

HLA 与 MAS 在分布式仿真领域的应用比较

张 毅, 胡勤友, 施朝健

(上海海事大学 航海科学研究所, 上海 200135)

摘 要: 高层体系结构(HLA)和多 Agent 系统(MAS)是目前分布式仿真领域的两个研究热点。两者适用性如何, 各方面有何异同之处, 在具体的仿真应用中如何选择这两个不同的系统构架来作为技术路线, 了解这两种不同技术的特点和它们之间的差异性为首要前提。该文通过介绍 HLA 和 MAS 在个体结构、通信和模型建立等各方面的优势和不足, 系统地比较了它们之间的性能差异。结果表明两者都是分布式仿真应用的优选方案, HLA 侧重资源重用, MAS 更具自主性, 需要根据具体仿真应用的要求来选择。

关键词: 高层体系结构; 多主体系统; 分布式仿真

中图分类号: TP391.9

文献标识码: A

文章编号: 1005-3751(2006)01-0150-04

A Comparison of MAS with HLA in Distributed Simulation Application

ZHANG Yi, HU Qin-you, SHI Chao-jian

(Research Institute of Nautical Science, Shanghai Maritime University, Shanghai 200135, China)

Abstract: HLA and MAS are two research focuses of the distributed simulation field at present. To learn their applicability, different and same aspects of them and how to select these two systematic frameworks as the technological route in application, learning their characteristic and otherness between them is prerequisite. This paper introduces the advantage and deficiency in their structure, communication and modeling, and then compares their different performance. The result has indicated these two technologies are optimum schemes that the distributed simulation is used. HLA emphasize resource reuse while MSA has more autonomy, choosing them according to different application.

Key words: high level architecture (HLA); multi-agent system (MAS); distributed simulation

0 引 言

当今社会, 随着计算机技术的飞速发展, 计算机仿真的应用领域不断拓宽, 需要解决的问题也越来越复杂, 必须依靠多个仿真系统进行联合协同仿真。因此, 分布交互仿真(Distributed Interactive Simulation, DIS)成为计算机仿真领域的热点之一。分布交互仿真技术的兴起主要为了解决两个问题: 一是使大规模复杂系统的仿真成为可能; 二是增强仿真系统的重用性, 降低开发费用。

1995年10月, 美国国防建模与仿真办公室(Defense M&S Office, DMSO)在提出的建模与仿真主计划^[1](M&S Master Plan, MSMP)中的一个主要内容即高层体系结构(High Level Architecture, HLA), 用于解决仿真系统的集成问题, 定义了构成分布交互仿真各部分的功能和相互关系, 为构造大规模的异构的仿真应用提供了一种集成方法。另一方面, 由于 Agent 所具有的自主性、协调性和智能性等特点, 国内外的研究人员也开始将从分布式人

工智能领域发展起来的多 Agent 系统(Multi-agent System, MAS)引入到分布交互仿真中, 期望加强仿真系统的规范性, 提高重用性, 增加互操作性。

由于 HLA 与 MAS 在仿真领域某些方面的相似和不同, 引起了研究人员广泛的关注和讨论。选用哪种系统结构能够更好地满足分布交互仿真的需要, 更具有优势性, 更有发展潜力, 这些问题是每个研究人员迫切需要的了解的, 也是在构建仿真应用时会面临的技术选择。文中将重点对 HLA 和 MAS 在结构、通信以及建模等方面的几个异同点进行阐述, 比较两者的优势和不足。

1 HLA 和 MAS 关于仿真应用的比较

由于 HLA 已经是成熟的标准框架, 因此关于它的研究更多集中于仿真应用方面。而 MAS 由于人工智能的渊源, 在理论研究和实际应用两条线上都发展迅速。但是由于两者产生根源不同, 发展背景和历程也不同, 使得它们各有侧重, 各有优势, 也有各自的不足。

1.1 联邦成员与 Agent

1.1.1 相似性

在基于 HLA 的仿真系统中, 联邦(Federation)和联邦成员是非常有特色的概念。联邦是指用于达到某一特定

收稿日期: 2005-04-21

作者简介: 张 毅(1979—), 男, 重庆江津人, 硕士研究生, 主要从事交通信息工程及控制技术、分布式仿真方面的研究; 施朝健, 博士, 教授, 博导, 研究方向为航海仿真、计算机图形技术和人工生命。

