ASEE'S VIRTUAL CONFERENCE
At Home with Engineering Education
JUNE 22 - 26, 2020 #ASEEVC
Paper ID #30994

A Healthcare Case-Study to Teach Simulation Techniques

Dr. Hassan Rajaei, Bowling Green State University

Hassan Rajaei is a professor of computer science at Bowling Green State University, Ohio. His research interests include cloud computing, High Performance Computing (HPC), distributed simulation, parallel and distributed processing, communication networks, wireless communications, and IoT. Rajaei received his Ph.D. from Royal Institute of Technology, KTH, Stockholm, Sweden, and he holds a M.S.E.E. from the University of Utah, and a BS from University of Tehran.

A Healthcare Case-Study to Teach Simulation Techniques

Abstract

Computer simulation is a proven technique in most engineering workplaces. Engineering students often are required to learn and practice modeling and simulation as part of their program studies. Teaching simulation techniques may need examples from the real world to challenge learners and teach the basics of modeling, scenario development, as well as verification, and validation. Real-world examples help students better understand and analyze the simulation results, as well as make their learning process a joyful experience.

This paper describes a simulation case-study for a drop-in healthcare center similar to an emergency room and/or urgent-care provider. The walk-in setup, as opposed to outpatient appointment scheduling, gives multiple challenging potentials to develop what-if scenarios for students to further develop their simulation project even after their targeted lab assignment. The goal is to inspire students' creativity and engage them in their learning experience. By providing user friendly tools that support changing the model, students learn to deal with changing and exploiting scenarios in the case.

We use a simple conceptual model with a few nurses, doctors, and staff to represent the clinic. Simplicity and real-world familiarity of this concept provides students with a jumpstart to take ownership of their simulation study. Soon, students realize the complexity of this case and how they can advance to more sophisticated scenarios. Like a computer game, students become excited to improve their level of knowledge and go beyond a simple laboratory. They develop the data model, implement a base, then improve to intermediate and advanced models. Like a game, several students often go beyond and develop additional scenarios of their own interest.

1. Introduction

Simulation in education is a well-known and an established field. Engineering education, defense training, and medical exercises are a few noticeable examples. As part of the degree requirements, engineering students often learn how to use modeling and simulations for their future workplaces. Whether designing and constructing bridges, buildings, auto vehicles, networks, artificial parts for a human body, or maintaining and improving such objects, experts need detailed knowledge of their subject of study. To acquire the needed skill sets in their education, practical and real-world examples play key roles in student curriculums. This paper describes one such example portraying a drop-in outpatient healthcare center. Students can often relate to this model since either themselves or close relatives have had experienced with similar clinics.

The goal of the simulation is to study performance of a system. Examples include waiting times in queues, average response times, and average throughputs of the systems. Conversely, the goal of teaching is to engage and enhance students in their learning. Multiple observations of this case-study by the author confirmed that the learning outcomes of students are well above the targeted expectations. Furthermore, these observations confirm that simulation tools have less significance than the models used for teaching. Numerous tools can support the case study model that is presented in this paper.

Outpatient care has become more common than overnight hospital stays. Numerous patients complain about waiting time in the hospitals or healthcare centers. Consequently, the need for decreasing the wait time becomes a key factor in a healthcare institute. Our model presents an emergency room or an urgent care where the patients need not to schedule, nor to stay overnight. While it may not show a typical layout, it has the ability to answer the problem at hand. Moreover, the model has two major flexibilities: platform independent and extending the base model to represent more complex system.

The rest of the paper is organized as follows: in Section 2 related work is reported, Section 3 describes the conceptual Drop-in Clinic Models, whereas Section 4 gives some details on implementations platforms and results. Section 5 gives extensions of the base model, and Section 6 gives concluding remarks and future work.

2. Related Work

Xianfei Jin et al [1] suggest that patients' irregular arrival pattern during a day can generate the main cause of long wait times, which also contributes to customer dissatisfaction. To improve outpatient care and reduce wait times for the incoming clients, one needs to determine where the bottleneck is and what causes the delay in the system. There are perhaps certain locations in the system that potentially cause more delay than others and thus block the system from having a seamless entity flow.

Improving wait times in the system is a key subject of this simulation study. In [2], the waiting time was improved by altering the beginning and end time of the doctors such that the doctors times were closer to the arrival of the patients. In [3], the authors used a triage-team method and assigned different urgency levels for patient needs—this method reduced the average time patients spent in the system. The authors in [4] postulated that the reason average patient wait times in a system become high is related to scheduling and late arrivals of the patients. To improve the flow, they suggested using penalties versus rewards. In [5] the author presented a method to reduce the time patients spend in the system by reducing the capacity of the physician's waiting rooms.

