The finite source queueing model for multiprogrammed computer systems with different CPU times and different I/O times

BRIAN D. BUNDAY and ESMAILE KHORRAM

Abstract

This paper discusses the finite source queueing model as it applies to a multiprogrammed computer system. The system processes N jobs using r Central Processing Units (CPU's) where $r<N$. The jobs emanate from peripheral devices, terminals, card readers etc. (I/O devices) at which it is assumed they suffer no delay.

If a CPU is available when a job requires service it is given this service. Otherwise a queue of jobs is formed. In the situation where there are more than r jobs requiring service, it is assumed that r randomly selected jobs are assigned to each of the r CPU's. It is assumed that the service time of job i has a negative exponential distribution with mean $1/\mu_i$. After service, job i returns to I/O devices for a random time before again calling for CPU service. This time is assumed to have a general distribution with mean $1/\lambda_i$.

A closed form solution for the steady-state probabilities that a particular set of jobs is at I/O processes is obtained. It is shown that the steady-state solution depends on the distribution of time at I/O devices only through the value $1/\lambda_i$. It is also shown how other important measures such as CPU utilisation, as well as waiting times and response times for the jobs, can be computed from this solution.

1. Introduction

A number of authors have applied the methods of queueing theory to the study of multiprogrammed computer systems. Following Sztrik [11] we can model such systems as follows. We suppose that there are N jobs in the system, each one emanating from a terminal at which it suffers no delay and to which it returns following CPU processing. There are $r(<N)$ CPU's in the system. If a CPU is available an arriving job (program) is immediately served by one of the available CPU's. Otherwise a queue of jobs is formed. The jobs would normally be served on a FIFO (first-in, first-out) basis. For job i we assume that its service time is exponentially distributed with mean $1/\mu_i$. We also suppose that the time job i spends at the peripheral devices (I/O operations) is a random variable with distribution function $F_i(x)$ or more conveniently survivor function $G_i(x)=1-F_i(x)$. These times are independent of each other and are different for the different jobs.

The queueing model just described was first used in the context of the "machine interference problem" by Ashcroft [1] who studied the $M/G/1$ case by way of the duration of the busy period of the operative (the CPU). Using the birth-death equations Benson and Cox [3] obtained a solution to the $M/M/1$ case and extended it to the $M/M/r$ case for which Peck and Hazelwood [9] computed extensive tables for work study applications. An important advance was made by Bunday and Scraton [4] who showed that the solution to the $G/M/r$ homogenenous case was the same as the $M/M/r$ solution.

There is a considerable literature showing applications of this and similar models to the computer systems situation. Early contributions were made by Gaver [6] and Avi-Itzhak and Heyman [2] while more recently we have the papers by Cohen [5], Schatte [10], Kameda [7] and Sztrik [11, 12]. The book by Kleinrock [8] contains an extensive bibliography as well as a discussion of other models.

The present paper extends the work of Sztrik and presents a closed form solution of the $\vec{G}/\vec{M}/r$ case in the steady-state situation for a queue discipline in which jobs are randomly allocated to CPU's whenever a new job calls for service or the service of a job is completed. From this it is easy to compute such quantities as the CPU utilisation and the expected waiting times and response times of the jobs. It is shown that these quantities depend on the distribution of the times spent at the I/O processes only through the means of these distributions.

2. The steady-state equations for the model

We consider a set of N jobs in a system with r CPU's. Service times for each job are assumed to have a negative exponential distribution with mean $1/\mu_i$ for job i. The times spent at I/O processes for each job are independently distributed.

Let $G_i(t)$ denote the probability that if job i arrives at I/O processes at time zero then it is still there at time t later. Thus

$$G_i(0) = 1 \quad \text{and} \quad G_i(\infty) = 0 \quad \text{for all } i.$$

Further if job i is at I/O processes at time t the probability that it will call for CPU service in the interval $(t, t+\delta t)$ is

$$-G_i'(t)\,\delta t/G_i(t) \quad \text{to first order in } \delta t. \tag{2.1}$$

The mean time spent at I/O by job i will be

$$\frac{1}{\lambda_i} = \int_0^\infty t[-G_i'(t)]\,dt = \int_0^\infty G_i(t)\,dt \tag{2.2}$$

Let $Q_{i_1 i_2 \dots i_n}(t_{i_1}, t_{i_2} \dots, t_{i_n}; \tau)\,dt_{i_1} dt_{i_2}, \dots, dt_{i_n}$ be the probability that at time τ a particular set $i_1, i_2, \dots, i_n$ of the N jobs are at I/O, one of them for a time in $(t_{i_1}, t_{i_1}+dt_{i_1})$, etc,, ..., another for a time in $(t_{i_n}, t_{i_n}+dt_{i_n})$, and the other jobs require CPU service. In the case of negative exponential service, n, $t_{i_1}, t_{i_2}, \dots, t_{i_n}$, and τ provide an adequate description of the system. We do not need to specify the state of each service at time τ since this will not influence the future behaviour of the system. Indeed we need not even specify which particular jobs are being serviced since the residual service time has the same distribution whether or not the service has been started.

