Teaching service modeling to a mixed class of engineering and information system students: a hybrid approach

Jeremiah D. Deng, Martin K. Purvis
Department of Information Science / Telecommunications Programme
University of Otago, PO Box 56, Dunedin, New Zealand
jeremiah.deng@otago.ac.nz, mpurvis@infoscience.otago.ac.nz

Abstract

Service modeling has become an important area in today's telecommunications and information systems practice. The teaching of the subject is not easy as it involves mathematical content difficult for especially students of the information systems background. We have developed a hybrid approach using a number of learning strategies to accommodate the diversity of students. This delivery approach seems to work well in bringing out comparable performance from different student groups. Teaching evaluation also reveals the effectiveness of our teaching approach.

1 Introduction

As an answer to the fast rising prominence of data communications and networking technologies in the IT industry, the Telecommunications programme at the University of Otago was established in 2001. The programme was initiated by the faculty members of the Information Science discipline, also incorporating foundational papers such as mathematics, physics, and electronics fundamentals, along with computer engineering and programming subjects. It has a few core papers such as Data Networks, Wireless Communications, Network Administration, and 4-th year subjects such as Digital Signal Processing, Multimedia Networking, and Mobile Computing. Teaching staff take on a broad range of expertise across the university: Computer Science, Information Science, Physics, and Electronics. Within the course students are allowed to study for a second major or a minor in other disciplines. The Programme has been also active in conducting research especially in key areas such as embedded systems, sensor networks, and mobile computing. Postgraduate student supervision has been fruitful.

Like other telecommunications and similar computer engineering programmes around the world, the teaching of various communication and networking technologies demands our focus. The fast paces of technological advances in various fronts make this a non-trivial task as technologies get invented, start to evolve, become old, and then phase out. All these imply that both for telcos and other enterprises there is a wide range of technologies to choose for adoption, to understand, optimize, and plan out against migration and sustainability. These activities are also under the constraints of non-technical aspects of business operation: finance, legislation, and policies etc.

For this reason from 2002 we have started offering a Computer Network Design paper for third year students that includes a systems methodology for enterprise communication network planning, analysis and management. In a sense it functions as a capstone paper of the Programme that provides the playground for Telecommunications students to exercise their understanding of existing technologies, including local area networks (LAN), wide area networks (WAN), virtual private networks (VPN), wireless technologies, and the Internet, and to use their features and synergy to construct a solution for enterprise networking as a service plan.

On the other hand, we noticed there seems to be a lack of sufficient attention paid to the service modeling aspect of telecommunication and computer engineering. Much pedagogic effort has been focused on teaching the technologies or system integration. The notion of effective modeling of the communication systems so that the best service performances can be achieved, however, comes much later into the pedagogic agenda. Mandelbaum and Zeltyn [10] recently argued that there is a gap between today's academic supply and the demand for service engineering, but for building successful systems of the future we must combine technological knowledge with process design.

Such a demand is also relevant to the Information Systems domain, albeit its being neglected in the curriculum design. In 2005 we amended the prerequisite and started to offer the paper to students majored in "Information Science" (whose background is similar to the more traditional "Information Systems"). As the new group of students are also familiar with networking technologies, there is almost no need to amend the course context. The real challenge, however, is on dealing with the diversity of a combined class. Differences lie in their mathematical background and skills, and their understanding about the business operations within enterprises. Throughout the years we have developed a hybrid delivery approach that seems to accommodate well the students' diversity, allow them to develop sound knowledge about service modeling and be able to apply this knowledge to solve real-world network design problems.

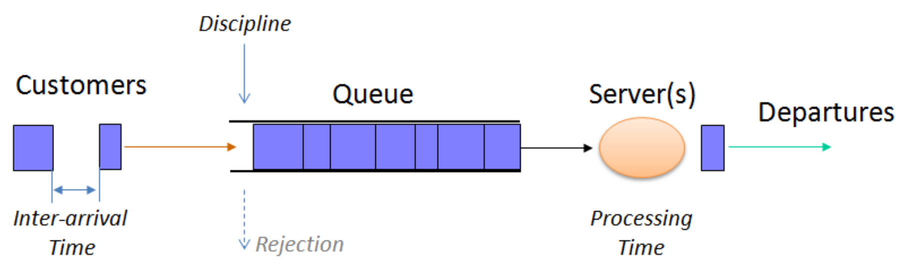


Fig. 1. Diagram of a simple queueing system.

