



Calhoun: The NPS Institutional Archive

Theses and Dissertations

Thesis Collection

1984

Normal approximation for response time in a processor-shared computer system model.

Pornsuriya, Suriya

Monterey, California. Naval Postgraduate School

<http://hdl.handle.net/10945/19471>



Calhoun is a project of the Dudley Knox Library at NPS, furthering the precepts and goals of open government and government transparency. All information contained herein has been approved for release by the NPS Public Affairs Officer.

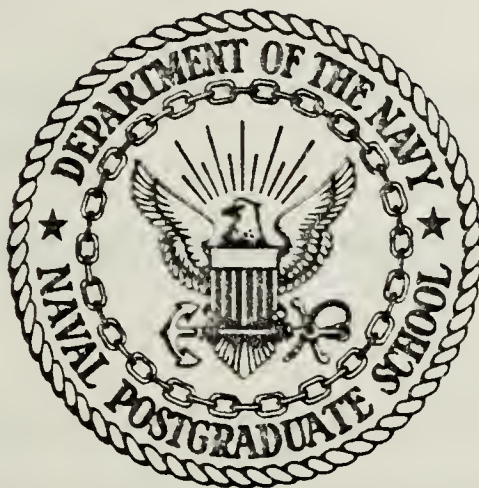
Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

<http://www.nps.edu/library>



NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

NORMAL APPROXIMATION FOR
RESPONSE TIME IN A
PROCESSOR-SHARED COMPUTER SYSTEM MODEL

by

Suriya Pornsuriya

March 1984

Thesis advisor:

P. A. Jacobs

Approved for public release; distribution unlimited

T215671

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA 93943

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Normal Approximation for Response Time in a Processor-Shared Computer System Model		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis March 1984
7. AUTHOR(s) Suriya Pornsuriya		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, CA 93943		8. CONTRACT OR GRANT NUMBER(s)
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, CA 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE March 1984
		13. NUMBER OF PAGES 107
		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for Public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Response Time, Central Limit Theorem, Heavy Traffic		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) In a time-shared computer system, the processor allocates its processing time equally to all jobs submitted for service from a fixed number of terminals. Under Markov assumptions, i.e., independent identically distributed exponential terminal think times and job requested service times, the distribution of response time of a tagged job theoretically can be determined by solving a system of differential equations derived for each initial system state. However, explicit closed form solutions to these equations are quite		

complex. The Central Limit Theorem and heavy traffic arguments suggest normal approximations to the distribution of the response time. Simulation of the response time is used to study the accuracy of these normal approximations to the response time distribution via moments and quantiles. Finally, the analysis is extended to a model for a system with two types of terminals.

Approved for public release; distribution unlimited.

Normal Approximation for
Response Time
in a Processor-Shared Computer System Model

by

Suriya Pornsuriya
Ensign, Royal Thai Navy

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 1984

ABSTRACT

In a time-shared computer system, the processor allocates its processing time equally to all jobs submitted for service from a fixed number of terminals. Under Markov assumptions, i.e. independent identically distributed exponential terminal think times and job requested service times, the distribution of response time of a tagged job theoretically can be determined by solving a system of differential equations derived for each initial system state. However, explicit closed form solutions to these equations are quite complex. The Central Limit Theorem and heavy traffic arguments suggest normal approximations to the distribution of the response time. Simulation of the response time is used to study the accuracy of these normal approximations to the response time distribution via moments and quantiles. Finally, the analysis is extended to a model for a system with two types of terminals.

TABLE OF CONTENTS

I.	INTRODUCTION	7
	A. BACKGROUND	7
	E. PROCESSOR-SHARING SYSTEM.	7
	C. MODEL AND APPROXIMATE DISTRIBUTION	9
II.	MEAN AND MOMENTS FOR A SINGLE JOB TYPE MODEL . . .	12
	A. INTRODUCTION	12
	B. CONDITION CN REQUIRED TIME AND SYSTEM STATE.	13
	C. CONDITION CN REQUIRED TIME	14
	D. MOMENTS AND VARIANCE OF RESPONSE TIME	16
	E. NUMERICAL RESULTS	18
III.	SIMULATION FOR ONE JOB TYPE MODEL	20
	A. INTRODUCTION	20
	B. WORK TIME	21
	C. SIMULATION FOR A 2-TERMINAL SYSTEM	22
	1. Algorithm	22
	2. Numerical Results	24
	D. SIMULATION FOR AN N-TERMINAL SYSTEM	25
	1. Algorithm	25
	2. Moments of Response Time	28
	3. Computation of Simulated Variance	30
	4. Computation of Simulated Skewness	30
	5. Computation of Simulated Kurtosis	31
	6. Standard Error of the Mean Response Time	32
	7. Numerical Results	33

IV.	NORMAL APPROXIMATION FOR RESPONSE TIME	36
	A. INTRODUCTION	36
	B. APPROXIMATION BY CENTRAL LIMIT THEOREM	37
	1. A Central Limit Theorem for $W(t)$	37
	2. A Central Limit Theorem for response time, $R(T)$	39
	C. APPROXIMATION BY HEAVY TRAFFIC ANALYSIS	40
	D. COMPARISON TO SIMULATION DATA	42
V.	MODEL FOR A SYSTEM WITH TWO TYPES OF TERMINALS	50
	A. INTRODUCTION	50
	B. STEADY-STATE DISTRIBUTION	51
	C. A CENTRAL LIMIT THEOREM FOR THE RESPONSE TIME	55
	D. HEAVY TRAFFIC APPROXIMATION FOR THE RESPONSE TIME	58
	E. SIMULATION	63
	1. Algorithm	63
	2. Computation of Simulated Mean, Variance, Skewness and Kurtosis	65
	3. Numerical Results	68
VI.	SUMMARY AND CONCLUSION	79
	A. SUMMARY	79
	B. CONCLUSION	81
	APPENDIX A: SIMULATION PROGRAM FOR ONE-TYPE MODEL	83
	APPENDIX B: SIMULATION PROGRAM FOR TWO-TYPE MODEL	93
	LIST OF REFERENCES	104
	INITIAL DISTRIBUTION LIST	106

I. INTRODUCTION

A. BACKGROUND

When a computer user, from his terminal, submits a job to the computer, it would be desirable for him that the job be processed right away. However, this rarely happens in the real world, because a computer system usually consists of only one Central Processing Unit (CPU) and many terminals. So if more than one person uses the computer, there will be jobs that request processing at the same time. Hence, some kind of queuing system or time-sharing technique will be needed to organize the allocation of processing time to those submitted jobs.

Generally, a computer system has two types of processing time allocation. One is called "Batch Processing". All the jobs submitted this way form a kind of queue and wait to be served (processed) according to the well known First-Come-First-Served (FCFS) policy, i.e. the first submitted job is processed to completion, then the computer is set up for the next waiting job; in some systems a priority policy based on the length of a job is used to determine the next job to get dedicated use of the processor. The other one which is of interest for this thesis is called "Time-Sharing Processing". It is based on a technique that permits concurrent processing of two or more jobs. Each job no matter when it is submitted gets an equal share of processing time until completion.

E. PROCESSOR-SHARING SYSTEM.

The so-called "Processor-Sharing" or "pure time-sharing" in computer engineering is the system in which the processor

shares its service (processing time) equally among all jobs submitted. In other words, if an individual job requiring a certain amount of processing time is tagged and submitted to the system and finds $(j-1)$ other jobs being processed, then from now on all j jobs, will each receive service (processing time) equal to $(1/j)$ -th of a (processing) time unit per time unit. Of course the rate at which submitted jobs receive service changes each time a new arrival joins the system and each time a completed job departs.

This abstraction of computer capacity allocation may be described in more formal terms as follows : if the chance that any single job, processed alone, finishes in time interval $(t, t+h)$ is $\mu h + o(h)$, (exponential-Markov service), then the chance that a particular "tagged" job in the company of $(j-1)$ others finishes in $(t, t+h)$ is $\mu(h/j) + o(h)$ as h approaches zero.

Clearly, there is no waiting line in processor sharing of the above type. It permits short jobs access to processing right away even if they arrive after longer jobs.

Processor sharing is an approximation to the processor sharing "Round-Robin" model. In this model once a particular tagged job enters the system, it joins the end of an ordered queue. When it reaches the service point it is allocated a fixed quantum (q) of service time. If the job completes within this time it simply leaves the system. If after q seconds it still requires more service, it is immediately returned to the end of the queue. This process then goes on and on until the required service is completed.

In the limit, however, as q approaches zero, the Round-Robin system becomes the Processor-Sharing system. The latter system here will be studied for its characteristics relating to the distribution of response time, i.e. the time that a tagged job requiring a certain amount of processing time actually taken to complete the service and leave the system.

C. MODEL AND APPROXIMATE DISTRIBUTION

Apparently the first study of delays to arriving jobs under processor sharing was conducted by Coffman, Muntz, and Trotter (1970). [Ref. 1] They assumed a steady state M/M/1 system under processor sharing, i.e. Poisson arrivals and exponential service times with a single server processor. They then derived an expression for the Laplace transform of the waiting time distribution of an arriving job conditioned on the processing time it requires and the number of jobs it finds in the system on arrival.

The properties of the response time, R , given the processing time required by the arriving job under processor sharing system was further analyzed by D. Mitra (1981) [Ref. 2] for the following model. A system consisting of N terminals and a single computer (CPU) can be modeled as a classical machine-repair situation: each thinking terminal (failure-prone machine) applies for computer service at rate λ , and queued or waiting jobs are served at rate μ as long as any jobs are present. If Markov assumptions are made throughout, then $X(t)$, the number of jobs at the service stage, is a birth and death process with transition rates:

$$\begin{aligned}
 X(t) = j & \longrightarrow X(t+h) = j + 1 & : & \lambda_j h + o(h) \\
 & \longrightarrow X(t+h) = j - 1 & : & \mu_j h + o(h) & (1.1) \\
 & \longrightarrow X(t+h) = j & : & 1 - (\lambda_j + \mu_j) h + o(h)
 \end{aligned}$$

where $\lambda_j = \lambda(N-j)$ is the rate at which a job is submitted to the computer when there are already j jobs in the system, and $\mu_j = \mu$ for $j \geq 1$, otherwise being zero, is

the rate at which the computer gives service to all j jobs submitted. Based on the above transition rates, the distribution of response time under processor sharing is characterized, and the moments such as mean and variance are found under interesting conditions, i.e. the conditional response time, given only the processing requirement T time units, derived from the condition that the tagged job requiring T time units of processing arrives to find $(j-1)$ others in the system. [Ref. 2]

Gaver, Jacobs and Latouche [Ref. 3] have generalized and extended the previous analysis by introducing the idea of processor sharing in an arbitrary birth and death process environment, thus allowing quite general terminal-computer interactions to be represented. In the process, the meaning of "system state at the moment of tagged job arrival" is also clarified by Lavenberg and Reiser. [Ref. 4] Response time characteristics are computed under the assumptions that processor-sharing service rates are processor-state-dependent in a more general way than that described earlier; this allows for approximate representation of overhead penalties and also of job scheduling. Other characteristics of tagged job response are also studied, e.g. the accumulated processing work, $W(t')$, actually performed on that job by elapsed time t' , with $t' < T$ (T = required processing time) following job introduction; note that $W(R) = T$, so the first passage of $W(t')$ to T is actually the response time. Although differential equations may be obtained for transforms of $W(t')$ under various initial conditions, and hence, implicitly for its distribution, the results are far from being explicit and informative. However, central limit theorems for additive functionals of Markov processes, or for cumulative processes, allow the conclusion that the accumulated work accomplished by fixed time t' on a "long" job is also approximately normally distributed.

Additionally, a normal approximation is shown to be valid for the simple model --and probably for others as well-- when the number of competing terminals becomes large, i.e. under heavy traffic conditions. The quality of the normal approximations for finite job lengths and for a finite number of terminals will be assessed by simulation methods in chapter IV of this thesis.

The differential equations for the mean and moments of the response time of a tagged job requiring a fixed amount of processing time, given that it enters to find an initial number of other jobs being processed, will be derived in chapter II. To remove the condition of initial system state, we will use the steady-state distribution of the number of jobs at the service stage. This will also be explained in the same chapter. A procedure to simulate the response time, given again an initial system state, will be described in chapter III. The empirical response times obtained from simulation will then be considered as stratified random samples. Hence, a method of computing the central moments will be given accordingly.

In chapter V, we will study a bivariate birth and death process model for a computer system having two types of terminals. This model allows relaxation of the independent identically distributed exponential service requirement and terminal think times of the model described by Mitra. [Ref. 2] Under the same conditions as the previous simple model with one job type, we will derive normal approximations for the distribution of the response time of a tagged job requiring t units of processing time. As before, simulation methods will be used to assess the accuracy of normal approximation to the distribution of the response time.

II. MEAN AND MOMENTS FOR A SINGLE JOB TYPE MODEL

A. INTRODUCTION

In this chapter, we consider the birth-death process model with rates in (1.1). Even though the unconditional mean and variance of response time of a Processor-Sharing system may be obtained by deriving the Laplace transform of the equilibrium waiting time distribution [Ref. 1] it is also interesting to develop some results for the conditional expectation of response time of a job requiring T units of processing time, since in the real world one might rather wonder how long a job that requires certain amounts of processing time will be delayed after being submitted to the system.

We will show that the equilibrium mean waiting time in the Processor-Sharing system varies linearly with the service time requirement T , i.e. $E(X) = \lambda T / (\mu (1 - \lambda / \mu))$, as $T \rightarrow \infty$. Thus for arrivals having a service time requirement less than the average, i.e. $T < 1/\mu$, the mean response time is less in the Processor-Sharing system than in the First-Come-First-Served system.

To derive the conditional mean response time, a given tagged job, i.e. a particular job that enters to find $(j-1)$ others waiting for service time, and that requires " T " units of processing time will be considered. Then under Markov assumptions a system of differential equations will be established to allow computation of the conditional mean response time by numerical methods. Other moments may also be obtained by means of solving other systems of differential equations.

E. CONDITION ON REQUIRED TIME AND SYSTEM STATE.

Under the assumptions of a Markov process, i.e. birth and death process, on the number of jobs at the service stage of a system of "N" terminals and a single computer (CPU) with transition rates as in equation (1.1), a system of differential equations for the mean response time may be derived as follows.

Let R refer to the response time of a newly arrived job, and

$$m_j(T) = E[R | X(0) = j, W(R) = T], \quad (2.1)$$

the conditional expectation of the response time, given that the tagged job is initially in the company of $(j-1)$ others, i.e. arrives to find $(j-1)$ jobs present, and requires "work" or processing time equal to T .

Let λ_j and μ_j be as in (1.1). Consider all the possible system changes in $(0, h)$, and subsequently; any of the following mutually exclusively events may occur:

(a) new job arrival, bringing the state to $j+1$, an event of probability $\lambda_j h$,

(b) accompanying job departure and return to think mode, an event of probability $(j-1) \mu (r(j)/j) h$,

(c) no change in accompanying system state but a reduction in remaining tagged-job service of $(r(j)/j) h$, an event of probability $1 - (\lambda_j + (j-1) \mu (r(j)/j)) h$.

All other possible events are of probability $o(h)$ and may be ignored.

The term $r(j)$ used above represents the fraction of time the processor actually spends processing when there are j jobs being processed. The fact that $r(j) < 1$ represents overhead.

Letting $\tilde{\mu}_j = (j-1) \mu (r(j)/j)$, (a), (b) and (c) lead to :

$$m_j(T) = h + m_j(T - (r(j)/j)h) [1 - (\lambda_j + \tilde{\mu}_j)h] \quad (2.2)$$

$$+ \lambda_j h m_{j+1}(T - (r(j)/j)h) + \tilde{\mu}_j h m_{j-1}(T - (r(j)/j)h) + o(h).$$

Subtract $m_j(T - (r(j)/j)h)$ from each side, then divide by h and let $h \rightarrow 0$ to get the differential equations :

$$(r(j)/j) m_j'(T) = 1 - (\lambda_j + \tilde{\mu}_j) m_j(T) \quad (2.3)$$

$$+ \lambda_j m_{j+1}(T) + \tilde{\mu}_j m_{j-1}(T).$$

This is a standard system of linear differential equations, initial conditions are $m_j(0) = 0$ for all j . A solution can be obtained in terms of Laplace transforms, by exponential formulas involving matrices, or, numerically, by use of standard computer codes for the solution of systems of linear differential equations. Simple explicit and comprehensible closed form results do not seem attainable.

