

层次式 RTI 服务器的设计与实现

姚益平 卢锡城 王怀民

(国防科学技术大学计算机学院 长沙 410073)

摘 要 提出了基于互操作协议的层次式 RTI 服务器的设计思想,给出了层次式 RTI 服务器 YH-*m*RTI 的体系结构,在此基础上介绍了 YH-*m*RTI 各管理服务的实现技术.由于 YH-*m*RTI 中多个 RTI 可以并行执行,因而可大大提高广域网上大规模仿真的效率.

关键词 HLA;RTI;大规模仿真;互操作

中图法分类号 TP391

Design and Implementation of Hierarchical RTI Server

YAO Yi-Ping LU Xi-Cheng WANG Huai-Min

(College of Computer, National University of Defense Technology, Changsha 410073)

Abstract To solve the performance bottleneck of centralized RTI-server, the radical problem is to solve load balance, time and space consistency, and interoperability among multi-RTI on WAN. The implementation of hierarchical RTI server YH-*m*RTI based on interoperability protocol is introduced in the paper. The architecture of YH-*m*RTI is presented. The implementation of the management services of YH-*m*RTI is described. Because all RTI servers in YH-*m*RTI can execute in parallel, the efficiency of large-scale simulation which uses YH-*m*RTI on WAN can be greatly improved.

Keywords HLA;RTI;large-scale simulation;interoperability

1 引 言

面临新世纪的挑战,2001 年世界计算机仿真大会的主题就是:“用仿真构造未来”,并明确提出基于作战模拟的大规模、复杂系统仿真是未来十年计算机仿真研究的重点.未来计算机仿真将朝着超大规模、模糊化、智能化方向发展.

高层体系结构 HLA(High Level Architecture)已于 2000 年 9 月被定为国际分布仿真通用标准 IEEE1516^[1],它为建模与仿真提供了一个通用的技术框架和开放的标准,HLA 的具体实现为 RTI

(RunTime Infrastructure),但 HLA 只规定了 RTI 应提供哪些服务功能,并未具体规定应如何实现及部署 RTI.目前实现的 RTI 大都为集中式的,而由于 HLA 规定,盟员之间不能直接通信,盟员之间的所有通信都必须经过 RTI,因此,集中式的 RTI 服务器在大规模仿真中容易成为影响系统效率的瓶颈,不利于仿真规模的扩充,不适合大规模仿真的需要.因此,为了满足大规模、多层次的分布式仿真需要,我们考虑采用分布的多个 RTI 服务器来平衡负载,但是,多 RTI 服务器环境给分布仿真系统管理带来新的挑战,即如何配置部署多 RTI 服务器的功能和角色,以实现 RTI 服务器间的有效通信,保证

分布环境下系统的时空一致.从逻辑上讲,多个 RTI 服务器的部署有两种结构:分布式、层次式.对于分布式 RTI 结构,每一仿真分组内都有自己的局部 RTI 服务器,盟员只需向本地 RTI 服务器提出请求,本地的服务器如能独立完成该请求则直接提供给本地盟员相应的服务,否则请求远地的其它 RTI 服务器来协同完成.这种结构中由于不存在中心服务器,因此缺乏集中式的全局性操作管理,系统的时空一致性很难得到保证.层次式为分布和集中式相结合的实现方法,在这种结构模型下有一个中心服务器,用于执行一些全局的操作,如时间管理等.为了获得较好的系统可扩缩性,在中心服务器之下设置了一组子服务器,每个子服务器负责一组仿真成员的服务请求,涉及到全局操作的请求则由中心服务器协调完成.层次结构可以解决分布式和集中式结构各自存在的问题,适合于大规模、多层次的分布式仿真需要.

由于 HLA 并没有规定 RTI 之间的互操作模型,因此,在层次式结构中,如何实现多 RTI 服务器之间的互操作成了问题的关键.归纳起来,实现 RTI

之间的互操作有如下 4 种方式^[2]:盟员网关(federate gateway)、盟员代理(federate proxy)、RTI 代理(RTI broker)、RTI 互操作协议(RTI interoperability protocol),其中 RTI 互操作协议是 RTI 之间实现通信的最有效的方法.

因此,基于互操作协议的层次式 RTI 服务器是解决大规模分布式仿真效率瓶颈的最佳方案.本文首先介绍了基于互操作协议的层次式 RTI 服务器 YH-*m*RTI 的体系结构,在此基础上介绍了 YH-*m*RTI 各管理服务的实现技术.