In the remainder of the paper, we will first present a quick introduction to a few key concepts of service modeling, particularly the queueing theory. Our intention is, based on this introduction in Section 2 our further discussion will make more sense. This is followed by course information in Section 3 and our delivery approach in Section 4. To evaluate how effective the delivery approach is, student assessment results and teaching evaluation results are presented in Section 5. Finally we conclude our paper along with some discussion on future work.

2 Service modeling: key concepts

The core element in service modeling is a mathematical tool called "queueing theory" [10]. A queueing system consists several key components:

- Customer arrivals. Customers arrive at a service facility under a point process. The inter-arrival time is modeled by a probability distribution.
- A queue. Under a discipline (e.g., First-in-first-out, or priority queueing), customers enter a queue and wait for their turns when the server is busy. If the queue size is limited, further arrivals when queue is full will be "rejected" or dropped by the system. If rejection is relevant, then there is also a dropping discipline involved, i.e., deciding which customer to be dropped - at the end, start, or a random position in the queue.
- Service. Depending on the job size, the server spends some time serving a customer before fetching an enqueued customer or wait for a new customer to arrive. Service time is also modeled with a probability distribution. The served customer then departs from the system.

The queueing process and its relevant components and factors can be demonstrated by Figure 1.

Queueing theory is a powerful tool for service modeling and engineering. In fact, queueing is everywhere: in banks, supermarkets, when getting on a bus, and checking in at the airport etc. In telecommunications, "customers" can be data packets waiting to be transmitted over a transmission line, phone calls to be established through the 3G radio, or HTTP requests to a web server. For a simple queueing scenario as shown in Figure 1, several factors will affect the efficiency and reliability of the system, including:

- Interarrival time. Usually time between arrivals always change for a queueing system. Whether the arrival pattern is smooth or bursty will affect the length of customer queue.
- Queueing discipline: whether it is FIFO or priority queueing will affect the time that a customer spends in the system. There are other disciplines utilized in telecommunications systems such as fair queueing and weighted fair queueing (WFQ) which is an important part of Quality of Service (QoS) implementations.
- Service time. This is determined not only by the processing speed of the server, but also by the size of the job the customer brings in. For instance, larger data packets tend to cost more service time to a server or a transmission link.

To characterize the behaviour of queueing systems usually Kendall's notation, in the form of $A/B/c/K/m/Z$, is used. Here A stands for the interarrival time distribution, B for service time distribution. The most frequently used notations for the first two items are M (exponential distribution), D (deterministic, i.e., constant), and G (for general distributions). The third item c is number of servers, while K is maximum number of allowed customers in the system, m indicates the size of the customer population, and Z is the queueing discipline which is typically FIFO.

