2023 Annual Conference & Exposition
Baltimore Convention Center, MD | June 25 - 28, 2023
The Harbor of Engineering Education for 130 Years
Paper ID #38297

Integration of OMNeT++ into a Networking Course in an Electrical Engineering Technology Program

Dr. Murat Kuzlu, Old Dominion University

Murat Kuzlu joined the Engineering Technology Department at Old Dominion University (ODU) as an Assistant Professor in 2018. He received his B.Sc., M.Sc., and Ph.D. degrees in Electronics and Telecommunications Engineering from Kocaeli University, Turkey,

Brian Emmanuel Tamayo
Salih Sarp, Old Dominion University

Salih Sarp is a Ph.D. student in the Electrical and Computer Engineering department at Old Dominion University, USA. Currently, he is developing AI applications and sensor fusion models. Previously, he received his BS degree in Electronics and Communicati

Dr. Otilia Popescu, Old Dominion University

Dr. Otilia Popescu received the Engineering Diploma and M.S. degree from the Polytechnic Institute of Bucharest, Romania, and the PhD degree from Rutgers University, all in Electrical and Computer Engineering. Her research interests are in the general areas of communication systems, control theory, signal processing and engineering education. She is currently an Associate Professor in the Department of Engineering Technology, at Old Dominion University in Norfolk, Virginia, and serves as the Program Director for the Electrical Engineering Technology Program. In the past she has worked for the University of Texas at Dallas, University of Texas at San Antonio, Rutgers University, and Politehnica University of Bucharest. She is a senior member of the IEEE.

Dr. Vukica M. Jovanovic, Old Dominion University

Dr. Vukica Jovanovic is a Chair of Department of Engineering Technology and Associate Professor and Batten Endowed Fellow in Mechanical Engineering Technology Program. She holds a Ph.D. from Purdue University in Mechanical Engineering Technology, focuses on Digital Manufactur

Integration of OMNeT++ into a Networking Course in an Electrical Engineering Technology Program

Abstract

Networking courses are an integral part of electrical engineering technology programs as most electronics in the modern day are required to communicate with each other. They are also getting more attention in manufacturing engineering technology programs because of the development of emerging technologies in Industry 4.0 arena. From laptops, computers, cellphones, modern day vehicles, and smart refrigerators, these devices require a certain level of networking to communicate with other devices, whether it be locally or worldwide. The objective of networking courses in an electrical engineering program is to demonstrate concepts such as local-area networks (LANs), wide-area networks (WANs), network topologies, data transmission, wireless communication protocols, supporting industry-related devices such as radio frequency supported devices and wireless sensors. Within the fields of electrical engineering technology and manufacturing engineering technology, these topics became essential for emerging technologies such as the Internet of Things (IoT), computing systems, and cyber-physical systems. Therefore, these concepts must be integrated into modern curricula, including nontraditional electrical-related courses. Most of the time, course material in networking classes is dispersed through lectures and textbooks since the concepts may be challenging to demonstrate and implement with hands-on activities in the classroom. However, simulation tools may prove beneficial in learning about topics in networking. One of these tools is OMNeT++, a powerful and flexible simulation environment with the ability to display a variety of communication networks. This paper discusses the significance of computer networking courses and modules within electrical and manufacturing engineering technology programs, the advantages of integrating the OMNeT++ simulator within these courses, as well as sample cases of communication network simulations.

Introduction

Computer or communication networks are systems of connections between special devices that allow them to communicate with each other. These types of devices consist of switches, routers, access points, hosts, and more. Each of these devices has a specific purpose. For example, a switch aims to directly connect other devices locally, such as computers, printers, and servers on the same network [1]. Computer networks are designed to accommodate various sizes, depending on the use case. Local-area networks (LANs) connect devices in a limited physical location, such as a home or office network. On the other hand, wide-area networks (WANs) connect multiple LANs and can span over a large geographic area. [1]. The importance of computer networking as a research topic continues to grow as more and more devices communicate with each other. From 1981 to 2008, over 500 million computers were connected to the internet, even reaching the rate of two new computers connecting every second [2]. With the widespread availability of internet access across the globe, the reliance on computer networks for communication has grown significantly [3]. Today, computer networks play an essential role in business, education, government, and other infrastructures [2, 4]. An example of computer networking in companies would be in advertising, where large amounts of data can be collected on the internet based on customers' preferences, affecting business decisions.

In education and schooling, computer networks allow instant access to course material and educational media available on the internet [2]. The governments on state and federal levels also require the use of computer networks as a means of communication. Job opportunities for computer network specialists have also expanded to retail, service, and public services [1]. Given the importance of computer networking for society, it is imperative that the students who will eventually be leading the networking industry be provided with a quality education to have the students thoroughly understand computer networking concepts. To ensure the students' learning, the presentation of course material should also be interesting enough so that students will not get easily lost or bored amidst the vast knowledge and technical depth of computer networking [4-6]. Education of computer networking initially emphasized the specialization of networking equipment, but as time went on, the focus shifted to network design [1]. The design of networks is complex as it contains many technologies that operate simultaneously together [6, 7].

