

Reinforce Networking Theory with OPNET Simulation

Jinhua Guo, Weidong Xiang, and Shengquan Wang
University of Michigan-Dearborn, MI, USA

jinhua@umich.edu; xwd@umich.edu; sqwang@umich.edu

Executive Summary

As networking systems have become more complex and expensive, hands-on experiments based on networking simulation have become essential for teaching the key computer networking topics to students. The simulation approach is the most cost effective and highly useful because it provides a virtual environment for an assortment of desirable features such as modeling a network based on specific criteria and analyzing its performance under different scenarios with no cost.

In this paper, we present our approach to develop an OPNET simulation networking laboratory that complements classroom lectures. Our simulation labs emphasize the understanding of the dynamics of network protocols instead of configuration and management. Students learn, through these experiments, a wide range of networking aspects including the design and the limitations of protocols, simulation and performance evaluation techniques, interpretation of data and packet analysis. Furthermore, we try to ensure that labs contain some extension or development of the topic beyond the lecture/reading and provide students additional active learning opportunities to discover knowledge.

We have been using OPNET simulation in an introductory computer networks course for the past three years. Feedback from the students has been very positive. Overwhelmingly, students have indicated that the OPNET labs help them better understand the intricate details of actual networking protocols, and they generally indicate that they enjoy these labs as well. In summary, students benefit from the OPNET simulation laboratory in the following three ways. First, the OPNET simulation labs reinforce the networking theory taught by regular lectures. Second, the open design of the labs encourages active learning. Third, students gain the knowledge of modeling and simulation techniques for performance evaluation of networking systems. This active learning approach gives students experience in the subtleties of the design of a complex system, as well as prepares them for the networking industry.

Keywords: OPNET, simulation, modeling, networking, active learning, hands-on experiment.

Introduction

Material published as part of this publication, either on-line or in print, is copyrighted by the Informing Science Institute. Permission to make digital or paper copy of part or all of these works for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage AND that copies 1) bear this notice in full and 2) give the full citation on the first page. It is permissible to abstract these works so long as credit is given. To copy in all other cases or to republish or to post on a server or to redistribute to lists requires specific permission and payment of a fee. Contact Publisher@InformingScience.org to request redistribution permission.

The explosive growth of the Internet in recent years has created a need for scientists and engineers, who can maintain, tune, debug, and innovate the networking infrastructure. Mastery of these technologies involves both theory and practice. The IEEE/ACM Computing Curriculum 2001 (Computing Curriculum, 2001) strongly recommends integrating hands-on experimentation and

analysis into networking courses as they reinforce student understanding of concepts and their applications to real-world problems. It also has been stated that laboratory components are absolutely essential for a networking curriculum and deep understanding of networking requires laboratory facilities that allow one to build, observe, experiment, and measure (Comer, 2004).

Many different approaches have been taken in developing “hands-on” laboratory-based networking courses. Some laboratory courses focus on network management and configuration (Brown, 2002; Fabrega, 2002; Fitzhugh, 2002; Kneale & Box, 2003) and even domain network administration (Nakagawa, Suda, Ukigai, & Miida, 2003). There is also some laboratory environments focus on one layer of networking protocols, such as link-layer network traffic analysis (Jipping, Bugaj, Mihalkova, & Porter, 2003) and implementation of transport level protocols (Richards, 2001). A few large networking education programs (Mayo & Kearns, 1999; Steenkiste, 2003) provide dedicated hardware in which students can experiment with a number of real networks and develop network functionality at the kernel level. However, such environments require extensive resources to setup and manage, and the high cost of providing dedicated networks makes it financially unworkable for most programs. The Virtual Network System (Casado & McKeown, 2005) is a teaching tool designed to allow hundreds of students working remotely to develop user space programs that function as network infrastructure components.

Most of those networking laboratory courses are introduced as a second course in computer networks. However, due to limited faculty and facility resources, most colleges and universities are only able to offer one networking course, mostly without any laboratory components. In fact, courses that expose students to actual network environments are still mostly absent in undergraduate and graduate curriculum (Kurose, Leibeheer, Ostermann, & Ott-Boisseau, 2002). There is an urgent need to integrate introductory networking courses with laboratory components.

