第 25 卷第 4 期
2005 年 4 月

计算机应用
Computer Applications

Vol. 25 No. 4
Apr. 2005

文章编号:1001 - 9081(2005)04 - 0799 - 03

基于消息报告机制的多 Agent 调度系统研究

熊　成[1,2],罗小宾[1,2],王小敏[1,2],张家树[1,2]
(1. 西南交通大学 信号与信息处理四川重点实验室,四川 成都 610031;
2. 西南交通大学 计算机与通信工程学院,四川 成都 610031)
(xiongcheng11@sohu.com)

摘　要:针对生产车间调度系统中因资源庞大以及动态变化而引起的监管困难问题,构建了一种基于消息报告机制的多代理调度系统分层体系结构。根据 Agent 在调度系统中的作用和角色的不同,将 Agent 划分为管理 Agent、任务 Agent 和资源 Agent,并通过一个具有消息报告机制的管理 Agent 来有效管理和监控系统的运行。实例分析表明,消息报告机制不仅能保证系统的有效运行,而且能提高系统的工作效率。

关键词:多代理;多代理调度系统;消息报告机制

中图分类号:TP273[+].5　　**文献标识码**:A

Study on multi-agent scheduling system based on message report mechanism

XIONG Cheng[1,2], LUO Xiao-bin[1,2], WANG Xiao-min[1,2], ZHANG Jia-shu[1,2]
(1. *Sichuan Province Key Lab of Signal and Information Processing, Southwest Jiaotong University, Chengdu Sichuan* 610031, *China*;
2. *College of Computer Science and Communication Engineering, Southwest Jiaotong University, Chengdu Sichuan* 610031, *China*)

Abstract: Aimed at management difficulty caused by enormous resource of scheduling system, a kind of hierarchical architecture of MASS (Multi-Agent Scheduling System) based on message report mechanism was designed. According to the difference of role and function in the MASS, Agent was divided into manage Agent, task Agent and resource Agent. In addition, the system operation was managed and monitored by a management Agent with message report mechanism. Results of instance analysis show that the message report mechanism can not only assure the effective operation of system, but also improve the efficiency of system.

Key words: multi-agent; multi-agent scheduling system; message report mechanism

0 引言

敏捷制造的实质是通过人、组织、管理、技术的高度集成,实现对客户需要的市场做出迅速的反应。敏捷制造的组织形式是动态联盟或虚拟企业,动态联盟就是在虚拟的模式下如何进行组织管理,包括如何组织动态联盟、如何重构车间和单元、如何安排计划、如何进行调度等问题。其中车间作业调度是实现生产高效率、高柔性和高可靠性的关键,有效实用的调度方法和优化技术的研究和应用已成为先进制造技术实践的基础。

车间作业调度问题的研究方法主要有经典调度理论和智能方法。经典调度理论的核心问题就是对调度算法的研究,它的局限性就是算法会随着生产环境的变化而失效。智能方法主要包括专家系统、人工神经网络以及多 Agent 系统技术。其中专家系统存在知识获取和推理速度这两个瓶颈;人工神经网络存在训练速度慢、探索能力弱等缺点[1]。多 Agent 系统技术由于系统中 Agent 具有高度自治性、智能性、协作性、交互性、适应性等特点,较好满足了敏捷制造环境下车间生产管理的需要,并且多 Agent 系统技术能将车间作业调度系统的研究从传统的集中式转向分布式。基于多代理技术的调度系统即为多代理调度系统(Multi-Agent Scheduling System, MASS)。

文献[4]的多 Agent 车间调度系统采用了分层体系结构,当系统运行时,管理 Agent 即要监管任务 Agent,同时又要监管资源 Agent。在这种情况下,若车间的设备资源数量庞大且车间设备动态变化时,由一个管理 Agent 对系统进行实时监控会因为其通信量过大使系统的工作效率降低。为此,本文借鉴文献[5]中工作流技术中的消息报告机制,提出了一种基于消息报告机制的 MASS 分层体系结构,当系统运行时,管理 Agent 只监控少数任务 Agent,任务 Agent 对与其发生关系的资源 Agent 进行监控,从而达到通过一个管理 Agent 来有效管理和监控动态变化系统的目的。实例分析表明,此种方法适合资源数量庞大、动态变化环境下的车间调度系统,具有良好的开放性和动态可重组性。

1 MASS 的体系结构

1.1 多 Agent 系统

多 Agent 系统(Multi-Agent System, MAS)由于系统中 Agent 具有高度自治性、智能性、协作性、交互性、适应性等特点,较好满足了敏捷制造环境下车间生产管理的需要,并且多 Agent 系统技术能将车间作业调度系统的研究从传统的集中式转向分布式,适应了敏捷制造环境下资源分布性和功能分

收稿日期:2004 - 09 - 21;**修订日期**:2004 - 11 - 23
作者简介:熊成(1977 -),男,江西南昌人,硕士研究生,主要研究方向:敏捷制造及多代理技术;　罗小宾(1966 -),男,四川南充人,博士,主要研究方向:先进制造、图像处理及信息安全;　王小敏(1973 -),男,江西萍乡人,博士研究生,主要研究方向:先进制造及信息安全;　张家树(1965 -),男,四川西充人,教授,博士生导师,博士,主要研究方向:图像处理及混沌信号处理.

