维普资讯 http://www.cqvip.com

基于强化学习的多机群网格资源调度模型*)

陈庆奎
(上海理工大学计算机工程学院　上海 200093)

摘　要　在由多个计算机集群构成的多机群网格环境下，为了解决数据并行型计算(DPC)与计算资源的有效匹配问题，提出了一个基于强化学习机制的网格资源调度模型；给出了由多个计算机机群组成的多机群网格、逻辑计算机机群、数据并行型计算和一系列 Agent 的定义；利用多 Agent 的协作做竞争机制、基于强化学习的匹配知识库的修正方法，研究了逻辑计算机机群与 DPC 资源供需之间的有效匹配问题；描述了网格的资源调度模型。理论分析和实践表明，该模型有效地解决了多机群网格环境之下数据并行型计算所需的资源优化使用问题。该模型适合于基于多机群网格的数据并行型计算。

关键词　多机群网格，多智能体，资源调度，数据并行型计算，强化学习

Multi-cluster Grid Resource Dispatch Model Based on Reinforcement Learning

CHEN Qing-Kui
(College of Computer Engineering, Shanghai University of Science and Technology, Shanghai 200093)

Abstract　For resolving the problem to effectively match between Data Parallel Computing (DPC) and computational resources in Multi-cluster Grid that composed of many computer clusters, a grid resource dispatch model based on reinforcement learning is discussed. A series of formal definitions, such as the Multi-cluster Grid (MCG), the Logical Computer Cluster(LCC), the DPC and the Agents, are given. Using the mechanism of cooperation and competition of Multi-Agent, the knowledge-base revising techniques based on reinforcement learning, the effective match methods are studied. The resource dispatch model is described. The analysis and experiment results show that this model effectively resolves the problems of optimized use of resources in Multi-cluster Grid. It can be fit for the data parallel computing in Grid.

Keywords　Multi-cluster grid, Multi-agent, Resource dispatch, Data parallel computing, Reinforcement learning

1　引言

随着信息技术的飞速发展，信息爆炸已经成为当今信息技术产业界和学术界所面临的急需解决的主要问题。网格[1,2]成为一个可行的解决方案。在网格中，强大的数据处理、存储中心是一个研究焦点。实现这种中心的有效手段是构建由多个网络构成的大规模机群。另一方面，由多个计算机网络构成的 Intranet 越来越多，大量廉价的个人计算装置随处可见，但是它们的资源利用率非常低。挖掘利用由多个计算机机群构成的网格上的空余的计算资源、存储资源、通信资源，可以得到大量的、非专用的、廉价的、大规模的高性能处理和计算资源。这些计算资源可以用来解决组织部门(企业、科研机构、大学、政府)的计算密集型任务(CSCW 环境任务)和海量信息查询(大规模并行搜索器)的需要。因此，这些资源的挖掘和使用理论技术具有重大的应用价值和市场前景。这样由多个自然分布的计算机机群通过 Internet 或 Intranet 连接而成的大规模计算环境被称作多机群网格。这种网格不同于其它著名网格系统(如 Globus, DataGrid)的主要因素是其资源分布的自然性、资源的不可控性、资源的动态性、资源的无边界性和异构性等特点。所以，随着机群节点数目和网络数目的增加，系统的可靠性和资源的动态调动能力就会下降。同时，随着网格资源规模的扩大、网格资源和服务的日益多样化，传统的技术已经无法适合异构的、多可信级别的多机群网格资源管理的需求。Agent[3]和多 Agent[4,5]技术带来了有效的可行的方案。已经有基于 Agent 的网格技术的研究，如从第一个面向计算网格的 Agent[6]，到基于 Agent 的网格组织结构[7]，到基于 Agent 的网格资源管理[8]，以及支持大规模 Internet 管理的多 Agent 系统[9]等。目前有关多机群网格的多 Agent 的理论和技术的研究还不多，需要有效的多 Agent 资源调度模型。多 Agent 的学习机制十分重要，因为 Agent 的智能是通过学习机制[10~14]来完善智能知识，从而提高行为的准确性和正确性。多智能体学习方法有主动学习和被动学习。目前以强化学习[15]为主流的主动学习成为业界研究的热点。本文采用多 Agent 协同合作理论，运用强化学习原理的知识修正机制，对多机群网格系统上的动态资源的调度理论技术进行了探讨，提出了一个基于强化学习的网格资源的调度模型(Multi-cluster Grid Resource Dispatch Model Based Reinforcement Learning-MCGRM)。该模型根据多机群网格上的不同类型的数据并行型计算对计算资源类别需求的不同，研究了多机群网格上的逻辑计算机机群资源的优化分配问题。实践表明，该模型有效地提高了多机群网格的资源利用率。