One way to integrate laboratory components into an introductory networking course is with simulation. Network simulation allows students to examine problems with much less work and of much larger scope than are possible with experiments on real hardware. An invaluable tool in this case is the free OPNET network simulator (OPNET, 2006) that offers the tools for modeling, design, simulation, data mining and analysis. OPNET can simulate a wide variety of different networks which are linked to each other. The students can therefore exercise various options available to networks and visually see the impact of their actions just by sitting at their workstations. Data message flows, packet losses, control/routing message flows, link failures, bit errors, etc. can be seen by the students at visible speed. This is the most cost effective solution for universities to demonstrate the behaviors of different networks and protocols.

This paper describes our OPNET simulation laboratory development and our experience using it in introductory networking courses. We begin by describing our teaching objectives and discussing why we choose it over other major simulation software packets. We then present the representative projects of the OPNET simulation labs. Our simulation labs emphasize the understanding of the dynamics of network protocols instead of configuration and management and contain some extension or development of the topic beyond the lecture/reading. Finally, we report our experiences and assessment using it in an introductory networking course.

Network Simulator Selection

Our teaching goal is to effectively integrate laboratory components into the introductory networking course without significantly increasing the workload of both instructors and students. The main objectives of our simulation laboratory experiments are:

- To reinforce the networking theory taught in classes with hands-on experiments. In our lectures, we teach networking concepts and protocols at a relatively abstract level. We

hope that hands-on lab exercises lead to a deeper understanding of networking principles and concepts.

- To allow students to build, observe, experiment, and measure variety of networks including direct link networks, switched networks, wireless networks, and inter-networks.
- To balance the breadth and depth of knowledge in an introductory networking course and drive some topics down to a level of details where students understand the elegance of the engineering that make this all work.
- To provide additional learning opportunities to discover knowledge.
- To provide an open lab environment so that all the lab experiments can be completed without supervision and in relatively short time (a few hours).

To meet these objectives, the following properties are essential for the network simulator to be used for the laboratory experiments:

- Ability to simulate a wide range of networking technologies: The simulation software could be used to model the entire network, including its routers, switches, protocols, servers, and the individual applications they support. It should support a large range of communication systems from a single LAN to global satellite networks.
- Ease of use: the simulation software should be easy to install and use. Students should be able to use the software to complete the lab assignments independently without any formal training.
- Free or low cost: The software should be free or low cost. In order to provide the open lab, students should be able to download and install the software on their personal computers.
- Higher simulation performance: For each lab assignment, students are required to create network model, run simulation, analyze results, and write a report. It is very important to have a high performance simulation engine so that simulations of most lab experiments can be completed in relatively short time (less than 30 minutes).

Other properties desired but not absolutely necessary are:

- Suitability of the software for use in research: the simulation software can be used for the simulation-based networking research.
- Better industry employment opportunities for students: the software should have a large user community and should be widely used by industry. So, students who have been taught using the software should be able to immediately apply their knowledge of network simulation when first employed.

Why OPNET?

There are various simulation experiment environments. Many target a specific area of research interest – a particular network type or protocol, such as wireless networks by GloMoSIM (GloMoSIM, 2001). Some systems, such as x-Sim (Brakmo & Peterson, 1996) and Maise (Bagrodia & Liao, 1994), focus on allowing the same code to run in simulation and on a live network. OPNET and NS-2 (NS2, 2006) are the two most popular network simulators, targeting a wider range of networks and protocols. NS-2, derived from REAL (Keshav, 1988), is an open source network simulator. NS-2 is widely used for network research in academia. NS-2 is also free. However, NS-2 is more difficult to learn and lacks a user interface. It requires the users to learn and use non-standard scripting interfaces such as tcl. It takes a significant amount time to get

familiar with NS-2. OPNET is the best network simulator to meet our teaching goals for the following reasons:

- OPNET is much easier to use than NS-2. It provides a very convenient Graphic User Interface (GUI) and is very easy to learn.
- OPNET can be used to model the entire network, including its routers, switches, protocols, servers, and the individual applications they support. A large range of communication systems from a single LAN to global inter-networks can be supported.
- OPNET software (with model source code) is available for FREE to the academic research and teaching community. Students can download and install OPNET IT Guru Academic Edition at home.
- The OPNET's discrete event engine for network simulations is the fastest and most scalable commercially available solution. It usually takes just a few minutes to complete simulations of most lab experiments.
- OPNET has a large user community. OPNET software is used by major fortune-500 companies, service providers, and government organizations worldwide. Students who have experiences with OPNET simulator will have much better future employment opportunities in industry.

