

基于高速以太网的 Ad Hoc 网络的测试仿真系统

秦正龙,胡建平

(北京航空航天大学 计算机学院,北京 100083)

(qinzhenglong@163.com)

摘 要:针对移动 Ad Hoc 网络路由协议调试的重要性和困难性,设计了一个动态的测试仿真系统。通过采取分布式的体系结构,虚拟传输的方法,该系统建立了真实的大规模移动 Ad Hoc 网络测试环境,可以在上面仿真测试设计的网络层和更上层的协议。同时,系统也解决了隐藏节点和暴露节点问题。

关键词:移动 Ad Hoc 网络;仿真;分布式;隐藏节点;暴露节点

中图分类号: TP393.04 **文献标识码:** A

Test and simulation system of the Ad Hoc network based on height speed ethernet

QIN Zheng-long, HU Jian-ping

(School of Computer, Beijing University of Aeronautics and Astronautics, Beijing 100083, China)

Abstract: A dynamic test and simulation system for mobile Ad Hoc network was designed for the importance and difficulty of testing Ad Hoc routing protocols. The system could make the mobile Ad Hoc networks test environment in large scale by using distributed architecture and virtual transmission. The network layer and the upper layer protocols were able to be tested and simulated on it. Moreover, the hided and the exposed nodes were be avoided by the system.

Key words: mobile Ad Hoc network; simulation; distributed; hided node; exposed node

由于有线网络的局限性,无线网络便成为一种必然的发展趋势。但在许多特殊的场合下,有中心的移动网络并不能胜任。于是移动 Ad Hoc 网络以其无中心、自组织、多跳路由、动态拓扑的优点便很快地发展起来,并被广泛的用于军事、抢险、紧急临时场合等诸多领域。由于移动 Ad Hoc 网络的特殊性,人们对其网络的路由和协议的调试就变得十分困难,但网络路由和协议的调试却非常重要,传统的仿真工具在仿真大规模移动 Ad Hoc 网络时,往往不能满足被测协议的准确性、实时性和可扩展性等。

本文以无线局域网传输 IEEE802.11b 协议标准为对象,对其标准的物理层和媒体接入控制层进行仿真。我们采取多线程、分布式结构的技术,基于 Linux 系统内核和系统提供的软件开发工具 GTK+,设计了一个仿真测试系统,最终在高速以太网中完成对移动 Ad Hoc 网络的仿真和测试。

1 系统的体系结构

1.1 系统的网络拓扑

本系统是一个分布式系统,运行在高速以太网中,它至少由四部分组成,分为三类,分别是观察节点(ON)、虚拟节点容器(VNC)、环境控制服务器(ECS),它们由集线器连接,如图 1。搭建了一个仿真测试系统,不需要其他设备,只需在一般的局域网环境下就可以运行。

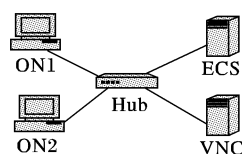


图 1 系统网络拓扑

1.2 系统的体系结构

本系统由 ON、VNC 和 ECS 三种部件组成。ON 上运行着被测协议。ON 就如同真实节点,用户在上面运行应用就如同在真实的移动 Ad Hoc 网络中的节点上运行一样。如果系统中运行多个 ON,两两 ON 之间可以建立多组对话。VNC 上运行着多个 VN。与 ON 不同,VN 不是真实的节点,而只是 VNC 中的逻辑对象。在结构上,VN 和 ON 运行着同样的协议栈,但是从 ON 看来,VN 只是路由器和带宽的竞争者。VNC 负责创建和维护 VN,并负责 VN 和 ECS 之间的通信。系统中可以部署一个或多个 VNC,通过增加虚拟节点的数量来达到增强系统扩展性的目的。ECS 是本系统的核心。它负责维护模拟网络的拓扑信息,模拟移动 Ad Hoc 网络的无线通信信道。ON 和 VN 在开始工作之前要询问 ECS,就像节点在加入一个移动 Ad Hoc 网络时需要向其他节点注册一样;ON 和 VN 在发送数据之前也要询问 ECS,就如同在真实环境下节点在发送数据前需要侦听信道一样。ECS 根据当前的拓扑信息和信道状态来应答 ON 和 VN 的询问。系统的体系结构如图 2 所示。

系统中种有控制通道(Control Channel,CC)和数据通道(Data Channel,DC)两种通道。ON 和 VNC 是通过控制通道 CC 与 ECS 通信的;ON 和 VNC 之间是通过数据通道 DC 通信。CC 和 DC 都是局域网中的 TCP 连接。ECS 首先初始化,当有 ON 或 VNC 加入到系统时,建立 CC。CC 用于系统各部件之间传递控制消息。当 ON 或 VNC 获准传输数据后,使用 DC 传输数据。在 ON 和 VNC 以及 VNC 上运行的 VN 在运行