2　基本定义

2.1　多机群网格

定义 1(计算机机群)　是一个二元组 $CC(Ma, CS, NC)$，其中 Ma 为 CC 主控制器；$CS=\{C_1, C_2, \cdots, C_p\}$ 为 CC 的所有计算节点的集合，NC 为该机群的互联网络。

定义 2(多机群网格)　一个多机群网格(Multi-cluster Grid-MCG)为一个四元组 $MCG(Ma, CCS, NG, R)$，其中 Ma 为 MCG 主控制器，$CSS=\{CC_1, CC_2, \cdots, CC_c\}$ 为计算机机群的集的集合，$NG=\{N_1, N_2, \cdots, N_g\}$ 为连接各个机群网络的

*)国家自然科学基金(60573108)、上海市重点学科建设项目(T0502)、上海教委发展基金(06QZ002,07ZZ92)。**陈庆奎**　教授，主要研究领域为网格、计算机机群、并行数据库。

维普资讯 http://www.cqvip.com

集合，R 为连接规则。

定义 3（*MCG* 上的数据并行型计算 DPC） 为了描述方便，缩写一个数据并行型计算为 DPC(Data Parallel Computing)。

定义 4（DPC 的资源需求权重向量） *MCG* 中的每类数据并行型计算 DPC 对计算节点的计算性能、存储容量、I/O 速度和通信能力的需求是不一样的。对每类数据并行型计算进行分析，可以给出对上述四种资源的需求权重，用向量 $W=(w_1,w_2,w_3,w_4)$，称作该类 DPC 的资源需求权重向量。这样，*MCG* 上的一类 DPC 可以用一个四元组 DPC(w_1,w_2,w_3,w_4)来表示。

定义 5（基本通信单元） 网格 *MCG* 的通信机制需要一个标准的通信单元结构，大量连续的数据被分解为若干个这样的通信单元在网络中传输，称该单元为基本通信单元，其大小记为 N；N 的单位可以是字节或与 DPC 相关的特定长度，本文采用字节数目。

定义 6（基本计算单元） 已知 *MCG* 的计算节点的中央处理器速度、内存容量、高速磁盘速度、网卡都需要统一的度量单位，用向量 $u=(u_{cpu},u_{mem},u_{disk},u_{net})$来表示。所有值可以根据基准测试程序计算获得。基本计算单元的值可以根据 DPC 的需求来定义，例如可以设每百兆赫兹处理器频率为一个 CPU 单位，每兆存储器为一个存储单位等。u_{net} 可以以基本通信单元为单位。

定义 7（机群网络的通信能力） 构成机群 *CC* 的网络 *NC* 带宽为$|NC|$字节，则该机群网络的通信能力为$|NC|/N$，N 为通信单元。

定义 8（多机群网格的网络的通信能力） 构成网格 *MCG* 的网络 *NG* 中的最小网络带宽为$|NG|$字节，则该网格网络的通信能力为$|NG|/N$。

定义 9（计算节点资源提供向量） 相对 DPC 的资源需求权重向量，各个计算节点由于其配置情况的不同，通过基本单元量化（基本通信单元、基本计算单元），可以形成其所提供的各类资源的能力的比例，用权向量 $PW=(p_{cpu},p_{mem},p_{disk},p_{i/o})$，称为计算节点资源提供向量。

定义 10（逻辑计算机机群） *MCG* 中的一个逻辑计算机机群（Logical Computer Cluster）是一个四元组 $LCC(Ma,LCS,B,CC)$，其中 Ma 为 *LCC* 主控制器；$LCS=\{C_1,C_2,\cdots,C_l\}$为 *LCC* 的所有计算节点的集合；*CC* 为 *LCC* 所在的物理计算机机群；B 为该逻辑机群所分配到的 *CC* 的物理网络带宽的一部分。

