ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526
INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 3, Issue 5, May 2015

Comparison & Simulation of Different Queuing Models

Lijo John

Lecturer In Computer Engineering, Thiagarajar Polytechnic College, Alagappanagar, Thrissur, Kerala.

Abstract: Queuing theory employs mathematical analysis to determine the system's effectiveness measures. Such critical and effective measures are then used as data in the development of an optimization model for determining system efficiency and requirements. In this paper, we will discuss and provide the steps for developing the computer simulation model as well as the mathematical algorithms required to analyse the complex service network as queuing systems. A computer simulation is a realistic simulation of how a real-world process or system works over time. It is a very useful and valuable tool for analysing and evaluating any available or developed process. It can be used to learn, test, and evaluate the behaviour of many different types of problems. Simulation provides a low-cost, secure, and quick analysis tool with the necessary sensitivity analysis.

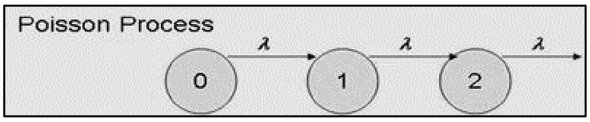
Keywords: Simulation, Queuing Models, Queuing Theory, simulation framework, queueing system

I. INTRODUCTION

Simulation is always used to assist decision makers and designers in better understanding the expected performance of the real system, as well as to test the effectiveness of the system design. The performance evaluation process is an essential activity in both the design of a new service system and the tuning of an existing one. A discrete event level performance evaluation of a service system can be implemented. It can be used and imitated by creating or developing a valid discrete event simulation model. This study will concentrate on queuing theory and simulation as effective tools for modelling the behaviour of service networks in order to optimise their performance while staying within the given constraints. Its importance is to create a simulation model to evaluate the performance of the proposed network utilisation. A model is a system abstraction obtained by making a set of assumptions about how the system works. It must capture the system's essential characteristics. The model is important during the planning stage of the system to test its performance and the effects of each variable on the system. [1]

Queuing Theory is a set of mathematical models of different queuing systems. It is widely used to analyse manufacturing and service processes with random variability in market demand (arrival times) and service times.
A queuing model is used in queuing theory to approximate a real queuing situation or system so that the queuing behaviour can be mathematically analysed. A number of useful steady-state performance measures can be determined using queueing models, including:
The average number of people in the queue or system, the average time spent in the queue or system, the statistical distribution of those numbers or times, the likelihood that the queue is full or empty, and the likelihood that the system is in a specific state These performance indicators are critical because issues or problems caused by queuing situations can lead to customer dissatisfaction with service or be the root cause of economic losses in a business. The analysis of the relevant queuing models enables the identification of the cause of queuing issues and the evaluation of the impact of proposed changes.

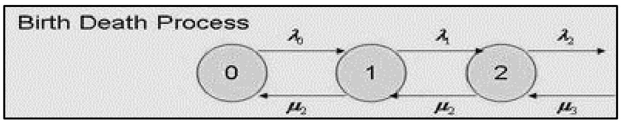
Markovian queueing model: A queuing model is said to be Markovian if the arrival process is Poisson and the service time distributions are exponential.



A Markov chain is a sequence of random variables X 1, X 2, X 3,... with the Markov property, which states that the future and past states are independent given the current state. Formally,

$$\Pr(X_{n+1}=x \mid X_1=x_1, X_2=x_2, \ldots, X_n=x_n) = \Pr(X_{n+1}=x \mid X_n=x_n).$$

Birth-death process:
The birth-death process is a subset of the continuous-time Markov process in which the states represent the current population size and the transitions are restricted to births and deaths. Birth-death processes have a wide range of applications, including demography, queuing theory, performance engineering, and biology. When a child is born, the process moves from state n to n+1. When someone dies, the process moves from state n to state n-1. Birth rates? I i=0.8 and death rates? I i=1.8 are used to define the process. [2]



Applications Of Queuing Theory

Queuing Theory has a wide range of applications, and this section will illustrate a few of them. It has three major

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526
INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 3, Issue 5, May 2015

sections: traffic flow, scheduling and facility design, and employee allocation. The examples provided are by no means the only applications where queuing theory can be put to good use; additional examples of areas where queuing theory is used are also provided.

- Traffic Flow

This is concerned with the movement of objects through a network while avoiding congestion and maintaining a steady flow in all directions.

- Queueing on roads

Queues at a highway interchange and during rush hour

- Scheduling

Computer scheduling

- Facility Planning and Management

Queues in a bank
A Mail Sorting Office

II. REVIEW OF LITERATURE

Queues are widely used in computer systems. A queue typically has one service facility and a waiting room [3]. The service facility may contain one or more servers. In general, a queue waiting room can have a finite or infinite capacity. Because of the infinite waiting room, the number of jobs waiting in line will not be limited. Not all jobs in the queueing system must be treated the same.

