

基于移动 Agent 的网格计算资源管理模型设计

曾正军¹,舒万能²

ZENG Zheng-jun¹,SHU Wan-neng²

1.东华理工大学 核工程技术学院,江西 抚州 344000

2.中南民族大学 计算机科学学院,武汉 430074

1.College of Nuclear Engineering and Technology,East China Institute of Technology,Fuzhou,Jiangxi 344000,China

2.College of Computer Science,South-Central University for Nationalities,Wuhan 430074,China

E-mail:zhjzeng@ecit.edu.cn

ZENG Zheng-jun,SHU Wan-neng.Grid resource management model based on mobile agent.Computer Engineering and Applications,2008,44(21):138-141.

Abstract: Grid computing is a hot research area in distributed and parallel computing field.Resources in grid are of distributive, heterogeneous, autonomous and dynamic, thus managing the resources is complex work. This paper presents an agent-based model of the resource management and uses the hybrid genetic simulated annealing algorithm(HGSAA) in the scheduling policy, which serves the need of the grid resource management scheduler well. Simulation experiment is given to show the modeling procedures and efficiency of the HGSAA.

Key words: grid computing; mobile agent; resource management; task scheduling; hybrid genetic simulated annealing algorithm

摘 要: 网格计算是当前高性能计算领域的一个研究热点,由于网格计算的资源具有分布性,异构性,自治性,动态性等特点,因而其资源管理比一般系统的资源管理具有更大的复杂度。文中提出了一种基于 Agent 的网格计算资源管理模型,并且采用了混合遗传模拟退火算法作为调度策略,满足了网格对调度系统可扩展性和全局最优调度的需求。实验结果证实了该方法的有效性。

关键词: 网格计算; 移动 Agent; 资源管理; 任务调度; 混合遗传模拟退火算法

DOI:10.3778/j.issn.1002-8331.2008.21.038 **文章编号:**1002-8331(2008)21-0138-04 **文献标识码:**A **中图分类号:**TP391

1 引言

传统的任务调度和资源管理方法已经不能很好地适应分布式系统,特别是那些用来进行大吞吐量计算的分布式系统。问题在于异构的资源使统一的分配算法很难适用,资源的分布,归属等一系列问题导致需要很多的分配策略。对于上述问题,需要开发和应用一种灵活的、通用的方法在分布式系统中进行资源管理和任务调度。近几年兴起的网格计算(Grid Computing)技术就是基于这种需要应运而生的^[1]。然而,在网格计算这样一个动态复杂环境下,资源是分散的、异构的、动态变化的,其性能还不完善,没有得到广泛的运用。因此如何有效地管理资源、任务调度和性能监控是影响网格计算是否成功的最重要因素之一^[2]。

基于上述背景,将移动 Agent 技术应用于网格计算资源管理。移动 Agent 技术是分布式技术与 Agent 相结合的产物,它除了具有智能 Agent 的最基本特性还具有移动能力、可靠性和安全性^[3-5]。移动 Agent 能在异构网络中自主地从一台主机迁移到另一台主机,并可与其它 Agent 或资源交互的程序,其独特的对象传递思想和卓越的特性给分布式计算乃至开发系统带来了巨大的革新^[6]。由此可见,移动 Agent 技术在复杂的网格计算中具有明显优势。

2 基于移动 Agent 的网格资源管理模型设计

在网格计算中,资源和资源管理是两个非常广泛的概念,资源包括硬件资源(计算机,存储器,网络基础设施等)、软件资源(虚拟服务)和数据等。资源管理则是资源消费者在使用资源的整个生命周期中识别需求、匹配资源、分配资源、调度和监视资源的一个过程。网格资源管理涉及到 4 个角色:资源消费者、资源调度器、网格中间件和网格资源提供者。下面将分析 Globus 中资源管理结构的分层调度模型。

2.1 Globus 中资源管理分层模型

网格资源管理结构的分层模型主要包括被动和主动两部分构件:被动构件主要有资源,任务,工作,调度;主动构件包括调度器,信息服务,域控制代理,执行代理,用户,监管器,工作控制代理^[7]。

当一个用户提交一个工作给工作控制代理时,该代理会调度许可控制代理,许可控制代理和网格信息系统通信,检查工作的资源需求是否能得到满足,决定是否提交给调度器,调度器使用网格信息服务发现资源并和域控制代理协商确定目前资源的状态和可用性,然后做出影射决策并将之传给执行代理,执行代理和资源所在域的域控制代理协商使用资源,有时