In [6] Rajaei and Khakzad describe a real-world scenario in a student cafeteria as an example in education. While their case has more in-depth complexities than for a single lab assignment (e.g. a term-project), a simplified version of the illustrated case could be used for a simple lab assignment. Such assignment, however, needs to have clear objectives of the expected outcomes.

3. The Healthcare Clinic Models

This section describes the details of the models associated with this clinic, including training for queuing networks and the associated input/output data model.

3.1. Conceptual Model

In the base model, as illustrated in Figure 1, there is an administrator who takes care of the arrival and registration of incoming patients. Based on the given symptoms, the patients are forwarded to one of the two nurses to be examined before the doctors to visit them. There are two doctors in the facility: one as generalist and one as a specialist. The examinations of the patients for the latter doctor normally take more time than the former doctor. In addition, both nurses and the doctors can re-route patients to the other doctor based on their diagnosis of the patient. For simplification, the doctors re-route option is not shown in Figure 1. At the end of the visit, the patients are sent to an exit checkout station for release.

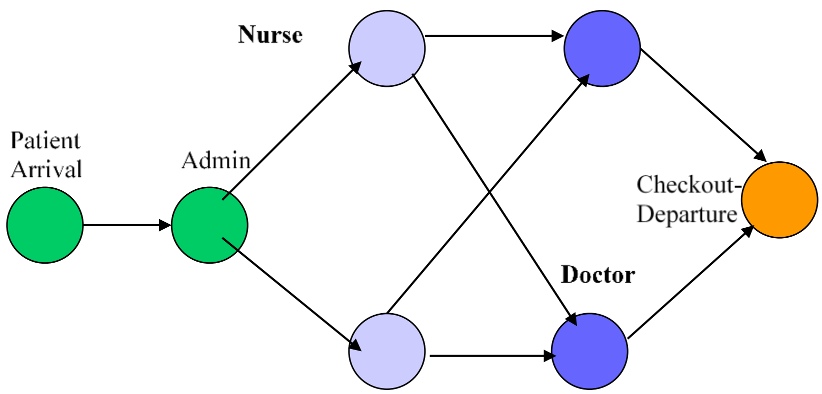

Figure 1: Conceptual model of the Drop-in Healthcare Clinic

The simplicity and the flexibly of the conceptual model is striking for multiple reasons. For the first, it is a queuing network that can be implemented cross platforms, tools, and simulation techniques. Second, each node of the model can be a single server queue or be further developed internally with complex sub-nodes. Third, multiple additional nodes and links and be added to this network to build a fairly sophisticated healthcare model. Finally, the conceptual model can be implemented with multiple simulation techniques such as discrete-event, Mont-Carlo simulation, or agent-based simulation. We used discrete-event simulation and implemented the model with multiple platforms such as ProModel, OMNeT++, and Arena among many others.

3.2. Training Students on the Basic Blocks

Students need training on how to work with the basic blocks of each node as well as how to correctly implement the conceptual model. This subsection describes such details whereas the next section details the data model.

3.2.1. A Single Queue Service Node

To teach students to realize the behavior of each node and thus the queuing network of the healthcare model, a single Serer Queuing system, illustrating a service node such as an ATM, a bank, or post office, was first assigned to students. As illustrated in Figure 2, the customers' arrival in the system is an M/M/1 queueing model. The system has a Markovian arrival, Markovian service, and 1 server. With this exercise, students learn the main characteristics of M/M/1 system, and how to exponentiate arrival, service, and departure. Further, they experience what Little Law explained as the impact of the arrival rate to the rest of the system. Finally, students comprehend why M/M/1 is a basic block of a discrete event simulation. Students are asked to vary the arrival and service rates, as well as the total number of customers in the system in the simulation duration unit (e.g. 1000 to 10000). Students experience changes in the arrival rate, the impact in the queue, and thus the dependency in the system waiting time. In addition, students are also asked to collect data, analyze the results, observe their findings, and submit a written report.

