

Simulation Tool for Queuing Models: QSIM

Pratiksha Saxena, Lokesh Sharma

Gautam Buddha University, Department of Mathematics, School of Applied Sciences,

Gautam Buddha University, Greater Noida, India 201308.

mathematics.pratiksha@gmail.com

ABSTRACT

This paper proposes simulation software QSIM (Queuing Simulation) to simulate queuing models. QSIM is a tool which simulates and compares the results of different queuing models. This tool is programmed in JAVA and is based on analytical approach to guide optimization strategy. In Qsim, Simulation is carried out by providing the inputs for arrival rate, service rate and number of servers and using these values the performance measurement of a particular model is stored in the database for further comparisons and study. Simulation results are stored in excel file and it also provides graphical results to compare the outcome.

General Terms

Simulation and Modeling.

Indexing terms

Simulation, JAVA, Queuing models, Performance Measures, calling population.

Academic Discipline And Sub-Disciplines

Modeling, Queuing Models and Simulation

SUBJECT CLASSIFICATION

Mathematics ; Queuing System

INTRODUCTION

Queuing modeling is very important for optimization of length queue, customer time and for best use of resources. Simulation, however, does take into account the randomness and interdependence which characterize the behavior of real-life business environment. Using simulation, randomness can be included through properly identified probability distributions taken directly from study data. However, as the complexity of the analysis increases, so does the need to employ computer-based tools. While spreadsheets can perform many calculations to determine the operational status of simple systems but it uses averages to represent schedules, activity times, and resource availability. This does not accurately reflect the randomness and interdependence present in reality with resources and other system elements. In 1960, at the initial stage of the development of packet-switched telecommunication networks, queuing theory was applied to models of such networks (Kleinrock L. 1964). A number of queuing models have been studied for commercial applications (Iliadis, I. and Denzel W.E. 1993). QNs are generated as powerful modeling notation and can be applied to many different domains, including computer networks, supply chain analysis, software systems, street traffic and others (Serazzi G. 2008). An overview of large-deviations theory and its applications to telecommunication systems is provided by Alan Weiss (Weiss A. 1995). Simulation of queues for a hospital is optimized by using simulation (Hang J. 1998). The use of queuing analysis and simulation of various hospital departments, such as inpatient, ICU (Green, L. V. 2003) and (Kaplan, E.H. et al 1981), obstetrics units and ED has been widely discussed in the literature (Kim, S. et al 1999). The application of basic queuing principles and models to the hospital inpatient admitting process has been studied by Green (Green, L. V. 2003). Kaplan, Sprung and Shmueli used the queuing modeling to analyze the impact of various admissions policies to ICU facilities (Kaplan, E.H. et al 1981). Law and Kelton proposed an algorithm of a successful computer simulation study (Law, A.M. and Kelton, W.D. 2000). A number of researchers (Banks and Carson, 1987; Mahachek, 1992; Vissers, 1998; Isken et al., 1999; Eldabi and Paul, 2001; Harper, 2002; Morrison and Bird, 2003; and others) have addressed the core principles for performing a discrete-event simulation study of a healthcare system (Banks, J. and Carson, J.S. 1987). A QuickPass model is proposed to improve bank queuing system based on queuing theory (Wang Y. et al 2000). A large scale simulation model for estimating the system behavior for capacity planning before real establishment or execution is proposed (Wang X. et al 2008). A model for bank counter is proposed with queuing models (Yang M. 2009).

This paper proposes a simulation tool to calculate and analyze the performance measurement for five queuing models. It discusses single server, multi server, first-in-first-out, service in random order, poisson arrivals, and exponential service time queuing model. This is a JAVA based simulation tool which calculates output measurements on providing the inputs and stores the result to the database for further analysis. QSIM also compares the output for different models and provides optimized result. It also facilitates comparison of single and multi server results. In the first phase algorithm and flowchart is described for QSIM. Second phase represents working of QSIM and third phase shows comparison of results for a study data.