基金项目:国家自然科学基金(the National Natural Science Foundation of China under Grant No.61572137)。

作者简介:曾正军(1974-),男,讲师,研究方向:计算机应用;舒万能(1981-),男,硕士,研究方向:网格计算,演化计算。

收稿日期:2008-04-30 **修回日期:**2008-06-03

可能需要预制资源。监视器监视工作的进程,如果性能比预期要低,则决定是否要重新调度^[8]。图1是分层调度模型。

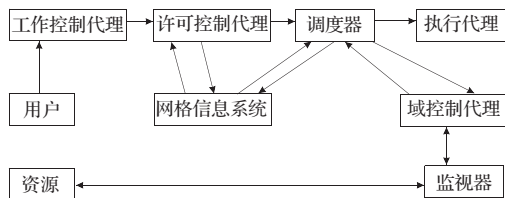


图1 Globus 的分层调度模型

2.2 基于移动 Agent 的网格资源管理模型

现有的 Globus 网格环境着重处理了资源发现和标识问题,而资源的分配和任务的调度方法都不完善,其中资源的分配没有考虑到资源的动态负载变化,任务的调度采用轮循方法,容错性差,没有考虑负载均衡^[9,10]。针对现有网格资源管理系统的缺点,采用了一种基于 Agent 的层次模型:每个本地的网格资源节点均被封装成为一个 Agent,这样,所有分布于网格系统中不同地方的网格资源节点(比如一个集群)就构成了底层的 Agent 系统,它们为基于网格的应用程序提供高性能计算能力。基于移动 Agent 的网格计算资源管理模型如图2所示。

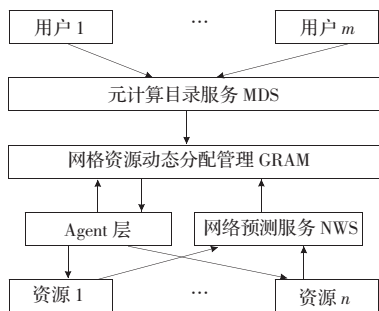


图2 基于移动 Agent 的网格计算资源管理模型

基于移动 Agent 的网格计算资源管理模型包括四大模块:元计算目录服务 MDS(Meta-computing Directory Service, MDS),网络资源动态分配管理 GRAM(Globus Resource Allocation Manager, GRAM),网络预测服务 NWS(Network Weather Server, NWS),Agent 层。

(1)元计算目录服务 MDS:MDS 是网格计算中信息中心,相当于一个智能 Agent 群。其主要功能是完成对网格计算环境中的信息的发现、注册、查询、修改等工作,提供对网格计算资源一个真实、实时的动态反映,同时能极大地减少网络拥挤,提高搜索速度。

(2)网络资源动态分配管理 GRAM:GRAM 其主要任务包括两方面:一是处理资源描述,根据预测确定 Agent 如何分配到各个资源上;另一方面是把任务请求分解,并将分解的子任务发送到各个相关的 Agent 上进行任务调度。

(3)网络预测服务 NWS:NWS 是一个设计用于跟踪现有资源和网络状况的分布式监控系统,也能够提供短期的网络性能预测,它将收集到的各资源节点当前负载信息,并实时反馈给 GRAM。

(4)Agent 层:在 Agent 层,每个单独的 Agent,将收到由 GRAM 分发的子任务队列,然后继续采用本地的进化调度策略让子任务在本地集群上进行最优化执行。

3 基于混合遗传模拟退火算法的移动 Agent 任务调度策略

在开放的网格计算复杂环境下,任务动态地加入和退出,使得系统的可利用资源动态变化,以及任务的执行时间往往是时变的,这一切都说明我们需要自适应地调节对网格计算资源的分配,合理高效地调度移动 Agent,以保证其高效稳定运行^[11,12]。

基于遗传算法任务调度策略,携带算法的移动 Agent 在搜索过程中自动获取和积累有关搜索空间的知识,并且自适应地控制搜索过程,不断引导算法向更小的空间去寻找最优解或满意解。一方面可以极大地减少寻找最优解或可行解的开销,另一方面它所找到的解执行任务时开销也较小,能够充分提高整个系统的效率^[13,14]。

