基于市场方法的 MAS 系统任务调度算法*

毛雪岷 白石磊
(中国科学技术大学 自动化系 合肥 230027)
(中国科学院合肥智能机械研究所 合肥 230031)
熊范纶 王儒敬
(中国科学院合肥智能机械研究所 合肥 230031)

摘 要 本文探讨了在 Internet 环境下,知识型 MAS 系统的任务负载分配问题,指出了它与网格计算的区别.本文的方法与传统的计算资源分配方法相反,将任务负载看作资源,而将 Agent 作为消费者,每处理一个任务,就获得一定数量的边际效用,利用市场机制,使 Agent 自主进行任务负载的分配.设计了基于均衡市场和连续 CDA 市场的两种调度算法,对算法的效果和特点进行了分析,证明市场机制的调度算法可以有效地实现 MAS 系统任务负载的平衡.此外,本文还给出了一个简单的确定均衡价格的方法.

关键词 Multi-Agent 系统,均衡市场,CDA 市场
中图法分类号 TP18

1 引 言

Internet 的出现促使传统的分布式人工智能(DAI, Distributed Artificial Intelligence)的研究从传统封闭的、基于局域网的系统结构转向开放的、基于 Internet 的、以 Multi-agent 理念为思想基础的体系结构[1].但智能系统的计算资源和带宽资源都是有限的,不能保证为大量的用户提供满意服务.

如果这些智能系统(以下将智能系统称之为 Agent)具有自主的协调、协作和通讯能力,则可以将这些 Agent 组织起来,构成开放式的 MAS 系统,在系统内采用一个好的调度机制使得每个 Agent 能够将过多的用户请求或提交的任务分配给其他合适的 Agent 完成,提高系统效率和服务质量.

这一思想与网格计算有相似之处,但本文所讨论的 Multi-agent 系统是特指由多个知识型 Agent 组成的,每个 Agent 都有其特有的问题求解机制和相应的知识库,问题求解过程是不能迁移的.网格计算是把需大量计算资源的大规模计算任务分解为可并行执行的资源需求较小的多个子任务,通过 Internet 将这些子任务分配到多个计算节点去执行[2],计算过程是可迁移的.这就导致 Multi-agent 系统中的任务调度与网格计算有本质的不同.我们不能采用与网格计算完全相同的策略,把 MAS 系统的任务调度作为单纯的计算资源分配问题来处理,必须考虑到 MAS 系统的特点,设计适当的调度机制.

2 MAS 系统任务调度机制的设计

在 Internet 环境下,Agent 的数量较大,并且是变化的,Agent 属于不同的所有者,其能力和计算资源也是动态变化的,采用集中式的方法来收集和管理 Agent 的状态信息并做出调度决策是极为困难的.分布式的调度机制允许 Agent 根据自身的任务负载做出判断,发出调度请求,能够及时反映整个系统的任务分布状态,以小的计算和通信开销实现较好的调度效果.

在设计分布式的调度机制时,需要相应的机制设计理论,来研究如何在一群利已主义(Self-Inter-

* 国家 863 计划资助项目(No. 2001AA110464)
收稿日期:2002-09-12;修回日期:2003-11-19

ested))的 Agent 中实现全局范围的优化解决方案[3].机制设计的理论基础是微观经济学和博弈论,近年来在分布式人工智能、通信网和调度等领域都有应用[4-9].

本章所探讨的 Internet 环境下 MAS 系统的任务调度,核心问题是如何对一个初始任务分布 X 作重新配置,得到新的任务分配 X',在公平性的前提下提高系统效用.在网格计算研究中,通常考虑的是如何充分利用各个节点的计算资源,但在我们讨论的问题中,Agent 的计算资源是难以获知的.我们认为,可以反过来将用户提交的任务视作资源,Agent 则利用这些资源"生产"对用户的服务,这样任务调度问题就转化为资源配置问题,从而可以利用经济学中的成果来解决这一问题.

