

PMS 中的一种匹配通信机制

A Matching Communication Mechanism in PMS

刘欣然 胡铭曾

王文敏

(哈尔滨工业大学320信箱 哈尔滨150001) (哈尔滨理工大学计算机工程系)

摘要 PMS(Parallel Member System) is a parallel reasoning system based on production system. This paper presents a matching communication mechanism suited for PMS, and describes the capacity of its communication primitives.

关键词 Production system, Parallel member system, Matching communication

过去在产生式系统的并行实现方面,人们往往将注意力放在共享存储的多处理器系统和专用体系结构的机器上,而对基于消息传递的计算机系统却没有给予太多的关注。其中主要原因是第一代基于消息传递的计算机系统具有较大的通信延迟,使得在产生式系统的并行实现中由于通信开销过大而不能获得令人满意的加速比。现在由于技术的进步,网络通信延迟降低了2-3个数量级,因此在基于消息传递的计算机系统上实现并行产生式系统又重新受到了人们的关注^[1]。

并行成员系统 PMS 就是这样一个在群机环境下实现的并行推理系统,它是在产生式系统 OPS5 基础之上发展而来的,它的基本执行单位是成员(Member),相当于一个小的产生式系统。执行时,多个成员按一定的策略分配到各个处理机上进行并行推理。本文对 PMS 的通信机制进行介绍,并描述了其通信原语的表现能力。

一、背景材料

产生式系统是以 Post 机为理论基础的一种计算模型,它与以 Turing 机为理论基础的计算模型的根本区别在于,产生式系统的特征不是控制驱动方式,而是一种数据驱动方式。由卡内基·梅隆大学开发的 OPS5^[2]是一种颇受欢迎的产生式系统程序设计语言,主要用于人工智能、专家系统和认知心理学领域。

一个产生式系统是由存放在产生式存储器(PM)中规则的集合和被称为工作存储器(WM)的公共数据库组成的。一个产生式规则具有如下形式:

if $C_1 \& C_2 \& \dots \& C_m$ then $A_1; A_2; \dots; A_n$

其中条件部分称为左手部(LHS),结合部分称

为右手部(RHS), $C_i (i=1, \dots, m)$ 称为条件元素, $A_j (j=1, \dots, n)$ 称为动作。它的意义是:当 WM 中的数据能使条件 $C_1, C_2, \dots, C_m$ 同时为真时,则执行动作序列 $A_1, A_2, \dots, A_n$ 。

产生式系统的执行方式是在推理机中循环进行“匹配-冲突归结-执行”的周期操作,如图1所示。

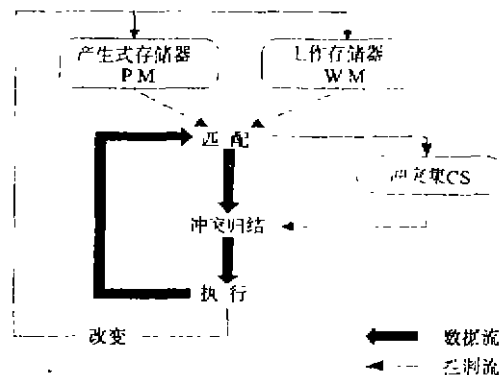


图1 产生式系统的运行模式
(虚框内所示为推理机)

由于产生式系统的执行与人类的思维过程非常相似,所以已用于书写了许多成功的专家系统。但由于产生式系统缺少显式模块化和结构化的知识描述方式,而且它是以产生式规则而不是以任务为单位,当规则数目庞大时(如著名的专家系统 R1 已达4000多条规则)易产生不可维护性而失败,以至没有人知道系统在做什么。

成员系统^[3](MS)就是为解决这一问题基于产生式系统而设计的推理模型,它的基本执行单位是成员而不再是产生式。这里成员是一个结构和功能清晰的、相对独立的知识处理单元,相当于一个小的产生式系统,拥有自己的产生式存储器和工作存储

