

网格调度中具有价格激励的改进蚁群算法^①

An Improved Ant Colony Algorithm Based on Price Competition in Grid Scheduling

兰 静 黄文明 张 阳 (桂林电子科技大学 计算机与控制学院 广西 桂林 541004)

摘要: 为了进一步提高网格资源的利用率, 本文提出了一种具有价格激励的改进蚁群算法, 并将其应用于网格资源调度中, 通过增加动态价格因子, 实现了资源提供者和消费者各自的经济利益, 并且能够有效地解决资源调度的各种问题和提高网格系统的性能。

关键词: 网格资源调度 价格激励 蚁群算法 价格因子 经济利益

网格^[1]技术是近年来国际上兴起的一种新的计算模式, 它利用高速互联网把分布于各个地理位置的动态资源连成整体, 为用户提供一体化信息服务。资源调度是连接网格底层和高层功能的纽带, 是协调整个网格系统有效运转的中枢。一个高效的资源调度策略对于提高网格系统性能尤为重要。

1 引言

本文首先提出了一种具有价格激励的改进蚁群算法, 此改进算法根据蚁群算法基本原理, 引入了价格因子, 刺激资源提供者提供动态的资源价格和资源消费者作出合理的预算, 在一定程度上增强了算法的搜索能力和系统的负载均衡, 缩短了网格调度的总执行时间。然后设计了基于改进蚁群算法的资源调度策略, 根据资源价格高低对资源进行调度, 按照优先级高低对任务进行调度, 并对失效资源节点上的任务进行迁移和重新调度, 提高了网格的资源利用率和系统的容错性能。

2 蚁群算法

蚁群算法^[2]是由 M.Dorige 等人于 1991 年首先提出来的一种源于生物世界的仿生类方法。蚂蚁是群居昆虫, 互相协作的一群蚂蚁很容易找到从蚁穴到食物源的最短路径。根据蚂蚁这一特性, 蚁群算法利用与环境的动态交互获得反馈信息, 采用逐步收敛的方

式搜索问题的最优解。蚁群算法的发现为各个领域解决诸多复杂的优化组合问题提供了有力的工具, 它主要应用于诸如旅行商问题、动态网络路由问题、分配问题、图形着色等 NP 难问题的求解, 网格资源调度本质上也是 NP 难问题, 因此蚁群算法非常适合于解决资源调度问题。

2.1 蚁群算法基本原理

经过对蚁群的觅食行为的研究发现, 蚂蚁在觅食的过程中能够在其所经过的路径上留下一种叫做信息素(pheromone)的化学物质, 它们通过这种化学物质相互传递信息。蚂蚁一般总倾向于走那些信息素浓度高的路径。这样, 某一路径上通过的蚂蚁数目越多, 留下的信息素浓度就越高, 则后续蚂蚁选择该路径的概率就越大。而其他路径上的信息素会随着时间的推移慢慢挥发, 这就形成了一种信息正反馈现象, 使得最终所有的蚂蚁都走蚁穴到食物源的最短路径。

如图 1 所示, 假设蚁穴用 A 表示, 食物源用 E 表示。有甲、乙两只蚂蚁从蚁穴 A 出发寻找食物源 E, 它们分别沿 ABE 和 ACDE 两条路径进行觅食, 其中 ABE 长为 1, ACDE 长为 1.5, 当发现了食物源 E 后, 蚂蚁甲和蚂蚁乙又会沿原路返回, 同时它们还会在经过的路径上留下信息素。如果两只蚂蚁的速度一样, 由于 ABE 的路径较短, 当蚂蚁甲返回蚁穴 A 时, ABE 上留下了两次信息素。而 ACDE 的路径较长, 蚂蚁乙只走到了 D 点, 则 ACD 路径上只留下了一次信息素。

① 基金项目: 广西研究生创新计划项目(2008105950812M428); 广西教育厅项目(2004(20)); 桂林电子科技大学 06 年度学科软环境项目
收稿时间: 2008-08-26

