

一种基于序贯博弈的网格资源分配策略*

李志洁^{1,2}, 程春田²⁺, 黄飞雪³, 李欣^{1,2}

¹(大连理工大学 计算机科学与工程系, 辽宁 大连 116024)

²(大连理工大学 水电与水信息研究所, 辽宁 大连 116024)

³(大连理工大学 经济系, 辽宁 大连 116024)

A Sequential Game-Based Resource Allocation Strategy in Grid Environment

LI Zhi-Jie^{1,2}, CHENG Chun-Tian²⁺, HUANG Fei-Xue³, LI Xin^{1,2}

¹(Department of Computer Science and Engineering, Dalian University of Technology, Dalian 116024, China)

²(Institute of Hydropower System and Hydroinformatics, Dalian University of Technology, Dalian 116024, China)

³(Department of Economics, Dalian University of Technology, Dalian 116024, China)

+ Corresponding author: Phn: +86-411-84708768, Fax: +86-411-84674141, E-mail: ctcheng@dlut.edu.cn, <http://www.dlut.edu.cn>

Li ZJ, Cheng CT, Huang FX, Li X. A sequential game-based resource allocation strategy in grid environment. *Journal of Software*, 2006,17(11):2373–2383. <http://www.jos.org.cn/1000-9825/17/2373.htm>

Abstract: In grid environment, resource load prediction is one of the most important problems in resource allocation optimization. But load status is difficult to estimate accurately due to the dynamic nature and heterogeneity of grid resource. In response to this issue, a resource allocation strategy that uses sequential game method to predict resource load for time optimization in a proportional resource sharing environment is proposed. The problem of multiple users bidding to compete for a common computational resource is formulated as a multi-player dynamic game. Through finding the Nash equilibrium solution of the multi-player dynamic game, resource load is predicted. Using this load information, a set of user optimal bids is produced to partition resource capacity according to proportional sharing mechanism. The experiments are performed based on the GridSim toolkits and the results show that the proposed strategy could generate reasonable user bids, reduce resource processing time, hence overcome the deficiency of Bredin's strategy, which is not concerned with resource load variation. The conclusion indicates that employing sequential game method for load prediction is feasible in grid resource allocation and adapts better to the dynamic nature of heterogeneous resource in grid environment.

Key words: grid computing; resource allocation; sequential game; proportional resource sharing; GridSim

摘要: 网格环境中资源的负载预测是实现资源优化分配的关键任务之一,而网格资源的动态性和异构性使得准确判断资源的负载状态十分困难.针对已有的分配策略对资源负载评估的不足,提出了一种基于序贯博弈的优化用户时间的网格资源分配策略.该策略将正比例资源共享的网格环境中多用户竞争同一计算资源的问题形式化为一个多人序贯博弈,通过寻求该序贯博弈中各个阶段博弈的纳什均衡解来预测资源负载;然后利用此负载信息生成所有用户的最优出价组合和资源的优化价格;最后根据各用户出价,按比例分配资源的计算能

* Supported by the National Natural Science Foundation of China under Grant No.50479055 (国家自然科学基金)

Received 2006-06-09; Accepted 2006-07-22

力.通过对网格模拟器 GridSim 的实验研究,结果表明,该策略能够得到合理的用户出价,降低资源占用时间,从而弥补了 Bredin 提出的优化策略中未考虑资源未来负载变化的缺陷,实现了资源的优化分配.其结论说明运用序贯博弈方法预测资源负载是可行的,且能更好地适应网格环境下异构资源的动态性.

关键词: 网格计算;资源分配;序贯博弈;正比例资源共享;GridSim

中图法分类号: TP393 文献标识码: A

由于现有技术无法解决虚拟组织(virtual organization)^[1]的一些问题,所以出现了“网格计算(grid computing)”^[2-4],它与传统的分布式计算的区别在于:它侧重于大规模资源共享、革新的应用以及高性能计算.但是,作为一个新兴的领域,网格计算面临着许多挑战^[5],资源管理^[6-9]就是其中一个复杂的问题.资源管理的核心目的是要在资源提供者和资源消费者之间建立一种共同协议,通过该协议,资源提供者同意向资源消费者提供执行任务的能力.网格环境下的资源管理和本地系统中资源管理的主要区别在于:网格环境所管理的资源跨越多个管理域.这种分布造成了相似的资源在进行配置和管理时所产生的异构性问题,而且网格资源的供需状况始终处于动态的变化之中.因此,许多研究^[10,11]运用经济学原理来解决网格环境中的资源管理问题.

这些研究大致可分为两类.一类是基于一般均衡(general equilibrium)^[12]的网格资源管理:在假定用户都是理性(个人效用最大化)的前提下,利用价格和价格浮动反映资源的异构性和供需状况,各个参与者通过价格自我调节,使整个网格系统协作运行.这种方法通过供需均衡实现资源的有效配置,具有代表性的学术观点和方法有:Buyya^[13]提出了一个基于经济的网格框架 GRACE(grid architecture for computational economy),将分布式计算经济应用于资源管理和应用调度;Wolski^[14]从计算经济的角度研究网格资源的分配问题,研究了商品市场模型和拍卖模型的资源分配效率;Abramson^[15]给出了一个网格资源代理,利用经济模型动态选择资源.但是,这些方法在很多情况下是理想化的,因为只考虑了资源和用户之间的关系,没有考虑用户和用户之间的相互影响.这样确定的资源均衡价格即使能达到帕累托(Pareto)最优,也难以满足实际网格环境的需要.另一类是基于纳什均衡(Nash equilibrium)^[16]的网格资源管理:侧重于多个理性利益主体的行为所产生的相互影响与作用的分析,其中个人的最优选择是其他人选择的函数,并将资源配置看成是一个博弈问题,通过寻求纳什均衡解得到资源的优化配置方案.其主要成果和方法有:Kwok^[17]提出了一个层次化网格的博弈论模型,考虑了资源的自私行为对整个网格作业执行性能的影响,但没有考虑用户的自私行为;Bredin^[18]研究了具有串行任务的多个网格用户竞争同一资源的博弈问题,提出了以预算为约束的优化作业执行时间的资源分配策略;Maheswaran^[19]提出了基于纳什均衡的拍卖可划分资源的优化用户费用的资源分配策略.然而,他们的分配策略都是基于历史的 CPU 负载信息进行用户决策,没有考虑未来资源负载的变化,不能得到合理的资源价格,因而也就无法有效地实现资源优化分配.