2 YH-*m*RTI 体系结构

为满足大规模、多层次分布式仿真的需要,解决基于不同软(硬)件平台的仿真软件特别是 RTI 服务器与服务器、RTI 服务器和盟员之间的互操作性,我们提出了基于 CORBA 实现分布式多 RTI 服务器的思想,并对此进行了深入研究和实现,该系统的体系结构如图 1 所示.

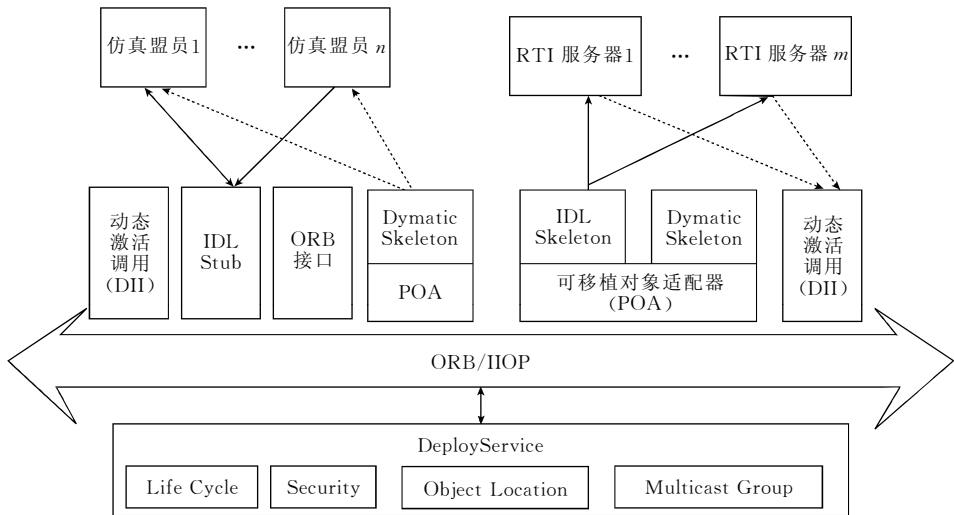


图 1 基于 CORBA 的 YH-*m*RTI 系统结构

图中 RTI 服务器及盟员均构建在 CORBA 中间件之上,其中 RTI 服务器实现 IEEE1516 标准规定的管理功能,为盟员提供一致的接口,盟员通过 RTI 的本地代理透明访问 YH-*m*RTI 的服务,RTI 代理把盟员的请求传递给管理该盟员的 RTI 服务器,RTI 服务器之间通过内部的互操作协议进行数据和状态信息的交换,RTI 服务器通过盟员代理用回调的方式将请求执行结果通知盟员.

图中虚线框中各局部 RTI 服务器与中心 RTI 服务器构成一个逻辑的整体,对盟员来讲这个逻辑的整体就如同一个集中式的 RTI 服务器.盟员只与管理它的本地 RTI 进行通信,本组内的盟员之间的数据交互由本地 RTI 处理并完成,不同组内的盟员之间的数据交互通过管理它们的局部 RTI 之间的通信完成.由于不同服务器可以并发执行,从而提高了仿真系统的效率.

