

# TEACHING COMPUTER NETWORKING USING OPNET IN WINDOWS AND UNIX ENVIRONMENTS

Donald U. Ekong, PhD,  
Department of Electrical and Computer Engineering  
Mercer University, Macon, GA 31207

## Abstract

Teaching computer networking courses can be enhanced by using network simulators. OPNET is a useful simulator that is used for designing networks and evaluating network performance under various test conditions. This paper describes a set of laboratory exercises developed with OPNET in UNIX and Windows environments. One added advantage of this approach is to expose students to working in both environments.

## Introduction

Computer networks have transformed the way we live and the way we do business. Some examples of this transformation include communication (via e-mail), information and resource sharing, and electronic commerce[1, 2].

Most computer networking classes are taught using networking hardware equipment and/or software packages[3-5]. Hardware equipment such as routers, switches, and hubs provide students with hands-on experience and expose them to real world equipment. However, maintaining, upgrading, and managing these equipment can be expensive [5].

Software packages also provide some hands-on experience. These packages can be divided into two main groups: (a) freeware, e.g. Ethereal [6], which are downloadable from the Internet; and (b) commercial packages, e.g. OPNET [7], which may have student editions. Freeware are usually the least expensive, but they suffer from limitations such as lack of support, potential changes in operating systems and/or protocols[8].

OPNET (OPTimized Network Engineering Tools) is a comprehensive software environment for modeling, simulating, and analyzing the performance of communications networks (including the Internet). Recent articles[9-14] have described how OPNET has been used in teaching and research. OPNET has become the industry standard tool for designing networks, and evaluating network performance under various test conditions[7]. Hence, students will benefit from exposure to this product.

In 2003, the W. M. Keck Foundation awarded Mercer University a three-year grant to establish an Advanced Engineering Analysis Center, to enhance the undergraduate experience in Mercer University School of Engineering (MUSE). The center was equipped with OPNET Modeler 9.1.A installed on a Sun Microsystems SUN Blade 2000 server and nineteen SUN Blade 150 workstations. The operating system in this lab was UNIX. OPNET also has an academic version, called OPNET IT Guru, which is free and has a limited feature set. In 2005, this academic version was installed in another laboratory containing Windows-based PCs.

ECE 455 (Computer Networks) is a computer networks class that is taught at the undergraduate (senior) level in the Department of Electrical and Computer Engineering, at MUSE. This course is aimed at introducing the students to various aspects of networks, such as protocols, implementation, and quality of service. In Fall 2003 semester, OPNET was introduced as a laboratory component of the class, and to aid in reinforcing many networking concepts. The first set of labs were developed for use in the UNIX environment[10]. This was further enhanced by adding more labs for use in

the Windows environment, thereby exposing students to working in both environments. This paper describes OPNET-based lab experiments that were developed for both environments.

### Laboratory Exercises

Four laboratory exercises are discussed in this paper. The first exercise was adapted from OPNET’s *Teaching with IT Guru* website[16]. This lab was run in Windows XP environment. The remaining exercises were developed for use in the UNIX environment. These labs were adapted from OPNET Modeler 9.1.A tutorials.

Configuring network models with OPNET usually requires many steps, and this can be overwhelming for new users. In the first exercise, the initial network has been configured for the students. All the students are required to do is to learn how OPNET is used to solve a networking problem. In the remaining exercises, a basic network model has been built for each exercise. The basic models are stored in XML

format. At the beginning of each lab, the respective network model is imported by the student into the student’s project workspace. The student then performs analyses on the basic model. The next step involves modifications of the basic network model, and more analyses of the modified networks. This way the students can focus on learning computer network concepts, and gradually be introduced to learning the steps on how to build network models with OPNET.

