计算机科学 1996 Vol. 23 №. 6

一个基于 PVM 的异构型智能分布式任务调度系统

周笑波　汲　化　谢　立
（南京大学计算机科学与技术系　南京 210093）

摘　要　The authors propose a heterogeneous intelligent distributed task scheduleing model based on belief network, using the approach of DAI and knowledge processing. On the top of PVM(Parallel Virtual Machine), consisted of Sun Workstation and Wyse multiprocessor(4 CPUs), a prototype System IDTS has been implemented. The results show a better performance.

关键词　Belief network, PVM.

1 引　言

分布计算环境是基于分布式系统上进行的计算服务系统，分布式任务调度问题就是寻找将一组相互协作的任务分配到一组处理器上运行的最优解。在任意多个处理器组成的系统中，求最佳的分布式任务调度一般是 NP 完全问题，因此传统的研究一般都局限于寻找满足特定功能目标的次优算法。由于传统调度系统的单一策略模式，使得调度决策在某些条件下是有效的，而在更多的情况下却不能令人满意。

在一个大型的分布式系统中，一方面，由于系统状态的不确定性；状态知识的不完整性；调度策略的不稳定性和系统缺乏自我调节能力，现有的许多分布式任务调度算法将失去其有效性而变得低效或无能。

另一方面，随着工作站、小型机等计算平台的普及和网络技术的成熟，组成系统的这些结点已不完全相同，而具有多种形式的异构性，例如，系统配置异构、体系结构异构和操作系统异构。如何将任务在这些异构的结点间快速而有效地分派和转移是目前分布式计算研究所面临的一个重大而复杂的问题。系统的异构性极大地增加了调度工作的复杂性，而现有的许多 分布式任务调度都缺乏对异构型分布式计算环境的有力支持。

本文在基于智能分布式任务调度[1]的基础上，引进信度推理网络技术、算法知识库技术和学习技术，设计和实现了一个基于 PVM 的异构智能分布式任务调度，它是在由并行计算机 Wyse series 7000i (4CPUs)和 SUN SPARC Ⅱ 组成的异构环境中的一个较为实用和有效的分布式任务调度系统。

2 系统模型概述

智能分布式任务调度模型如图 1 所示。该系统模型主要分为状态管理器和决策管理器。每一个处理结点上都有该模型的一个实现，称为局部调度系统。分布式任务中处理器的管理采用分散控制方式，由局部调度系统通过合作来完成。

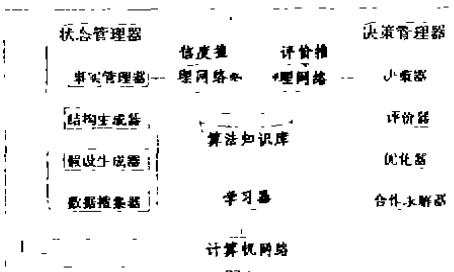

图 1

2.1 状态管理器

状态管理器负责搜集、推理和形成全局状态知识，并维护状态知识库，是局部调度系统对整个分布式系统所具有的认识。它用于搜集一切在一定开销范围内可能搜集得到的有用的状态信息，然后利用局部系统所掌握的不完整的全局知识，辅助以少量的必不可少的系统状态查询工作，通过信度推理网

周笑波　硕士生，主要研究领域为分布式系统。汲化　博士生，主要研究领域为分布式系统。谢立　副校长、博士生导师，主要研究领域为分布与并行系统。

络尽可能快地(保证时效性)推导出尽可能接近实际的系统状态,存放在状态库中,作为整个分布式任务调度的决策基础。状态管理器有四个基本模块:I/O解释器,事实管理器,学习器,结构生成器。

状态知识可以分为三个层次:①基本数据集:是由低级模块通过观察而收集到的状态数据的集合;②基本事实集:是由假设生成器对基本数据集的解释,又称基本命题集;③高级事实集:是从基本事实集推出的状态知识,又称高级命题集。

(1)I/O 解释器。是一个双向解释系统,局部调度系统通过它与低层操作系统交互。其中的数据搜集器负责搜集各种状态数据,网络信息。包括:①测试数据;②观察数据;③统计数据;④时间数据。数据搜集器将搜集到的各种数据存放在一个基本数据库中,用信度来表示数据的可靠程度。

