Journal of the Nigerian Association of Mathematical Physics
Volume **15 (November, 2009), pp 425 - 430**

Simulated queues in dynamic situations

H. I. Ojarikre
Department of Mathematics and Computer Science
Delta State University, Abraka, Nigeria.

Abstract

Discrete event simulation of dynamic situation of queuing systems has been carried out using the next-event simulated time procedure for the Monte Carlo and Area approaches. Simulation of the queuing system will be from the arrival and departure of customers from a local commercial bank. Using simulation of events, certain number k of cashiers attend to customers in queues. We designed a situation whereby in a month, we have three periods: most busy days from 25th to 5th, more busy days from 6th to 12th and busy days from 13th to 24th of the month. The dynamic nature of the periods will change the status of a number of random variables like the length of each queue, the time delay of each customer and the cashiers involved. The computational problem of this approach has been accommodated by making the dimensionality of the problem both in terms of the number of choices and sizes of the state space small.

Keywords
Monte Carlo approach, Dynamic situation, Queuing System, service times, simulation run

1.0 Introduction

Simulation is a technique for conducting numerical experiments using a model that portrays important characteristics of real world counterpart [4]. The aim of this paper is to analyze discrete-event simulation of queuing systems in dynamic situations with a view to deriving some parameters for measuring the performance of queuing system. A local commercial bank which we have nicknamed Bank EX, has been chosen for case study and collection of data.

Each point in the problem where a decision must be made is regarded as a stage while the state will be information describing the problem at each stage, generally in the form of specific values of the variable. Each customer arriving at the bank must make a decision whether to join the queue or not. Unlike in dynamic programming problems, establishing a basic recursion relationship, whereby the return at each stage is expressed as a function of data relevant to that stage [11] is avoided.

Corresponding author:
e-mail address: ojarikreify@yahoo.com
Telephone: +234-0803-422-8587

A queuing system can be described by its input (arrival process), its queue discipline and its service mechanisms, [12]. Miller [9] presented a queue-series model for reaction time. As a result of the varying complexity, the distributed nature of the queuing processes and the wide range of queueing applications, characteristics, capacity planning and performance turning of queueing process integration middleware has been a challenge for scientific managers who develop and provide solutions to queueing problems, [1]. Keane and Wolpin [7] developed an approximate solution method of discrete choice dynamic programming models by simulation using Monte Carlo integration and interpolation of the non simulated values using regression function. Monte Carlo simulation according to Taha [10] refers to the use of random sampling to estimate the output of an experiment. Simulation may not be an optimization technique but it is a technique for estimating the measure of performance of the modeled system. Several modifications of some models to fit better into more practical problems have resulted in many complex problems which can be solved using simulation techniques. It is the aim of this paper to present simulated queueing models with the aim of deriving some parameters for measuring the performance of queueing systems. Aurelio de Mosquita and Hernandez [2] described the use of spreadsheets combined with simple VBA code as a tool for teaching queueing theory and discrete –event simulation.

2.0 Formulating and implementing a simulation model

As contained in [8] simulation is the process of experimenting or using a model and noting the results which occur. Simulation provides an insight into a problem which would be unobtainable by other means. By means of simulation, the behaviour of a system can be observed over time and the actual time span can be compressed. According to [5] there are two methods for handling the simulation of arrival of customers and the servicing of customers which we shall discuss in this work. There are: fixed-time incrementing and next-event incrementing.

In fixed-time incrementing, the system is assumed to be in the initial state at a given point in time. The time is advanced by a small fixed amount, added to a register that serves as the master clock for the system to record the passage of time and then the system is updated by determining what events occurred during the elapsed time unit. The state of the system is then recorded. For a queueing theory model, the events that may occur during the elapsed time units are the arrival of customers, the servicing of customers or the departure of customers. In some cases the servicing and departure of a customer can be regarded as a single action; the service completion of a customer.

However, next-event incrementing is different from fixed-time incrementing. In next-event incrementing, the master clock is incremented by a variable amount of time rather than by a fixed amount of time each time. This procedure keeps the simulated system running without interruption until an event occurs at which point the researcher or computer pauses for a while to record the change in the system. The variable that gives the current value of simulated time is what we call the simulation clock (master clock). The idea of next event incrementing is implemented by keeping track of when the next few simulated events are scheduled to occur, jumping in simulated time to the first of these events and updating the system. In other words, there is need to keep track of two future events, namely, the next arrival and the next service completion if a customer is currently being served. These times are obtained by taking a random observation from the probability distribution of inter arrival and service times respectively.