**Lab 1- Introduction to OPNET:** In this lab, the students evaluate the Internet connection choices for a small Home PC Network. The network model in this lab consists of three personal computers (PCs) that are connected to the Internet for web browsing, E-mail, audio streaming, games, and FTP (file transfer protocol), as shown in Figure 1. The WAN link connects the home PC network to the Internet, and three Internet servers provide different services to the client PCs. The complete topology is preconfigured except the WAN link data rate between the Router and the Internet

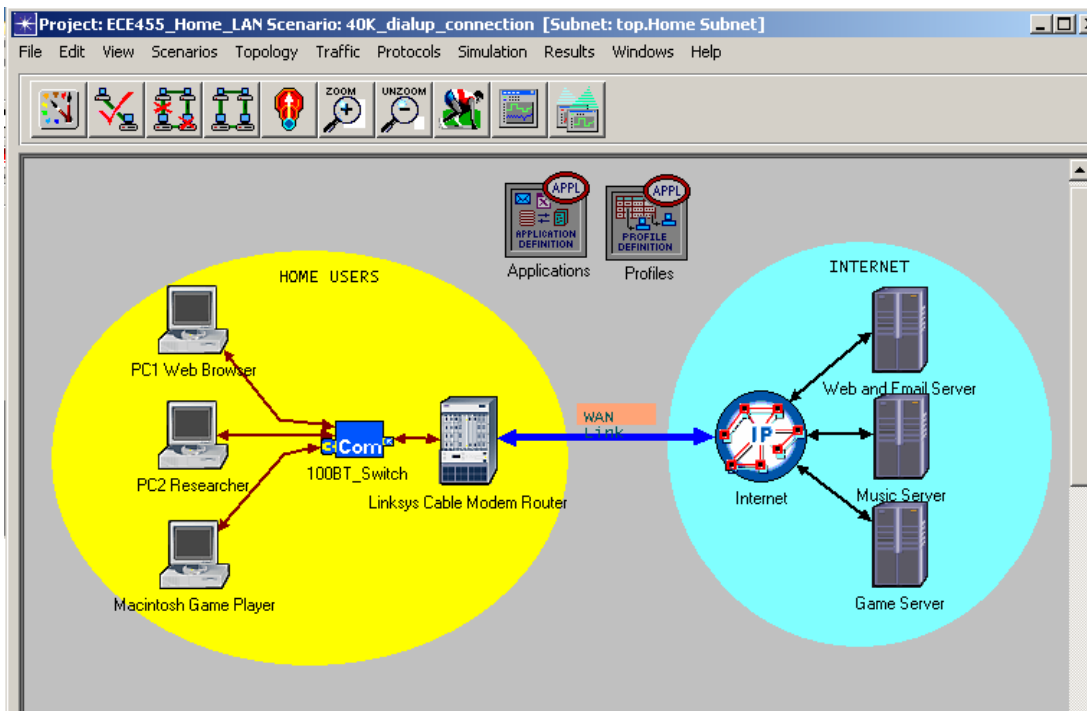


Figure 1: Computer Network For Lab 1[16].

cloud. The students conduct a series of what-if simulations to see how performance differs if the family connects the Internet using (a) a 40 Kbps dial-up line, (b) a cable modem or DSL line downloading at 512 kbps, or (c) a T1 line with download speed of 1.544 Mbps.

Figure 2 shows a typical simulation result for one of the users of this home network, the researcher, when using the 40 Kbps dial-up line. This result shows that the download link utilization averages about 40% of the capacity of the link between the researcher’s PC and the web server; the upload link utilization about 2%. Figure 3 shows that the average response time the researcher experiences when attempting to access a website is about 2 seconds.

Results from the 40Kbps, 512 Kbps, and T1 simulations are compared, and a recommendation on which Internet connection mode is made by the student. The recommendations take into consideration the following factors: cost, and future increase in the number of users.

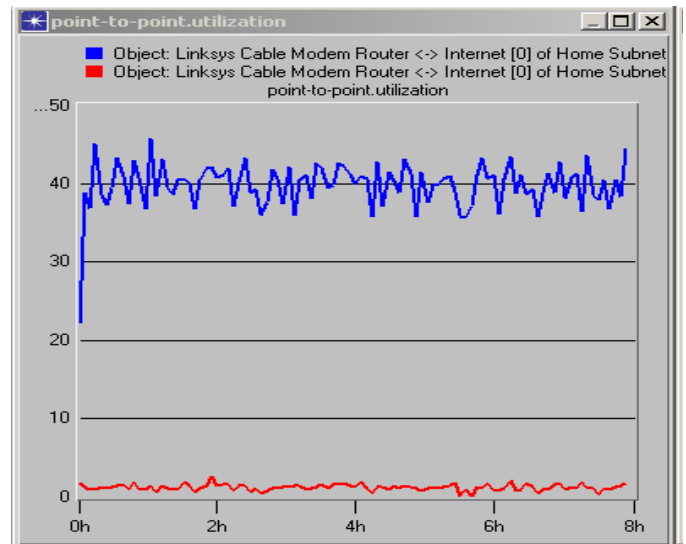


Figure 2: Link Utilization of Researcher’s PC.

