

## NS-2 的仿真模拟技术分析

杨玉华, 刘培宁, 刘际炜, 陈涵生

(华东计算技术研究所, 上海 200233)

**摘要:** 网络仿真是研究通信网络技术的有效工具。仿真模拟功能提供了实际网络和仿真器交互的能力。该文介绍了 NS-2 的结构及网络仿真的设计流程, 重点分析了网络仿真 NS-2 中仿真模拟功能的设计和实现。

**关键词:** 仿真; 模拟; 网络仿真; 代理; 网络对象

## Analysis of Network Emulation Technology in NS-2 Simulator

YANG Yuhua, LIU Peining, LIU Jiwei, CHEN Hansheng

(East China Institute of Computer Technology, Shanghai 200233)

**【Abstract】** Network simulation is one of the efficient analysis tools to study the technology of communication networks. Employing an emulation capability in network simulation provides the ability for real-world traffic to interact with a simulation. This paper firstly introduces the architecture of NS-2 simulator, and describes the design of simulation, emphatically analyzes the design and implementation of the emulation facility in the NS-2 simulator.

**【Key words】** Simulation; Emulation; Network simulation(NS); Agent; Network object

网络仿真(Network Simulation, NS)是近几年来被提出的一个新概念。它能迅速地建立起现有网络的模型, 并能够方便地修改模型进行仿真, 使系统具备较好的灵活性和扩充性, 为方案的验证和比较提供了可靠的依据。将通信网络仿真与试验网相结合来研究通信网络的特定性能, 为通信网络的规划、网络通信技术等提供定量分析工具。

NS 是一个事件驱动的网络仿真, 由 LNBL (Lawrence Berkeley National Laboratory) 的网络研究小组开发, 它的前身是 S.Keshav 研制的 REAL 仿真器<sup>[1]</sup>。这是一个可扩展、易配置、可编程的事件驱动仿真引擎 (Simulation Engine), NS 具有开放体系结构, 并带有大量协议库支持, 支持多个流行的 TCP (如 SACK、Tahoe 和 Reno) 和路由调度算法, 其源代码全部公开, 提供开放的用户接口。NS 在网络研究界得到了广泛的应用和支持。

### 1 NS-2 的结构与仿真设计

#### 1.1 NS-2 结构

(1) NS 是一个离散事件模拟器, 仿真网络中传送的包就是事件, 根据包的产生、发送、到达、丢弃等事件驱动时钟。NS-2 中由 scheduler 类处理事件调度。NS-2 结构如图 1 所示。

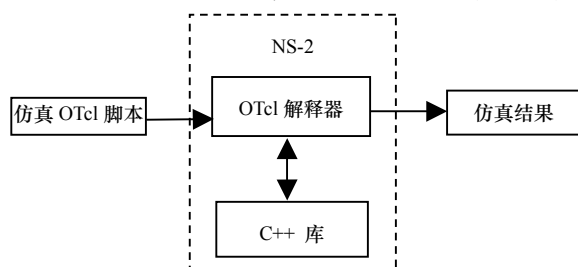


图 1 NS-2 仿真器一般结构

(2) NS 的前端是一个 Otcl 解释器(包含仿真事件调度器、网络组件对象库和网络设置模块库)。仿真器内核定义了有层

次结构的多种类, 称为编译类结构。在 Otcl 解释器中有相似的类结构, 称为解释类结构。这两种类结构关系密切, 从用户的角度来看, 两种结构的类之间有一一对应的关系。用户通过解释器创立新的仿真对象之后, 解释器对它进行初始化, 与编译类结构中相应的对象建立映射。

(3) NS 源于 REAL network simulator 的一个变种, 受到 DARPA 的 VINT 项目的支持, 由 LBL、Xerox PARC、UCB 和 USC/ISI 共同开发。

(4) 模拟平台采用 C++ 语言编写, 并使用 Otcl 语言编写命令和配置接口。

NS-2 采用分层软件结构, 采用面向对象的编程语言 C++ 和 Otcl 语言描述对象。在实现过程中, 控制和数据分离, 控制操作作为外壳部分, 使用 Otcl 实现, 数据处理为内核部分, 使用 C++ 实现。大致可分为 4 部分: 控制语言部分, 事件调度模块, 网络组件模块部分, NS-2 模拟部分。

(1) 控制语言部分。Tcl 是一脚本语言, 它本身不支持面向对象的编程方法, Otcl 是对 Tcl 进行面向对象的扩充后的语言, 它不能单独存在, 必须依附 Tcl 语言, Tccl 是 NS-2 中映射 C++ 与 Otcl 的中间软件层。

(2) 事件调度部分。事件调度部分是 NS-2 的总控部分, NS-2 是一个事件驱动的仿真器, 网络仿真就是在不断地处理事件。在仿真开始前, 已按照网络模拟脚本中的描述生成所有的事件, 按时间顺序排在事件总表中, 仿真开始后, 就依次从事件总表中取事件, 修改时钟, 处理事件。