布性的要求,并可以将复杂问题分解为多个较小的相关子问题,以降低系统设计的复杂性。正因如此,MAS 的协作调度技术被广泛用于车间作业调度领域。

1.2 MASS 中功能代理的划分

根据代理在 MASS 中的作用或角色不同,可将代理划分为管理 Agent、任务 Agent 和资源 Agent。

1.2.1 管理 Agent

管理 Agent 就是一个 MASS 的管理者,主要负责任务规划、管理,并协调和监控其他任务 Agent,其结构如图 1 所示。它首先通过人机接口接收用户的产品定单,然后将定单分解为多个任务,并产生任务 Agent 来完成各个任务的执行。管理 Agent 通过通信接口监控各个任务 Agent 来管理和监控调度系统的运行。

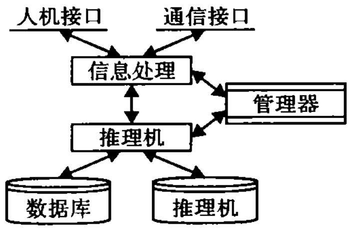

图 1 管理 Agent 结构

管理 Agent 用 BNF(Backus-Naur-Form 巴科斯范式)[1]表示的一个八元组为:

< *ManageAgent* > ::= < 标识符 > < 人机接口 >
< 通信接口 > < 数据库 > < 知识库 >
< 管理器 > < 推理机 > < 信息处理 >

1.2.2 任务 Agent

任务 Agent 由管理 Agent 产生,接收管理 Agent 分解的任务,将任务消息通知资源 Agent 并对它们进行招标。根据资源 Agent 的招标信息,选择合适的资源 Agent 来完成任务。一般包括以下过程:

(1)接收管理 Agent 的任务信息;

(2)向相关的资源 Agent 发出招标信息;

(3)搜集各资源 Agent 回应的标书;

(4)评价各标书;

(5)根据对标书的评价结果选择资源 Agent,并通知其他落选的代理;

(6)系统运行时,分阶段向管理 Agent 发送信息;

(7)提交任务。

1.2.3 资源 Agent

资源 Agent 与生产资源相连接,代表资源与外界交谈。它们能够根据自身的能力、运行状态、任务要求等信息进行判断,并向任务 Agent 投标竞争工序操作任务,因此一般都有推理、规划等模块以及相应的知识库。它主要产生以下动作或反应:

(1)报告自己当前的能力或可提供的处理时间;

(2)搜集所有的任务招标信息;

(3)生成标书;

(4)对任务招标信息进行整理和评价;

(5)接受任务。

资源 Agent 的结构用 BNF 表示的八元组为:

< *ResourseAgent* > ::= < 标识符 > < 设备接口 >
< 通信接口 > < 数据库 > < 规则库 > < 规划器 >
< 推理机 > < 信息处理 >

1.3 MASS 的分层体系结构

在 MASS 的分层体系结构中,各个 Agent 之间采用松耦合的形式,单个 Agent 的故障仅仅引起局部效率的降低或功能的丧失,不会影响其他的 Agent 的正常运行,提高了整个系统的可靠性和容错能力。

在现有的调度系统中,当系统运行时,管理 Agent 既要监管任务 Agent,又要监管资源 Agent。管理 Agent 直接监控资源 Agent,会使管理 Agent 的通信量过大,特别是定单较大,任务工序较多,所需的资源 Agent 庞大时,管理 Agent 监控的通信成本就很高,同时可能涉及资源 Agent 的保密性。故本文提出一种基于消息报告机制的分层体系结构,如图 2 所示。在这个体系结构中,管理 Agent 的主要负责是同外界交互、任务规划和全局管理。管理 Agent 只需要在任务 Agent 上设计合理的消息报告机制,然后监控各个任务 Agent,对资源 Agent 的功能、执行状态等信息不需要直接监控。通过这种分层监控结构,管理 Agent 能够实现对资源庞大且动态变化的调度系统有效地进行管理和监控。

