

# 基于动态粒子群算法的工作流服务主体优选方法

陈 鹏<sup>1,2</sup> 何 涛<sup>2,3</sup>

(华中科技大学计算机学院 武汉 430074)<sup>1</sup> (湖北省电力公司信息通信分公司 武汉 430077)<sup>2</sup>  
(武汉大学电气工程学院 武汉 430072)<sup>3</sup>

**摘 要** 在研究工作流服务时间-费用双重优化问题的基础上,提出一种基于动态粒子群算法的工作流服务主体优选方法。通过区域划分,在每个粒子所在区域内,当适应值小于最佳适应值时,对区域重新进行初始化,从而使算法具有更强的全局收敛性和动态的自适应性;同时引入随机扰动、回退等算子,将搜索范围扩大到整个解空间以大大提高获得最优解的概率。结合动态粒子群算法建立工作流调度问题的目标模型,并从跨时间粒度、跨时区、跨工作时间 3 个方面对工作流服务主体优选方法进行了讨论分析。实验结果表明,该方法比其他应用工作流调度的算法具有更短的执行时间和费用,具有更高的效率、更好的优越性。

**关键词** 动态粒子群,工作流,调度,遗传算法

中图法分类号 TP183 文献标识码 A

## Optimization Method of Workflow Service Subject Based on Dynamic Particle Swarm Algorithm

CHEN Peng<sup>1,2</sup> HE Tao<sup>2,3</sup>

(College of Computer Science, Huazhong University of Science and Technology, Wuhan 430074, China)<sup>1</sup>

(Information & Communication Branch of Hubei epc, Wuhan 430077, China)<sup>2</sup>

(College of Electrical Engineering, Wuhan University, Wuhan 430072, China)<sup>3</sup>

**Abstract** Research on workflow business hours - cost optimization problem. This paper proposed a novel dynamic particle swarm algorithm optimization method of workflow service subject. Through regional division, in each of the particles located in the region, when adapting to a value less than the best fitness, reinitializes region, so that the algorithm has better global convergence and dynamic adaptability, while the introduction of random disturbance, reverse operator, the search scope are expanded to the entire solution space in order to greatly improve the optimal solution probability. Combined with dynamic particle swarm algorithm based grid workflow scheduling problem of target model, and from three aspects of the cross time granularity, across time zones, the across working system, this paper discussed the workflow service subject selection method. The experimental results show that this method than other applications of grid workflow scheduling algorithm has shorter execution time and cost, higher efficiency, better superiority.

**Keywords** Dynamic particle swarm, Workflow, Scheduling, Genetic algorithm

## 1 引言

随着办公自动化和网络的普及,现代企业的信息系统多为跨平台、多系统的应用环境。工作流技术能够全部或部分自动化和协同工作,有效提高工作效率。工作流服务调度就是为活动选择并分配合适的资源,不仅要以最少的时间完工,而且要考虑运行费用。工作流服务时间-费用的优化问题实质上是一个复杂的 NP-Hard 问题,目前在项目管理方面已经得到广泛的研究。文献[1]详细介绍了分值界限法、动态规划法等求解方法;文献[2]提出了基于不同约束的独立任务调度启发式算法;文献[3]指出在服务资源多、并行任务同时执行的情况下,DAG(Directed Acyclic Graph)的时间-费用双重优化问题可归结为项目管理中的 DTCTP 问题;文献[4]提出了

截止期分解策略,将截止期分解为局部截止期,通过局部优化得到全局最优解;文献[5]提出遗传算法求解工作流调度方法。这些方法都是针对元任务的,没有考虑现代工作流服务的跨组织、跨地域而引起的跨时区、跨工作体制等问题,因此不能满足复杂的时区和因果约束条件的工作流服务调度需要。

本文在研究工作流服务时间-费用双重优化问题的基础上,提出了一种基于动态粒子群算法的工作流服务主体优选方法。通过对粒子群进行区域划分,每个粒子在所在区域内,当适应值小于最佳适应值时,重新进行区域划分,因此使算法具有更强的全局收敛性和动态的自适应性,同时引入随机扰动、回退等算子,将搜索范围扩大到整个解空间以大大提高获得最优解的概率。利用动态粒子群算法建立网格工作流调度