假设生成器完成从基本数据集到基本事实集的推理。

(2)事实管理器和信度推理网络。事实管理器负责维护知识库中由事实命题构成的信度网络,完成从基本数据集到基本事实集的推理。信度推理网络是一个具有层次结构的非强连通图,其中每个结点都有一个信度值用以表示该结点所代表的命题的真实程度。每个非叶结点的信度值由它与各子结点之间的有向边的强度组合得到。当新的证据到来时,首先改变叶结点的信度值,并通过有向边在整个信度网络中逐层向上传播。

(3)结构生成器。主要目的是根据任务组的通信结构和当前状态知识决定的硬件结构状态,决定一个合适的硬件结构,以减少任务组的通信代价。为减少开销,我们不考虑正在运行的任务的动态迁移。

(4)学习器。学习是智能分布式任务调度的一个重要功能,是从未知或不确切知到到知的过程。A)学习专家经验;B)询问学习;C)概念学习。

2.2 决策管理器

决策管理器负责当前的状态知识和已掌握的调度知识,通过评价、优化,产生出一个既能满足用户响应时间的要求,又能满足系统动态变化的性能要求的调度方案。同时,决策管理器还具有在各局部调度系统之间协调、合作的能力,以解决知识不完整性问题。决策管理器有以下五个模块:

(1)算法库。是决策器进行决策的对象,提供可供选用的各种算法。在算法库中,各个可供选用的算法呈并列结构,每个算法包含两部分:算法的适用状态和算法体,这两部分之间建立一一对应关系,算法库和学习器相连。

(2)决策器。根据当前的系统状态知识,从算法库中挑选适用于当前状态的算法,调度算法的选择依赖于三个方面:①算法的调用条件;②算法的预期目标;③算法的运行代价。我们在决策器中也引入信度网络机制。

(3)评价器。负责对由决策器产生的一组侯选调度方案进行评价,评价标准是一组代价函数,其目标是要极小化这些函数。可接受调度可能是不唯一的,这可以通过一个加权函数来决定最后的选择,加权函数一般表示调度系统对某些评价标准的侧重。

(4)优化器。完成对调度的优化工作,优化的目标是寻找一个可接受的调度。我们提出一个称为缆车爬山法的启发式算法[4]。

(5)合作求解器。由于分布式系统的自治性,各局部调度系统所拥有的知识是局部的、不完整的,从而各个局部调度系统的调度合成在全局可能不是最优的,合作求解器用来解决此知识不完整性带来的冲突问题。

3 系统实现

3.1 实现环境

根据上述模型,我们在由三台采用 SUN OS 4.1 的 SUN SPARC Ⅱ工作站和一台采用 SCO Unix For MPX3.0 操作系统的 Wyse series 7000i(CPUs)多处理机通过以太网组成的环境里,实现了异构型的智能分布式任务调度系统 IDTS,如图 2 所示。

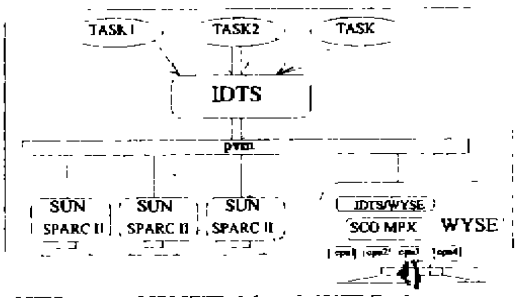

图2

(1)并行虚拟机 PVM。由于本调度系统的环境是跨体系结构,操作系统也不完全兼容的异构环境,解决其中统一的通信体制问题成为关键的问题,我们采用了 PVM 作为该异构型调度系统的通信平台。

PVM 是由美国 Oak Ridge 国家实验室,Tennessee 等大学联合研究开发的一个分布并行计算平台。它建立在以 Unix 为操作系统的、基于 TCP/IP

协议的、异构的网络环境中，采用 Message-passing 的通信模型，提供给用户一个一致的 API 接口，使得用户不必过多考虑和关注网络中不同厂商提供的硬件平台和资源的分布情况，可以动态而方便地将不同体系结构的网络结点加入 PVM 环境中，而在用户的角度，其面对的只是一个一致的、单一的并行虚拟机，从而可以充分利用网络中各种计算资源。

目前 PVM 最高版本所支持的硬件体系结构达 40 余种，如 SUN、SGI、CRAY、MIPS、ALPHA 等。

(2)SCO MPX Version 3.0。多处理器计算机使用多个 CPU 同时执行多个任务来提高系统性能，但 SCO UNIX 系统是一个多任务、多用户、单处理器的操作系统。SCO MPX Version 3.0 是一个 SCO UNIX System V 的多处理机扩展，在标准的 UNIX 核心上添加 SCO MPX 软件后，UNIX 系统便可以自动识别和使用多个 CPU。