遗传算法 GA(Genetic Algorithm, GA)采用了生物进化论的思想,通过自然选择和适者生存的竞争策略来求解优化问题^[15,16]。虽然 GA 有较强的全局搜索性能,但它在实际应用中容易产生早熟收敛的问题,而且进化后期搜索效率较低。模拟退火 SA(Simulated Annealing, SA),起源于统计物理学方法,并首次被 Kirkpatrick 等引入优化问题的求解。SA 算法具有很好的局部搜索能力,但对参数的依赖性较强。混合遗传模拟退火算法 HGSSA(Hybrid Genetic Simulated Annealing Algorithm, HGSSA)是将 GA 和 SA 算法相结合而构成的一种优化算法。GA 的局部搜索能力较差,但把握搜索过程总体的能力较强;而 SA 算法搜索总体的能力较差,但把握局部搜索的能力较强^[17]。HGSSA 算法将 GA 算法和 SA 算法的优点充分结合起来,有效地提高了算法的求解效率和问题的求解质量。

3.1 基于移动 Agent 的任务调度建模

在资源动态变化的资源管理系统中,为了便于分析问题,下面给出该问题的一般性定义:

定义1 定义 $A=[A_1, A_2, \dots, A_n]$ 表示网格中资源 Agent 的集合, $T=[T_1, T_2, \dots, T_m]$ 表示网格中任务的集合。其中, A_j 表示第 j 个资源 Agent, T_i 为第 i 个子任务。

定义2 定义任务分配矩阵 $D=[d_{ij}]_{m \times n}$, 表示 T 到 A 的映射关系,其中 $d_{ij} = \begin{cases} 1 & \text{表示任务 } t_i \text{ 分配到 Agent } A_j \\ 0 & \text{表示任务 } t_i \text{ 未分配到 Agent } A_j \end{cases}$ 。

定义3 假定资源 Agent A_j 当前的 CPU 利用率,内存利用率,当前网络流量,磁盘 I/O 访问率,进程总数等负载参数分别用 $C_j\%$, $M_j\%$, N_j , $IO_j\%$, P_j 表示,则 A_j 动态负载权值可以表示为:

$l_j = \pi_1 \times C_j\% + \pi_2 \times M_j\% + \pi_3 \times N_j + \pi_4 \times IO_j\% + \pi_5 \times P_j$ 。其中, $\sum \pi_j = 1$, π_j 反映各个负载参数的重要程度, $j=1, 2, \dots, n$ 。

定义4 定义 $F(X)$ 为子任务在各资源 Agent 上运行的优化分配函数。

$$F(X) = \sum_{i=1}^m \sum_{j=1}^n \xi_i \times d_{ij} \times e^{-l_j}$$

3.2 编码及群体初始化

基因编码表示必须能唯一地表示搜索空间中的所有搜索节点。在 HGSSA 算法中,由分配矩阵 $D=[d_{ij}]_{m \times n}$ 的定义可以分析出矩阵 D 满足以下两个特征:(1)每行有且仅有一列的元素值为 1,其他的为 0;(2) D 中所有元素和为 m ,即 $\sum_{i=1}^m \sum_{j=1}^n d_{ij} = m$ 。对分配矩阵 $D=[d_{ij}]_{m \times n}$ 采用自然数编码方式,把矩阵 D 每行中元素 1 所在列号作为基因,各基因之间是相互独立的,基因编

号为 K_1, K_2, \dots, K_n , 其中 $K_j \in [1, m], j \in [1, n]$, 且为可重复相等的自然数。

采用随机分配的方法产生适量个体组成的初始群体为了能够取得全局最优解, 群体要有一定的规模。具体方法: 随机产生 $popsize$ 个位数为 n 的个体, 然后用自然数编码的染色体串作为初始种群。

3.3 适应度函数

适应度函数是 HGSAA 算法搜索进化过程中用来评价个体的优劣, 并作为以后遗传操作的依据, 它是对优良染色体的定量描述。模型的目标函数为求最小值问题, 需要将其转化为求最大值的适应度函数。适应度函数可以表示为:

$$f(x) = \frac{1}{F(x)}$$

3.4 选择算子

选择是将父代的个体信息传递到子代。每代中的每一个个体按照适应度函数的大小决定它能够复制到下一代的概率。通过选择, 使得群体中的优秀个体数目不断增加, 整个进化过程朝着更优解的方向进行。体现了“适者生存, 优胜劣汰”的进化原则。采取“轮盘赌”式的选择策略:

(1) 计算所有个体的选择概率:

$$P(i) = \frac{f(i)}{\sum_{i=1}^{popsize} f(i)}$$

(2) 随机生成一个数 $r = \text{random}[0, 1]$;

(3) 若 $P(0) + P(1) + \dots + P(i-1) < r < P(0) + P(1) + \dots + P(i)$, 则第 i 个个体被选择到下一代。

3.5 交叉算子

交叉是指把两个父代个体的部分结构加以替换重组而生成新个体的操作。本文定义的交叉算子如下:

(1) 在 $[0, 1]$ 内随机生成一个数 $r_c = \text{random}[0, 1]$, 若 $r_c < P_c$, 则按下式进行交叉操作: $V_1' = r_c \times V_1 + (1-r_c) \times V_2, V_2' = r_c \times V_2 + (1-r_c) \times V_1$ 。 V_1, V_2 分别是父个体, V_1', V_2' 分别为子个体。

(2) 计算 V_1' 和 V_2' 的适应度 $f(V_1'), f(V_2')$, 若 $\min\{1, \exp(-(f(V') - f(V))/T_k)\} > \text{random}[0, 1]$, 则接受新解。 T_k 为第 k 次进化的温度。

3.6 变异算子

变异操作的作用是在种群出现局部收敛时通过变异算子的突变, 使整个种群能保持一定的多样性。本文定义的变异算子如下:

(1) 在 $[0, 1]$ 内随机生成一个数 $r_m = \text{random}[0, 1]$, 若 $r_m < P_m$, 则用随机方法产生一个正整数 $w = [\text{random}[1, m]]$, 然后用 P 替换个体 V_i 中第 w 位对应的基因值 K_w 。

(2) 按照交叉算子步骤(2)的方法决定是否接受变异后的解。

3.7 终止条件

当冷却流程的温度 T 不能再降低时, 退火过程自然停止。

3.8 算法的求解过程

- (1) 初始化控制参数: $popsize, T_0, k=0$;
- (2) 根据 3.2 方式对基因进行编码, 并随机产生初始种群 p_0 ;
- (3) 评价群体 p_k 中每个个体的适应函数值 $f(x_i), i=1, 2, \dots, popsize$;

- (4) 执行选择操作, 随机地选取父代种群 F_k ;
- (5) 对 F_k 进行交叉操作, 产生杂交种群 C_k ;
- (6) 对 C_k 进行变异操作, 得到中间种群 M_k ;
- (7) 由父代种群和中间种群共同组成新的种群 $p_{k+1} = F_k \cup M_k$;
- (8) 当满足终止条件时, 退火过程自然结束, 输出最优解;

否则, $T_{k+1} = T_k \times (1 - \frac{k}{popsize}), k=k+1$, 返回到步骤(3)。

4 仿真实验与结果分析

基于上述网格资源调度算法 HGSAA 和 Agent 资源管理模型, 设计了算法的仿真实验。算法中用到的主要参数有: 群体规模 $popsize$ 为 100, 交叉概率 P_c 为 0.85, 变异概率 P_m 为 0.1, 初始温度 $T_0=0$ 。

表 1 是各算法收敛时得到的最优解的完成时间和收敛时的进化代数, 以及相对误差。为了评价算法的性能, 定义 E_m 为相对误差, 衡量算法对问题的最佳优化度, 其值越小意味着算法的优化性能越好:

$$E_m = \frac{C_1 - C^*}{C^*} \times 100\%$$

式中, C_1 为算法运行所得的实际优化值; C^* 为问题的期望最优值。

表 1 仿真比较结果

资源 Agent 个数	任务 个数	完成时间/s			收敛时的进化代数			相对误差 $E_m/\%$		
		GA 算法	SA 算法	HGSAA 算法	GA 算法	SA 算法	HGSAA 算法	GA 算法	SA 算法	HGSAA 算法
2	20	275	280	261	128	134	120	0.32	0.42	0.01
	30	382	398	360	250	254	213	0.57	0.45	0
3	30	339	332	298	162	187	157	0	0.12	0
	50	653	667	573	369	370	248	0.26	0.26	0.02
4	40	453	450	397	215	221	189	0.15	0	0
	60	598	587	526	307	345	251	0.54	0.26	0.01
5	50	546	546	465	375	375	265	0.37	0	0
	100	821	825	730	524	587	413	0	0.02	0