Representative Projects

Our labs emphasize the understanding of the dynamics of network protocols instead of configuration and management. Students learn through these experiments a wide range of networking aspects including the design and the limitations of protocols, simulation and performance evaluation techniques, interpretation of data and packet analysis. Furthermore, we try to ensure that labs contain some extension or development of the topic beyond the lecture/reading.

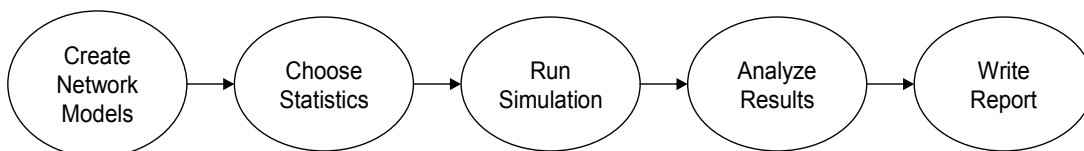


Figure 1: Lab Experiment Workflow

Each lab experiment consists of the following five steps: create network model, choose statistics, run simulation, analyze results, and write a report, as seen in Figure 1. In addition, each lab has a few questions based on the reports generated from the simulation to test the students' understanding as well as analytical and reasoning skills.

OPNET provides four editors to develop a representation of a system being modeled. These editors, the Network, Node, Process, and Parameter Editors, are organized in a hierarchical fashion, as seen in Figure 2. Each level of the hierarchy describes different aspects of the complete model being simulated. Models developed at one level of the hierarchy are used (or inherited) by models at the next higher level. This leads to a highly flexible simulation environment where generic models can be developed and used in many different scenarios.

We design detailed lab manuals so that all lab experiments can be completed without supervision. The following are some representative lab assignments designed around OPNET.

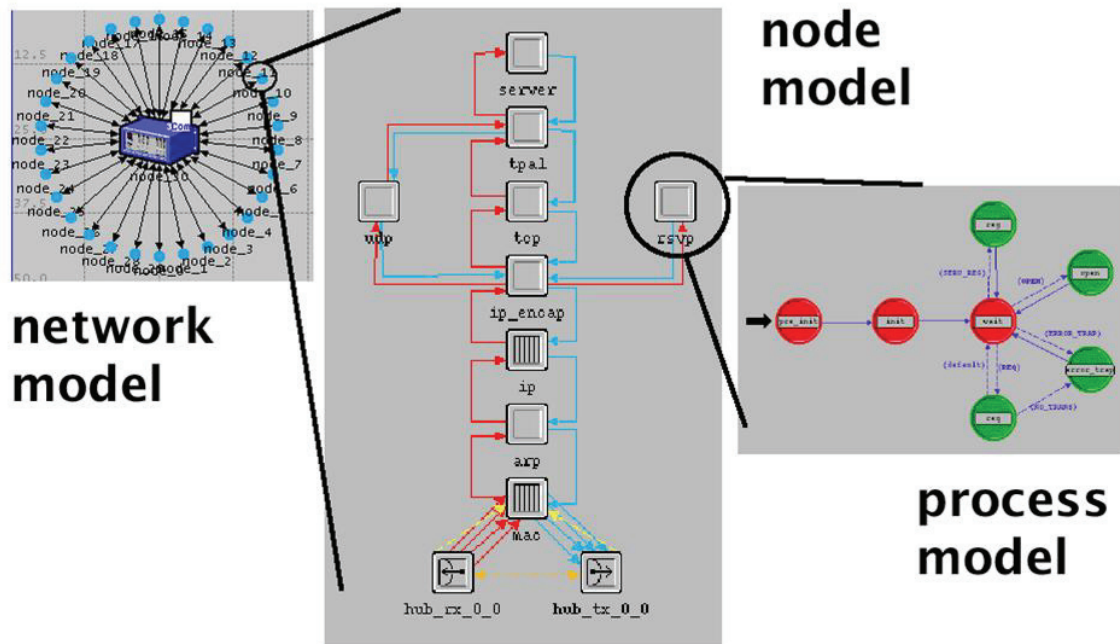


Figure 2: OPNET Graphic Editors for Network, Node, and Process Models

Laboratory 1: Ethernet

This lab is designed to study the Ethernet Carrier-Sense Multiple Access with Collision Detection (or CSMA/CD) protocol. Students will learn that the throughput of Ethernet drops when the load is very high due to the increasingly frequent collisions that are occurring as the network reaches saturation, as seen Figure 3.