Furthermore, the interaction and relationships between all these technologies are not explained by a single theory [1, 8]. One solution that may help resolve these concerns about computer network education would be the use of computer network simulations [9]. Computer network simulation is the imitation of real-world network communication scenarios using the software. The purpose of this simulation software is to reflect the quality of a given network design through the analysis of the performance of the simulation [7, 10]. Since computer network simulation is a tool that is used regularly in the professional setting of network design and research, it would make sense that it would also be useful for educational purposes [9]. This paper aims to introduce the integration of OMNeT++, a network simulation environment, in networking courses to address the difficulty of teaching networking concepts to electrical engineering technology students.

OMNet++ Network Simulation Platform

OMNeT++ is a framework and library that is essentially used to create and primarily simulate communication networks. The simulations are composed of individual modules written in C++ and then connected to form larger components using Network Description Language (NED), OMNet++'s proprietary high-level language [11-13]. During the simulation, OMNeT++ will perform calculations given the user's parameters and then output the results. These results include throughput, queue times, round trip time, and more. These results represent the performance of the communication network set up by the user. A library that is often used in conjunction with the OMNeT++ framework is the INET library. INET provides a plethora of modules used in communication networks, such as routers, hosts, switches, etc., which can all be connected using wired or wireless connections, including Ethernet, PPP, IEEE 802.11, etc. [14, 15].

Furthermore, the INET library also provides the ability to use different transport layer protocols within the simulation, such as TCP, UDP, and STCP [14]. Given the features listed, one can assume that the INET library is a highly beneficial tool for communication networks in the OMNeT++ simulation environment. OMNeT++ is indeed a powerful tool in the realm of communication network simulation. Some core projects utilized with the OMNeT++ simulation environment include 5G and LTE networks using the Simu5G framework and inter-vehicular communication simulation using the Veins and SUMO framework [16]. Figure 1 displays a sample simulation being run using the Veins and SUMO framework with OMNeT++ visualizing the spread of information in this scenario.

While the applications of the OMNeT++ simulator can seem complex and intimidating to students, the integrated development environment (IDE) is made to be user-friendly and provide some quality-of-life features. One of these features allows the user to drag and drop certain elements onto a field representing the network and to visualize the network's topology simulated [17]. Figure 2 shows an example of a network created by dragging elements made available through the INET framework. A list of connections is also provided to link the components together.

The OMNeT++ simulation software is also suitable to be adopted by educational institutions because it is open source and free to use for non-profit reasons [12, 18]. By being open source, the OMNeT++ software source code is available to all its users. It allows creating and sharing custom libraries for technologies such as 5G and mobile networks for all the OMNeT++ community members [18, 19]. This is beneficial in a learning environment for students in computer networking courses because the concern for the coding aspects of simulation software is lessened, and the focus can be on the essential concepts taught in communication networks. In addition to being open source, the OMNeT++ software is free and available on most operating systems, such as Windows, Linux, and Mac OS, which should be compatible with almost all educational institution computers.

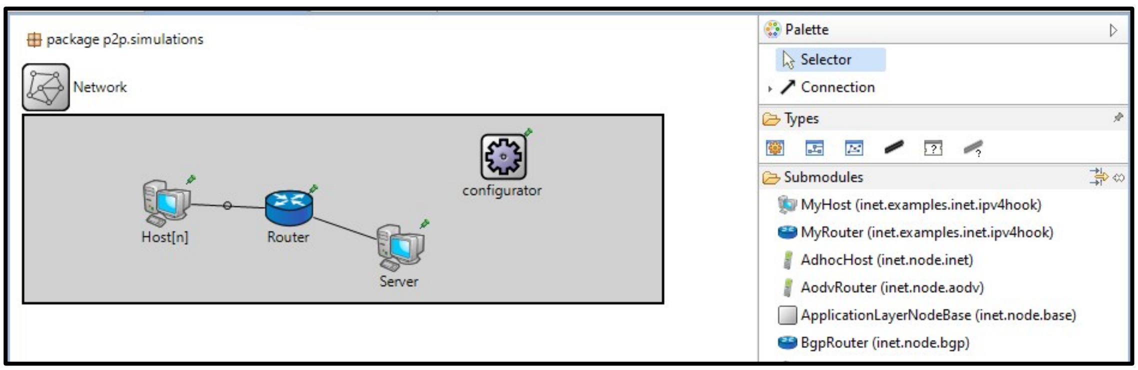

Figure 1: Sample network created by OMNeT++

Methods

This section discusses five sample cases of communication networks simulated using the OMNeT++ simulation environment. The purpose of each case is to represent different network models that can be found in real-life scenarios. Each of the cases will also include a visual representation of the network topology, which is sourced from the OMNeT++ IDE, as well as a sample of graphical data which is provided by OMNeT++ as a result of the simulation.