YH-*m*RTI 采用层次式的部署结构(如图 2 所

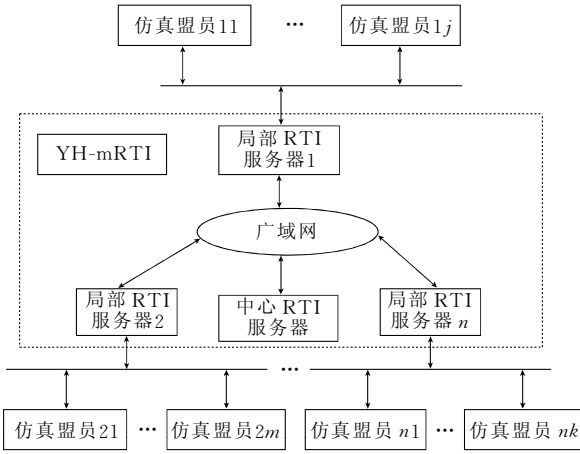


图 2 YH-mRTI 部署结构模型

3 YH-mRTI 各管理服务的实现

使用层次式 RTI 的联盟执行过程大致可分为启

动、执行和结束三个阶段。在启动阶段联盟管理的工作主要有联盟的创建、盟员加入联盟、初始化数据等；声明管理的工作主要有盟员信息的公布/订购，对象管理的工作主要有对象的登记/发现、属性值的更新/反射等；数据分发管理的工作主要有区域的创建和关联等；时间管理的工作主要有事件排序类型缺省值及时间前瞻值(Lookahead)的设置。在执行阶段对象管理的工作主要有盟员属性值的更新/反射、交互的发送/接收等；时间管理的工作主要有时间推进的请求/允许、相关时间值的查询等；声明管理的工作主要有信息的进一步(或重新)公布/订购，数据分发管理的工作主要有区域的创建、修改、删除和关联等；联盟管理的工作主要有联盟状态的保存、恢复等；所有权管理的工作主要有属性所有权的转移。在结束阶段主要由联盟管理完成盟员的退出及联盟执行的删除。使用层次式 RTI 的联盟执行过程概要图如图 3 所示。

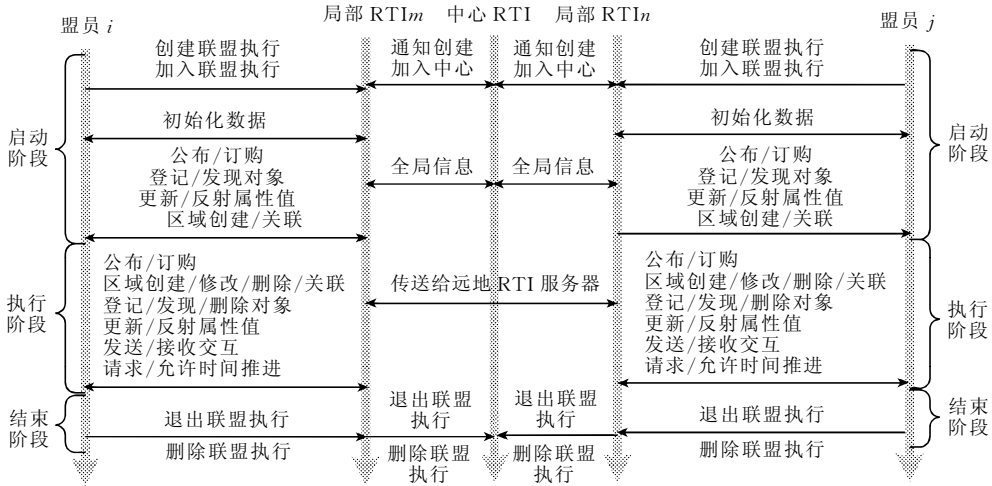


图 3 使用层次式 RTI 的联盟执行过程概要图

下面具体介绍层次式 RTI 各管理服务的实现技术。

3.1 联盟管理(FM)的实现

联盟管理主要提供对盟员与联盟执行的管理服务,用于在分布仿真环境中创建、动态控制和删除一个联盟执行及盟员的加入与退出。在这个过程中,需要作一些初始化工作。联盟的初始化是通过处理本联盟的 FED 文件完成的,每个 RTI 服务器都会根据同一个初始化数据文件生成全联盟一致的各类数据列表^[3]。盟员的初始化主要是对本盟员的公布、订购信息,对象、区域信息的初始化。

联盟执行的创建与删除是由联盟中标明为“管理者”的盟员所使用。联盟管理者在局部 RTI 服

器上创建一个联盟执行后,局部 RTI 调用中心 RTI 服务器中的创建联盟执行服务在中心 RTI 上创建联盟执行,然后中心 RTI 告知其它局部 RTI,联盟执行已经创建,这样其它的盟员就可以通过本地局部 RTI 加入联盟执行。当局部 RTI 上的所有盟员都退出联盟执行后,局部 RTI 从中心 RTI 中退出,同时删除局部 RTI 上的联盟执行,当所有局部 RTI 都从中心 RTI 中退出后,才能删除中心 RTI 上的联盟执行。

3.2 声明管理(DM)的实现

声明管理主要是对分布仿真中盟员能够产生的数据和盟员感兴趣的数据进行说明和动态控制的一种服务。声明管理服务实现了基于类层次的信息过滤,RTI 服务器根据盟员公布和订购信息进行过