到稿日期:2012-02-13 返修日期:2012-06-28 本文受国家自然科学基金项目(9140C1101061001)资助。

陈 鹏(1982—),男,硕士,工程师,主要研究方向为电力系统自动化、计算机安全,E-mail:Chenpeng\_wuhan@163.com;何 涛(1977—),男,学士,高级工程师,主要研究方向为电力通信。

问题的目标模型,并从跨时间粒度、跨时区、跨工作时间 3 个方面对工作流服务主体优选方法进行了讨论分析。实验结果表明,该方法比其他应用网格工作流调度的算法具有更短的执行时间和费用,具有更高的效率、更好的优越性。

## 2 工作流模型

工作流(Workflow)是业务流程的计算机化或自动化,是为有效处理工作中的具有固定模式的日常活动而提出的一个概念,把工作活动定义为良好的任务、角色、规则和过程进行执行和监控,从而提高生产组织水平和工作效率。业务流程中的活动及关系可以转化为有向无环图 DAG 来描述, $G = \langle V, E \rangle$ ,  $V$  为顶点集,表示活动任务集合; $E$  为边集,表示活动之间的约束关系。

工作流总体费用计算公式:

$$C_{total} = \min \sum_{i \in V} \sum_{1 \leq k \leq l(i)} y_{ik} c_{ij} \quad (1)$$

$$\begin{cases} \sum_{k=1}^{l(i)} y_{ik} = 1, \forall i \in V \\ f_i \leq f_j - \sum_{k=1}^{l(i)} t_{ik} y_{ik}, \forall (i, j) \in E \\ f_n \leq \delta_n \\ y_{ik} \in \{0, 1\} \end{cases} \quad (2)$$

式中,当第  $i$  个任务选择服务序列中第  $k$  个服务时,  $y_{ik}$  为 1, 否则  $y_{ik}$  为 0。  $f_i$  是第  $i$  个任务的完成时间,  $f_n$  是整个工作流的完成时间,  $C_{total}$  是工作流最优的费用值,  $\sum_{k=1}^{l(i)} y_{ik} = 1$  表示每次只能选择一个服务,  $f_i \leq f_j - \sum_{k=1}^{l(i)} t_{ik} y_{ik}$  表示选择要满足偏序约束关系,  $\delta_n$  为工作截止期。

## 3 动态粒子群算法

### 3.1 粒子群算法原理

粒子群算法(Particle Swarm Optimization, PSO)是美国心理学家 James Kennedy 和电气工程师 Russell Eberhart 共同提出的一种群体智能优化算法<sup>[6-8]</sup>。该算法是从鸟群觅食的过程得到启发,利用群体中粒子的合作和竞争产生最优解的优化算法。粒子群算法数学模型:设粒子群规模为  $N$ , 目标搜索空间为  $D$  维,第  $i$  个粒子的  $D$  维空间位置向量为  $x_i = (x_{i1}, x_{i2}, \dots, x_{ik}, \dots, x_{iD})$ , 每个  $x_i$  表示解空间中一个潜在的可行解,可根据目标函数计算出的适应值来判断其是否为最优解。第  $i$  个粒子的  $D$  维空间速度向量为  $v_i = (v_{i1}, v_{i2}, \dots, v_{ik}, \dots, v_{iD})$ , 第  $i$  个粒子个体最优位置  $P_i = (P_{i1}, P_{i2}, \dots, P_{ik}, \dots, P_{iD})$ , 粒子群群体最优位置  $Q_i = (Q_{i1}, Q_{i2}, \dots, Q_{ik}, \dots, Q_{iD})$ , 迭代公式如下:

$$P_{i,t+1}^k = \begin{cases} x_{i,t+1}^k, & f(x_{i,t+1}) < f(P_{i,t}) \\ P_{i,t}^k, & \text{else} \end{cases} \quad (3)$$

$$v_{i,t+1}^k = \omega v_{i,t}^k + c_1 r_1 (P_{i,t}^k - x_{i,t}^k) + c_2 r_2 (Q_{i,t}^k - x_{i,t}^k) \quad (4)$$

$$x_{i,t+1}^k = x_{i,t}^k + v_{i,t+1}^k \quad (5)$$

式中,  $c_1$  为认知加速因子,可调整个体最优粒子方向飞行的步长,  $c_2$  为社会加速因子,可调整全局最优粒子方向飞行的步长。  $r_1, r_2$  是  $[0, 1]$  之间的随机数,加入随机性以扩大搜索空间。  $v_{i,t}^k$  和  $x_{i,t}^k$  分别表示第  $i$  个粒子在第  $t$  次迭代中第  $k$  维

