

# 一种基于角色 Agent 的教育网格任务调度策略<sup>①</sup>

## Task Scheduling Policy Based on Role Agent in Educational Grid

周燕霞 (浙江东方职业技术学院 工程技术系 浙江 温州 325011)

**摘要:** 针对教育网格任务调度存在的困难,提出了一种基于角色 Agent 的教育网格任务调度模型。该调度模型将教育网格资源进行角色化分层组织与分解,并对教育网格任务进行角色化任务分解。在 OptorSim 网格模拟系统中检验了该角色 Agent 调度模型的优越性,为教育网格的任务调度提供了新的解决方案。

**关键词:** 教育网格 任务调度 角色 Agent

### 1 引言

网格的最终目标是实现对网格资源的优化使用,以完成用户提交的任务,为用户提供更好的服务质量。在其实施过程中,系统必须根据任务的资源需求和网格资源状态,对任务所要求的资源进行选择 and 分配,并进行任务调度与控制。

任务调度就是在任务和资源之间建立一种映射关系。任务调度是网格处理环境中的一个重要环节,一个好的任务调度机制可以很好地实现任务的合理分配及资源的有效利用,从而保证网格中各部件协同工作,为用户提供良好服务。因此,网格环境下的任务调度研究是一项很有意义的工作。

### 2 教育网格<sup>[1,2]</sup>任务调度难点分析

为任务分配合适的资源,同时又要平衡资源之间的工作负载,是任务调度追求的最终目标。结合教育网格的资源特点及任务特征,其任务调度存在如下难点:

(1) 在计算资源方面:在教育网格中,要为任务分配合适的资源,首先要发现适合的资源。同时,还要制定合理的优化调度策略。在任务调度之前,首先要对资源的工作状态有充分的了解。教育网格中的资源发现与资源监控都是相当困难的。这是由网格资源的多层次异构性、资源使用者的异构性、资源拥有者的自治性、动态变化性、资源分配决策的分布性等特

点所导致的。

(2) 在计算任务方面:教育网格中的任务计算量非常巨大,同时还涉及巨大的数据传输量。因此将一个巨大的任务调度到一个计算节点去执行,在数据传输上要花费很多的时间,这种做法是不可取的。如果能将任务分解,发现其中的并行子任务,不但可以并行传输数据,而且可以使这些子任务得以并发执行,这样将有效缩短任务的执行时间。在教育网格的任务调度方面,要制定合理的任务分解方案,以提高任务调度的效率。

(3) 在优化算法方面:在教育网格环境中,准确预测任务执行时间和通信延迟是非常困难的。这是因为教育网格中的资源存在异构性和动态性。在教育网格的优化调度策略中,应该能够根据资源的异构性和动态性,做出自适应调整。

### 3 基于角色 Agent 的任务调度策略

#### 3.1 角色的基本定义

角色理论认为<sup>[3]</sup>,角色是某一类对象结构、性质、行为、职能等方面所共有的特征集合,能够作为事物分类的合理标准。本文将角色描述为模式解决方案中的一个具体元素,角色的结构对应于解决方案中的元素结构。角色被组织为层次结构,以使得角色可以包含子角色。模式使用者创建、修改、定位元素,从而使角色得以满足。

<sup>①</sup> 收稿时间:2009-03-26

在教育网格系统中，一个处理某类型信息的网格节点可以被定义为一种角色 **R**，其可以被看作在网格系统中特定位置个体的一系列控制动作的意识态度，包括关于该角色的信念、目标和意图等等。

角色特征的语法定义如下：

```
<Role_Model> ::= "ROLE"
<ID> [<Description>] <Goals> <Qualifications>
<Relationship> <Obligations> <Concurrency_Constraints>
<Permissions> <Protocols> <Priorities> <Methods>
"END_ROLE"
```