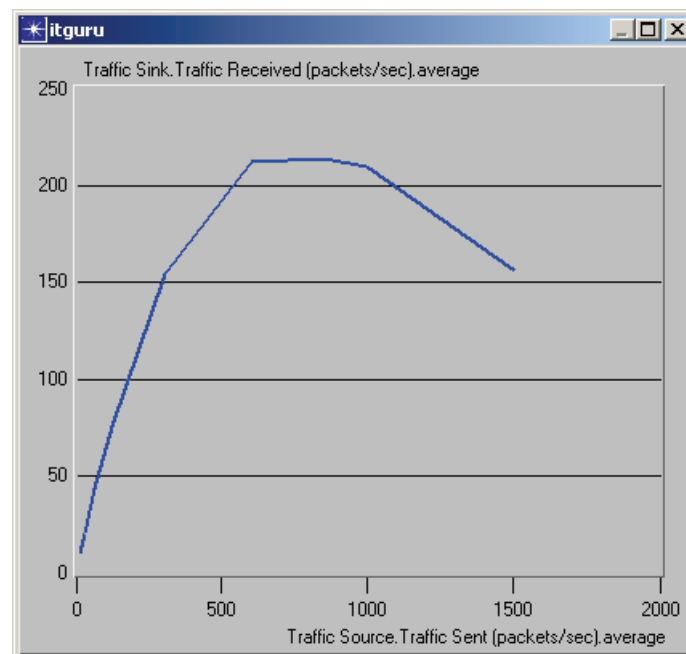


Figure 3: Main results of Ethernet Lab

Laboratory 2: Token Ring.

This lab is designed to demonstrate the implementation of a token ring network. The simulation in the lab will help students examine the performance of the token ring network under different scenarios. In this lab, students will set up a token ring network with 14 nodes connected in a star topology, as seen in Figure 4. Students will study how the utilization and delay of the network are affected by the network load as well as the THT.

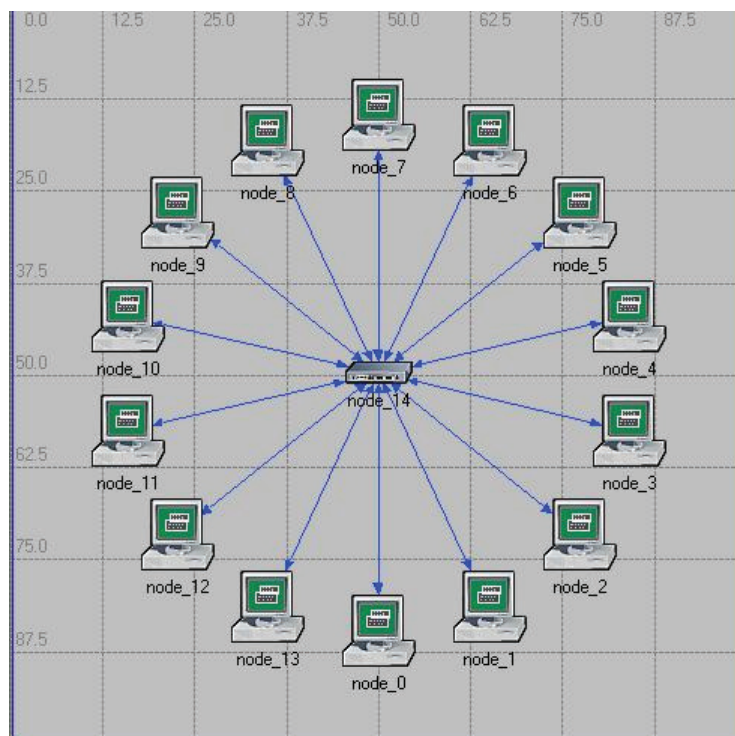


Figure 4: Token Ring Network, Star Configuration

Laboratory 3: Switched LANs

This lab covers LAN switching in Ethernet networks. The lab exercise studies how packets are forwarded between LANs and how the path of packets between source and destination are determined. The simulation in this lab will help students examine the performance of different implementations of local area networks connected by switches and hubs.