器,具有一定的推理能力,多个成员还可以通过通信彼此之间交换信息以进行协作求解。

将成员系统模型映射到多机系统上就构成了并行推理系统 PMS,它由分布在多个处理器节点上的、彼此平等的、松散耦合着的多个知识处理单元(成员)构成,本文讨论了 PMS 的通信部分设计。

二、PMS 通信方式的确定

迄今为止所提出的通信方式,如消息传递,远程过程调用等,大都出现在以控制驱动为特征的过程性语言中;而 PMS 是在产生式基础上发展而来的,所以 PMS 的通信也应当具备产生式系统的典型特征——数据驱动的特点。

吸收产生式系统工作存储器 WM 中的内容与产生式存储器 PM 中产生式左手部的条件元素相匹配的思想,PMS 中也将通信原语分为通信动作原语和通信模式原语两类。通信动作原语是在产生式规则右手部的用于发送消息的动作,通信模式原语则是在产生式左手部的用于接收通信消息的条件元素,并不是通常意义的通信原语,PMS 中称规则库为广义产生式存储器,因为每条产生式的左手部除含有普通条件元素外还可能含有通信模式元素,称公共数据库为广义的工作存储器,由普通的工作存储器和通信消息存储器构成,见图2。

两个成员通信时,源成员的某条产生式通过右手部的通信动作原语将通信消息发送给目标成员,当目标成员接收到此消息后,将其存放到自己的通信消息存储器中,当目标成员进行下一轮的“匹配—冲突归结—执行”时,就会将广义工作存储器(包括工作存储器和通信消息存储器)中的内容与广义产生式存储器中的产生式(左手部包含通信模式原语)进行重新匹配,从而对目标成员中产生式的执行顺序产生影响,这正好体现了多个知识源间(成员)相互协同进行问题求解的思想。

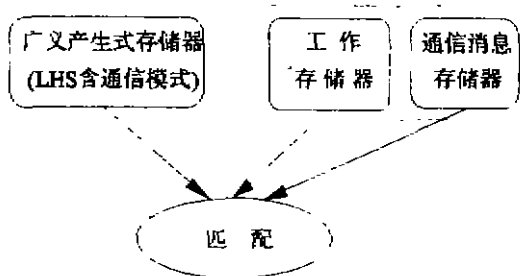


图2 PMS 中的产生式匹配示意图
(图中虚框所示为广义工作存储器)

观察日常生活中人类的活动也不难发现,人与人之间的通信方式不外乎有两类:一类是非交互式的,如信件、电报等;另一种是交互式的,如电话及正常交谈。因此为了便于并行人工智能问题求解,更加准确地模拟人类活动,在 PMS 中也提供了两种通信方式:一种是交互式的称作同步通信,另一种是非交互式的称作异步通信。

三、PMS 通信原语的设计

PMS 是通过通信动作原语和通信模式原语的匹配来实现通信的。异步通信原语只有一对,它们是通信动作原语 Send 和通信模式原语 Receive;同步通信原语有两对,它们是同步请求动作原语 Request 和同步应答动作原语 Reply,与它们相匹配的通信模式原语是 Req-accept 和 Rep-accept。

在 PMS 中的通信具有一对多(一个通信动作原语可向多个成员发送消息)的特点,因此,在通信原语之间存在着一个通信协议,如图3所示。经过通信协议作用之后,形成若干条消息项并被送给相应的成员。

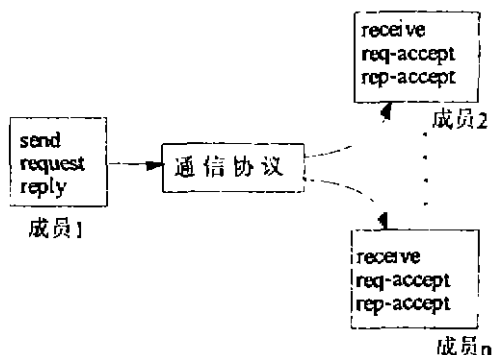


图3 PMS 中通信协议的作用

在 PMS 中,同步通信的过程是:发送方通过同步请求动作原语 Request 发送同步请求消息后,转入活动等待状态,直到接收到目标成员通过同步应答动作原语 Reply 发送过来的应答消息后才继续执行;异步通信则是发送异步消息后继续运行,而不等待对方的应答消息。