从表 1 中可以看出, 相对 GA 算法, SA 算法, 本文 HGSAA 算法能找到更好的解, 并且相对误差和波动率都较小。主要原因是对适应度函数进行线性尺度变换, 并对杂交和变异算子取值采取自适应方式, 有效避免 GA 的早熟收敛现象, 同时在进化的过程中, 融入退火的思想, 使算法逐渐向最优解搜索。

图 3 为 Agent 资源在 HGSAA 算法部署前的负载统计情况。图 4 为该算法部署后, Agent 资源的负载统计情况。从图 3 可以看出, 在网格计算资源优化调度模型部署前, 资源 3 由于任务过多导致资源负载过重, 以致于部分网格用户无法继续提

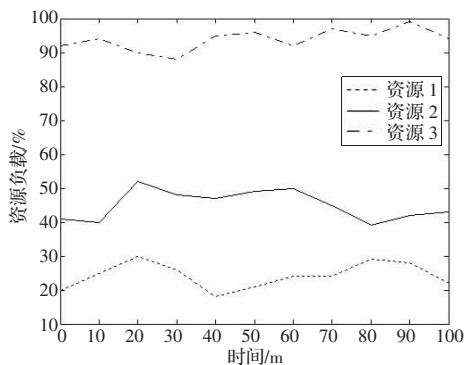


图 3 HGSAA 算法部署前资源的负载统计

交任务,而资源 1 和资源 2 的负载过低,资源没有得到充分地利用。从图 4 可以看出,系统部署后,虽然资源 3 的负载有所降低,但是资源 1 和资源 2 的负载有了很大的提高,从整体上做到了对资源的负载均衡。

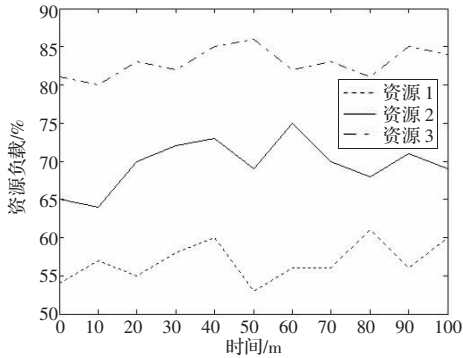


图 4 HGSAA 算法部署后资源的负载统计

5 结论

本文设计了一种基于移动 Agent 网格计算资源管理模型,能够实现网格计算资源的高效动态自适应管理。在模型基础上,采用混合遗传模拟退火算法,对网格计算环境下移动 Agent 任务调度问题展开了深入的研究,同时在调度策略上把握住了网格资源管理的启发性本质,使得系统能够进行全局最优调度。文中提出的网格计算资源管理模型与移动 Agent 技术相结合,为网格计算领域的研究提供新的研究工具和方法,对网格技术的应用与发展具有深刻推动意义。

参考文献:

- [1] Foster I, Kesselman C, Tuecke S. The anatomy of the grid: enabling scalable virtual organizations[J]. International J Supercomputer Applications, 2001, 15(3).
- [2] Foster I, Kesselman C, Lee C, et al. A distributed resource management architecture that supports advance reservations and co-allocation[C]//Intl Workshop on Quality of Service, 1999.
- [3] Karnik N M, Tripathi A R. Design issues in mobile agent programming systems[J]. IEEE Concurrency, 1998, 6(3): 52-61.
- [4] Wilhelm U G, Staamann S M, Buttyan L A. A pessimistic approach to trust in mobile agent platforms[J]. IEEE Computing, 2000, 9(10):

(上接 112 页)

同时,在 K-L 第六主成分下提取出了错分入沼泽的行道树,初步分析原因认为:在该地区,由于到田间的行道树较为稀疏,在遥感影响上相当一部分是离散的点,容易被误以为是噪声,于是可以利用集中了较多噪声的 K-L 变换的第 6 主成分来找回这些树木信息。

参考文献:

- [1] Wani M M, Choubey V K, Himanshu J. Quantification of suspended

(上接 115 页)

- [6] Law N L, Szeto K Y. Adaptive Genetic Algorithm with mutation and crossover matrices[C]//IJCAI-07, 2007: 2330-2333.
- [7] 金吾伦,郭元林.国外复杂性科学的研究进展[J].学术博览,2003(6).
- [8] 常青,钟民先.基于耗散结构的改进遗传算法求取红外图像二维团

40-48.

