基于遗传编程的网格资源调度算法

李 钧,王忠群,刘 涛
(安徽工程科技学院 计算机与科学工程系,安徽 芜湖 241000)

摘 要:网格资源调度是一个非常重要的研究课题。由于因特网的开放、动态性,传统的资源调度和分配方法已经不再适用网格计算,基于经济模型的资源管理和调度成为研究热点。在计算市场模型中,构造有效的效益函数又是提高算法性能的关键。有关文献中采用的是线性效益函数,虽然降低了复杂度,但不能很好地反映用户的效益。文中提出了基于遗传编程来寻找和构造非线性效益函数的方法,并将其应用到网格调度算法中。实验结果表明该算法可以提高网格中的资源调度性能。

关键词:网格;遗传编程;调度
中图分类号:TP393 **文献标识码**:A **文章编号**:1673-629X(2008)02-0129-04

Grid Resource Scheduling Algorithm Based on Genetic Programming

LI Jun, WANG Zhong-qun, LIU Tao
(Dept. of Computer Science & Technology, Anhui University of Technology and Science, Wuhu 241000, China)

Abstract: Management and scheduling is a very important problem in grid computing. Because of openness and dynamics of Internet traditional resource management and scheduling algorithms have no longer been valid in grid computing, and now resource management and scheduling based on economic model is a research hotspot. In computational market model, constitution of benefit function is key to improvement of scheduling algorithm. The relevant literatures present linear benefit function, it reduces complexity, but it can not reflect benefit of users. This paper proposes a method for finding and constructing benefit function, based on genetic programming, and it is used in grid scheduling algorithm. The experimental results show that this algorithm can improve performance of grid resource scheduling.

Key words: grid; genetic programming; scheduling

0 引 言

随着 Internet 的迅猛发展,网格计算日益成为一个重要的研究领域。网格通过互联网将空间分散的、系统异构的计算资源组合起来[1,2]。这些资源涵盖了PC机、工作站、集群以及超级计算机,具有异构性、动态性、分布性的特点[3],这决定了网格资源的管理和调度是一个非常复杂的 NP 问题。

R.Buyya 提出了运用经济模型来解决资源管理问题[4~6],该模型将资源使用者作为市场模型中的消费者,资源所有者作为市场模型中的生产者。资源使用者以一定的价格向资源所有者购买资源,而资源所有者根据自己的计算能力和提供服务的稳定性,以不同价格出售自己计算资源,利用价格和价格浮动反映资源的稳定性和供需状况,让各个参与者通过叫价与还价来达成交易,由市场机制来实现资源的有效配置[7]。在计算经济模型中,作为消费者的使用者和作为生产者的资源提供者,都要通过效益函数来衡量自己交易的得失,从而指导下一次交易策略的选择。文中提出了一个新的效益函数的构造方法,与以往的线性效益函数[8,9]不同的是,该方法利用遗传编程技术来寻找非线性的效益函数,通过更为有效的构造效益函数来提高资源调度的性能。

1 遗传编程

遗传编程[10]是在遗传算法基础之上发展而来,借鉴了自然界生物进化理论和遗传的原理,采用了层式编码方式自动随机产生程序的方法。遗传编程区别于遗传算法不再以 0 或 1 来构成个体,而是以树型结构的程序来表示,如图 1 所示。

在该树型结构中包含两种节点:终节点集合[10](Terminal Set):包含程序中可能出现的值的集合,如

收稿日期:2007-05-29
基金项目:安徽省自然科学基金资助项目(070412058);安徽教育厅自然科学基金重点项目(2006KJ016A,2005KJ065)
作者简介:李 钧(1979-),男,安徽芜湖人,硕士研究生,助教,研究方向为分布式计算、网格计算等;王忠群,教授,研究方向为软件工程、分布式计算、工作流技术等。