仿真目的的分布仿真系统,它由联邦对象模型、若干联邦成员和运行时间基础结构(RTI, Run-Time Infrastructure)构成(如图1所示)。在一个联邦执行中,各种仿真应用加入进来,每个仿真应用就可以是一个联邦成员。这些联邦成员为了实现某种特定的仿真目的,集合起来组成联邦,通过RTI进行交互作用,并且可以在联邦运行过程中随时“插入”和退出。

较之联邦,MAS中的Agent也是其中的特色单元。若干Agent组成MAS,Agent是其中基本的计算单元,一个Agent需要和其它的Agent不断进行交互,以对复杂的问题进行求解,合作完成一个整体的仿真目标。在大规模的仿真系统中,一个Agent也可以对应于一个仿真应用,成为这个仿真应用与整个MAS链接的桥梁,或者确切说是它的“代理”,负责处理和MAS交互的一切事务。

HLA中的联邦成员与MAS中的Agent都好像是人类社会或组织中一个个单独的人类个体,它们在特定环境中独立存在,又相互联系。这种组织架构的好处在于保证了仿真应用的独立性,将具体的仿真功能与通用的支撑服务分离开来,屏蔽了各个仿真应用的具体实现细节,使各部分可以相对独立地进行开发,然后方便地加入到分布仿真系统中,从而充分利用各自领域的先进技术。

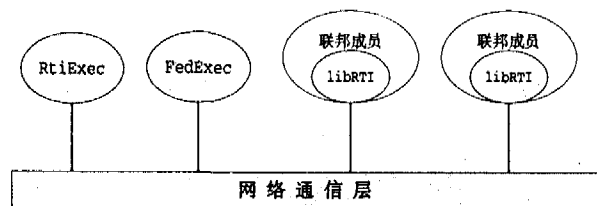


图1 RTI结构逻辑图

1.1.2 相异性

Agent与HLA中联邦成员最大的区别就在于Agent具有更高级更智能化的属性与行为。从Agent的弱概念^[2]来看,几乎所有被称为Agent的硬件或软件都应该具有以下特性:自治性(Autonomy)、社会能力(Sociality)、反应能力(Reactivity)、自发行行为(Pro-activeness)。

而HLA中的联邦成员的定义规范中则没有涉及到独立成员个体的自主行为,只是在联邦运行过程中,赋予联邦成员动态的转移和接收实例属性所有权的能力,使得一个联邦成员可以广泛地应用于其他联邦。RTI提供的服务集是较低水平的行为,仅仅类似于Agent概念中的“行为原语”(Performatives)^[3],因此联邦成员不可能具有目标导向的自主操作或智能行为。

1.2 RTI与Agent通信

通信是分布式系统中的关键性技术。由于系统中各个仿真节点都独立存在,它们之间需要相互联系,通过共同的努力实现系统的目标^[3-13]。

1.2.1 通信机制

HLA框架下一个联邦中各成员间的交互作用是通过

RTI所提供的服务来实现的,即在一个联邦的执行过程中,所有的联邦成员应按照HLA的接口规范同RTI进行数据交换实现成员间的交互作用。RTI在每个联邦成员的主机中都有驻留程序(Local RTI Component, LRC),是联接系统各部分的纽带。联邦成员在开发过程中遵守相应的规则和与RTI的接口规范,在运行过程中也只与本机中的RTI本地组件LRC进行直接交互,其余的交互任务全部由RTI来完成。可见,HLA系统的通信任务实际上是在分布的各个RTI部件之间完成的。在MAS中,通信可以使得多个Agent协调它们之间的动作和行为,使得MAS的行为更好地保持连贯性,从而完成特定的任务。Agent个体之间的交互既可以通过通信服务器(Facilitator)发生信息交互,也可以在它们之间直接交互(通过Facilitator查询其它Agent地址)。与之相比,HLA中的联邦成员之间则永远不会直接发生交互,只能通过RTI。另外,由于Agent所具有的智能特性,因此它们在通信时可以加入更加智能的属性,例如可以区分不同的言语行为采用诸如诚恳、承诺的方式,还可以控制自己的信念(愿望、意图),并且还可具有移动性。

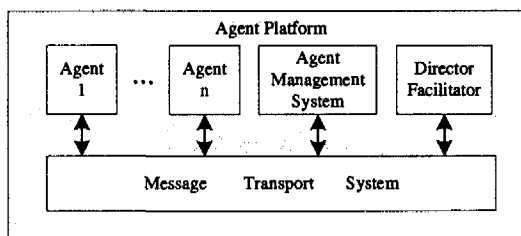


图2 多Agent系统平台结构(FIPA标准)