C. CONDITION ON REQUIRED TIME

The condition that the tagged job entered to find $(j-1)$ others in the system, i.e. $X(0) = j$, can be removed according to the stationary distribution that corresponds to the system state found by the arriving job. The resulting expression allows the conclusion that the expected response time is "linear" in the required processing time, T . The result here holds for quite general birth-and-death process model, not only for the simple machine-repair setup detailed here. [Ref. 5]

The derivation of linearity of the expected response time is developed, in outline, as follows.

First, observe that the long-run distribution of $X(0)$, i.e. the number of jobs present (including the tagged job) just after the tagged job enters, is

$$q_j = c \pi_{j-1} \lambda_{j-1} = c \pi_j \mu_r(j) \quad , \quad j = 1, 2, \dots, N \quad (2.4)$$

where c is selected so that the q 's sum to one, and for all j 's, $\pi_j = \pi_0 (\lambda_0 \lambda_1 \dots \lambda_{j-1}) / (\mu_1 \mu_2 \dots \mu_j)$ is the stationary distribution (assumed to exist) of the Markov chain $X(t)$, i.e. the number of jobs at the service stage, with rates as in (1.1) with $\mu_j = \mu_r(j)$. The equation (2.4) is intuitively apparent, for the long-run probability that a transition from $j-1$ to j occurs in $(t, t+h)$ is $\pi_{j-1} \lambda_{j-1} h$ as $h \rightarrow 0$ and hence equation (2.4) follows by normalization. A formal proof can be provided based either upon an embedded Markov chain formulation, or upon the theory of additive functionals of a Markov process. [Ref. 6] The distribution $\{q_j\}$ has also been given by Kelly. [Ref. 7]

Next, use equation (2.4) to remove the condition that $X(0) = j$. Put

$$m(T) = E_{X(0)} E[R | X(0), W(R) = T] = \sum_{j=1}^N q_j m_j(T) \quad (2.5)$$

Then in terms of the differential equations (2.3); after multiplying through by $j/r(j)$, one obtains, with initial conditions $m_j(0) = 0$,

$$m'(T) = \sum_{j=1}^N (j/r(j)) q_j + \sum_{j=1}^N q_j (j/r(j)) [-(\lambda_j + \tilde{\mu}_j) m_j(T) + \lambda_j m_{j+1}(T) + \tilde{\mu}_j m_{j-1}(T)]$$

$$m'(T) = \sum_{j=1}^N (j/r(j)) q_j. \quad (2.6)$$

Thus it follows that the long-run conditional expected response time of the processor-sharing system is linear in the processing time requirement T :

$$\begin{aligned} E[R|W(R)=T] &= T \sum_{j=1}^N (j/r(j)) q_j \\ &= TE[X(0)/r(X(0))]. \end{aligned} \quad (2.7)$$

Apparently no such simple form exists for $\text{Var}[R|W(R)=T]$, although Mitra [Ref. 2] has given a formula for a particular case. It will be shown, however, that the above variance is indeed proportional to T if T is large.

D. MOMENTS AND VARIANCE OF RESPONSE TIME

The conditional moments for response time of a job requiring T units of processing time may be obtained by a similar derivation to that used for the expected conditional response time. For example, to find an expression in differential equation form of the second moment, one has to consider all the possible system changes during time period $(0, h)$ as has been done for the mean response time in previous section. If the conditional second moment of response time of a tagged job that requires T units of processing time is:

$$m_j^2(T) = E[R^2 | X(0)=j, W(R)=T], \quad (2.8)$$

then, the following results subsequently occur:

$$m^2_j (T) - m^2_j (T - (\tau(j)/j)h) \quad (2.9)$$

$$= 2m_j (T - (\tau(j)/j)h) - (\lambda_j + \tilde{\mu}_j) m^2_j (T - (\tau(j)/j)h) \\ + \lambda_j m^2_{j+1} (T - (\tau(j)/j)h) + \tilde{\mu}_j m^2_{j-1} (T - (\tau(j)/j)h).$$

As $h \rightarrow 0$ one obtains the differential equations:

$$(\tau(j)/j) [dm^2_j (T)/dT] = 2m_j (T) - (\lambda_j + \tilde{\mu}_j) m^2_j (T) \\ + \lambda_j m^2_{j+1} (T) + \tilde{\mu}_j m^2_{j-1} (T). \quad (2.10)$$

Again, the distribution found in (2.4) can be used to remove the condition that the job enters to find $j-1$ others in the system.

The variance may then be computed by usual formula, i.e.

$$\text{Var}[R(T)] = E[R^2(T)] - [E[R(T)]]^2.$$

Likewise, the differential equations for the third and fourth moments, $m^3_j(T)$ and $m^4_j(T)$, may be obtained by the procedure used to evaluate the first and second moments. The expressions for these moments are as follow:

$$(\tau(j)/j) [dm^3_j (T)/dT] = 3m^2_j (T) - (\lambda_j + \tilde{\mu}_j) m^3_j (T) \\ + \lambda_j m^3_{j+1} (T) + \tilde{\mu}_j m^3_{j-1} (T), \quad (2.11)$$

$$(\tau(j)/j) [dm^4_j (T)/dT] = 4m^3_j (T) - (\lambda_j + \tilde{\mu}_j) m^4_j (T) \\ + \lambda_j m^4_{j+1} (T) + \tilde{\mu}_j m^4_{j-1} (T). \quad (2.12)$$

Once the condition of initial system state is removed using the expression of (2.4) we can compute the third and fourth central moments by expanding the powers in order to calculate them in terms of moments around the origin obtained from solving the system of differential equations described above. The central moments may be expressed in the following forms:

$$E[(R(T) - E[R(T)])^3] = E[R^3(T)] - 3E[R^2(T)]E[R(T)] + 2[E[R(T)]]^3$$

$$E[(R(T) - E[R(T)])^4] = E[R^4(T)] - 4E[R^3(T)]E[R(T)] + 6E[R^2(T)][E[R(T)]]^2 - 3[E[R(T)]]^4.$$

The skewness and kurtosis of response time are then computed as follows:

$$\text{Skewness} = E[(R(T) - E[R(T)])^3] / (\text{Var}[R(T)])^{3/2},$$

$$\text{Kurtosis} = \{E[(R(T) - E[R(T)])^4] / ((\text{Var}[R(T)])^2)\} - 3.$$

E. NUMERICAL RESULTS

The conditional expected response time $m_j(T)$ and the conditional moments $m_j^2(T)$, $m_j^3(T)$ and $m_j^4(T)$ can be computed by solving the differential equations (2.10)-(2.12) using numerical methods, i.e. either linear or Runge-Kutta methods. Once these results are obtained, the condition that the tagged job enters to find $j-1$ others in system is removed. Hence we will obtain the mean and the second, third and fourth moments of response time of a job that requires T units of processing time. These values can then be used to compute the central moments, and eventually allow us to determine the variance, skewness and kurtosis for the distribution of response time.

Table I shows the means, variances, skewnesses and kurtosises of response time of a job requiring T units of processing time for a system of 2 terminals with arrival rate $\lambda = 1$ and service rate $\mu = 2$. We see that as the required work time becomes large the distribution of the response time is moderately close to a symmetric one, since the skewnesses are small. The kurtosis is also approaching zero as the required work time becomes large. This suggests that the distribution of the response time may be approximated by a normal distribution for large required work time.

TABLE I
Numerical Results for 2 Terminals

$$\lambda = 1, \mu = 2$$

<u>Time</u>	<u>Mean</u>	<u>Variance</u>	<u>Skewness</u>	<u>Kurtosis</u>
0.1000	0.1335	0.0020	0.6633	-1.3415
0.2000	0.2668	0.0074	0.7078	-1.3042
0.3000	0.4002	0.0152	0.7053	-1.2428
0.5000	0.6668	0.0358	0.6876	-1.0823
1.0000	1.3325	0.1031	0.5729	-0.6024
2.0000	2.6719	0.2656	0.5118	-0.4626
3.0000	4.0084	0.4569	0.4614	-0.3802
5.0000	6.6815	0.7711	0.2554	-0.1976

III. SIMULATION FOR ONE JOB TYPE MODEL

A. INTRODUCTION

The numerical method of computing the moments of the conditional response time of a tagged job that requires some fixed amount of processing time indicated in the previous chapter is generally sufficiently accurate, especially if carried out by the Runge-Kutta method. However, the distribution of the response time is also of interest. We will use simulation to study this distribution. We will describe a simulation routine for response time of a job requiring a fixed amount of processing time for the model with one job type described in chapters I and II.

The conditional response time of a tagged job that enters to find $j-1$ others initially present in the system, and requires T units of processing time can be simulated for the model if the job submission rate to the processor of each terminal and the processing rate of the processor, i.e. λ and μ , are known. Under Markov assumptions, the number of jobs, including the tagged one, in the Processor-Sharing system which consists of one processor and N terminals is considered as a birth and death process with transition rates $\lambda_j = \lambda(N-j)$ and $\mu_j = \mu(j-1)r(j)/j$, where $r(j)$ is defined as a fraction of time the processor actually spends processing when there are j jobs being processed including the tagged job. Thus the interarrival time and the departure time (work completed) of the jobs in the system are exponentially distributed with parameters λ_j and μ_j respectively. We use the LLRANDCMII package available for the Naval Postgraduate School computer system to generate the two exponential times with rates λ_j and μ_j respectively.

The generated arrival and departure times are compared and the sojourn time in state j is determined as well as the next state of the system and the amount of processing time the tagged job gets from the processor during the sojourn time. We then repeat the procedure until the accumulated processing time for the tagged job meets the requirement of work required to completion.

B. WORK TIME

It turns out to be especially convenient to measure time in terms of the amount of actual work or processing that has been accomplished on the tagged job. Let $C(w)$ denote the number of jobs undergoing service at a moment when exactly w units of processing have been accomplished on the tagged job. We will also assume $r(j) = 1$ for all j . The rate of accretion of clock or response time at work time w is $C(w)$: if $C(w) = 1$ then the tagged job is alone and response (clock) time and work time advance at the same rate, while if $C(w) = 17$ the tagged job is accompanied by 16 others and 17 units of response time accrue for every single work time unit. It follows that the response time for the tagged job requiring T units of processing time is simply

$$R(T) = \int_0^T C(w) dw.$$

The process $\{C(w)\}$ is a birth and death process related to $X(w)$. It has arrival and departure rates $\tilde{\lambda}_j = \lambda_j(N-j)$ and $\tilde{\mu}_j = \mu_j$. All the simulations described in this thesis will be done in work time.

To obtain the conditional expected response time of a tagged job that requires T units of work time as in previous chapter, we remove the condition that the tagged job entered to find $j-1$ others initially present in the system by applying the same steady-state distribution of the number of jobs in system, q , found in (2.4).

C. SIMULATION FOR A 2-TERMINAL SYSTEM

1. Algorithm

We will first describe the simulation by considering a simple computer system which consists of one processor and only two terminals. Each terminal submits jobs to be processed by the processor at rate λ and the processor has a service rate of μ for jobs already present in the system. The service effort is allocated equally to all jobs present in the system at any time. Therefore, if a job requiring T units of work time enters to find the system empty, the arrival rate of the other job will be λ while there will be no departure. Similarly, if the job enters to find the other one already present in the system, there will be no arrival while the service rate will be $\mu r(2)/2$.

To transform the rates into the terms of work time we multiply them by $1/r(1)$ and $2/r(2)$ respectively. Thus, the arrival rate, if any, becomes $\lambda/r(1)$ and the service rate, if any, is simply μ .

Based on the above transition rates an algorithm to perform simulation for the conditional response time and eventually the mean response time of a job requiring T units of processing time will be given as follows.

Algorithm to simulate response time of a job requiring T units of processing time in a 2-terminal system.

Let w_0 = amount of work time remains to accomplish for the tagged job.

c_0 = amount of clock time accumulated towards the response time of the tagged job.

Step 1 : Set $w_0 = T$ and $c_0 = 0$

Step 2 : If the tagged job enters to find the system empty, otherwise go to step 3, generate an exponential time with parameter $\lambda/r(1)$. Call this t' .

- a) If $t' \geq w_0$, set the conditional response time
 $R_1 = c_0 + w_0 / r(1)$.
 STOP
- b) If $t' < w_0$, set :
 $w_0 = w_0 - t'$, $c_0 = c_0 + t' / r(1)$.
 GO TO Step 3.

Step 3 : If the job enters to find another one already present in the system, i.e. $j = 2$, generate an exponential with parameter μ . Call this t'' .

- a) If $t'' \geq w_0$, set the response time
 $R_2 = c_0 + 2w_0 / r(2)$.
 STOP
- b) If $t'' < w_0$, set :
 $w_0 = w_0 - t''$, $c_0 = c_0 + 2t'' / r(2)$.
 GO TO Step 2.

Repeat the above procedure until we obtain the conditional response times for both cases.

To find the mean response time we use the long-run distribution of the number of jobs in system just after the tagged job entered, i.e.

$$q_j = c \pi_{j-1} \lambda_{j-1} = c \pi_j \mu r(j),$$

where $j = 1, 2$ and $q_1 + q_2 = 1$ and π_j is the stationary distribution of the continuous time Markov chain, $\{X(t)\}$.

The mean response time of a job requiring t units of work time is then

$$E[R] = R_1 q_1 + R_2 q_2,$$

where R_1, R_2 are the conditional mean response times, given the tagged job enters to find the system empty and one job already present respectively, generated by the algorithm.

The variance, skewness and kurtosis of the response time may also be obtained by deriving the usual central

moments. These calculations will be detailed in the next section.

2. Numerical Results

The numerical results shown in Table II are obtained from the outputs of a computer program written in FORTRAN .

TABLE II
Simulation results for 2 terminals

$$\lambda = 1, \mu = 2$$

<u>Time</u>	<u>Mean</u>	<u>Std.Dev.</u>	<u>Skewness</u>	<u>Kurtosis</u>
0.1000	0.1339 (.0006)	0.0452	0.6839	-1.4462
0.2000	0.2648 (.0014)	0.0851	0.7550	-1.2457
0.3000	0.4031 (.0027)	0.1229	0.6459	-1.3044
0.5000	0.6688 (.0047)	0.1913	0.6477	-1.1835
1.0000	1.3669 (.0098)	0.3200	0.6017	-1.0297
2.0000	2.6864 (.0157)	0.5051	0.5690	-0.4870
3.0000	4.0128 (.0211)	0.6516	0.5184	-0.2896
5.0000	6.6863 (.0269)	0.8224	0.2974	-0.1623

using the above algorithm to simulate the conditional response time, given the initial system state when the tagged job arrives. The steady state distribution is then used to evaluate the unconditional moments, and measures of skewness and kurtosis for various work time requirements of the job. The values between parentheses below the means are their corresponding standard errors which depend on the number of simulation replications. These outputs are evaluated based on 500 replications of the response time for each initial condition. The $r(j)$'s are all assumed to be unity.

Note that the moments obtained from the simulation agree well with those obtained by solving directly the system of differential equations for the moments of the response times. This fact provides a check for the simulation. The simulated response times show diminishing values of skewness and kurtosis as the required work time becomes large, again suggesting that there may be an increasingly accurate normal approximation to the response time distribution.

D. SIMULATION FOR AN N-TERMINAL SYSTEM

1. Algorithm

Now, consider a more general computer system with one processor and N terminals. As before, each terminal has a submission rate λ to the processor and the processor processes each job to completion with rate μ . The allocation of processing time always follows the method of "Processor-Sharing".