目标成员接收到源成员通过通信动作原语发送的消息后,将其存放到自己的通信消息存储器中。进入匹配阶段后,如果某条产生式左手部通信模式原语的形式与通信消息存储器中某条消息的形式相一致的话,则源成员的通信动作原语与目标成员的通信模式原语匹配成功。

通信动作原语与通信模式原语之间的一一对应

反映了 PMS 通信原语的数据驱动特征和“通信模式匹配+通信动作执行”的风格。如果单从同步通信和异步通信的角度来看, PMS 与传统的并行过程性语言是相同的,但在驱动方式和功能上则有较大的甚至是本质上的差异,这主要表现在:1)传统的并行过程性语言反映的是控制点与控制点之间的同步,而 PMS 反映的则是数据流与数据流之间的同步;2)传统的并行过程性语言是单消息激活一个进程,而 PMS 可以是多消息激活一个成员(只要目标成员中某条产生式的左手部包含有多个通信模式原语,即可接收来自多个成员的通信消息)。

四、PMS 通信原语的实现

在并行成员系统中,由于所有的成员分布在多个处理机上,因此成员与成员之间的通信相应地可以分成两类:节点内两成员间的通信和节点间两成员间的通信。虽然这两者都包含有同步通信和异步通信,但它们之间存在着本质上的区别。

当两个成员处在同一节点机时,两成员间所谓的通信实际上是源成员将通信消息放入目标成员的通信消息存储器中,然后再与目标成员 PM 中产生式左手部的通信模式原语进行匹配,继而决定该成员在可执行队列中的位置以及此成员内产生式的执行顺序。整个过程中发送动作是模拟进行的,并且不存在真正的接收动作。

当两成员处在不同的节点机时,情况就完全不一样了。这时两成员间的通信必须由源成员通过真正的发送动作将消息发送到目标成员所在的节点,节点接收后根据消息的内容将其送入相应的目标成员通信消息存储器中,再与其广义产生式存储器进行匹配,从而决定该成员在可执行队列中的位置以及此成员内产生式的执行顺序。

4.1 通信的软硬件基础

为了通信可靠性的需要和便于系统的普及,我们选用了当前最为流行的局域以太网网络作为 PMS 的运行环境。通信软件包括两类函数:

一、三个底层通信函数

(1) **Rec.info**: 中断接收其他节点发送过来的通信消息,并将其放入事先定义好的通信消息存储器中;

(2) **Send.info**: 将消息发送到系统中的某个节点上;

(3) **Fetch.info**: 到本节点的通信消息存储器中取一条其他节点发送过来的通信消息。

以上三个函数是 PMS 通信实现的关键之一,节点间通信时,所有高层通信函数都是在这三个底层通信函数的基础上建立的。

二、格式转换函数

(1) **Constr**: 实现表结构到串结构的转换;

(2) **Strcon**: 实现串结构到表结构的转换。

因为节点间传送的消息是字符串,而 PMS 的解释器是用 LISP 语言写成的,其基本数据结构是二叉树的表结构,不能直接发送,所以发送之前首先要将消息由表结构转化成串结构,然后调用发送动作原语将其发送到相应的节点;当接收方接收到此消息时,还需将此消息重新由串结构转换成表结构后送给 PMS 解释器。这样才能保证消息的正常发送和接收。

这两个函数是 PMS 通信实现的关键之二,节点间通信时,所有高层通信函数都调用了这两个函数。

4.2 通信的同步处理

异步通信原语 **Send** 不存在同步问题,所以节点内与节点间的通信流程基本一样,只是在节点间通信时比节点内通信多了目标节点的查询和节点间发送动作的调用而已。但对于同步请求动作原语 **Request** 和同步应答动作原语 **Reply** 这对同步通信原语来说就存在着一个很关键的同步问题。