实现资源分配的经济模型中,可适用于 MAS 系统任务调度的是均衡市场模型、合同网模型、议价模型和拍卖模型.

议价模型需要供求双方寻找适合的交易对象,并对交易数量和价格进行反复的协商,负载过重的 Agent 可能需要与不止一个较为空闲的 Agent 交易,才能够将过多的负载分配出去.若系统中有 n 个 Agent,对于任务负载为 x 的 Agent,共通讯开销为 $O(x)\cdot O(n)$.在合同网模型中,Agent 需要发出 $n-1$ 个招标通告,接收 $n-1$ 个投标书,通讯开销至少为 $2n-2$,当 n 的数值较大时,对于需实时处理的任务调度问题,这两种模型都是不适用的.下面我们将讨论基于均衡市场模型的任务调度算法.

3　调度机制的评价

任务调度机制应当保证在系统资源约束下实现用户满意度和资源利用率的最大化,同时保证公平性要求.用户满意度可用用户等待时间和结果准确度两个参数来表示,用户等待时间越短,结果准确度越高,则用户满意度越高.资源利用率可用 Agent 承担的任务数来反映.本文的方法是将用户等待时间 t,结果准确度 a,Agent 承担的任务数 x 映射为 Agent 的效用函数 $u(t,a,x)$,用满足公平性要求这一前提下效用函数的高低来衡量调度效果.

本文所讨论的 MAS 系统是由向用户提供专家系统、决策支持等服务的知识型智能 Agent 组成,Agent 所给出的结果的准确度是衡量 Agent 性能的极其重要的因素,MAS 系统需要给 Agent 的结果准确度设定一个下限 a_0,不满足准确度要求的 Agent 不能加入系统,保证系统中每个 Agent 都能够提供准确度高于用户最低要求的服务.

在结果的准确度 a 一定的情况下,显然用户等待时间 t 越小,其满意度越高.设 t_0 为用户等待时间的上限,对于某一用户提交的任务,若有 $t\leqslant t_0, a\geqslant a_0$,则称 Agent 满足用户的基本满意度要求.

Agent 对用户提交给它的任务有两种处理方式,一是为每个任务开辟一个进程,并发处理所有任务(目前的操作系统对 Web 服务均采用并发方式);二是按照用户提交的先后排队处理.在网络环境下,我们可以认为任务到达概率符合泊松分布.即在 t 时间内到达 x 个任务的概率为

$$V_x(t)=e^{-\lambda t}\frac{(\lambda t)^x}{x!},\ x=0,1,2,\cdots, \tag{1}$$

平均到达概率为 λ.

本文讨论的 Agent 提供的是知识型服务,不同问题所需求解时间是不同的、不可预知的,求解一个问题所需时间的分布函数 $S_0(t)$ 同样为负指数分布,即

$$S_0(t)=\begin{cases}e^{-\mu t}, & t>0\\ 1, & t\leqslant 0\end{cases}, \tag{2}$$

可得平均问题求解时间为 $\frac{1}{\mu}$.

若 Agent 为每个任务创建一个进程,进行并发处理,可看作一个特殊的 $M/M(n)/c(n)$ 排队系统,在这个系统中,服务台的数量 $c(n)=n$,平均服务时间为 $\mu(n)=\frac{1}{n\mu}$,μ 为仅有一个任务时的平均处理速率,代入 $M/M/c$ 排队系统的稳态方程

$$\begin{cases}\lambda P_{n-1}+c\mu P_{n+1}-(\lambda+c\mu)P_n=0\\ \mu P_1+\lambda P_0=0\end{cases} \tag{3}$$

可得 Agent 的平均任务负载为 $L_s=\sum_{n=1}^{\infty}nP_n=\frac{\lambda}{\mu-\lambda}$.

用户平均等待时间为 $T_s=\mu(n)=\frac{1}{n\mu}=\frac{L_s}{\mu}$,与任务负载成正比.采用排队处理方式仍有类似结果.

由于用户等待时间 t 与任务负载 x 成正比,这样 Agent 的效用函数 u 可以只用 a 和 x 两个参数定义,记作 $U(x,a)$.