The simulation for response time of a job requiring T units of processing time will be done under the condition that when the job arrives there are j terminals active, i.e. j jobs, including the tagged one that just arrived, are being served by the CPU. Here, j can be $1, 2, \dots, N$.

Under Processor-Sharing scheduling, if j jobs are present then in a short time interval of length h the tagged job gets $hr(j)/j$ units of work done. Thus, if $W(t)$ is the amount of work done on tagged job by the time it has been in the system for t units of clock time, and if the number of jobs in system during this time t is j , then $W(t) = tr(j)/j$.

As long as $W(t)$ is less than the required amount of work T for the tagged job, we will have to accumulate the amount of work done computed according to the number of jobs in system at that time. The conditional response time will be the clock time t for which the accumulated completed work time $\int_0^t W(t') dt'$ is equal to T .

Again it is convenient to measure time in terms of work time. The work time process, $C(w)$, described in the previous section is a birth-death process with rates $\lambda(N-j)j = \lambda'_j$ and $\mu_j = \mu(j-1)$, $1 \leq j \leq N$. The response time is simply $R(T) = \int_0^T C(w) dw$, if $r(j) = 1$.

To simulate the work time process, we generate two exponential times with parameters λ'_j and μ'_j respectively. The minimum of the two will indicate which event, arrival or departure, occurs first. If an arrival occurs first and the accumulated completed work time is still less than the requirement, T , the number of jobs being processed by the CPU, i.e. system state, will be $j+1$. Likewise, when a departure occurs first the number of jobs for next computation will be $j-1$.

The above observations allow us to construct an algorithm to perform a simulation for conditional response time and lead eventually to the estimation of statistics for the response time of a job requiring T units of processing time as follows.

Algorithm to simulate response time of a job requiring T units of processing time for an N -terminal system, given when the tagged job begins processing there are $(j-1)$ jobs also being processed.

Let w_0 = amount of work that remains to accomplish for the tagged job.

c_0 = amount of clock time accumulated towards the response time of the tagged job.

Step 1 : Set $w_0 = T$ and $c_0 = 0$.

Step 2 : If $j=1$, i.e. the job enters to find the system empty, otherwise GO TO step 3, generate an exponential time with parameter λ_1 . Call this t' .

a) If $t' \geq w_0$, set the response time

$$R_1 = c_0 + w_0/r(1)$$

STOP

b) If $t' < w_0$, set

$$w_0 = w_0 - t'$$

$$c_0 = c_0 + t'/r(1)$$

$$j = 2$$

GO TO step 3.

Step 3 : If $j = 2, 3, \dots, N-1$, generate two exponential times with parameters λ_j and μ_j . Call them t' and t'' respectively.

a) If $\min(t', t'') \geq w_0$, set the response time

$$R_j = c_0 + w_0 j/r(j)$$

STOP.

b) If $\min(t', t'') < w_0$:

i) If $\min(t', t'') = t'$, set

$$w_0 = w_0 - t'$$

$$c_0 = c_0 + t' j/r(j)$$

$$j = j + 1$$

GO TO step 2 or 3 or 4 according to j .

ii) If $\min(t', t'') = t''$, set

$$w_0 = w_0 - t''$$

$$c_0 = c_0 + t'' j/r(j)$$

$$j = j - 1$$

GO TO step 2 or 3 or 4 according to j .

Step 4 : If $j = N$, i.e. all terminals become active, generate an exponential time with parameter μ'_N . Call this t'' .

a) If $t'' \geq w_0$, set the response time

$$R_N = c_0 + w_0 N / r(N)$$

STOP

b) If $t'' < w_0$, set

$$w_0 = w_0 - t''$$

$$c_0 = c_0 + t'' N / r(N)$$

$$j = N - 1$$

GO TO step 3.

Each run of the algorithm for fixed initial j gives a realization of the conditional response time of the tagged job given there are $(j-1)$ other jobs in the system when the tagged job arrives for processing.

2. Moments of Response Time

The simulation based on the above algorithm provides a batch of conditional response times of a tagged job for each initial system state.

Suppose we simulate a batch of size K for each conditional response time at initial system state j , for all j 's.

Let R_{jk} be the k -th realized conditional response time given the initial condition is j ; (that is, the tagged job arrives when $(j-1)$ other jobs are being processed), for $j = 1, 2, \dots, N$ and $k = 1, 2, \dots, K$.

Mathematically, we can use the averages of $(R_{jk})^i$, $i = 1, 2, 3, 4$, over each batch to compute empirical first, second, third and fourth conditional moments given initial condition j respectively. To compute the unconditional empirical moments for response time of a tagged job that requires T units of processing time, we can multiply the j -th conditional empirical moment by the steady-state

probability q_j given by (2.4) and sum over all $j = 1, 2, \dots, N$. The empirical mean, variance, third and fourth central moments then are computed by power expansions, i.e.

$$\begin{aligned} \widehat{E}[R(T)] &= \sum_{j=1}^N \bar{R}_j q_j = \bar{R} \\ \widehat{\text{Var}}[R(T)] &= \sum_{j=1}^N \bar{R}_j^2 q_j - (\bar{R})^2 \\ \widehat{E}[(R(T)-R)^3] &= \sum_{j=1}^N \bar{R}_j^3 q_j - 3\bar{R} \sum_{j=1}^N \bar{R}_j^2 q_j + 2(\bar{R})^3 \\ \widehat{E}[(R(T)-R)^4] &= \sum_{j=1}^N \bar{R}_j^4 q_j - 4\bar{R} \sum_{j=1}^N \bar{R}_j^3 q_j + 6(\bar{R})^2 \sum_{j=1}^N \bar{R}_j^2 q_j - 3(\bar{R})^4, \end{aligned} \quad (3.1)$$

where $\bar{R}_j^i = (\sum_{k=1}^K R_{jk}) / K$, for $i = 1, 2, 3, 4$.

However, in practice the above procedure can be numerically unstable since the averages of the second, third and fourth moments over the batch may be very large numbers. Hence, when we add or subtract these numbers to compute the central moments, it is possible that the computation produces round-off errors which may be substantial. Therefore, we would rather rewrite the central moments in terms of conditional expectations, i.e.

$$E[(R-E(R))^i] = E[E[(R-E(R))^i | X(0)]], \quad i=1, 2, 3, 4. \quad (3.2)$$

where $X(0)$ is the number of jobs requesting processing by the CPU, including the tagged job, when the tagged job arrives. This allows us to obtain the central moments in a more numerically stable manner. Further details will be shown for the computations of variance, skewness and kurtosis of the response time.

3. Computation of Simulated Variance

Let $\bar{R}_j = (\sum_{k=1}^K R_{jk}) / K$ and $\bar{R} = \sum_{j=1}^N \bar{R}_j q_j$, then the estimated variance

$$\begin{aligned} \widehat{\text{Var}}[R] &= \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N (R_{jk} - \bar{R})^2 q_j \\ &= \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N (R_{jk} - \bar{R}_j + \bar{R}_j - \bar{R})^2 q_j \\ &= \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N [(R_{jk} - \bar{R}_j)^2 + 2(R_{jk} - \bar{R}_j)(\bar{R}_j - \bar{R}) + (\bar{R}_j - \bar{R})^2] q_j. \end{aligned}$$

Since $(\sum_{k=1}^K R_{jk} - \bar{R}_j) / K = 0$, the estimated variance becomes

$$\widehat{\text{Var}}[R] = \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N (R_{jk} - \bar{R}_j)^2 q_j + \sum_{j=1}^N (\bar{R}_j - \bar{R})^2 q_j.$$

This is the sampling version of the formula

$$\text{Var}[R] = E[\text{Var}(R|X(0))] + \text{Var}[E(R|X(0))],$$

which is a known general result that applies to any random variable, R , that also depends upon another random variable, namely $X(0)$. The first component represents the overall variability of R for a fixed value of $X(0)$, and the second component represents the variability of R due to the variability in $X(0)$.

4. Computation of Simulated Skewness

To compute the estimated skewness we first compute the third estimated central moment of the response time by rewriting it in a conditional expectation form as we have done for the computation of the variance. The third central moment is derived as shown below.

$$\widehat{E}[(R_{jk} - \bar{R})^3] = \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N (R_{jk} - \bar{R})^3 q_j$$

$$\begin{aligned} \widehat{E}[(R_{jk} - \bar{R})^3] &= \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N (R_{jk} - \bar{R}_j + \bar{R}_j - \bar{R})^3 q_j \\ &= \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N [(R_{jk} - \bar{R}_j)^3 + 3(R_{jk} - \bar{R}_j)^2 (\bar{R}_j - \bar{R}) \\ &\quad + 3(R_{jk} - \bar{R}_j) (\bar{R}_j - \bar{R})^2 + (\bar{R}_j - \bar{R})^3] q_j. \end{aligned}$$

We can remove the 3-rd component on the right hand side, since $(\sum_{k=1}^K R_{jk} - \bar{R}_j) / K = 0$. Thus

$$\begin{aligned} \widehat{E}[(R_{jk} - \bar{R})^3] &= \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N (R_{jk} - \bar{R}_j)^3 q_j + 3 \sum_{k=1}^K \sum_{j=1}^N (R_{jk} - \bar{R}_j)^2 (\bar{R}_j - \bar{R}) q_j \\ &\quad + \sum_{j=1}^N (\bar{R}_j - \bar{R})^3 q_j. \end{aligned}$$

The measure of skewness of response time is then

$$\widehat{\text{Skewness}}[R_{jk}] = \frac{E[(R_{jk} - \bar{R})^3]}{(\text{Var}[R_{jk}])^{3/2}}.$$

5. Computation of Simulated Kurtosis

We start by rewriting the expression for the estimated fourth central moment as follows:

$$\begin{aligned} \widehat{E}[(R_{jk} - \bar{R})^4] &= \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N (R_{jk} - \bar{R})^4 q_j \\ &= \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N (R_{jk} - \bar{R}_j + \bar{R}_j - \bar{R})^4 q_j \\ &= \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N [(R_{jk} - \bar{R}_j)^4 + 4(R_{jk} - \bar{R}_j)^3 (\bar{R}_j - \bar{R}) \\ &\quad + 6(R_{jk} - \bar{R}_j)^2 (\bar{R}_j - \bar{R})^2 + 4(R_{jk} - \bar{R}_j) (\bar{R}_j - \bar{R})^3 \\ &\quad + (\bar{R}_j - \bar{R})^4] q_j. \end{aligned}$$

We then simplify, as before, by removing the 4-th term on the right hand side. Hence,

$$\begin{aligned} \widehat{E}[(R_{jk} - \bar{R})^4] &= \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N (R_{jk} - \bar{R}_j)^4 q_j + 4 \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N (R_{jk} - \bar{R}_j)^3 (\bar{R}_j - \bar{R}) q_j \\ &\quad + 6 \frac{1}{K} \sum_{k=1}^K \sum_{j=1}^N (R_{jk} - \bar{R}_j)^2 (\bar{R}_j - \bar{R})^2 q_j + \sum_{j=1}^N (\bar{R}_j - \bar{R})^4 q_j. \end{aligned}$$

Therefore, the measure of kurtosis is

$$\widehat{\text{Kurtosis}}[R_{jk}] = \frac{\widehat{E}[(R_{jk} - \bar{R})^4]}{(\widehat{\text{Var}}[R_{jk}])^2}.$$

Since for the normal distribution the kurtosis has the value 3, we then subtract 3 from the kurtosis computed above. So the new value of kurtosis will be 0 when the distribution of response time has the normal degree of kurtosis.

6. Standard Error of the Mean Response Time

In order to assess the accuracy of the simulated average response time, we may compute a standard error for the mean response time of a tagged job that requires T units of work time from our batches of simulated conditional response times.

We have derived previously that the mean response time is

$$\widehat{E}[R_{jk}] = \bar{R} = \sum_{j=1}^N \left(\frac{1}{K} \sum_{k=1}^K R_{jk} \right) q_j.$$

Now, we apply a property of the variance function by considering q and K fixed constants and noting that the R_{jk} 's are obtained from independent realizations. This allows us to write the estimated variance of the expected response time as follows :

$$\widehat{\text{Var}}[\widehat{E}[R_{jk}]] = \frac{1}{N} \sum_{j=1}^N \left(\frac{1}{K} \sum_{k=1}^K (R_{jk} - \bar{R}_j)^2 \right) q_j^2$$

To obtain the standard error of the mean response time we simply take the square root of $\widehat{\text{Var}}[\widehat{E}[R_{jk}]]$.

7. Numerical Results

Tables III, IV and V show the outputs from a simulation program based on the previously described algorithm. The number of replications for each initial condition is

TABLE III
Numerical results from simulation

$N = 10, \lambda = 15, \mu = 100$

<u>Time</u>	<u>Mean</u>	<u>Variance</u>	<u>Skewness</u>	<u>Kurtosis</u>
0.0100	.040368 (.000177)	.000295	.119865	-.715162
0.0250	.100510 (.000503)	.001405	.088115	-.712345
0.0375	.151623 (.000733)	.002508	.015783	-.587745
0.0500	.201580 (.000952)	.003865	-.046461	-.572722
0.0625	.251340 (.001092)	.004953	-.012371	-.510326
0.1000	.402916 (.001510)	.008842	-.092651	-.378387
0.1500	.607487 (.001935)	.013806	-.061429	-.186003
0.2000	.806743 (.002303)	.019704	-.023100	-.130595

TABLE IV
Numerical results from simulation

$N = 10, \lambda = 25, \mu = 100$

<u>Time</u>	<u>Mean</u>	<u>Variance</u>	<u>Skewness</u>	<u>Kurtosis</u>
0.0100	.060567 (.000192)	.000253	-.773193	.685650
0.0250	.150692 (.000467)	.000992	-.717666	.637294
0.3750	.226364 (.000652)	.001775	-.697005	.532480
0.0500	.302138 (.000791)	.002466	-.662308	.538648
0.0625	.378130 (.000925)	.003266	-.649630	.527906
0.1000	.603584 (.001228)	.005601	-.538279	.509166

500. The numbers below the means are their standard errors. We can see that the results in the tables indicate a somewhat symmetric distribution for the response time (skewnesses are very small), and the kurtosis values do not strongly indicate non-normality (they decrease towards zero), especially when the processing time requirement becomes large. The kurtosis values in table III seem to indicate smaller tails than those in table IV, and the skewness values indicate that we have a more symmetric distribution in the case $\lambda = 15$ than when $\lambda = 25$ for an equal $\mu = 100$. In the next chapter some normal approximations to the response time distribution will be described.

TABLE V
Numerical results from simulation

$N = 25, \lambda = 5, \mu = 100$

<u>Time</u>	<u>Mean</u>	<u>Variance</u>	<u>Skewness</u>	<u>Kurtosis</u>
0.0100	.063016 (.000202)	.001011	.346171	-.463719
0.0250	.158884 (.000637)	.005204	.213798	-.555656
0.0375	.239094 (.001008)	.010175	.170748	-.541305
0.0500	.316021 (.001354)	.015985	.159817	-.565249
0.0625	.394780 (.001686)	.021523	.117365	-.496104

$N = 25, \lambda = 10, \mu = 100$

<u>Time</u>	<u>Mean</u>	<u>Variance</u>	<u>Skewness</u>	<u>Kurtosis</u>
0.0100	.149847 (.000252)	.000659	-.527812	.471482
0.0250	.375932 (.000588)	.002529	-.644450	.428531
0.0375	.562173 (.000834)	.004470	-.667446	.366822
0.0500	.750556 (.001020)	.006467	-.641420	.314407
0.0625	.938072 (.001165)	.008259	-.593361	.275313

IV. NORMAL APPROXIMATION FOR RESPONSE TIME

A. INTRODUCTION

The Markov assumptions we make on the processor-sharing system allow us to infer that the distribution of the response time of a tagged job that requires T units of processing time may be approximated by the normal distribution when T is large and/or when the system is in heavy traffic.