Many authors have written about the use of the process-driven approach based on spreadsheets. Banks [3] employed these methods to generate a table called ‘ad hoc simulation table’ which contains information such as arrival time, waiting time in the queue and the total time in the system about each customer in a single server queueing system. Ingolfesson and Grossman [6] in addition to obtaining the ‘ad hoc simulation table’, also provided graphical interface displaying the customers and the servers statuses.

Tables 2.1 to 2.3 are tables showing the arrival and departure pattern in a simulation run.

Table 2.1: Events and server status

Time	**Event**	***N*(*t*)**	**Next Arrival**	**Next departure**		
				Server 1	**Server 2 …**	**Server *k***
t_o	Open	0	T_{0+i}	0.00	0.00	0.00
t_{0+i}	…	…	…	…	…	…
t_{o+i+k}	…	…	…	...	…	…

Table 2.2: Customers’ arrival and departure pattern#

Customer	Arrival Time	Service Stating time	Server	Service Finish time

Table 2.3: Servers statuses

Server 1		Server 2			…	Server *k*	
Time	**Status**	**Customer**	**Status**	**Customer**	**…**	**Status**	**Customer**
t_o	Idle	-	Idle		…	Idle	
t_{o+i}	Busy	*J*	Busy		…	Busy	⋮
	⋮	⋮	⋮	⋮	…	⋮	⋮

Table 2.1 is the event table for a given run which shows the main state variables and highlights the underlying discrete –event logic. Table 2.2 displays what happens to each customer. That is, upon arrival, the customers either join the queue or receive service immediately depending on the availability of server, each customer picks one of the servers randomly. Table 2.3 gives information about the server status throughout the simulation run.

From our observation at EX bank, Abraka, during the “busy period”, the inter-service time is less than inter arrival time. Here, few customers will have to wait or no queue at all .In other words, the queue length will go on diminishing. During the “more busy period”, the inter-service time is equal to the inter arrival time. If initially, the queue length was zero, few new arrivals will have to wait otherwise the queue length is constant. Finally, during the “most busy period”, the inter-service time is greater than inter arrival time. As a result the queue length increases indefinitely.

3.0 Numerical illustration

As a result of routine analysis of cash flow in a local commercial bank, the manager wishes to recommend to management for an additional staff. From careful study, it has been observed that customers arrive in accordance with table 3.1:

Table 3.1: Customers' arrival

	Busy period	**More busy period**	**Most busy period**
Probability of 0 customer arriving in any minute	0.5	0.4	0.1
Probability of 1 customer arriving in any minute	0.5	0.2	0.3
Probability of 2 customers arriving in any minute	0.3	0.2	0.5
Probability of 3 customers arriving in any minute	0.1	0.3	0.6

The service times also vary according to table 3.2 below:

Table 3.2: Service time

	Busy period	**More busy period**	**Most busy period**
Service time in minutes	5	12	20

The arrival and departure distribution and time spent by customers at bank EX in Abraka, is shown in table 3.3, for busy days. Arrival on the average is one customer in every 2minutes described by Poisson input and the service times for each of the 3 desk clerks are given by an exponential distribution. A single waiting line was formed and customers are processed on the basis of first come, first served with a little variation on the total time spent in the system. The tendency for a customer not to enter the system as the line gets longer and for a customer who has already joined the system to leave before being served respectively was also observed.