Case 1 - Two Hosts:

In this first case, two hosts are set to communicate with each other in a simple network. Within the network description file in the OMNeT++ IDE, a network is defined, and the components are placed, as seen in Figure 2 below. A host, in this case, is a device that can send data to other components via a wired connection, for example, a desktop computer. Another device used is an Ethernet switch which allows communication of devices connected locally with a wired connection. The last component used in this first simulation is a *configurator* module which can automatically or manually assign IPv4 addresses and configure routing. During the simulation, a UDP application is sent from Host1 to Host2 in intervals of 1 second for a total of 10 seconds.

The two hosts, named *Host1* and *Host2,* are connected to the Ethernet switch named *etherSwitch* using an Ethernet connection.

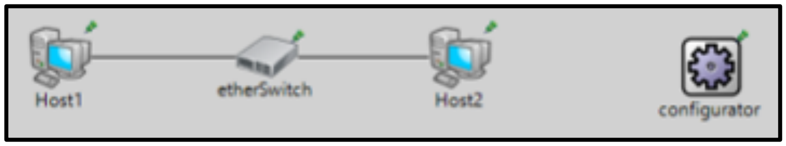

Figure 2: Topology of Case 1, Two Hosts

Case 2 - Server with N number of Hosts:

In the second case, there are *N* number of hosts connected to a router, with one of the hosts being designated as a server. A router is a module that oversees handling packets between computer networks but is also capable of transferring data with a wired connection. In this case, the host designated as the server will receive all the data from the other hosts, similar to how a server can handle the computer resources in its network. In the network description file, all components are connected with Ethernet cables, as shown in Figure 3.

In this simulation, each host named *Host[1], Host[2], ..., Host[N]* will send a UDP application similar to the first case. However, the destination of each application will be the *Server* host. To reach the host, the data first passes through the *Router,* which acts as the medium connecting all the hosts. Assuming there are at least three (3) hosts, packets of sizes 16B, 50B, and 200B will be sent.

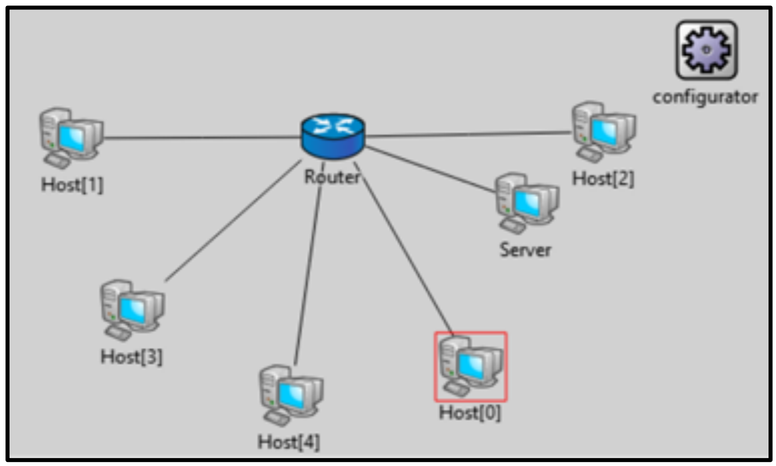

Figure 3: Topology of Case 2, Server with N Hosts

Case 3 - Two Routers with N number of Hosts:

In the third case, shown in Figure 4, there are two *routers*, each with their own group of *N* hosts connected to them, representing individual computer networks or areas. In the network description file, the hosts on the *Router1* network are named *Host[0], Host[1], ... , Host[N]* and are connected to *etherSwitch1* with an Ethernet connection. In the second network on *Router2* the hosts *Host2[0], Host2[1], ..., Host2[N]* are connected to *etherSwitch2* using an Ethernet connection.

The two routers, which represent two computer networks, are then connected with an Ethernet connection. During the simulation, each of the hosts in the *Router1* computer network will send a UDP program to a host in the *Router2* network.