滤,只发送盟员订购的类信息,从而有效地减少了网络上的信息流量和盟员的信息处理量。

在多 RTI 服务器系统中,为了提高效率,避免频繁解析远地 RTI 服务器的对象引用和服务调用,减少网络中的消息数,对属于不同局部 RTI 服务器的仿真盟员之间的数据交换,我们采用“精”“粗”两层匹配和过滤策略,其过程是:当订购盟员向本地 RTI 请求订购特定的信息时,本地 RTI 将本地的各个盟员的订购信息合并成一个集合,称为“本地订购集合”,传递给远地 RTI 服务器,该集合就成为远地 RTI 的“远地订购集合”。这样,当需要进行匹配(如公布盟员登记对象实例)时,本地 RTI 服务器 *i* 通过对“本地订购信息”进行精匹配和“远地订购集合”进行粗匹配,就可知道有哪些本地盟员及远地 RTI 需要获取属性更新信息;粗匹配成功后远地 RTI 服务器 *j* 再将自己的“本地订购信息”与粗匹配成功的 RTI 服务器 *i* 的公布信息进行精匹配,从而确定 RTI 服务器 *j* 的属性,更新接收盟员。这种多服务器环境下基于类的两层匹配过滤模型见图 4 所示。

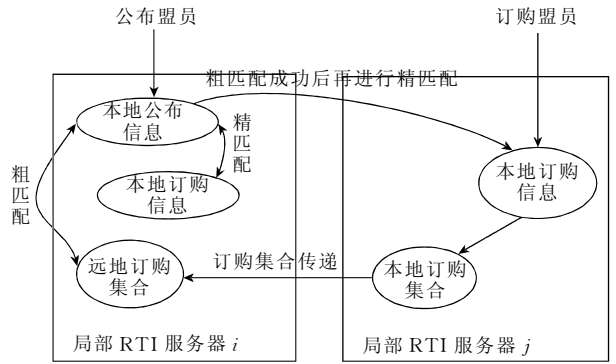


图 4 基于类的两层匹配过滤模型

3.3 对象管理(OM)的实现

对象管理服务主要处理对象的登记/发现/删除、属性的更新/反射、交互的发送/接收等。对象通过发送属性更新或交互来传送对象状态信息或事件。对象管理提供了两种数据交换方式:一种“推”方式,公布盟员主动提供属性更新值;另一种“拉”方式,订购盟员请求对象属性的更新。对象管理服务过程如图 5 所示。

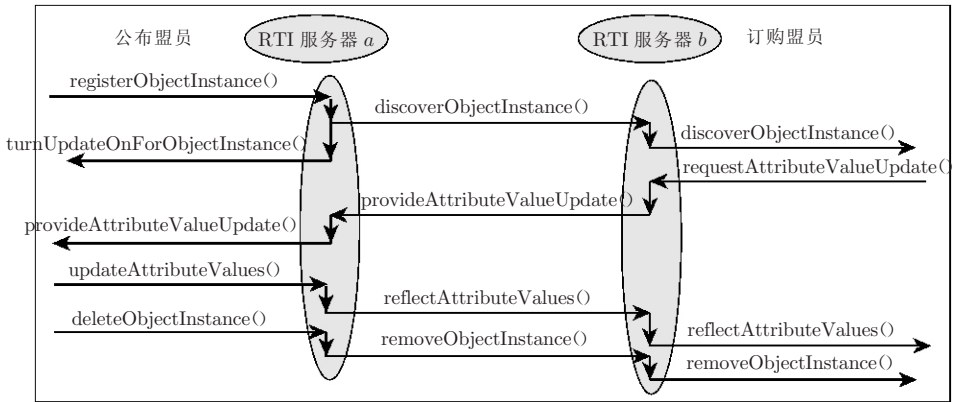


图 5 YH-mRTI 对象的创建、数据交换及对对象删除过程

为了保证层次式服务器中数据的一致性,我们规定对象实例只存在本地,当公布盟员在本地 RTI 服务器登记了对象实例后,不需要向远地 RTI 登记该实例;登记了对象实例的盟员更新了属性值后,本地订购盟员和远地订购了该对象属性的 RTI 服务器就会收到反射属性值的回调。