Two methods are used to argue the approximate normality of the distribution of response time. One is based on the Central Limit Theorem for additive functionals of a birth and death process, and the other follows from a heavy-traffic diffusion approximation of the birth and death process. The formulas to compute the approximate mean and variance will be described. More details concerning the analytic form of the approximations are given by Gaver, Jacobs and Latouche. [Ref. 3] The approximations will be compared with the results from simulation to study their accuracy.

As mentioned before, the Central Limit Theorem for additive functionals of Markov processes allows the conclusion that the accumulated work accomplished in time t' of a job requiring a large amount of processing time, T , is approximately normally distributed. This in turn allows the conclusion that the corresponding response time is also approximately normally distributed. Hence, we will start by considering $W(t')$, the total work expended by the computer on the tagged job by clock time t' after its arrival, given that the tagged job requires exactly T time-units of work for completion.

It is observed that if when a job arrives there are $X(0) = j$ customers, including the new arrival, present in the system for processing, then:

$$W(t') = \int_0^{t'} (\tau(X_c(u))/X_c(u)) du, \quad X_c(0) = j \geq 1, \quad (4.1)$$

where $X_c(t)$ is the number of jobs at the service stage at clock time t .

From this an appropriate central limit theorem for $W(t)$ can be established by using results for finite birth-and-death models [Ref. 8] or by making use of the theory of cumulative processes. [Ref. 9] We note here that the latter development of the central limit theorem is adaptable to models more general than the simple birth-and-death process.

In the case in which the system is in heavy traffic we can approximate $\{X_c(t); t \geq 0\}$ by an Ornstein-Uhlenbeck process, see e.g. Iglehart. [Ref. 10] The process $W(t')$ is then approximated by an integral of an Ornstein-Uhlenbeck process. A normal approximation for the response time distribution then follows.

After deriving some formulas for the above two methods of approximation, we will make comparison for goodness of fit of those approximations to simulation data.

B. APPROXIMATION BY CENTRAL LIMIT THEOREM

1. A Central Limit Theorem for $W(t)$

Under Markov assumptions throughout the processor-sharing system, the number of jobs being at service stage at clock time t , $X_c(t)$, is a finite ergodic stationary time reversible Markov chain.

If we define a function $f(X(t)) = \tau(X(t))/X(t)$, then as outlined in Keilson [Ref. 8] the process in equation (4.1) may be proven in a variety of ways to be asymptotically normal in distribution for large t . Hence, the accumulated accomplished work-time at time t , $W(t)$, satisfies a central limit theorem.

In order to apply the central limit theorem derived by Keilson we will have to redefine the infinitesimal generator according to the number of jobs in system without including the tagged one.

The relevant generator is now

$$X'(t) = j \rightarrow X'(t+h) = j+1 : \lambda_{j+1} h + o(h) \quad (4.2)$$

$$\rightarrow X'(t+h) = j-1 : \mu_{j+1} h j / (j+1) + o(h)$$

$$\rightarrow X'(t+h) = j : 1 - (\lambda_{j+1} + \mu_{j+1} j / (j+1)) h + o(h),$$

for $j = 0, 1, \dots, N' = N-1$.

The process in equation (4.1) then becomes

$$W(t) = \int_0^t f(X'(u)) du, \quad (4.3)$$

where $f(X'(u)) = r(X'(u)+1) / (X'(u)+1)$.

We can now express a statement for the central limit theorem as :

$$\frac{W(t) - \xi t}{\sigma \sqrt{t}} \rightarrow N(0,1), \text{ as } t \rightarrow \infty \quad (4.4)$$

where the constants ξ and σ^2 are such that

$$\xi = \sum_{j=0}^{N'} f(j) \pi_j' \quad (4.5)$$

$$\sigma^2 = 2[f(0), f(1), \dots, f(N')] \begin{bmatrix} \pi_0' & & & \\ & \pi_1' & & \\ & & \ddots & \\ & & & \pi_{N'}' \end{bmatrix} \begin{bmatrix} f(0) \\ f(1) \\ \vdots \\ f(N') \end{bmatrix} \quad (4.6)$$

with \underline{z} being a matrix defined below

$$\underline{z} = ([I - \underline{A} + \underline{L}]^{-1} - \underline{L}) / \gamma, \quad (4.7)$$

I is an identity matrix, and \underline{L} is an N by N matrix whose rows are steady-state probabilities, $\{\pi'_j\}$, of the birth and death process with rates in (4.2), i.e.

$$\underline{L} = \begin{bmatrix} \pi'_0 & \pi'_1 & \dots & \pi'_{N'} \\ \pi'_0 & \pi'_1 & \dots & \pi'_{N'} \\ \vdots & & & \\ \pi'_0 & \pi'_1 & \dots & \pi'_{N'} \end{bmatrix} \quad (4.8)$$

and \underline{A} is a matrix defined as follows :

$$A_{0,1} = \lambda_1 / \gamma, \quad A_{0,0} = 1 - \lambda_1 / \gamma$$

$$A_{j,j+1} = \lambda_{j+1} / \gamma, \quad A_{j,j-1} = \mu_{j+1} (j / (j+1)) / \gamma, \quad A_{j,j} = 1 - \nu_j / \gamma \quad (4.9)$$

$$\nu_j = \lambda_{j+1} + \mu_{j+1} (j / (j+1)), \quad \gamma = \max_j \nu_j$$

$$(\nu_0 = \lambda_1, \nu_{N'} = \mu_{N'} (N' / N)), \quad (\text{again with rates in (4.2)}).$$

2. A Central Limit Theorem for response time, R(T)

The accumulated accomplished work-time at clock time t' of a tagged job requiring T units of processing time, $W(t')$, increases in random straight-line segments from $W(0) = 0$ until $W(t') = T$. The value of t' here will be the response time of the job, $R(T)$. It is the first-passage time to the required work time. So we can state the following:

$$P(W(t') < T) = P(R(T) > t'). \quad (4.10)$$

From equations (4.4) and (4.10), we can derive, by the same standard argument of renewal theory as given in Karlin and Taylor [Ref. 11] pp. 208-209, the following statement of a central limit theorem for response time, $R(T)$, as T approaches infinity, i.e.

$$\frac{R(T) - (T/\zeta)}{\sqrt{T\sigma^2/\zeta^3}} \rightarrow N(0,1), \text{ as } T \rightarrow \infty. \quad (4.11)$$

C. APPROXIMATION BY HEAVY TRAFFIC ANALYSIS

Now, we consider the processor-sharing system as a machine repair model in which $\lambda_j = \lambda(N-j)$, $\mu_j = \mu$, and $r(j) = 1$ for all j 's. Let N the number of terminals be large and the traffic intensity ρ , which is defined as the ratio of the expected service time to N times the expected inter-arrival time, be a fixed value less than one, i.e.

$$\rho = \mu / (N\lambda).$$

Under the above conditions, when a job requiring T units of processing time enters the system it will be processed in the company of many others. This indicates a system with heavy traffic situation. As mentioned earlier in the introduction section we may apply the properties of the limiting diffusion process developed by Iglehart. [Ref. 10]

Therefore, if $X_c(t)$ is the number of jobs at the processing stage at clock time t , then $X_c(t)$ can be approximated by a diffusion process as follows:

$$X_c(t) = Na(t) + \sqrt{N} \cdot Y(t), \quad (4.12)$$

where $a(t)$ is a deterministic function of time and when t approaches infinity it has a finite limit, i.e. $a(\infty) = 1-\rho$, and $\{Y(t)\}$ is, for the present model, a particular Ornstein-Uhlenbeck process.

The accumulated completed work-time by fixed time t' of a tagged job requiring T units of processing time in equation (4.1) becomes

$$W(t') = \int_0^{t'} du / X_c(u) = \int_0^{t'} du / (Na(u) + \sqrt{N} \cdot Y(u)). \quad (4.13)$$

Next, we apply the approximation and expand the expression in (4.13) to second order terms in N ; then assume that the tagged job arrives when the system is in steady state so that we can use the finite limit of $a(\infty) = 1-\rho$ in place of $a(u)$. The approximation is now:

$$W(t') = \int_0^{t'} du / (N(1-\rho)) - (\sqrt{N} / [N(1-\rho)]^2) \int_0^{t'} Y(u) du, \quad (4.14)$$

for $0 \leq t' \leq t$.

Hence, the expectation of accumulated amount of work-time completed on the tagged job is approximately $t' / (N(1-\rho))$, and the actual distribution of total work done is also approximately Gaussian (integral of an Ornstein-Uhlenbeck process), where the Gaussian property results from the assumption of many accompanying jobs, and not necessarily because the tagged job requires a long processing time.

As t' approaches infinity, we can evaluate (4.14) and show that $E[\int_0^{t'} Y(u) du] = 0$, and $\text{Var}[\int_0^{t'} Y(u) du] = (2\mu / (N\lambda^2)) t'$. So the normal approximation to accumulated completed work time, $W(t')$, has the parameters:

$$\bar{z} = 1 / (N(1-\rho)), \quad \sigma^2 = 2\mu / (\lambda^2 [N(1-\rho)]^2).$$

By reasoning the same way as to obtain a central limit theorem for response time from a known approximated normal distribution of accumulated completed work-time for a tagged job requiring T units of processing time, we can derive a normal approximation for the distribution of response time as having the parameters as follow:

$$E[R(T)] = N(1-\rho)T, \quad (4.15)$$

$$\text{Var}[R(T)] = 2\mu T / (\lambda^2 N(1-\rho)). \quad (4.16)$$

We can also improve the value of the variance by using the formula below:

$$\text{Var}[R(T)] = [2\mu T / (\lambda s)] \delta, \quad (4.17)$$

where $s = \lambda N - \mu$ and $\delta = 1 - [(1 - e^{-sT}) / (sT)]$.

D. COMPARISON TO SIMULATION DATA

We will consider some particular simulation results from a system consisting of 10 terminals and one processor in the cases of light traffic and heavy traffic to make comparison to normal approximations described in previous section.

First we can use the measures of skewness and kurtosis of the response time resulting from simulation data to roughly assess the degree of normality or non-normality. We know that if a distribution is symmetric its skewness will be zero and for the normal distribution its kurtosis has the value 3. However, even if a distribution of the simulated

response time has the measures of skewness and kurtosis close to those of the normal distribution, it does not imply that the distribution is necessarily normal. It only suggests that a normal distribution may be a reasonable approximation.

Secondly, we will compute the empirical distribution of the simulation data at various quantiles of the normal distribution whose mean and variance are approximated by either central limit theorem or heavy traffic analysis (limiting diffusion). For example, to compare the one-tenth quantile of the approximated normal distributions to the simulated response time at the initial system state j , we first compute

$$R_{.10} = \text{Approx. Mean} + (\text{Approx. Std. Dev.}) (Z_{.10})$$

where $Z_{.10}$ is the (1/10)th quantile of the standard normal distribution. We then determine the conditional relative frequency of the simulated response time, given the initial system state is j , as

$$r.f._j(R_{.10}) = \frac{\text{No. of simulated response time} \leq R_{.10}}{\text{No. of replications}}$$

We then use the distribution q_j in (2.4) to remove the initial state condition. Hence, we obtain the one-tenth quantile of the simulated response time (corresponded to the normal approximations). We may also compute the standard error of the estimated quantile by taking the square root of

$$\text{Var}(\hat{p}) = \left(\sum_{j=1}^N q_j^2 \hat{p}_j (1 - \hat{p}_j) \right) / K,$$

where $\hat{p}_j = r.f._j(R_{.10})$ and $K =$ number of replications used in simulation.

Now consider a system of 10 terminals with arrival rate $\lambda = 15$ and service rate $\mu = 100$. The expected number of active terminals is, therefore,

$$E[X(t)] = 10(1 - (100/150)) = 3.3333 ,$$

which indicates a light traffic situation.

TABLE VI
Comparison of Simulation Data to Normal Quantiles

$$N = 10, \lambda = 15, \mu = 100$$

<u>Time</u>	<u>Normal</u>	<u>.10</u>	<u>.25</u>	<u>.50</u>	<u>.75</u>	<u>.90</u>	<u>.95</u>	<u>.99</u>
0.0100	CLT	0.0	.123	.503	.887	.995	1.0	1.0
	HTA I	0.0	0.0	.361	.948	1.0	1.0	1.0
	HTA II	0.0	.102	.361	.677	.909	.972	.999
0.0250	CLT	.042	.211	.487	.806	.967	.994	1.0
	HTA I	0.0	.012	.334	.832	.996	1.0	1.0
	HTA II	0.0	.094	.334	.668	.914	.977	1.0
0.0500	CLT	.083	.232	.486	.766	.938	.977	.999
	HTA I	0.0	.032	.300	.737	.970	.998	1.0
	HTA II	0.0	.067	.300	.649	.916	.973	1.0
0.0750	CLT	.082	.241	.494	.766	.922	.967	.997
	HTA I	0.0	.033	.266	.688	.948	.991	1.0
	HTA II	.001	.049	.266	.630	.900	.969	1.0
0.1000	CLT	.097	.243	.483	.761	.919	.969	.998
	HTA I	.000	.027	.238	.647	.934	.987	1.0
	HTA II	.001	.037	.238	.603	.893	.972	1.0

TABLE VII

Comparison of Simulation Data to Normal Quantiles

$$N = 10, \lambda = 25, \mu = 100$$

<u>Time</u>	<u>Normal</u>	<u>.10</u>	<u>.25</u>	<u>.50</u>	<u>.75</u>	<u>.90</u>	<u>.95</u>	<u>.99</u>
0.0100	CLT	.042	.153	.447	.854	.997	1.0	1.0
	HTA I	.048	.158	.434	.826	.990	1.0	1.0
	HTA II	.106	.224	.434	.721	.911	.975	1.0
0.0250	CLT	.077	.191	.453	.804	.975	.998	1.0
	HTA I	.084	.192	.438	.774	.955	.993	1.0
	HTA II	.104	.216	.438	.729	.917	.976	1.0
0.0500	CLT	.092	.215	.454	.769	.951	.990	1.0
	HTA I	.095	.212	.435	.728	.927	.979	1.0
	HTA II	.108	.224	.435	.707	.909	.965	.998
0.0750	CLT	.093	.198	.447	.754	.936	.983	1.0
	HTA I	.095	.193	.419	.713	.908	.967	.998
	HTA II	.102	.200	.419	.701	.892	.959	.996
0.1000	CLT	.101	.236	.474	.756	.933	.976	.999
	HTA I	.102	.226	.448	.711	.898	.958	.997
	HTA II	.106	.232	.448	.704	.891	.953	.996

Table III shows the values of skewness and normal kurtosis of the response time of a tagged job for various work-time requirements. Those values are quite small especially for large work-time requirements. It indicates the possibility of a successful normal approximation.

Table VI shows how the simulated response time of a job requiring some processing time units is distributed

comparing to the various quantiles (.10, .25, .50, .75, .90, .95, .99) of the normal distributions approximated by central limit theorem and the two heavy traffic (limiting diffusion) approaches (HTA I is (4.16), HTA II is (4.17)).

We see that in this situation the limiting diffusion approximation for normality of the distribution of response time is not so good an approach even for large amount of processing time requirement. This is easily anticipated since the system has only an expected proportion of one-third of its terminals active. However, the central limit theorem seems to work pretty well when the required work time becomes very large.

Let's consider a new system with arrival rate $\lambda = 25$ and the service rate $\mu = 100$, with $N = 10$ again. The expected number of active terminals is now

$$E[X(t)] = 10 (1 - (100/250)) = 6,$$

which indicates a moderately heavy traffic in the system.

Table IV provides us some good feelings about the measures of skewnesses and normal kurtosises not being too far from those of actual normal distribution. Now, observe the table VII which shows the distribution of the simulated response time of a tagged job for various work time requirements comparing to the various quantiles of the normal distributions approximated by central limit theorem and limiting diffusion approaches.

We see that the limiting diffusion methods, in this case, work pretty well even for small work-time requirements. The one with second formula (4.17) of approximating the variance works better than the other, because it provides a smaller standard deviation. The central limit theorem approximation still works very well when the work time requirement becomes large.