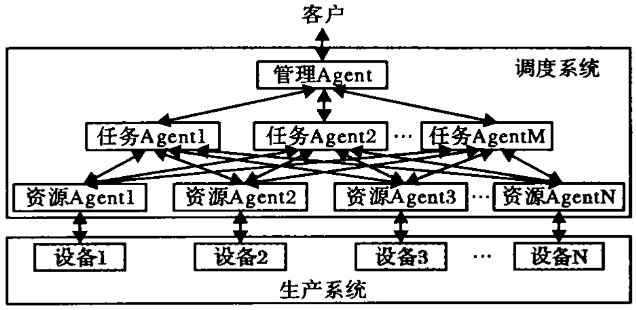

图 2 基于多代理的调度系统分层体系结构

1.4 基于招投标机制的谈判协议

管理 Agent 从客户或生产计划系统中接收到定单时,通过自身的判断推理,将定单分解为多个并行或串行的任务,并动态产生任务 Agent 负责相应的任务,每个任务可以有多个工序组成,每个工序根据各个工序的要求,由任务 Agent 向资源 Agent 进行招标,资源 Agent 根据自身的能力和状态,产生一个标书进行投标,任务 Agent 在规定时间内,比较收集到的标书,选择一个资源 Agent 来执行工序,并通知没有中标的资源 Agent。图 3 即是一个资源 Agent 和一个任务 Agent 的招标—投标—中标过程。

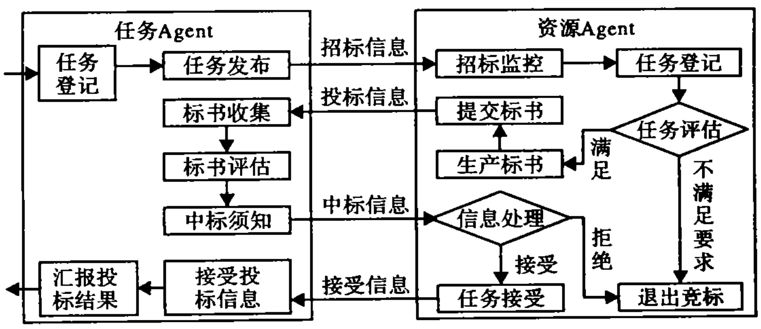

图 3 一个资源 Agent 和一个任务 Agent 的招投标过程

2 消息报告机制的监控机理

建立分层体系结构的目的就是为了管理 Agent 更好地管理和监控调度系统的有效运行。要使系统有效运行,就必须对系统运行过程中任务的各个工序进行监控,确保任务按计划完成。如果管理 Agent 失去了对任务的各个工序的监控,

就会失去对整个过程的控制。

在分层体系结构中,由于管理 Agent 不直接监控每个资源 Agent 的状态和具体操作运行,为了监控资源 Agent 上工序的运行情况,本文采用消息报告机制,通过任务 Agent 来间接监控资源 Agent 的工序运行情况。管理 Agent 把任务划分为多个工序,每个工序完成后,任务代理就要向管理 Agent 发送一个消息报告工序完成。当设备出现故障使资源 Agent 不能完成计划任务时,任务 Agent 也要及时向管理 Agent 发送消息报告故障。管理 Agent 通过监控这些消息的结果就可以间接监控资源 Agent 的运行状态。图 4 是消息报告机制原理图。

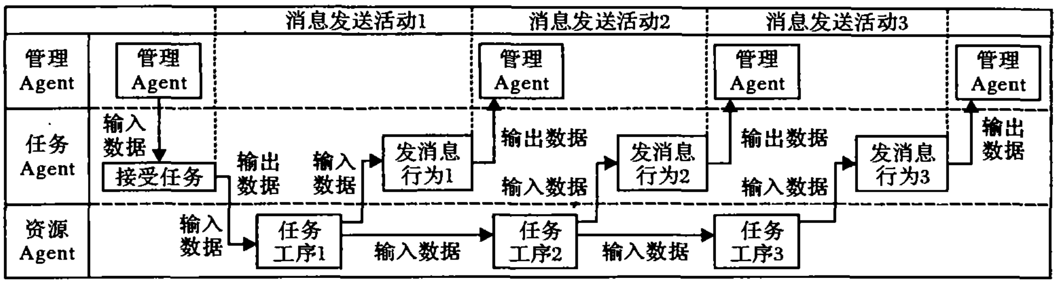

图 4　消息报告机制原理

3　消息报告机制实例

某制造企业接收到一个客户的产品定单,并启动其生产调度系统来完成,如图 5 所示。管理 Agent 接收到产品定单 a 后,将其分解为产品设计任务 b 和产品加工任务 c,并产生相应的任务 Agent1 和任务 Agent2;任务 Agent 根据资源自身的状态和能力评估,通过与资源 Agent 进行招投标的方式将任务的各个工序外包给相应的资源 Agent,并发送接受任务的消息通知给管理 Agent;管理 Agent 再回复客户一个接受定单的通知。任务 Agent 完成任务 b 和任务 c 以后,发送任务完成消息给管理 Agent,任务 Agent 就自动消除。管理 Agent 再发送 a 给客户,最后管理 Agent 开发票和客户付款。