**Lab 2 – Office LAN Modeling:** In this lab, the basic model is a company’s LAN (local area network) on a first floor office building ( Figure 4). The XML format of the network model is imported by each student into the student’s respective workspace. The 30 nodes represent workstations that are linked to a database server via a switch. The Applications object in the

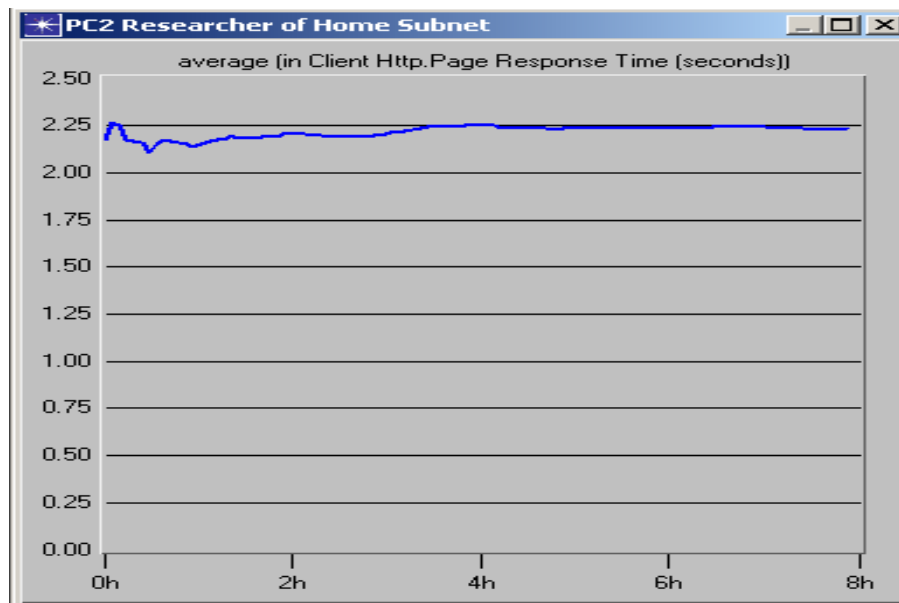


Figure 3: Average Page Response Time of Researcher’s PC.

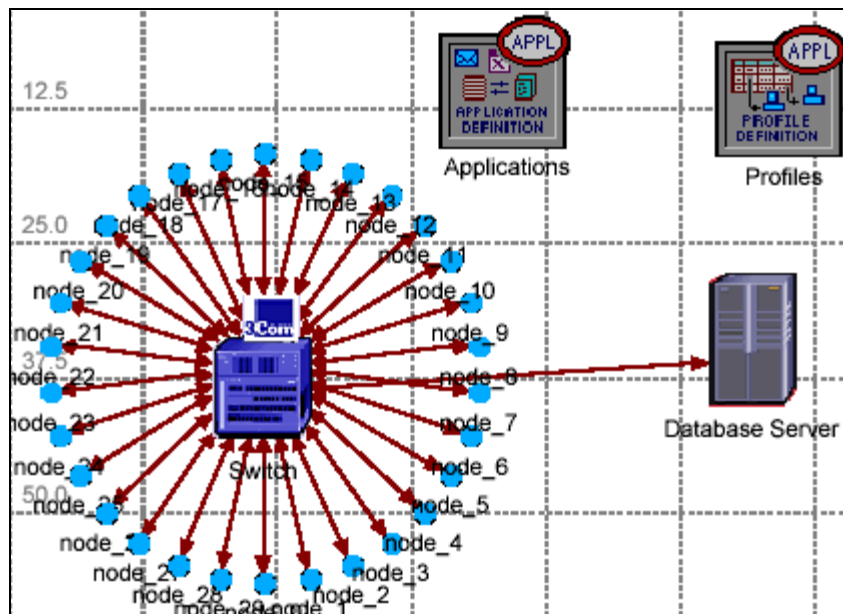


Figure 4: Office LAN Network For Lab 2.

figure specifies the application traffic (e.g. database access, e-mail, web browsing, and file transfer) that can exist in the network. The Profiles object specifies the user profiles that can be modeled in the network. The user profile modeled in this network is light database access - workstations accessing the database server at a low rate.