若 $U(x,a)$ 满足

$$\frac{\partial U}{\partial a}>0,\frac{\partial U}{\partial x}\Big|_{x<x_0}>0,\frac{\partial U}{\partial x}\Big|_{x=x_0}=0,\quad \frac{\partial U}{\partial x}\Big|_{x>x_0}<0, \tag{4}$$

式中 x_0 满足 $t(x_0)=t_0$.

满足上式的效用函数 $U(x,a)$ 有如下特性：

1. $U(x,a)$ 随着准确度的提高而单调增加.

2. 在满足用户对等待时间的最低要求的前提下(即 $t \leqslant t_0$)，$U(x,a)$ 随着 Agent 承担的任务数 x 单调增加.

3. 若准确度保持一定，当 Agent 承担的任务数 x 使得用户等待时间达到用户最低要求 t_0 时，效用函数 $U(x,a)$ 达到极大值，即效用函数的极值点就是在保证用户基本满意度要求的情况下，资源利用率最高的状态.

4 基于均衡市场机制的 MAS 系统任务调度算法

我们先来定义 Agent 的效用函数. 设 A_i 为系统中的 Agent，A_i 的平均任务处理时间为 τ_i，x_i 为 A_i 承担的任务数，a_i 为 A_i 的问题求解结果的平均准确度，m_i 为 A_i 所拥有的货币数量，t_0 为用户最低满意度的等待时间，当 A_i 的系统利用率过低时，用货币向其他 Agent 购买任务，任务负载过重时，出售一定数量的任务. 设 A_i 的用户平均等待时间为 t_i，由上节讨论可知

$$t_i = \tau_i \cdot x_i. \tag{5}$$

记满足 $t_i(x_i) = t_0$ 的 x_i 为 $\overline{x_i}$，定义 A_i 的效用函数如下：

定义 1 A_i 由消费 x_i 个任务得到的效用为

$$u(x_i) = \begin{cases} -a_i(x_i - \overline{x_i})^2, & x_i \leqslant \overline{x_i} \\ -a_i^{-1}(x_i - \overline{x_i})^2, & x_i > \overline{x_i} \end{cases}. \tag{6}$$

定义 2 $U_i(x_i, m_i)$ 为 Agent A_i 在任务数为 x_i，货币量为 m_i 时的效用，且有

$$U_i(x_i, m_i) = m_i + u(x_i), \tag{7}$$

由(6)式得

$$U_i(x_i, m_i) = m_i - \begin{cases} a_i(x_i - \overline{x_i})^2, & x_i \leqslant \overline{x_i} \\ a_i^{-1}(x_i - \overline{x_i})^2, & x_i > \overline{x_i} \end{cases}. \tag{8}$$

设 A_i 在调度前的任务量为 x_i，其价格为 p，货币量为 m_i，则其总收入为 $M = p \cdot x_i + m_i$，A_i 在与其他 Agent 交易任务时要受 M 的约束，其效用最大化问题可表述如下：

$$\max_{x_i, m_i}[m_i + u(x_i)] \quad s.t.\ m_i + px_i = M. \tag{9}$$

效用函数只需能表明一种状态好于另一种状态即可，数理经济学的研究表明基数效用与基于"偏好"的序数效用得出的结论是相同的. 上式所定义的为拟线性(quasi-linear)效用函数，优点是可以在进行效用最大化分析时不考虑总收入约束. 显然上式满足前面对效用函数的要求.

Agent 进行任务交易时的供需关系应受价格影响. 当价格上升时，需求下降，供给增加；价格降低时，需求增加，供给减少. 反之，供大于求时，价格上升；供小于求时，价格下降.