其中：**ID** 用来唯一标识角色 **role**；

**Description** 是对 **role** 在系统中权利和义务的概要描述；

**Goals** 是担当该角色对象所负责的一个或多个目标；

**Qualifications** 限定对象实现目标必需的技能和一个或多个必要的预备条件；

**Relationships** 是系统中该角色和其它角色的多个关系；

**Obligations** 是一组职责，每个 **Obligation** 均规定对象必须执行的一个特定行为；

**Concurrency\_Constraints** 提供一组并发机制，如竞争机制、转移机制等；

**Permissions** 是一组许可令或禁止令。一个 **Permission** 表明在某一机器上执行的一个特定操作被允许发生；而一个 **Prohibition** 则表明某一机器上的某一操作是不允许的；

**Protocols** 是一组用以控制对象间通信的规则；

**Priorities** 优先权用于限定扮演角色的对象使用某一种资源的优先顺序；

**Methods** 是一组方法，每一种 **Method** 制定了一种对象导航的方法。

### 3.2 角色 Agent

角色可引入到多 **Agent** 系统分析和设计中，作为连接多 **Agent** 系统宏观模型与微观模型的桥梁<sup>[4,5]</sup>。角色 **Agent** 被理解为接受信息、加工信息和发送信息的 **Agent** 对象。从概念角度上讲，角色是一种约束，在这种约束下，**Agent** 参与某些交互并以某种方式进

行演变；从执行角度上讲，角色是 **Agent** 的某些属性和行为的封装，并且绑定在 **Agent** 上。

整个基于角色的过程主要从活动的角度来讲，工作流程由活动及其逻辑关系组成。从满足用户的要求分析，工作流程的基本组成元素是角色及其协调关系，每一角色有输入与输出参数集。复杂的流程是由各种各样的角色智能体组成，它们具有不同的职能、能力和任务，相互协作完成流程的目标。活动及其关系只是角色及其合作的外在表现，流程本质上是角色的集合。如图 1 是角色的处理流程图：

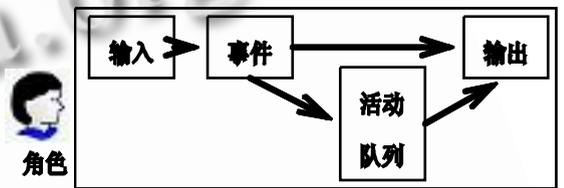


图 1 角色处理流程图

通过 **Agent** 来设定一个角色所采用的这些基于角色的态度。角色除了为 **Agent** 提供指导其动作的目标和意图，还可以为 **Agent** 提供一些执行动作时所需要的一些信息。采用一个角色时可以有二类意识状态：必选态度和可选态度。一方面，必选态度是当 **Agent** 承担某一角色时必须接受的意识状态；另一方面，可选态度和角色并没有密切的关联，当 **Agent** 承担某一角色时可以选择是否接受该意识状态。这样说来，角色对 **Agent** 的思维状态具有深远的影响。

### 3.3 基于角色的 Agent 模型定义

一般来讲，一个角色 **Agent** 的完整定义应该包含局部数据、历史经验库、处理过程和处理机几个部分。其中，局部数据是私有的，只有具有相应存取权限的外部用户才能通过请求的相应服务来访问这些数据。历史经验库定义处理事务的经验、知识规则等，**Agent** 的处理过程可以根据历史经验库中的知识规则或历史经验有效地完成任务或者不断地完善自身。处理过程是对外的窗口，所提供的服务都是通过对处理过程的调用请求而实现的。

在本文提出的调度模型中，提出了下列几种类型的角色 **Agent**，用来控制任务的处理过程，如图 2 所示：

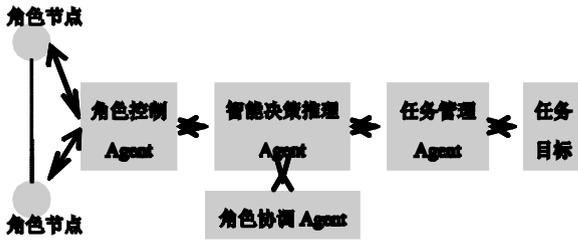


图 2 基于角色的 Agent 模型

一个处理任务创建之后,任务管理 Agent 根据用户传递的信息来触发智能决策推理 Agent 以及角色协调 Agent,以确定当前执行实体的角色,决定任务的下一个执行角色(如果为空,即尚无用户执行操作,则默认为 0)。然后触发角色控制,将该角色的角色名传递给它。