Laboratory 4: ATM

The objective of this lab is to examine the effect of ATM adaptation layers and service classes on the performance of the network. ATM provides QoS capabilities through its five service classes: CBR, VBR-rt, VBR-nrt, ABR, and UBR. In this lab, students will set up an ATM network, as seen in Figure 5, which carries four applications: Video, Voice, Email, and FTP. Student will study how the choice of the adaptation layer as well as the services classes can affect the performance of the applications.

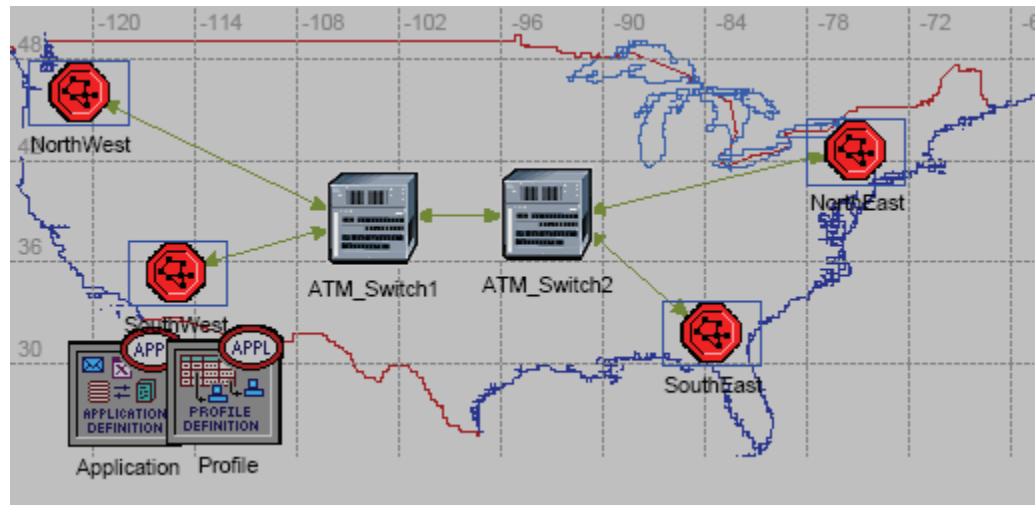


Figure 5: ATM Network

Laboratory 5: RIP: Routing Information Protocol

This lab explores the routing Information Protocol (RIP), as seen Figure 6. Students configure and analyze the performance of the RIP under different scenarios. Experiments illustrate the count-to-infinity problem in the distance vector routing protocol.