如何获取未来的资源负载呢?这是实现资源优化分配的关键环节.网格环境的分布性和动态性使得准确判断资源的负载状态是十分困难的,有一些负载预测方法^[20,21]基于资源性能的不变性或缓慢变化,然而这种假设没有考虑未来的负载变化,在一些情况下是不合理的;另一些方法^[22,23]基于自适应策略,但是实现起来非常复杂.为此,本文提出了一种简单而有效的负载预测方法——序贯博弈(sequential game)策略,先将正比例资源共享的网格环境中多用户竞争同一计算资源(CPU)的问题形式化为一个多人动态博弈,通过有限次序序贯博弈获取未来的资源负载信息,并形成资源的最终价格和用户的优化出价组合,实现了网格资源的动态优化分配,改善了 Bredin 策略对负载评估的不足.

1 网格中正比例资源共享的经济模型

正比例资源共享的经济模型适于管理大规模的共享资源,能够解决一个用户长期独占资源的问题,主要应用于校园网格.我们主要讨论在这种模型中网格资源的优化分配问题.在正比例资源共享模型中,用户出价相对越高,使用的资源数量相对越多.本模型中有 N 个网格用户竞争一个有限的计算资源.这些网格用户需要购买网格资源才能完成作业(一个类型各异的任务序列).因为作业由一个任务序列构成,故作业的执行时间就是所有

任务的执行时间之和.假设有 K 个类型的资源,每个用户在一个特定类型的资源上只能完成一个任务,我们定义如下参数:

- (1) $\{q_k^i\}_{k=1}^K$: 网格用户的任务序列,任务必须按顺序执行(即任务之间有数据依赖),其中, q_k^i 是第 i 个网格用户的第 k 个类型任务的大小.
- (2) c_k^i : 第 i 个网格用户为了完成 k 类型任务而选择资源的能力.
- (3) b_k^i : 第 i 个网格用户为使用 k 类型资源每秒的出价.
- (4) A_k : 网格用户集合.
- (5) B_k : 类型 k 的网格资源从 A_k 中接收的总的用户出价.
- (6) B_k^{-i} : 除第 i 个用户以外所有网格用户对第 k 个类型资源的出价集合,即 $B_k^{-i} = \sum_{j \in A_k, j \neq i} b_k^j$.

因为第 i 个网格用户分到的资源数量与整个资源的比例等于它的出价相对于所有用户出价的的比例,所以,第 i 个网格用户分到的资源数量是

$$r_k^i = c_k^i \left(\frac{b_k^i}{B_k} \right) = c_k^i \left(\frac{b_k^i}{b_k^i + B_k^{-i}} \right) \quad (1)$$

本文假定网格用户具有各种资源价格状态的完美信息,即 B_k^{-i} 已知.既然 b_k^i 不依赖于 B_k^{-i} , ($B_k^{-i} + b_k^i$) 可以替代 B_k ,那么,第 i 个网格用户为完成 k 类型任务所需的时间就是

$$t_k^i = \frac{q_k^i}{r_k^i} = \frac{q_k^i (b_k^i + B_k^{-i})}{c_k^i b_k^i} \quad (2)$$

且费用是

$$e_k^i = t_k^i \cdot b_k^i = \frac{q_k^i (b_k^i + B_k^{-i})}{c_k^i} \quad (3)$$

2 网格资源分配的动态优化策略

2.1 网格资源分配中的纳什均衡

在使用正比例共享的网格模型中,每个网格用户必须在其他用户可能的出价基础上确定自己的最优出价.也就是说,每个网格用户的出价都受到参与竞争的其他用户出价的影响.这种竞争和决策行为正是博弈论要研究的问题,其结果是一个纳什均衡,它可能不是各方及整体利益的最大化,但它是在已给定信息条件下的一种必然结果.网格用户的时间优化问题是在有限的预算 E_i 下,尽可能快地完成作业.该优化问题可以表示为

$$\min \sum_{k=1}^K t_k^i, \text{ s.t. } \sum_{k=1}^K e_k^i \leq E_i \quad (4)$$

式(4)可以用拉格朗日方法求解,其拉格朗日方程为

$$L = \sum_{k=1}^K t_k^i + \lambda \left(\sum_{k=1}^K e_k^i - E_i \right) \quad (5)$$

对方程式(5)两边取 b_k^i 的偏微分,令其等于 0,导出一个用户的任意两个任务出价 b_k^i 和 b_j^i 之间的关系

$$\frac{\partial L}{\partial b_k^i} = -\frac{q_k^i B_k^{-i}}{c_k^i (b_k^i)^2} + \lambda \frac{q_k^i}{c_k^i} = 0 \Rightarrow \lambda = \frac{B_k^{-i}}{(b_k^i)^2} \Rightarrow b_k^i = b_j^i \sqrt{B_k^{-i} / B_j^{-i}} \quad (6)$$

再对方程式(5)两边取 λ 的偏微分,令其等于 0,然后根据方程式(6),用含有 b_1^i 的表达式代替 $\{b_k^i\}_{k=2}^K$,得到

$$\frac{q_1^i}{c_1^i} (b_1^i + B_1^{-i}) + \sum_{k=1}^K \frac{q_k^i}{c_k^i} \sqrt{\frac{B_k^{-i}}{B_1^{-i}}} b_1^i + \sum_{k=1}^K \frac{q_k^i}{c_k^i} B_k^{-i} - E_i = 0 \quad (7)$$

