

多 Agent 系统通信设施的设计与实现^{*}

游明英¹ 李益才² 彭 军¹

(重庆科技学院电子信息工程学院 重庆 400050)¹ (重庆交通学院计算机与信息学院 重庆 400074)²

摘要 理性的 Agent 是智能的 Agent, 在多 Agent 系统中, Agent 之间的通信不单是信息的传递问题, 还有 Agent 对所传递信息的共同理解。本文在分析 Agent 通信模型的基础上, 详细讨论了在多 Agent 系统平台的开发过程中如何设计及实现信息传输子系统, 提出了一种共享数据结构与私有数据结构相结合, 设置信息路由的方法来实现多 Agent 系统中 Agent 之间的通信。该方法在实践中取得了较好效果。

关键词 多 Agent 系统, 通信, 缓冲区, 信息对象

The Design and Realization of Communication Facilities in Multi-Agent System Development

YOU Ming-Ying¹ LI Yi-Cai² PENG Jun¹

(College of Electronic Information Engineering, Chongqing University of Science and Technology, Chongqing 400050)¹

(School of Computer and Information, Chongqing Jiaotong University, Chongqing 400074)²

Abstract Rational agent is intelligent agent. In multi-agent system, there not only needs that the information can be transported to the correct target agent but also the agent can understand the information it received correctly. Based on the communication model, this paper discusses in details how to design and realize the information transport subsystem in multi-agent system, and proposes a method to implement the communication in multi-agent system.

Keywords Multi-agent system, Communication, Buffer, Information object

1 引言

超媒体、多媒体技术和人工智能的发展使基于 Web 的网络应用系统的发展进入到一个新的阶段, 利用多媒体、超媒体技术以及网络技术, 人们开发出许多网络应用系统。这些网络应用系统建立在超媒体及多媒体技术之上, 让用户可以根据自己的需要和时间情况来共享网上丰富的资源及享受网络应用系统提供的服务。由于这类系统的高度动态性、开放性以及复杂性, 在基于 Web 的适应性系统中使用 Agent 技术来提高系统的适应性是很自然的想法^[1]。

智能主体 (Agent) 一般用来特指那些具有一定智能和自治能力, 能根据环境变化做出响应和进行相应处理的软件系统。由于基于智能体的方法具有自治性、适应性、稳健性, 而且易于实现, 因此对那些结构性不好或者定义不明确的任务有很大的优势。

在 Agent 的研究中, 认知模型、主体结构和协同机制是 Agent 的三个主要研究方向。协同机制是指在多 Agent 系统中, 智能主体之间进行协商共同完成任务的方法, 主要有 Resenschein 对策论协商机制^[2]、Koworesow 的行为与目标相关性自发协同^[3]以及 Jennings 的共同责任模型^[4]等。这些协同机制在较高层次上讨论了 Agent 与 Agent 进行交互的协商方案等的控制策略, 它们是 Agent 对所交换的信息或知识的共同理解。但对底层通信信息的传输与分配较少涉及。本文从多 Agent 系统通信协议出发, 对多 Agent 系统中 Agent 与 Agent 之间通信基础设施的设计与实现进行详细的讨论。

2 Agent 通讯模型

Agent 的最基本特征是其自主性, 即与周围环境主动相

互作用的能力。一个多 Agent 系统由许多相互独立的 Agent 组成, 它们之间的相互协调以及它们与外界环境的相互协调是多 Agent 系统能否成功的关键。Agent 之间的相互协调与通讯需要有通讯协议与 Agent 能够理解的通讯语言作保障。

2.1 Agent 通讯协议

在多 Agent 系统中, 各个 Agent 要做到相互协调和有效的互操作, 需要三个基本的条件, 首先是它们有共同的语言, 其次是它们能对所交换的知识有共同的理解, 最后是系统必须有对语言 and 知识进行交换的能力。因此在多 Agent 系统中, Agent 通信协议表现为一种层次模型, 如图 1 所示。

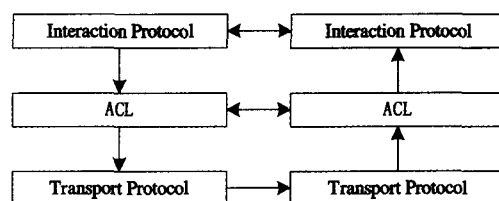


图 1 通信协议

图 1 中, 交互协议通常指在较高层次上 Agent 与 Agent 进行交互的协商方案等的控制策略, 它们是 Agent 对所交换的信息或知识的共同理解; 而 ACL (Agent 通信语言) 则是 Agent 进行通信的媒介, 通常表示由 Agent 通信语言来表达所交互的信息是命令、请求或者是告知; 而传输协议则是真正的 Agent 通信底层所使用的传输协议, 如 TCP/IP, SMTP 或 HTTP 协议等, 它的目的是保证交互信息能从一个 Agent 传输到另一个目标 Agent, 特别是当信息传输目的地是远程的 Agent 时。