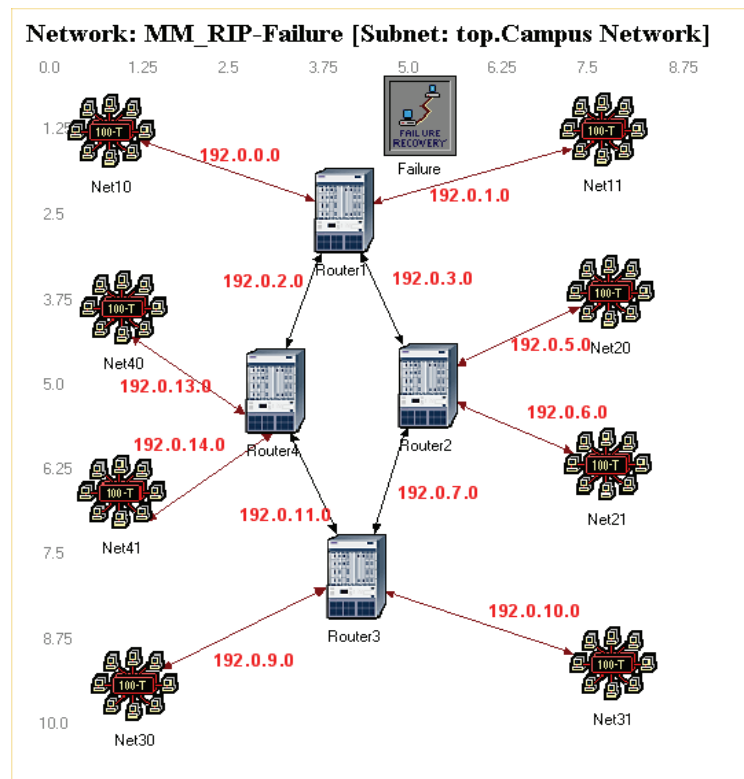


Figure 6: RIP Routing Example

Laboratory 6: OSPF: Open Shortest Path First

The objective of this lab is to configure and analyze the performance of the Open Shortest Path First (OSPF) routing protocol. Students explore the advantage of hierarchical routing and load balancing with the OSPF.

Laboratory 7: TCP: Transmission Control Protocol

This lab is designed to demonstrate the congestion control algorithms implemented by the Transmission Control Protocol (TCP). The lab provides a number of scenarios to simulate these algorithms. Students compare the performance of the algorithms through the analysis of the simulation results.

Laboratory 8: Networking System Design

The objective is to have a design that maximizes the network performance, taken into consideration the cost constraints and the required services to be offered to different types of users. In this lab, students will design a network for a company that has four departments: Research, Engineering, E-Commerce, and Sales, as seen in Figure 7. Students will utilize a LAN model that allows them to simulate multiple clients and servers in one simulation object. Students will define a profile that specifies the pattern of applications employed by the users of each department in the company. Students will study how different design decisions can affect the performance of the network.

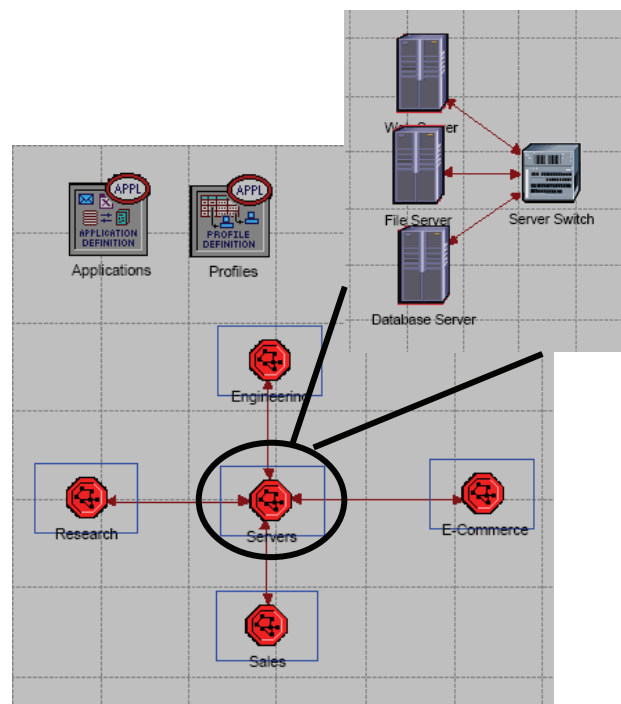


Figure 7: Networking System Design

Assessment

We have been using OPNET simulation in an introductory computer networks course since Fall 2003. Before the start of the simulation labs, we usually spend one hour discussing how the Discrete Event Simulation works, how to install and use the OPNET software, and some basic traffic models and statistical approaches. Students then work without supervision in persistent small

groups (two or three students). We also implement a Principal Investigator approach. For each project, students take turns being the Principal Investigator, who will lead the design, implementation, results analysis, and report writing of the lab experiment in his team. We assign four or five simulation experiments each semester.

To assess the efficacy of the simulation experiments in enhancing student learning, we utilize lab report questions and student survey. Each lab has several questions to test the students' understanding of relevant basic concepts. Based on the results generated from the simulation, over 95% students are able to correctly analyze the data collected and draw the accurate conclusions. In his Ethernet lab report, one student wrote,

“Using the OPNET simulator was very beneficial in visually setting up and observing how the above values impact the overall throughput of the network. The simulator makes it much easier to develop the network. OPNET definitely made understanding the Ethernet easier, especially when a person can graphically see the results. Once the user understands how to select the various parameters to view, he/she can easily look at different areas of the network. The network simulator is definitely a good economic way to get a “first-impression” view of the system being designed.”

