Journal of Computations & Modelling, vol.1, no.2, 2011, 27-38
ISSN: 1792-7625 (print), 1792-8850 (online)
International Scientific Press, 2011

Throughput of Queueing Networks on Based Discrete review policy

G.S. Mokaddis[1], C.H. Matta[2] and Mariam K. Metry[3]

Abstract

This paper is analyzed one of the closed multi-class queueing networks model by using the discrete review policies. The size of the batches is selected in the network according to discrete uniform distribution and is served according to Erlang distribution. Our aim is to obtain model performance measures when the number of jobs in the system is fixed, the generating parameter of service times is changed, calculate the efficiency measures of the network by using discrete-review policies results through an example.

Mathematics Subject Classification: 60K25, 90B22

Keywords: Queueing Networks, Manufacturing System, Linear programming

[1] Department of Mathematics, Faculty of Science, Ain Shams University, Cairo, Egypt, e-mail: gmokaddis@hotmail.com

[2] Department of Statistics, Faculty of Economics and Political Sciences, Cairo University, Cairo, Egypt, e-mail: cecilehm@aucegypt.edu

[3] Department of Mathematics, Faculty of Science, Ain Shams University, Cairo, Egypt, e-mail: mari25eg@yahoo.com

Article Info: *Revised* : November 7, 2011. *Published online* : November 30, 2011

1 Introduction

In this paper such processing systems are modeled as closed with multiple classes queueing networks. It is a group of service centers representing system's resources, and customers who represent users or procedures in the computer systems obtained by [1, 2, 6]. The size of the batches are selected in the network according to discrete uniform distribution and are served according to Erlang distribution. The model comprises a number of workers; that is the servers obtained by [4, 8]. Each server can treat a batch of jobs immediately. The jobs treated in the network are categorized into a number of classes in which the jobs for each class are treated in batches by only one server. When a server finishes treating a batch of a certain class, all the elements of the treated batch will show up at once, then will be set up in a queue for another server, where they become one batch under another class, waiting for a different treatment by this server obtained by [3, 5, 7].

In this way all the batches of different classes will be treated through repeated visits of all servers in the network, completing all the jobs in the network. At this point all the jobs will leave the system at once, and the system will be ready to treat other new jobs obtained by [4, 5]. Our aim is to obtain model performance measures when the number of jobs in the system is fixed, the generating parameter of service times is changed and also when the generating parameter of service times is fixed, the number of jobs in the system is changed. Finally, we calculate the efficiency measures of the network by using discrete-review policy results through an example. These are networks populated by many job classes that may differ in their service requirements and routes through the network and there is many to one relation between job classes and servers System. The controller has discretion as to the sequencing of jobs of the various classes at each server.

2 Description of Queueing Network Model

The Queueing network consists of a number (i) of servers, where $i = 1, 2, \ldots, S$. These are servers capable of handling more than one job simultaneously, which are common in some manufacturing systems such as semiconductor wafer fabs (well-drive furnace for example). The work handled in the network can be satisfied into (j) class where $j = 1, 2, \ldots, J$ and each class has infinite capacity. The services of class (j) are processed in batches by a unique servers (j), and each server can only process one job class at any point in time. Therefore, the constituency of server (i) will be denoted C_i, where $C_i = \{ j : \ s(j) = i \}$, while the $S \times J$ constituency matrix $C = [C_{ij}]$ will be following incidence matrix:

$$C_{ij} = \begin{cases} 1, & \text{if } \ s(j) = i \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

Upon completion of service at server $s(j)$, a job in a class (j) batch becomes a job of class (m) with probability P_{jm} and exits the network with probability $1 - \sum_m P_{jm}$ independent of all previous history. Since the network is closed, so the following conditions must be available for all tasks of class (j).

$$\sum_m P_{jm} = 1 \tag{2}$$

The transition matrix is symbolized as $P = [P_{jm}]$.