We consider the transitions that may occur in $(\tau, \tau+\delta\tau)$ working always to first order in $\delta\tau$.

$$Q_{i_1 i_2 \dots i_N}(t_{i_1}+\delta\tau, \dots, t_{i_N}+\delta\tau; \tau+\delta\tau) =$$

$$= Q_{i_1, i_2 \dots i_N}(t_{i_1}, t_{i_2}, \dots, t_{i_N}; \tau)\left[1-\delta\tau \sum_{s=1}^{N} \{-G'_{i_s}(t_{i_s})/G_{i_s}(t_{i_s})\}\right]. \tag{2.3}$$

It is convenient to denote $\{i_1 i_2 \dots i_n\}$ the set of jobs at I/O by A_n while $B_n = A_n^c$ denotes the set of jobs calling for CPU service.

$$Q_{i_1, i_2 \dots i_n}(t_{i_1}+\delta\tau, \dots, t_{i_n}+\delta\tau; \tau+\delta\tau) =$$

$$= Q_{i_1 i_2 \dots i_n}(t_{i_1}, t_{i_2}, \dots, t_{i_n}; \tau)\left[1-\delta\tau \sum_{s=1}^{n} \{-G'_{i_s}(t_{i_s})/G_{i_s}(t_{i_s})\} - \delta\tau \sum_{j \in B_n} \mu_j\right]+$$

$$+\delta\tau \sum_{j \in B_n} \int_0^{\tau} Q_{i_1 i_2 \dots i_n, j}(t_{i_1}, t_{i_2}, \dots, t_{i_n}, t_j; \tau)\{-G'_j(t_j)/G_j(t_j)\}\, dt_j \tag{2.4}$$

for all groups $i_1, i_2, \dots, i_n$ such that $N-r \leqq n \leqq N-1$.

$$Q_{i_1 i_2 \dots i_n}(t_{i_1}+\delta\tau, \dots, t_{i_n}+\delta\tau; \tau+\delta\tau) =$$

$$= Q_{i_1 i_2 \dots i_n}(t_{i_1}, t_{i_2}, \dots, t_{i_n}; \tau)\left[1-\delta\tau \sum_{s=1}^{n} \{-G'_{i_s}(t_{i_s})/G_{i_s}(t_{i_s})\} - \frac{r}{N-n}\delta\tau \sum_{j \in B_n} \mu_j\right]+$$

$$+\delta\tau \sum_{j \in B_n} \int_0^{\tau} Q_{i_1 i_2 \dots i_n, j}(t_{i_1}, t_{i_2}, \dots, t_{i_n}, t_j; \tau)\{-G'_j(t_j)/G_j(t_j)\}\, dt_j \tag{2.5}$$

for all groups $i_1, i_2, \dots, i_n$ such that $1 \leqq n \leqq N-r$.

In (2.5) the particular set of r jobs being serviced is equally likely to be any one of the $\binom{N-n}{r}$ sets possible. This is equivalent to assuming that whenever the number of jobs requiring service exceeds r, then we have a SIRO (service in random order) queue discipline. This is a somewhat artificial situation and is certainly different from the more natural FIFO discipline. In the latter case the resulting system of equations has no explicit solution. We shall show that for the queue discipline adopted the equations can be solved. In many cases, provided the inhomogeneity is not excessive, our solution, particularly in respect of the important properties of the system, will be a good approximation to the FIFO case. Its closed and easily computed form is its merit.

$$Q_0(\tau+\delta\tau) = Q_0(\tau)\left[1-\frac{r}{N}\delta\tau \sum_{j=1}^{N} \mu_j\right]+$$

$$+\delta\tau \sum_{j=1}^{N} \int_0^{\tau} Q_j(t_j; \tau)\{-G'_j(t_j)/G_j(t_j)\, dt_j. \tag{2.6}$$