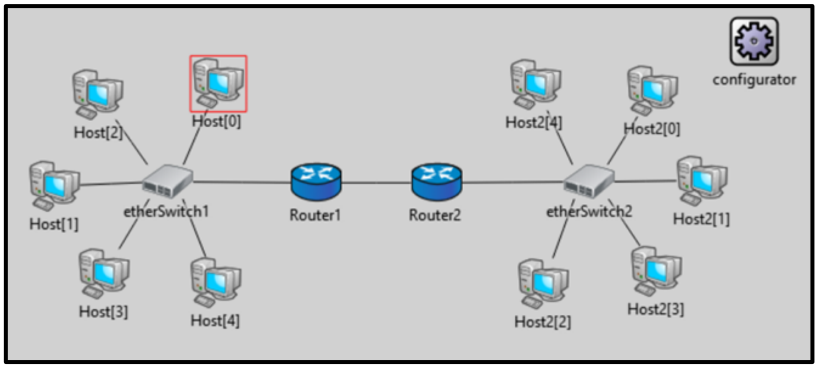

Figure 4: Topology of case 3, Two Routers with N Hosts

Case 4 - Wireless Host:

In the fourth case, shown in Figure 5, there are two wireless hosts, an access point, a wired host, and a configurator. This simulation requires the use of a *radioMedium*, which is a module used for wireless communications to keep track of transmissions going to receivers [20]. Only the wired host is directly connected to the access point using an Ethernet connection.

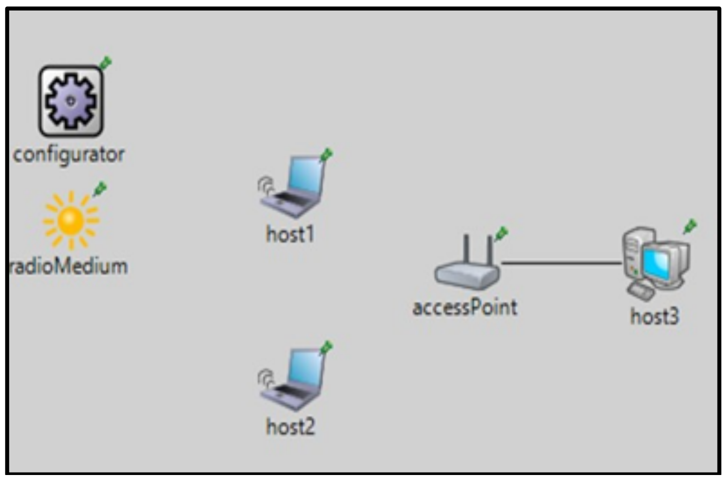

Figure 5: Topology of Case 4, Wireless Host

Case 5 - Three Areas:

In the final case, shown in Figure 6, three routers are connected to each other to represent three network areas. The first area surrounding *router1* consists of wired hosts connected to *etherSwitch1*, as seen in previous cases, and the second area surrounding *router2* and *etherSwitch2*. The third area consists of *router3* connected to an access point and a wireless host.

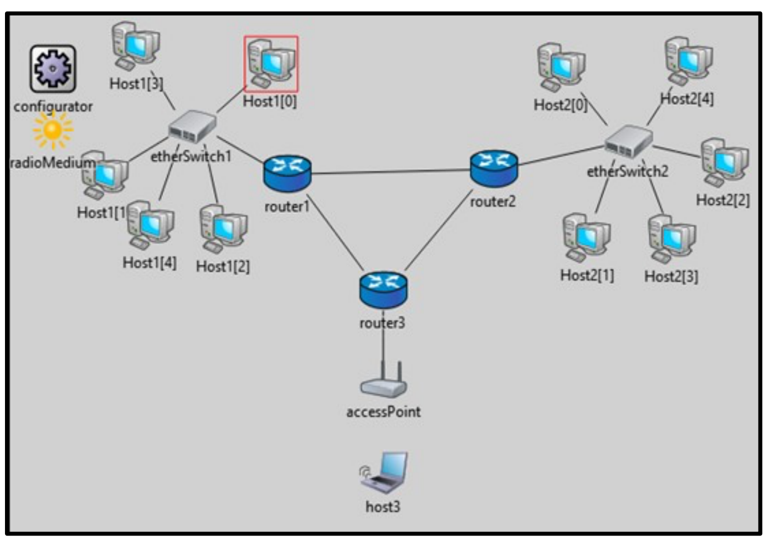

Figure 6: Topology of Case 5, Three Routers

The connections between the wired hosts to Ethernet switches, Ethernet switches to routers, and router to access point are Ethernet connections, while the access point to host3 is wireless. The simulation also calls for the use of a *configurator* and *radioMedium* module. In the simulation, *host3* is set to ping the hosts in the second area, the hosts in the first area are set to send a UDP application to the wireless host in area 3, and the hosts in the second area are set to send a UDP application to the hosts in the first area.

Results

In this section, the results for the five sample case simulations are presented. It includes a description of sample data collected at the end of each simulation, as well as a visual plot or graph to accompany that data.

Case 1 - Two Hosts (Results):

The sample of data collected in Case 1 shows the throughput of data being sent between the two hosts. Throughput is a characteristic of computer or communication networks that describes the amount of data being transmitted over a given period and usually is measured in bits per second [21]. In this case, throughput is seen to increase linearly and peak at around 16 Kbps (Kilobits per second) every second, see Figure 7. Other than these peaks every second, the throughput is measured at 0 Kbps and is represented by a flat line.