设系统中有 n 个 Agent，记作 $A_1, A_2, \cdots, A_n$，对于任一 A_i，在任务调度前的任务和货币的数量为 x_{i0} 和 m_{i0}，称向量 $\omega_i = (x_{i0}, m_{i0})$ 为 A_i 的初始禀赋，$X_i(P) = (x_i, m_i)$ 为在价格向量 P 下使 A_i 达到效用最大化的最优消费束，$P = (p, 1)$ 为表示当前任务和货币价格的价格向量(由于本文采用拟线性效用函数，m_i 的价格始终为 1). $X_i(P)$ 可由下式解出：

$$\frac{\partial u(x_i)}{\partial x_i} = p = \begin{cases} -2a_i(x_i - \overline{x_i}), & x_i \leqslant \overline{x_i} \\ -2a_i^{-1}(x_i - \overline{x_i}), & x_i > \overline{x_i} \end{cases}. \tag{10}$$

我们称(10)式为 A_i 的价格函数，表明在 A_i 对任务的需求量为 x_i 时，为达到效用最大化应将任务的价格定为 p. 价格函数的反函数称为 A_i 的需求函数，表明在价格为 p 时，A_i 对任务的需求量. 要说明的是，现实和经济学研究中，不存在负值的价格，而在本文的方法中，若供大于求，价格为负值. 但从下面的讨论中可以看出，负值的价格仍然满足经济学的要求，并不影响结论.

从(10)式还可以看出，p 作为 A_i 在当前负载为 x_i 时的报价，当任务负载不足时 $p > 0$；达到最佳负载时 $p = 0$；过载时 $p < 0$.

记 $\partial u(x_i)/\partial x_i$ 为 $u_x(x_i)$，称为 x_i 的边际效用，表示 A_i 每多"消费"一个任务引起的效用变化. 价格函数表明当边际效用等于价格时，Agent 达到效用最大化.

定义 3 记 $Z_i(P) = X_i(P) - \omega_i$，称 $Z_i(P)$ 为 A_i 的净需求向量，$z_{xi}(p) = x_i(p) - x_{i0}$ 为 A_i 对任务的净需求，$Z(P) = \sum_{i=1}^{n} Z_i(P)$ 为 MAS 系统的超额需求向量. 如果通过调节价格 P，使得在价格 P^* 下，有

$$Z(P^*) = \sum_{i=1}^{n} Z_i(P^*) = 0, \tag{11}$$

则称市场达到 Walrasian 均衡，P^* 称为均衡价格.

从上式可以看出，市场达到均衡意味着市场出清(market clear)，此时供需相等，所有交易都可以完成. 结合价格函数还可以得到以下均衡市场的性质：

性质 1 市场出清后，所有 Agent 对于任务 x 的边际效用相等，且等于均衡价格 p^*.

MAS系统的任务调度过程可看作是交换市场的交易过程,在系统中设定一个Agent作为市场组织者,称为Auctioneer,负责收集各Agent的供需要求并据此调节价格.各个Agent根据价格变动调整自己的供需要求,并反馈给Auctioneer,直至达到市场均衡.下面我们证明在本文定义的效用函数下必然存在均衡价格 P^*,并给出调度过程的算法描述.

定理1　在(8)式定义的效用函数下,必存在唯一的均衡价格 P^*,使得 $Z(P^*)=0$.

证明　由Walrasian均衡存在性定理,我们只需证明以下三点:

1) 超额需求 $Z(P)$ 连续;

2) $Z(P)$ 满足Walrasian法则,即 $P\times Z(P)=0$;

3) (8)式表示的效用函数为严格凸偏好.

1) 由于(8)式为拟线性效用函数,只需考虑 x 的超额需求 $Z_x(p)$ 是否连续,由(10)式得 A_i 的需求函数为

$$x_i(p)=\begin{cases}\overline{x_i}-\dfrac{p}{2a_i}, & p\geqslant 0\\ \overline{x_i}-\dfrac{pa_i}{2}, & p<0\end{cases},\qquad (12)$$

显然 $x_i(p)$ 连续.由于 $z_{xi}(p)=x_i(p)-x_{i0}$,可得 $z_{xi}(p)$ 连续,而 $Z_x(p)=\sum_i z_{xi}(p)$,所以 $Z_x(p)$ 也是连续的.