之前要向 ECS 注册,在结束运行前要向 ECS 注销;ON 和 VNC 在发送数据之前要向 ECS 请求。这些请求都是以消息的形式通过 CC 传到 ECS 的,ECS 的应答消息也是通过 CC 传回到节点。在得到允许之后,节点之间的数据传输是通过数据通道传输的。

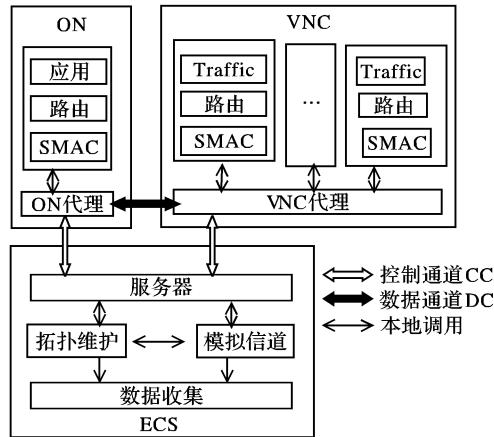


图2 系统的体系结构

2 系统各部分的设计

2.1 观察节点 ON 的设计

观察节点由观察节点代理、SMAC、路由和应用几个模块组成。其中观察节点代理相当于履行了节点物理层职能,模拟信号的接收和发送。另外,观察节点代理还负责通过控制通道与环境控制服务器通信。SMAC(Simulation Access Media Control)是媒体接入模块,采用分布式协调函数 DCF(Distributed Coordination Function)的 IEEE 802.11 协议模型,作为协议栈的数据链路层。SMAC 向上为被测的路由协议提供标准的接口,使路由协议能够直接运行在 SMAC 模块之上;向下 SMAC 通过观察节点代理与内核空间进行数据交换,从而和其他部件通信。所以,SMAC 是设计的核心所在。路由层运行真实的或是仿真的路由协议,对包进行转发。应用层运行真实的应用程序或者自己设计的数据发生器,产生应用数据,以及处理从下层传来的数据。

2.2 环境控制服务器的设计

此部分模拟 IEEE 802.11b 通信信道,维护模拟移动 Ad Hoc 网络的拓扑信息。接收来自观察节点或是虚拟节点容器的发送请求,根据当前的信道情况和拓扑信息,判断是否能发送。如果不能,要在应答消息中说明原因。收集模拟数据,以供分析。

服务器接收来自观察节点或虚拟节点容器的请求传送消息,提取消息中的信息,更新拓扑,查询到目的节点的通信信道是否可用。如果不可用,找出原因,生成应答消息;如果可用,找出请求节点的传输范围和影响范围,将范围内节点的虚拟 IP 地址分类放到应答消息中。发送应答消息,更新信道状态。

802.11b 模拟信道根据发送消息中的信息,计算拟发送节点周围的信道状态的变化,如占用信道时间。用最新的计算结果更新网络拓扑中的信道状态。

拓扑维护模块维护仿真网络的拓扑。接受来自服务器的更新命令,更新节点的位置和运动状态等信息;接受来自服务器的查询命令,返回查询结果。接受来自 802.11b 模拟信道的更新命令,更新节点之间传输信道的状况。

2.3 虚拟节点容器的设计

虚拟节点容器上运行多个虚拟节点,以模拟观察节点之间的移动 Ad Hoc 网络。虚拟节点产生虚拟数据包,与真实节点竞争信道,转发观察节点(和其他虚拟节点)的包。虚拟节点代理管理虚拟节点容器上运行的多个虚拟节点,负责虚拟节点和环境控制服务器以及其他节点的通信。

2.3.1 虚拟节点的设计

虚拟节点由数据生成器、移动 Ad Hoc 网络 TCP/IP 和通信接口三个部分组成。

数据生成器按照观察节点上应用层协议的特点产生虚拟包。TCP/IP 模块运行被测协议,负责将数据生成器产生的数据封装成 IP 包,或负责找到要转发数据包的下一条路由。通信接口负责虚拟节点和虚拟节点代理之间的通信。虚拟节点通过通信接口向虚拟代理注册和注销。还负责与虚拟节点代理的 SMAC 接口通信,发送封装好的虚拟包,提供需转发数据包的下一跳路由。