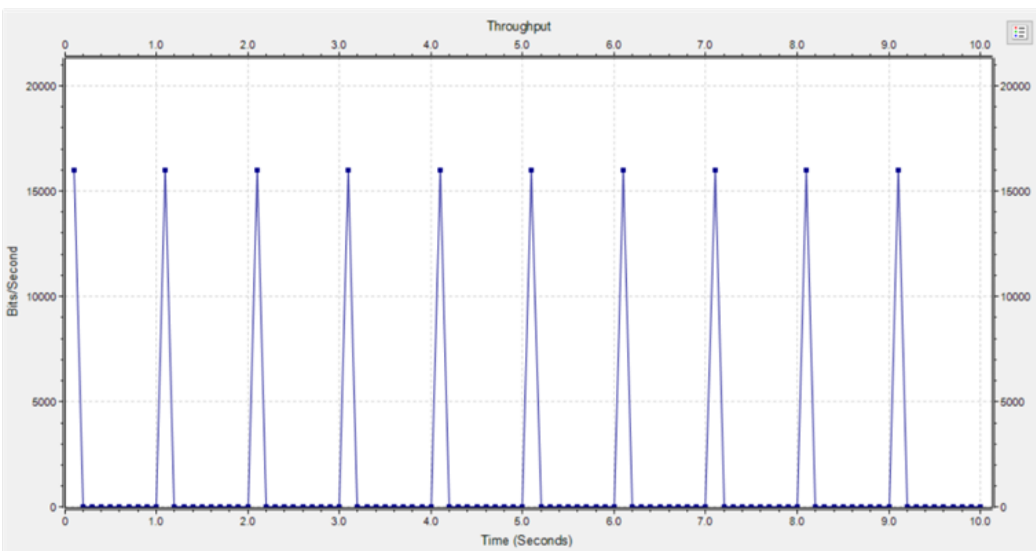

Figure 7: Throughput of Case 1

Case 2 - Server with N number of Hosts (Results):

The sample of data collected shows the throughput being sent between a variable number of hosts with varying packet sizes. The throughput of the 16B packet size is blue, the 50B packet size red, and the 200B packet size green, respectively. On the graph in Figure 8, the throughput of the 200B packet rises to 16 Kbps every second, the 50B packet increases to around 4 Kbps every second, and finally, the 16B packet rises to around 2 Kbps every second. Other than these peaks every second, the throughput for each of the packet sizes is measured at 0 Kbps and is represented by a flat line.

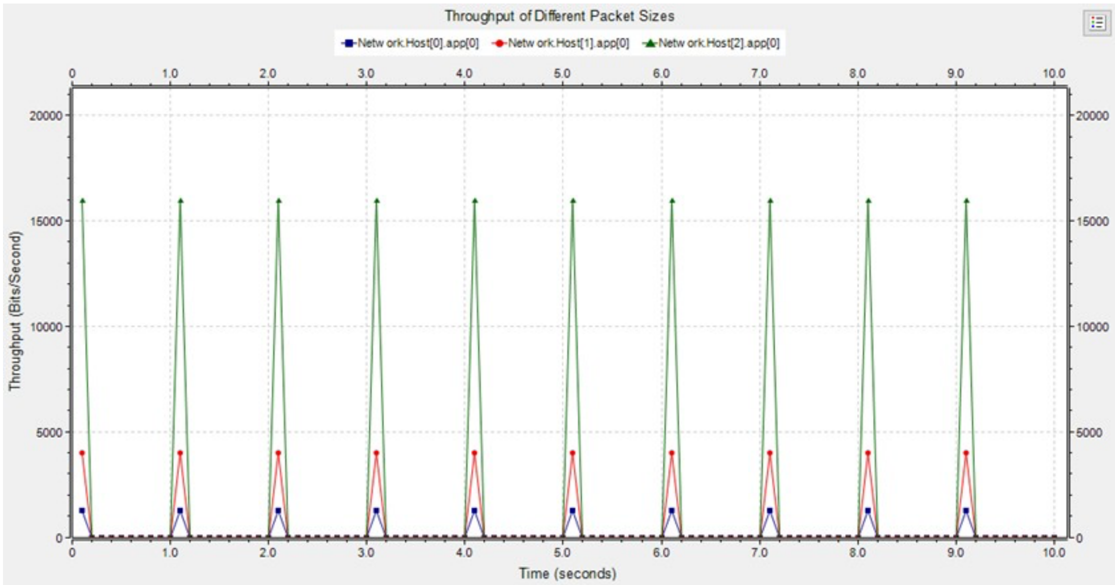

Figure 8: Throughput of different packet sizes in Case 2

Case 3 - Two Routers with N number of Hosts (Results):

In the sample data collected, the bits received and sent are shown for each of the components in the simulation. Unlike the previous cases, this data shows the total amount of data sent during the simulation instead of the rate at which data was transmitted. The bits received are shown in blue, and the bits sent are shown in red; see Figure 9. Each column represents a connection between the designated servers and the connected hosts.