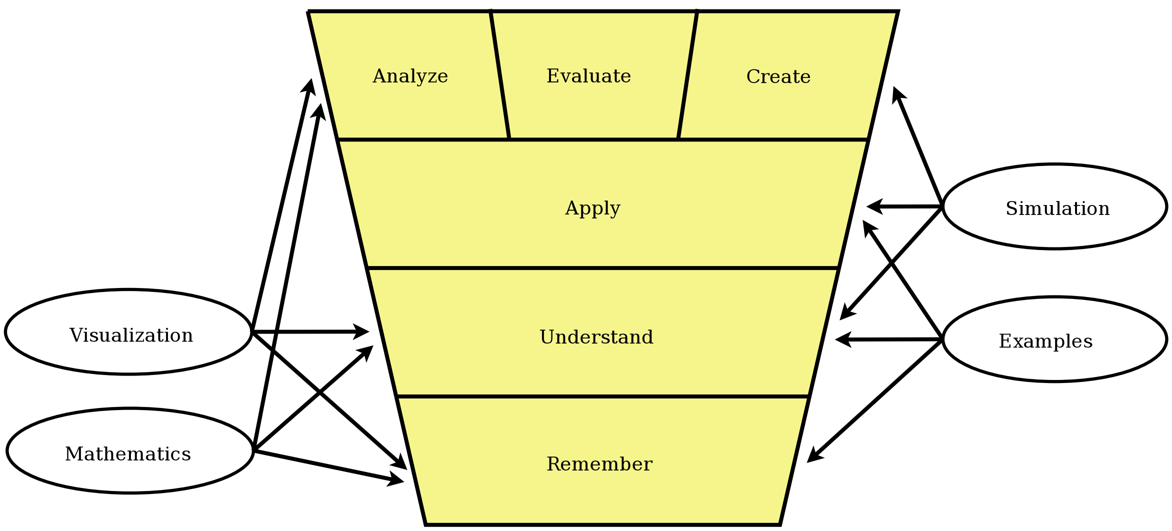


Fig. 2. The revised Bloom's Taxonomy [1] assisted with learning strategies.

For instance, an $M/M/1$ system has an exponentially distributed interarrival time, an exponentially distributed processing time, and one server; whereas an $G/D/1/K$ system has a general distribution for interarrival time, a deterministic processing time, one server, and the system only allows up to K customers.

For a single server queueing system to stay stable, naturally the average arrival rate (i.e., the reciprocal of average interarrival time, usually denoted as λ) should not exceed the average service rate (i.e., the reciprocal of average processing time, denoted as μ).

To assess the performance of a queueing system, there are several performance indicators, e.g., number of customers in the system, and average time spent in system (by a customer). For more details of queueing theory and its application in computer networks, one can refer to the classic work written by Bertsekas and Gallager [2].

3 Course Information

3.1 Syllabus

The prerequisite is a Network Management paper where students get familiar with networking technologies and configuration. The students background difference is resulted from earlier difference in their curricula. The Telecommunications (noted as "TELE") students have studied at least one first-year mathematics paper (algebra and calculus), while the Information Systems ("INFO") students normally do not take these papers. On the other hand, their previous training in programming is about the same.

The workload is similar to other papers in both TELE and INFO curricula. This includes two 1-hour lectures and a 2-hour tutorial or lab session each week. The paper lasts for 13 weeks of the entire semester.

3.2 Resources

Because of the specially designed course content we cannot find a single textbook that suits our purposes. Selected sections are taken from several books and these are combined with other online readings as the core course material.

We use a course website as the central portal that contains or links to all teaching materials (including lecture notes, readings, tutorials, laboratory books, and assignments). It also links to a Web blog created on Tumblr on which we encourage students to post out interesting links they have found that relate to service modeling and networking in general). Choice of blog instead of the Blackboard system is due to the latter's dull interface and lack of use as observed in the past.

To allow the students to complete queueing performance of complex systems such as $M/M/c$ and priority queues (whose solution formulae are not for a faint-hearted third-year student), we also give out a specially designed spreadsheet so that the students can bypass the annoyance of a complicated calculation process and get assignment problems solved.

To pair with theoretical queueing analysis, we also adopt the simulation approach. Simulation packages, either of general purpose such as OMNeT++ [5], or for special purpose such as the Network Simulator (NS-2) [9]) have been used in the course. After a few trials we have settled down on NS-2 since it provides easy implementation of abstract queueing model simulations (e.g., by using UDP traffic to simulate $M/M/1$) as well as network simulations with realistic traffic modeling.

4 Teaching and evaluation methods

4.1 General approach

Work on teaching queueing theory: inquiry-based [4], or using computer algebra systems (CAS) [7].

We take a problem-solving approach in the delivery of our paper, different from the data-oriented approach in [10]. Our reasons are two-fold: first, there are many real-world scenarios in telecommunication and information system engineering that require us to use appropriate modeling for performance analysis and optimization; secondly, this allows the students to get engaged early on and initialize a problem-solving process that includes data gathering. From a teaching point of view, this is also convenient/effective.