注：• 由定义可知，一个物理计算机机群可以对应多个逻辑计算机机群，一个逻辑计算机机群只能存在于一个物理机群中。通过逻辑机群的建立，我们可以构造大规模计算环境。

2.2 多 Agent 定义

定义 11（计算节点资源管理智能体 A_{rm}） *MCG* 上的资源管理智能体 A_{rm} 是在 *CCS* 的所有计算节点上的用于资源动态管理的部件；其信念是其所在的计算节点是最具有能力的；其愿望是挖掘其所在节点的计算、存储、通信能力，尽可能提高其所在计算节点在网格中的作用；其意图是根据本节点的资源状况、状态同网格中的其它 Agent 合作、竞争，达到其所辖计算节点的优化工作状态。

定义 12（可靠性管理智能体 A_a） *MCG* 上的可靠性管理智能体 A_a 是存在于在 *CCS* 的所有计算节点上的用于资源稳定性状态监测的部件，主要检测其所在计算节点的 CPU 资源、内存资源、网络资源以及各种服务的工作状态，并根据这些状态的数据来修正所在计算节点的可靠性参数；其信念是相信所在计算节点都会出现各种故障；其愿望是发现其所在节点的计算、存储和通信故障，尽可能降低其所在计算节点在网格中的作用，它与 A_{rm} 是相矛盾的；其意图是根据本节点的资源状态信息，与网格中其它 Agent 合作、竞争，达到其所辖计算节点的优化工作状态。

定义 13（机群管理智能体 A_{cc}） *MCG* 上的机群管理智能体 A_{cc} 是存在于在 *CCS* 中每个机群的 Ma 上，用来对其所辖的所有计算节点的能力进行综合管理，根据它们的计算、存储、通信以及服务能力等进行综合排队、管理，同时负责与其它机群的 A_{cc} 协调、协同工作；其信念是相信其所辖机群能力是最强的；其愿望是发现其所辖机群的尽可能多的计算、存储、通信和服务资源，尽可能地提高其在多机群网格中的地位；其意图是根据本机群的资源状态信息，与网格中其机群的 A_{cc} 协作、竞争，达到其所辖机群的优化工作状态。

定义 14（网格管理智能体 A_{grid}） *MCG* 上的网格管理智能体 A_{grid} 是存在于在 *MCG* 的 Ma 上、用来对其所辖所有机群的能力进行综合管理，根据所有机群的计算、存储、通信、服务能力等进行综合排队、管理，同时负责与机群的 A_{cc} 协调、合作工作；其信念是相信其所辖网格能力是最强的；其愿望是发现所辖网格的尽可能多的计算、存储、通信、服务资源，尽可能提高网格的吞吐率和效率；其意图是根据本网格的资源状态信息，与服务代理 Agent 合作，达到其所辖网格的优化工作状态。

定义 15（用户代理智能体 A_{user}） *MCG* 上的网格管理智能体 A_{user} 是存在于 *MCG* 的 Master 上、用来代理用户的服务请求；根据请求的计算、存储、通信和服务等要求，分配资源；负责与机群的 A_{grid} 协调、合作工作。

定义 16（网格服务智能体 $A_{service}$） *MCG* 上的网格服务智能体 $A_{service}$ 是存在于 *MCG* 的所有计算节点上、用来完成并行计算的特定的计算功能。

每个网格服务智能体都有自己的 BDI，一般情况下，一个 $A_{service}$ 的信念相信是相信其能提供最好的服务；其愿望是尽可能多地提供自身的服务；其意图是寻找所在网格上的对自己有益的最佳资源，并迁移到最佳资源节点，进行优化服务。

3 资源管理 Agent 描述

各个计算节点上的、各个机群上的 Agent 负责资源信息的收集、整理、汇总等工作，并把这些信息汇总到网格管理智能体。这些 Agent 统称为资源管理 Agent。