2) A_i 为任一Agent,其初始禀赋为 $\omega_i=(x_{i0},m_{i0})$,则(9)式中约束条件为

$$m_{i0}+px_{i0}=M_i,$$

A_i 的需求函数应满足这一约束,即有 $PX_i=P\omega_i$,则 $P\times Z(P)=P\times(X_i-\omega_i)=PX_i-P\omega_i=0$,满足Walrasian法则.

3) 由(10)式可得,边际效用 $u_x(x_i)$ 满足

$$\frac{\partial u_x(x_i)}{\partial x_i}=\begin{cases}-2a_i<0, & x_i\leqslant\overline{x_i}\\ -2a_i^{-1}<0, & x_i>\overline{x_i}\end{cases},$$

故 $u_x(x_i)$ 是严格递减的,设 $X_1=(x_1,m_1)$ 和 $X_2=(x_2,m_2)$,且 $U(X_1)\geqslant U(X_2)\geqslant K$,令 $X_3=tX_1+(1-t)X_2$, $t\in(0,1)$,则有 $X_3=(x_3,m_3)=(tx_1+(1-t)x_2,tm_1+(1-t)m_2)$.设 $x_1>x_2$,则 $x_3=x_2+t(x_1-x_2)$, $m_3=m_2+t(m_1-m_2)$,而

$$\begin{aligned}U(X_3)&=m_3+u(x_3)\\&=m_2+t(m_1-m_2)+\int_0^{x_2}u_x(x)\mathrm{d}x+\int_{x_2}^{x_2+t(x_1-x_2)}u_x(x)\mathrm{d}x\\&=m_2+t(m_1-m_2)+u(x_2)+\int_{x_2}^{x_2+t(x_1-x_2)}u_x(x)\mathrm{d}x,\end{aligned}$$

由 $u_x(x_i)$ 严格递减可得

$$\int_{x_2}^{x_2+t(x_1-x_2)}u_x(x)\mathrm{d}x>t\int_{x_2}^{x_1}u_x(x)\mathrm{d}x=t(u(x_1)-u(x_2)),$$

故 $U(X_3)>m_2+t(m_2-m_1)+u(x_2)+t(u(x_1)-u(x_2))=U(X_2)+t(U(x_1)-U(X_2))$,

由 $t\in(0,1)$, $U(X_1)\geqslant U(X_2)\geqslant K$ 可得 $U(X_3)>K$.

若 $x_2>x_1$,同理可得 $U(X_3)>tU(X_1)+(1-t)U(X_2)>K$.

所以 $U(X)$ 的严格凸性得征.

从1)、2)、3)得,存在唯一的均衡价格 P^*,证毕.

定理2　在均衡价格 P^* 下,MAS系统的任务分配达到Pareto最优.

证明　由(10)式可得,此时所有Agent的边际效用相等,均为 p^*,假定此时未达到Pareto最优,则必存在 A_i 和 A_j, $X_i=(x_i,m_i)$, $X_j=(x_j,m_j)$,仍可在某一价格 p 下交易,且一方效用提高,另一方效用至少不降低.设 A_i 以价格 p 出售 ε 个任务给 A_j,则 $X_i=(x_i-\varepsilon,m_i+p\varepsilon)$, $X_j=(x_j+\varepsilon,m_j-p\varepsilon)$,若 A_i 的效用增加,则有 $\Delta u_i=u_i(x_i-\varepsilon)-u_i(x_i)>-p\varepsilon$,即 $u_i(x_i)-u_i(x_i-\varepsilon)<p\varepsilon$,由于边际效用严格递减,$u_i(x_i)-u_i(x_i-\varepsilon)>p^*\varepsilon$,故有 $p^*<p$.而 $\Delta u_j=u_j(x_j+\varepsilon)-u_j(x_j)\geqslant p\varepsilon$,同样由于边际效用严格递减,有 $u_j(x_j)-u_j(x_j-\varepsilon)<p^*\varepsilon$,故有 $p^*\geqslant p$,得出矛盾.证毕.