2.2 Agent 通讯语言

^{*} 基金项目: 重庆市科委自然科学基金 (CSTC, 2005BB2050); 重庆市教委科学技术研究项目基金 (KJ051402)。游明英 讲师, 硕士, 研究方向为计算机应用; 李益才 副教授, 博士, 研究方向为网络应用系统。彭 军 副教授, 博士, 研究方向为网络安全。

由于多 Agent 中各个 Agent 是有一定智能和自主性的软件实体,各个 Agent 之间所传送的信息就不能仅仅表示为字符流或二进制数据流,它们之间的通信应该在知识层面上进行表达,以便于 Agent 之间的理解和交流,于是 Agent 通信语言(即 ACL)应运而生。ACL 是 Agent 之间进行信息交互和信息共享的协议性语言,它不但能够表示语义,而且能够表示语气。

2.3 本体

为便于 Agent 相互之间(也包括系统各部分)理解系统中所传递的信息,我们需要对系统的本体进行定义,本体(Ontology)是在一个系统中对共享的概念进行显示的、形式化的规范说明^[5],它是 Agent 能够理解领域术语,很好地完成交互协议(Interaction Protocol)所必需的。在具体的多 Agent 系统中,本体的定义与问题领域有关。

2.4 Agent 通讯模型

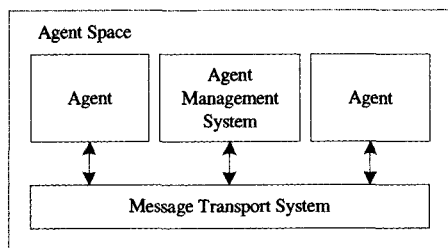


图 2 多 Agent 平台示意图

在多 Agent 系统中,Agent 之间需要相交通信和协作以完成共同的工作,Agent 通信模型是指对 Agent 之间的通信进行表达的形式化框架,它表达了 Agent 之间通信的机制。

在一个多 Agent 系统中,所有的 Agent 都在一个叫做 Agent Space 的空间中活动,Agent Space 运行在一个虚拟机上,它主要负责管理其中的 Agent,包括 Agent 的创建与删除、提供 Agent 之间进行通信的能力、Agent 在多 Agent 系统中的分布与移动等,如图 2 所示。

图 2 中,我们可以看出,在 Agent Space 中的所有 Agent 都接受 Agent Management System 的管理,所有 Agent 之间的通信都通过 Message Transport System(MTS)来完成。其中,Message Transport System 包括两部分,一部分通信媒介,它是使系统中各个 Agent 能够进行通信成为可能的各类载体的统一抽象描述,如黑板、元组空间等,它包括了通信能

够实现的机制;另一部分是通信规则,它主要用于定义与 Agent 之间的通信事件相对应的通信媒介的行为,它由通信语言和交互协议来进行定义,其中通信语言包含表达和交换数据结构的语法,在多 Agent 系统中,通常的作法是将要进行通信的信息看成一个对象,对象中有多个属性,包括语气(如命令、请求或者是告知),发送者,接收者,通信的内容等。

如果多 Agent 系统是在一个分布式环境中,则其中的 Message Transport System 要利用网络协议来完成分布在不同机器上的 Agent 之间的通信。

我们可以将 Agent 通信模型(ACM)定义为如下的一个二元组:

定义 1 Agent 通信模型 $ACM = (Agents, MTS)$

其中:Agents 是在一个 Agent Space 中活动的所有 Agent 的集合;

MTS 是为 Agent 之间通信的信息进行传输的信息传输系统,它包括通信媒介和通信规则,它负责依据一定的通信规则对 Agent 之间的通信信息进行传输。

3 多 Agent 系统通信信息传输子系统设计

由于 Agent 本身是一个软件实体,Agent 之间的通信可以自然而然地想到过程调用,通过对目标 Agent 的接收过程进行调用,将所要传递的信息作为参数传递过去,而过程的返回结果作为应答信息传递回来。根据目标 Agent 的物理位置来决定过程调用是本地过程调用还是远程过程调用(RPC)^[6]。但是,由于众所周知的原因,过程调用所传递的信息较简单,无法在交互协议及 Agent 通信语言的层次上实现 Agent 对所交换信息的共同理解,故在多 Agent 系统通信中较少使用这种方法。