From the students' lab reports, we found that the OPNET labs benefit students in the following ways:

- Reinforce the theory (deep understanding of complicated details of actual systems). For example, through the Ethernet lab, students better understood that the throughput of multiple access Ethernet is limited to approximately 30% due to the higher rate of collision when the load increases.
- Encourage active learning. For example, in the classroom lectures, we discuss only the general ideas and basic principles of distance-vector routing and link-state routing algorithms. However, we leave the details of actual implementations of both algorithms, RIP and OSPF, to the labs. We found that students did very well in understanding the details of these protocols through the lab experiments by themselves and were able to correctly answer all the questions. Similarly, students learned the details of ATM five service classes, CBR, VBR-rt, VBR-nrt, ABR, and UBR, on their own through the laboratory experiments.

Table 1: Student Evaluation of Simulation Labs (Average Scores)

	Fall 2003	Fall 2004	Fall 2005
Lab covered course content/objectives	4.4	4.9	4.8
Lab enhanced understanding of basic concepts	4.7	4.7	4.9
Lab well planned and organized	3.9	4.5	4.5
Labs interesting	4.4	4.8	4.6
Labs appropriate difficulty	4.1	4.2	4.4
Number of lab assignments reasonable	3.9	4.0	4.3
Time to complete for PIs (hours)	5.5	5.8	5.7
Time to complete for other participants (hours)	2.5	2.2	2.3
Number of students responding to survey	33	17	26

5 = Strongly Agree, 4 = Agree, 3 = Neutral, 2 = Disagree, 1 = Strongly Disagree

At the end of each semester, we surveyed all students. Feedback from the students has been very positive, as seen in Table 1. Students believe that the simulation labs are well organized and cover course content and objectives. Overwhelmingly, students have indicated that the OPNET labs help them better understand the intricate details of actual networking protocols, and they gener-

ally indicate that they enjoy these labs as well. Students also have indicated that OPNET software is very easy to learn and use. It takes an average of 5.6 hours for the Principal Investigators and 2.3 hours for other team members to complete a lab experiment assignment. On average, a student can complete all the lab assignments in 16 hours. Since students mostly work on their own, it doesn't introduce any workload for the instructors except the grading.

Through the simulation labs, students gain knowledge of modeling and simulation as well. Modeling and simulation is a powerful tool of performance evaluation of networking systems. After several simulation lab experiments, students learned how some ideas and concepts are evaluated through simulations. In particular, they learned how to model the network and network traffic and evaluate new protocols through simulation. We found that students who had experiences with OPNET in the introductory course are more likely to use the OPNET in the advanced networking course and the capstone design course. And, they usually can do a better project.

We also observed that students' technical writing skills were improved. The experimentation reports follow the formats of regular technical reports, including abstract, introduction, design, implementation, experimentation, result analysis, and conclusions. We usually give students some sample reports before their first lab. This really helps students with their technical writing.

Conclusions

An important complement to classroom lectures is laboratory experiments. In networking, this often implies programming, protocol design, experiments and measurement. We believe that simulation has an important role here, since it allows students to examine problems with much less work and of much larger scope than are possible with experiments on real hardware. Simulation can be easier than experimentation because simulators do not need to reproduce all the details of the real world and they can be easily instrumented. In addition, simulation of dozens or hundreds of nodes are easy on limited hardware, many more than is affordable if physical hardware was required.

We have been very happy with our use of the OPNET simulator. Our experiences show that students benefit from the OPNET simulation laboratory in many ways. The OPNET simulation labs reinforce the networking theory taught by regular lectures. The open design of the labs encourages active learning. In addition, students gain the knowledge of modeling and simulation technique for performance evaluation of networking systems.