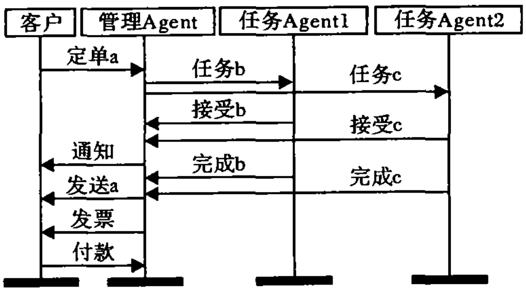

图 5　某企业生产调度系统的信息交互序列

任务 Agent1 在接收到产品设计任务 b 后,将起分为三个设计工序:产品概念设计、产品详细设计和产品工艺设计。每完成一个工序,任务 Agent1 都要及时向管理 Agent 发送消息报告工序完成。下面既是一个任务 Agent1 完成产品概念设计后给管理 Agent 发送的消息代码:

```
…
<action_name>概念设计方案</ action_name >
<action_ID = action_ID_001 >
    <sender>TaskAgent1 </sender>
    <receiver>ManageAgent </receiver>
    <files>概念设计文档 </files>
    <action >finished </action>
    <action_send_time>
        <second>00 </second> <minute>00 </minute>
            <hour>10 </hour>
        <day>30 </day> <month>8 </month>
        <year>2004 </year>
    </action_send_time>
<action_ID>
…
```

在系统运行时,任务资源 Agent 能实时监控相应资源 Agent 的执行状态。若因为某个设备的损坏,产品概念设计不能按计划完成。此时任务 Agent 就能及时(而不用等到此工序计划完成时间到达)知道故障,并立即向管理 Agent 发送消息报告故障。因为系统采取的是松耦合结构,单个 Agent 不能正常工作不会影响其他 Agent。如此管理 Agent 就可以通过局部调整计划,通过对产品概念设计工序进行再招标。

如果管理 Agent 对任务 Agent1 的产品概念设计不满意,可发出挂起产品详细设计信息并重新启动产品概念设计信息。任务 Agent1 在收到这两个消息后,就随之挂起相应资源 Agent 的产品详细设计工序,并重新开始进行产品概念设计。当然如果管理 Agent 对概念设计虽然不满意,但认为概念设计虽需要修改但对详细设计不影响,也可以发送继续详细设计信息并重新启动概念设计信息。上述对两个信息采取的控制命令包括"挂起"、"回滚"和"继续"等控制消息。由此可见,管理 Agent 可以通过消息报告机制能够实现对系统的有效运行。

通过消息报告机制,管理 Agent 不但能保证 MASS 按计划运作,同时还能提高整个 MASS 的工作效率。假设按计划任务 Agent1 需要三天完成的产品设计任务 b,由于该任务 Agent 监控下的资源 Agent 的设备更新,任务 b 一天内就可以完成。这样任务 Agent1 可以在第一天内完成,而到第三天才将结果报告给管理 Agent,也可以在第三天才开始执行任务。也就是说某个资源 Agent 采用了先进设备提高了这个任务 Agent 工作效率,但整个 MASS 的工作效率并没有提高。通过消息报告机制,任务 Agent 在完成其任务后会及时地向管理 Agent 报告,管理 Agent 就能够及时了解任务 Agent 的任务完成状态。当某个任务被提前完成时,管理 Agent 即可以提前启动其后序任务。因此,当系统中某个任务 Agent 工作效率提高时,整个 MASS 的工作效率也能得到提高。

多代理系统技术应用于车间作业调度系统是一个行之有效的方法,采用多代理调度系统的分层体系结构,能够有效地解决调度系统中庞大资源的协调管理问题。

参考文献:

[1]　包振强. 基于多 Agent 的智能制造执行系统研究[D]. 南京: 南京航空航天大学, 2003.

[2]　KHALID K, HENRI P, MEBARKI N. Using multi-agent architecture in FMS for dynamic scheduling[J]. Journal of Intelligent Manufacturing, 1997, 16(8): 41 - 47.

[3]　王艳红, 尹朝万, 张宇. 基于多代理和规则调度的敏捷调度系统研究[J]. 计算机集成制造系统 CIMS, 2000, 6(4): 45 - 49.

[4]　韦文斌, 杨建军, 曾波, 等. 基于多代理的车间调度系统结构和调度流程的研究[J]. 航空精密制造技术, 2001, 37(5): 23 - 27.

[5]　张锋铭. 虚拟企业工作流研究及基于多代理和 Web 服务的原型系统实现[D]. 西安: 西北工业大学, 2002.