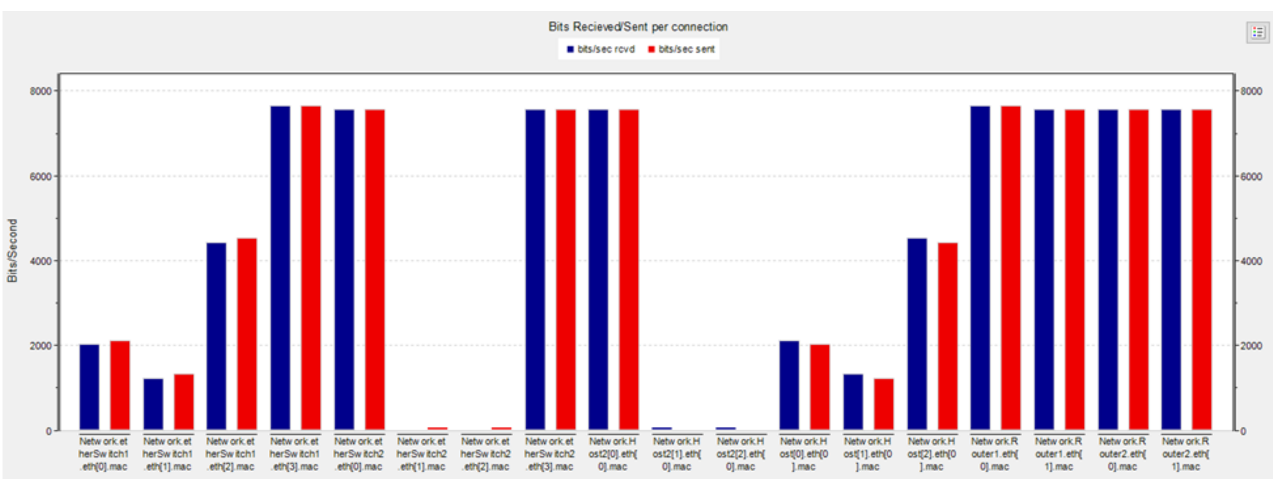

Figure 9: Bits received/sent per connection in Case 3

Case 4 - Wireless Host (Results):

In Figure 10, the data represents the queue times of each of the wireless hosts in seconds versus the simulation time. The blue plot points represent the queue times for *host1,* and the red plot points represent the queue times for *host2*. The queue times for both hosts are seemingly random but range between a minimum of 0.0001 seconds and a maximum of about 0.001 seconds.

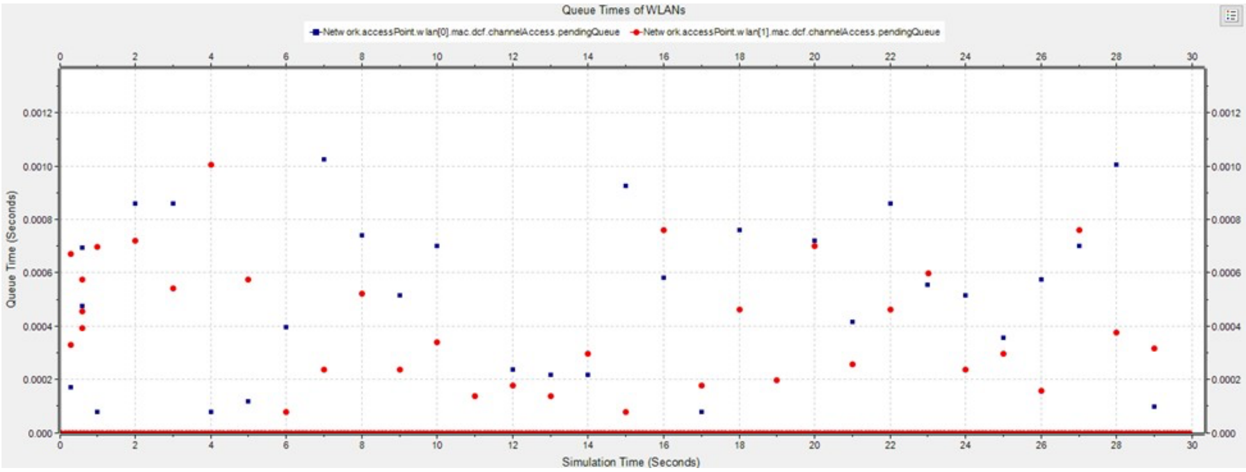

Figure 10: Scatter Plot of Wireless Host queue times in Case 4

Case 5 - Three Areas (Results):

A sample of data collected in this simulation (Figure 11) showcases the round-trip time of each packet sent by the wireless host. Round trip time is the measure of the total time elapsed for a packet being sent to its destination and the acknowledgment that the packet was received at the origin. In the plot, the data points are somewhat randomized, with a minimum of about 0.0023 seconds and a maximum of about 0.0035 seconds, with an average of about 0.003 seconds.