SCO MPX 是模块化的，它对 UNIX 系统完全兼容，丝毫不影响原系统已存在的功能、系统管理和文件系统。它仅在 UNIX 核心上作了支持多处理器的修改工作，系统界面并无改变，对用户来说也是透明的。

3.2 实现技术

(1)统一的通信平台。为了实现统一的通信平台，我们利用 PVM 源代码公开的优势，首先将 PVM 移植到基于 SCO Unix for MPX 操作系统的 Wyse 并行机上，并以 PVM 为平台，编写了智能操作系统 KZ3/WYSE 的通信软件，为异构型的智能分布式调度系统 IDTS 提供了统一的通信平台。

该平台提供最主要的两个通信原语：pvm_sys_send(hostname,msgtype,message,length)和 pvm_sys_receive(hostname,msgtype,message,length)。这样，本异构系统中的工作站和 Wyse 多处理机间便可用这两个通信原语，进行机间进程级的通信。其中，msgtype 用来区分信件的类型，目前共有 11 种类型。

另外，pvm_sys_query(hostname,msgtype)原语用来对指定的处理器的信箱查询有无指定的 msgtype 的信件。

在机内不同进程的通信手段上，除了用上述两个通信原语利用不同的信件类型进行通信外，本系统还用 PRC 机制，实现并提供了三个原语：sys_put(name,data,length)将 data 数据区内容放在 name 标识的信箱里。sys_get(name,data,length)将由 name 标识的信箱里的内容放入 data 数据区内，sys_release(name)释放信箱内由 name 标识的内容。

(2)智能分布式任务调度器 IDTS 的知识获取。分布式任务调度的基础在于准确及时地获取各种有用的知识，IDTS 的学习器学习两类知识：专家知识和状态知识。专家知识是一种永久性知识，它包括现有的任务调度算法和设计分布式任务调度的经验，我们是用规则的形式来描述这些知识的。

学习状态知识的方法有两种，一种是通过观察的学习方法，另一种是询问学习方法。通过观察学习是一种高级形式的学习方法，它以观察和询问为基础，通过归纳的方法生成规则。IDTS 主要对通信产生的结果进行观察和统计，例如，在经过 100 次观察和询问学习之后，学习器发现当对方结点处理器繁忙时，50%RPC 失败，则得到规则：RPCTO→处理器忙{0.5}。询问学习是一种简单学习方法，它是获得准确的状态知识的唯一途径。

UNIX 作为较适宜分布计算的平台，它把一个进程的运行地址空间分为核心空间和用户空间，应用程序只能运行在用户的地址空间，要想读取核心空间的信息和服务，就必须通过系统调用。但系统调用只限于提供比较通用的功能，它并没有向我们关心的每个结点的几个状态：CPU 就绪队列长度、剩余内存、磁盘剩余量和磁盘 I/O 量提供直接的负载信息。

在 IDTS 中，我们通过使用系统调用之外，通过对代表 UNIX 系统内核的内存映象的一对伪设备驱动程序/dev/mem 和/dev/kmem 进行了读写，即对内存映象的访问，来准确地知道我们所需要的各种核心状态。

当然，这种获取内核信息的方法必须以内核变量的符号表为前提，否则就因无法定位内核变量的存贮地址，而无法读取它们的数值。

(3)IDTS/WYSE。在单处理机的 UNIX 系统中，调度者关心的是什么时候，一个进程被调度。而在多处理机的 UNIX 中，调度者除此之外，还需关心进程在哪个处理器上执行。

在采用 SCO UNIX For MPX 操作系统的 WYSE 多处理机上，我们根据上述智能调度的思想，在 MPX 之上设计了一个智能局部调度系统 IDTS/WYSE。我们使用系统调用和通过对代表 UNIX 系统内核的内存映象的伪设备驱动程序/dev/mem 和 dev/kmem 进行读写，得知我们所需要的各种核心状态后，再用 MPX 提供的 lockpid()原语，对分配到 WYSE 多处理机上的任务组进行智能调度，将任务