TABLE VIII

Comparison of Simulation Data to Normal Quantiles

$N = 25, \lambda = 10, \mu = 100$

<u>Time</u>	<u>Normal</u>	<u>.10</u>	<u>.25</u>	<u>.50</u>	<u>.75</u>	<u>.90</u>	<u>.95</u>	<u>.99</u>
0.0100	CLT	.030	.150	.456	.859	.987	.999	1.0
	HTA I	.047	.161	.456	.845	.980	.998	1.0
	HTA II	.109	.233	.456	.727	.908	.965	.997
0.0250	CLT	.070	.192	.460	.799	.961	.992	1.0
	HTA I	.082	.204	.460	.780	.948	.986	1.0
	HTA II	.112	.233	.460	.730	.910	.967	.997
0.0375	CLT	.087	.212	.470	.784	.948	.986	1.0
	HTA I	.097	.226	.470	.763	.934	.978	.999
	HTA II	.115	.243	.470	.731	.910	.964	.997
0.0500	CLT	.089	.210	.462	.778	.940	.982	.999
	HTA I	.099	.223	.462	.760	.927	.976	.999
	HTA II	.111	.236	.462	.734	.910	.963	.997
0.0625	CLT	.090	.220	.467	.772	.933	.977	.999
	HTA I	.100	.233	.467	.751	.917	.968	.998
	HTA II	.112	.244	.467	.727	.903	.960	.997

$N = 40, \lambda = 10, \mu = 100$

<u>Time</u>	<u>Normal</u>	<u>.10</u>	<u>.25</u>	<u>.50</u>	<u>.75</u>	<u>.90</u>	<u>.95</u>	<u>.99</u>
0.0100	CLT	.072	.198	.477	.789	.954	.988	1.0
	HTA I	.076	.203	.477	.784	.949	.986	1.0
	HTA II	.110	.243	.477	.739	.902	.961	.997
0.0375	CLT	.094	.224	.477	.749	.917	.966	.997
	HTA I	.098	.227	.477	.743	.912	.963	.996
	HTA II	.106	.236	.477	.736	.900	.958	.995
0.0625	CLT	.097	.232	.482	.748	.917	.965	.997
	HTA I	.100	.236	.482	.744	.909	.962	.997
	HTA II	.105	.241	.482	.737	.903	.957	.995

The above results for the two situations, one light traffic and another moderately heavy traffic, lead to the confirmation that for a tagged job requiring large amount of processing time the distribution of response time approaches, asymptotically, normality. In heavy traffic case the limiting diffusion approximations seem to work better than the central limit theorem approach. They do not work well at all when the system has the terminals active less than a half of its full capacity.

To conclude this chapter we show some more results in comparing the approximated normal quantiles to the values of simulated response times in table VIII for different (larger) number of terminals and various transitional rates. We can see easily that as the number of terminals is large the distribution of simulated response time approaches the approximated normals more rapidly. The heavy traffic approximation seems to work better, in these cases, than the central limit theorem approach, especially for small required work-time. This corresponds to our observation that the central limit theorem requires large work-time requirement to be a good approximation, while the only condition that the heavy traffic approximations require is to have a certain amount of jobs waiting to be served at any instant. Finally, table IX shows the mean and standard deviation of the response time computed by the central limit theorem and heavy traffic approximations for all the cases that we have been studying the comparison of the quantiles.

TABLE IX

Approx. Mean and Std. Dev. for One-Type Model

$N = 10, \lambda = 15, \mu = 100$

<u>Time</u>	<u>CLT</u>		<u>HTA I</u>		<u>HTA II</u>	
	<u>Mean</u>	<u>Std.Dev</u>	<u>Mean</u>	<u>Std.Dev.</u>	<u>Mean</u>	<u>Std.Dev.</u>
0.0100	.0404	.0323	.0333	.0516	.0333	.0238
0.0250	.1010	.0510	.0833	.0816	.0833	.0535
0.0500	.2019	.0722	.1667	.1155	.1667	.0919
0.0750	.3029	.0884	.2500	.1414	.2500	.1216
0.1000	.4038	.1020	.3333	.1633	.3333	.1462

$N = 10, \lambda = 25, \mu = 100$

0.0100	.0605	.0245	.0600	.0231	.0600	.0160
0.0250	.1513	.0388	.1500	.0365	.1500	.0314
0.0500	.3027	.0548	.3000	.0516	.3000	.0481
0.0750	.4540	.0672	.4500	.0632	.4500	.0604
0.1000	.6053	.0776	.6000	.0730	.6000	.0706

$N = 25, \lambda = 10, \mu = 100$

0.0100	.1500	.0390	.1500	.0365	.1500	.0254
0.0250	.3750	.0617	.3750	.0577	.3750	.0497
0.0375	.5625	.0755	.5625	.0707	.5625	.0641
0.0500	.7500	.0872	.7500	.0816	.7500	.0760
0.0625	.9375	.0975	.9375	.0913	.9375	.0863

$N = 40, \lambda = 10, \mu = 100$

0.0100	.3000	.0264	.3000	.0258	.3000	.0213
0.0375	1.1250	.0512	1.1250	.0500	1.1250	.0477
0.0625	1.8750	.0661	1.8750	.0646	1.8750	.0628

V. MODEL FOR A SYSTEM WITH TWO TYPES OF TERMINALS

A. INTRODUCTION

The processor-sharing model we have been describing so far deals with a computer system having a single exponential service time distribution only. To generalize it we will consider a model of a computer system that consists of one Central Processing Unit, M terminals of type I and N terminals of type II. Type I terminals submit jobs to be processed by the processor at rate λ_1 , which need an expected amount of work $1/\mu_1$. Likewise for type II terminals, the arrival rate to the processor is λ_2 , and the expected work needed is $1/\mu_2$. Think times and amounts of work requested are to be independent and exponential.

This computer system may be viewed as the one having the ability to process two types of jobs. Each type of jobs must be submitted from its corresponding type of terminal.

The expected response time of a tagged job that requires T units of processing time will be derived by the same approach we used for the one-type model. However, now, we have to consider conditioning the given tagged job to be either type I or type II. The continuous time Markov-chain to be considered for the model is bivariate, with one variable being the number of jobs of type I being processed and the other being the number of jobs of type II being processed. The response time of the tagged job will depend on the numbers of both types of jobs in the system with it.

Once we obtain the expected conditional response time given the initial condition of how many others as described above, we will have to remove those conditions by the steady-state distribution of the tagged job given it is of

type I or of type II. The application of this steady state distribution to the conditional response time will enable us to compute the mean, standard deviation, and higher moments of response time of a tagged job that requires T units of work time, and hence the skewness and kurtosis as well. These values can help judge the goodness of the normal approximation.

Simulation will be used to generate the conditional response times as before. The steady state distribution of the tagged job will be used to compute estimates of the mean, variance, skewness and kurtosis of response time for a tagged job requiring T units of processing time. Finally, we will describe an approach for normal approximations by the central limit theorem, and by a heavy traffic approach as for the one-type model, and study their accuracy through simulation.

B. STEADY-STATE DISTRIBUTION

When the system has two types of terminals to deal with, the direct derivation to find differential equations or to apply the Laplace transforms for the conditional response time of a tagged job is much more complicated than when the system has only one type of terminal. Since, we now have to consider not only the condition that the job is tagged to find how many of them already present in the system but also the condition that the job is of what type, I or II.

However, under the processor-sharing concept with Markov assumptions, exponential think times and work request times, it will not be too hard to simulate the expected conditional response time, given that the job is one of the two types, and that it initially finds $i-1$ others of the same type and j of the other type present in the system. But then to remove these conditions we will have to apply the joint

steady-state distributions of the number of jobs present in the system just after the tagged job enters, given the tagged job is of specified type. The similar computation is performed for the case the tagged job is of another type.

Let M and N be the number of type I and type II terminals respectively. If $\widetilde{X}_k(t)$ is the number of type k jobs at clock time t for $k = 1, 2$, then, before a tagged job enters and let (i, j) , for $i = 0, 1, 2, \dots, M$ and $j = 0, 1, 2, \dots, N$, be the system state i type I jobs and j type II jobs being processed, the limiting distribution of the number of type I and type II jobs in the system is

$$\widetilde{\pi}(i, j) = \lim_{t \rightarrow \infty} P[(\widetilde{X}_1(t), \widetilde{X}_2(t)) = (i, j)].$$

Consider the local balance equations for type I tagged jobs:

$$\begin{aligned} \widetilde{\pi}(0, 0) M \lambda_1 &= \mu_1 \widetilde{\pi}(1, 0) \\ \widetilde{\pi}(1, 0) (M-1) \lambda_1 &= \mu_1 \widetilde{\pi}(2, 0) \\ &\vdots \\ \widetilde{\pi}(i, 0) (M-i) \lambda_1 &= \mu_1 \widetilde{\pi}(i+1, 0) \end{aligned} \tag{5.1}$$

which imply that, for $i = 0, 1, 2, \dots, M$, the probability that the system is in state $(i, 0)$ is

$$\widetilde{\pi}(i, 0) = M(M-1) \dots (M-i+1) (\lambda_1 / \mu_1)^i \widetilde{\pi}(0, 0). \tag{5.2}$$

Likewise, we can derive the same thing for type II tagged jobs:

$$\begin{aligned}
 \widetilde{\pi}(0,0) N \lambda_2 &= \mu_2 \widetilde{\pi}(0,1) \\
 \widetilde{\pi}(0,1) (N-1) \lambda_2 &= \mu_2 \widetilde{\pi}(0,2) \\
 &\vdots \\
 \widetilde{\pi}(0,j) (N-j) \lambda_2 &= \mu_2 \widetilde{\pi}(0,j+1)
 \end{aligned} \tag{5.3}$$

which in turn imply that, for $j = 0, 1, \dots, n$,

$$\widetilde{\pi}(0,j) = N(N-1)\dots(N-j+1) (\lambda_2/\mu_2)^j \widetilde{\pi}(0,0). \tag{5.4}$$

Other local balance equations are :

$$\begin{aligned}
 \widetilde{\pi}(i,0) N \lambda_2 &= \frac{1}{i+1} \mu_2 \widetilde{\pi}(i,1) \\
 \widetilde{\pi}(i,1) (N-1) \lambda_2 &= \frac{2}{i+2} \mu_2 \widetilde{\pi}(i,2) \\
 \widetilde{\pi}(i,2) (N-2) \lambda_2 &= \frac{3}{i+3} \mu_2 \widetilde{\pi}(i,3) \\
 &\vdots \\
 \widetilde{\pi}(i,j) (N-j) \lambda_2 &= \frac{(j+1)}{i+j+1} \mu_2 \widetilde{\pi}(i,j+1).
 \end{aligned} \tag{5.5}$$

Finally we can comfortably guess, from the equations (5.5), the steady-state distribution of the total number of jobs (of both types) before the tagged job enters as, for $i = 0, 1, \dots, M$ and $j = 0, 1, \dots, N$,

$$\begin{aligned} \widetilde{\pi}(i, j) &= \binom{i+j}{j} M(M-1)\dots(M-i+1) (\lambda_1/\mu_1)^i \\ &\times N(N-1)\dots(N-j+1) (\lambda_2/\mu_2)^j \widetilde{\pi}(0,0). \end{aligned} \quad (5.6)$$

This distribution also satisfies the full balance equations. The steady-state distribution of the number of jobs in the system can be found by choosing $\widetilde{\pi}(0,0)$ so that $\sum_{i=0}^M \sum_{j=0}^N \widetilde{\pi}(i, j) = 1$. More details can be found in Gaver and Jacobs. [Ref. 12]

The steady state distribution of the entering tagged job being of type I and there being i jobs of type I and j jobs of type II processing when the tagged job enters is

$$q(i, j, I) = k \widetilde{\pi}(i-1, j) (M-i) \lambda_1, \quad (5.7)$$

and for type II tagged job

$$q(i, j, II) = k \widetilde{\pi}(i, j-1) (N-j) \lambda_2, \quad (5.8)$$

where k is selected so that $\sum_{i=0}^M \sum_{j=0}^N [q(i, j, I) + q(i, j, II)] = 1$.

Similarly, the conditional distribution of there being i jobs of type I and j jobs of type II processing when the tagged job arrives given the tagged job is of type I is

$$q_I(i, j) = k_I \widetilde{\pi}(i-1, j) (M-i) \lambda_1, \quad (5.9)$$

where k_I is chosen so that $\sum_{i=0}^M \sum_{j=0}^N q_I(i, j) = 1$.

C. A CENTRAL LIMIT THEOREM FOR THE RESPONSE TIME

In this section we will present a central limit theorem for the conditional distribution of the response time of a tagged job requiring T units of work given it is of a particular type. More details can be found in Gaver and Jacobs. [Ref. 12]

In what follows we will assume the tagged job is of type I. Let $X(t)$ be the number of type I jobs (excluding the tagged one) being processed at work time t ; that is when the tagged job has acquired t units of work. Then $\{X_1(t), X_2(t); t \geq 0\}$ is a continuous time Markov chain with rates

$$(i, j) \rightarrow (i+1, j) : (M - (i+1)) \lambda_1 (i+j+1), \quad 0 \leq i < M-1 \quad (5.10)$$

$$(i, j) \rightarrow (i, j+1) : (N - (j+1)) \lambda_2 (i+j+1), \quad 0 \leq j < N \quad (5.11)$$

$$(i, j) \rightarrow (i-1, j) : i \mu_1, \quad 1 \leq i \leq M-1 \quad (5.12)$$

$$(i, j) \rightarrow (i, j-1) : j \mu_2, \quad 1 \leq j \leq N. \quad (5.13)$$

Similar arguments to those used in deriving (5.6) show that the limiting distribution $\pi(i, j)$ for the Markov chain having rates (5.10), (5.11), (5.12) and (5.13) is of the form

$$\pi(i, j) = \binom{i+j}{j} (M-1)(M-2)\dots(M-i) (\lambda_1/\mu_1)^i \quad (5.14)$$

$$\times N(N-1)\dots(N-j+1) (\lambda_2/\mu_2)^j \pi(0, 0),$$

where $\pi(0,0)$ is chosen so that $\sum_{i=0}^M \sum_{j=0}^N \pi(i,j) = 1$.

The response time of the tagged job which requires work T is

$$R(T) = \int_0^T [X_1(u) + X_2(u) + 1] du, \quad (5.15)$$

an integral of a function of a continuous time Markov chain.

As a result the central limit theorem of Keilson [Ref. 8] applies to give that as $T \rightarrow \infty$, there are constants m and σ such that

$$\frac{R(T) - mT}{\sigma\sqrt{T}}$$

has an asymptotic normal distribution with mean zero and variance one. In this case

$$m = \sum_{i=0}^{M-1} \sum_{j=0}^N (i+j+1) \pi(i,j). \quad (5.16)$$

The variance σ^2 is computed as follows.

Let Q be the infinitesimal generator of $(X_1(t), X_2(t))$ with system states in the order, i.e. $(0,0), \dots, (M-1,0), (0,1), \dots, (M-1,1), \dots, (0,N), \dots, (M-1,N)$. Then Q is an $M \times (N+1)$ square matrix having the form

$$Q = \begin{bmatrix} A_0 & R_0 & 0 & 0 & 0 & \dots \\ M_1 & A_1 & R_1 & 0 & 0 & \dots \\ 0 & M_2 & A_2 & R_2 & 0 & \dots \\ 0 & 0 & M_3 & A_3 & R_3 & \dots \\ \cdot & \cdot & \cdot & \cdot & \cdot & \dots \end{bmatrix}$$

where the non-zero elements are the rates of those system states.

Let γ be the maximum absolute value of a diagonal element of Q , then by uniformization of the chain we obtain a matrix $\tilde{A} = I + \frac{1}{\gamma} Q$.