的速度和位置。  $P_{i,t}^k$  和  $Q_{i,t}^k$  分别表示第  $i$  个粒子在第  $t$  次迭代中第  $k$  维的个体和全局极值点位置。  $\omega$  为惯性因子,可以控制搜索速度,快速收敛到局部,然后再局部搜索得到最后解。惯性因子  $\omega$  更新公式:

$$\omega = \omega_{max} - (\omega_{max} - \omega_{min}) t / T_{max} \quad (6)$$

### 3.2 动态区域划分

假设粒子  $X = \{x_1, x_2, \dots, x_n\}$ , 以粒子为中心,划分半径为  $R$  的区域,因为粒子之间的欧氏距离不同,所以这些区域有可能重叠。每个粒子不仅代表自身,也代表其所在的区域。按照此方法,如果群体数量为  $m$ , 就可以把全局区域划分  $m$  个不同的子区域。  $x_i = (x_{i1}, x_{i2}, \dots, x_{ik}, \dots, x_{iD})$ , 其中  $D$  代表维数,  $n$  代表粒子数。如果  $D=1$ , 划分的区域是一维的,每个区域是一段线段,这些线段允许重叠;如果  $n=2$ , 划分的区域是二维的并且每个区域是一个圆,允许重叠。

每个粒子代表一个子区域,并且也是该区域当前最优解,即  $P_{best}$ 。为得到区域最优解,使每个粒子在其区域内进化并更新区域最优解,进化规则如下:

$$x_i = N(x_i, R) \quad (7)$$

式中,  $N(x_i, R)$  为高斯分布随机函数,如果  $x_i$  跳出区域范围,则重新进行进化计算,将其限制在区域之内。半径  $R$  的设置不能太大或太小,太大则区域重叠严重,而太小则个体区域不能覆盖全部的空间区域,易陷入局部最优,而找不到全局最优解。因此设置  $R$  为动态变化量:

$$R = R * \exp\left(\frac{1}{\sqrt{D}} * N(0, 1)\right) \quad (8)$$

式中,  $D$  为维数,  $N(0, 1)$  是 0 到 1 之间的随机数。

### 3.3 引入随机扰动和回退

如果对每一维向量  $x_{ij}$  ( $j=1, \dots, n$ ) 都引入随机扰动因子,则会破坏个体最优解信息和全局最优解信息,从而降低解的质量。因此引入随机扰动因子,采用随机挑选并引入极值扰动,既可以使粒子脱离局部极值,又不会因为极值扰动破坏局部最优值信息,扩大了粒子搜索的路径和区域,使其能更好地发现最优解。扰动因子公式如下:

$$e_1 = \begin{cases} r_1^{t_0} > T_0, & \text{if } U(0, 1) \geq 0.5 \\ 1, & \text{else} \end{cases} \quad (9)$$

$$e_2 = \begin{cases} r_2^{t_g} > T_g, & \text{if } U(0, 1) \geq 0.5 \\ 1, & \text{else} \end{cases} \quad (10)$$

进化机制具有随机性,因此粒子求解可能会出现偏差,无法找到最优解的位置。在此使用回退因子,如果在某一时刻粒子最优解没有得到改进或者出现倒退的状况,则粒子需要在自身位置的基础上向后回退,即原来的加法变成减法。这样粒子会以更大的概率寻找到最优解。需要将式(4)、式(5)改为:

$$\begin{aligned} v_{i,t+1}^k &= \omega v_{i,t}^k + c_1 r_1 (e_1 P_{i,t}^k - x_{i,t}^k) + c_2 r_2 (e_2 Q_{i,t}^k - x_{i,t}^k) \\ x_{i,t+1}^k &= x_{i,t}^k \pm v_{i,t+1}^k \end{aligned} \quad (11)$$