分配到资源较为空闲的处理器上执行，以更高效地使用多处理机的并行工作能力。IDTS/WYSE 既可以作为 IDTS 的组成部分，也可单独成为一个调度系统。这样，本系统在工作站和多处理机间支持粗、中粒度的进行，而且在多处理机内能实现细粒度的并行，形成一个并行计算环境。

4 系统使用及性能分析

4.1 系统使用

本调度系统的接口方式采用 UNIX 通用文件接口方式，在系统装入之后，用户首先必须建立与处理器知识有关的数据文件 ioshosts，格式如下：

host-name　　host-no　　host-type

host-name 必须与 UNIX 系统中/etc/hosts 中的工作站名一致，host-no 是一个区别各工作站和 WYSE 多处理机的顺序号，host-type 表示该机器类型，如 SUN4 和 WYSE 等。

当用户的任务组希望通过本系统进行调度运行的时候，用户必须首先填写任务说明书，它以文件的形式提交给本系统，其格式如下：

主关键字　说明项　〖次关键字【说明项】…〗

其中，主关键字分不可缺省和可缺省两种，共有八类：任务组名，任务组所在路径名，任务说明表，运行说明表，通信说明表，完成时间说明，局部任务说明，指定处理器名表。用户提交任务说明书以后，就可以调用调度系统的命令运行该任务说明书说明的任务组。

4.2 性能分析

IDTS 的状态获取是从信度推理网络推导出的状态知识库中得到的，而状态数据库将根据调度系统当前知识的变化而不断刷新。它在产生调度前首先进行一次处理器选择。其调度产生是根据算法知识库中的算法知识，通过决策器中的评价推理网络，进行决策、评价、优化的结果，目前我们的算法库中有两个算法：LPT 算法(Graham69，LO85)和随机算法。[4]

我们主要通过输入样本空间并行性分析、满足响应时间与增加系统流量的关系、最大到达率与处理器数的关系和系统总完成时间与处理器数的关系等四个方面的实验进行了系统性能分析，同时，与传统的 BID 投标算法[4]的性能作了比较，实验结果表明，在一个大型分布式系统中，IDTS 在满足响应时间要求和增加系统流量方面均优于 BID 算法，并且在任务组所包含的任务个数增加时，IDTS 的性能优于纯粹通过通信方式获得状态的方法。

另外，我们对由三台 SUN 工作站组成的同构环境和再加入一台 WYSE 多处理机后的异构环境下的 IDTS 的性能变化进行了分析比较。两者的输入样本空间是一样的，均由 100 个任务组(每个任务组包括 20 个任务)构成，每个任务组的调度属性包括运行代价、通信代价、任务截取时间和任务到达时间。

由于充分利用了 WYSE 多处理机 4 个 CPU 的并行工作能力，本异构系统(实际上有 7 个处理器，3 个 Sparc I 和 4 个 Intel 486)的系统性能大大增强。100%的任务组都可以达到加速比 2，85%的任务组可以达到加速比 2.5，45%的任务组可以达到加速比 3，有 20%的任务组可以达到加速比 3.5，而无 WYSE 机加入，由三台工作站组成的同构系统仅有 35%的任务组可以达到加速比 1.5，几乎没有任务组可达到加速比 2。

结　论　本文提出了一个异构型的智能分布式任务调度模型，并在由 WYSE 并行计算机和 SUN 工作站组成的异构环境中，设计和实现了一个基于 PVM 的较为实用有效的智能分布式任务调度系统。它具有如下特点：1)具有处理知识不确定性的能力。2)具有自适应和学习的能力。3)具有在异构型结点间合作协调的能力。

参考文献

[1]Casavant, T. L., et al., A Taxonomy of Scheduling in General-Purpose Distributed Computing Systems, IEEE Trans. Soft. Eng., SE-14(2), 1988

[2]Blair G. S., A Knowledge-based Operating System, The Computer Journal, 30(3)1987

[3]Chen J., Xie, L. and Sun Z. X., A Model for Intelligent Task Scheduling in a Large Distributed System, ACM SIGOPS Operating System Review, Oct. 1990

[4]陈军，博士论文，1990

[5]Shafer G., A Mathematical Theory of Evidence, Princeton University Press, 1976

[6]Chou T. C. K. et al., Load Balancing in Distributed Systems, IEEE Trans. Soft. Eng., SE-8(4)1982

[7]陈军、谢立等，基于知识处理的分布式任务调度，计算机研究与发展，1990

[8]郭锐峰，UNIX 系统内核信息的获取方法，小型微型计算机系统，1/1995