Let $f(i, j) = i+j+1$, $\pi_j(i) = \pi(i, j)$, $f_j(i) = f(i, j)$ and $(\pi f)_j = \sum_{i=0}^{M-1} \pi_j(i) f_j(i)$. Then according to Keilson [Ref. 8] the central-limit theorem variance term, σ^2 , for the integral $\int_0^T f(X_1(u), X_2(u)) du$ is as follows.

$$\sigma^2 = 2 [(\pi f)_0, (\pi f)_1, \dots, (\pi f)_N] \tilde{Z} \begin{bmatrix} f_0 \\ f_1 \\ \vdots \\ f_N \end{bmatrix} \quad (5.17)$$

where $\tilde{Z} = \frac{1}{\gamma} [I - \tilde{A} + \tilde{L}]^{-1} - \tilde{L}$, and

$$\tilde{L} = \begin{bmatrix} \pi_0 & \pi_1 & \dots & \pi_N \\ \pi_0 & \pi_1 & \dots & \pi_N \\ \vdots & & & \\ \pi_0 & \pi_1 & \dots & \pi_N \end{bmatrix} \quad (5.18)$$

If we define $\underline{X} = (x_0, x_1, \dots, x_N) = [I - \tilde{A} + \tilde{L}]^{-1} f$, then, since $I - \tilde{A} = -\frac{1}{\gamma} Q$, we have

$$\left[-\frac{1}{\gamma} Q + \tilde{L} \right] \underline{X} = f. \quad (5.19)$$

Multiplying both sides of (5.19) by $\underline{\pi}$, and since $\underline{\pi} Q = 0$ and $\tilde{L} \underline{X} = \underline{\pi} \underline{X}$, we then have $\underline{\pi} \underline{X} = \underline{\pi} f$. Thus (5.19) becomes

$$-\frac{1}{\gamma} Q \underline{X} = f - (\underline{\pi} f), \quad (5.20)$$

or, better,

$$Q_X = -\gamma(f - \underline{\Pi}f). \quad (5.21)$$

We can then solve equation (5.21) for X and thus the central limit theorem variance σ^2 from equation (5.17).

D. HEAVY TRAFFIC APPROXIMATION FOR THE RESPONSE TIME

As before, consider a computer system consisting of M type I terminals with arrival rate λ_1 and service rate μ_1 , and N type II terminals with arrival rate λ_2 and service rate μ_2 respectively. In this section we will present a heavy traffic approximation for the response time. More details of the approximation are described by Gaver and Jacobs. [Ref. 12]

Let $L_1 = \lambda_1 M$, $L_2 = \lambda_2 N$ and $c = M/N$. To simplify the notations used in deriving heavy traffic mean and variance, we define the following expressions.

First we solve for m_1 which is the positive solution of the quadratic equation;

$$0 = -L_2 L_1 c + (\mu_1 - L_1) \mu_2 L_1 \quad (5.22)$$

$$+ m_1 [L_1^2 \mu_2 + (\mu_1 - L_1) (\mu_1 L_2 - \mu_2 L_1) + c L_2 L_1 \mu_1]$$

$$+ m_1^2 [L_1 (\mu_1 L_2 - \mu_2 L_1)],$$

then set $m_2 = [m_1 \mu_1 / (L_1 (1 - m_1))] - m_1$.

$$\text{Let } a_1 = -[L_1 (1 - m_1) - \mu_1 - L_1 (m_1 + m_2)]$$

$$a_2 = -[L_1 (1 - m_1)]$$

$$b_1 = -[L_2 (c - m_2)]$$

$$b_2 = -[L_2 (c - m_2) - \mu_2 - L_2 (m_1 + m_2)]$$

$$\sigma^2 = L_1 (1 - m_1) (m_1 + m_2) + m_1 \mu_1$$

$$\sigma_2^2 = L_2 (c - m_2) (m_1 + m_2) + m_2 \mu_2.$$

Now put $S_0 = (-(b_2 + a_1) + \sqrt{(b_2 + a_1)^2 - 4(a_1 b_2 - a_2 b_1)}) / 2$

and $S_1 = (-(b_2 + a_1) - \sqrt{(b_2 + a_1)^2 - 4(a_1 b_2 - a_2 b_1)}) / 2$

(Note that S_0 and S_1 are solutions to a quadratic equation).

Also put $\beta_{11} = -(S_0 + b_2) / (S_1 - S_0)$; $\beta_{21} = b_1 / (S_1 - S_0)$

$$\beta_{22} = -(S_0 + a_1) / (S_1 - S_0) ; \beta_{12} = a_2 / (S_1 - S_0)$$

$$K_{11} = (S_1 + b_2) / (S_1 - S_0) ; K_{12} = -a_1 / (S_1 - S_0)$$

$$K_{22} = (S_1 + a_1) / (S_1 - S_0) ; K_{21} = -b_1 / (S_1 - S_0)$$

$$\gamma_{31} = (b_2 - b_1) / (a_1 b_2 - a_2 b_1) = (b_2 - b_1) / (S_0 S_1)$$

$$\gamma_{32} = (a_1 - a_2) / (a_1 b_2 - a_2 b_1) = (a_1 - a_2) / (S_0 S_1)$$

$$\beta_{31} = (\gamma_{31} S_1 + 1) / (S_0 - S_1) ; \beta_{32} = (\gamma_{32} S_1 + 1) / (S_0 - S_1)$$

$$K_{31} = -(\gamma_{31} S_0 + 1) / (S_0 - S_1) ; K_{32} = -(\gamma_{32} S_0 + 1) / (S_0 - S_1)$$

Now, let $X_i(t)$ = number of type i jobs, $i = 1, 2$, processing at work time t (tagged job not included). Suppose the tagged job is of type 1. Put

$$Y_i(t) = (X_i(t) - M m_i) / M . \quad (5.23)$$

Then, $Y_i(t)$ satisfies the following stochastic differential equations

$$dY_1(t) = -a_1 Y_1(t) + (-a_2) Y_2(t) + \sigma_1 dW_1(t) , \quad (5.24)$$

$$dY_2(t) = -b_1 Y_1(t) - (-b_2) Y_2(t) + \sigma_2 dW_2(t) , \quad (5.25)$$

where $W_1(t)$ and $W_2(t)$ are independent Brownian motions.

The response time for the tagged job requiring t units of processing time is

$$\begin{aligned}
 R(t) &= \int_0^t [X_1(u) + X_2(u)] du & (5.26) \\
 &= \int_0^t [\sqrt{M} Y_1(u) + Mm_1 + \sqrt{M} Y_2(u) + Mm_2] du \\
 &= [M(m_1 + m_2)]t + \sqrt{M} \int_0^t [Y_1(u) + Y_2(u)] du.
 \end{aligned}$$

Let $Z(t) = \int_0^t [Y_1(u) + Y_2(u)] du$. Then $Z(t)$ satisfies the stochastic differential equation

$$dZ(t) = Y_1(t) + Y_2(t). \quad (5.27)$$

The heavy traffic approximation for the distribution of $R(t)$ is that $R(t)$ has a normal distribution with mean

$$[M(m_1 + m_2)]t$$

and variance equal to $M(\text{Var}[Z(t)])$.

To derive an expression for $\text{Var}[Z(t)]$ we will apply results found in Arnold [Ref. 13] to solve the following system of stochastic differential equations:

$$d\underline{V}(t) = A\underline{V}(t) + B d\underline{W}(t), \quad (5.28)$$

where $\underline{V}(t) = [Y_1(t), Y_2(t), Z(t)]$, $d\underline{W}(t) = [dW_1(t), dW_2(t)]$

and

$$A = \begin{bmatrix} -a_1 & -a_2 & 0 \\ -b_1 & -b_2 & 0 \\ 1 & 1 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \\ 0 & 0 \end{bmatrix}$$

By corollary (8.2.4) on page 130, Arnold states the solution of equation (5.28) as follows.

$$Y_1(t) = [\beta_{11} e^{s_0 t} + K_{11} e^{s_1 t}] Y_1(0) + [\beta_{12} e^{s_0 t} + K_{12} e^{s_1 t}] Y_2(0) \quad (5.29)$$

$$+ \int_0^t \sigma_1 [\beta_{11} e^{s_0(t-u)} + K_{11} e^{s_1(t-u)}] dW_1(u)$$

$$+ \int_0^t \sigma_2 [\beta_{12} e^{s_0(t-u)} + K_{12} e^{s_1(t-u)}] dW_2(u),$$

$$Y_2(t) = [\beta_{21} e^{s_0 t} + K_{21} e^{s_1 t}] Y_1(0) + [\beta_{22} e^{s_0 t} + K_{22} e^{s_1 t}] Y_2(0) \quad (5.30)$$

$$+ \int_0^t \sigma_1 [\beta_{21} e^{s_0(t-u)} + K_{21} e^{s_1(t-u)}] dW_1(u)$$

$$+ \int_0^t \sigma_2 [\beta_{22} e^{s_0(t-u)} + K_{22} e^{s_1(t-u)}] dW_2(u),$$

$$Z(t) = [\gamma_{31} + \beta_{31} e^{s_0 t} + K_{31} e^{s_1 t}] Y_1(0) \quad (5.31)$$

$$+ [\gamma_{32} + \beta_{32} e^{s_0 t} + K_{32} e^{s_1 t}] Y_2(0)$$

$$+ \int_0^t \sigma_1 [\gamma_{31} + \beta_{31} e^{s_0(t-u)} + K_{31} e^{s_1(t-u)}] dW_1(u)$$

$$+ \int_0^t \sigma_2 [\gamma_{32} + \beta_{32} e^{s_0(t-u)} + K_{32} e^{s_1(t-u)}] dW_2(u).$$

Let $C_1 = \gamma_{31} + \beta_{31} e^{s_0 t} + K_{31} e^{s_1 t}$ and $C_2 = \gamma_{32} + \beta_{32} e^{s_0 t} + K_{32} e^{s_1 t}$. It then follows that

$$\begin{aligned} \text{Var}[Z(t)] &= C_1^2 \text{Var}[Y_1(0)] + C_2^2 \text{Var}[Y_2(0)] \\ &+ 2C_1 C_2 (\text{Cov}[Y_1(0), Y_2(0)]) \end{aligned} \quad (5.32)$$

$$\begin{aligned} &+ \int_0^t \sigma_1^2 [\gamma_{31} + \beta_{31} e^{s_0(t-u)} + K_{31} e^{s_1(t-u)}]^2 du \\ &+ \int_0^t \sigma_2^2 [\gamma_{32} + \beta_{32} e^{s_0(t-u)} + K_{32} e^{s_1(t-u)}]^2 du. \end{aligned}$$

where : $\text{Var}[Y_1(0)] = \sigma_1^2 \int_0^\infty [\beta_{11} e^{s_0 u} + K_{11} e^{s_1 u}]^2 du$

$$+ \sigma_2^2 \int_0^\infty [\beta_{12} e^{s_0 u} + K_{12} e^{s_1 u}]^2 du$$

$$\begin{aligned} \text{Var}[Y_2(0)] &= \sigma_1^2 \int_0^\infty [\beta_{21} e^{s_0 u} + K_{21} e^{s_1 u}]^2 du \\ &+ \sigma_2^2 \int_0^\infty [\beta_{22} e^{s_0 u} + K_{22} e^{s_1 u}]^2 du \end{aligned}$$

$$\text{Cov}[Y_1(0), Y_2(0)]$$

$$\begin{aligned} &= \sigma_1^2 \int_0^\infty [\beta_{11} e^{s_0 u} + K_{11} e^{s_1 u}] [\beta_{21} e^{s_0 u} + K_{21} e^{s_1 u}] du \\ &+ \sigma_2^2 \int_0^\infty [\beta_{12} e^{s_0 u} + K_{12} e^{s_1 u}] [\beta_{22} e^{s_0 u} + K_{22} e^{s_1 u}] du \end{aligned}$$

Substitute the integral forms for $\text{Var}[Y_1(0)]$, $\text{Var}[Y_2(0)]$ and $\text{Cov}[Y_1(0), Y_2(0)]$ in equation (5.32). It remains now to solve the simple integrals to obtain a formula for the variance of the response time by heavy traffic approximation.

E. SIMULATION

1. Algorithm

Practically, the simulation for response time in a system with M type I terminals and N type II terminals is performed by applying the same procedure as for the one-type model. This means that, for each initial system state of a tagged job of each type, we generate the exponential arrival and departure times to be able to determine the next system state after a sojourn time. The simulation is for the work time process.

If the minimum of the two exponential times is greater than or equal to the amount of work-time requirement, we simply determine the response time by converting it into real time term. If not, we determine the next system state, the work-time remaining to be accomplished and the accumulated clock time, and generate new exponential times. We repeat this procedure until we obtain a response time for each initial system state of a tagged job of each type. Once we obtain those conditional response times, we can easily calculate the mean, moments, and measures of skewness and kurtosis by applying the steady-state distribution derived previously.

We now describe in detail how to simulate a conditional response time, given an initial system state is (i, j) and the tagged job is of type I, for a computer system consisting of M type I terminals having λ_1 and μ_1 as arrival and requested work rates, and N type II terminals having λ_2 and μ_2 as arrival and requested work rates respectively; (i does not include the tagged job).

In units of work-time the arrival rate of the next tagged job of either type is

$$A = [(M-(i+1))\lambda_1 + (N-j)\lambda_2](i+j+1), \quad (5.33)$$

and the service completion rate of a job in system is

$$D = [i\mu_1 + j\mu_2]. \quad (5.34)$$

We, then, generate two exponential times with parameters A and D. Call them t' and t'' respectively.

If $\min(t', t'') \geq w_0$, then set the response time

$$R(i, j, I) = c_0 + w_0(i+j+1),$$

where c_0 is the accumulated clock time (initially = 0) and w_0 is the amount of work-time remains to be completed (initially = T work time requirement of the job).

If $\min(t', t'') < w_0$, set

$$w_0 = w_0 - \min(t', t'')$$

$$c_0 = c_0 + \min(t', t'')(i+j+1).$$

If $\min(t', t'') = t'$, which means the system state is changed by an arrival, we generate two more exponential times with parameters $(M-(i+1))\lambda_1(i+j+1)$ and $(N-j)\lambda_2(i+j+1)$ respectively. If the latter quantum of time is less than the previous one, i.e. an arrival of type II job occurs first, the system state changes from $(i+1, j)$ to $(i+1, j+1)$. Otherwise, it changes from $(i+1, j)$ to $(i+2, j)$. In the contrary, if $\min(t', t'') = t''$, which means the system state is changed by a departure, we also generate two exponential times for service time with parameters $i\mu_1$ and $j\mu_2$

respectively. If the latter is the minimum of the two, i.e. a departure of type II job occurs first, the system state will then change from (i, j) to $(i, j-1)$. Otherwise, it changes from (i, j) to $(i-1, j)$.

Based on new system state and new work-time requirement, we repeat the same procedure until we obtain the corresponding response time.

Note that we will have in total M times $N+1$ response times for type I tagged job and $M+1$ times N for type II tagged job. We can then apply the steady-state distributions $q_I(i, j)$ and $q_{II}(i, j)$ to compute the mean, variance, skewness and kurtosis of the response time of a tagged job that requires T units of processing time as for the one type model.

2. Computation of Simulated Mean, Variance, Skewness and Kurtosis

a. Simulated Mean Response Time

Let $R_k(i, j, I)$ and $R_k(i, j, II)$ be the k -th simulated conditional response time, given the initial system state (tagged job included) is (i, j) and the tagged job is of type I and type II respectively.

If we perform K replications for simulation of response time at each initial system state, then the mean response time, given that the tagged job is of type I, is

$$\bar{R} = \sum_{i=0}^M \sum_{j=0}^N \bar{R}(i, j, I) q_I(i, j), \quad (5.35)$$

where $\bar{R}(i, j, I) = \left(\sum_{k=1}^K R_k(i, j, I) \right) / K.$

b. Standard Error of the Mean Response Time

Under an assumption that $R_k(i, j, I)$ and $R_k(i, j, II)$ are independent, we can compute the variance of the estimated mean response time, given the tagged job is of type I, based on the value obtained from equation (5.35) as follows.

$$\text{Var}(\bar{R}) = \frac{1}{K} \sum_{i=0}^M \sum_{j=0}^N \text{Var}[\bar{R}(i, j, I)] q_I^2(i, j) \quad (5.36)$$

where $\text{Var}[\bar{R}(i, j, I)] = \left[\sum_{k=1}^K (R_k(i, j, I) - \bar{R}(i, j, I))^2 \right] / K$.