## 4 基于粒子群算法的工作流服务主体优选方法

现代网络技术的发展形成了一个大的分布式协作系统,工作流服务也具有跨时间粒度、跨地域、跨工作体制等复杂性,且很多计算和处理具有动态性,需要并行计算。传统的解

决方法已经无法有效地解决这些问题,本文利用改进的粒子群算法从跨时间粒度、跨地域、跨工作体制 3 个方面解决 workflow 服务优选问题。

#### 4.1 关键技术分析

##### (1) 跨时间粒度

在 workflow 服务中,很多时候存在多个时间粒度,例如某个 workflow 站点任务完成时间为 2 小时 40 分 25 秒,而另一个站点任务完成时间为 4 小时 30 分,workflow 服务的开始时间是 2012 年 7 月 1 日 8 点 30 分。这里有 3 个时间粒度:时、分、秒。文献[9]提出时间粒度转换方法,包括世纪、年、月、日、时、分、秒等转换方式,例如:“2010-10-13-07:00”+10 天 20 小时=“2010-10-24-03:00”。本文按此方式先计算站点第 1 个任务所需要完成的天数及站点完成任务的日期;再计算该任务完成时间的月数和站点完成任务的月份;然后计算该任务完成时间的年份和站点完成任务的年份;依次下去,最后得到最后一个任务的完成时间。

##### (2) 跨地域

在复杂 workflow 服务中,有些站点是全球分布,存在时区问题。假设有两个连续的任务  $i$  和  $j$ ,第  $i$  个任务所在的时区为  $S$ ,第  $j$  个任务所在的时区为  $M$ ,它们对应的标准时区差分别为  $H_i$  和  $H_m$ ,则第  $j$  个任务的开始时间为:

$$Starttime(j) = Endtime(i) + H_m - H_i \quad (12)$$

##### (3) 跨工作体制

由于在 workflow 服务中,很多站点所在企业、城市、国家不同,因此上班时间、休息日、节假日也不同,工作体制就出现了多种形式。针对工作体制不同的问题,workflow 站点完成任务的时间  $CT(j)$  需要包括任务到达站点后到站点开始工作时段之间的等待时间  $Wait(j)$  和站点在执行任务过程中可能存在的休息时间  $Rest(j)$ ,则任务完成时间和开始时间为:

$$CT(j) = ET(j) + Tinput(j) + Toutput(j) + Rest(j) \quad (13)$$

$$Starttime(j) = Endtime(i) + Wait(j) \quad (14)$$

式中, $ET(j)$  为第  $j$  个任务的执行时间, $Tinput(j)$  为输入数据的时间, $Toutput(j)$  为输出数据的时间。

#### 4.2 算法步骤

依据粒子群算法的原理,设定 workflow 服务的目标函数作为适应度函数,约束条件作为惩罚函数。算法描述如下:

(1) 设置问题的任务、站点数、权值和边界条件。

(2) 初始化种群,计算每个粒子的适应度,并把初始化值作为个体最优解,来记录个体最优解中的最优值作为全局最优解。初始化迭代次数为 1,初始化区域划分。

(3) 计算每个粒子的位置和速度,超出范围的按边界取值。

(4) 进行动态区域划分,引入扰动因子和回退,更新粒子个体最优解和全局最优解。

(5) 判断是否达到迭代次数,如达到,算法停止,记录最优解,否则转向(3)。

#### 5 实例验证及分析

为验证本文算法的优越性,在 MATLAB7.0 下对 workflow

服务主体优选进行仿真实验。实验环境:软件采用 Java 编程,Windows XP 操作系统,硬件采用 Pentium IV4GHz, RAM 为 2GB 的 PC 机。粒子种群数  $N$  是站点数的 2 倍,迭代次数为 50,认知加速因子  $c_1$  和社会加速因子  $c_2$  为 2.0,动量惯性系数  $\omega$  为 0.8,区域划分半径  $R$  为 1。本文通过 DAG 生成器对 workflow 服务进行建模。实验中节点数  $|V| = \{4, 8, 12, 16, 20, 24, 28\}$ ,图的最大出度小于站点数。任务的服务站点数量采用均匀分布,取值范围在 5 到 15 之间;服务的执行时间分别取在 5 到 50、50 到 100 之间产生的随机数,workflow 费用初始化为 1000 除以 workflow 服务的执行时间,由此可以模拟不同的粒度和不同 QoS 值服务。根据每种参数的不同组合产生 30 个实例,截止期以 workflow 下界完工时间为准,每个实例按 5% 递增取 12 个不同截止期,实验结果为所有实例的平均值。