另一种方法是采用公用数据结构来传递信息,即黑板模式。所有要发送信息的 Agent 向公用数据结构中写信息,而所有要接收信息的 Agent 则在信息到达黑板之后与之相联系的 Agent 接收自己的信息。这种方法能提供 Agent 之间在交互协议及通信语言层次上对所交换信息的共同理解,但由于这种通信方式并不考虑 Agent 本身在网络中的具体位置,所有需要传输的信息直接写到黑板,同时读数据也是从黑板直接读数据,加大了网络的负载,阻碍了它在网络中的应用。本文提出了一种共享数据结构与私有数据结构相结合,设置信息路由的方法来实现多 Agent 系统中 Agent 之间的通信。

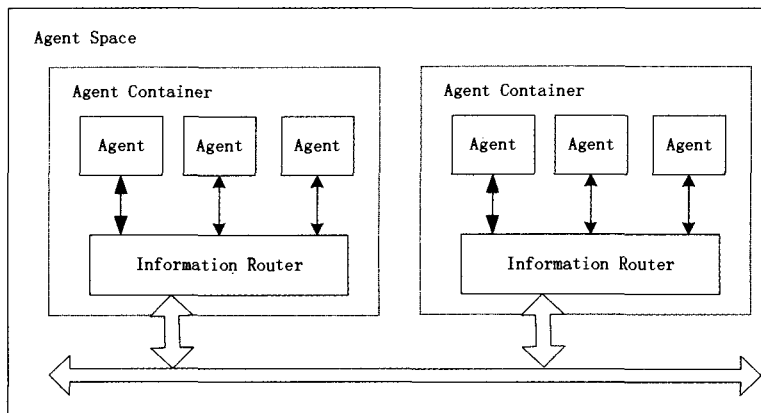


图 3 多 Agent 系统中信息传输子系统

3.1 Agent 通信方法

考虑到多 Agent 系统一般是跨主机的分布式系统,其信

息的传输不仅仅与本机有关系,因此,设计的多 Agent 系统中信息传输子系统如图 3 所示。

如图 3 所示, Agent Space 是整个多 Agent 系统中 Agent 的生存空间,它可以是跨平台的; Agent Container 是本地机 Agent 容器,本地机中可以有多个 Agent 容器,也可以只有一个 Agent 容器;在多 Agent 平台中,每一个 Agent 都有一个全局的名字,其名字的组成为“Agent 本地名/Agent 容器名@主机名:端口号”; Information Router 是信息路由,由它对 Agent 所发送信息的目标地址进行解析,并发送到相应的信息接收 Agent,同时信息路由还具有管理本 Agent Container 内 Agent 的功能。

3.2 Agent 信息传输机制

在一个 Agent Container 内由 Information Router 管理一个公用数据结构,它是一个 Agent 信息缓冲池,每个要发送信息的 Agent 只需成功申请一个信息缓冲区,并填上相应的内容即可。同时每个 Agent 都带有一个接收信息用的私有信息队列,该私有信息队列只允许其本身和 Information Router 访问。在一个 Agent Container 内 Information Router 和 Agent 之间的信息交互如图 4 所示。

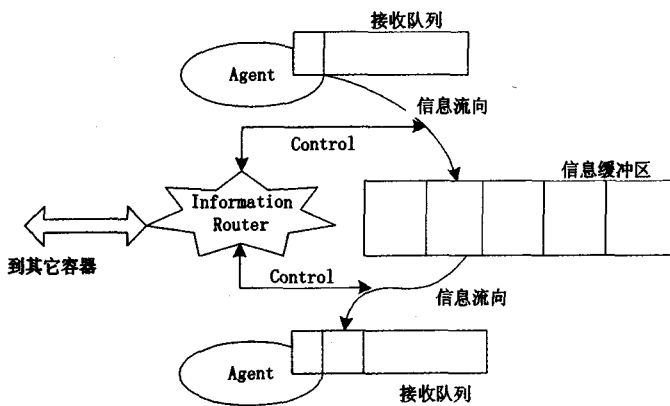


图 4 Information Router 和 Agent 之间的信息交互

从图中可以看出,如果一个 Agent 要发送信息,首先在 Information Router 的控制下向信息缓冲池申请一个缓冲区,成功之后填上相应的要发送的信息,Information Router 发现缓冲区中有待发信息,则将该待发信息的目标地址提取出来进行分析,如果目标地址是在同一个容器中,则直接将该信息转发到目标地址;如果是在同一主机但非同容器,则将其转发到相应容器的信息路由;如果不在同一主机,则找到目标地