图 1 中的 X;操作节点[10](Function Set):包含对终节点进行操作的运算子,如图 1 中的 +,*。

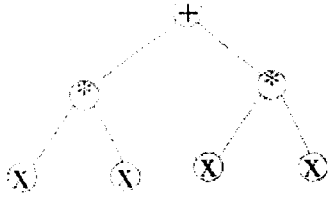

图 1 遗传编程中的程序结构

遗传编程首先从终节点集合(Terminal Set)和操作节点(Function Set)中随机选取节点产生树型结构程序的个体,但操作节点不能出现在树的叶子节点上。并以树型结构程序个体构成初始种群(generation 0),计算每个个体的适应度值;选择遗传算子,如复制、交叉、变异,依照一定的概率产生新的个体,并将新个体与旧个体进行适应度比较,选取最优,构成下一代种群,依此对种群不断进行迭代,直到某一代中个体的适应度值符合设定的要求,即找到整个搜索空间的最优解或者近似最优解[10]。与遗传算法相比,遗传程序设计是在可能空间中寻找合适的计算机程序的自适应搜索算法。其搜索过程从本质上属于随机搜索算法,其空间遍历性比遗传算法要好,并可随机跳跃到不同的子空间,从而在全局空间中从若干不同搜索路径以并行方式逼近问题的解。

遗传算法的交叉、变异是对编码串进行操作,而在遗传编程中遗传算子的操作涉及到树型结构的操作,其有着不同的实现方式[10]:

①交叉:根据锦标赛、轮盘赌等算法,从当前种群中选取两个父代个体;从父代个体中随机设定交叉点,拆出子树,交换两个个体的子树,形成新的两个子代个体(如图 2 所示)。

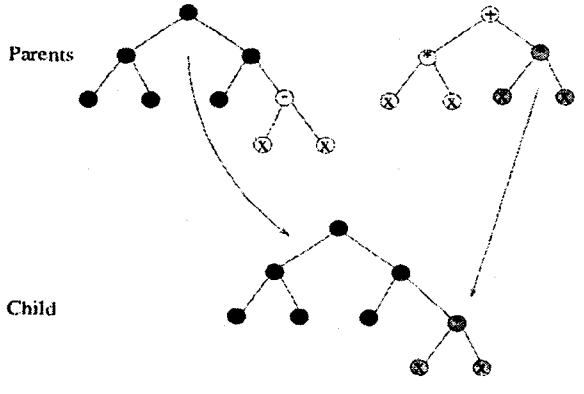

图 2 遗传编程中的交叉操作

②变异:在当前种群中随机选取一个个体,从终节点集合和操作节点中随机选取一个或若干个节点,构成子树或单个节点,并在变异点上对父代个体进行替换。

2 算法模型

2.1 问题描述与定义

任务集 $T = \{T_0, T_1, \cdots, T_{n-1}\}$,$T_i$ 为第 i 个任务;

资源集 $R = \{R_0, R_1, \cdots, R_{n-1}\}$,$R_j$ 为第 j 个资源的信息;

Deadline_i 为第 i 个资源的任务的最终期限;

Wait_j 为第 j 个资源执行下一个任务的等待时间;

Trans_{ij} 为第 i 个任务将代码和数据传输到第 j 个资源所需的时间;

$\text{Bene\_run}(T_i, R_j, \text{Deadline}_i, \text{Wait}_j)$ 表示为任务 T_i 分配到资源 R_j 上,并在任务 T_i 给定预期 Deadline_i,等待时间 Wait_j 和 Trans_{ij} 传输时间的条件约束下求的时间效益;

Budget_i 为任务 i 提出的完成计算任务的预算成本;

Peak_time_price_j 为第 j 个资源使用高峰时资源的价格;

$\text{Offpeak\_time\_price}_j$ 为第 j 个资源使用率低时的资源的价格;

$\text{Bene\_cost}(T_i, R_j, \text{Budget}_i, \text{Peak\_time\_price}_j, \text{Offpeak\_time\_price}_j)$ 表示为任务 T_i 分配到资源 R_j 上,并在任务 T_i 给定预算成本 Budget_i,高峰时资源的价格 Peak_time_price_j 和使用率低时的资源价格 $\text{Offpeak\_time\_price}_j$ 的条件约束下求的费用效益;