2.3.2 虚拟节点代理的设计

虚拟节点代理和观察节点代理一样提供两个接口层:SMAC 接口和网络接口。同样虚拟节点代理被分成三个部分:SMAC 接口、SMAC 和网络接口。

SMAC 接口负责接受来自虚拟节点的数据包,验证包的有效性,封装生成帧,然后将帧传到 SMAC,无效则丢弃该包。当有帧从 SMAC 传入,SMAC 除去帧头,生成包,将包转发到相应的虚拟节点。SMAC 与观察节点的 SMAC 相同。只是在准备发送虚拟包时,请求消息中要加入虚拟包的信息。网络接口是如果要求发送的包是虚拟包,那么请求发送消息中要加入虚拟包的信息。

3 系统中各部分的通信

各部分之间的通信有两条数据流:消息传输和数据传输。消息传输是发生在环境控制服务器和其他部件之间;数据传输是发生在观察节点与观察节点、观察节点与虚拟节点容器以及同一个虚拟节点容器和不同虚拟节点容器之间的。并且部件种类的不同,传输的消息和数据种类也随之变化。消息是通过控制通道传输的。使用这些控制消息来驱动仿真,使整个仿真工作正常进行。系统在初始化时建立好控制通道,环境控制服务器随时准备接收来自观察节点和虚拟节点容器的各种请求消息,根据消息的内容构造应答消息予以应答。观察节点和虚拟节点容器的注册、注销和数据传输的控制都是通过这种方式完成的。

4 软件的仿真测试和结果分析

本系统是在 Linux 下用 C 编程实现。环境控制服务器、观察节点和虚拟节点容器都使用多线程编程技术来实现,程序运行在 Linux 的用户空间,因为不需要对 Linux 的内核做任何的修改,所以不用重新编译 Linux,软件就可以正常运行。系统的图形化的用户界面的设计使用 Linux 下的软件开发工具 GTK+ 来完成的,各种网络仿真参数都通过用户界面进行配置。

我们对测试仿真环境作了初步的仿真测试,主要考察节点数量和网络负载对仿真结果的影响。

4.1 节点数量对仿真结果的影响

图3显示了节点数量对吞吐率^[4]的影响。从图3可以看出随着节点数量的增加,吞吐率也随着下降。相同节点数时,

CSMA/CA 机制的吞吐率比 RTS/CTS 的吞吐率高,这也是因为没有考虑隐藏节点问题。节点数量影响着网络负载,节点数量越多,那么网络的负载越重。另外,节点数量越多,无线信道的竞争者就越多,那么信道中的通信量就越多,信道利用率也就越低。所以随着节点数量的增加,网络的吞吐率下降,传输延迟也会增加。信道中的通信量越多就意味着在仿真中产生和处理的通信量也就越多。这样一来,仿真的速度就会降低。

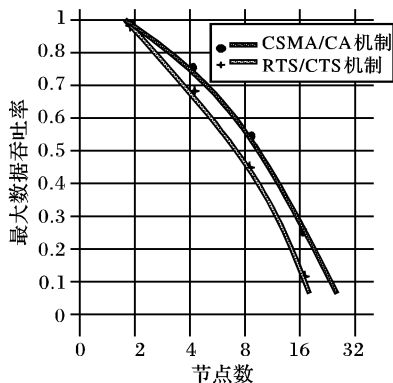


图3 节点数量和吞吐率的关系

4.2 网络负载对仿真结果的影响

在正常情况下,MAC 层对移动 Ad Hoc 网络吞吐率的影响是:点对点机制的吞吐率随网络负载线性增长直到饱和。CSMA/CA 机制和 RTS/CTS 机制在网络负载较轻的时候也随网络负载呈线性增长。但是因为移动 Ad Hoc 层网络中存在隐藏节点问题,当网络负载继续增加时,使用 CSMA/CA 机制,网络开始出现阻塞,吞吐率逐渐趋近于 0。而使用 RTS/CTS 机制,网络并不出现阻塞,而是逐渐趋近于饱和。

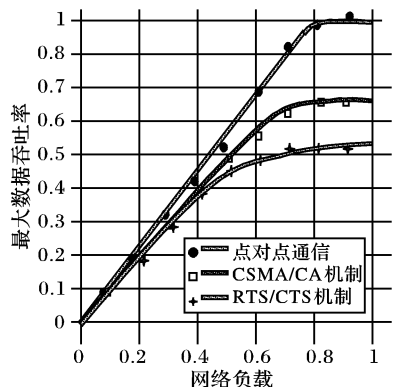


图4 网络负载与数据吞吐率的关系

仿真得出的实验数据如图 4 所示。点对点机制的数据是基本与正常情况下的结果一致,但是 CSMA/CA 机制和 RTS/CTS 机制却有明显的不同。一是 CSMA/CA 机制没有出现阻塞,二是 CSMA/CA 机制的吞吐率比 RTS/CTS 机制的吞吐率