Table 3.3: Arrival and departure distribution of customers

S/N	**Service time begins**	**Service time ends**	**Server**	**Time**
1	8.05	8.09	2	4
2	8.09	8.15	2	6
3	8.15	8.19	3	4
4	8.19	8.24	2	5
5	8.24	8.31	1	7
6	8.31	8.37	2	6
7	8.37	8.40	3	3
8	8.40	8.46	1	6
9	8.46	8.51	3	5
10	8.51	8.57	3	6
11	8.57	9.00	2	3
12	9.12	9.16	3	4
13	9.19	9.26	1	7
14	9.26	9.30	2	4
15	9.30	9.35	3	5

Table 3.4: The frequency distribution table for the arrival in table 3.3

X	0	1	2	3	4	5	6	-7
F	0	0	0	2	4	3	1	2
FX	0	0	0	6	16	15	24	14
X^2	0	0	0	9	16	25	36	49
FX^2	0	0	0	18	64	75	144	98

$$\Sigma F = 15,\ \Sigma \mathrm{FX} = 75,\ \Sigma \mathrm{FX}^2 = 399,\ \mu = \frac{\sum FX}{\sum F} = \frac{75}{15} = 5$$

On the average, 5 minutes is spent for providing service for each customer on a busy day. This implies that in one hour 12 customers are attended to. Since 1 customer arrives in every 2 minutes on the average, then the arrival rate, λ, is 30 customers per hour on the average. Then the expected time in the system

$$E(Time\ in\ system = \frac{\mu(\frac{\lambda}{\mu})^k p_0}{(k-1)!(k\mu-\lambda)^2} + \frac{1}{\mu} = 0\cdot 186\, per\ hour = 11\cdot 15 \min s.$$

where μ is the service rate while k is the number of server and P_o is the probability that the system capacity is idle. In this case, the number of facility, k, is 3.

$$P_o = \frac{1}{\sum_{n=0}^{k-1} \frac{1}{n!}(\frac{\lambda}{\mu})^n + (\frac{\lambda}{\mu})^k \frac{k\mu}{k\mu-\lambda}} = 0\cdot 0394$$

4.0 Conclusion

Discrete Models deal with systems whose behaviour changes only at given instants. Simulation model describes the operation of the system in terms of the individual events of the component of the system, in this case, arrivals and departure of customers rather than describing the overall behaviour of the system directly. We have used the simulation run to determine the number of customers arriving and served within a specified period of time and the length of time spent in the system. The dynamic interactions between decisions and subsequent events are recorded such as the arrival time, adjustment in the length of queue if the customer decides to join the system and the next departure time. Since the events encountered in the system are not very large, next event incrementing method was adopted. However, when the number of events occurring in small interval of time is very large, a computer programme is preferred. We discovered that during ‘most busy period’, customers spend much time waiting for service. To cut down on the length of time spent waiting for service, we strongly advised that management should employ more staff.

•

References

[1] Alexander, W.P., Berry, R.F. Levine, F.E. and Urquart, R. J., (2000) “A unifying approach to performance analysis in the Java environment” IBM System Journal, Vol. 39, No. 1, PP. 118-133.

[2] Aureliode Masquita M. and Hermandez A.E. (2006). Discrete Event Simulation of Queues with spreadsheet, A teaching case. Winter Simulation Conference.

[3] Banks. J. (1998) Principles of Simulation in handbook of simulation: Principles, methodology, advances, applications and practices. John Wiley, New York.
[4] ECK, R.D. (1976) “Operations Research for Business”, Wordsworth Publishing Company, California.
[5] Hillier, F.S. and Lieberman, G.J. (1986) “An Introduction to Operations Research” Holden Day, San Francisco.

[6] Ingolfsson. A. and Grossman. T.A. (2002) “Graphical Spreadsheet Simulation of Queues”Informs Transactions on Educations, vol 2(2), Pp.27-39.
[7] Keane, M.P. and Wolpin K.I (1994). “Solution and estimation of discrete choice dynamic programming models by simulation and interpolation: Monte Carlo evidence” The Review of Economics and Statistics, Vol. 76, No. 4 pp. 648-672.
[8] Lucey, T. (1996) “Quantitative Techniques”, ELST, London.
[9] Miller J.O.(1993) “A queue-series model for reaction time with discrete-stage and continuous flow models as special cases” Psychological Review, 100, 702-715
[10] Taha, H.A. (1976) “Operations Research an introduction,” Prentice-Hall, New Jersey.
[11] Trueman Richard E. (1981) Quantitative Methods for Decision Making in Business: CBS College Publishing, New York.
[12] Wagner, H.M.(2000) “Principles of Operations Research with applications to Managerial Decisions”, Prentice-hall of India, New Delhi.