市场均衡时,任务调度是否达到了最好的效果呢?从(10)式可以看出,对于每个 A_i 来说,$\overline{x_i}$ 是其最佳负载,此时在保证用户可以得到满意服务的前提下,A_i 的资源利用率达到最高.因此在MAS系统的总的任务负载给定的情况下,应使每个Agent的负载尽可能靠近最佳负载点 $\overline{x_i}$.我们可以用各个Agent偏离 $\overline{x_i}$ 的均方差 S 来衡量任务调度的效果,即

$$S=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i-\overline{x_i})^2}$$

$$s.t.\ \sum_{i=1}^{n}x_i=N.$$

而基于均衡市场的调度算法有如下性质:

性质1　若MAS系统中每个Agent的准确率 a_i

均为1,则市场出清后 S 达到最小.

证明 用 Lagrange 乘子法可证.

由于 a_i 趋近于1,故此均衡市场的调度算法可使 S 趋近于最小化,这表明该算法的确可以改善系统的任务分布.

性质2 在均衡市场价格 p^* 下,其它条件相同时,准确度较高的 Agent 将承担较多的任务负载.(证明略)

我们在效用函数定义中当 $x_i > \overline{x_i}$ 时,将系数由 a_i 变为 a_i^{-1} 就是出于这种考虑.

性质3 在均衡市场价格 p^* 下,其它条件相同时,计算资源多的 Agent 将承担较多的任务负载.(证明略)

性质2和性质3表明,基于均衡市场的调度算法较之单纯追求偏差 S 最小化的方法更为合理.

在传统的经济学理论中,均衡价格的确定是通过试错过程(tatonnement process)来确定的,也就是由 auctioneer 先随机给出一个价格,再根据此价格下的市场供求状况调整,直至市场出清.一些采用市场方法进行资源分配的文献也是通过试错来确定价格的[11].但本文给出的效用函数使我们可以直接计算出均衡价格和各个 Agent 的净需求.由于在市场达到均衡时,所有 Agent 的边际效用相等,因此,若 $\sum_i x_i > \sum_i \overline{x_i}$ 时,对所有的 Agent 均有 $x_i > \overline{x_i}$;若 $\sum_i x_i \leqslant \sum_i \overline{x_i}$ 时,对所有的 Agent 均有 $x_i \leqslant \overline{x_i}$.每个要求调度的 Agent 向 Auctioneer 报告自己当前的状态向量 $(x_{i0}, \overline{x_i}, a_i)$, Auctioneer 计算 $\sum_i x_i$ 和 $\sum_i \overline{x_i}$,假定 $\sum_i x_i > \sum_i \overline{x_i}$, $\sum_i x_i = N$,我们用 $X = (x_1, K, x_n)$ 表示所有 Agent 对任务的需求向量,$X_0 = (x_{10}, K, x_{n0})$ 表示调度前各个 Agent 的负载,则用下面的方法计算均衡价格和各个 Agent 的净需求.

当 $\sum_i x_i \leqslant \sum_i \overline{x_i}$ 时,必有 $p \geqslant 0$,从需求函数可得

$$x_i(p) + \frac{1}{2a_i}p = \overline{x_i}, \quad i = 1,2,\cdots,n,$$

再加上约束条件

$$x_1 + x_2 + \cdots + x_n = N.$$

用矩阵形式表示为

$$\begin{bmatrix} 1 & 0 & \cdots & 0 & 1/2a_1 \\ 0 & 1 & \cdots & 0 & 1/2a_2 \\ & & O & & \\ & & & 1 & 1/2a_n \\ 1 & 1 & \cdots & 1 & 0 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ M \\ x_n \\ p \end{bmatrix} = \begin{bmatrix} x_1 \\ \overline{x_2} \\ M \\ \overline{x_n} \\ N \end{bmatrix}. \tag{13}$$

解上式方程组,即可得对任务需求向量 $X = (x_1, K, x_n)$ 和均衡价格 p^*,由于 x_i 应为整数,对其进行取整操作,令 $x_i = \lfloor x_i + 0.5 \rfloor$,对任务的净需求向量由 $Z = X - X_0$ 得出.若 $\sum_i x_i > \sum_i \overline{x_i}$,则将上式中的 $1/2a_i$ 替换为 $a_i/2$ 即可.