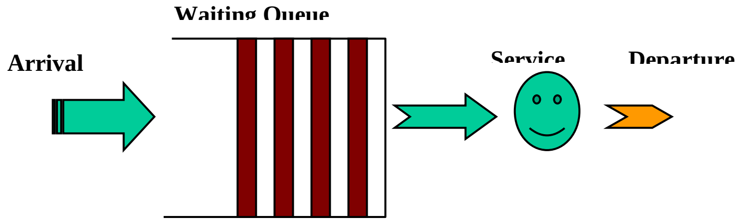

Figure 2: A Single-Server Queuing System to teach students the behavior of the queuing system in the healthcare model

3.2.2. Tandem Queues and Routing Decision

Decision making and understanding the use of random numbers are two additional training techniques students need to learn. The tandem model of Figure 3 is configured as an extension of the single queue described above. In this model, students learn how to split the outcome of a departure in one queue to form the arrival of a second or third queue. Further, students should explain and select which random distribution suits best for their decision making, and how the departure of one node will become the arrival of the tandem nodes.

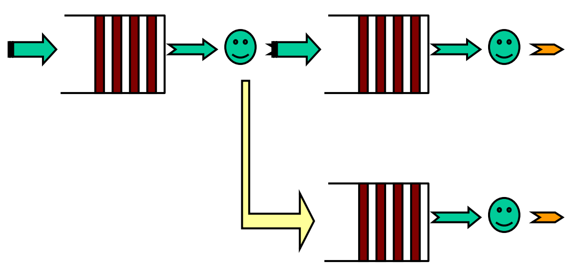

Figure 3: A Tandem Queue model used for training students

3.3. Input and Output Data Model

Data models are vital to any simulation study. There are two significant parts of data that must be worked, understood, and documented: the inputs data and the outputs (or result). Each part needs to be studied, analyzed, and determined. The results need to be documented for the implementation of the model. Data models of the Healthcare Clinic of Figure 1 include:

Input Data:

- Arrival rate of the patients
- Administration service time
- Probability of selecting which doctors visit
- Time at each Nurse station (average time and distribution probability)
- Time at each Doctor station (average time and distribution probability)
- Redirecting Probabilities (both nurses and doctors), and why
- Checkout service time

Output Data (or Objectives)

- Total throughput of the system (per one hour or more simulation time)
- Efficiency of the clinic (needs to be defined)
- Average queuing time of each nodes
- Potential bottleneck of the clinic (by changing e.g. arrival rate and service rates)
- Other collecting data of interest

Clearly, the above task and the rest of the simulation process would overwhelm a single student. As a result, we use groups of 3-4 students for this assignment.

4. Implementation and Results

The provided conceptual model of Figure 1, along with its corresponding related models, are independent of any platform, and thus can be implemented with multiple simulation tools. This section gives examples of several implementation platforms with some details using specific simulation tools.

4.1. Platforms and Tools

Simulation tools and textbooks are plentiful. During the course of multiple years of teaching simulation techniques, the author used different platforms and one at a time, assigned them to students for a specific term. We started with NS2 and found that was not easy to use. Then we used SimPlus [7] and JSimPlus [8], developed and built by the author and his students. SimPlus is the C++ version of Simple simulation engine from Law [9], but with additional functionalities to build a complete and user friendly simulation tool for students to achieve their tasks. JSimPlus is the Java version of SimPlus, with additional functionalities and tutorials, library deposits, and example models. Then we switched gears to commercial tools such as OPNET, ARENA, and ProModel. Recently we changed to popular open-source OMNeT++. Students liked ProModel for its ease of use and good tutorial examples. Perhaps the best of ProModel we can mention here is a good textbook [10], which is easy to read and follow for modelling lab assignments. Nevertheless, students disliked some the hidden method when they needed to change the model (e.g. the logic). Changing model description in OMNeT++ is straightforward by using IDE and C++ codes. In addition, we found a few technical issues with ProModel as well as its licensing issue. As a result, we are now experimenting with OMNeT++, which has a good platform and sufficient building models, in addition to using follow up documents, and books [11, 12].

4.2. ProModel Examples

Over several years, we used ProModel both as the textbook and as a simulation platform. It is convenient to use one for both purposes. Nevertheless, it is difficult to achieve all of the learning outcomes of a simulation course with one tool and its book, even though with good examples available. Adding a simulation example like Figure 1 would help facilitating using different tool and the textbook. This section provides some implementation details using ProModel.