MATERIAL AND METHODS

QSIM supported operating systems are Windows XP (32-bit), Vista (32- or 64-bit), or Windows 7 (32- or 64-bit). It supports 64-bit distributions capable of running 32-bit applications. QSIM is developed in Netbeans (IDE-Integrated development environment) version 6.8. This simulation tool is designed in Java (J2SE) JDK 1.7 and MS Access is used as database. Fundamental SQL queries are used in the program.

QSIM includes five queuing models; $M/M/1:\infty/FCFS$, $M/M/1:\infty/SIRO$, $M/M/1:N/FCFS$, $M/M/S:\infty/FCFS$, $M/M/S:N/FCFS$. Algorithms for these models are described as:

$M/M/1:\infty/FCFS$ and $M/M/1:\infty/SIRO$

Step1: Design a frame having defined space to show outputs of the models. Say framework().

Step2: Get the values of μ and λ . Convert to Double if required. Use Double.parseDouble().

Step3: Calculate value of ρ , P_o , L_s , L_q , W_s , W_q using queuing model mathematical formulas.

Step4: Call the method to connect the framework to database.

Step5: check if Selected item in drop down is :- $M/M/1:\infty/FCFS$ (model 1)

Insert calculated values of model into model1 table of database.

Step6: else if Selected item in drop down is :- $M/M/1:\infty/SIRO$ (model 2)

Insert calculated values of model into model2 table of database.

Step7: Show results on framework and End.

$M/M/1:N/FCFS$

Step1: Call the frame designed in the model 1/2. Say framework().

Step2: Get the values of μ , λ and N . Convert μ and λ to Double and N to Integer if required.

Step3: Calculate value of ρ , P_o , L_s , L_q , W_s , W_q using queuing model mathematical formulas.

Use Math.pow() methods to implement power functions.

Step4: Call the method to connect the framework to database.

Step5: Check if Selected item in drop down is :- $M/M/1:N/FCFS$ (model 3)

Insert calculated values of model into model3 table of database.

Step6: Show results on framework and End.

$M/M/S:\infty/FCFS$

Step1: Call the frame designed in the model 1/2. Say framework().

Step2: Get the values of μ , λ and S . Convert μ and λ to Double and S to Integer if required.

Step3: Calculate value of ρ , P_o , L_s , L_q , W_s , W_q using queuing model mathematical formulas.

Use Math.pow() methods to implement power functions.

Step4: Call the method to connect the framework to database.

Step5: Check if Selected item in drop down is :- $M/M/S:\infty/FCFS$ (model 4)

Insert calculated values of model into model4 table of database.

Step6: Show results on framework and End.

$M/M/S:N/FCFS$

Step1: Call the frame designed in the model 1/2. Say framework().

Step2: Get the values of μ , λ , N and S . Convert μ and λ to Double and N and S to Integer if required.

Step3: Calculate value of ρ , P_o , L_s , L_q , W_s , W_q using queuing model mathematical formulas.

Use Math.pow() methods to implement power functions and factorial using inbuilt method.

Step4: Call the method to connect the frame to database.

Step5: Check if Selected item in drop down is :- M/M/S : N/FCFS (model 5)

Insert calculated values of model into model5 table of database.

Step6: Show results on framework and End.

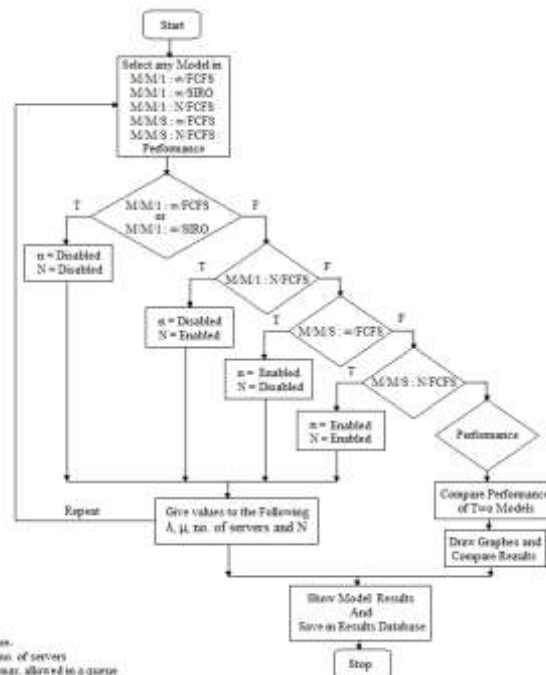


Fig. 1 Flowchart for QSIM

For connecting to database:

Step 1: call class.forName method addressing the JdbcOdbcDriver.