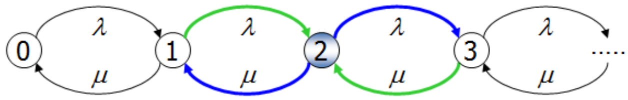
Our delivery is designed based on a revised Bloom's Taxonomy [1] as shown in Figure 2. Bubbles around the taxonomy indicate a number of learning strategies we have employed to impart the knowledge of service modeling to the class of diversity.

4.2 Learning strategies

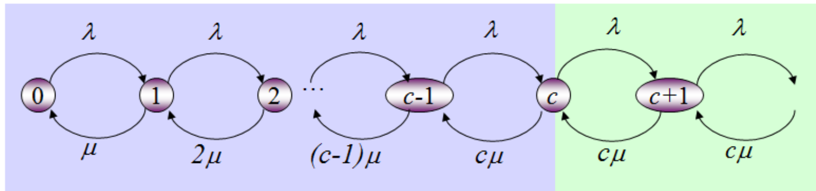
4.2.1 Learning by Mathematics

The queueing theory is the core of the service modeling and it is inherently mathematical. Queueing theory is largely missing in Information Systems curricula. There is a long-time (mis-)perception that teaching mathematics to IS students would be a total disaster. The lack of necessary mathematical content in the curriculum perhaps in return has encouraged the students' maths-phobia.

Our argument is even INFO students are teachable in mathematics. They might be less informed and less skilled, their understanding about the problem solving ideas may not be bad once they overcome the initial fear of notations and formulae. In fact, Thomas et al. [16] looked at the influence on university entry scores and engineering students performance and concluded that the correlations between the two either do not exist or are too weak to base any educational inventions on. Our intention is to equip INFO students with the understanding and knowledge about queueing models so that they can use the skills to



(a) $M/M/1$



(b) $M/M/c$

Fig. 3. Markov chain diagrams used to explain the queueing models of (a) $M/M/1$, and (b) $M/M/c$.

optimize the performance of an information system or a computer network. This can be achieved using some proper guidance, encouragement, and some good tools.

To understand and be able to apply queueing models, one of course needs to capture the main concepts of queueing systems, and be able to use the Kendall notation (which does require a little bit memorization). We also need to inform the students about the properties of various probability distributions so that they can understand why different traffic profiles may affect the network performance. All these are not possible without an adequate level of exposure to mathematics.

With a combined class of IS and Engineering students, we have to orchestrate an engaging delivery process. This means rather than teaching on notations and formulae per se, we need to take advantage of other strategies .

4.2.2 Learning by seeing

A picture is worth a thousand words. Visualization is employed through out the teaching of queueing theory via diagrams and animations. Sometimes simple diagrams such as Figure 1 can become instrumental in developing conceptual understanding.

Figure 3 gives another example. The difference of these two diagrams, especially the details of death rate between system states (links going left on the diagrams) are related to the number of servers in the systems. The green and blue links in Figure 3(a) are inflows and outflows for state “2”, and the sum of the rates for inflows and outflows on any state of a stable queueing system should be equal (on which we bring in the analogy of a sink with water flow in and out), and we then show that by simple mathematical derivation one can work out the solutions for such a queueing system.

The colouring of the diagram in Figure 3(b) indicates that there are two patterns of the states’ departure rates, and this visual clue is passed onto students who can then align the derivation process of the solution with these two different regions so that better understanding is achieved.

Apart from these illustrations, other visualizations are also used in class, including PowerPoint animations and animated computer simulations. Simulation labs also help students visualize the queueing systems and observe their behaviours.

4.2.3 Learning by examples

To make students able to better comprehend the abstract queueing models, we further make constant references to real-world examples, such as queues in banks or supermarkets, call-centre services etc., relating to students’ experience. When talking about the self-similarity in the Internet traffic, we use fractal images as an analogy in an effort to demonstrate the idea of fractals and impress them with those beautiful patterns.