这样蚂蚁乙留下的信息素浓度低于蚂蚁甲的。而后续的蚂蚁根据信息素浓度的高低进行选择路径，即它们总是选择信息素浓度高的路径 ABE 进行觅食，如此循环下去，这条路径 ABE 就是蚁穴 A 与食物源 E 之间的最短路径。

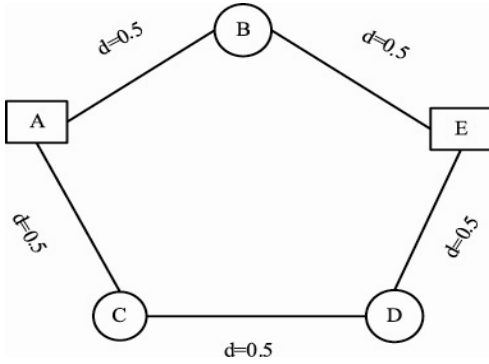


图 1 蚁群算法示意图

2.2 蚁群算法的优缺点

研究表明，蚁群算法具有分布并行性、鲁棒性、可扩展性等优点。

①分布并行性：蚁群在觅食的过程中，如果仅是靠单个蚂蚁来完成任务，因为单个蚂蚁的智力和体力极其有限，就容易收敛于局部最优，但是多个个体之间通过不断的进行信息交流和传递，有利于收敛于解空间的某一子集，蚁群算法这种同时构造问题的多个解体现了算法的分布并行性。

②鲁棒性：蚁群算法不会因为单个蚂蚁寻找到较差的解或因为问题空间发生变化而失去原有的作用。

③可扩展性：此优点体现在蚁群可在原来的最优解基础上快速找到已新增了计算资源的问题的最优解。

虽然蚁群算法拥有许多优点，但是同样也存在着一些缺陷。

①根据对蚁群算法的复杂度分析，该算法需要较长的搜索时间，这在一定程度上影响了网格系统的总执行时间的降低。

②当蚁群搜索进行到一定程度后，所有个体所发现的解趋于一致，就不能对解空间进一步进行搜索，容易出现了停滞现象，使算法的收敛速度变慢，不利于算法发现更好的解。

③由于蚁群算法总是依赖于路径上的信息素浓度来选择路径，在网格资源调度过程中会造成过多的任

务都迁移到一个最优资源节点上，产生资源负载不均衡现象。

3 改进的蚁群算法介绍

本文通过对基本蚁群算法的改进，引入价格因子，动态调整资源提供者对资源设定的价格，资源消费者则可以在自己的预算范围内选择合适的资源来完成任务，进一步提高蚁群算法的性能和资源的利用率，为实现高效、安全的网络资源调度打下基础。

3.1 资源价格因子的引入

在网格资源调度中，资源提供者为消费者提供计算资源，并从中获利；而资源消费者为满足自己的资源需求需要作出一定的预算。为了保证资源获得充分利用，同时还要使资源消费者能以最小的代价完成任务，本文引入了一个资源价格因子 T_{pr} ，即每个节点的单位价格。它不是一个固定的常数，而是与任务被成功完成的次数有关的动态函数。初始的单位价格 $T_{pr}(0)$ 是系统预先定义的，当任务分配给资源时

$$T_{pr}^{new} = \begin{cases} \lambda_s T_{pr}^{old} \\ \lambda_s = T_s / T_a \end{cases} \quad (1)$$

其中， λ_s 表示价格调整系数， T_s 表示节点成功完成的任务数， T_a 表示该节点已经接受的所有任务总数。当分配给资源的任务被完成的数量较多时，资源提供的单位价格也较高；当分配给资源的任务被完成的数量较少时，单位价格则较低。通过此价格激励机制，促使了资源提供者提供的资源价格和资源消费者的预算趋于合理化。