引入下面 3 个变量:

$$\alpha_1^i = E_i - \sum_{k=1}^K \frac{q_k^i}{c_k^i} B_k^{-i}, \beta_1^i = \frac{q_1^i}{c_1^i}, \gamma_1^i = \sum_{k=1}^K \frac{q_k^i}{c_k^i} \sqrt{B_k^{-i}} \quad (8)$$

解方程式(7),得到 b_1

$$b_1^i = \frac{(\alpha_1^i - \beta_1^i B_1)^2}{2\gamma_1^{i^2}} \left(-1 + \sqrt{1 + \frac{4\gamma_1^{i^2} B_1}{(\alpha_1^i - \beta_1^i B_1)^2}} \right) \quad (9)$$

其中, $B_1 \in \left(0, \frac{\alpha_1^i}{\beta_1^i} \right)$,若 B_1 超出此区间,则 $b_1^i=0$.方程式(9)中的 b_1^i 就是第 i 个网格用户的第 1 个任务的最优出价.我们

在图 1 中给出了不同用户数量下资源价格的形成过程.竞争用户的增加上抬了资源价格,而且随着资源价格负载的上升,用户的总体出价先升高后下降,这是因为当资源价格超出用户预算时,部分用户会选择退出竞争.图 2 显示了用户出价 b 随资源负载 B 变化的曲线,以及在不同的富有程度下,用户出价的变化.如果用户的预算充裕,可供选择的出价空间就比较大.

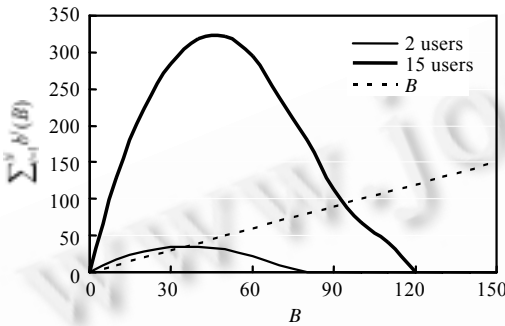


Fig.1 Searching resource equilibrium price

$$B = \sum_{i=1}^N b^i(B)$$

图 1 搜索资源均衡价格 $B = \sum_{i=1}^N b^i(B)$ 的过程

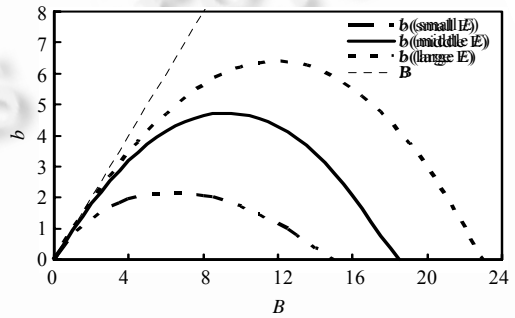


Fig.2 Comparison of user bids under different budgets

different budgets

图 2 比较不同预算条件下的用户出价

2.2 基于序贯博弈的负载预测

在第 2.1 节的资源分配策略中,资源负载是根据历史负载均值评估的,为了适应网络资源的动态性,提出了利用有限次序贯博弈预测资源负载信息的资源优化分配策略.由于采用了正比例资源共享模型,所以本文的负载是指资源的当前价格.序贯博弈使用序贯理性(sequential rationality)假设,即不论过去发生了什么,参与人应该在博弈的每一个时点上最优化自己的决策.在第 2.1 节的博弈中,参与人是网格用户,战略则是各个网格用户的出价函数(方程式(9)),支付(payoff)反比于用户所花费的时间,资源价格及各用户出价组合的纳什均衡可由方程 $B = \sum_{i=1}^N b^i(B)$ 求解.由于用户出价函数中的系数 α_k^i 和 γ_k^i 分别是资源负载的函数,因此资源负载评估的差异可以产生不同的纳什均衡解(资源分配方案),进而影响到作业执行时间.而在有限次序贯博弈中,用户根据上次博弈的结果不断地调整出价,如图 3 所示,逐渐形成一种均衡局面,使得资源负载最终趋向于一个稳定值.

序贯博弈从时间角度讲属于动态博弈,从信息角度看,每次博弈使用上一次博弈的结果作为博弈条件,于是产生了不同于上一次博弈的纳什均衡.令 G 是序贯博弈的阶段博弈, $G(n)$ 是第 n 次阶段博弈;产生的资源价格记为 $B^{G(n)}$.先利用历史的资源负载均值 $B^{G(0)}$ 和时间优化策略进行第 1 轮博弈,将 $B^{G(0)}$ 代入方程(8),并求解方程式 $B = \sum_{i=1}^N b^i(B)$,即

$$B = \sum_{i=1}^N \left(\frac{(\alpha^i(B^{G(0)}) - \beta^i B)^2}{2\gamma^{i^2}(B^{G(0)})^2} \left(-1 + \sqrt{1 + \frac{4\gamma^{i^2}(B^{G(0)})^2 B}{(\alpha^i(B^{G(0)}) - \beta^i B)^2}} \right) \right) \quad (10)$$

得到资源均衡价格 $B^{G(1)}$,此即第 1 轮博弈预测的资源负载.然后进行第 2 轮博弈,将资源价格负载 $B^{G(1)}$ 代入方程式(8),再求解方程式 $B = \sum_{i=1}^N b^i(B)$,即

$$B = \sum_{i=1}^N \left(\frac{(\alpha^i (B^{G(1)}) - \beta^i B)^2}{2(\gamma^i (B^{G(1)}))^2} \left(-1 + \sqrt{1 + \frac{4(\gamma^i (B^{G(1)}))^2 B}{(\alpha^i (B^{G(1)}) - \beta^i B)^2}} \right) \right) \quad (11)$$