3.1 计算节点队列

MCG 的计算节点 C_i 有三个状态：主态、竞争态、空闲态。如果 C_i 正在执行某个 *DPC* 的计算过程，那么称其为主态；如果 C_i 处于已经准备接受运行某一服务，但还没有启动执行，那么称其为就绪态；如果 C_i 处于非正常状态，那么称其为失效态；

针对网格的计算节点的上述的 3 个状态，网格 *MCG* 的每个机群 $CC_i(1\leqslant i\leqslant c)$上构造 4 个节点队列，分别是主态队列 $Q_{master}[i]$、就绪队列 $Q_{ready}[i]$、失效队列 $Q_{failure}[i]$。4 个队列的长度分别记为：$LQ_{master}[i]$、$LQ_{ready}[i]$、$LQ_{failure}[i]$。

按照定义所描述的各类 Agent 的 BDI，所有 Agent 通过协作竞争的目的是各自所辖的计算节点、计算机机群尽可能发挥主要作用，即尽可能处于多机群网格的主态队列中；按照这种驱动机制，把三个队列按优先级排队如下：

维普资讯 http://www.cqvip.com

主态队列>就绪队列>失效队列

计算节点的状态级别如下：

主态>就绪态>失效态

3.2 资源管理智能体 A_{rm}

一个网格计算节点 C_i 的性能的衡量参数一般可有下面四个方面：①C_i 当前可提供的 CPU 计算能力 p_{cpu}；②C_i 当前可提供的内存存储能力 p_{mem}；③C_i 当前可提供的网络通信能力 p_{net}；④C_i 当前可提供的磁盘 I/O 能力 $p_{I/O}$。资源管理智能体 A_{rm} 负责动态监测计算节点 C_i 四个参数 p_{cpu}、p_{mem}、p_{net}、$p_{I/O}$ 的变化，利用它们构成计算节点当前时刻的能力函数：

$$p_{node}=f(p_{cpu},\ p_{mem},\ p_{net},\ p_{I/O}) \tag{1}$$

设本计算节点的上一时刻的能力函数值为 pl_{node}，资源管理智能体 A_{rm} 利用公式(1)计算当前时刻的能力函数值 pc_{node}。

如果 $pc_{node}>pl_{node}$，则 A_{rm} 向 A_{CC} 申请向上一级计算节点状态转变。

3.3 可靠性管理智能体 A_a

一个网格计算节点 C_i 的可靠性能的衡量参数一般可有下面两个方面：①当前节点的正常状态标志器 p_{mark}；②计算节点成功完成网格服务智能体的比率：p_{succ}＝计算节点成功完成的服务数目/计算节点所执行的服务总数目。设 l_1 小数，并且 $0<l_1<1$。那么，如果计算节点 C_i 的 p_{mark} 为 $false$，则 A_{rm} 向 A_{CC} 通报 C_i 变为失效状态；如果 $p_{succ}\in(0,l_1)$，则 C_i 变为失效状态；如果 $p_{succ}\in(l_1,1)$，则 C_i 变为就绪状态；

3.4 机群管理智能体 A_{cc}

CC_i 的 A_{cc} 接收来自机群 CC_i 所有计算节点上的 A_{rm}、A_a 定期发来的 p_{node}、p_{mark}、p_{succ}；同时利用心跳侦测技术检测所有节点的存活有效状态。

机群 CC_i 根据自身当前的就绪队列 $Q_{ready}[i]$ 的实际以及空闲网络资源的带宽构造支持 DPC 计算的逻辑计算机机群的集合 $LCCS$；算法如下：

算法 1(逻辑计算机机群形成算法)