3.2 蚁群算法的改进思想

3.2.1 信息素初始化

在任务分配给资源之前，必须初始化网格系统中各个资源节点的信息素。本文选用资源所包含的处理器个数 n 和其处理能力 p (MIPS)、通信能力 b (Mb/s) 以及节点的单位价格 T_{pr} (元/M) 作为资源的初始信息素，如公式 2 所示：

$$T_k(0) = aT_p(0) + bT_b(0) + cT_{pr}(0) \quad (a + b + c = 1) \quad (2)$$

其中， a 、 b 、 c 分别表示资源节点的处理能力信息素、通信能力信息素和单位价格信息素在整个初始信息素中所占的比重。

3.2.2 路径选择规则

系统初始化完毕后，根据各资源节点上的信息素，利用公式 3 计算出节点分配到任务的概率，此概率随

着资源信息素的改变而改变，是蚂蚁选择下一个节点的参考值。

$$P_k(t) = \begin{cases} \frac{[T_k(t)]^{\alpha} \times [\eta_k]^{\beta}}{\sum_u \{ [T_u(t)]^{\alpha} \times [\eta_u]^{\beta} \}} & (k \text{ 和 } u \text{ 为未访问节点}) \\ 0 & (\text{其他}) \end{cases} \quad (3)$$

其中， $P_k(t)$ 表示节点 k 选择下一个节点的概率， $T_k(t)$ 表示计算节点 k 在 t 时刻的信息素浓度，是一个启发因子，它表示节点 k 的固有属性，参数 α 和 β 用于调节 α 和 β 之间的权重，当 α 变小时，则算法收敛速度较快，当 β 变小时，算法收敛速度较慢。

3.2.3 局部更新机制

当任务被分配到某个资源节点上时，资源的信息素会随之发生改变，我们可以根据蚁群经过路径所耗资源量，利用信息素局部更新机制对节点进行更新，如公式 4 所示：

$$T_k^{new} = \begin{cases} \rho T_k^{old} + \Delta T_k & (\rho \text{ 是信息素的持久性}) \\ \Delta T_k = -K & (K \text{ 是任务运算量和通信量}) \end{cases} \quad (4)$$

信息素局部更新机制可以适当的减少路径上的信息素，有利于蚁群探索不同的路径，增加解的多样性，增大获得最优解的概率。

3.2.4 全局更新机制

当资源节点完成任务并返回时，进行资源信息素全局更新，如公式 5 所示：

$$T_k^{new} = \rho T_k^{old} + \Delta T_k \quad (5)$$

当任务成功返回时， $\Delta T_k = C_e \times K$ ，其中 C_e 是奖励因子，表示成功经验增加，一般取值为 0.6。当任务返回失败时， $\Delta T_k = C_p \times K$ ，其中 C_p 是惩罚因子，表示失败经验增加，一般取值为 -1.2。

4 网格资源调度流程

基于改进蚁群算法的网格资源调度流程如下：

①资源消费者提交任务，并插入到任务队列中等待调度。同时系统为每一个任务设置优先级，按照优先级的高低进行分配任务，原则是优先级高的任务先调度。优先级随着任务的长度变化而变化。长度越长的任务的优先级越高，长度越短的任务的优先级越低。

②利用公式(2)初始化网格各个资源节点的初始信息素。

③根据资源 ID 号查询资源当前状态信息，返回资源的单位价格，并根据资源单位价格对资源进行升序

排序。

④调度器从任务队列中取出优先级高的任务，并且根据公式(3)计算出的概率，把任务分配到对应的资源节点上。若该资源节点的单位价格在资源消费者提供的预算范围内，则此任务正常分配给资源，若该资源节点的单位价格超出了资源消费者提供的预算范围，则此任务分配失败，重新插入任务队列，等待新的调度。

⑤当任务正常分配给资源时，利用公式 4 进行局部信息素更新。

⑥任务分配后，等待结果返回。若任务成功完成，利用公式 1 对节点的单位价格进行更新，并采用公式(5)： $T_k^{new} = \rho T_k^{old} + \Delta T_k$ (其中 $\Delta T_k = C_e \times K$) 进行节点信息素全局更新。