1.2.2 语言载体

如上节所述,HLA中联邦成员之间的通信是完全依靠RTI来实现的。在联邦运行期间,联邦成员在RTI中声明对象类和交互类的实例,对象间的信息交换是通过其属性值的更新或彼此间发送交互实例来实现的,属性值更新的职责可以由联邦中所有的联邦成员来承担,而所有这些对象和交互的规范都已在对象模型模板(Object Model Template, OMT)中被定义好。

联邦成员间交互的典型过程为:发起交互的联邦成员通过RTI服务发出交互请求,RTI收到请求后将首先根据交互实例的内容更新内部数据,然后RTI将查询内部数据结构(如各种表格),再根据一定的规则(如“公布/订购关系”)将该交互实例正确地传递给需要接收交互实例的联邦成员。

在MAS中,Agent之间是使用标准化的Agent通信语言(Agent Communication Language, ACL)来进行交流的,KQML(Knowledge Query Manipulation Language)和FIPA-ACL是目前流行的并广为接受的两个Agent通信语言标准。KQML提供了一种在MAS中交换知识与信息的语言,是一个由41条通信原语(Performatives)所组成的可扩充的原语集合。它既是一种信息格式也是一种信息操纵协议,支持Agent之间运行时的知识共享。FIPA

协会提出的 FIPA-ACL 中 ACL 规范由一组消息类型以及它们的语言组成,和 KQML 一样,其消息本身表达通信动作。

1.2.3 声明管理

构建一个大规模的分布仿真环境时,如果每个仿真应用的数据都广播给其它所有的仿真应用,就会浪费大量的处理时间和通信资源。为了解决仿真交互中的此类问题,HLA 体系和 MAS 中都提供了一种相似的“匹配”机制,即它们各自在通信环境中所提供的声明服务。

在 HLA 中采用的这种“匹配”机制,即是数据“生产者”向 RTI 声明自己所能“生产”的数据,数据“消费者”向 RTI 订购自己所需要的数据,由 RTI 负责在“生产者”和“消费者”之间来进行匹配。

由此在联邦范围内建立了一种公布和订购关系,以利用 RTI 的控制机制减少网络中的数据量。图 3 描述了声明管理对联邦中信息流的影响。

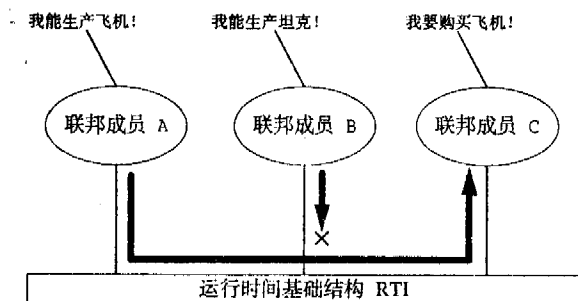


图 3 RTI 声明管理

类似的,在 MAS 中所提供的匹配机制中,服务中介者一般是由黄页服务器和白页服务器(Facilitator)来充当,这些服务器都是一个 Agent 角色。黄页服务器的作用与目录的作用类似,它维护着当前多主体系统中所有主体提供的服务索引;白页服务器其实就是名称服务器(Name Server),它维护着多主体系统中所有主体的网络地址。这样当一个服务请求者从黄页服务器查询到提供某种服务的所有服务提供者后,它可以根据每个服务提供者的名称从白页服务器处查询到它们对应的网络地址,然后直接与它们建立联系。

1.2.4 时间管理

时间是分布仿真中的核心概念。

在 RTI 提供的六大服务中包括了时间管理服务,并支持多种时间管理策略。由时间管理服务来控制 and 协调不同局部时钟管理类型的联邦成员,控制各盟员在仿真时间轴上的推进,保证 RTI 能在适当的时间以适当的方式和顺序将来自成员的事件转发给相应的成员。

与 HLA 不同的是,在传统的 Agent 理论中没有时间管理的概念,不存在类似的实现时间同步的管理模块。当多 Agent 理论应用到系统仿真中时,研究人员通常在 MAS 中开发一个时间管理模块,或者直接设计一个 Agent,负责事件的同步控制,从而实现消息和事件的有效传送和一致。