(3) 网络基本模块部分。网络模块是 NS-2 的主体部分, 它由大量的协议仿真模块、基本网络行为仿真模块组成, 实

**作者简介:** 杨玉华(1978—), 男, 硕士生, 研究方向: 网络通信技术及应用; 刘培宁, 硕士生; 刘际炜, 助理; 陈涵生, 研究员

**定稿日期:** 2004-06-15 **E-mail:** hqy0921@sohu.com

现了基本网络运行和大量网络协议的模拟。

(4) NS-2 模拟部分。编写网络模拟脚本, 模拟各种网络运行, 取得仿真结果。

## 1.2 网络仿真

NS-2 仿真设计流程如图 2 所示。

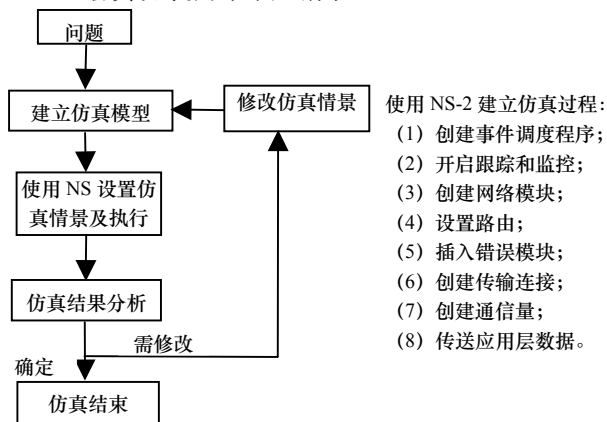


图 2 NS 仿真设计流程

网络仿真是一种利用数学建模和统计分析的方法模拟网络行为, 从而获取特定的网络特性参数的技术。实际上, 建立仿真模型, 编制程序对所研究网络的全部或部分行为进行仿真, 然后运行该程序。在程序的运行过程中采集数据, 最后对该程序的运行结果进行统计分析。在对网络进行仿真的过程中, 不断地对仿真模型及其结果进行一定的推理和验证, 如果正确, 采用此仿真模型; 否则重新确定仿真模型。通信网络的仿真(或模拟)在网络规划和设计、网络运行性能的研究、网络安全、网络通信及协议的研究开发, 以及网络新技术的研究等方面具有相当重要的作用。

## 2 仿真模拟

Emulation 是指将仿真器引入到实际网络。使用仿真器的特别对象能够将实际通信量引入仿真器, 并且能将仿真器加进的通信量引入实际网络<sup>[3]</sup>。对于这样的设备, 依靠仿真器是否作为一个充当路由器的终端站, 还是另一个终端站, 有两种主要的运用。在第 1 种情况下, 实际通信量能够通过仿真器传递(对于端点是透明的), 受仿真器的对象, 或者实际网络中其它通信量的影响。在第 2 种情况下, 仿真器能包含与真实实体通信的通信源或接收器。

### 2.1 仿真模拟的模式

仿真模拟功能可分成两种模式:

(1) 不透明模式 (Opaque Mode): 把真实数据作为不透明数据包。网络数据包是不被解释的, 仿真器不处理它。如图 3 所示。

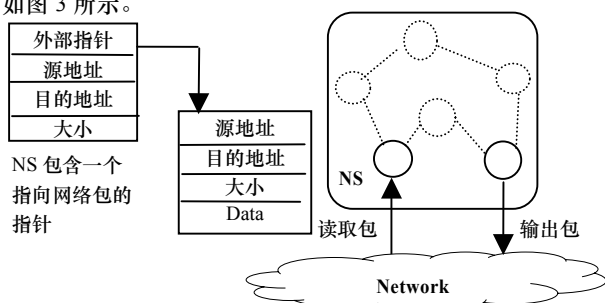


图 3 不透明模式下仿真器传递包

仿真器没有直接处理真实数据包, 真实数据包可能被仿真器丢失、延迟、重新排序或复制, 仿真器充当了一个路由

器, 允许真实数据包不加处理地传递。NS 仿真的数据包含有一个指向网络数据包的指针。不透明模式常用于端到端的应用测试。

(2) 协议模式 (Protocol mode): 真实数据可以被仿真器解释或产生。如图 4 所示。在协议模式中, 仿真器能解析或产生真实网络通信数据。仿真器中协议实现能够与实际网络中同等协议实现符合。这种模式要求 NS 中 TCP 代理能够译解和解析标准协议消息。当前, 此模式下 nse 支持 ICMP, ARP 和 TCP NAT 代理。协议模式能用于协议和一致性测试, 评测 DDoS 攻击的影响和无线网络。