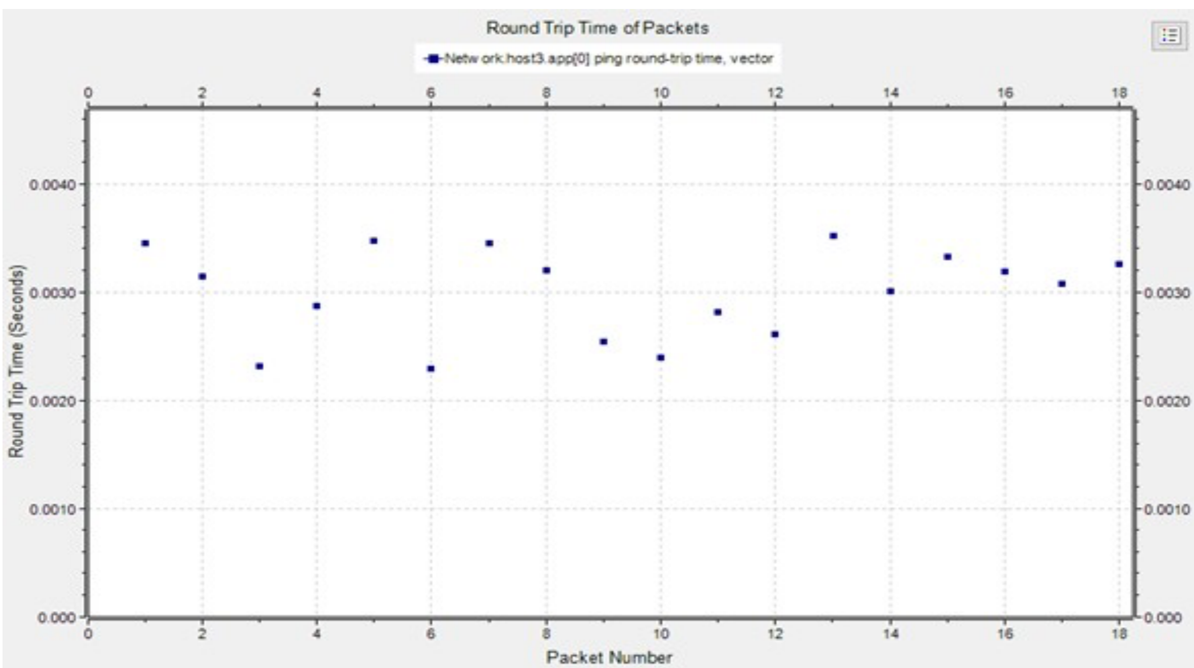

Figure 11: Scatter Plot of round-trip times in Case 5

Discussion

This section further discusses the results showcased in the previous section. In Case 1, the throughput is seen to increase and peak at intervals of every second, which makes sense because the simulation had been set to send an application between the two hosts every second. When an application is not being sent or has finished being sent, there should not be any data being transferred between the hosts, which is observed by the flat line in the throughput graph. In Case 2, the throughput of different packet sizes is shown in different colors. It is observed that the throughput measured is directly proportional to the size of the application being sent, which makes sense because the larger the application or workload, the higher the amount of data needs to be sent over the network. In Case 3, an observation made is that some of the connections may not have received or sent a packet within the time that the simulation was running. The connections that had varying levels of bits received and sent could be due to packet loss. In Case 4, the queue time of each of the wireless hosts is observed to be randomized because of the radio medium being used in the simulation. This makes sense because a wireless connection will be less stable than a wired connection. This observation also extends to Case 5, which shows varying measures of round-trip times due to the use of a wireless medium. Since 2019, the Electrical Engineering Technology Program at ABC University has been offering a data communication and networking course. In this course, students use a commercial network simulator called NetSim for classroom activities, allowing them to model communication networks and gain hands-on experience while utilizing Cisco IOS commands. Although the instructor initially intended to use an open-source network simulator called OMNeT++ as an alternative to NetSim, it appears to be more complex than NetSim. Consequently, the instructor has decided to continue using NetSim for hands-on activities. However, OMNeT++ is recommended for advanced data communication and networking courses and labs.