Priority queueing systems are queueing systems in which some jobs receive preferential treatment [4]. The queues in a priority queueing system are ordered, and the higher priority jobs are served first. All jobs are assumed to be divided into different priority classes, which are numbered from 1 to n. The smaller number denotes the higher priority class. For example, the priority 1 job will be handled before the priority 10 job. However, for jobs with the same priority, the service discipline FIFO still applies (First-in, Firstout). Priority policies are classified into two types: preemptive-resume policies and non-preemptive priority policies.

According to the preemptive-resume policy, a job of higher priority has the right to interrupt the service of a lower priority job [5]. The non-preemptive priority policy, on the other hand, states that once a server begins to handle a job, it will not be stopped until the job is completed. When using the non-preemptive priority policy, the higher priority job cannot interfere with the process of another job.
The authors of [6] analyze how to dynamically change the queueing structure to minimize the waiting time. Similar to our work, they also study a general class of queueing systems with multiple job types and a flexible service facility. This work inspires us to study the effect of priority in the queueing system. Building a simulation system is a popular method for analysing queueing performance. describes a method for analysing queueing system performance using discrete event simulation. They create a framework for fuzzy discrete event simulation (FDES) that can calculate queue performance metrics such as average queue length and waiting time. This paper is a good starting point for us to learn how to calculate queue performance metrics. In our project, we must also compute queue performance metrics like average waiting time and server utilisation.

Objectives:

• Define and parametrize the queueing system based on data collected
• Create a simulator that can measure queueing system performance, such as waiting time and server utilisation
• Using historical data, analyse the distribution of interarrival time and service time. • Examine the impact of server count on waiting time and server utilisation.

III. RESEARCH METHODOLOGY

First, we create and implement a discrete event simulation simulation framework (DES). Then, to evaluate the simulation results, we simulate various queueing systems, including those with a single server and those with multiple servers. Finally, we conduct experiments to determine the best way to increase server utilisation.

IV. RESULT AND DISCUSSION

The flexible queueing system is a queueing system with multiple job classes and heterogeneous servers in which jobs can be processed by more than one server and servers can process more than one job class [7]. Figure 1 depicts a queueing system with full flexibility, which is a type of flexible queueing system. A fully flexible queueing system means that in this system

1) Each server is capable of handling any job class. It can be seen that servers 1 and 2 can handle jobs from queues 1 and 2.
2) Any server can process the job. Jobs from queues 1 and 2 can be assigned to server 1 and server 2.

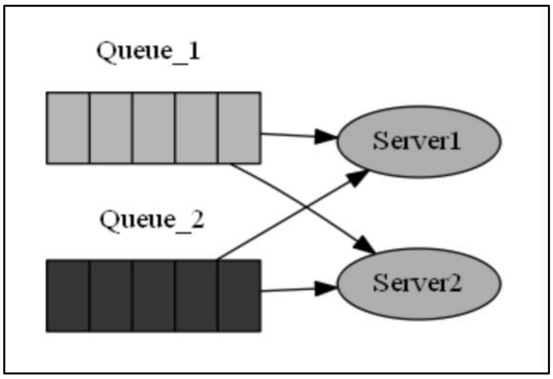


Figure 1: An example of the queueing system with full flexibility

Figure 2 depicts a queueing system with limited flexibility, which is a more general type of queueing system. A queueing system with limited flexibility entails the following:
1) Each server is capable of handling one or more job classes. Server 1 can handle jobs from queues 1 and 2, while Server 2 can handle jobs from queues 2 and 3.
2) The task can be handled by one or more servers.
The job from queue 3 can only be processed by server 2, whereas the job from queue 1 and 2 can be processed by both servers.

DOI 10.17148/IJIREEICE.2015.3558

ISSN (Online) 2321 – 2004
ISSN (Print) 2321 – 5526

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN ELECTRICAL, ELECTRONICS, INSTRUMENTATION AND CONTROL ENGINEERING
Vol. 3, Issue 5, May 2015

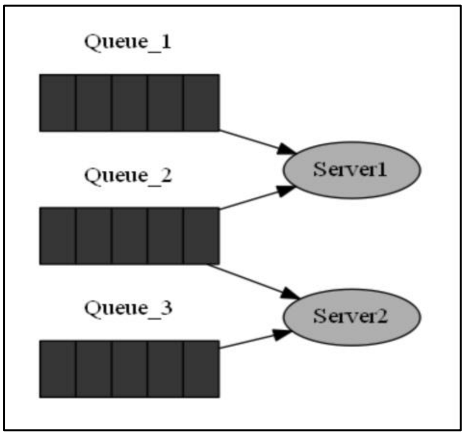


Figure 2: Example of the queueing system with limited flexibility

In the replay, we use the database timestamps to simulate the existing queueing system, which has stringent data integrity requirements. As mentioned in the previous section, several timestamps are required to define a job; missing even one of the timestamps results in a missing job in the system.