高。这些不同是由于在仿真时没有考虑隐藏节点问题。在这种情况下,对于 CSMA/CA 机制,即使出现隐藏节点,数据照样可传输,所以网络不会出现阻塞。如果考虑隐藏节点的话,当网络负载继续增加时,使用 CSMA/CA 机制,网络开始出现阻塞,吞吐率逐渐趋近于 0。而使用 RTS/CTS 机制,网络并不出现阻塞,而是逐渐趋近于饱和。

4.3 解决隐藏节点和暴露节点问题的策略

采用双信道的方法,即利用数据信道收发数据,利用控制信道收发控制信号和 802.11b 中的 MAC 协议的 RTS/CTS 握手机制来解决,隐藏节点和暴露节点的问题基本得以解决。

但由于需要发送 RTS 和 CTS 帧,所以 RTS/CTS 机制会给无线局域网带来很多额外的开销。当发送短数据包时,RTS/CTS 机制增加了许多开销,所以在发送短数据包时,可以不使用 RTS/CTS 机制。不用 RTS/CTS 机制发送数据包时,需要定义出数据的最小长度,这个长度叫做 RTS 阈值。只有数据包的长度大于 RTS 阈值时,才可以使用 RTS/CTS 机制发送数据。当帧长度超过 RTS/CTS 阈值时,工作节点就采用 RTS/CTS,若没有超过就采用 DCF,这是一种混合模式。用户也可以关闭 RTS/CTS 机制,这样帧长度即使超过 RTS/CTS 阈值,工作节点仍然使用 DCF 来传输数据。

5 结语

网络的路由和协议是当前移动 Ad Hoc 网络的研究重点,而且也是仿真测试的难点,为了让这些协议被有效测试,仿真工具应当能提供逼近实际网络的仿真环境,得出准确的仿真数据供协议研究者使用。本文提出的针对移动 Ad Hoc 网络的特殊性的仿真测试系统,能够在有线网络环境中模拟出无线移动的网络环境来。为了更加逼真地仿真真实的网络环境,应进一步完善 MAC 层仿真,提供上层的接口,使那些用到了 MAC 层功能的上层协议也能仿真,以支持安全性,服务质量等问题的研究。

参考文献:

- [1] SUN J-Z. Mobile Ad Hoc networking: an essential technology for pervasive computing[Z]. 0-7803-7010-4/01/2001 IEEE.
- [2] LUNDGREN H, *et al.* A large-scale tested for reproducible Ad Hoc protocol evaluations[A]. In Proceedings of IEEE Wireless Communications and Networking Conference (WCNC 2002) [C]. Orlando, Florida, Mar. 2002.
- [3] KE Q, MALTZ DA, JOHNSON DB. Emulation of multi-hop wireless Ad Hoc networks[A]. In Proceedings of 7th International Workshop on Mobile Multimedia. Communications (MoMuC'00) [C]. Oct. 2000.
- [4] [http://www.computeruser.com/resources/dictionary/popup_definition.php?Lookup=8410\[EB/OL\]](http://www.computeruser.com/resources/dictionary/popup_definition.php?Lookup=8410[EB/OL]), 2005.

(上接第 1294 页)

- [4] 刘莹,吴建平,王颖. 延迟受限启发式组播路由算法[J]. 高技术通讯, 2003, 13(9): 1-4.
- [5] SHAIKH A, SHIN KG. Destination-driven routing for low-cost multicast[J]. IEEE Journal of Selected Areas in Communications, 1997, 15(3): 373-381.
- [6] ZHANG BX, MOUFTAH HT. A destination-driven shortest path tree algorithm[A]. IEEE International Conference on Communications[C]. Canada (Kingston): 2002. 2258-2262.
- [7] 王涛,李伟生. 低代价最短路径树的快速算法[J]. 软件学报, 2004, 15(5): 660-665.

- [8] WAXMAN BM. Routing of multipoint connections[J]. IEEE Journal on Selected Areas in Communications. 1988, 6(9): 1617-1622.
- [9] SUN Q, LANGERDORFER H. A new distributed routing algorithm with end-to-end delay guarantee[A]. In Proc. IFIP Fifth International Workshop on Quality of Service[C]. New York: Columbia University, 1997.
- [10] SUN Q, LANGENDORER H. Efficient multicast routing for delay-sensitive applications[A]. In proceedings of Second International Workshop on Protocols for Multimedia Systems (PROMS'95) [C]. Austria (Salzburg): Mozart on Multimedia Highways, 1995. 452-458.