由此我们可以给出即与均衡市场的调度算法的描述:

1. 每个 Agent 向 Auctioneer 发出当前时刻的 $(x_{i0}, \overline{x_i}, a_i)$;

2. Auctioneer 计算出 $\sum_i x_i$ 和 $\sum_i \overline{x_i}$,并根据两者的大小用(13)式解出 Z;

3. 依次将 Z 中小于0的分量对应的 Agent 放入需求队列 D,将大于0的分量对应的 Agent 放入供应阶列 S;

4. Auctioneer 分别取出 D 和 S 中的第一个 Agent,记为 D_1 和 S_1;

5. 若 $Z(D_1) > Z(S_1)$,通过 S_1 将 $Z(S_1)$ 个任务转移给 D_1,将 S_1 移出队列 S,
$Z(D_1) = Z(D_1) - Z(S_1)$,转8;

6. 若 $Z(D_1) = Z(S_1)$,通知 S_1 将 $Z(S_1)$ 个任务转移给 D_1,将 D_1 和 S_1 移出队列 S,转8;

7. 若 $Z(D_1) < Z(S_1)$,通过 S_1 将 $Z(D_1)$ 个任务转移给 D_1,将 D_1 移出队列 D,
$Z(S_1) = Z(S_1) - Z(D_1)$;

8. 若队列 D 和 S 不空,转4,否则转9;

9. 算法结束.

基于均衡市场的调度算法可以保证较好的实现负载均衡,但为避免频繁地进行任务分配,应每隔一个固定时间段进行一次任务调度,且由于存在均衡价格和净需求向量的计算过程,在此期间不考虑用户新提交的任务,不能完全反映任务分布状态的变化.为此也可以采用较简单的基于 CDA 市场模型的算法.

5 试验结果

下面是均衡市场算法进行任务调度的模拟实验结果.在4个 Agent 组成的 MAS 系统中进行任务调度,参数设置如下表,各时段任务负载按泊松分布随机产生.

表 1　各 Agent 的准确度、最佳负载及不同时刻的实际负载

	准确度 a	最佳负载	$T=1$	$T=2$	$T=3$	$T=4$	$T=5$	$T=6$
$A1$	0.9	20	20	35	47	53	56	54
$A2$	0.8	30	5	7	13	9	6	4
$A3$	0.8	20	17	30	23	9	28	17
$A4$	0.7	80	30	7	6	40	36	39

结果如图 1 ～ 图 4 所示,图中"o"、实线、"*"和虚线分别代表 $A1$、$A2$、$A3$ 和 $A4$.

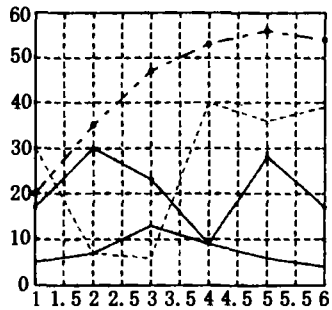

图 1　Agent $A1$、$A2$、$A3$ 和 $A4$ 在调度前的任务负载

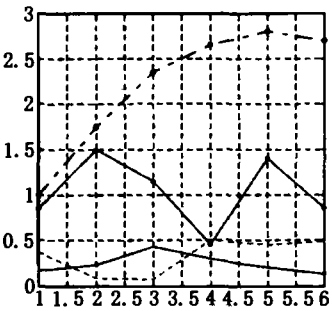

图 2　用户感到的相对时延(任务负载 / 最佳负载)

大于 1 的相对时延表明不满足用户对响应时间的要求.