在节点内通信时,其同步过程是:同步请求动作原语 **Request** 发送完消息后将自己挂起,然后把目标成员插入到可执行队列的头部进行匹配处理,以决定下一步动作。若匹配成功,则目标成员将应答消息通过同步应答动作原语 **Reply** 送回发此请求的成员;如果匹配不成功,则退出可执行队列以便让源成员接着执行。

在节点间通信时,同步过程是:源成员通过同步请求动作原语 **Request** 将消息发送到目标节点后开始查询目标成员的应答消息,如果目标成员通过同步应答动作原语 **Reply** 发回匹配成功的应答消息,则源成员将其接收并加以相应的处理;如果目标成员发回匹配不成功的应答消息,则源成员退出查询状态接着执行。

从以上的分析可以看出,节点内通信时不存在查询应答消息的动作,所谓等待应答消息的过程实质上是让目标成员先执行而已。另一个重要的区别是:节点内通信如遇匹配不成功,目标成员只需简单地将控制权交还给源成员即可;而节点间通信如遇匹配不成功,目标成员必需通过同步应答动作原语 **Reply** 发回一特定消息来标明,这是节点内和节点

间通信在同步问题上最本质的区别。

上述过程是由 PMS 解释器自动完成的,对使用者透明,用户编写程序时无须知道通信的详细过程,只要按其语法格式书写即可。

五、通信原语的语法和语义

5.1 同步通信原语

1)同步请求动作原语

语法形式: (Request <target-member> <message>)

语义描述:将同步请求消息 <message> 发送给目标成员 <target-member>)

2)同步请求模式原语

语法形式: (Req-accept <source-member> <message-form>)

语义描述:如果来自源成员 <source-member> 的同步请求消息满足 <message-form> 的模式,则匹配成功,并进行相应的值约束

3)同步应答动作原语

语法形式: (Reply <target-member> <message>)

语义描述:将同步应答消息 <message> 发送给目标成员 <target-member>

4)同步应答模式原语

语法形式: (Rep-accept <source-member> <message-form>)

语义描述:如果来自源成员 <source-member> 的同步应答消息满足 <message-form> 的模式,则匹配成功,并进行相应的值约束

5.2 异步通信原语

1)异步通信动作原语

语法形式: (Send <target-member> <message>)

语义描述:将异步通信消息 <message> 发送给目标成员 <target-member>

2)异步通信模式原语

语法形式: (Receive <source-member> <message-form>)

语义描述:如果来自源成员 <Source-member> 的异步通信消息满足 <message-form> 的模式,则匹配

成功,并进行相应的值约束

六、通信原语的表现能力

例1: (Send m1 message) 将异步消息 message 发送给成员 m1,然后继续执行。

例2: (Request m1 message) 向成员 m1 发出同步请求消息 message,然后进入活动等待状态。

例3: (... (Receive m1 message-form1)
(Receive m2 message-form2)

→

)右手部动作

如果接收到成员 m1 的满足 message-form1 模式的消息,并且接收到成员 m2 满足 message-form2 模式的消息,则执行右手部动作。

例4: (... (Receive m1 message-form1)

(Req-accept m2 message-form2)

→

(Reply m2 message)

)

如果接收到成员 m1 的满足 message-form1 模式的消息,并且接收到成员 m2 满足 message-form2 模式的消息,则向成员 m2 回送一同步应答消息 message。

结束语 并行成员系统 PMS 是对产生式系统进行重大改造而设计的用于支持并行 AI 问题求解的推理系统,它在保留了产生式系统优点的同时又克服了其缺陷。由于具有比较丰富的通信原语,从而给程序员提供了灵活控制进程间相互作用的手段。

参考文献

- [1] Anurag Acharya et al., Implementation of Production Systems on Message-Passing Computers, IEEE Trans. on Parallel and Distributed Systems, 3(4) 1992
- [2] C. L. Forgy, OPS5 User's Manual, Technical Report CMU-CS-81-135, Department of Computer Science, Carnegie-Mellon University, July 1981
- [3] 王文敏,支持协同式 AI 问题求解的成员系统(博士学位论文),哈尔滨工业大学,1989,12