Another group of examples come from the computer networking domain, where various networking technologies are brought under the light of the queueing modeling, so that the students can see how they can use their conceptual understanding to solve real-world problems inherent in these technologies.

Example queueing problems are given and attempted in lectures and tutorials, sometimes we even ask student volunteers to come afore to the blackboard and show his/her solutions to the class. The invitation normally will succeed and students get involved and are also encouraged.

By referring to everyday examples and networking implementations, students are also challenged to balance the queueing theory with real-world factors beyond pure engineering considerations. For instance, students are first shown for pure efficiency a multi-server single queue model is always preferred, but they are then challenged by the question why a different multi-server multi-queue model remains frequently used in supermarkets or in web service deployment. These examples serve to clarify their understanding about the pros and cons of different models, and they are easy to remember.

4.2.4 Learning by doing

While the previous strategies help to establish sound understanding about queueing models and the performance analysis skills, they are much related to the lower two layers in the taxonomy, we use practicals for the students to gain knowledge of higher levels. In particular, when we ask students to conduct queueing analysis to derive the performance of system in question, additional tasks of conducting simulations are also assigned to them. The motivation is two-fold. First by coding the simulation they will gain more understanding about the overall system process and its key parameters; Secondly, by comparing the theoretical results obtained by queueing theory formulae with performance indicators obtained by the simulation, they will have true appreciation about the stochastic nature of the system, as well as understand the impact of not only the average scenario, but also the worst scenario which is often the culprit of telecommunication service outage.

Computer simulations have been used in teaching queueing theory [12], [6] and TCP/IP networking [14]. Packages such as NS-2 and OMNeT++ have been used frequently used in computer networking research [8], [17]. By learning how to write simulation, and the same importantly, analyze simulation outcome, students are equipped with research interests and skills for a potential postgraduate study.

4.3 Assessments

Assessment plays an important role in student learning. In our paper, it is crucial for developing the student learning on the upper layers in the Bloom's Taxonomy, including application, analysis, evaluation, and creation.

The assessment package consists of two parts: internal and external, each taking up 50% of the full marks. For internal assessment three assignments are used, all designed around real-world scenarios:

- Assignment 1 is on performance optimization of a networking system using queueing analysis and simulations. Over the years two scenarios have been alternately used: multi-core computer systems for network intrusion detection (SNORT [15]); and a web server serving to production queries and managerial queries (therefore with different service requirements to be met).
- Assignment 2 is topology design task that aims at connecting branch offices of a nation-wide company in New Zealand. It allows the students to exercise their simulation skills to imitate real-world network traffic manipulation and challenges them on the redundancy considerations subject to the long narrow geographical layout of the two islands.
- Assignment 3 is a final project that focus on high-level system design that incorporates scenarios dealt with in the previous two scenarios, plus more services requirements and constraints (such as policies and financial limits).

The final exam consists of basically two parts: short-answer questions for testing on students' understanding of network design methodologies and modeling issues, such as components of queueing systems and their Kendall annotations; long-answer questions are designed for testing on problem-solving. For instance, given an enterprise network and a key application, the students are asked to choose a queueing model for performance analysis and suggest measures to achieve the best service quality under potential constraints.

As the focus has been on conceptual understanding and application rather than on memorization, we list all the queueing model solution formulae on a "cheat sheet" which is provided in addition to the exam questions in the final exam. Calculators are also allowed.

TABLE 1
Pair-wise T-test results on difference of average marks between TELE and INFO students.

Assessments	p-value
Queueing	0.57
Project	0.018
Exam	0.58

The long-answer part also has a question that asks the students to reflect on their Assignment 3, summarizing their design process they went through and problem-solving approaches they took.

5 Results and Interpretations

5.1 Assessment result comparison

To evaluate the effectiveness of our delivery approach, especially how well it addresses class diversity, first we compare the performance of two groups of students across three assessments: the queueing assignment ('Queueing'), the final design project ('Project'), and the final exam. As explained the first two have opposite focuses: Queueing is a quantitative task that assesses students' understanding about queueing theory, and Project is rather on high-level critical thinking and qualitative by nature. We would like to see whether the diversity of students affects their performance in these assessments. To ensure whether the difference in average performance between the two groups is statistically significant or not, independent t-tests are carried out and the results are given in Table 1.