3.4 所有权管理(OWNM)的实现

所有权管理服务允许在联盟的执行过程中动态转移对象属性的所有权。当多个盟员同时请求拥有某一对象的同一属性时,由于 RTI 服务器是按接收序处理请求的,这就需要一定的数据结构和算法来保证盟员间的公平竞争,为避免 RTI 服务器数据与盟员数据的不一致性,我们设计了一个获取队列

acq 和一个移交队列 div。acq 队列记录盟员获取对象属性的意图,div 队列记录盟员移交对象属性的意图。用这两个队列来解决属性所有权的竞争问题。

在层次式 RTI 服务器实现中,因为客户和服务器的数据修改总存在一定的延迟,不可能完全同步。为了不使网络延迟影响仿真的正确性,我们采用如下策略:RTI 服务器认为调用 attributeOwnershipAcquisition(),negotiatedAttributeOwnershipDivestiture()只是表达了盟员的一种属性所有权转移愿望,并未进行真正的转移,故不进行全局数据的修改,而只将相应项加入到获取队列或移交队列中。盟员在修改属性所有权本地记录之后,发出 attributeOwnershipReleaseResponse()调用,RTI 服务器接收到该调用之

后,才修改全局数据.同理,盟员只有在接收到回调函数 attributeOwnershipAcquisitionNotification()之后,才修改本地所有权信息.

3.5 数据分发管理(DDM)的实现

数据分发管理是在声明管理实现基于类的数据过滤后,实现基于值的过滤,进一步减少盟员接收冗余数据的可能性和网络中的数据流量^[4].在HLA 1516接口规范中,DDM提供与区域相关的服务,部分DM服务等价于以缺省区域作为区域参数的相应DDM服务的调用.但是,由于盟员不存在引用缺省区域的任何方式,所以DM和DDM都要进行实现.同时在IEEE 1516中又规定,不带区域属性订购(DM中服务)和带区域属性订购(DDM中服务)不存在任何相互影响,盟员订购/更新对象属性时可以关联或不关联区域集.当一个盟员不关联区域而更新属性时,它并不能要求其它盟员必须采用不带区域订购服务.DM与DDM中相关服务对数据分

发的影响见图6所示.

是否发送	不带区域属性订购	带区域属性订购
区域无关属性更新	发送	发送
区域相关属性更新	发送	根据区域匹配结果判定

图6 DM与DDM中相关服务对数据分发的影响

大规模仿真系统中,不同服务器之间盟员的公布与订购存在大量的区域,为了减少网络数据流量和RTI服务器处理的区域数,解决过滤效率与系统规模之间的矛盾,我们提出了一种基于区域的两层数据过滤结构模型,实现粗细两层信息的过滤.其中精信息是来自本地RTI服务器所管理盟员的公布与订购信息,粗信息是来自其它RTI服务器在精信息的基础上通过一定的聚集合并算法产生的订购信息.每一个RTI服务器都维护这两类信息.该模型逻辑结构如图7所示.