1.3 OMT 与 Agent 建模

分布式仿真环境中会出现大量的仿真对象,在这些仿真对象之间产生交互,形成一个虚拟的世界。而这些对象模型的建立在 HLA 和 MAS 中也是互有不同。

上文提到了 HLA 中的一种统一的表格——对象模型模板(OMT),它被使用来规范对象模型的描述,从而方便不同类型仿真系统的互操作和重用。在 HLA OMT 中,HLA 定义了两类对象模型。一类是描述仿真联邦的联邦对象模型(Federation Object Model, FOM),另一类是描述联邦成员的成员对象模型(Simulation Object Model, SOM)。OMT 作为一种记录 HLA 对象模型的模板,其目的就是为了使对象模型的各个使用者对该对象模型有准确、一致的理解。

基于 Agent 的建模仿真方法的基本思想是通过模拟现实世界,将复杂系统划分为与之相应的 Agent,以自底向上的方式,从研究个体微观行为着手,进而获得系统宏观行为。

面向 Agent 的仿真建模也会从多个不同的角度为真实系统定义多个对象模型,包括对象和对象状态转移的各种静态及动态关系的完整描述。相对而言,HLA 对象模型(FOM 和 SOM)对系统描述的范围要窄得多,它仅仅关注联邦和联邦成员进行信息交换所必需的要求和能力。在 HLA 中,对象由标识其特征的属性完全定义,属性值更新可以实现信息交换,更新的职责可以由联邦中所有的联邦成员来承担。但在 Agent 模型中,状态以及更新状态的操作(方法)都通过类封装在一起^[11~15]。

1.4 结果讨论

经过以上比较,可看出 HLA 和 MAS 在分布式仿真领域的相似性。这些相似性很大程度上是由于两者所应用的领域相同,所仿真系统的结构和目的也类似的缘故。

同时,HLA 和 MAS 在更多的方面具有差异性。HLA 更多的侧重于仿真资源的重用与互操作,它不考虑如何由对象构建成员,而是在假设已有成员的情况下考虑如何构建联邦,即如何设计联邦成员间的交互以达到仿真的目的,它是一个层次化的体系框架;而 Agent 的优势在于它的自主性能,关注个体的建模和交互,更强的表达能力使之能够有效控制分布式计算的复杂性,这是其它传统方法所不能及的。

2 HLA 规范结合 MAS 的相关研究

虽然关于 HLA 和 MAS 孰优孰劣的讨论一直都存在,但是实际上,随着研究的深入和扩展,越来越多的研究人员更倾向于将两者结合起来,进行优势互补。这种结合不仅能使 MAS 体系建立在较为成熟的 HLA/RTI 框架结构基础之上,而且使 HLA/RTI 的框架更具有 MAS 的智能化和协作的特点。以下是一些相关研究。文献[16]中提出了一种基于 HLA 的多 Agent 系统体系结构和技术框架;文献[17]中提出了一种基于 HLA 的多智能体组织协

作方式;文献[18]在研究多 Agent 仿真方法中采用了 HLA/RTI 作为多 Agent 仿真环境的支撑环境。这些都是将 HLA 和 MAS 相结合的尝试,也是把 AI 技术和仿真技术进行融合的有意义的探索。

3 结束语

HLA 和 MAS 在仿真领域的发展都颇受关注。文中对两者在个体结构、通信和模型建立方面进行的比较,以及对于比较结果的讨论,对于研究人员开发分布式仿真系统是有参考价值的。而把两者结合起来,在 HLA 规范框架下实现多 Agent 系统,将是未来分布式仿真的新方向。

参考文献:

- [1] Crosbie R, Zenor J. High Level Architecture Module1 Basic Concepts(HLA1516)[R]. SCS, Mcleod Institute of Simulation Sciences, California State University, 2001.
- [2] Wooldridge M J, Jennings N R. Intelligent agent: Theory and Practice[J]. Knowledge Engineering Review, 1995, 10 (2): 115 - 152.
- [3] 王文杰,叶世伟.人工智能原理与应用[M].北京:人民邮电出版社,2004.
- [4] 周彦,戴剑伟. HLA 仿真程序设计[M].北京:电子工业出版社,2002.
- [5] FIPA Agent Management Specification[EB/OL]. Foundation for Intelligent Physical Agents. <http://www.fipa.org/>, 2000.
- [6] FIPA ACL Message Structure Specification[EB/OL]. <http://www.fipa.org/specs/fipa00061/>. 2002.
- [7] 胡勤友. Web 上的多主体系统若干关键技术研究[D].上海:复旦大学,2003.
- [8] DMSO. RTI 1.3 - Next Generation Programmer's Guide Version 4[EB/OL]. <http://www.dmsol.com>, 2001.
- [9] 魏晓斌,周盛宗, Bachmendo B, 等. Agent 通信机制探讨[J]. 计算机工程与应用, 2002, 38(5): 66 - 70.
- [10] 黄健,黄柯棣. HLA 中的时间管理[J]. 计算机仿真, 2000, 17(4): 69 - 73.
- [11] 陈凌云,姜振东. HLA 中对象模型的研究[J]. 计算机仿真, 2003, 20(1): 79 - 82.
- [12] 李军,苏国庆. 基于 HLA OMT 的通信对抗建模仿真[J]. 舰船电子对抗, 2004, 27(5): 27 - 30.
- [13] Palmer M. HLA Simulations vs. Agents, Object Services and Consulting[J/OL]. <http://www.objs.com/agility/tech-reports/9807-HLA-vs-ACL.html>. 1998 - 07.
- [14] Lutz R. A Comparison of HLA Object Modeling Principles With Traditional Object - Oriented Modeling Concepts[A]. 97F-SIW-025, Simulation Interoperability Workshop[C]. Orlando, Florida: [s. n.], 1997.
- [15] 罗批,司光亚,胡晓峰,等. 基于 Agent 的复杂系统建模仿真方法研究进展[J]. 装备指挥技术学院学报, 2003, 14 (1): 78 - 82.
- [16] 高志年,邢汉承. 基于 HLA 的多 Agent 系统体系结构研究[J]. 小型微型计算机系统, 2003, 24(13): 336 - 339.
- [17] 陈艳彪,李志刚,黄建明,等. 基于多智能体的坦克分队对抗仿真模型研究[J]. 系统仿真学报, 2004, 16(4): 705 - 707.
- [18] 侯锋,陈洪辉,罗雪山. 基于多 Agent 的 C4ISR 系统建模与仿真方法研究[J]. 光电技术应用, 2004, 19(3): 25 - 30.

(上接第 131 页)

致性是指多个处理器程序的执行次序与共享存储器中共享数据存取次序之间的一致性。高速缓存与内存的数据一致性问题几乎都是由硬件完成的, Intel 在 Pentium CPU 提供了一种“窥探”机制,监听系统总线上的操作,因而 SMP 结构中的高速缓存与内存的数据一致性问题软件透明的。但页面映射目录和页面映射表的缓冲存储(TLB)的一致性,要通过软件来辅助实现。

当一个 CPU 改变内存某个页面目录或映射表中的内容,都可能引起其他 CPU 中的 TLB 内容不一致,就向系统中使用这个映射表的 CPU 发出 flush_to_others(), 毁灭 TLB 中的内容。如果修改的是一个页面表则调用 flush_tlb_one()毁灭一个页面,若修改的是页面目录则调用 local_flush_tlb()冲刷整个 TLB,以保证各 CPU 中 TLB 的一致性。

5 结束语

SMP 系统是一种利用共享内存空间与外围设备实现多进程的计算机体系结构,它最重要的机制在于对内存空间的共享。通过 SMP 所获得同等性能的提高,要比购买

几台独立的机器把它们组合在一起更加便宜和简单,并且它比等待下一代 CPU 面世要快得多。SMP 系统能够以较小的成本换来更大效益,是一种经济的选择。但是,采用 SMP 提高性能也要付出代价的,为了支持 SMP 系统, Linux 内核复杂度和协同的开销大大增加, CPU 必须不能互相干涉彼此的工作,必须要解决 SMP 的启动与进程调度问题,以及一致性问题。随着 Linux 内核不断发展, Linux 操作系统对 SMP 的支持也会越来越完善。

参考文献:

- [1] 杨孟辉,李伟,廖建新,等. SMP 的结构分析研究[J]. 高技术通讯, 2002(2): 21 - 25.
- [2] Schimmel C. 现代体系结构上的 Unix 系统[M]. 张辉译. 北京:人民邮电出版社, 2003.
- [3] Cesati B. 深入理解 Linux 内核[M]. 陈莉君译. 北京:中国电力出版社, 2001.
- [4] 徐晓磊,董兆华,吴建峰,等. Linux 可抢占内核的分析[J]. 计算机工程, 2003, 29(15): 115 - 117.
- [5] 毛德超,胡希明. Linux 内核源代码情景分析[M]. 杭州:浙江大学出版社, 2001.