To obtain the standard error we simply take the square root of the expression (5.36).

c. Variance of Simulated Response Time

We start by computing higher central moments of response time, given the tagged job is of type I, by applying the same method as for one-type model. The second central moment which is also the variance of response time may be computed, based on simulation data, as follows.

$$\text{Var}[R] = \frac{1}{K} \sum_{k=1}^K \sum_{i=0}^M \sum_{j=0}^N [(R_k(i, j, I) - \bar{R})^2 q_I(i, j)] \quad (5.37)$$

$$= \frac{1}{K} \sum_{k=1}^K \sum_{i=0}^M \sum_{j=0}^N [(R_k(i, j, I) - \bar{R}(i, j, I))^2 + (\bar{R}(i, j, I) - \bar{R})^2] q_I(i, j).$$

d. Skewness of response time

The skewness of response time can be computed by the usual formula, i.e.

$$\gamma_3 = \mu_3 / \sigma^3,$$

where μ_3 is the third central moment and σ is the standard deviation of the response time. As for the calculation of the variance,

$$\mu_3 = \frac{1}{K} \sum_{k=1}^K \sum_{i=0}^M \sum_{j=0}^N [(R_k(i, j, I) - \bar{R})^3 q_I(i, j)] \quad (5.38)$$

$$= \frac{1}{K} \sum_{k=1}^K \sum_{i=0}^M \sum_{j=0}^N [(R_k(i, j, I) - \bar{R}(i, j, I))^3$$

$$+ 3(R_k(i, j, I) - \bar{R}(i, j, I))^2 (\bar{R}(i, j, I) - \bar{R})$$

$$+ (\bar{R}(i, j, I) - \bar{R})^3] q_I(i, j) .$$

e. Kurtosis of response time

The kurtosis is defined as $\gamma_4 = (\mu_4 / \sigma^4) - 3$, where μ_4 is the fourth central moment and σ is the standard deviation of the response time. As before;

$$\mu_4 = \frac{1}{K} \sum_{k=1}^K \sum_{i=0}^M \sum_{j=0}^N (R_k(i, j, I) - \bar{R})^4 q_I(i, j) \quad (5.39)$$

and by simple polynomial expansion;

$$\begin{aligned} \mu_4 = & \frac{1}{K} \sum_{k=1}^K \sum_{i=0}^M \sum_{j=0}^N [(R_k(i, j, I) - \bar{R}(i, j, I))^4 \\ & + 4 (R_k(i, j, I) - \bar{R}(i, j, I))^3 (\bar{R}(i, j, I) - \bar{R}) \\ & + 6 (R_k(i, j, I) - \bar{R}(i, j, I))^2 (\bar{R}(i, j, I) - \bar{R})^2 \\ & + (\bar{R}(i, j, I) - \bar{R})^4] q_I(i, j). \end{aligned}$$

3. Numerical Results

Tables X and XI show the values of the means, their standard errors (numbers between parentheses below the means), standard deviations, skewness and kurtosis of the empirical response times for a system consisting of 5 type I terminals and 5 type II terminals. We also attach the values of the mean and the standard deviation computed by the central limit theorem and the limiting diffusion in heavy traffic. Those values are computed for two cases. In the first case, Table X, the arrival and departure rates are $\lambda_1 = 30$, $\mu_1 = 100$ for type I jobs and $\lambda_2 = 20$, $\mu_2 = 50$ for type II jobs. But in the second case, Table XI, those rates are in reverse order, i.e. $\lambda_1 = 20$, $\mu_1 = 50$, $\lambda_2 = 30$ and $\mu_2 = 100$.

We see that the results shown in tables X and XI indicate that in both cases the distribution of the empirical response time of a tagged job that requires a fixed amount of processing time is about the same. This indicates that the distribution of the response time is almost independent of the job type of the tagged job. The measures

TABLE X

Simulation results for a two-type model

 $M = 5, N = 5, \lambda_1 = 30, \lambda_2 = 20, \mu_1 = 100, \mu_2 = 50$

<u>Time</u>	<u>Mean</u>	<u>Std.Dev.</u>	<u>Skewness</u>	<u>Kurtosis</u>
0.0100	0.07102 (.00014)	0.01365	-0.80425	-0.77956
CLT	0.07156	0.02038		
HTA	0.07101	0.01315		
0.0250	0.17770 (.00032)	0.02676	-0.87139	-0.73673
CLT	0.17890	0.03222		
HTA	0.17752	0.02528		
0.0375	0.26720 (.00043)	0.03506	-0.93571	-0.70581
CLT	0.26835	0.03947		
HTA	0.26628	0.03249		
0.0500	0.35671 (.00054)	0.04185	-0.98596	-0.70335
CLT	0.35780	0.04557		
HTA	0.35503	0.03840		
0.0625	0.44607 (.00062)	0.04758	-0.82744	-0.67426
CLT	0.44725	0.05095		
HTA	0.44379	0.04352		

TABLE XI
Simulation Results for two-type model

$M = 5, N = 5, \lambda_1 = 20, \lambda_2 = 30, \mu_1 = 50, \mu_2 = 100$

<u>Time</u>	<u>Mean</u>	<u>Std.Dev.</u>	<u>Skewness</u>	<u>Kurtosis</u>
0.0100	0.07128 (.00013)	0.01345	-0.79647	-0.76541
CLT	0.07065	0.02038		
HTA	0.07101	0.01315		
0.0250	0.17832 (.00031)	0.02632	-0.86195	-0.76107
CLT	0.17663	0.03223		
HTA	0.17752	0.02528		
0.0375	0.26638 (.00045)	0.03525	-0.87967	-0.76051
CLT	0.26495	0.03947		
HTA	0.26627	0.03249		
0.0500	0.35704 (.00053)	0.04139	-0.86977	-0.75428
CLT	0.35327	0.04558		
HTA	0.35503	0.03840		
0.0625	0.44385 (.00063)	0.04777	-0.82988	-0.72768
CLT	0.44158	0.05096		
HTA	0.44379	0.04352		

TABLE XII

Approx. Normal VS Empirical quantiles

$M = 5, N = 5, \lambda_1 = 30, \lambda_2 = 20, \mu_1 = 100, \mu_2 = 50$

<u>Time.</u>	<u>Normal</u>	<u>.10</u>	<u>.25</u>	<u>.50</u>	<u>.75</u>	<u>.90</u>	<u>.95</u>	<u>.99</u>
0.0100	CLT	.049	.159	.457	.869	.998	1.0	1.0
	HTA	.114	.229	.442	.718	.919	.979	1.0
0.0250	CLT	.078	.194	.462	.809	.978	.997	1.0
	HTA	.113	.228	.442	.717	.916	.973	.999
0.0375	CLT	.088	.204	.455	.787	.962	.995	1.0
	HTA	.111	.224	.432	.702	.909	.965	.999
0.0500	CLT	.088	.212	.459	.771	.955	.989	1.0
	HTA	.111	.226	.434	.696	.898	.961	.998
0.0625	CLT	.093	.212	.465	.773	.945	.986	1.0
	HTA	.110	.222	.432	.705	.889	.954	.997

$M = 5, N = 5, \lambda_1 = 20, \lambda_2 = 30, \mu_1 = 50, \mu_2 = 100$

<u>Time</u>	<u>Normal</u>	<u>.10</u>	<u>.25</u>	<u>.50</u>	<u>.75</u>	<u>.90</u>	<u>.95</u>	<u>.99</u>
0.0100	CLT	.041	.137	.433	.838	.998	1.0	1.0
	HTA	.109	.221	.443	.719	.915	.977	1.0
0.0250	CLT	.062	.170	.425	.765	.969	.997	1.0
	HTA	.103	.223	.438	.708	.911	.975	1.0
0.0375	CLT	.079	.191	.429	.754	.952	.991	1.0
	HTA	.118	.229	.447	.713	.905	.969	.999
0.0500	CLT	.075	.181	.417	.724	.939	.986	1.0
	HTA	.106	.222	.431	.693	.897	.965	.998
0.0625	CLT	.081	.198	.436	.745	.932	.980	1.0
	HTA	.115	.238	.451	.719	.898	.960	.997

of skewness for these particular systems indicate that the distribution of the empirical response time is a little skewed to the left, and the measures of kurtosis show thinner tails than it should be for the normal distribution even for a moderately large work-time requirement. We will, therefore, observe the difference between the normal quantiles, approximated by the central limit theorem and by heavy traffic analysis, and the empirical response time to make further judgement on the appropriateness of the approximations.

We now compare the empirical response time at each initial system state to the various quantiles of the normal distributions whose mean and variance are approximated by the central limit theorem and the heavy traffic analysis. The concept of the computations of the normal quantiles and the relative frequencies of the empirical response time is the same as that described for the one-type job model in chapter IV. The results shown in table XII indicate almost the same behavior in distribution of the empirical response time for both cases. Again, this fact indicates that the distribution of the response time is almost independent of the job type of the tagged job. Those values indicate that our normal approximations agree pretty well with the empirical response times obtained by simulation, especially when the work-time requirement becomes large. They also agree with the results shown in tables X and XI where the measures of skewness show a slightly left-skewed distribution and the measures of kurtosis (small negative values) indicate that the distribution of the empirical response times has a slightly flatter peak, fatter shoulders and thinner tails than the normal distribution. Table XII also shows that the empirical response time has a slightly higher mean than that approximated by either central limit theorem or heavy traffic analysis. However, we can still say that the

TABLE XIII

Simulation results for a two-type model

 $M = 5, N = 5, \lambda_1 = 40, \lambda_2 = 10, \mu_1 = 125, \mu_2 = 25$

<u>Time</u>	<u>Mean</u>	<u>Std.Dev.</u>	<u>Skewness</u>	<u>Kurtosis</u>
0.0100	0.07246 (.00014)	0.01378	-0.74325	-0.76697
CLT	0.07239	0.02491		
HTA	0.07197	0.01328		
0.0250	0.18158 (.00035)	0.02902	-0.88632	-0.76209
CLT	0.18096	0.03938		
HTA	0.17993	0.02708		
0.0375	0.27167 (.00051)	0.03886	-0.92108	-0.74667
CLT	0.27144	0.04823		
HTA	0.26990	0.03590		
0.0500	0.36227 (.00064)	0.04723	-0.98223	-0.76223
CLT	0.36193	0.05569		
HTA	0.35986	0.04326		
0.0625	0.45220 (.00077)	0.05566	-0.94014	-0.74837
CLT	0.45241	0.06227		
HTA	0.44983	0.04964		

TABLE XIV
Results for a two-type model

$M = 5, N = 5, \lambda_1 = 10, \lambda_2 = 40, \mu_1 = 25, \mu_2 = 125$

<u>Time</u>	<u>Std.Dev.</u>	<u>Skewness</u>	<u>Kurtosis</u>	
0.0100	0.07169 (.00014)	0.01335	-0.81156	-0.75757
CLT	0.07170	0.02369		
HTA	0.07197	0.01328		
0.0250	0.17877 (.00035)	0.02802	-0.85737	-0.81866
CLT	0.17925	0.03746		
HTA	0.17993	0.02708		
0.0375	0.26904 (.00049)	0.03755	-0.92639	-0.82570
CLT	0.26888	0.04587		
HTA	0.26990	0.03590		
0.0500	0.35883 (.00062)	0.04561	-1.00062	-0.79135
CLT	0.35851	0.05297		
HTA	0.35986	0.04326		
0.0625	0.44810 (.00074)	0.05272	-0.93762	-0.76072
CLT	0.44814	0.05922		
HTA	0.44983	0.04964		

TABLE XV

Approx. Normal VS Empirical quantiles

 $M = 5, N = 5, \lambda_1 = 40, \lambda_2 = 10, \mu_1 = 125, \mu_2 = 25$

<u>Time</u>	<u>Normal</u>	<u>.10</u>	<u>.25</u>	<u>.50</u>	<u>.75</u>	<u>.90</u>	<u>.95</u>	<u>.99</u>
0.0100	CLT	.024	.116	.445	.907	1.0	1.0	1.0
	HTA	.110	.228	.433	.710	.905	.973	1.0
0.0250	CLT	.054	.164	.435	.820	.993	1.0	1.0
	HTA	.112	.219	.422	.682	.901	.967	1.0
0.0375	CLT	.070	.186	.428	.795	.981	.999	1.0
	HTA	.113	.224	.414	.693	.899	.967	.999
0.0500	CLT	.075	.182	.439	.784	.972	.998	1.0
	HTA	.110	.216	.422	.689	.896	.969	1.0
0.0625	CLT	.083	.200	.443	.765	.959	.997	1.0
	HTA	.118	.222	.427	.683	.891	.958	1.0

 $M = 5, N = 5, \lambda_1 = 10, \lambda_2 = 40, \mu_1 = 25, \mu_2 = 125$

<u>Time</u>	<u>Normal</u>	<u>.10</u>	<u>.25</u>	<u>.50</u>	<u>.75</u>	<u>.90</u>	<u>.95</u>	<u>.99</u>
0.0100	CLT	.027	.119	.447	.906	1.0	1.0	1.0
	HTA	.110	.230	.458	.737	.931	.982	1.0
0.0250	CLT	.061	.178	.451	.824	.991	.999	1.0
	HTA	.121	.236	.464	.734	.930	.985	1.0
0.0375	CLT	.068	.191	.441	.790	.979	.999	1.0
	HTA	.117	.239	.451	.725	.927	.983	1.0
0.0500	CLT	.081	.189	.430	.776	.975	.997	1.0
	HTA	.117	.232	.444	.729	.935	.985	1.0
0.0625	CLT	.085	.199	.442	.772	.960	.995	1.0
	HTA	.121	.237	.456	.737	.923	.979	1.0

distribution of the empirical response time in table XII is not far from being normal especially when the work requirement is large. The heavy traffic analysis gives a better approximation for small amount of work requirement, while for large work requirement the central limit theorem shows a slightly better approximation.

More results are shown in tables XIII and XIV for the case of $\lambda_1 = 40$, $\lambda_2 = 10$, $\mu_1 = 125$, $\mu_2 = 25$ and the case of reverse rates respectively. The values in both tables do not differ much. This again indicates that the distribution of the response time is independent of the job type of the initial tagged job. The comparison of the empirical response time to the quantiles approximated by the central limit theorem and the limiting diffusion in heavy traffic shown in table XV also suggests the same conclusion. Finally, tables XVI and XVII show the results for a single job type model with the arrival and service rates from the average rates of the two job type model, i.e. $N = 10$, $\lambda = 25$, $\mu = 75$. The values in those tables suggest that there might be some particular cases where we can use the single job type model to approximate the two job type model. However, further investigation is not included in this thesis.