If we consider the situation when a service is completed in the interval $\tau, \tau+\delta\tau$

$$Q_{i_1 i_2 \dots i_n j}(t_{i_1}+\delta\tau, t_{i_2}+\delta\tau, \dots, t_{i_n}+\delta\tau, 0; \tau+\delta\tau)\cdot\delta\tau =$$

$$= \mu_j \delta\tau Q_{i_1 i_2 \dots i_n}(t_{i_1}, t_{i_2}, \dots, t_{i_n}; \tau) \tag{2.7}$$

for all $j \in B_n$ and all groups $i_1, \ldots, i_n$ such that $N-r \leqq n \leqq N-1$.

$$Q_{i_1 i_2 \ldots i_n j}(t_{i_1}+\delta\tau, t_{i_2}+\delta\tau, \ldots, t_{i_n}+\delta\tau, 0; \tau+\delta\tau)\cdot\delta\tau =$$

$$= \frac{r\delta\tau}{N-n}\mu_j Q_{i_1 i_2 \ldots i_n}(t_{i_1}, t_{i_2}, \ldots, t_{i_n}; \tau) \tag{2.8}$$

for all $j \in B_n$ and all groups $i_1, \ldots, i_n$ such that $0 \leqq n \leqq N-r$.

If we consider the situation as $\tau \to \infty$ so that

$$Q_{i_1 i_2 \ldots i_n}(t_{i_1}, t_{i_2}, \ldots, t_{i_n}; \tau) \to Q_{i_1 i_2 \ldots i_n}(t_{i_1}, t_{i_2}, \ldots, t_{i_n})$$

and further write

$$Q_{i_1 i_2 \ldots i_n}(t_{i_1}, t_{i_2}, \ldots, t_{i_n}) = G_{i_1}(t_{i_1})G_{i_2}(t_{i_2})\ldots G_{i_n}(t_{i_n})R_{i_1 i_2 \ldots i_n}(t_{i_1}, t_{i_2}, \ldots, t_{i_n}) \tag{2.9}$$

then (2.3) to (2.8) take the form

$$\left[\frac{\partial}{\partial t_{i_1}}+\ldots+\frac{\partial}{\partial t_{i_N}}\right] R_{i_1 \ldots i_N}(t_{i_1}, t_{i_2}, \ldots, t_{i_N}) = 0 \tag{2.10}$$

$$\left[\frac{\partial}{\partial t_{i_1}}+\ldots+\frac{\partial}{\partial t_{i_n}}\right] R_{i_1 \ldots i_n}(t_{i_1}, t_{i_2}, \ldots, t_{i_n}) = -R_{i_1 \ldots i_n}(t_{i_1}, t_{i_2}, \ldots, t_{i_n}) \sum_{j \in B_n} \mu_j -$$

$$-\sum_{j \in B_n} \int_0^{\infty} R_{i_1 \ldots i_n j}(t_{i_1}, t_{i_2}, \ldots, t_{i_n}, t_j) G_j'(t_j)\, dt_j \tag{2.11}$$

for all groups $i_1, \ldots, i_n$ such that $N-r \leqq n \leqq N-1$.

$$\left[\frac{\partial}{\partial t_{i_1}}+\ldots+\frac{\partial}{\partial t_{i_n}}\right] R_{i_1 \ldots i_n}(t_{i_1}, t_{i_2}, \ldots, t_{i_n}) = -\frac{r}{N-n} R_{i_1 i_2 \ldots i_n}(t_{i_1}, \ldots, t_{i_n}) \sum_{j \in B_n} \mu_j -$$

$$-\sum_{j \in B_n} \int_0^{\infty} R_{i_1 \ldots i_n j}(t_{i_1}, \ldots, t_{i_n}, t_j) G_j'(t_j)\, dt_j \tag{2.12}$$

for all groups $i_1, \ldots, i_n$ such that $1 \leqq n \leqq N-r$.

$$0 = R_0 \frac{r}{N} \sum_{j=1}^{N} \mu_j + \sum_{j=1}^{N} \int_0^{\infty} R_j(t_j) G_j'(t_j)\, dt_j. \tag{2.13}$$

$$R_{i_1 \ldots i_n j}(t_{i_1}, t_{i_2}, \ldots, t_{i_n}, 0) = \mu_j R_{i_1 \ldots i_n}(t_{i_1}, \ldots, t_{i_n}) \tag{2.14}$$

for all $j \in B_n$ and groups $i_1, \ldots, i_n$ such that $N-r \leqq n \leqq N-1$.

$$R_{i_1 \ldots i_n j}(t_{i_1}, t_{i_2}, \ldots, t_{i_n}, 0) = \frac{r}{N-n} \mu_j R_{i_1 \ldots i_n}(t_{i_1}, \ldots, t_{i_n}) \tag{2.15}$$

for all $j \in B_n$ and groups $i_1, \ldots, i_n$ such that $0 \leqq n \leqq N-r$.