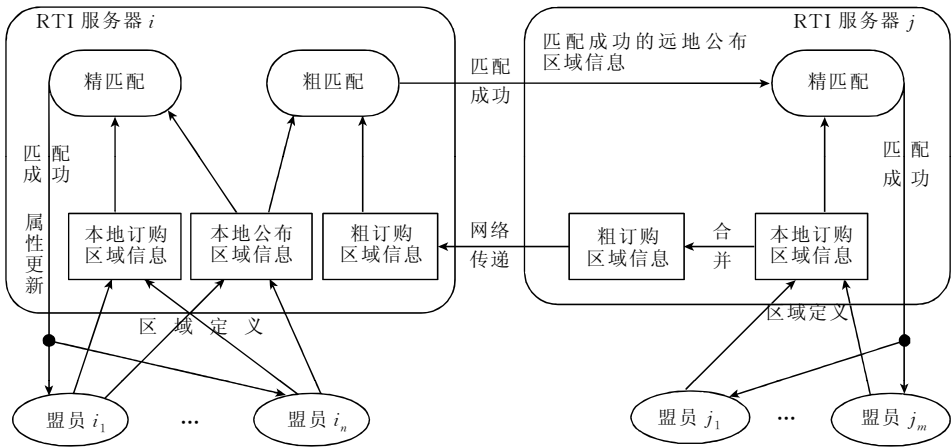


图7 基于区域的两层过滤结构模型

当RTI服务器*i*上的公布盟员更新区域信息时,该更新了的“本地公布区域信息”首先与本地RTI服务器上的盟员提供的“本地定购区域”进行“精匹配”,如果匹配成功,则通知RTI发送更新信息给本地的定购盟员.然后,RTI服务器*i*上的“本地公布区域信息”再与来自远地RTI服务器*j*上的“粗定购区域信息”进行“粗匹配”,如果粗匹配成功,则表示远地RTI服务器*j*上一定有盟员对RTI服务器*i*上的“本地公布区域信息”感兴趣,于是RTI服务器*i*将匹配成功后的“本地公布区域信息”传递给RTI服务器*j*,在RTI服务器*j*中再与“本地定购区域信息”进行“精匹配”,并根据匹配结果确定其后属性更新的接收盟员.当某一盟员的订购区域发生变化时,重新合并订购信息,如果合并后区域没有发

生变化,则不需进行粗匹配;否则,将合并后的订购区域通过网络传递给远地RTI服务器,重复前述的粗匹配及其以后的相关工作.

3.6 时间管理(TM)的实现

时间管理服务的目的是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盟员的消息转发给相应的盟员.YH-*m*RTI中提供了两种类型的消息,即接收序消息(Receive-order,RO)和时戳序消息(Time-stamp-order,TSO),每条消息,不论是发送或是接收,将会或者是一个TSO消息或者是RO消息.中心RTI保存时间管理所需的全局信息,如各盟员的Look-ahead值及各盟员本地逻辑时间等.各局部RTI中对它所管理的每个盟员都维持两个队列:RO队列及

TSO 队列. 当局部 RTI 收到属性更新的消息时, 首先查询该属性的有效订购(即匹配成功的)盟员列表, 如订购者为本地盟员, 则将反映属性更新的消息送相关队列, 如订购者为远地 RTI, 则将消息送远地 RTI, 远地 RTI 再根据该属性的有效订购盟员列表将消息送相关队列. 当局部 RTI 收到时间推进、更改 Lookahead 值等请求时, 则请求中心 RTI 协助完成相关操作(如计算 GALT 值等).

4 结束语

随着大规模复杂系统仿真应用的不断扩展, 研究广域网上多 RTI 服务器之间的负载平衡、时空一致及互操作方式等已成为当前基于 HLA 的分布式仿真应用面临的关键问题. 本文介绍的层次式 RTI 服务器 YH-*m*RTI 既解决了大规模仿真中集中式 RTI 服务器的瓶颈问题, 又减少了全局操作的延迟, 同时也为广域网上各仿真盟员的时空一致性提供了保障. 当然, 由于时间原因, 该系统无论在功能还是性能上, 都有待进一步的测试、完善和优化.



YAO Yi-Ping, born in 1963, M. S. and professor. His research interests include software engineering, distributed interactive simulation and Virtual Reality.

LU Xi-Cheng, born in 1946, vice president of National U-

参 考 文 献

- 1 Simulation Interoperability Standards Committee (SISC) of the IEEE Computer Society. IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)—IEEE Std 1516-2000, 1516. 1-2000, 1516. 2-2000. New York: The Institute of Electrical and Electronics Engineers Inc., 2000
- 2 Myjak M D, Clark D, Lake T. RTI interoperability study group final report. Fall Simulation Interoperability Workshop: Tech Rep: 99F-SIW-001, 1999
- 3 Yao Yi-Ping, Lu Xi-Cheng, Liu Bu-Quan, Lu Gang, Han Lin. Design of data structure of hierarchical RTI server. Journal of Computer Research and Development, 2002, 39(Supplement): 274~278(in Chinese)
(姚益平, 卢锡城, 刘步权, 卢刚, 韩林. 层次式 RTI 中数据结构的设计. 计算机研究与发展, 2002, 39(增刊): 274~278)
- 4 Yao Yi-Ping, Shi Xiang-Quan, Wan Jiang-Hua. Research and implementation of high level architecture. Journal of System Simulation, 2000, 12(4): 364~366(in Chinese)
(姚益平, 时向泉, 万江华. HLA/RTI 的研究与实现. 系统仿真学报, 2000, 12(4): 364~366)

niversity of Defense Technology, member of Chinese Academy of Engineering, Ph. D. supervisor. His research interests focus on high performance computer and high performance network.

WANG Huai-Min, born in 1962. Ph. D., professor, Ph. D. supervisor. His research interests include distributed computing, distributed interactive simulation and information security.