The boxplot of the students' performance in Queueing is given in Figure 4. Marks are shown as percentage on the y-axis. As expected the INFO group gives a bigger range and there is a low-end outlier of around 10 marks. The ranges given by the first and third quartiles for INFO and TELE are however quite similar. The INFO group even reports a slightly higher average mark than TELE, which is a little surprising as normally we have expected the latter, with better mathematics background, should have achieved more. The difference is however statistically *not* insignificant, as indicated by Table 1.

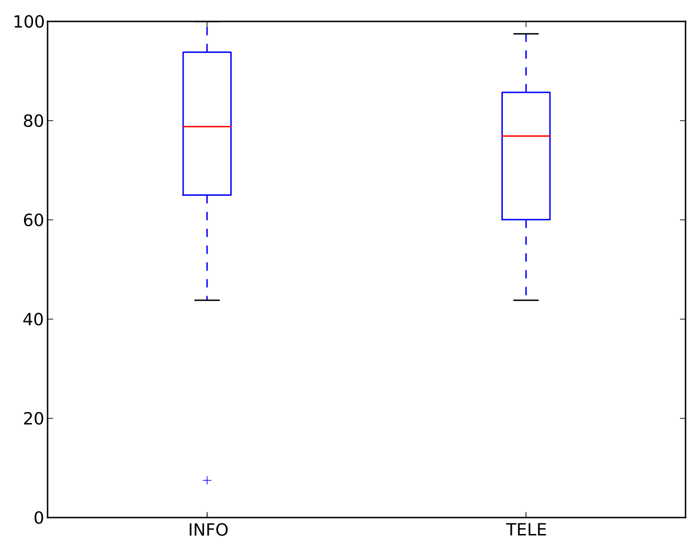


Fig. 4. Assessment result comparison: Queueing.

Interestingly, the result seems to suggest that some diversity even exists within the INFO group. While the outlier on the low-end drags the INFO average down, there have been a couple of highly motivated

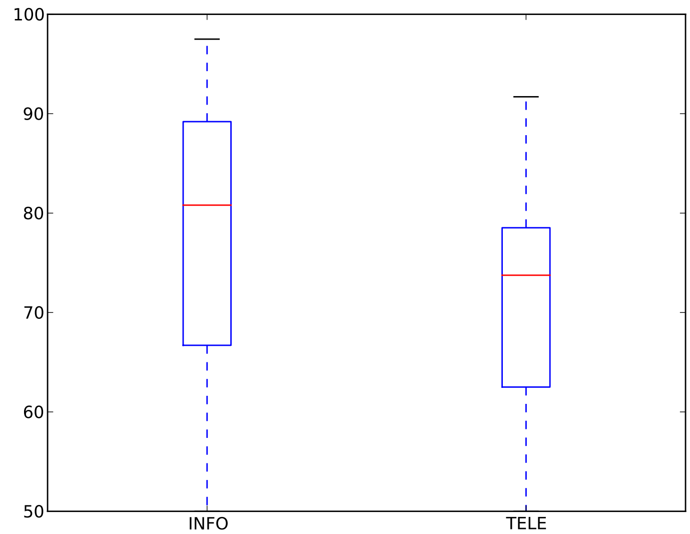


Fig. 5. Assessment result comparison: Project.

TABLE 2
Teaching evaluation results.

Questions	Feedback		
	Positive	Neutral	Negative
How valuable is the paper?	86	14	0
Are you encouraged to think?	100	0	0
Do you spend more effort doing this paper?	57	14	29
Are the assessments helpful?	85	15	0
Overall how effective is the Paper taught?	93	7	0

INFO students who achieved full marks for the Queueing task by successfully thinking out all possible options and assessing the relevant system performance.