⑦若任务执行失败，则将任务重新插入任务队列中，等待调度器的重新分配，并利用公式 1 对该节点的单位价格进行更新。同时按照公式(5)：对信息素进行实时更新。

⑧重复③-⑧步骤，直到所有任务都被成功执行。

5 实验结果

在 GridSim 仿真平台中，随机生成 4 个资源和 4 个资源消费者进行模拟实验。资源情况如表 1 所示：

表 1 资源情况

ID	处理器 个数(n)	处理能力 (MIPS)	通信能力 (Mb/s)	单位价格 (元/M)
1	2	380	20	1
2	4	377	30	2
3	4	380	40	2
4	4	515	20	1

每个资源消费者包含的任务数大约为 20 个，每个任务的平均长度大概 1000MI。图 2 显示了每个资源消费者拥有的任务被完成的情况。

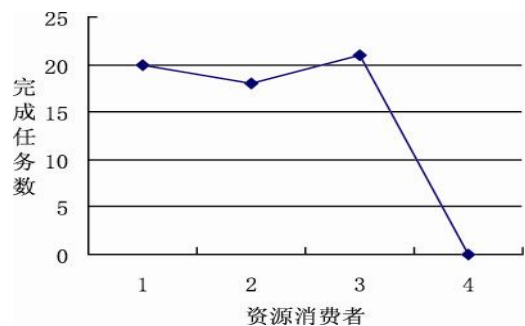


图 2 仿真结果

由图2可知,资源消费者4由于设置的预算较低,在竞争中处于劣势,很难得到资源;而预算较高的消费者则能够较早的完成任务,处于竞争的优势地位。

6 总结

针对网格资源的复杂特点,本文提出的改进蚁群算法综合考虑了网格系统中各个资源节点的参数和任务的优先级别,通过不同的信息素更新规则,保障了资源节点上的负载均衡和调度效率。同时算法可以根据成功完成的任务数量动态的调整资源的价格,避免了资源提供者提出不合理的价格和资源消费者作出不合理的预算,让双方在交易时获得更高的满意度,体现了一定的价格灵活性。将经济学上的价格竞争^[6]引入到网格资源调度策略中,为提高系统资源利用率提供了一个极具创新意义的思路。下一步研究的重点是将改进的蚁群算法与其他算法相结合,进一步增强算法收敛的能力和网格系统性能。

参考文献

- 1 都志辉,陈渝,刘鹏.网格计算.北京:清华大学出版社,2002.
- 2 Dorigo M, Caro GD, Gambardella LM. Ant algorithms for discrete optimization. *Artificial Life*, 1999,5(3):137-172.
- 3 须文波,张涛.网格计算资源调度算法研究.计算机工程,2006,(14).
- 4 段海滨.蚁群算法原理及其应用.北京:科学出版社,2005:35-36.
- 5 Merkel D, Middendorf M, Schmeck H. Ant colony optimization for resource-constrained project scheduling. *IEEE Transactions on Evolutionary Computation*, 2002.
- 6 Buyya R. Economic-based distributed resource management and scheduling for grid computing. Melbourne, Australia: Monash University, 2002.
- 7 Qi XG, Liang ZY. Ant colony algorithm based resource allocation and task scheduling of grid. *Journal of Guangxi University for Nationalities: Natural Science Edition*, 2006(2):123-125.
- 8 Buyya R, Vazhkudai S. Compute power market: Towards a market-oriented grid. 1st IEEE/ACM International Symposium on Cluster Computing and the Grid(CCGRID 2001). Los Alamitos, California, USA: IEEE Computer Society Press, 2001.
- 9 舒勤,刘一萌.基于 GridSim 的网格资源调度算法研究[博士论文]成都:四川大学,2004.