角色控制将提取该角色的相关信息,同时确定下一个执行实体,将角色和下一个执行实体同时反馈给任务管理 Agent。

任务管理 Agent 为该执行实体创建用户,将相关信息交给该用户(包括执行实体需要的资源数据、任务列表、执行任务起止时间、需要输出的数据、任务完成可采取的动作等),并为其提供相关数据的访问接口。用户为执行实体提供操作接口和必要的服务(如从系统中调取数据和向系统中写入数据等),由执行实体完成任务。执行实体完成任务后,可能采取不同的动作,用户将收集执行实体的输出数据及采取的动作,反馈给角色节点。

### 3.4 基于角色 Agent 的任务处理过程

本文中整个教育网格服务被看作为一个目标,对复杂任务处理的实施过程一般是通过目标的分解、角色任务的分配及活动的执行来完成的。角色之间通过目标分解和活动执行建立相互关系。

从宏观上看,教育网格可以视为高抽象层次的角色。整个网格系统的服务目标又是由多个核心流程完成。从系统内部来看,每个流程可以由承担不同角色的网格节点来完成。从角色的观点来说,角色的确定依赖于网格目标的分解,高层角色负责高层目标。具体的任务处理过程如图 3 所示。

角色是个层次概念,其粒度要和目标分解的粒度相对应。根据工作流程的组织范围,小到一个工作线程、进程,大到一个网格节点,都担负一定的角色。

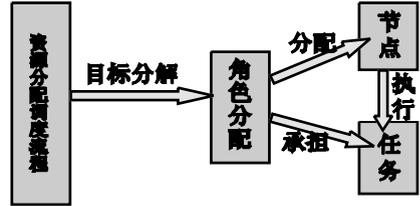


图 3 基于角色的任务处理图

任务目标也是个层次型概念,根据一定规则可以逐级分解,直到原子目标(单个线程可以承担的任务)。目标的实现依赖角色的活动,活动和角色是可分解的。角色的分解和任务目标的分解对应,并由参与者负责相应层次的活动。角色和活动的层次性是和目标层次性相对应的。高层目标可能涉及不同的角色,角色负责成员的活动分配。

## 4 引入角色 Agent 的优势

将角色 Agent 引入调度模型中存在如下优势:

(1) 效率优势:角色 Agent 机制可以通过引入角色 workflow 分解,使任务量按照供需情况浮动,即可做到量力而行,最终达到一种对资源供需双方都有利的均衡状态。

(2) 协作优势:在整个教育网格系统任务调度中,可以把不同的角色节点作为协作关系的载体,利用角色间的相关继承关系、连接关系、组合关系构成群组协作行为的多个方面,以角色为依据描述和构造群体协作机制中的群体感知、协调、通讯和协作控制机制。

(3) 应用角色强化 Agent 的特征<sup>[6]</sup>:如果能给 Agent 确定某种角色,那么其行为的选择空间将会减少很多。理想的方法是对任一个状态,只提供给 Agent 一个或两个可能的行为选择,这样 Agent 将很容易进行决策,而且是反应式的。

(4) 引入角色能够减少复杂度<sup>[6]</sup>:在多 Agent 系统中,减少复杂度意味着减少推理。虽然它们可能缺乏智能性,然而对整个多 Agent 系统来说,角色之间有效的组合将增加系统的有效性、反应性和鲁棒性。此外,Agent 可以根据队友的角色来预测它们的下一步动作。