Figure 5 gives the boxplot of students performance in the Project. INFO group has a higher average mark than the TELE group. The result correlates well with our expectations as we think INFO students are better trained in high-level system evaluation with a big picture in mind, but TELE students are normally more oriented towards finding solutions to problems of smaller scale (not necessarily of less difficulty) and tend to ignore factors such as budgets and management concerns. The difference of average marks between the two groups is statistically significant.

However, looking on the performance in the final exam as shown in Figure 6, both groups again display similar patterns and there is no statistical difference between the average marks of the two groups. We believe this reveals that the learning outcome of the two groups has been similar despite their diversity.

5.2 Teaching evaluation results

Course evaluations have been conducted twice and some of the agglomerated results are shown in Table 2. As evaluation questionnaires are returned anonymously, we are not able to tell students' identities. Without looking at the two groups, we can only assess the evaluation by the whole class. Overall as we can see, all students agreed to that they are challenged to think. The effort they spent in the paper seems to correlate with the class diversity, about 29% of them spending less effort (compared with other papers), while 59% spending more. On the other hand, very positive feedback is obtained on the value of the paper, the helpfulness of assessments, and the overall effectiveness of the paper delivery.

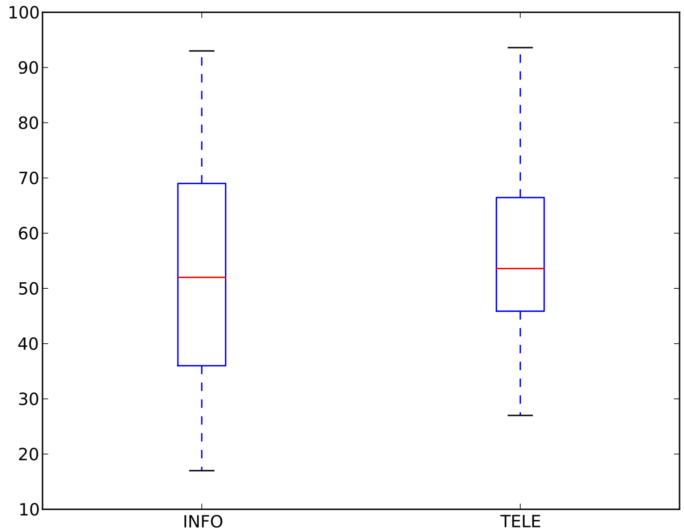


Fig. 6. Assessment result comparison: Exam.

5.3 Other observations

As explained there are a number of details when characterizing a queueing model. Despite our frequent emphasizing on potential pitfalls in model selection, students fall into them now and then. For instance, only about 20% of students managed to identify a model in a final exam question correctly as $M/D/1$; the others all fell for $M/M/1$.

Most students lack confidence with their numerical skills and this seems to be a continuing trend. One anecdote was that when it came to calculate the job arrival rate (λ) in "one job arrives every five minutes", the student started his web browser and went to the WolframAlpha website to calculate "1/5". This almost made the tutor speechless for the rest of the tutorial.

Another issue that is under consideration is the use of computer algebra software (CAS). There has been research conducted on using CAS, e.g., Maple, in teaching engineering mathematics and positive outcome has been achieved [13]. There is also concerns about CAS, without proper uses designed, operate only as some magic blackboxes hence without the capability of enhancing students learning [11]. So far we have not used any CAS. First, there will be extra overhead introduced by using CAS: we need additional lecture and lab time to spend on teaching the students how to use CAS. Secondly, even though CAS can be used to help students derive queueing solutions, it is not the focus of our teaching goals. Rather, the students are expected to understand the basic mathematical ideas behind various mathematical models (and hence their differences subtle or significant), match real-world networking scenarios with some specific queueing models, and then use tools (formula, spreadsheets etc.) to calculate the service specifications.

On the other hand, we also realize that while the current delivery and assessment packages manage to enable students to do problem-solving (i.e., analyze and evaluate) and produce network designs (to "create" to a certain extent), their capabilities in using mathematics to model and solve a new model are not fostered. Maybe in the future we can offer a few bonus questions and provide CAS tools for students of more mathematics and programming prowess. This would encourage them to develop research interest in relevant directions and get prepared for a postgraduate study.