It is perhaps worth mentioning that these same equations would result from a second queue discipline which Tomkó refers to as "processor sharing". In this situation whenever there are more jobs demanding service than CPU's, i.e. $N-n>r$, then all jobs receive service on each CPU in such a way that during a unit time every job receives an amount $1/(N-n)$ CPU service on every CPU. This will approximate the case when all CPU's operate in time sharing. See also Cohen [5].

3. The solution of the steady state equations

The general solution of (2.10) is

$$R_{i_1 \ldots i_N}(t_{i_1}, t_{i_2}, \ldots, t_{i_N}) = g(t_{i_1}-t_{i_2}, t_{i_1}-t_{i_3}, \ldots, t_{i_1}-t_{i_N})$$

where g is an arbitrary function.

But $R_{i_1 \ldots i_n}(t_{i_1}, t_{i_2}, \ldots, t_{i_n})$ is a symmetric function for all A_n so that

$$R_{i_1 \ldots i_N}(t_{i_1}, t_{i_2}, \ldots, t_{i_N}) = \varkappa \tag{3.2}$$

where $\varkappa$ is a constant is a solution.

From (2.14) and (2.15) we obtain in turn

$$R_{i_1 \ldots i_n}(t_{i_1}, t_{i_2}, \ldots, t_{i_n}) = \frac{\varkappa}{\prod_{k=1}^{N-n} \mu_{j_k}} \tag{3.3}$$

where $j_k \in B_n$ and $N-r \leqq n \leqq N-1$.

$$R_{i_1 \ldots i_n}(t_{i_1}, t_{i_2}, \ldots, t_{i_n}) = \frac{(N-n)!\,\varkappa}{r^{N-n-r}\, r! \prod_{k=1}^{N-n} \mu_{j_k}} \tag{3.4}$$

where $j_k \in B_n$ and $0 \leqq n \leqq N-r$.

These solutions also satisfy (2.11) to (2.13).

Thus the probability that a particular group $i_1, i_2, \ldots, i_n$ of jobs are at I/O and the rest are not is

$$Q_{i_1 i_2 \ldots i_n} = \int_0^\infty \cdots \int_0^\infty Q_{i_1 i_2 \ldots i_n}(t_{i_1}, t_{i_2}, \ldots, t_{i_n})\, dt_{i_1} \ldots dt_{i_n}$$

so that from (2.2)

$$Q_{i_1 \ldots i_n} = \frac{(N-n)!\,\varkappa}{r^{N-n-r}\, r! \prod_{k=1}^{N-n} \mu_{j_k}} \prod_{j=1}^{n} \lambda_{i_j}^{-1} \tag{3.5}$$

for all groups with $0 \leqq n \leqq N-r$.

$$Q_{i_1 \ldots i_n} = \frac{\varkappa \prod_{j=1}^{n} \lambda_{i_j}^{-1}}{\prod_{k=1}^{N-n} \mu_{j_k}} \tag{3.6}$$

for all groups with $N-r \leqq n \leqq N$.

Thus the probability that n jobs are at I/O is

$$q_n = \sum_{\{i_1 \ldots i_n\}} \frac{(N-n)!\,\varkappa}{r^{N-n-r}\, r! \prod_{k=1}^{N-n} \mu_{j_k}} \prod_{j=1}^{n} \lambda_{i_j}^{-1} \tag{3.7}$$

for $0 \leqq n \leqq N-r$.

$$q_n = \sum_{\{i_1 \ldots i_n\}} \frac{\varkappa}{\prod_{k=1}^{N-n} \mu_{j_k}} \prod_{j=1}^{n} \lambda_{i_j}^{-1} \tag{3.8}$$

for $N-r \leqq n \leqq N$.