Step 2: initiate an object of DriverManager addressing database name, user name and password.

Step 3: put all these statements in try catch and handle the exceptions.

Step 4: End.

Flowchart shown in fig. 1 represents simulation steps of QSIM.

DISCUSSION

QSim calculates the output measurements and stores these variables in excess file for analysis of result. Output of QSIM stored in various variables as Average server utilization(ρ), Average number of customers waiting(L_q), Average number in system(L_s), Average waiting time(W_q), Average time in the system(W_s), Probability of zero customers in system(P_0), Probability of exactly n customers in system(P_n).

The evaluation of systems performance consists of the following steps:

- Start the toolbox. By double clicking the executable file.
- Select any of the model given in the Drop down list.
- Enter the value of no. of servers, length of queue, arrival rate and service rate in the text boxes.
- Click on the simulate button of the toolbox. (model results are saved in the database)
- End
- A new window will open containing text boxes for each model so now provide values in the text boxes for both models.
- Click on the compare button of the toolbox. (graph will be drawn between both models and model results are saved in the database)
- End.

Main Interface window has a drop down with six choices. Five choices represent five different queuing models and one choice is for performance comparison. Depending on the model, it enables and disables the text boxes. Clicking on the SIMULATION, Text input boxes take input variable values and these values reflects to respective text boxes. Fig. 2 shows main interface of QSIM.

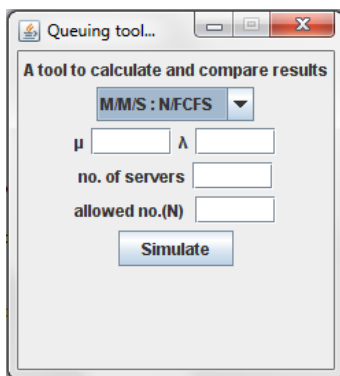


Fig. 2 Main Interface

After that, it calculates result according to selected model in drop down box. Fig. 3 represents drop down box from main interface of QSIM.

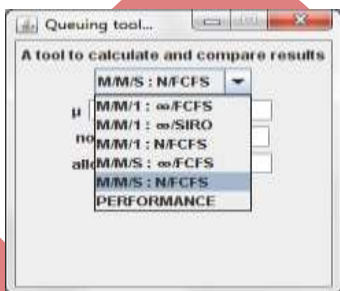


Fig. 3 Choice of model in interface

Results appear in the new adjacent window and also stored in database simultaneously. It calculates six values of output measurements for different queuing models say ρ , ρ_0 , L_s , L_q , W_q , W_s . Fig. 5 represents working of QSIM and substitution of input values.

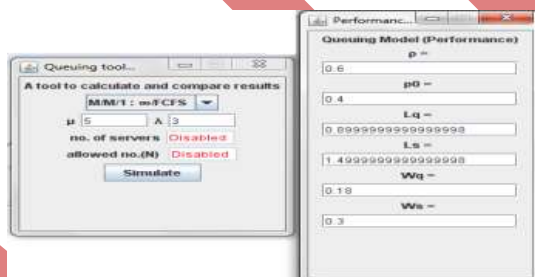


Fig. 4 Input values for model

Now, If we select the PERFORMANCE in drop down list, a new window appears which has two drop down lists to compare two queuing models. With the help of this feature, performance of two queuing models and be compared by selecting models from drop downs. These two models are compared on the basis of μ , λ , s and N . Fig. 5 represents calculation interface of QSIM.