输入：绪队列 $Q_{ready}[i]$ 及其长度 $LQ_{ready}[i]$
输出：机群 CC_i 的逻辑计算机的集合 $LCCS$
算法：
①根据 CC_i 的各个计算节点的资源提供向量进行模糊聚集，使资源提供向量相近的计算节点构成一个逻辑计算机机群；这样得到初步的逻辑计算机的机群的集合 $LCCS$；
②[为每个逻辑机群分配网络带宽(通信能力)]
当前机群 CC_i 的空闲网络能力为 SB；
对 $LCCS$ 中的逻辑计算机机群按其规模由大到小排序，得到有序队列 $LCCS[]$；
设临时集合 $TMP=LCCS[]$；
$i=1$；
While $TMP\neq\phi$ and $SB>0$ do
取一个逻辑机群 $LCC\in TMP[i]$；
$NB=LCC$ 所有节点网络卡的通信能力和；
/*分配 LCC 的通信带宽为 NB 的通信能力*/
$SB=SB-NB$；/* 分配网络带宽*/
$TMP=TMP-\{LCC\}$；/*去掉已分配的逻辑机群*/
i++；
End do；
③$LCCS=LCCS\text{-}TMP$；/* 更新 $LCCS$ 为最终 LCC 集合*/
④如果 $TMP\neq\phi$，则释放所有的 TMP 中的机群节点为就绪状态节点，加入就绪队列。
⑤输出 $LCCS$；算法结束。

4 资源调度知识模型

4.1 强化学习与资源调度

强化学习[15] Agent 一般由三个部分构成：状态感知器、学习器和动作选择器。感知器把环境状态 s 映射成内部感知 i；动作选择器根据当前策略选择动作 a 作用于环境 E；学习器根据环境状态的奖赏值 r 及内部感知 i，更新 Agent 的知识策略。E 在动作 a 的作用下将导致环境状态的变迁。其基本原理是：如果 Agent 的某个动作将得到环境的正的奖赏(强化信号)，那么 Agent 以后产生这个动作的趋势便会加强，反之 Agent 产生这个动作的趋势减弱。

资源调度模型需要在各类 DPC 与各个 LCC 之间做出有效的匹配，以便达到资源的优化使用。我们可以通过强化学习机制，来优化资源调度规则。

4.2 资源调度的知识表示

调度信息由三个方面构成：逻辑机群信息、DPC 信息以及二者之间的匹配度。

按照算法 1 的过程可知，逻辑机群信息也可以用一个六元组表示 $ILCC(aw_1,\ aw_2,\ aw_3,\ aw_4,\ n,\ NB)$，其中 aw_1，aw_2，aw_3，aw_4 分别为该机群的 CPU、内存、磁盘和网卡的平均资源提供量；n 为逻辑机群包含的计算节点数目，NB 为其分配到的网络通信能力。

对一类 DPC 来说，其信息可以由资源需求向量来表示，即 $DPC(w_1,w_2,w_3,w_4)$。LCC 和 DPC 之间的匹配度可以用数 d 表示。这样一个匹配知识规则可以表示成：$(ILCC,DPC,d)$。所以，MCG 的匹配知识库是由若干个 $(ILCC,DPC,d)$ 知识规则构成的元组空间。

由于 MCG 资源的异构性、动态性，以及 DPC 需求描述的不精确性，使得这些规则的有效性很低。解决的方法是通过强化学习机制进行修正。

4.3 学习模型

根据强化学习的原理，在对匹配规则 $r(ILCC,DPC,d)$ 选用后，要根据 DPC 在 $ILCC$ 上实际运行的情况来检验 r 的有效程度。其有效度可以根据 $ILCC$ 资源的利用率来检测。如果 DPC 在 LCC 上运行后，资源利用率较高，则说明规则 r 是有效的，否则 r 便无效。进而，可以根据 r 的有效性作为强化学习的奖赏函数。

设 u_1，u_2，u_3，u_4，u_5 分别为 DPC 在 $ILCC$ 上运行后统计出的处理器、内存、磁盘、网络适配器、逻辑机群网络的实际利用率；a_1，a_2，a_3，a_4，a_5 为权系数，且 $a_1+a_2+a_3+a_4+a_5=1$；那么奖赏函数为：

$$p=a_1*u_1+a_2*u_2+a_3*u_3+a_4*u_4+a_5*u_5 \tag{3}$$

设 $y_1(0\leqslant y_1\leqslant 1)$ 是一个有效规则阈值，x 为规则匹配度校正量，那么：

如果 $p\geqslant y_1$，则规则 $r.d=r.d+x$；匹配度增加；

如果 $p<y_1$，则规则 $r.d=r.d-x$；匹配度减弱。

MCG 的匹配智能体将根据规则 $r.d$ 的值的大小来确定下一次匹配动作。

4.4 网格管理智能体 A_{grid}

A_{grid} 接收来自 MCG 所有机群的 A_{cc} 定期发来的 p_{node}、p_{mark}、p_{succ}、pc_{cc}；其主要工作过程如下：