$\varkappa$ is determined by the condition

$$\sum_{n=0}^{N} q_n = 1. \tag{3.9}$$

4. Some useful measures for the system

The probability that all N jobs are at I/O is

$$q_N = \varkappa/[\prod_{j=1}^{N} \lambda_j]. \tag{4.1}$$

Since if n jobs are at I/O the probability that a particular CPU is servicing is $\frac{N-n}{r}$ if $N-n \leqq r$ or 1 if $N-n > r$ then the proportion of time each CPU is servicing, the CPU utilisation, is given by

$$U = [\sum_{k=1}^{r} kq_{N-k} + \sum_{k=r+1}^{N} rq_{N-k}]/r. \tag{4.2}$$

For a particular job i if $q^{(i)}$ denotes the long run proportion of time that job i is at I/O processes, then

$$q^{(i)} = \sum_{n=1}^{N} \sum_{i \in \{i_1 \ldots i_n\}} Q_{i_1 \ldots i_n}. \tag{4.3}$$

Using a result due to Tomkó [13] we also have

$$q^{(i)} = 1/\lambda_i/\{1/\lambda_i + W_i + 1/\mu_i\} \tag{4.4}$$

where W_i is the mean time that job i waits not being serviced by a CPU. Thus

$$W_i = (1-q^{(i)})/(\lambda_i q^{(i)}) - 1/\mu_i. \tag{4.5}$$

Of course with the queue discipline being considered the total waiting time may be made up of a number of such periods. The particular job may, at some stage, be in the selected set of those being serviced, and following a service completion or the arrival of another job may then not be in the selected set and will have to wait.

The mean response time of job i is given by

$$T_i = W_i + 1/\mu_i = (1-q^{(i)})/(\lambda_i q^{(i)}) \tag{4.6}$$

so that the mean number of jobs that are calling for and receiving CPU service is given by

$$\bar{N} = \sum_{i=1}^{N} (1 - q^{(i)}) = \sum_{i=1}^{N} \lambda_i T_i q^{(i)}. \tag{4.7}$$

Of course in the case of processor sharing as mentioned at the end of Section 2 there is no waiting time. However the mean response time as given by (4.6) is still appropriate for this discipline.

Acknowledgement

We are very grateful to Professor J. Tomkó who commented on an earlier version of this paper. His valuable and constructive criticism has, we believe, led to an improved presentation of this work.

SCHOOL OF MATHEMATICAL SCIENCES,
UNIVERSITY OF BRADFORD, U.K.

References

[1] H. Ashcroft, The Productivity of Several Machines under the Care of One Operator, J. Roy. Stat. Soc. B, 12 (1) (1950), 145—151.
[2] B. Avi-Itzhak and D. P. Heyman, Approximate Queueing Models for Multiprogramming Computer Systems, Opns. Res., 21 (1973), 1212—1230.
[3] F. Benson and D. R. Cox, The Productivity of Machines Requiring Attention at Random Intervals, J. Roy. Stat. Soc. B, 13 (1951), 65—82.
[4] B. D. Bunday and R. E. Scraton, The G/M/r Machine Interference Model, Eur. J. Operation Res., 4 (1980), 399—402.
[5] J. W. Cohen, The Multiple Phase Service Network with Generalised Processor Sharing, *Acta Informatica,* 12 (1979), 245—284.
[6] D. P. Gaver, Probability Models for Multiprogramming Computer Systems, J. ACM, 3 (1967), 423—438.
[7] H. Kameda, A Finite-Source Queue with Different Customers, J. ACM, 29 (1982), 478—491.
[8] L. Kleinrcck, Queueing Systems, Vol. 2: Computer Applications, Wiley-Interscience, New York, 1976.
[9] L. G. Peck and R. N. Hazelwood, Finite Queueing Tables: ORSA Publications in Operations Research 2, Wiley, New York, 1958.
[10] P. Schatte, On the Finite Population GI/M/1 Queue and its Application to Multiprogrammed Computers, *Journal of Information Processing and Cybernetics,* 16 (1980), 433—441.
[11] J. Sztrik, Probability Model for Non-Homogeneous Multiprogramming Computer System, *Acta Cybernetica,* 6 (1983), 93—101.
[12] J. Sztrik, A Queueing Model for Multiprogrammed Computer Systems with Different I/O Times, *Acta Cybernetica,* 7 (1984), 127—135.
[13] J. Tomkó, Semi-Markov analysis of the inhomogeneous machine interference model: Lecture Notes in Control and Information Sciences, 84. System Modelling and Optimization. Proc. of the 12th IFIP Conf. Hungary, 1985, 992—1001.

(Received March 27, 1987, revised Nov. 3, 1987)