得到资源均衡价格 $B^{G(2)}$,此即第 2 轮博弈预测的资源负载.重复这个过程,直到预测负载的变化小于指定阈值.设最后一轮博弈为第 $n-1$ 轮博弈,则以此轮博弈预测的资源负载 $B^{G(n-1)}$ 为参数,可以计算出所有用户的最优出价组合(纳什均衡解)和资源最终价格 $B^{G(n)}$.

$$\left\{ b^i = \max \left\{ 0, \frac{(\alpha^i - \beta^i B^{G(n-1)})^2}{2\gamma^{i^2}} \left(-1 + \sqrt{1 + \frac{4\gamma^{i^2} B^{G(n-1)}}{(\alpha^i - \beta^i B^{G(n-1)})^2}} \right) \right\} \right\}_{i=1}^N \quad (12)$$

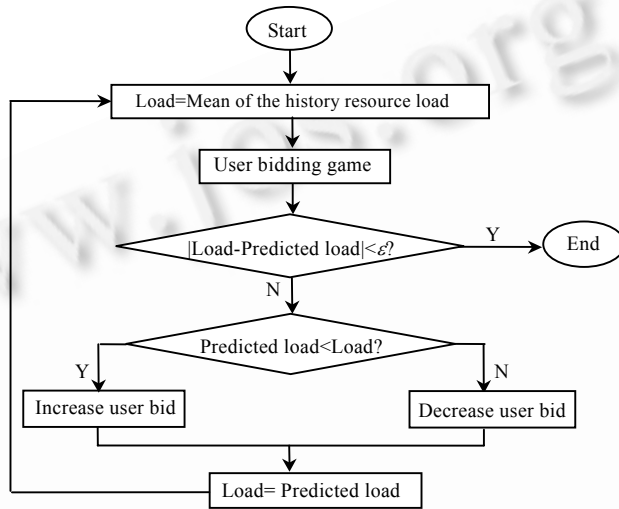


Fig.3 The process of price adjusting in the sequential game-based allocation

图 3 序贯博弈策略的调价过程

序贯博弈策略的输入、输出数据流如图 4 所示.

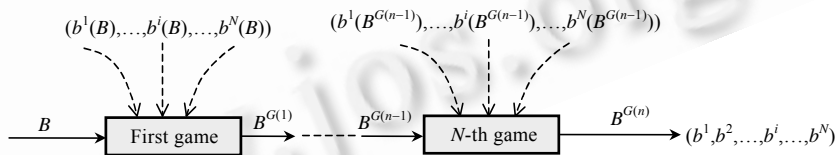


Fig.4 Data flow of input and output using sequential game-based allocation strategy

图 4 基于序贯博弈的分配策略的输入、输出数据流

图 5 和图 6 分别给出了 3 个用户和 15 个用户参与竞争的序贯博弈中,资源的分配比例随阶段博弈次数的变化趋势.在用户数量较少时,序贯博弈能够很快收敛;而在用户数量较多时,需要多次阶段博弈才能达到资源分配的稳定状态.

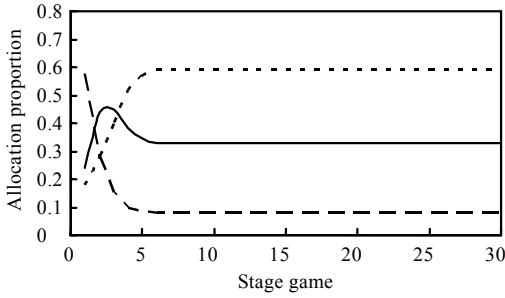


Fig.5 Convergence process of resource allocation proportion with three competing users

图 5 3 个竞争用户时,资源分配比例的收敛过程

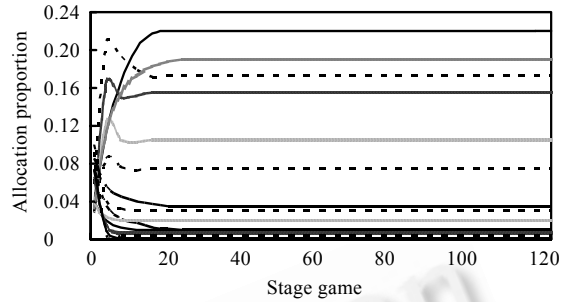


Fig.6 Convergence process of resource allocation proportion with fifteen competing users

图 6 15 个竞争用户时,资源分配比例的收敛过程

3 网格环境中序贯博弈策略的资源分配算法

本节描述如何在网格环境中实现基于重复博弈的资源分配策略.一个简单的网格模型主要有网格用户、网格信息服务和网格资源 3 类实体.网格用户是资源的消费者;网格资源是资源的提供者;网格信息服务负责资源的注册与查询,具有中介功能.具体分配算法如下:

算法 1. 网格用户 i 的出价算法.

Identify resources that can be used in this job execution through

GIS (grid information services)

For all resource available **do**

 identify their capability c_k and estimate their load B_k^{-i}

While receive resource load $B_k^{G(n)}$ **do**

For all task t_k^i **do**

 calculate three coefficients $\alpha^i(B_k^{G(n)}), \beta^i(B_k^{G(n)}), \gamma^i(B_k^{G(n)})$

 send $\alpha^i(B_k^{G(n)}), \beta^i(B_k^{G(n)}), \gamma^i(B_k^{G(n)})$ to the resource of type k

End for

End while

Submit all the tasks to their particular resources...

算法 2. k 类网格资源的分配算法.