References

- Bagrodia, R.L. & Liao, W.-T. (1994). Maisie: A language for the design of efficient discrete-event simulations. *IEEE Transaction on Software Engineer, April*, 225-238.
- Brakmo, L. & Peterson, L. (1996). Experiences with network simulation. *Proceedings of ACM SIGMETRICS 96*, 80-90.
- Brown, R. (2002). Network management: A capstone design experience. *32nd ASEE/IEEE Frontier in Education Conference*, Boston, MA, November 2002.
- Casado, M. & Mckeown, N. (2005). The virtual network system. *The Proceedings of the 36th SIGCSE Technical Symposium on Computer Science Education (SIGCSE'05)*, 76-80.
- Computing Curriculum. (2001). *IEEE/ACM CC-2001 Task Force*. Retrieved January 6, 2006, from <http://www.acm.org/sigcse/cc2001/index.html>
- Comer, D. (2004). *Hands-on networking with internet technologies* (2nd ed.). Prentice Hall.
- Fabrega, L. (2002). A virtual network laboratory for learning IP networking. *ACM 7th Annual SIGCSE Conference on ITiCSE 2002*, 161-164

- Fitzhugh, S. (2002). Portable network laboratory. *32nd ASEE/IEEE Frontier in Education Conference*, Boston, MA, November 2002.
- GloMoSim. (2001). *Global mobile information systems simulation library*. Retrieved January 26, 2006, from <http://pcl.cs.ucla.edu/projects/glomosim>
- Jipping, M. J., Bugaj, A., Mihalkova, L. & Porter, D. E. (2003). Using Java to teach networking concepts with a programmable network sniffer. *Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education*, 120-124.
- Keshav, S. (1988). *REAL: A network simulator*. Technical report 88/472, University of California, Berkeley.
- Kneale, B. & Box, L. (2003). A virtual learning environment for real-world networking. *Information Science*, 71.
- Kurose, J., Leibeherr, J., Ostermann, S., & Ott-Boisseau, T. (2002). Workshop report. *ACM SIGCOMM Workshop on Computer Networking: Curriculum Designs and Educational Challenges*.
- Mayo, J. & Kearns, P. (1999). A secure unrestricted advanced systems laboratory. *The Proceedings of the 30th SIGCSE Technical Symposium on Computer Science Education (SIGCSE'99)*, 165-169.
- Nakagawa, Y., Suda, H., Ukigai, M., & Miida, Y. (2003). An innovative hands-on laboratory for teaching a networking course. *33rd ASEE/IEEE Frontier in Education Conference*, Boulder, CO, November 2003.
- NS-2 (2006). *The Network Simulator – NS2*. Retrieved February 18, 2006, from <http://www.isi.edu/nsnam/ns>
- OPNET (2006). *OPNET network simulator*. Retrieved February 21, 2006, from <http://www.opnet.com>
- Richards, B. (2001). Rtp: a transport layer implementation project. *Proceedings of the Sixth Annual CCSC Northeastern Conference on the Journal of Computing in Small Colleges*, 134-141.
- Steenkiste, P. (2003). A network project course based on network processors. *Proceedings of the 34th SIGCSE Technical Symposium on Computer Science Education*, 262-266.

Biographies



Dr. **Jinhua Guo** is the director of Vehicular Networking Systems Research Laboratory and an Assistant Professor in the Department of Computer and Information at the University of Michigan-Dearborn. He received his Ph.D. in Computer Science from the University of Georgia in 2002. He has worked on a range of important problems in experimental computer systems, spanning distributed systems, scientific computing, mobile and wireless networks, vehicular ad hoc networking, and security and privacy. He was a recipient of the IEEE/ASEE Frontiers in Education New Faculty Fellow Award.



Dr. **Weidong Xiang** received his M.S.E.E. and Ph.D. degrees from Tsinghua University, Beijing, China, in 1996 and 1999, respectively. From 1999 to 2004, he worked as a Postdoctoral Fellow and then Research Scientist in the Software Radio Laboratory (SRL) at Georgia Institute of Technology, Atlanta, USA. In September 2004, he joined the ECE Department, University of Michigan-Dearborn, as an Assistant Professor. His research interests include vehicular networks, software radio and cognitive radio, ultra-wideband (UWB), and wireless networked control systems.



Dr. **Shengquan Wang** is an assistant professor in the University of Michigan - Dearborn. He received the Ph.D. degree in Computer Science from Texas A&M University in 2006. He received the BS degree in mathematics from Anhui Normal University, China, in 1995, the MS degree in applied mathematics from the Shanghai Jiao Tong University, China, and the MS degree in mathematics from Texas A&M University in 2000. His research interests are in the areas of networking and distributed systems with a primary focus on real-time computing and communication, and secure computing and communication.