## 5 任务分解和调度

在本文提出的任务处理过程中,首先要对教育网格中活动的任务进行有效的分解,才能发挥基于角色 Agent 的任务处理效率,而调度是实现教育网格资源透明服务的保证。

### 5.1 基于角色 Agent 的教育网格资源组织和分解

教育网格中的数据资源种类繁多,数据量大,其静态信息由资源信息服务提供。资源的静态信息是指资源的属性值不随或很长时间内不随时间变化。虽然静态信息不反映资源当前的能力。但是它从一定方面反映了资源的特征,而这些特征对于资源选择是相当重要的。

由于教育网格资源的动态性,资源的静态信息不是绝对不变化的,因此必然要求所提供的信息服务具有一定的可扩展性和动态性。资源信息服务应提供资源的动态信息。资源信息模型主要实现虚拟资源、软件及物理设备的逻辑表示。

本文提出的角色资源信息模型中,可以设置此类资源的属性中哪些是必须在每个资源信息记录中存在的,哪些是可以选择的。这种方式使资源信息库具有了极大的灵活性和更宽的适应面。设计者可以将各类资源的属性范围扩大到最大值,但是限定几个最基本的属性名,这样使用者可以用这些“角色”定义出系统需要的“对象”,也就是一个个资源信息实体。这些资源信息实体的属性范围也是随当时的需求而定,并非完全一致。使用者也可以在后来的应用过程中随时添加需要的属性值。角色节点整体结构如图 4 所示:

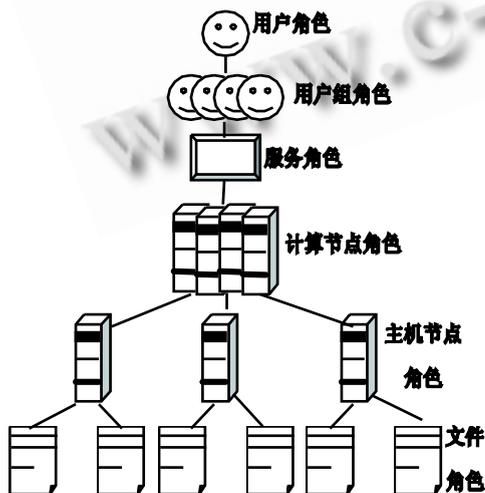


图 4 教育网格角色资源信息服务模型结构

其中,计算节点是构成教育网格的主体,在资源信息服务中保存它们的信息是非常重要的;由于教育网格之间的访问和互操作接口主要通过 Web 服务的方式来进行,因此,利用服务角色来保存服用接口信息是非常必要的;另外,可以将具有相同工作的用户划分为一个用户组,以便统一管理;其中,用户角色描述了使用者的姓名,账号,权限等。

在定义好角色信息模型后,还需要定义命名模型来描述资源的定位以及资源信息之间的关系。一般可以按照树型结构组织教育网格资源信息。每个资源信息条目以树型结构中的位置来命名。

### 5.2 基于角色 Agent 的教育网格任务分解

如果将一个巨大的计算任务,不经分解就直接分配给某个计算节点去执行,必将导致任务执行速度慢,且其他计算节点得不到充分利用。

基于此,本文引入了角色任务分解,其目的就是充分开拓任务的并发性和可扩放性。需要注意的是,角色任务分解是将任务分解成角色对象资源,而这里的分解不仅是并行分解,其中还包括串行分解。

不同要求的计算子任务分配给合适的计算节点,这样可以节省计算费用。发现其中可并行的子任务,使一部分计算子任务分配到不同的计算节点上并行执行,可以提高计算速度和资源的利用率,同时降低单个计算节点的工作负载。

根据计算节点的处理能力,为其分配工作负载大小合适的计算任务。任何计算都需要数据,在计算开始之前要把数据传到相应的计算节点,因此用户的计算任务可以首先划分成数据传输子任务和计算子任务,数据传输子任务需要消耗存储资源和通信资源,计算子任务则要消耗网格的计算资源,而这两种任务又可以按照一定的原则进行再分解。

教育网格中各种功能的实现需依托各个网格节点,各个类型的角色节点群又有其子任务角色节点。例如可以划分任务处理角色节点群、任务分配角色节点群和任务调度角色节点群,每种类型的角色节点群还包含若干角色节点。