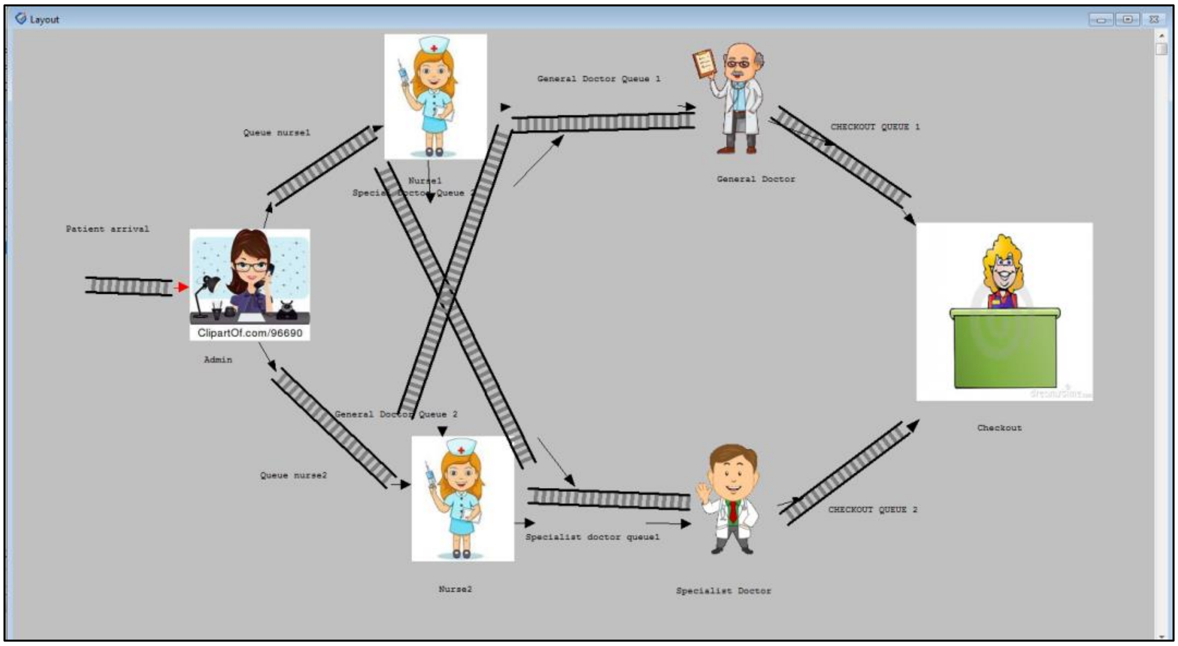

Figure 4: Simulation of the Healthcare Clinic in ProModel

4.2.1. Implementation Model

Figure 4 portraits an implementation example of the healthcare clinic in ProModel. Students are first trained with the single queue of Figure 2 and the tandem queues of Figure 3. That training also provides practice for students on how to deal with changes in their scenarios. As shown in the implementation model of the healthcare clinic, each node either becomes a single queue or contributes to a tandem queue. Consequently, the students' task becomes a natural follow up of their earlier assigned task. The data model described in Section 3.3 is applied to the implementation model both for the input parameters and for the output results.

A good simulation tool with a graphic editor significantly increases students' creativity. Multiple simulation platforms, including ProModel, facilitate such creativity to build and add graphics that students desire to use in their model. Figure 5 illustrates one such example where a group of students created the entities of their simulation assignment as illustrated in Figure 6.

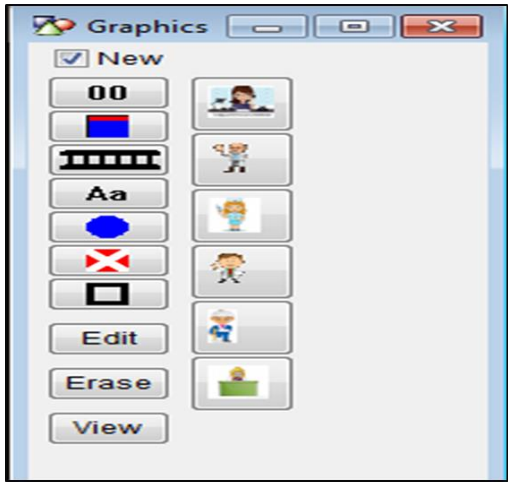

Figure 5: Example of students' creativity using graphic editor

4.2.2. Runtime Animation

Modern simulation platforms often have animation module that helps simulation analysts to develop their implementation and debug potential issues by visual observation. In addition, the animation tool facilitates demonstrating a good presentation of the simulation problem and the obtained results. Figure 6 shows a snapshot of the simulation run, where patients move through diverse parts of the system. For simplicity, only one type of patient is portrayed in the figure.