$$\lim_{n \to \infty} P^n = \pi \tag{3}$$

where π refers to the unique invariant probability measure associated with the routing matrix $P = [p_{jm}]$,

$$\pi = \pi P. \tag{4}$$

Finally, denote by $Q_j(t)$ the total number of class j tasks in the system at time t, and by $Q(t)$ the corresponding J-vector of queue lengths such that $N = Q(0)$ these N tasks will circulate indefinitely among servers inside the network without any departures or arrivals. A generic value of $Q(t)$ will be denoted by q and the size of this vector is defined as $|q| = \sum_j q_j$. The service times of sever (i) for class (j) batches are symbolized as $\{ST_j,\ i(I_j),\ I_j \geq 1\}$, where I_j is batch number, while service rate μ_j for class (j) tasks is:

$$\mu_j = \frac{1}{E(ST_{j,i})} = \frac{1}{S_{j,i}} \qquad (5)$$

Thus, the matrix will be as follows:

$$M = diag\{S_{1,1}, \ldots, S_{j,i}\} \qquad (6)$$

3 Basic Assumptions of Queueing Network

In this paper, we make the following assumptions on distributional characteristics of service time processes:

1. The service times $ST_{j,i}$, $j = 1, 2, \ldots, J$ are independent random variables.
2. All tasks are served in accordance with Erlang distribution with heterogeneous means for each server.
3. $E(ST_{j,i}) < \infty$, for $j = 1, 2, \ldots, J$.
4. The size of batches NB_i, $i = 1, 2, \ldots, S$, which are subtracted from each class for servicing are independent random variables for each server.
5. The size of batches is selected according to discrete uniform distribution with heterogeneous parameter for each server.
6. The maximum batch size for each server (i) will be denoted by $C_{S(j)}$, so the matrix Γ will be as follows:

$$\Gamma = diag\{C_{S(1)}, C_{S(2)}, \ldots, C_{S(j)}\} \qquad (7)$$

7. The service system inside the network starts with first class service which are performed randomly (SIRO) a service in random order, until all tasks of the first class are done, then the service order of a single class will be as follows First come First served (FCFS) while the service order of different classes will be as follows Last buffer First served (LBFS).

4 Analyzing of Queueing Network Model

In order be to understand the mechanism of queueing network model and get efficiency measurements we have to Analyze such a model by using discrete review policies. Among the most important characteristics of the discrete review policies are:

- System status is reviewed at discrete time points, $0 = t_0 < t_1 < t_2 \cdots$ and each such point the controller formulates a processing plan for the next review period, based on the queue length vector observed.
- Formulation of the plan requires solution of a linear program (LP) whose objective function involves a dynamic reward function $r(.)$. This function assigns a reward rate to the devoted in processing the different classes of jobs as a function of the observed queue length vector. Implementation of the plan involves enforcement of certain safety stock (θ) requirements in order to avoid unplanned server idleness.
- During each review period the system is only allowed to process jobs that were present in the beginning of that period which makes the implementation of the associated processing plans very simple.
- The duration of review periods and the magnitudes of safety stocks are dynamically adjusted review periods get longer and safety stocks increase as queues lengthen, but both grow less than linearly as functions of queue length.

- A discrete review policy $DR(r, L, \mu)$ is derived from a real valued, strictly positive, concave function $L(.)$ on R_+, and a real valued, strictly positive.
- Applying the discrete review policy by finding the maximum of objective in order to determine the utilization of each server in the system though solving this linear program.

$$Max\ \ r^T y$$

subject to

$$q - Ry \geq \theta\,,\quad y \geq 0\,,\quad cy \leq Le\,,\ \ \theta = LR\mu\,,\ \ R = (1 - P^T)M^{-1}\Gamma \tag{8}$$

where,

r^T : represents reward vector with dimension J, which is chosen regarding such $r_j = j$.

y : represents a vector with dimension J.

y_j : is the required time for processing the tasks of class j.

L : represents scheduled time for planning and

e : represents unit vector with dimension i.

5 Model performance Measures

We can get the most important measures of performance as follow (outputs):

5.1 Utilization

We can obtain the utilization of server (i) for class (j) by using the following:

$$U_{j,i} = \frac{y_i}{L} \tag{9}$$

and also, we can evaluate the total utilization of server (i) for all classes from the following Equations

$$U_i = \sum_{j=1}^{J} U_{j,i} \tag{10}$$

5.2 Throughput

According to the following equations we can calculate the throughout of each class individually

$$X_{j,i} = \frac{U_{j,i}}{S_{j,i}} \tag{11}$$

The total throughput for all classes at server (i):

$$X_i = \sum_{j=1}^{J} X_{j,i} \tag{12}$$

And also the throughout of the system:

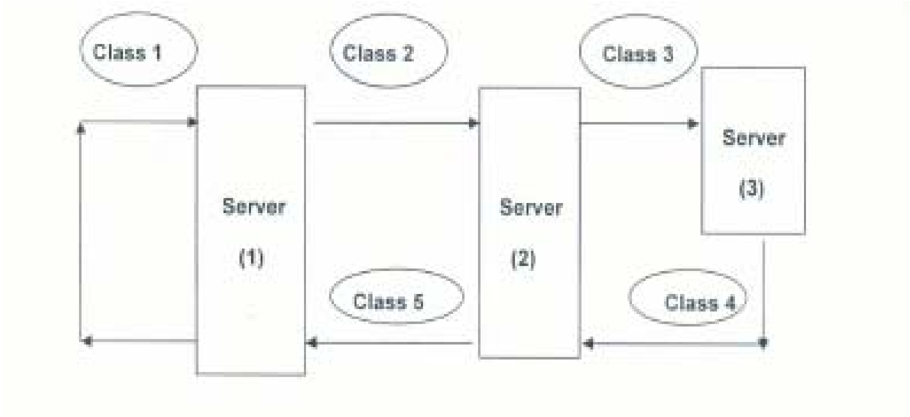
$$X = \sum_{i=1}^{S} X_i \tag{13}$$

5.3 Queue Length

To Calculate the mean queue length for the tasks of class (j) at sever (i) we can combine queue lengths for each class at the beginning of service to the end of observation period $QL_{j,i}$, then divides it by number of batches $n_j(L)$ for class j at time L as shown in the following equation

$$Q_{j,i} = \frac{QL_{j,i}}{n_j(L)} \tag{14}$$

6 Motivating Example



Firgure1: Queueing Network Model

We will present an example of a reentrant line with batch servers shown in Figure 1 and Table 1.

It consists of three servers. In addition to that, there are five classes of tasks which are being serviced in the network. The server one offers its tasks for classes one and five, the server two services classes three and four, and server three services class three only. The starting point for servicing process begins with class one and at the end of this stage the following stage begins with class two and so one thus, the total number of tasks in the network will be limited number N during service period until the integrated service process has been accomplished for the whole tasks; then the system as organization will be free of tasks and ready to welcome a new batch of tasks. The results in Table 2 and 3 and in Figure 2 and 3 were obtained by using personal computer; IMSL library; the simulation program SIM.FOR and the Excel program. We generate service times for each server from Erlang distribution with different parameters $\theta_1 = 13.0$, $\theta_2 = 6.5$, $\theta_3 = 3.0$ by using the routine RN- EXP from IMSL library by using the routine RNUND. For the generated data used in this example were obtained by giving the computer the different input in each case θ_1, θ_2, θ_3, $n_5(t_1)$, $n_5(t_2)$, $n_5(t_3)$, $n_5(t_4)$, $n_5(t_5)$ as shown in Table 1; calling the routine RNEXP from IMSL library and running the program, we calculated all following observed values L, S_j, i, QL_j, i, $n_j(L)$, q_j and μ_j.

Table 1: The different input of generated data

θ_i	$n_5(t_1)$	$n_5(t_2)$	$n_5(t_3)$	$n_5(t_4)$	$n_5(t_5)$	N
13.0	15	30	45	60	75	179
6.5	30	60	90	120	150	501
3.0	45	90	135	180	225	813
	60	120	180	240	300	1177
	75	150	225	300	375	1543
	90	180	270	360	450	1709
	105	210	315	420	550	2086
	120	240	360	480	600	2511

7 Computational results

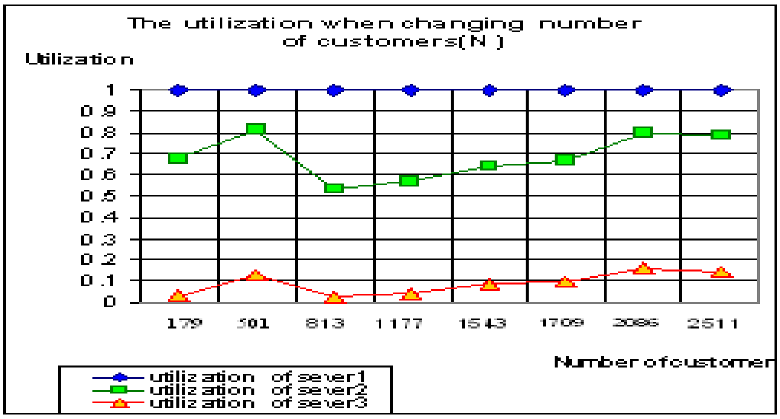


Figure 2: The utilization when changing number of customers (N)