While ($|B_k^{G(n)} - B_k^{G(n-1)}| > \varepsilon$) **do**

For all grid user i **do**

 receive $\alpha^i(B_k^{G(n)}), \beta^i(B_k^{G(n)}), \gamma^i(B_k^{G(n)})$ from grid user i

 to build $b_k^i(B_k^{G(n)})$

End for

Search for $B_k^{G(n)} = \sum_{i=1}^N b_k^i(B_k^{G(n)})$ in $\left(0, \max\left\{\frac{\alpha^i(B_k^{G(n)})}{\beta^i(B_k^{G(n)})}\right\}\right)$

For all grid user i **do**

 send $(B_k^{G(n+1)}, b_k^i / B_k^{G(n+1)})$ to grid user i

End for

End while

Schedule tasks with policy...

模拟过程:(1) 网格资源先到网格信息服务(grid information service,简称 GIS)中注册,这个资源注册过程类似于 Globus^[24]系统中 GRIS (grid resource information server)到 GIIS(grid index information server)中注册;(2) 网格用户则到 GIS 中查询资源,GIS 返回一个注册资源的列表和详细的联系方式后,网格用户再向网格资源发送请求索要资源的配置和特性等信息,网格资源返回动态信息,如资源能力、时区、可用性、当前负载和其他配置参数;(3) 网格用户根据资源的动态信息提交出价函数,即方程式(12),包含 3 个系数($\alpha_k^i, \beta_k^i, \gamma_k^i$);(4) 网格资源根据所有用户的出价函数采用二分搜索计算价格负载,资源价格是方程 $B_k = \sum_{i=1}^N b_k^i(B_k)$ 的解,并反馈给网格用户;(5) 网格用户根据资源反馈的价格负载,再次提交出价函数,网格资源再次计算价格负载,如此往复直至预测负载的变化小于指定阈值;(6) 网格资源根据所有用户的出价函数计算资源的最终价格和用户的最优出价组合,并向网格用户发送一对反馈值($B^{G(n)}, b^i/B^{G(n)}$),其中, $B^{G(n)}$ 表示当前时间资源的负载状况,而 $b^i/B^{G(n)}$ 表示分配给用户的资源比例.此分配算法的优点是,由于竞争用户的出价函数是一致的,所以资源能够进行精确的分配.

4 性能分析

在序贯博弈策略中,系统由资源和用户组成.负载的预测是通过有限次序贯博弈来实现,博弈的终止条件是连续两次负载预测结果相差一个很小的常数.用户向资源提交出价函数,资源计算此次出价竞争的纳什均衡,并预测下一次阶段博弈的资源负载,用户根据预测的负载信息调整出价,再次提交出价函数,重复上述过程,直至价格负载的变化在指定范围之内,结束此序贯博弈.每次阶段博弈得到用户出价的局部均衡,从大的时间尺度来看,整个系统趋于整体均衡,即满足收敛性的要求.上述价格负载的预测过程对于动态的网格环境有更好的适应性,且简单易行.对于工作着的网格系统而言,网格资源随时都可能收到用户的使用请求,而且当一些作业执行完成时,资源会有部分空闲能力.因此,最常见的情况是:每当资源有空闲能力时,利用序贯博弈策略为等待队列中的用户作业分配资源.由于阶段博弈的纳什均衡解是资源采用二分搜索迭代计算的,所以,一次阶段博弈的时间复杂度为 $O(N \log(M))$,其中 N 为竞争资源的用户数量, M 为最小的用户出价上界 α_k^i / β_k^i .另外,资源在获取用户的出价函数时产生 $O(N)$ 的网络传送开销.因为网格环境中的远程作业一般都是本地资源无法执行或运行时间较长的,所以,虽然序贯博弈策略的时间复杂度较大,但相对于网格作业执行时间,其开销仍然是可容忍的.而且,序贯博弈在负载预测方面相对于许多基于时间序列的预测方法而言具有优势,能够较准确地预测到历史信息所不能反映的情况,如负载变化趋势的突变,进而优化了用户的作业执行时间.

5 模拟实验结果及讨论

5.1 负载预测方法

如第 2 节所述,在基于出价的正比例资源共享的网格环境中,资源价格负载评估可影响到整体作业执行时间.为了更准确地预测负载以提高资源利用率、缩短作业执行时间,我们提出了一个利用序贯博弈预测负载的资源分配策略,并与目前常用的 3 种负载预测方法进行比较.

(1) 一步超前预测(one-step-ahead,简称 OSA).这个模型是基于趋势的时间序列预测,能够根据历史的 CPU 负载信息提供一步超前的 CPU 负载预测.其基本思想是,根据时间序列变化的趋势预测下一个值,如果当前值上升(下降),那么下一个值也上升(下降).在利用此模型时,一个需要处理的问题是上升(下降)常量的确定.我们将这个常量的初始值设为 0.5,然后在每个时间步计算当前负载值和前一步负载值的差值,然后根据这个差值改写相应的上升(下降)常量来预测下一步的负载值.

(2) 区间均值预测(interval mean,简称 IM).由于 CPU 负载的时间序列具有高度的自相似性,所以多个连续时间段内的负载平均值不会过于平滑.为了预测一定时间区间内的负载均值,需要对初始 CPU 负载的时间序列进行集合运算(取平均值).对集合运算后区间均值的时间序列使用一步超前方法预测下一区间的平均负载.这就面临一个集合度的确定问题,集合度用来计算时间区间内负载均值的初始数据点的个数.集合度的取值不应

太大,否则会屏蔽掉负载变化.一般可按下式来计算集合度 M :

$$M = \text{任务执行时间} / \text{初始负载时间序列的测量时间间隔}$$

(3) 历史均值预测(history mean,简称 HM):应用开始前 5 分钟内的历史资源负载均值.对于变化较大的负载,可缩小时间尺度;而变化较为平缓的负载可适当放宽时间尺度,依具体的环境设定.

(4) 序贯博弈预测(sequential game,简称 SG):包含若干个阶段博弈,每一阶段博弈的结果(预测的价格负载)是下一阶段博弈的条件,直至价格负载的变化在指定范围之内,结束此序贯博弈.初始博弈使用历史均值预测的资源负载为条件,用户在每次博弈中调整出价.