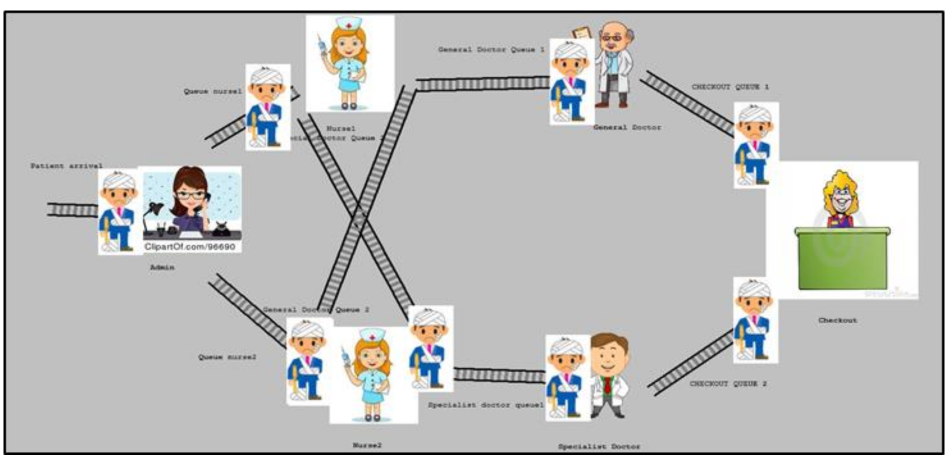

Figure 6: Snapshot of simulation run for 1000 patients in the system

4.2.3. Selected Result examples

Figure 7 demonstrates one example result of the healthcare simulation collected by the same group of students who worked on Figures 4-6. Figure 8 shows the result obtained by another group of students using a different tool for the healthcare model.

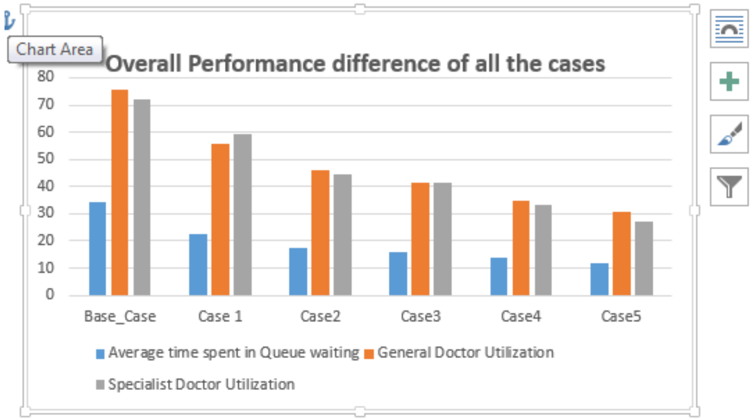

Figure 7: Performance result obtained by a group of students for the healthcare model

Figure 7 shows the average waiting time in the system for the generalist doctor as well as the specialist one. Figure 8 shows the result of the healthcare simulation by another group of students using another simulation tool.

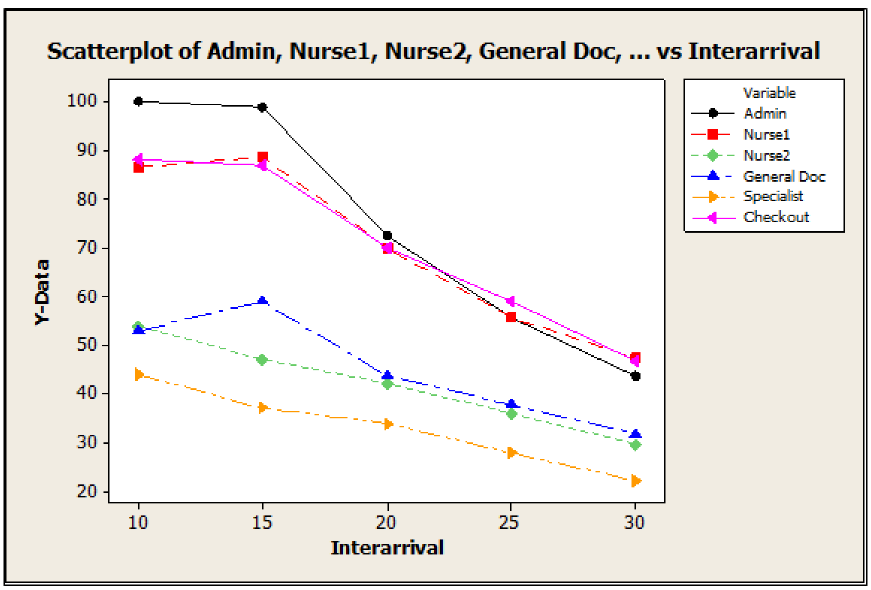

Figure 8: Performance of the Healthcare Clinic by another student group using different simulation platform; impact of the arrival rates on service times are observed.