$\text{Bene\_qos}(T_i, R_j)$ 表示为任务 T_i 分配到资源 R_j 上时 Qos 效益。

2.2 效益函数的构造

效益函数是在多个性能指标之间建立函数关系,通过构造的效益函数评价当前调度的性能,指导下一次的资源调度以期达到性能指标的最优化。在以往的效益函数的构造过程中往往采用线性函数来构造,其一般形式为:

$$\text{Benefit}(T_i, R_j) = a * \text{Bene\_run}(T_i, R_j, \text{Deadline}_i, \text{Wait}_j) + b * \text{Bene\_cost}(T_i, R_j, \text{Budget}_i, \text{Peak\_time\_price}_j, \text{Offpeak\_time\_price}_j) + c * \text{Bene\_qos}(T_i, R_j) \quad (1)$$

在(1)式中,$a + b + c = 1, 0 \leqslant a, b, c \leqslant 1$。任务 T_i 分配到资源 R_j 上的效益函数为 $\text{Benefit}(T_i, R_j)$ 由 $\text{Bene\_run}(T_i, R_j, \text{Deadline}_i, \text{Wait}_j)$,$\text{Bene\_cost}(T_i, R_j, \text{Budget}_i, \text{Peak\_time\_price}_j, \text{Offpeak\_time\_price}_j)$ 和 $\text{Bene\_qos}(T_i, R_j)$ 加权平均得到,但时间效益、费用效益、Qos 服务质量效益三个性能指标之间并不是简单的线性关系,该效益函数不能有效地反映各个性能指

标的非线性关系。因而文中通过遗传编程技术来寻找和构造效益函数,较好地反映了时间效益、费用效益和Qos服务质量效益三个性能指标之间的非线性关系。

终节点集 Terminal Set = {time,cost,qos},其中time为 Bene_ run(T_i,R_j,Deadline$_i$,Wait$_j$)计算出的时间效益值;cost 为 Bene_ cost(T_i,R_j,Budget$_i$,Peak_ time_ price$_j$,Offpeak_ time_ price$_j$)计算出的费用效益值;qos为 Bene_ qos(T_i,R_j)计算出的服务质量效益值。

操作节点集 Function Set = {Add,Sub,Mul,Div},其中 Add,Sub,Mul,Div 为对终节点操作的运算子。其集合中运算子不能出现在树结构的叶子节点。

文中将遗传算子的交叉概率设为85%,突变概率设为10%,复制概率设为5%,迭代的代数设为50,树型结构的深度为20。适应度函数值为0表示找到最优解,则停止迭代;但如果适应度函数值不为0,迭代代数超过50或树的深度超过20,也停止迭代。其运行后,所得到的效益函数:

```
Tree (generation 37):
Mul(ADF1X2(4.000000, time), Add(qos, Div(Add(time, Div
(time,time)), cost)))[Add(ARG;0.538743, ARG;0.715719)]
[Mul(ARG;0.699453, ARG;0.924662)][Mul(4.000000, 4.
000000)]
Fitness:0.000000
Size:22
CPU time:17.397087 seconds (17310000 clock ticks)
Averaged Approximately:
1367 Individuals per second (1383 since last checked)
0.6 Generations per second
0.1 Attempts per second
```

2.3 调度算法

```
Begin
  For i=1 to max_nubmber_of_jobs
  For j=1 to max_number_of_resources
    time=Bene_run(Ti,Rj, Deadlinei,Waitj)
    cost=Bene_cost(Ti,Rj,Budgeti,Peak_time_pricej,Offpeak_
time_pricej)
    qos=Bene_qos(Ti,Rj)
    将 time、cost,qos 代入计算 Benefit(Ti,Rj)
  Endfor
  Endfor
  For 任务 Ti
    找到 fitness 为0的 Benefit(Ti Rj)所对应的资源 Rj
    将任务 Ti 分配到资源 Rj 上
    从任务队列中删除 Ti
    更新资源 Rj 的信息
  Endfor
End
```