算法 2(多机群网格资源调度算法)

①从所有物理计算机机群接收机群信息；
②驱动所有物理计算机机群构建逻辑计算机机群(按照算法 1)；
③从用户代理智能体 A_{user} 接受 DPC 任务，形成 DPC 队列；
④根据 LCC 集合以及 DPC 队列，查询匹配知识库，选择最佳匹配规则，进行匹配。
⑤驱动所有 LCC 进行 DPC 计算；所有 LCC 统计计算过程中各类资源的平均利用率(如公式(3))；
⑥对所有已经确定实施的匹配规则进行奖赏修正(按照 4.3 节的学习模型)；如果一个规则 r 的匹配度 d 低于某个淘汰阈值 y_2，则删除该规则；
⑦ 执行步骤①。

5 分析及试验

5.1 理论分析

维普资讯 http://www.cqvip.com

定理 1 MCGRM 对多机群网格资源的调度是有效的。

证明：根据强化学习原理，通过环境反馈作为为奖赏来修正 Agent 的行为，可以实时、动态地适应环境的变化，符合 DPC 的实际需求。根据 *LCC* 和 DPC 的定义，它们的匹配度是根据对应资源的实际情况而定的。*LCC* 的状态集是可以通过离散设定而得到有限的状态集合。同理，各类 DPC 对计算资源的需求也是相对稳定的，其状态数目亦是有限的。所以 *LCC* 状态个数与 DPC 状态数目的乘积组合亦是有限的。所以构成 MCGRM 的匹配知识库的规则数目是有限的。通过淘汰阈值(算法 2⑥)的设定，可以减少无效规则的数目，缩小搜索空间。所以，匹配知识库的构建和维护是可行的。

根据知识规则 $r(ILCC, DPC, d)$ 的结构和算法 2 的第⑥步，匹配度 d 的值随着逻辑计算机资源的利用率的提高而增大，规则 r 将被优先使用，即资源利用率高的规则被优先驱动。所以随着学习过程的深入，MCGRM 将会提高 *MCG* 资源的利用率，更适合 *MCG* 的异构、动态的特性。

综上所述，MCGRM 对多机群网格资源的调度是有效的。

5.2 实验

利用一个 1000M 交换机和一个 100M 的交换机构造两个物理机群，每个机群各由 12 个计算节点构成，节点类型如表 1 所示。每个物理机群内各类节点均为三个。在某些 DPC 计算要求下，CT1、CT2 可以模糊化为一类，CT3、CT4 可以模糊化为一类。

表 1 各类计算节点配置信息

节点类别	处理器	内存	硬盘	网卡
CT1	2.0Ghz	256M	5400 转	100M
CT2	2.0Ghz	128M	5400 转	100M
CT3	2.8Ghz	256M	7200 转	100M
CT4	2.8Ghz	128M	7200 转	100M

DPC 类型选取三个代表类型的数据并行性运算{矩阵乘法、迭代法、JOIN 算法}。矩阵乘法代表通信和计算都适中的计算，迭代法代表计算资源比通信资源需求高的 DPC 计算，JOIN 算法代表通信资源比计算资源需求高的 DPC 计算。

根据网络的情况初始假设逻辑机群的规模为 2、4、6、8、14 等，匹配知识库的初始规则有 15 个，后经过测试数据调整为 137 个左右。各类运算分成 7 个阶段被重复驱动运行，每个阶段 12 小时，通过随机函数动态变化 *MCG* 的资源使用。取阈值 $y_1=0.8$，$y_2=0.2$，$x=0.01$，匹配度 d 为小于 1 的正小数。

测试对学习的各个阶段的资源利用率，以及最佳机群规模等进行了记录。

对三类 DPC 计算，随着学习过程的深入，运行它们的 MCG 的资源平均利用率逐渐提高，试验结果如图 1 所示。图中横轴分别代表该模型运行的 7 个阶段(每阶段 12 小时，运行到第 7 阶段时，共累计 84 小时)；纵轴为各个阶段的三类 DPC 计算所对应的 *MCG* 的资源利用率。图中细实线为矩阵乘算法、粗实线为 JOIN、虚线为迭代算法。根据测试结果可知，随着运行阶段的逐步累加(学习时间逐渐累积)，*MCG* 资源的利用率逐渐提高，并达到一定的稳定值；又由于 JOIN 对通信资源的需求过高，其平均资源利用率最低，是因为网络成为通信的瓶颈。