Table 2: The utilization when changing number of customers (N)

θ	Utilization			N
	U1	U2	U3	
13.0 6.5 3.0	1.00	0.6783	0.0283	179
	1.00	0.8124	0.1262	501
	1.00	0.5331	0.0244	813
	1.00	0.5729	0.0414	1177
	1.00	0.6410	0.084	1543
	1.00	0.6685	0.950	1709
	1.00	0.7988	0.1598	2086
	1.00	0.7901	0.1437	2511

At the total utilization of server 1 is constant and equal to unity. The total utilization of servers 2 and 3 are increasing except at $N = 813$.

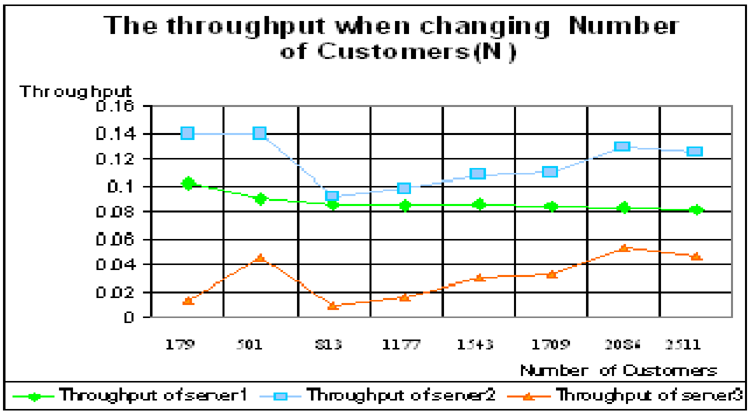


Figure 3: The throughput when changing number of customers (N)

Table 3: The throughput when changing number of customers (N)

θ	Throughput			N
	X1	X2	X3	
13.0 6.5 3.0	0.1018	0.1396	0.0132	179
	0.0899	0.1395	0.0412	501
	0.0862	0.0915	0.0090	813
	0.0854	0.0976	0.0150	1177
	0.0857	0.1085	0.0295	1543
	0.0840	0.1098	0.0324	1709
	0.0834	0.1295	0.0524	2086
	0.0813	0.1256	0.0470	2511

From Table 3 and Figure 3, we note that increasing N of the total throughput of server 1 is decreasing the total throughput of servers 2 and 3 are between increasing and decreasing.

References

[1] Baris ATA and Sunil Kumar, Heavy Traffic Analysis of open processing Networks with complete Resource pooling: Asymptotic optimality of Discrete Review Policies, *The Annals of Applied Probability*, 15(1), (2005), 331-391.

[2] G.S. Mokaddis, S.A. Matta and M.M.El-Genaidy, On Poisson Queue with three heterogeneous servers, *International Journal of Information and Management Sciences*, Taiwan-China, **9**(4), (1998), 53-66.

[3] Jingang Liu, Feng Yang, Hong Wan and John W. Fowler, *Capacity planning through queueing analysis and simulation-based statistical methods: a case study for semiconductor wafer fabs*, 1998.

[4] S. Kumar and P.R. Kumar, Queueing network models in the design and analysis of semiconductor wafer fabs, *IEEE Transactions on* 1042, **296X**, (2001), 548-561.

[5] S. Kumar and D.A. Nottestad, Flexible capacity design for the Focus Factory: a case study, *International Journal of Production Research*, **47**(5), (2009), 1269-1286.

[6] A. M. Law and W. D. Kelton, *Simulation Modeling and Analysis*, 3rd edition, McGraw-Hill, New York, 2000.

[7] Mariam K. Metry and N.A. Hassan, Performane analysis of multiclass queueing Network model with positive Poisson batch sizes following discrete review policy, *MASA*, **4**, (2009).

[8] F. Yang, Neural network meta modeling for Cycle time throughput profiles in manufacturing, *Journal of Operational Research*, (2010), 172-185.