The company also plans to add an additional network on another floor. The students start by configuring the simulator to collect the following statistics for the first floor network: the database server's Ethernet load and the entire network's Ethernet delay.

A typical simulation result is shown in Figure 5. The vertical axis represents the time average of the database server's load, in bits/sec, while the horizontal axis represents the simulation time. From this result, the students can see that the average load appears to be leveling off (i.e. not monotonically increasing), indicating a stable network.

After the first network's statistics are collected, the students expand the network model to include a second floor segment that

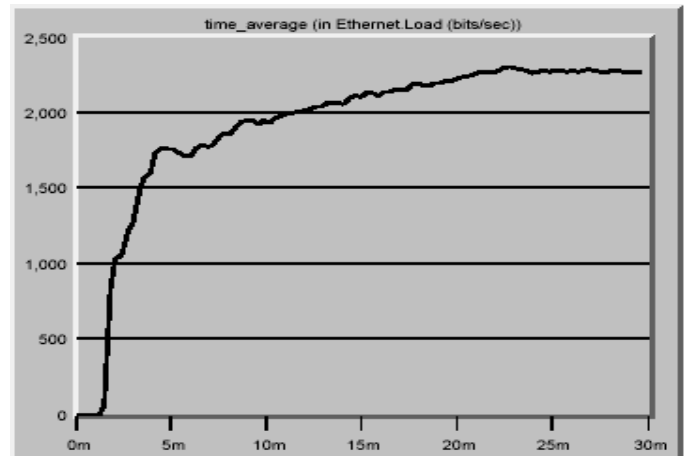


Figure 5: Database Server's Load.

consists of a 30-workstation star topology, similar to the first network. The second floor network does not have a database server; it is connected to the first floor database server via a router. After the expanded network model is completed, students collect statistics (the database server's Ethernet load and the entire network's Ethernet delay), and compare them with previous statistics that were collected for the first network. The statistics are used in determining if the database server will be able to handle the additional load of a second network,

and if the total delay across the final network will be acceptable once the second network is installed.

**Lab 3 – Modeling a WAN:** In this lab, the basic model is a company’s WAN (wide area network). The company has subnets in its offices in Atlanta, Dallas, Denver, and Minneapolis. All the offices are connected to the Internet via 56K point-to-point protocol (PPP) links (Figure 6). The network model also includes profiles of the following types of users and their locations: Engineers in the Atlanta and Denver offices, Researchers in the Minneapolis office, and Salespersons in the Dallas office. The model makes use of the following profiles that have been defined by OPNET: engineers – light web browsing, light e-mail use, light telnet use, and light file transfer; researchers – heavy web browsing and light e-mail use. e-commerce customer – light web browsing; and sales persons – light database access, light e-mail use, light web browsing, and light file printing. The applications (e.g. web browsing) are supported by servers in the Atlanta office (Figure 7).

Users of the company’s network are experiencing high delays. The students create models of two upgraded networks. The first upgraded model replaces the 56K PPP links with DS-1 PPP links; the second upgraded model uses DS-3 PPP links. The students run simulations for all three networks (56K, DS-1, and DS-3) in which the HTTP Page Response Time statistic is collected for each network. This is the average time it takes to retrieve an entire web page. A comparison of the data collected for each network is used to determine whether the company should upgrade to a DS-1 link or a DS-3 link.

The average HTTP page response time for the 56K link is shown in Figure 8. The vertical axis represents the average time, in bits/sec, while the horizontal axis represents the simulation time. The students then try to improve the response time by upgrading the links in the WAN. The simulation results enable the students to determine whether the company should upgrade to a DS1 link or a DS3 link.