3 实验结果与分析

3.1 实验设置

GridSim[11]是一个基于 SimJava 的事件驱动的网格仿真工具包,提供了计算经济模型的有效模拟。GridSim 通过创建资源实体对象、用户实体对象和任务实体对象来模仿实际网格中的资源、用户和任务,利用 Input 类和 Output 类完成通信,模拟任务分配和运行。

实验步骤为以下3步:

(1)创建网格资源,创建了3个具有不同 MIPS,操作系统异构的网格资源;

(2)创建用户和任务,随机产生0～100个任务,并对任务的长度、数据文件大小等属性进行了设置;

(3)以文中算法模型编写调度器实体,进行算法的模拟。

3.2 实验结果分析

文献[9]提出了如下的效益函数:

$$\text{Benefit}(T_i,R_i)=\alpha\times\text{Benefit\_ETC}(T_i,R_i)+\beta\times\text{Benefit\_Cost}(T_i,R_i) \quad (2)$$

在公式(2)中 $\alpha+\beta=1,0\leqslant\alpha,\beta\leqslant1$。将 Benefit_ ETC($T_i$,$R_i$)和 Benefit_ Cost($T_i$,$R_i$)加权平均得到效益函数 Benefit($T_i$,$R_i$)。

文献[8]利用柯布—道格拉斯函数来建立效益函数:

$$U(S,R)=\alpha\times\ln(QS)+(1-\alpha)\times\ln(R) \quad (3)$$

上述两个公式都是以简单的线性函数来描述时间效益和价格效益的关系,然而在实际的网格资源的调度中,二者存在着复杂的非线性关系。简单的线性效益函数只能近似地评价调度的性能,在任务数和资源数都增加的情况下,调度的性能会有所降低。文中通过遗传编程构造的非线性的效益函数较好反映了时间效益、价格效益和 Qos 效益的非线性关系,与公式(2)、(3)相比,更好评价了调度算法的效用,从而有效地指导了后继调度的改进,如图5所示,随着 x 轴所表示的任务数的增加,效益最优算法完成所有任务的总时间(y 轴)有比较大的增加,而文中算法在同等条件下完

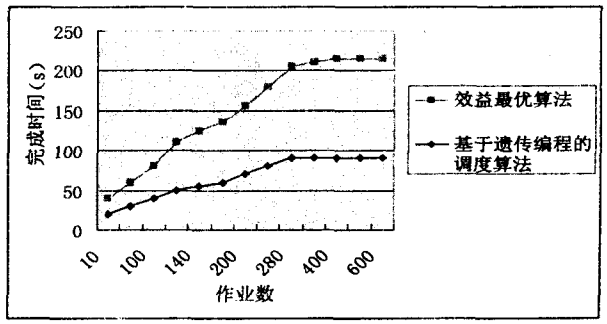

图3　调度性能比较

成任务的总时间与效益最优算法相比，有较大的降低，提高了调度性能。

4 结 论

基于经济模型的网格资源调度算法的关键是构造效益函数。效益函数构造的方便性和它的性能决定资源调度算法的优劣。借助计算经济模型的思想，提出了一种基于遗传编程改进的网格资源调度算法，利用遗传编程技术来寻找和构造的有效的效益函数，一方面提高了效益函数构造的方便性，另一方面提高了资源调度算法的性能。

参考文献：

[1] Foster I, Kesselman C. The Grid: Blueprint for a Future Computing Infrastructure[M]. USA: Morgan Kaufmann Publishers, 1999.

[2] Oram A. Peer - to - Peer: Harnessing the Power of Disruptive Technologies[M]. USA: O'Reilly Press, 2001.