## 6 网格模拟器试验分析

要取得最佳的资源调度效果,除了需要有效的任务调度模型之外,还需要结合高效的资源调度优化算法<sup>[7]</sup>。由于蚁群算法中人工蚂蚁的划分思想能灵活的

应用于角色 Agent 调度模型中。因此本文提出将自适应信息素改进蚁群算法应用于教育网络的调度优化中，并在 OptorSim 网络模拟系统中检验了该角色 Agent 调度模型的优越性。通过对 OptorSim 中自带的 cms 数据进行测试，结果分析如下：

(1) 网络文件转换率分析：网络文件转换率<sup>[8]</sup>越高，表明网格调度中分解的文件的失误率越高(转换率越低表明，文件都已经分解到位，不需要进行额外的转换)通过图 5 可以看出该调度模型的转换率比较低，而且随着时间增长其转换率总体比较平衡。

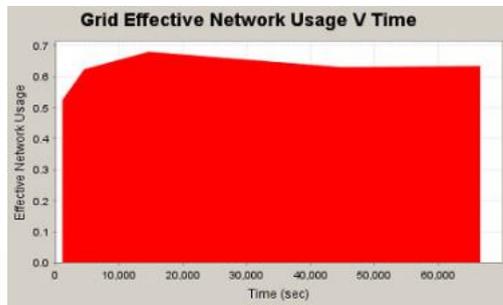


图 5 角色 Agent 调度模型的网络文件转换率

(2) 网格作业平均消耗时间分析：通过图 6 可以看出，随着时间的增长，该调度模型的作业平均消耗时间不大。



图 6 角色 Agent 调度模型的网格作业平均消耗时间

(3) 网格工作量消耗分析：通过图 7 可以看出，随着时间的增长，该调度模型的工作量消耗是极小的。

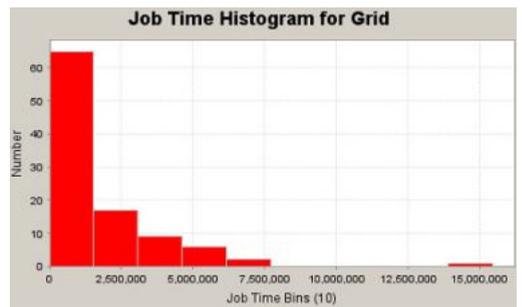


图 7 角色 Agent 调度模型的网格工作量消耗分析

## 7 结语

教育网格任务调度必定是教育网格研究的重点和难点。由于作者的水平有限，本文的研究只能解决其中的某几个方面的问题。基于角色 Agent 的调度模型本身虽然存在很多的优势，但由于是初步应用于教育网格系统分析中，模型的实现问题还需进一步的研究和探讨。

### 参考文献

- 1 李国杰. 信息网格——未来的互联网应用. 中国科学院计算所网, 2002.
- 2 金江军. 网格技术在教育行业的应用. 北京大学: 中国国家教育网格, 2004.
- 3 赵丽. 角色在多 Agent 系统中的应用. 哈尔滨师范大学自然科学学报, 2006, 22(2): 75 - 78.
- 4 Lander SE. Issues in multivalent design systemize. Expert, 1997, 12(2).
- 5 王丽侠, 梅成才. 基于多 Agent 协作的下载模型研究. 浙江师范大学学报, 2005, 28(2): 168 - 171.
- 6 范植华, 白光野, 等. 角色编辑器 Rib-MDE 的设计与实现. 计算机学报, 1996, 19(5): 398 - 400.
- 7 Rafael AM. Job Scheduling and Resource Management Techniques in Dynamic Grid Environments, 2003.
- 8 CIO J, Kerbyson DJ, Nudd GR. Performance evaluation of an agent-based resource management infrastructure for grid computing. Proc. of the 1st IEEE Int. Symp. on Cluster Computing and the Grid. 2001. 311 - 318.