工作流的各任务的时序控制流程中包含 9 个任务,每个任务可选的执行站点数为 1 到  $n$  个。在编程实验过程中,可以依次设定  $n=5, 10, 15, 20, 25, 30$ , 分别进行实验。为使实验结果具备一般性,在编程实验中,本文设置标准时区为 2012 年 6 月 5 日 14 点 30 分。采用随机函数产生其他各参数,分别采用本文改进的粒子群算法、遗传算法进行了仿真实验。其中改进的粒子群算法中微粒群规模、学习因子、权重等参数的设置值和遗传算法的参数设置相同。采用上面的阶梯式站点数,对两种算法分别运行 50 次,其实验结果如图 1 所示。

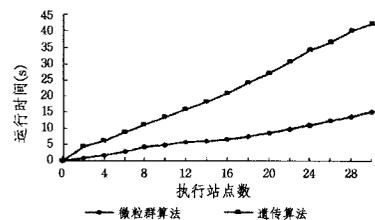


图 1 两种算法运行时间比较

从图 1 可以看出,对于具有复杂的工作流服务选择问题,本文改进的粒子群算法比遗传算法具有以下优势:对于找到最优路径具有更好的成功率,算法得到的路径费用少,运行时间较少,达到了时间-费用的双重优化,并且随着站点数的增多,改进的粒子群算法的执行效率优越性明显。再利用两种算法运行 100 次,得到其达到最优路径的次數、平均代数及平均时间,如表 1 所列。从表 1 可以看出,对于 workflow 优化调度问题,本文算法要比传统的遗传算法达到最优路径的成功率高,达到最优路径的迭代次数少,而且程序运行时间要短。

表 1 两种算法的性能比较

算法	达到最优路径次数	未达到最优路径次数	达到最优路径平均代数	达到最优路径平均时间
遗传算法	65	35	64.7	9.6
本文算法	100	0	39.5	3.2

**结束语** 本文在研究 workflow 服务时间-费用双重优化问题的基础上,提出了一种基于动态粒子群算法的 workflow 服务主体优选方法。通过对粒子群进行区域划分,每个粒子在所在区域内,当适应值小于最佳适应值时,重新进行区域划分,因此使算法具有更强的全局收敛性和动态的自适应性,同时引入随机扰动、回退等算子,将搜索范围扩大到整个解空间以

大大提高获得最优解的概率。利用动态粒子群算法建立网格 workflow 调度问题的目标模型,并从跨时间粒度、跨时区、跨工作时间 3 个方面对 workflow 服务主体优选方法进行了讨论分析。实验结果表明,该方法比其他应用网格 workflow 调度的算法具有更短的执行时间和费用,实现了时间-费用的双重优化,具有更高的效率、更好的优越性。

## 参 考 文 献

- [1] De P, Dunne E J, Ghosh J B, et al. The discrete time-cost tradeoff problem revisited[J]. *European Journal of Operational Research*, 1995, 81(2): 225-238
- [2] Buyya R, Abramson D, Giddy J, et al. Economic models for resource management and scheduling in grid computing[J]. *Concurrency and Computation: Practice and Experience Journal (Special Issue on Grid Computing Environments)*, 2002, 14(13-15): 1507-1542
- [3] Lin M, Lin Z X. A cost-effective critical path approach for service priority selections in grid computing economy[J]. *Decision Support Systems*, 2006, 42(3): 1628-1640

(上接第 170 页)