5.2 实验设置

设计一个网格系统是一个非常复杂的系统工程,需要考虑许多问题,比如资源广域共享带来的异构性、动态性、有效的资源管理和调度、网络性能、容错能力、可扩展性以及自适应能力等.而网格模拟器为帮助验证设计方案、测试设计性能提供了极大的便利.模拟器的作用是模拟一个网格环境,在这个环境中可以研究不同的问题,比如可行性和性能.本文选用的网格模拟器 GridSim^[25]是一个基于 Java 的事件驱动的网格仿真工具包,它的主要目标是通过模拟来研究基于计算经济模型的有效资源分配方法.GridSim 有许多优势:在 SimJava 基础上开发、支持 Windows 操作系统;采用面向对象技术,模拟开发比较简单;另外,模拟采用了虚拟时间,不受主机性能的影响,而且不需要开发实际的任务.在 GridSim 中模拟应用调度,需要创建 GridSim 实体.实体继承 GridSim 类,并重写 body()方法来实现自己的功能.每个实体都有两个通信端口,通过类 Input 和 Output 实现,其中每个端口又有自己的 body()方法用于处理发送和接受的事件.本文主要涉及以下实体:网格资源、网格用户、网格信息服务及 I/O.

(1) 网格资源.模拟网格环境下的计算资源见表 1.创建网格资源需要 3 个步骤:首先,创建具有不同 MIPS(million instructions per second)的 CPUs(也称为 PEs(processing elements));然后,一个或多个 PE 形成一台机器;最后,一台或多台机器形成一个网格资源.因此,网格资源可以是单个处理器、共享内存的多处理器(shared memory multiprocessors,简称 SMP),或者一个分布式内存的簇.我们模拟了 4 个异构网格资源,文中资源异构性定义为不同的特性,如 PE 数量、处理器速度、时区等等,这些特性均包含在 ResourceCharacteristics 类的对象中.从 I/O 端口进来的信息流要通过重写 processOtherEvent()方法进行处理,即允许资源定义自己的分配策略.用户和资源之间的网络通信速度以数据传输带宽率定义为 100Mbps.这些网格资源采用正比例资源共享机制管理.资源能力(即总的 PE MIPS rating)定义为 MIPS rating 乘以 PE 数量,按照正态分布从 0.5 变化到 1.5.资源的背景负载(background load)由类 TrafficGenerator 产生.负载的时间序列取 240s,粒度为 1Hz.本文定义高均值为负载均值大于 50,低均值为负载均值小于 10;高变化为负载变化大于 5,低变化为负载变化小于 2.表 1 中的 4 个资源负载分别为低均值低变化(LL)、低均值高变化(LH)、高均值低变化(HL)和高均值高变化(HH).

Table 1 Grid resource characteristics simulated using GridSim

表 1 利用 GridSim 模拟的网格资源特性

Name	Resource characteristics: Vendor, resource type, OS	A PE MIPS rating	PE number	PE load
R_0	Sun, Ultra 5, Solaris	377	8	LL
R_1	Sun, Ultra 4, Solaris	377	16	LH
R_2	Intel, Pentium/VC820, Linux	380	5	HL
R_3	Intel, Pentium/VC820, Linux	380	10	HH

(2) 网格用户.模拟了 20 个竞争资源的网格用户.每个用户包含 2~4 个任务,一个任务就是一个小的网格应用,它包含任务执行管理的所有细节信息,如任务处理需求(表现为 MIPS),磁盘 I/O 操作,输入、输出文件的大小等,这些信息有助于计算远程资源的执行时间.每个任务的任务长度和输入、输出文件的大小是异构的.任务长度至少是 2 000 MI(million instructions),且以 0%~10%正向变化.引入 0%~10%变化的目的是模拟任务的异构性.与此类似,输入文件和输出文件的长度至少是 100 MI 和 250 MI,且在 10%~50%正向变化的区间内取随机数.用户预算根据帕累托 80/20 法则设定,即 20%的用户占有 80%的财富;而 80%的用户只占有财富的 20%.

(3) 网格信息服务.提供网格资源的注册、检索和发现服务.它维护着一张网格资源列表.网格资源在其上注册表明愿意处理网格任务.当网格用户查询资源时,此服务返回一个可用的注册资源列表及其地址.

(4) I/O.提供与其他实体的通信服务.其中,send()方法处理发送调用,get_data()处理接收调用.

5.3 实验结果与讨论

本节给出了4种负载预测方法 OSA,IM,HM 和 SG 在网格资源分配中性能比较的模拟实验结果.文中使用如下性能评价指标:(1) 不同负载条件下资源的占用时间,也即所有竞争用户作业的执行时间;(2) 平均作业执行时间;(3) 作业执行时间的标准偏差.由于我们生成的背景负载在不同的时段各不相同,因此有必要在不同时段开始模拟,以便更全面地搜集评估数据.

图 7~图 10 分别给出了4种负载预测方法的资源占用时间的曲线变化.在不同负载条件下,资源占用时间呈现较大差异.对于低均值的负载,占用时间较短;高均值的负载,占用时间较长.对于低变化的负载,占用时间变化不大;高变化的负载,占用时间变化较大.SG 的占用时间基本位于4条曲线的底部,效果最好,原因是经过连续的博弈,每个用户对其他用户的实力有了大致了解,不仅预测了资源负载,而且优化了此次资源分配.相比而言,OSA 十分不稳定,主要是因为它只能预测短期变化,对长期变化则无法预见,所以这种方法一般在负载变化缓慢的环境中使用.IM 属于利用均值进行预测的方法,因而对于临时或突发性的负载变化,其预测效果不好.HM 效果最差,主要是由于它跨越较大时间尺度,适用于预测中长期变化.