[3] Parastatidis S, Watson P, Webber J. Grid Resource Specification[R]. North East Regional e - Science Centre, School of Computing Science University of Newcastle, Newcastle - upon - Tyne, NE1 7RU, United Kingdom, 2003: 2 - 3.

[4] Steiglitz K, Honig M L. A computational market model based on individual action[M]//Market - based control: a paradigm for distributed resource allocation table of contents. [s. l.]: [s. n.], 1996: 1 - 27.

[5] Buyya R, Abramson D, Giddy J. A Case for Economy Grid Architecture for Service - Oriented Grid Computing[C]//Proceedings of the International Parallel and Distributed Processing Symposium: 10th IEEE International Heterogeneous Computing Workshop (HCW 2001). San Francisco, California, USA: IEEE CS Press, 2001.

[6] Buyya R, Abramson D, Giddy J. An Economy Driven Resource Management Architecture for Global Computational Power Grids[C]//Proceedings of the 2000 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2000). Las Vegas, USA: CSREA Press, 2000.

[7] 刘一萌，舒 勤. 基于 Bargain 的经济模型的网格资源交易管理算法[J]. 计算机工程与应用，2004(17)：191 - 193.

[8] 陈冬娥，杨 扬，刘 丽. 基于效用最优的网格计算资源调度算法[J]. 计算机工程与应用，2006(2)：191 - 193.

[9] 胡自林，徐 云，毛 涛. 基于效益最优的网格资源调度[J]. 计算机工程与应用，2005(7)：69 - 70.

[10] Koza I J R. Genetic Programming: On the Programming of Computers by Means of Natural Selection[M]. Cambridge, MA: MIT Press, 1992.

[11] Buyya R, Murshed M. GridSim: A Toolkit for the Modeling and Simulation of Distributed Resource Management and Scheduling for Grid Computing[J]. The Journal of Concurrency and Computation: Practice and Experience (CCPE), 2002, 14(13 - 15): 312 - 317.

(上接第 128 页)

科级") → $D = 1$，可信度为 1。

即根据表 5 现有的数据分析，当满足年龄段在 40～50 岁之间，职称为"高级"，学历为"硕士"的教师，教学评价结果为"良"。当满足年龄段在 30～40 岁之间，学历为"硕士"，职称为"初级"或"中级"的教师，教学评价结果为"良"。当满足年龄段在 50 岁以上，职称为"高级"，学历为"本科"的教师，教学评价结果为"良"。继续对表 5 进行分析，可以得到所有的规则。

4 结 论

信息系统经过粗糙集方法[7]处理后，可以实现以下目标：

1)从信息系统中去除冗余对象，即值约简；

2)从信息系统中去除冗余属性，即属性约简；

3)得到独立属性的最小子集，同时保证与原属性的分类质量相同，即约简的属性集；

4)得到各约简集的交集——核(core)，这是最优约简集的必要元素；

5) 得到分类规则。粗糙集方法得到的分类规则一般是符号形式的显示规则，正是数据挖掘所追求的，近年来得到越来越广泛的应用。

对于不完备的信息系统，应用粗糙集理论进行数据挖掘是非常有效的，可广泛的应用多个领域[7]。

参考文献：

[1] Dunham M H. 数据挖掘教程[M]. 北京：清华大学出版社，2005.

[2] 曾黄麟. 粗集理论及其应用[M]. 重庆：重庆大学出版社，1998.

[3] 王国胤. Rough 集理论与知识获取[M]. 西安：西安交通大学出版社，2001.

[4] Pawlak Z. Rough Set Theoretical Aspects of Reasoning about Data[M]. Dordrecht: Kluwer Academic Publishers, 1991.

[5] 温有奎. 知识元挖掘[M]. 西安：西安电子科技大学出版社，2004.

[6] Pyle D. 业务建模与数据挖掘[M]. 北京：机械工业出版社，2005.

[7] 纪 滨. 粗糙集理论及进展的研究[J]. 计算机技术与发展，2007，17(3)：69 - 72.