址的主机地址并转发信息到目标机器的相应容器的信息路由。

3.3 Agent 信息对象设计

Agent 信息对象是 Agent 之间交流的语言,也是 Information Router 能将信息传递到目标 Agent 的依据,因此它的设计应遵循一定的规范。回顾前面曾谈到 Agent 之间的通信应能使 Agent 对所传递的信息有共同的理解,能对传递的信息进行理性化的处理,这包括语气、语用等方面。目前对于 Agent 通信语言有多种,如 KQML, FIPA 等,根据某一种规范的 Agent 通信语言,可以设计 Agent 信息对象,如表 1 所示。

表 1 Agent 信息对象

SpeechActType; <text>
Content; <statement>
Sender; <Agent global name>
Receiver; <Agent global name>
Ontology; <Ontology>

其中: SpeechActType 表示语气方面的信息,包括告知 (INFORM)、询问 (QUERY)、拒绝 (DENY) 等, Content 是具体的信息,它也根据领域特性有具体的格式, Sender 和 Receiver 指定了通信的发起者和接收者, Ontology 则指明了 Agent 应以何种方式来理解信息的具体内容,它指明了对 Content 的解析方法。

结论 理性的 Agent 是智能的 Agent, 在多 Agent 系统中, Agent 之间的通信不单是信息的传递问题, 还有 Agent 对所传递信息的共同理解。本文在分析 Agent 通信模型的基础上, 详细讨论了在多 Agent 系统平台的开发过程中如何设计及实现信息传输子系统, 提出了一种共享数据结构与私有数据结构相结合, 设置信息路由的方法来实现多 Agent 系统中 Agent 之间的通信。该方法在实践中取得了较好效果。

参考文献

- 1 Wooldridge M 著. 石纯一, 张伟, 徐晋晖译. 多 Agent 系统引论 [M]. 电子工业出版社, 2003, 10
- 2 Kraus S. Negotiation and cooperation in multi-agent environments. Artificial Intelligence, 1997, 94(1): 79~97
- 3 Shehory O, Kraus S, Yadgar O. Emergent cooperative goal-satisfaction in large scale automated agent systems. Artificial Intelligence, 1999, 110(1): 1~55
- 4 Kraus S, Sycara K, Evenchik A. Reaching agreements through argumentation: A logical model and implementation. Artificial Intelligence, 1998, 104(1): 1~69
- 5 Henze N, Nejd W. Knowledge Modeling for Open Adaptive Hypermedia
- 6 Brenner W, Zarnekow R, Wittig H. Intelligent Software Agents: Foundations and Applications. Springer-Verlag Berlin Heidelberg, 1998. 79~111

(上接第 218 页)

分、微分变量的 S 函数, 这样可以方便遗传算法的求解。设 PID 调节器给定最优化初始值为 $K_p = 0.63$, $T_I = 0.0504s$, $T_D = 1.9688s$ 。对于这组初始值, 可得系统的单位阶跃响应。设群体规模 $N = 10$, 杂交概率 $P_c = 0.25$, 变异概率 $P_m = 0.001$, 最大代数 $k = 10$, 因此可得 PID 调节器最优化参数值为 $K_p = 3.6232$, $T_I = 0.1175s$, $T_D = 16.9950s$ 。

从上述测试结果可以看出, 采用遗传算法进行非线性控制系统参数优化时, 系统的超调量较低、上升时间较短、调节时间较短。

小结 采用遗传算法对非线性控制系统参数进行优化时, 系统具有超调量较低、上升时间较短、调节时间较短的特

点, 且系统的稳定性和快速性都较好。

参考文献

- 1 Murphey R A, Pardalos P M, et al. Frequency Assignment Problems, Handbook of combinatorial optimization, Kluwer Academic Publishers [S], 1999
- 2 Vittorio M, Antonella C. An ANTS Heuristic for the Frequency Assignment Problem[S]. <http://www.csr.unibo.it>
- 3 Bater J, Jeavons P, Cohen D. Are there optimal reuse distance constraints for FAPs with random Tx placement? [S]: [CSD-TR-98-01]. CS Royal Holloway Uni. Of London, 1998
- 4 Aardal K I, et al. Algorithms for Frequency Assignment Problems [S]. CWI Quarterly, Vol9(1&2), 1996
- 5 郑志蕴. 利用神经网络进行无损检测的探讨[M]. 微计算机信息, 2002, 18(6): 27~29