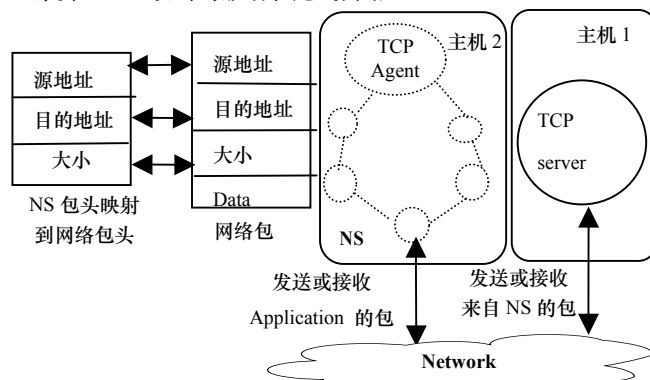


图 4 协议模式下传递包 (TCP 代理)

### 2.2 仿真器和网络通信量的接口

NS-2 提供了一些对象作为仿真器和真实网络的通信接口, 包括 tap agents 和 network objects。图 5 表示了通信接口和仿真器的交互。

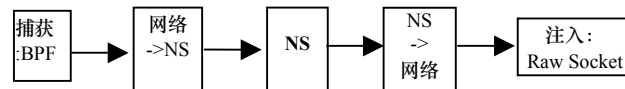


图 5 仿真模拟对象和仿真器的交互

网络对象通过 BPF 和 Raw Sockets 访问真实网络, 提供发送和接收真实数据的入口。Tap 代理在 NS 和真实网络间进行数据包格式转换。

类 TapAgent 是从基类 Agent 派生的一个简单类。那样的话, 它能产生仿真包, 在 NS 通用的包头包含任意指定值。对于引入到仿真器的包, 使用 PT\_LIVE 包类型。虽然多个 tap agent 可配置在一个模拟节点上, 但每个 tap agent 至多有一个相关的网络对象。tap agents 能够向相关的网络对象发送网络包, 和从相关网络对象接收网络包。

网络对象用于访问真实网络(或访问捕获的网络包的跟踪文件)。依靠访问底层网络的具体协议层, 有几种网络对象形式。网络对象提供一个进入真实网络的入口点。当前 NS2 支持 3 种网络对象: pcap/bpf, raw IP 和 UDP/IP。

(1) Pcap/BPF 网络对象。提供了一个 libpcap 库(the LBNL packet capture library)扩展接口。发送/接收链接层帧, 使用 tcpdump/pcap 滤除格式。

(2) Pcap/File 网络对象。类似于 Pcap/BPF 对象, 不同点是从跟踪文件获得网络数据, 而不是从真实网络。那样的话, 对于文件对象, 混杂方式的概念和特殊接口命名是不可行的。另外, 创建跟踪文件的能力仍在研究中。这个设备将提供创建兼容 tcpdump 的跟踪文件的能力。

(3) IP 网络对象。提供了对 IP 协议的原始访问, 允许 IP 网络包(包括头部)完全的详述。使用 raw socket 来实现。另外, raw sockets 接口的标准比其它类型 sockets 稍少。

(4) UDP/IP 网络对象。发送/接收 UDP/IP, IP 多点传送支持(处于开发中)。

### 2.3 Real-time 调度程序

在仿真方式下,使用一种特殊的系统调度程序:Real-time 调度程序。它实现一种软实时调度程序,将仿真器中的事件执行对应到真实时间。

这个实时调度程序试着使事件的执行和真实时间同步。通常它作为列举调度器(list scheduler)的子类来实现。NS 的实时能力仍然处于研究中,它用于将仿真网络引进到真实的拓扑结构中(使用易配置的网络拓扑结构, cross-traffic 等,来进行实验)。仿真器需要和实际网络包到达的速率保持一致,相关的工作是减缓通信数据传输速率,将来这个同步过程不再受限制。

## 3 实验设计

### 3.1 环境描述

测试平台: redhat linux7.3。网络仿真软件:网络仿真软件工具采用 ns-allinone-2.26。NS-2 软件的安装和使用参考文献[1]、[8],使用 nse 编译参考文献[1]。