5. Extensions of the Base Model

The base model of Figure 1 can be modified and extended with several scenarios and functionalities. This section demonstrates two such extensions.

5.1. Adding Scheduled Patients

A potential improvement of any healthcare provider can include adding scheduled patients, rather than walk-in patients, to their system to improve the wait times in the queues. A group of students studied such extension and found that it was fairly straightforward to extend such complexity in the model by adding a different arrival rate to the some entities of Figure 1. One such place would be in the administration node. While other configurations exist, adding a second node to the arrivals makes it much easier to study the behavior of the system. In reality, there is no need to have a second person to do this job, but rather the simulation model needs to separate the two to facilitate implementation.

5.2. Adding Resources and Other Complexities

Most notable extensions come from the need to add resources. We can add nurses, doctors, and different specialists. Additionally, waiting rooms can be changed, reconfigured, or added to the system. Other extensions we studied included adding laboratories and test locations which often prolonged the wait times in the system. Care should be taken for the average service times when added lab-works or other facilities. Finally, certain percentages of emergency room patients often end up having an overnight stay in the hospital. If a complete study is necessary, such extensions can be added to the system.

6. Concluding Remarks

We presented a simple yet powerful simulation model to teach our future engineers. The simulation case study exhibits a drop-in healthcare clinic with staff, nurses, doctors, and patients. Due to the familiarity of a healthcare system, most students immediately recognize themselves in this scenario. Further, the simplicity and usability of the presented model excite students to quickly gain the ownership of their study and make this assignment similar to a memorable gaming case. Another important feature of the provided model refers to the cross-platform independency of the conceptual model presented in this paper.

Bibliography

1. Jin, X, A Sivakumar A., and Lim S. "A Simulation Based Analysis on Reducing Patient Waiting Time for Consultation in Outpatient Eye Clinic." 2013. Web. 12 Dec. 2015.
2. Ho, Emily. "Improving Waiting Time and Operational Clinic Flow in a Tertiary Diabetes Center." Bmj Quality Improvements Reports. Group.bmj.com, 2014. Web. 15 Dec. 2015.
3. Ruohonene, T., Neittaanmaki P., and Teittinen J., "Simulation Model for Improving the Operation of the Emergency Department of Special Health Care." 2006. Web. 15 Dec. 2015.
4. Aeenparast, A., Tabibi S., Shahanaghi K, and Aryanejhad M., "Reducing Outpatient Waiting Time: A Simulation Modeling Approach." Iranian Red Crescent Medical Journal. Kowsar. Web. 15 Dec. 2015.
5. Cote, Murray. "Patient Flow and Resource Utilization in an Outpatient Clinic." Patient Flow and Resource Utilization in an Outpatient Clinic. Department of Health Care Administration, Trinity University, 1 June 1998. Web. 15 Dec. 2015.

6. Rajaei H, Khakzad R, 2017, “A Real-World Example for Student Learning: BTSU Cafeteria Simulation”, in ASEE 2017 Annual Conference Proceedings, June 25-28, Columbus, OH
7. Rajaei, Harper, Mancine, Scott, 2004, "SimPlus: An Experimental Simulation Tool", in Proceedings of the 2004 American Society for Engineering Education Annual Conference (ASEE'04), June 2004, Salt Lake City, Utah. Session 2420.
8. Rajaei, H. Eid E., Kannungo, D., Ringerberg, J., 2011, "JSimPlus: A Tool for Teaching Simulation Techniques", in the 14th Communications and Networking Simulation Symposium, CNS'11, sponsored by ACM/SCS, April 4-11, Boston.
9. Law, A. 2007, “Simulation Modeling & Analysis”, 4th Ed, McGraw Hill
10. Harrell C, Ghosh B, and Bowden R, 2012 “Simulation Using ProModel”, 3rd Ed, McGraw Hill
11. Chamberlain T, 2013, “Learning OMNeT++, Packt Publisher
12. Virdis A, 2019 “Recent Advances in Network Simulation, the OMNeT Environment and its Ecosystem” Springer