Conclusion

Computer/communication networks are an integral part of society in daily activities. These networks allow for communication between devices from local areas to countries across the world. Within the context of electrical engineering technology, communication networks enable electronics to communicate with one another to form more sophisticated systems. As the technology for networking developed over time, the dependence on computers and communication networks increased. However, the advancement of network technology also increased the complexity of teaching networking classes. For students to fully understand the topics covered in networking courses, external tools, such as simulation packages, should be used in addition to the information provided by lectures and textbooks. One of these tools is the OMNeT++ simulation environment, which can assist students in learning communication network topics by providing visual representations of the networks being studied, as shown previously in the sample cases above. Its open-source characteristic and free project database make it a promising tool for both students and researchers. The integration of OMNeT++ within networking courses, such as those found in an electrical engineering program, is a step in the right direction for teaching these courses, as the focus on the computer and communication networking industry itself shifts from equipment specialization to network architecture and design.

References

[1] "What is computer networking?" Feb 2022. [Online]. Available: http://www.cisco.com/c/en/us/solutions/enterprise-networks/what-is-computer-networking.html

[2] D. Comer, *Computer Networks and Internets*. Pearson Education, 2009.

[3] B. Wellman, "Computer networks as social networks," *Science*, vol. 293, no. 5537, pp. 2031–2034, 2001. [Online]. Available: https://www.science.org/doi/abs/10.1126/science.1065547

[4] H. K. Lu and P. C. Lin, "Effects of interactivity on students' intention to use simulation-based learning tool in computer networking education," in *2012 14th International Conference on Advanced Communication Technology (ICACT)*, 2012, pp. 573–576.

[5] N. Sarkar and T. Craig, "Teaching wireless communication and networking fundamentals using wi-fi projects," *IEEE Transactions on Education*, vol. 49, no. 1, pp. 98–104, 2006.

[6] I. Dronyuk, O. Fedevych, R. Stolyarchuk, and W. Auzinger, "OMNET++ and Maple software environments for IT Bachelor studies". in *Procedia Computer Science*. Jan 2019 1;155:654-9.

[7] J. Guo, W. Xiang, and S. Wang, "Reinforce networking theory with opnet simulation," *Journal of Information Technology Education: Research*, vol. 6, no. 1, pp. 215–226, 2007.

[8] A.-l. Barabasi, "The architecture of complexity," *IEEE Control Systems Magazine*, vol. 27, no. 4, pp. 33–42, 2007.

[9] G. F. Riley, "Using network simulation in classroom education," in *Proceedings of the 2012 Winter Simulation Conference (WSC)*, 2012, pp. 1–5.

[10] L. Campanile, M. Gribaudo, M. Iacono, F. Marulli, and M. Mastroianni, "Computer network simulation with ns-3: A systematic literature review," *Electronics*, vol. 9, no. 2, 2020. [Online]. Available: https://www.mdpi.com/2079-9292/9/2/272

[11] K. Wehrle, G. Mesut, and J. Gross, *Modeling and tools for network simulation*. Springer, 2016.

[12] Varga, "The omnet++ discrete event simulation system," *Proc. ESM'2001*, vol. 9, 01 2001.

[13] X. Xian, W. Shi, and H. Huang, "Comparison of omnet++ and other simulator for wsn simulation," in *2008 3rd IEEE Conference on Industrial Electronics and Applications*, 2008, pp. 1439–1443.

[14] "What is inet framework?" [Online]. Available: https://inet.omnetpp.org/Introduction

[15] Virdis, G. Stea, and G. Nardini, "Simulte - a modular system-level simulator for lte/lte-a networks based on omnet++," in *2014 4th International Conference on Simulation And Modeling Methodologies, Technologies And Applications (SIMULTECH)*, 2014, pp. 59–70.

[16] G. Nardini, D. Sabella, G. Stea, P. Thakkar, and A. Virdis, "Simu5g–an omnet++ library for end-to-end performance evaluation of 5g networks," *IEEE Access*, vol. 8, pp. 181 176–181 191, 2020.

[17] C. Sommer, "Tutorial." [Online]. Available: https://veins.car2x.org/tutorial/

[18] S. E. Lakhan and K. Jhunjhunwala, "Open-source software in education," *Educause Quarterly*, vol. 31, no. 2, p. 32, 2008.

[19] M. A. Khan and F. UrRehman, "Free and open-source software: Evolution, benefits and characteristics," *International Journal of Emerging Trends & Technology in Computer Science*, vol. 1, no. 3, pp. 1–7, 2012.

[20] "Radiomedium" [Online]. Available: https://doc.omnetpp.org/inet/api- current/neddoc/inet. physicallayer.wireless.common.medium.RadioMedium.html

[21] S. A. Jyothi, A. Singla, P. B. Godfrey, and A. Kolla, "Measuring and understanding throughput of network topologies," in *SC'16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE, 2016, pp. 761-772.