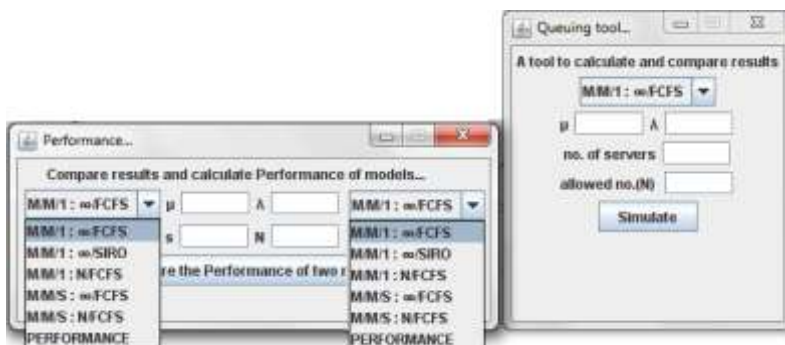


Fig. 5 Calculation interface

On substitution of input variables μ , λ , s and N for two models, it provides comparison of results for two models on the basis of their ρ , ρ_0 , L_s , L_q , W_q and W_s . Then it also draws a comparison graph between these two models and stores the

comparison results in back end database MS- Access. This feature helps to optimize the number of servers for the study. Fig. 6 represents comparison of results and graph representation of QSIM.

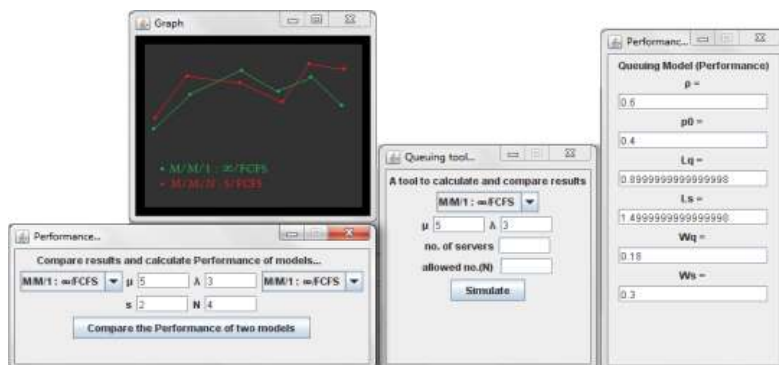


Fig. 6 Comparison of results

Every calculated result (μ , λ , Ls, Lq, Ws, Wq) and comparison result of two models are stored in back-end database (MS Access). These results can be retrieved for further calculation and comparison.

3.1 Numerical Experiment

Result calculation and analysis is explained with the help of a numerical experiment. Data is provided for input parameters λ, μ, S and N.

There are six different tables in the database, five for each model and one to store comparison results. Once we calculate the results these results are stored for future purpose and so these can be retrieved any time for future calculation.

Calculations for QSIM are shown in Fig. 7.

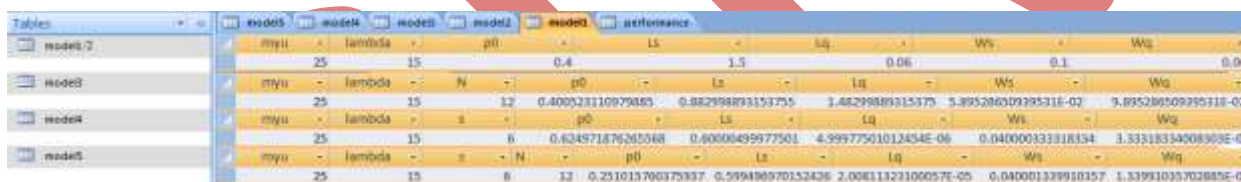


Fig. 7 Result storage interface

Table 1 represents compilation of input and output values for all the models of QSIM. For both models I and II results are same. In model III when number of allowed length in queue is fixed say 12, length of queue is decreases with some fraction although waiting time is increasing. For same input in model IV when number of servers is fixed to 6, length of queue in system falls down drastically from 1.4 to 0.6 and waiting time in system falls from 0.058 to 0.040. For model V, with similar input values all the output variables are at the lower end as compared to results of IV say lesser length of queue, waiting time.