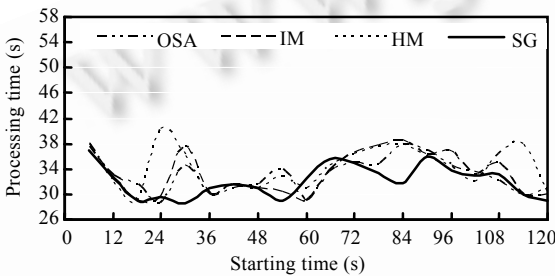


Fig.7 Resource processing time comparison of four load prediction methods on the resource with low-mean and low-variance load

图 7 在低均值低变化的负载条件下,4种负载预测方法的资源占用时间比较

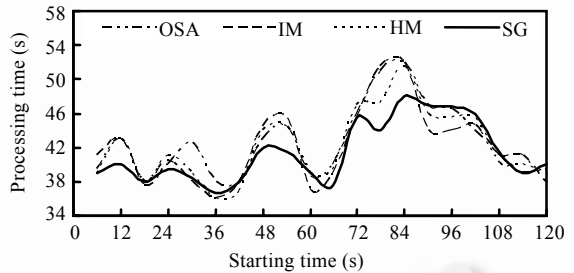


Fig.8 Resource processing time comparison of four load prediction methods on the resource with low-mean and high-variance load

图 8 在低均值高变化的负载条件下,4种负载预测方法的资源占用时间比较

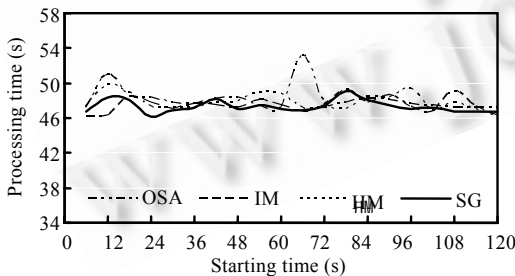


Fig.9 Resource processing time comparison of four load prediction methods on the resource with high-mean and low-variance load

图 9 在高均值低变化的负载条件下,4种负载预测方法的资源占用时间比较

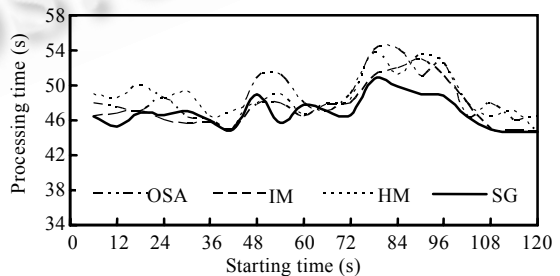


Fig.10 Resource processing time comparison of four load prediction methods on the resource with high-mean and high-variance load

图 10 在高均值高变化的负载条件下,4种负载预测方法的资源占用时间比较

表 2 给出了图 7~图 10 中模拟结果的统计数据.其中,SG 节省的资源占用时间相对于另 3 种方法分别是:比

HM 节省了 1%~6%;比 IM 节省了 0.3%~1.7%;比 OSA 节省了 1.7%~3.5%。另外,SG 缩减的标准差相对于其他方法分别是:比 HM 减少了 13%~27%;比 IM 减少了 18%~82%;比 OSA 减少了 9%~52%。

Table 2 Comparison of mean and standard deviation of total job execution time

表 2 总体作业执行时间的平均值和标准差比较

Experiments	OSA		IM		HM		SG	
	Mean	Standard deviation	Mean	Standard deviation	Mean	Standard deviation	Mean	Standard deviation
Fig.7	33	2.79	33.26	3.35	34.26	3.3	32.09	2.53
Fig.8	42.51	4.25	42.09	4.45	42.09	4.11	41.37	3.57
Fig.9	48.15	1.44	47.5	0.85	47.83	0.95	47.32	0.69
Fig.10	48.64	2.72	47.49	2.46	48.78	2.41	46.92	1.76

6 结论与未来的工作

(1) 针对 Bredin 的分配策略对资源负载评估的不足,本文提出了网格环境下基于预算限制的时间优化的资源分配策略,运用序贯博弈方法获取资源的未来价格负载,更好地适应了网格环境下异构资源的动态性。

(2) 该策略有两个关键技术:将相互影响的多个用户出价竞争同一资源的行为形式化为一个多人动态博弈,该博弈的纳什均衡解即是资源的优化分配方案;采用序贯博弈的方法确定资源的未来负载,根据此负载可以得到合理的用户出价。

(3) 利用网格模拟器 GridSim 评估了该策略的性能,并验证了该策略相对于 Bredin 策略的优越性,即减少了资源占用时间,克服了 Bredin 优化策略中资源价格虚高的缺点,更加接近于网格环境的动态情况,满足用户的服务质量需求。

(4) 今后的工作是进一步完善该分配策略,使其更好地满足用户的 QoS 需求。有待考虑的因素有任务之间的通信费用和通信时间、服务的可靠性及异构性等。把这些情况考虑到网格用户的优化目标中是非常复杂的,需要进一步深入地加以研究。

References:

- [1] Foster I, Kesselman C, Tuecke S. The anatomy of the grid: Enabling scalable virtual organizations. *Int'l Journal of Supercomputer Applications*, 2001,15(3):200-222.
- [2] Foster I. The grid: Computing without bonds. *Scientific American*, 2003,288(4):78-85.
- [3] Yang GW, Jin H, Li ML, Xiao N, Li W, Wu ZH, Wu YW, Tang FL. Grid computing in China. *Journal of Grid Computing*, 2004, 2(2):193-206.
- [4] Jin H, Zou DQ, Han ZF. Implementation of a grid system based on Web. *Mini-Micro Systems*, 2003,24(12):2053-2056 (in Chinese with English abstract).
- [5] Jin H. Challenges of grid computing. In: Fan W, Wu Z, Yang J, eds. *Advances in Web-Age Information Management: Proc. of the 6th Int'l Conf. LNCS 3739*, Berlin: Springer-Verlag, 2005. 25-31.
- [6] Krauter K, Buyya R, Maheswaran M. A taxonomy and survey of grid resource management system for distributed computing. *Software: Practice and Experience*, 2002,32(2):135-164.
- [7] Ding J, Chen GL, Gu J. A unified resource mapping strategy in computational grid environments. *Journal of Software*, 2002,13(7): 1303-1308 (in Chinese with English abstract). <http://www.jos.org.cn/1000-9825/13/1303.pdf>
- [8] Zhang WS, Yang GW, Shen MM, Zheng WM. RSDictionary—A global namespace for distributed computing environment. *Journal of Computer Research and Development*, 2005,42(8):1409-1414 (in Chinese with English abstract).
- [9] Li W, Xu ZW, Bu GY, Cha L. An effective resource locating algorithm in grid environments. *Chinese Journal of Computers*, 2003, 26(11):1546-1549 (in Chinese with English abstract).
- [10] Weng CL, Lu XD. A pricing algorithm for market-based resource management on grid computing systems. *Journal of Computer Research and Development*, 2004,41(7):1151-1156 (in Chinese with English abstract).
- [11] Cao HQ, Xiao N, Lu XC, Liu Y. A market-based approach to allocate resources for computational grids. *Journal of Computer Research and Development*, 2002,39(8):913-916 (in Chinese with English abstract).

- [12] Cheng JQ, Wellman MP. The WALRAS algorithm: A convergent distributed implementation of general equilibrium outcomes. *Journal of Computational Economics*, 1998,12(1):1-24.
- [13] Buyya R, Abramson D, Giddy J. A case for economy grid architecture for service-oriented grid computing. In: Proc. of the 10th IEEE Int'l Heterogeneous Computing Workshop. Washington: IEEE Computer Society, 2001. 776-790.
- [14] Wolski R, Plank JS, Brevik J, Bryan T. Analyzing market-based resource allocation strategies for the computational grid. *Int'l Journal of High Performance Computing Applications*, 2001,15(3):258-281.
- [15] Abramson D, Buyya R, Giddy J. A computational economy for grid computing and its implementation in the nimrod-G resource broker. *Future Generation Computer Systems*, 2002,18(8):1061-1074.
- [16] Nash JF. Non-Cooperative games. *Annals of Mathematics*, 1951,54(2):286-295.
- [17] Kwok YK, Song SS, Hwang K. Selfish grid computing: Game-Theoretic modeling and NAS performance results. In: Proc. of the IEEE Int'l Symp. on Cluster Computing and the Grid. Washington: IEEE Computer Society, 2005. 349-356.
- [18] Bredin J, Kotz D, Rus D, Maheswaran RT, Imer C, Basar T. Computational markets to regulate mobile-agent systems. *Autonomous Agents and Multi-Agent Systems*, 2003,6(3):235-263.
- [19] Maheswaran RT, Başar T. Nash equilibrium and decentralized negotiation in auctioning divisible resources. *Group Decision and Negotiation*, 2003,12(5):361-395.
- [20] Liu C, Yang L, Foster I, Angulo D. Design and evaluation of a resource selection framework for grid applications. In: Proc. of the 11th IEEE Int'l Symp. on High-Performance Distributed Computing. Washington: IEEE Computer Society, 2002. 63-72.
- [21] Wolski R, Spring N, Hayes J. The network weather service: A distributed resource performance forecasting service for metacomputing. *Future Generation Computing Systems*, 1999,15(5-6):757-768.
- [22] Arabe JNC, Beguelin A, Lowekamp B, Seligman E, Starkey M, Stephan P. Dome: Parallel programming in a distributed computing environment. In: Proc. of the 10th Int'l Parallel Processing Symp. Washington: IEEE Computer Society, 1996. 218-224.
- [23] Gehring J, Reinefeld A. Mars: A framework for minimizing the job execution time in a metacomputing environment. *Future Generation Computing Systems*, 1996,12(1):87-99.
- [24] Foster I, Kesselman C. Globus: A metacomputing infrastructure toolkit. *Int'l Journal of Supercomputer Applications*, 1998,11(2): 115-128.
- [25] Buyya R, Murshed M. GridSim: A toolkit for modeling and simulation of grid resource management and scheduling. *Journal of Concurrency and Computation: Practice and Experience*, 2002,14(13-15):1175-1220.

附中文参考文献:

- [4] 金海,邹德清,韩宗芬.基于 Web 的网格系统的实现. *小型微型计算机系统*,2003,24 (12):2053-2056.
- [7] 丁箐,陈国良,顾钧.计算网格环境下一个统一的资源映射策略. *软件学报*,2002,13(7):1303-1308. <http://www.jos.org.cn/1000-9825/13/1303.pdf>
- [8] 张武生,杨广文,沈美明,郑邦民.RSDictionary——一种用于分布式计算环境的全局名字空间. *计算机研究与发展*,2005,42(8): 1409-1414.
- [9] 李伟,徐志伟,卜冠英,查礼.网格环境下一种有效的资源查找方法. *计算机学报*,2003,26(11):1546-1549.
- [10] 翁楚良,陆鑫达.一种基于市场机制的网格资源调价算法. *计算机研究与发展*,2004,41(7):1151-1156.
- [11] 曹鸿强,肖依,卢锡城,刘艳.一种基于市场机制的计算网格资源分配方法. *计算机研究与发展*,2002,39(8):913-916.



李志洁(1978 -),女,黑龙江鸡西人,博士生,主要研究领域为网格计算,分布式计算.



黄飞雪(1971 -),男,博士生,副教授,主要研究领域为博弈论.



程春田(1965 -),男,博士,教授,博士生导师,主要研究领域为知识管理系统,决策支持系统和智能算法,空间信息,网格计算的建模技术.



李欣(1979 -),男,博士生,讲师,主要研究领域为网格计算.