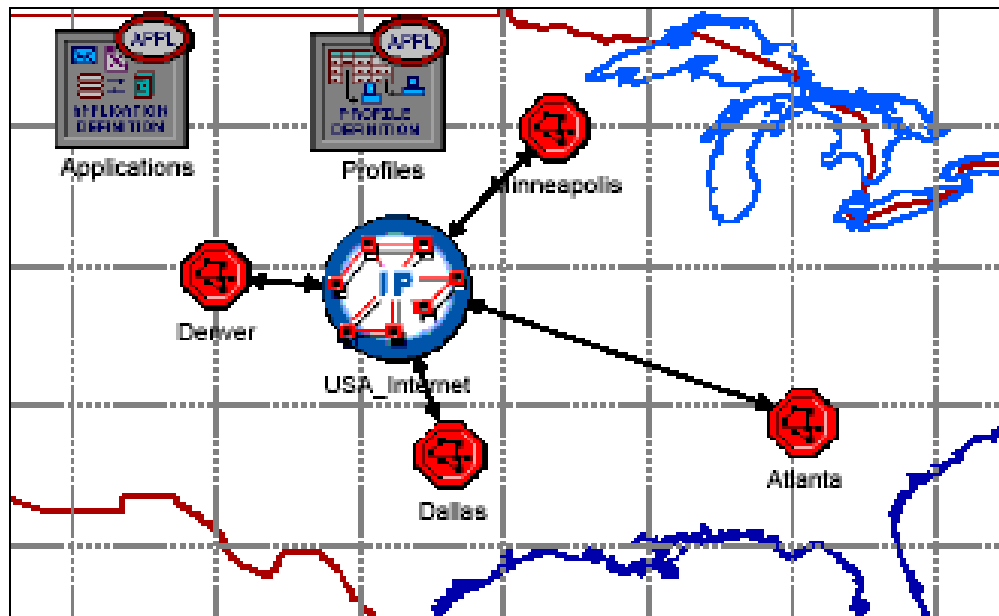


Figure 6: Company WAN Network For Lab 3.

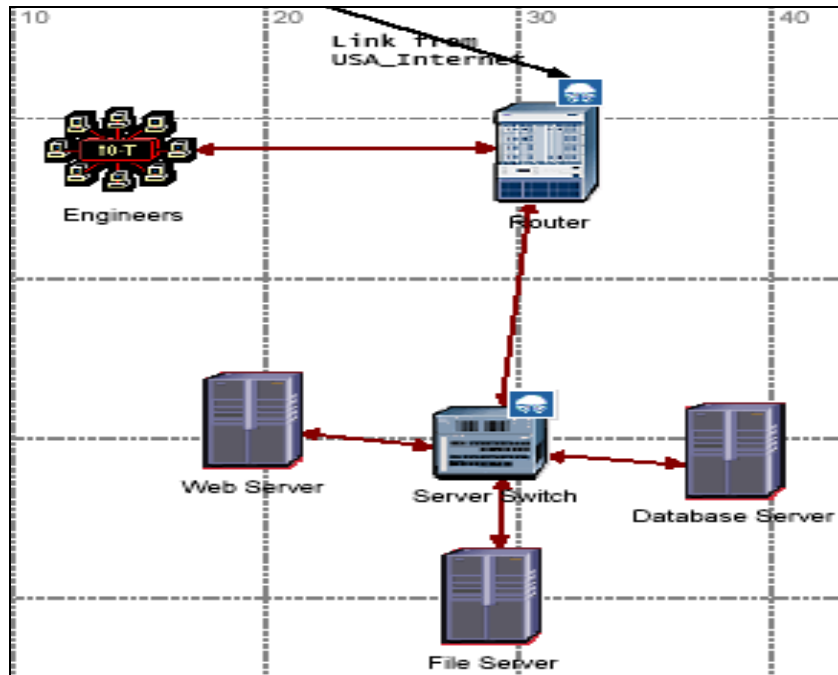


Figure 7: Atlanta Office Subnet (Lab 3).