These results can be analyzed as;

- Length of queue is at minimum value in case of model III.
- Length of queue in system is least in case of model V.
- Waiting time in queue is least in case of model I.
- Waiting time in system is least in case of model IV and V.

Table 1 input values and output

Model	Input				Output					
	λ	μ	N	S	ρ	Po	Lq	Ls	Wq	Ws
I or II	15	25	-	-	0.60	0.400	0.899	1.499	0.059	0.099
III	15	25	12	-	0.60	0.400	0.882	1.482	0.098	0.058
IV	15	25	-	6	0.10	0.624	4.999	0.600	3.333	0.040
V	15	25	12	6	0.10	0.251	2.008	0.599	1.339	0.040

This numerical experiment represents simulation done by QSIM.

CONCLUSION

The Multi-objective QSIM simulation Model of the Queuing System is presented in this paper. QSIM can calculate the six performance output measurements for five queuing models. The numerical experiments shows that the queuing system with QSIM is much efficient than the queuing system without it. Results of QSIM are stored in Excess file and retrieve the data set of various for further analysis of result set. After comparing the results, the best queuing model among the given models for provided values can be chosen. One of the important features of this simulation tool is to compare the output measurements for various numbers of servers in the system. This simulation tool can be applied to various queuing systems to optimize the results and to evaluate the optimum number of servers in a system. The further work can be focused on the queuing system with several QSIM and the design of the QSIM management information system. The further work on this toolbox can be done by implementing and comparing all available models in queuing system which will provide a choice of best suitable model for the given conditions.

REFERENCES

- [1] Banks, J., Carson, J.S. 1987 Applying the Simulation Process. Proceedings of the 1987 Winter Simulation Conference, Georgia, USA, 68-71.
- [2] Green, L. V. 2003 How many hospital beds? *Inquiry*, 39, 400-412.
- [3] Hang J. 1998 Simulation for a Hospital Clinic Queue System. *Systems Engineering-Theory and Practice*, 3, 140-144.
- [4] Iliadis, I., Denzel W.E. 1993 Analysis of packet switches with input and output queuing. *IEEE Transactions on Communications*, 41:5, 731 – 740.
- [5] Kaplan, E.H., Sprung, C.L., Shmueli, A., and Schneider D. 1981 A Methodology for the Analysis of Comparability of Services and Financial Impact of Closure of Obstetrics Services. *Medical Care*, 19, 395-409.
- [6] Kim, S., Horowitz, I., Young, K.K., and Buckley, T.A. 1999 Analysis of Capacity Management of the Intensive Care Unit in a Hospital. *European Journal of Operational Research*, 115, 36-46.
- [7] Kleinrock L. 1964 *Communication nets: stochastic message flow and delay*, McGraw-Hill, New York.
- [8] Law, A.M., Kelton, W.D. 2000 *Simulation Modeling and Analysis*, 3rd Ed., McGraw-Hill.
- [9] Serazzi G. 2008 *Performance Evaluation Modelling with JMT: learning by examples*. Technical Report.
- [10] Wang X., Zhihui D., Li S., Chen Y. 2008 Modeling and Simulation of Virtualized Autonomic Service Centers. *Simulation*, 84, 119-136.
- [11] Wang Y., Qian C., Cao J. 2000 Optimized M/M/c model and simulation for bank queuing system, *Software Engineering and Service Sciences, IEEE International Conference*, 474 – 477.
- [12] Weiss A. 1995 An introduction to large deviations for communication networks. *IEEE Journal on Selected Areas in Communications*, 13:6, 938-952.
- [13] Yang M., Yi K. 2009 Optimization of bank counters based on the queuing process. *Journal of Zhejiang University*, 36:9, 514-518.

Author' biography with Photo



Author is assistant Professor in Mathematics Department, School of Applied Sciences, Gautam Buddha University, Greater Noida, India 201308. She has three books and a number of papers in national and international journals.