Another interesting idea that is worthy consideration is a "by projects" approach [3], which might help more self-motivated students identify the gaps in their mathematical skill set and become eager to learn more in an effort to bridge those gaps.

6 CONCLUSION

Teaching mathematical content to students of computing background is always challenging. In this paper we presented a case study on our effort in delivering a service modeling paper to students of Telecommunications and Information Systems backgrounds. A number of learning strategies are developed to assist students learning and these seem to have worked to accommodate the class diversity. Statistical analysis on students performance and teaching evaluation highlights suggest that the delivery has been satisfactory.

In the future we will work on further improving our teaching approach. Items on the agenda include reassessing the possible use of CAS and integration of smaller projects that allow more capable students (from both backgrounds) to explore more topics of interests.

REFERENCES

[1] L. W. Anderson and D. R. Krathwohl. *Taxonomy for Learning, Teaching and Assessing: A Revision of Bloom's Taxonomy of Educaiton Objectives*. Longman, New York, 2001.

[2] D. Bertsekas and R. Gallager. *Data networks (2nd ed.)*. Prentice-Hall, Upper Saddle River, NJ, USA, 1992.

[3] G. Bischof, E. Bratschitsch, A. Casey, and D. Rubeša. Facilitating engineering mathematics education by multidisciplinary projects. In *Proc. 2007 ASEE Annual Conf.*, Washington, DC, 2007. American Society for Engineering Education. http://papers.asee.org/conferences/paper-view.cfm?id=4238.

[4] Y.-Y. Chan. Teaching queueing theory with an inquiry-based learning approach: A case for applying webquest in a course in simulation and statistical analysis. In *Proc. 37th ASEE/IEEE Frontiers in Education Conference*, pages F3C–1 – F3C–6, Oct. 2007.

[5] O. Community. OMNeT++ network simulation framework, 2009. http://www.omnetpp.org/.

[6] G. Dobson and R. Shumsky. Web-based simulations for teaching queueing, Little's law, and inventory management. *INFORMS Transactions on Education*, 7(1), 2006.

[7] S. Fitzgerald and J. Place. Teaching elementary queueing theory with a computer algebra system. *SIGCSE Bulletin*, 27(1):350–354, Mar. 1995.

[8] J. Heidemann, N. Bulusu, J. Elson, C. Intanagonwiwat, K. chan Lan, Y. Xu, W. Ye, D. Estrin, and R. Govindan. Effects of detail in wireless network simulation. In *Proceedings of the SCS Multiconference on Distributed Simulation*, pages 3–11, Phoenix, Arizona, USA, January 2001. USC/Information Sciences Institute, Society for Computer Simulation.

[9] ISI. The network simulator - ns-2, 2011. http://www.isi.edu/nsnam/ns/.

[10] A. Mandelbaum and S. Zeltyn. Service engineering: Data-based course development and teaching. *INFORMS Transactions on Education*, 11(1):3–19, 2010.

[11] W. Neuper. What teachers can request from CAS-designers. In *Proceedings of the 5th International Conference on Technology in Mathematics Teaching*, Klagenfurt, Austria, 2001.

[12] J. H. Reed. Computer simulation: A tool to teach queuing theory. In *Experiential Learning Enters the Eighties*, volume 7, 1980.

[13] C. Rielly. The application of computer algebra software in the teaching of engineering mathematics, 2005.

[14] N. Sarkar. Teaching computer networking fundamentals using practical laboratory exercises. *Education, IEEE Transactions on*, 49(2):285 –291, may 2006.

[15] Snort. Snort home page, 2010. http://www.snort.org.

[16] G. Thomas, A. Henderson, and G. T. The influence of university entry scores on student performance in engineering mechanics. *Australasian Journal of Engineering Education*, 17(1), 2011.

[17] A. Varga and R. Hornig. An overview of the OMNeT++ simulation environment. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, SIMUtools'08, pages 60:1–60:10, ICST, Brussels, Belgium, Belgium, 2008. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).