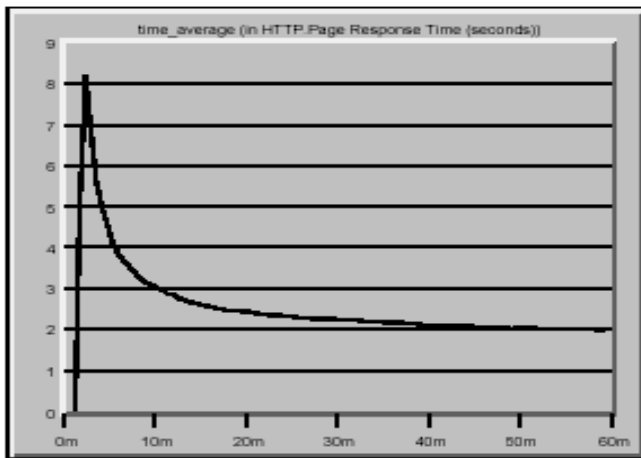


Figure 8: Average HTTP Page Response Time for 56K Link (Lab 3).

**Lab 4 – Transmission Control Protocol (TCP):** TCP is a network protocol that provides a reliable connection service to applications. TCP congestion control algorithms allow a source to determine how much capacity is available in the network, so it knows how many packets it can safely have in transit. For each connection, the source maintains a congestion

window in order to limit the amount of data that the source can have in transit at any given time. In this lab, the students study the effect of different network configurations on TCP congestion windows.

The basic network model in this lab consists of a FTP server located in Denver, CO, and a FTP client located in Chicago, IL, as shown in Figures 9 and 10. The FTP client is connected to a router, and the router is connected to the Internet. The FTP server is also connected to a router, and the server’s router is connected to the Internet.

The basic network model represents a perfect network that does not discard packets. The students create more network models. The new models discard packets, e.g. with an Internet packet discard ratio of 5%. Each network model is configured to collect statistics for the FTP server’s congestion window. Simulations are run and the students can see how discarded packets affect each network’s TCP congestion window.

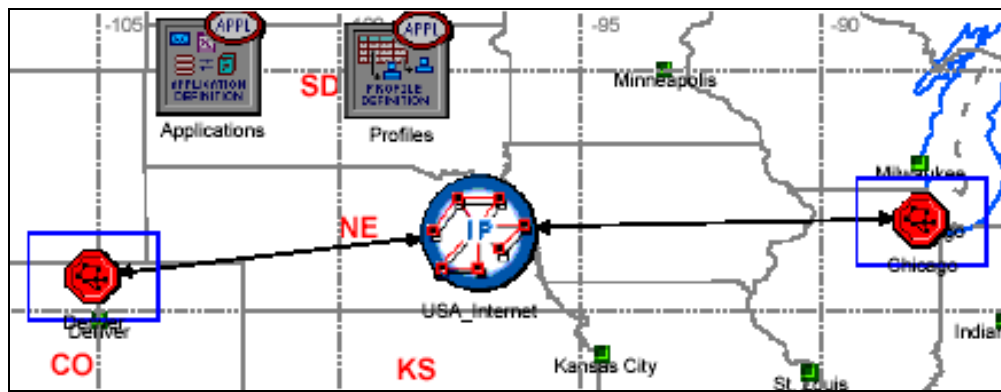


Figure 9: Network Model Used to Study TCP (Lab 4).

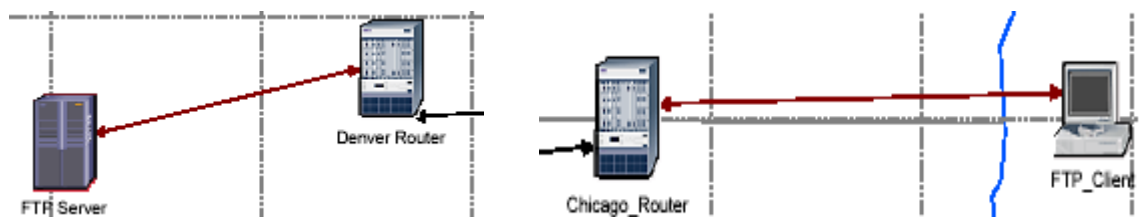


Figure 10: Client and Server Models (Lab 4).

### Students Comments

The students filled out surveys after completing the labs. Most students found the instructions in the modules to be easy to follow. They also appreciated the exposure to an industry standard tool for designing networks. Some of the students suggested that more information be provided on OPNET software, and on why they had to follow certain steps in the labs.