- [11] Chapelle O, Zien A. Semi-supervised classification by low density separation[C]//*Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics*. Barbados, 2005: 57-64
- [12] Zhu X, Ghahramani Z, Lafferty J. Semi-supervised learning using Gaussian fields and harmonic functions[C]//*Proceedings of the 20th International Conference on Machine Learning*. Washington DC, USA, 2003: 912-919
- [13] Zhou D, Bousquet O, Lal T N, et al. Learning with local and global consistency[C]//*Thrun S, Saul L, Scholkopf B, eds. Advances in Neural Information Processing Systems 16*. Cambridge, MA: MIT Press, 2004: 321-328
- [14] Blum A, Mitchell T. Combining labeled and unlabeled data with co-training[C]//*Proceedings of the 11th Annual Conference on Computational Learning Theory*. Madison, Wisconsin, USA, 1998: 92-100
- [15] Goldman S, Zhou Y. Enhancing supervised learning with unlabeled data[C]//*Proceedings of the 17th International Conference on Machine Learning*. Stanford, CA, USA, 2000: 327-334
- [16] Zhou Z H, Li M. Tri-training: Exploiting unlabeled data using three classifiers[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2005, 17(11): 1529-1541
- [17] Li M, Zhou Z H. Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples[J]. *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, 2007, 37(6): 1088-1098
- [18] Settles B. Active learning literature survey [R]. *Computer Sciences Technical Report 1648*, University of Wisconsin-Madison, 2009
- [19] Abe N, Mamitsuka H. Query learning strategies using boosting and bagging[C]//*Proceedings of the 15th International Conference*

- [4] Yu J, Buyya R, Tham C K. Cost-based scheduling of workflow applications on utility grids[C]//*Proceedings of the 1st IEEE International Conference on e-Science and Grid Computing*. Melbourne, Australia, 2005
- [5] 郭文彩, 杨扬. 基于遗传算法的网格服务 workflow 调度的研究[J]. *计算机应用*, 2006, 26(1): 54-56
- [6] Robinson J, Ragnat-Samii Y. Particle swarm optimization in electromagnetics[J]. *IEEE Transaction Antennas Propag*, 2004, 52(2): 397-407
- [7] Huang T, Mohan A S. A hybrid boundary condition for robust particle swarm optimization [J]. *Antennas and Wireless Propagation Letters*, 2005, 4(1): 112-117
- [8] Mikki S, Kishk. An improved particle swarm optimization technique using hard boundary conditions[J]. *Microwave Opt Technol Lett Sep*, 2005, 46(5): 422-426
- [9] Liu J X, Zhou C J. Research on the Workday Model in Business Service Grid Environment[C]//*Proc. of the 2005 International Workshop on Workflow Management Systems in Grid Environment*. Changsha: IEEE Publisher, 2006
- [10] Mitchell T. *Machine Learning*. Madison, Wisconsin, USA, 1998: 1-9
- [20] Seung H, Opper M, Sompolinsky H. Query by committee[C]//*Proceedings of the 5th Annual Conference on Computational Learning Theory*. Pittsburgh, PA, 1992: 287-294
- [21] Muslea I, Minton S, Knoblock C A. Active + semi-supervised learning = robust multi-view learning[C]//*Proceedings of the 19th International Conference on Machine Learning*. Sydney, Australia, 2002: 435-442
- [22] Zhu X, Lafferty J, Ghahramani Z. Combining active learning and semi-supervised learning using Gaussian fields and harmonic functions[C]//*Proceedings of ICML 2003 workshop on The Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining*. Washington DC, 2003
- [23] Zhou Z H, Chen K J, Dai H B. Enhancing relevance feedback in image retrieval using unlabeled data[J]. *ACM Transactions on Information Systems*, 2006, 24(2): 219-244
- [24] Xu J M, Fumera G, Roli F, et al. Training SpamAssassin with active semi-supervised learning [C] // *Proceedings of the 6th Conference on Email and Anti-Spam*, Mountain View, CA, 2009
- [25] Breiman L. Random forests[J]. *Machine Learning*, 2001, 45(1): 5-32
- [26] Li M, Zhang H, Wu R, et al. Sample-based software defect prediction with active and semi-supervised learning[J]. *Automated Software Engineering*, 2012, 19(2): 201-230
- [27] Quinlan J R. *C4. 5: programs for machine learning*[Z]. Morgan Kaufmann, San Francisco, CA, USA, 1993
- [28] Cortes C, Vapnik V. Support-vector networks [J]. *Machine Learning*, 1995, 20(3): 273-297
- [29] Witten I H, Frank E. *Data Mining: Practical machine learning tools and techniques with Java implementations*[Z]. Morgan Kaufmann, San Francisco, CA, USA, 2000