. A new optimizer using particle swarm theory [C] //Proceedings of the 7th International Symposium on Micro Machine and Human Science. Piscataway: IEEE Service Center, 1995: 39-43.
- [17] 潘峰, 李位星, 高琪, 等. 粒子群优化算法与多目标优化 [M]. 北京: 北京理工大学出版社, 2013: 16-19.
- [18] Shi Y H, Eberhart R. A modified particle swarm optimizer [C] //Proceedings of Evolutionary Computation Proceedings, The 1998 IEEE International Conference, IEEE World Congress on Computational Intelligence. IEEE, 1998: 69-73.
- [19] 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, 47 (1): 147-156.
- [20] 崔志华, 曾建潮. 微粒群优化算法 [M]. 北京: 科学出版社, 2011: 25-30.
- [21] Ratnaweera A, Halgamuge S K, Watson H C. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients [J]. IEEE Transactions on Evolutionary Computation, 2004, 8 (3): 240-255.

## Web services composition based on modified particle swarm optimization

LU Xiang-peng, YE Heng-zhou

(Guangxi University Key Laboratory of Embedded Technology and Intelligent Information Processing, Guilin University of Technology, Guilin 541006, China)

**Abstract:** It is an NP problem to generate the composition service fast and dynamically from a large number of candidate services under the users functional and non-functional requirements. Particle Swarm Optimization (PSO) is an effective approach to solve this kind of problem. There is some deficiency partly or entirely existing in services composition approaches based on PSO. It can only supporting specific workflow or the workflow included only sequence pattern. It can only use the number of iterations as the termination condition; not avoiding the premature convergence problem or the result of that being unsatisfactory. This paper proposes a web service composition method based on modified particle swarm optimization which supports sequence, parallel and choice patterns. The main improvements of our PSO are: adding a few initial particles generated by local optimization strategy; adjusting the setting proposal of cognitive factors and social factors; proposing the concept of diversity in the population and presenting a diversity of healing mechanisms to avoid premature convergence; setting an early termination condition of the iterative operations. The simulation experiments demonstrated the advantages of the proposal in time cost and method optimization.

**Key words:** particle swarm optimization; web services composition; premature convergence; early termination condition