### Concluding Remarks

This paper has described OPNET-based labs that have been developed for a Computer Networks class. The labs have enhanced the educational experience of the students, and have also exposed them to working with networking software in Windows and UNIX environments.

Future work includes the introduction of more labs, and the use of other networking software, for example, open source software[5, 8].

### ACKNOWLEDGEMENTS

The author acknowledges with thanks the assistance of the W. M. Keck Foundation through the provision of funds for the purchase of OPNET software and computer equipment. The author also thanks OPNET for their university program for research and education.

### References

1. A. S. Tanenbaum, Computer Networks, 4<sup>th</sup> Ed., Prentice Hall, Upper Saddle River, New Jersey, 2003.
2. B. A. Forouzan, Data Communications and Networking, 3<sup>rd</sup> Ed., McGraw Hill, New York, 2004.
3. E. Yaprak, L. Anneberg, "The Use of Freeware Network Analyzers in a Networking Laboratory," in Proceedings of

- ASEE Annual Conference, June 2002, Montreal, Canada.
4. S. Zeadally, E. Yaprak, Y. Li, X. Che, "A Survey of Network Performance Tools For Computer Networking Classes", in Proceedings of IASTED 2003 International Conference on Computers and Advanced Technology in Education (CATE 2003), Rhodes, Greece, June 2003, pg.381-386.
  5. J. Matthews, "Hands-on Approach to Teaching Computer Networking Using Packet Traces," in Proceedings of the ACM SIGITE Conference (SIGITE '05), Newark, NJ, October 2005, p. 361-367.
  6. Ethereal, <http://www.ethereal.com>, Viewed June 9, 2006.
  7. OPNET, <http://www.opnet.com>, Viewed June 9, 2006.
  8. E. Yaprak, L. Anneberg, "Utilizing Open Source Tools in the Networking Laboratory," in Proceedings of ASEE Annual Conference, June 2005, Portland, Oregon.
  9. Universities Applying OPNET Software, [http://www.opnet.com/services/university/univ\\_use\\_opnet\\_list.html](http://www.opnet.com/services/university/univ_use_opnet_list.html), Viewed June 12, 2006.
  10. D. Ekong, "Using OPNET Modules in a Computer Networks Class at Mercer University," in Proceedings of ASEE Southeast Section Conference, April 2004, Auburn, Alabama.
  11. Sambari, A., Bhagavathula, R., Thanthry, N. and Pendse, R., "Teaching with OPNET: WSU's Experience", Proceedings of OPNETWORK 2002, Washington, DC, Aug. 2002.
  12. Minton, Carl, Lee, David C., and Midkiff, Scott F., "On the Instructional Use of Opnet at Virginia Tech", Proceedings of OPNETWORK 2000, Washington, DC, Aug. 2000
  13. Cheng, C., Jiang, M., Lo, E., Hardy, R. H. S., and Trajkovic, Lj, "Opnet in an education environment: the Simon Fraser University experience", Proceedings of OPNETWORK '99, Washington, DC, Aug. 1999
  14. Vukadinovic V., and Trajkovic, Lj., "OPNET implementation of the Mobile Application Part protocol", Proceedings of OPNETWORK 2003, Washington, DC, Aug. 2003.
  15. W. M. Keck Foundation, <http://www.wmkeck.org/>, Viewed June 12, 2006.
  16. Teaching with IT Guru, [http://www.opnet.com/services/university/itg\\_panko.html](http://www.opnet.com/services/university/itg_panko.html), Viewed June 13, 2006.

### **Biographical Information**

Dr. Ekong is an Assistant Professor of Computer Engineering at Mercer University. He received his B.Eng. in Electrical Engineering at the University of Port Harcourt, and his M.Sc. and Ph.D. in Electrical Engineering at the University of Saskatchewan. He is also a Registered Engineer in the province of Alberta, Canada. Before coming to Mercer University in 2002, he worked in industry as a Senior Engineer at Ciena Corp, Senior Engineer at Motorola, and a Systems Engineer at Valmet Automation. His teaching and research interests include computer networks, microcontrollers/embedded systems, and digital logic.