### 3.2 仿真实例

本实验使用 BPF 和 IP 网络对象建立 emulation 测试。仿真器提供路由能力,从一个接口读取数据帧,通过仿真网络传递包,通过 raw IP 网络对象输出包。

```
set myname "yhy" # 指定主机名
set ifname eth0 # 指定接口
set myaddr "10.0.5.224" # 设置 IP 地址
set stoptime 200.0 # 仿真结束时间 (s)
set owdelay 1000ms # 设定延迟时间
...

set ns [new Simulator] # 创建 simulator 对象
$ns use-scheduler RealTime #采用实时同步的调度程序
set bpfl [new Network/Pcap/Live]; # 作读 IP 数据
$bpfl set promisc_ true # 使用混合模式
set ipnet [new Network/IP]; # 作输出 IP 数据
$ipnet open writeonly
$bpfl filter "icmp and dst $myaddr" # 仅接收 ICMP 包
set pfal [new Agent/Tap]
set ipa [new Agent/Tap]
set echoagent [new Agent/PingResponder]
$pfal network $bpfl # 绑定网络代理到 tap agent
$ipa network $ipnet # 绑定网络代理到 tap agent
# 创建拓扑
set node0 [$ns node]
set node1 [$ns node]
set node2 [$ns node]
# 设置结点间的连接
$ns simplex-link $node0 $node2 1Mb $owdelay DropTail
$ns simplex-link $node2 $node0 1Mb $owdelay DropTail
$ns simplex-link $node1 $node2 1Mb $owdelay DropTail
$ns simplex-link $node2 $node1 1Mb $owdelay DropTail
$ns attach-agent $node0 $pfal; # 包过滤代理
$ns attach-agent $node1 $ipa; # ip 代理
$ns attach-agent $node2 $echoagent # 响应代理
$ns simplex-connect $pfal $echoagent # 连接代理
$ns simplex-connect $echoagent $ipa
...
```

### 3.3 实验分析

实验简单地建立了一个协议模式的 emulation 实例,仿真器中包含一个简单的 ICMP 响应代理。结点结构和包的流向如图 6 所示,结点 n0 中含有一个 tap 代理和相应的使用包过滤器的网络对象。结点 n1 含有一个 tap 代理和相应的 raw IP 网络对象。结点 n2 含有 ICMP 响应代理,简单地接收 ICMP 请求包和构造 ICMP 回应包。结点 n0 上的 tap 代理将过滤得到的包构造新的仿真包,传递给结点 n2。结点 n2 将接收的包作为 ICMP 请求 IP 包,并回应。结点 n1 上的 tap 代理将接收的包引入网络对象,产生 raw IP 包输出。运行 nse,在别的主机上,通过 ping 10.0.5.73 (myaddr),进入仿真器,在输出屏幕上可以看到外部包发送的信息。同样,在 10.0.5.73 上 ping 其它主机,运行 nse 也看到获取发送包的信息。

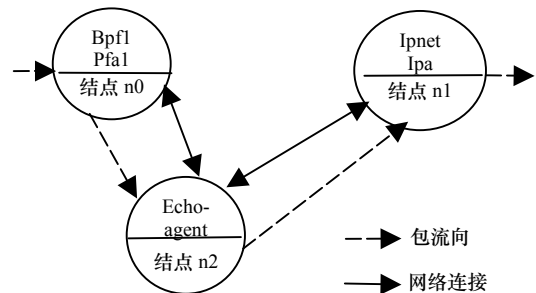


图 6 3 个结点的协议模式拓扑结构和包流程

### 4 结束语

网络仿真是一个分析网络模型、通信协议的有效工具。本文介绍和分析了网络仿真 NS-2 的仿真模拟能力,通过实验以仿真实例说明和分析了仿真模拟设计流程。网络仿真模拟已成功地应用于基于自组网的柯达文件系统<sup>[2]</sup>。当前 NS-2 的仿真模拟功能在开发中,未来的工作包括实现多点传送支持、提高仿真路由能力等方面<sup>[4]</sup>。

### 参考文献

- 1 Fall K, Varadhan K. The NS Manual. <http://www.isi.edu/nsnam/ns/doc/index.html>, 2003-11
- 2 Ke Q, Maltz D, Johnson D B. Emulation of Multi-hop Wireless Ad Hoc Networks. The 7<sup>th</sup> International Workshop on Mobile Multimedia Communications (MoMuC 2000), 2000
- 3 Hussain A. Emulation in NS. [www.isi.edu/nsnam/ns/ns-tutorial/tutorial-02/slides/enulation.ppt](http://www.isi.edu/nsnam/ns/ns-tutorial/tutorial-02/slides/enulation.ppt)
- 4 Fall K. Network Emulation in the Vint/NS Simulator. ISCC, 1999-07
- 5 Fall K. The New NS2 Emulation Facility. <http://www.cs.berkeley.edu/~kfall/tcpimpl98.pdf>
- 6 Greis M. Tutorial for the Network Simulator. <http://www.isi.edu/nsnam/ns/tutorial/index.html>
- 7 Chung J, Claypool M. NS by Example. <http://nile.wpi.edu/NS/index.html>
- 8 李方敏. 网络仿真软件 NS 的结果输出和分析. 计算机工程, 2000, 26(9)
- 9 陈文革. 仿真技术及其应用. 广东省电信科学技术研究院研究院报, 2000, (19)
- 10 姜宁康, 李毓麟. NS 网络仿真技术及其应用分析. 小型微机计算机系统, 2001, 22(4)