- [5] Kraus S, Sycara K, Evenchik A. Reaching agreement through argumentation: a logical model and implementation[J]. Artificial Intelligence, 1998, 104(1,2): 1-69.
- [6] Schoonderwoerd R, Holland O, Bruten J. Ant-like agents for load balancing in telecommunications networks[C]//Proceedings of the First International Conference on Autonomous Agents, 1997: 209-216.
- [7] Yao Junhan. Resource scheduling algorithm for grid computing and its modeling and analysis using Petri net[C]//Shanghai: The 2nd International Workshop on Grid and Cooperative Computing, 2003.
- [8] Dai Yuan-Shun, Wang Xiao-Long. Optimal resource allocation on grid systems for maximizing service reliability using a genetic algorithm[J]. Reliability Engineering and System Safety, 2006, 91: 1071-1082.
- [9] Weng Chuliang, Lu Xinda. A cost-based on-line scheduling algorithm for job assignment on computational grids[M]. Heidelberg: Springer Verlag, 2003.
- [10] Martino V D, Mililotti M. Sub-optimal scheduling in a grid using genetic algorithm[J]. Parallel Computing, 2004, 30(5/6): 553-565.
- [11] 颖峰. 基于进化算法的网格计算资源管理调度系统[J]. 计算机工程, 2003, 29(15): 110-175.
- [12] 魏伟, 李艳玮, 郑伟勇. 基于移动 Agent 动态自适应网格资源的管理模型[J]. 华东交通大学学报, 2006, 23(5): 109-111.
- [13] 李春林, 卢正鼎, 李蜡元. 基于 Agent 的计算网格资源管理[J]. 武汉理工大学学报: 交通科学与工程版, 2003, 27(1): 10.
- [14] 汝传, 韩光法, 陈宏伟. 网格计算环境下资源管理优化策略研究[J]. 通信学报, 2005, 26(7): 21-26.
- [15] Zomaya A Y, Yee-Hwei. The observations on using genetic algorithms for dynamic load-balancing[J]. IEEE Transactions on Parallel and Distributed Systems, 2001, 12(9).
- [16] Yao Wensheng. Genetic scheduling on minimal processing elements in the grid[M]. Heidelberg: Springer-Verlag, 2002.
- [17] Shu Wanneng, Zheng Shijue, Gao Li, et al. An improved genetic simulated annealing algorithm applied to task scheduling in grid computing[J]. Dynamics of Continuous Discrete and Impulsive Systems-series B-applications & algorithms, 2006: 831-835.
- [18] 张讲社, 徐宗本, 梁怡. 整体退火遗传算法及其收敛充要条件[J]. 中国科学: E 辑, 1997, 27(2): 154-164.
- [19] 王霞, 周国标. 整体退火遗传算法的几乎处处强收敛性[J]. 应用数学, 2003, 16(3): 1-7.

solids in Dal lake, srinagar using remote sensing technology[J]. International Journal of Remote Sensing, 17(7): 1425-1432.

- [2] 衣伟宏, 杨柳, 张正祥. 基于 ETM+ 影像的扎龙湿地遥感分类研究[J]. 湿地科学, 2004: 208-212.
- [3] 李爽, 丁圣彦, 许叔明. 遥感影像分类方法的研究[J]. 河海大学学报, 2002, 32(2): 70-73.
- [4] 王建业. 江苏沿海滩涂资源利用的新方向[J]. 海洋通报, 1999(18): 56-62.
- [5] Carpenter D J, Carpenter S M. Modeling inland water quality using Landsat data[J]. Remote Sensing of Environment, 1983, 13: 345-352.

值[J]. 华东理工大学学报: 自然科学版, 2005, 31(5).

- [9] Simoes A, Costa E. An evolutionary approach to the zero/one Knapsack problem: testing ideas from biology[C]//ICANNGA'2001, 2001: 22-25.