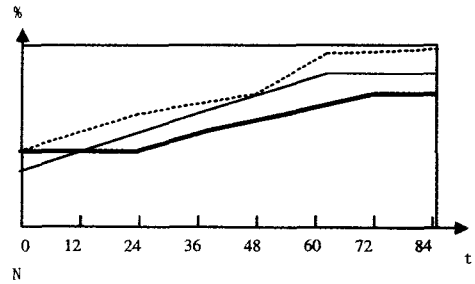

图 1 三类 DPC 的资源利用率变化

对三类 DPC 计算，在普通网络情况下，测试了 LCC 的有效规模，测试结果如表 2 所示。

表 2 各类 DPC 适合的 LCC 规模

矩阵乘法	迭代法	JOIN 算法
8	10	5

结束语 在由多个物理机群构成的多机群网格环境下，为了有效地支持 DPC 计算，采用逻辑机群技术是一个有效的资源划分方法。通过 LCC 和 DPC 各类资源的有效匹配，来提高 MCG 资源的利用率。由于 MCG 的动态性、异构性，这种匹配变得困难。基于强化学习的知识规则修正机制可以有效地解决这一问题，本文研究证明了这一点。作为初步结果，还有许多问题有待研究，如规则的自动生成问题，LCC 跨物理机群问题等。

参 考 文 献

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

2 Segal B. Grid Computing: The European Data Project [A]. In: IEEE Nuclear Science Symposium and Medical Imaging Conference [C], Lyon, 2000. 15～20

3 Tokoro M. Computational Field Model: Toward a New Computing Model/Methodology for Open Distributed Environment [C]. In: Proceeding of 2nd IEEE Workshop on Future Trends in Distributed Computing System, Sept. 1990

4 Osawa E. A Scheme for Agent Collaboration in Open MultiAgent Environment [C]. In: Proceeding of IJCAI'93, August 1993. 352～358

5 Wooldridge M. An Introduction to Multivalent System [M]. John Wiley & Sons (Chichester, England). ISBN 0 47149691X, February 2002

6 Manola F, Thompson C. Characterizing the agent grid [OL]. http://www. objs. com/agility/techreports/990623-characterizing-thr-agent-grid. html, June 1999

7 Foster I, Kesselman C, Tuecke S. The Anatomy of the Grid: Enabling scalable virtual organizations [J]. International Journal on High performance Computing Applications, 2001, 15(3): 200～222

8 Rana O F, Walker D W. The Agent Grid: Agent-based resource integration in PSEs [C]. In: Proceedings of the 16th IMACS World Congress on Scientific Computing, Applied Mathematics and Simulation. Lausanne, Switzerland, August 2000

9 Overeinder B J, Wijngaards N J E, van Steen M, et al. Multi-agent Support for internet-scale grid management [OL]. http://www. cs. vu. nl/pub/papers/globe/asib-grid. 02. pdf

10 Kaelbling L P, Littman M L, Moore. Reinforcement learning: A survey [J]. Journal of Artificial Intelligence Research, 1996, 4 (2): 237～285

11 Sutton R S. Learning to predict by the methods of temporal differences [J]. Machine Learning, 1988, 3: 9～44

12 Dayan W P. Q-learning [J]. Machine Learning, 1992, 8(3): 279～292

13 Rummery G, Niranjan M. On-line Q-learning using connectionist systems: [Technical Report]. CUED/F-INFENG/TR 166. Cambridge University Engineering Department, 1994

14 Horiuchi T, Katai O. Q-PSP learning: An exploitation-oriented Q-learning algorithm and its applications [J]. Transactions of the Society of Instrument and Control Engineer, 1999, 39(5): 645～653

15 高阳，陈世福，陆鑫. 强化学习研究综述[J]. 自动化学报，2004，30(1): 86～100