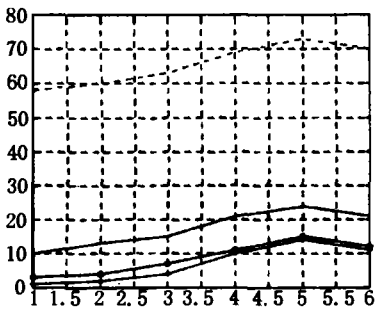

图 3　调度后的任务负载分布

从图 3 中可以看出,尽管 $A1$ 与 $A3$ 的计算资源相同,有相同的最佳任务负载,但 $A1$ 的准确度较高,故调度后 $A1$ 得到比 $A3$ 更多的任务.

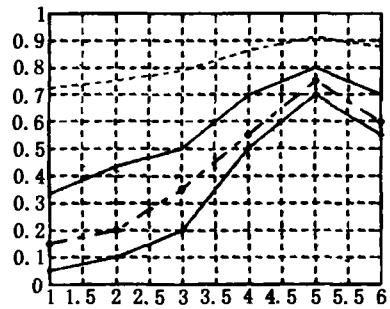

图 4　调度后用户感受到的相对时延

从图 4 中可以看出,调度后的相对时延均小于 1,可满足用户对响应时间的要求.

参 考 文 献

[1] Levy A, Weld D. Internet Intelligent System. Artificial Intelligence, 2000, 118(1): 1 - 14

[2] 丁　菁.网格环境下资源管理的研究.博士学位论文,中国科学技术大学,合肥,2002

[3] Semret N. Market Mechanisms for Network Resource Sharing. Ph. D Thesis, Columbia University, New York, USA, 1999

[4] Wellman M, Wurman P, *et al*. Designing the Market Game for a Trading Agent Competition. IEEE Internet Computing, 2001, 5 (2): 43 - 51

[5] Long D. The AIPS-98 Planning Competition. AI Magazine, 2000, 21(2): 13 - 33

[6] Schaerf A, Shoham Y, Tennenholtz M. Adaptive Load Balancing: A Study in Multi-Agent Learning. Journal of Artificial Intelligence Research, 1995, 2: 475 - 500

[7] Huberman B, Clearwater S. A Multi-Agent System for Controlling Building Environments. In: Lesser V, ed. Proc of the 1st International Conference on Multi-Agent Systems(ICMAS), San Francisco, USA, 1995, 171 - 176

[8] Gagliano R, Fraser M, Schaefer D M. Allocation of Computing Resources. Communications of the ACM, 1995, 38(6): 88 - 103

[9] 张尧学,等.一种基于市场模型的网络带宽分配方法.电子学报,1999, 27(9): 127 - 132

[10] 蒋殿春.高级微观经济学.北京:经济管理出版社,2000

[11] Ygge F, Akkermans H. Decentralized Markets Versus Central Control: A Comparative Study. Journal of Artificial Intelligence Research, 1999, 11: 301 - 333

AN ALGORITHM BASED ON EQUILIBRIUM MARKET FOR TASK ALLOCATION IN MAS

Mao Xuemin, Bai Shilei
(*Deparement of Automation*, *University of Science and Technology of China*, *Hefei* 230027)
(*Institute of Intelligent Machines*, *Chinese Academy of Sciences*, *Hefei* 230031)
Xiong Fanlun, Wang Rujing
(*Institute of Intelligent Machines*, *Chinese Academy of Sciences*, *Hefei* 230031)

ABSTRACT

In this paper, tasks allocation method for knowledge type MAS in Internet environment is discussed. Different to other methods of computation resource allocation, our metod take tasks as resource and take Agents as consumers. When an Agent completed a task, it would get amount of marginal utility. So Agent could allocate tasks by itself with market mechanism. An algorithm based on equilibrium market is proposed; effect and characteristic of the algorithms is also been discussed; the algorithm is proved that it could realize load balancing effectively. Furthermore, a very simple method for calculating equilibrium price is proposed.

Key Words Multi-Agent System, Equilibrium Market, CDA Market