To address this issue, we are attempting to provide a more general method of simulation by simulating the real system using the estimated distribution. In our study, we assume that the system is a G/G/1 queueing system with a general distribution of interarrival times and a general distribution of service times. It is important to note that the distribution of the interarrival time and the service time differs. [8].

To simulate the interarrival time and the service time, four continuous probability distributions are chosen: the exponential distribution, the gamma distribution, the lognormal distribution, and the Weibull distribution. Because the distributions of interarrival time and service time are assumed to be independent, four different distributions will produce 16 different combinations.

Figure 3 depicts the simulation results for various types of distribution.

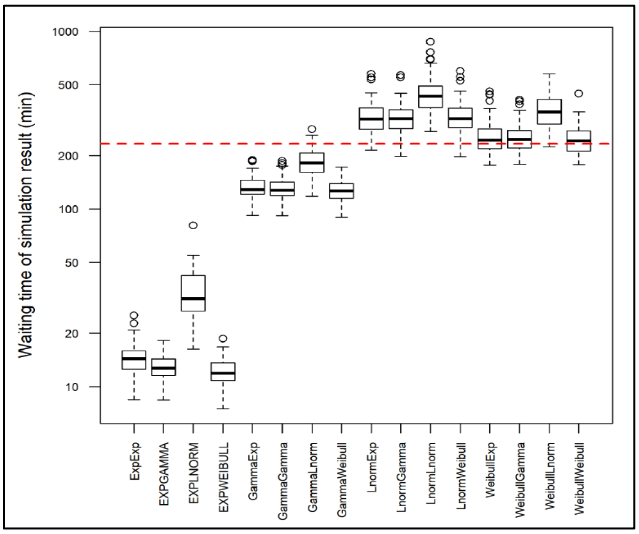


Figure 3: Simulation results based on various distribution types.

Figure 3 shows that the median value of 'WeibullWeibull' is roughly equal to the data clean result, which is strong evidence to support the prediction. Furthermore, ignoring the difference in service time distribution, the median value of interarrival time with Weibull distribution is closest to the red line. [9-10]

V. CONCLUSION

We chose a single server queueing system and used four different continuous distributions to estimate the real distribution of interarrival time and service time. The results demonstrated that the estimated distributions can be used to simulate the system. We used the queueing system transformation to analyse the impact of server numbers, allowing us to build a queueing structure with full flexibility using different numbers of servers. According to the findings, a queueing system with full flexibility and fewer servers may have a comparable waiting time and higher server utilisation than the source system.

REFERENCES

[1] A. Horv´ath, G.I.R´ozsa, M. Telek, “A map fitting method to approximate real traffic behaviour,” 8-th IFIP Workshop on Performance Modelling and Evaluation of ATM and IP Networks, Ilkley, U.K., July 2000.
[2] A. T. Andersen, B. F. Nielsen, “A markovian approach for modeling packet traffic with long-range dependence,” IEEE journal on selected areas in communications, Vol.16, No.5, pp. 719-732, 1998.
[3] P. Nain, “Application to the Modelling of Computer Systems,”. Lecture Notes, 2008
[4] S. Kelly-Bootle and B. W. Lutek, “Queueing Theory,” in Probability, Statistics, and Queuing Theory with Computer Science Applications, Elsevier, 1990.
[5] N. K. Jaiswal, Ed., “Edited by,” in Priority Queues, vol. 50, Elsevier, 1968.
[6] N. Gans and G. van Ryzin, “Optimal Control of a Multiclass, Flexible Queueing System,” Oper. Res., vol. 45, no. 5, pp. 677–693, Oct. 1997.
[7] S. Gurumurthi and S. Benjaafar, “Modeling and analysis of flexible queueing systems,” Nav. Res. Logist., vol. 51, no. 5, pp. 755–782, Aug. 2004.
[8] U. N. Bhat, “The General Queue G/G/1 and Approximations,” in An Introduction to Queueing Theory, Boston: Birkhäuser Boston, 2008, pp. 169–183.
[9] I. Buluta and F. Nori, “Quantum Simulators,” Science, vol. 326, no. 5949, pp. 108–111, Oct. 2009
[10] S. C. Borst, O. J. Boxma, and H. Levy, “The use of service limits for efficient operation of multistation single-medium communication systems,” IEEEACM Trans. Netw., vol. 3, no. 5, pp. 602–612, Oct. 1995.

DOI 10.17148/IJIREEICE.2015.3558