TABLE XVI
Simulation results for one-type model

$N = 10, \lambda = 25, \mu = 75$

<u>Time</u>	<u>Mean</u>	<u>Std.Dev.</u>	<u>Skewness</u>	<u>Kurtosis</u>
0.0100	0.07008 (.00024)	0.01391	-0.70894	0.70894
CLT	0.07008	0.02058		
ETA II	0.07000	0.01345		
0.0250	0.17658 (.00053)	0.02629	-0.94906	1.66179
CLT	0.17520	0.03253		
ETA II	0.17500	0.02576		
0.0375	0.26202 (.00078)	0.03607	-0.85598	1.32022
CLT	0.26280	0.03984		
ETA II	0.26250	0.03302		
0.0500	0.35014 (.00094)	0.04299	-0.76443	1.08030
CLT	0.35041	0.04601		
ETA II	0.35000	0.03897		
0.0625	0.43879 (.00106)	0.04783	-0.81254	1.31538
CLT	0.43801	0.05144		
ETA II	0.43750	0.04412		

TABLE XVII

Quantile comparison for one-job model

$$N = 10, \lambda = 25, \mu = 75$$

<u>Time</u>	<u>Normal</u>	<u>.10</u>	<u>.25</u>	<u>.50</u>	<u>.75</u>	<u>.90</u>	<u>.95</u>	<u>.99</u>
0.0100	CLT	.044	.157	.445	.849	.997	1.0	1.0
	HTA I	.060	.176	.443	.814	.988	1.0	1.0
	HTA II	.117	.235	.443	.722	.912	.976	1.0
0.0250	CLT	.067	.161	.424	.789	.972	.999	1.0
	HTA I	.080	.182	.418	.754	.946	.991	1.0
	HTA II	.093	.204	.418	.712	.912	.973	1.0
0.0375	CLT	.089	.203	.464	.764	.957	.994	1.0
	HTA I	.103	.221	.460	.740	.934	.984	.999
	HTA II	.114	.235	.460	.719	.907	.970	.999
0.0500	CLT	.092	.214	.447	.763	.947	.985	.999
	HTA I	.106	.234	.445	.734	.916	.968	.998
	HTA II	.116	.243	.445	.715	.901	.959	.997
0.0625	CLT	.083	.209	.440	.756	.940	.985	.999
	HTA I	.102	.223	.437	.717	.914	.975	.999
	HTA II	.111	.233	.437	.707	.899	.963	.998

VI. SUMMARY AND CONCLUSION

A. SUMMARY

We brought up a concept of processor-sharing and then described a model for processor-shared work-time allocation of a computer system with one type of terminals as a birth-death process under Markov assumptions in the first chapter. In the second chapter, the mean response time of a tagged job that requires a fixed amount of processing time was derived based on a condition that the tagged job enters to find an initial number of jobs present for processing. We also derived some higher moments of the response time by means of differential equations which led to the computations of variance, skewness and kurtosis of the distribution. Some numerical results obtained by solving those differential equations were then given as an example for a system with two terminals and the arrival and departure rates are 1 and 2 respectively.

A simulation procedure for the response time, given an initial system state when the job was tagged to the system, was described in Chapter III. We detailed an algorithm for a particular two-terminal system to check with the numerical method and generalized it for a computer system with a fixed number of terminals. Based on those empirical response times we showed how to compute the mean response time using the steady-state distribution. The standard errors of the mean response time, the empirical moments and the measures of skewness and kurtosis of the response time were also derived in that chapter. In Chapter IV, we derived the formulas to compute the approximate normal mean and variance of the response time by the central limit theorem for additive

functionals of a birth-death process and also by the limiting diffusion approximation in heavy traffic situation. Based on those approximated means and variances, we explained how to compute the various quantiles and compared them to the empirical response times obtained by simulation. The difference between the relative frequencies of the empirical data and the CDF of the approximated normal distributions indicates how well the normal distributions agree with the empirical response times.

In Chapter V, we described a model for a computer system with two types of terminals, again under Markov assumptions for a bivariate birth and death process. Only now we have to consider a condition that the tagged job is one of the two types. The steady-state distribution was then derived for this two-type system. This allows us to remove the condition of initial system state and leads to the computations of the mean response time of a tagged job that requires a fixed amount of processing time as for one-type model. The computations of higher moments, thus the measures of skewness and kurtosis, were also explained as well as the standard error of the mean. Two approximations for a normal distribution, one based on the central limit theorem for additive functionals and the other one based on the limiting diffusion in heavy traffic situation, were derived for this two-type model in the same way as for the one-type model. Next, a procedure to simulate the response time of a tagged job given an initial system state was described, and then based on these empirical data we make comparison to the quantiles of the approximating normal distributions to study the behavior of the empirical response times towards normality.

Finally, for appendixes, we attached two programs written in FORTRAN to perform the simulation of the response times as described in Chapters III and V respectively. The program in appendix A simulates the response time of a

tagged job based on its work-time requirement, the size of the system and the arrival and departure rates for a single job type model. It computes the statistic elements of the empirical response times such as mean, variance, skewness and kurtosis. It also computes the quantiles of the normal distributions whose mean and variance are approximated by the central limit theorem and heavy traffic analysis. It then computes the relative frequencies of the empirical response times compared to the quantiles. The standard error of each relative frequency is also computed. This will help us to make a judgement on the number of replications needed for simulation. The program in appendix B provides the same things for a two-type job model. We only have to input the mean and variance for the approximation by the central limit theorem from an APL program due to a large number of matrix operations.

E. CONCLUSION

The agreement of the normal approximations with the simulation is satisfactory for both one-type and two-type models, especially in the cases of large work-time requirement and/or under heavy traffic situations. The central limit theorem approximation always works when the work requirement is large enough even if we don't have a heavy traffic situation. The approximation based on the heavy traffic analysis seems, however, to work better, given that the system is under a heavy traffic situation, than the central limit theorem approach for small work requirement. But in a light traffic situation, the limiting diffusion approximation does not work well at all.

Based on the values of skewness and kurtosis of the empirical response time, we have tried to use the Edgeworth expansion to improve the normal distribution of the response

time approximated by the central limit theorem. The distribution computed by the Edgeworth approximation is close to the CLT normal distribution, and the results do not indicate significant improvement.

In the two-type model, we observe that the response time distribution appears to be almost independent of the type of the tagged job. It is hard to make a judgement on the behavior of the empirical response time based on the observations of the mean, variance, skewness and kurtosis, even though those values indicate that the distribution is well approximated by the normal. The simulation for the response time of the two-type model involves a lot of computations. For example, in a system consisting of 5 type I and 5 type II terminals, we have to consider 30 initial system states in total. If we want to perform a simulation with 500 replications, it will involve at least 15,000 computations. This fact indicates that the round-off error may be substantial. However, the relative frequencies of the empirical response time comparing to the approximate normal quantiles obtained for the particular cases in this thesis are in a rather good agreement with the theoretical suggestion. This means that, under the Markov assumptions, the distribution of the response time of a tagged job that requires a fixed (large) amount of processing time is approximately normal.

Throughout this thesis we have been assuming that the distributions of the arrival and service times are exponential. However, all the computations done for both models can as well be extended to the case of general distribution of service time. It might be of interest to study the models and derive a method to simulate the response time, since in the real world we would always have to deal with a more and more sophisticated computer system which is capable of dealing with many kinds of work. So further study might be needed for such system.

APPENDIX A
SIMULATION PROGRAM FOR ONE-TYPE MODEL

```

C***PROGRAM TO SIMULATE RESPONSE TIME IN PROCESSOR SHARING
C***OF N TERMINALS COMPARING TO NORMAL DISTRIBUTION.
C
C VARIABLES AND CONSTANTS
C N = NUMBER OF TERMINALS
C T = INITIALLY REQUIRED AMOUNT OF WORK
C W0 = AMT. OF WORK REQUIRED IN FUTURE
C CO = AMT. OF CLOCK TIME ACCUMULATES
C LAMDA = RATE AT WHICH A JOB APPLIES FOR COMPUTER SERVICE
C MU = FRACTION OF TIME THERE ARE J JOBS BEING PROCESSED
C RJ = PROCESSED WHEN J JOBS BEING PROCESSED
C RTJ = RESPONSE TIME WHEN J JOBS BEING PROCESSED
C LAMJ = RATE AT WHICH A JOB APPLIES FOR SERVICE WHEN J JOBS
C ARE IN SYSTEM
C MUJ = DEPARTURE RATE WHEN J JOBS ARE IN SYSTEM
C
C DECLARATIONS
C REAL T,LAMDA,MU,RJ(50),RTJ(50,1000),QJ(50),PSI,SMSQ
C *,Q0,Q(50),LATT,MUJJ(50),MCLT,VCLT,SDLT
C *,VAR,SKEW,KURT,LAMJ(50),MUJ(50),Q(7),Q(7),FX(7)
C *,PRJ(10,7),PQT(7),ALPHA,BETSQ,RO,F,VARP(7),SDP(7)
C *,ER1(50),ER2(50),ER3(50),E1R,E2R,E3R,E4R
C *,SER,SKN,KTS,DEV(50),D4(50),PI,DMI(7),DM2(7),DM3(7)
C *,BIAS,SKY,KTY,SDR,CDF(5,7),CF(7,5)
C INTEGER N,KO(7),KT
C
C PI = 3.1415927
C Q(1) = -1.2816
C Q(2) = -0.6745
C Q(3) = 0.
C Q(4) = 0.6745
C Q(5) = 1.2816
C Q(6) = 1.6449
C Q(7) = 2.3263
C
C ENTER SUBROUTINE VARIABLES
C READ(5,500) KT,N,T,LAMDA,MU
C
C READ FRACTION OF TIME PROCESSOR SPENDS PROCESSING
C DO 5 I=1,N
C   RJ(I) = 1.
C   5 CONTINUE

```

```

PSNO00010
PSNO00020
PSNO00030
PSNO00040
PSNO00050
PSNO00060
PSNO00070
PSNO00080
PSNO00090
PSNO00100
PSNO00110
PSNO00120
PSNO00130
PSNO00140
PSNO00150
PSNO00160
PSNO00170
PSNO00180
PSNO00190
PSNO00200
PSNO00210
PSNO00220
PSNO00230
PSNO00240
PSNO00250
PSNO00260
PSNO00270
PSNO00280
PSNO00290
PSNO00300
PSNO00310
PSNO00320
PSNO00330
PSNO00340
PSNO00350
PSNO00360
PSNO00370
PSNO00380
PSNO00390
PSNO00400
PSNO00410
PSNO00420

```



```

C COMPUTE LAMJ(J) AND MUJ(J)
DO 10 JJ=1,N
LAMJ(JJ) = LAMDA*(N-JJ)
MUJ(JJ) = MU*(JJ-1)*RJ(JJ)/JJ
10 CONTINUE
C COMPUTE PROB. OF BEING IN STATE J,I.E. QJ
DO 13 KK=1,N
LAMQJ(KK) = LAMDA*(N+1-KK)
MUQJ(KK) = MU*RJ(KK)
13 CONTINUE
DO 15 KJ=1,N
QQ(KJ) = 1.
DO 14 KL=1,KJ
QQ(KJ) = QQ(KJ)*LAMQJ(KL)/MUQJ(KL)
14 CONTINUE
15 CONTINUE
QQ = 0.
DO 17 JK=1,N
QO = QO+QQ(JK)
17 CONTINUE
DO 18 IK=1,N
QJ(IK) = QQ(IK)/QO
18 CONTINUE
C CALL SUBROUTINE SIMULATING RESPONSE TIME
CALL SIMPSN(KT,N,T,LAMJ,MUJ,RJ,RTJ)
WRITE(6,600) N,KT,I,LAMDA,MU
DO 990 J=1,N
ER1(J) = 0.
ER2(J) = 0.
ER3(J) = 0.
ER4(J) = 0.
990 CONTINUE
DO 1000 J=1,N
K=1,KT
ER1(J) = ER1(J)+RTJ(J,K)/FLOAT(KT)
995 CONTINUE
EIR = 0.
DO 1010 J=1,N
EIR = EIR+ER1(J)*QJ(J)
1010 CONTINUE
DO 1020 J=1,N
K=1,KT
ER2(J) = ER2(J)+(RTJ(J,K)-ER1(J))*2/FLOAT(KT)
ER3(J) = ER3(J)+(RTJ(J,K)-ER1(J))*3/FLOAT(KT)

```



```

1019 ER4(J) = ER4(J)+(RTJ(J,K)-ER1(J))**4/FLCAT(KT)
1020 CONTINUE
CONTINUE
E2R = 0.
E3R = 0.
E4R = 0.
SER = 0.
DO 1030 J=1,N
DEV(J) = ER1(J)-E1R
E2R = (ER2(J)+DEV(J)**2)*QJ(J)
E3R = (ER3(J)+3*ER2(J)*DEV(J)+DEV(J)**3)*QJ(J)
E4R = (ER4(J)+4*ER3(J)*DEV(J)+6*ER2(J)**2+DEV(J)**4)
SER = SER+ER2(J)*QJ(J)**2/FLOAT(KT)
1030 CONTINUE
SDR = E2R*.5
SKN = E2R/E2R**.5
KTS = E4R/E2R**2 - 3
SER = SQRT(SER)
WRITE(6,663) E1R, SER, SDR, E3R, SKN, E4R, KTS
C CALL SUBROUTINE TO APPROXIMATE MEAN RESPONSE TIME AND VARIANCE
CALL CLTR(N, LAMDA, MU, RJ, PSI, SMSQ)
DO 150 LM=1,5
IF (LM.EQ. 1) .OR. (LM.EQ. 2) ) GO TO 300
IF (LM.EQ. 3) GO TO 150
MCLT = T/PSI
VCLT = T*SMSQ/PSI**3
WRITE(6,660)
GO TO 200
150 IF (LM.NE. 4) GO TO 160
C APPROX. BY HEAVY TRAFFIC ANALYSIS I.
WRITE(6,661)
ALPHA = N-MU/LAMDA
BETSQ = 2*MU/(LAMDA**2*ALPHA)
MCLT = T*ALPHA
VCLT = T*BETSQ
GO TO 200
160 WRITE(6,662)
C APPROX. BY HEAVY TRAFFIC ANALYSIS II.
MCLT = T*(N-MU/LAMDA)
RO = LAMDA*N-MU
F = 1-(1-EXP(-RO*T))/(RO*T)
VCLT = 2*MU*T*(F/(LAMDA*RO))
200 SDLT = VCLT**.5
WRITE(6,601) MCLT, SDLT
C COMPUTE QUANTILES OF NORMAL DIST. WITH MEAN MCLT AND VARIANCE VCLT
C FOR .10, .25, .50, .75, .90, .95, .99

```

PSN00910
PSN00920
PSN00930
PSN00940
PSN00950
PSN00960
PSN00970
PSN00980
PSN00990
PSN01000
PSN01010
PSN01020
PSN01030
PSN01040
PSN01050
PSN01060
PSN01070
PSN01080
PSN01090
PSN01100
PSN01110
PSN01120
PSN01130
PSN01140
PSN01150
PSN01160
PSN01170
PSN01180
PSN01190
PSN01200
PSN01210
PSN01220
PSN01230
PSN01240
PSN01250
PSN01260
PSN01270
PSN01280
PSN01290
PSN01300
PSN01310
PSN01320
PSN01330
PSN01340
PSN01350
PSN01360
PSN01370
PSN01380


```

C
C
C
C
DO 210 I=1,7
  QT(I) = MCLT+SDLT*Q(I)
210 CONTINUE
WRITE(6,602) QT
C
C
C
C
WRITE(6,603)
C
C
C
C
COMPUTE RELATIVE FREQUENCIES FROM SIMULATION DATA
DO 50 I=1,N
  DO 19 L=1,7
    KO(L) = 0
  CONTINUE
DO 46 J=1,KT
  IF (RTJ(I,J) .GT. QT(1)) GO TO 30
  DO 100 K=1,7
    KC(K) = KO(K)+1
  CONTINUE
  GO TO 45
30 IF (RTJ(I,J) .GT. QT(2)) GO TO 31
  DO 101 K=2,7
    KC(K) = KO(K)+1
  CONTINUE
  GO TO 45
31 IF (RTJ(I,J) .GT. QT(3)) GO TO 32
  DO 102 K=3,7
    KC(K) = KO(K)+1
  CONTINUE
  GO TO 45
32 IF (RTJ(I,J) .GT. QT(4)) GO TO 33
  DO 103 K=4,7
    KC(K) = KO(K)+1
  CONTINUE
  GO TO 45
33 IF (RTJ(I,J) .GT. QT(5)) GO TO 34
  DO 104 K=5,7
    KC(K) = KO(K)+1
  CONTINUE
  GO TO 45
34 IF (RTJ(I,J) .GT. QT(6)) GO TO 35
  DO 105 K=6,7
    KC(K) = KO(K)+1
  CONTINUE
  GO TO 45
35 IF (RTJ(I,J) .GT. QT(7)) GO TO 45
  DO 45 KO(7) = KC(7)+1
  CONTINUE
46 CONTINUE
C

```

```

PSNO1390
PSNO1400
PSNO1410
PSNO1420
PSNO1430
PSNO1440
PSNO1450
PSNO1460
PSNO1470
PSNO1480
PSNO1490
PSNO1500
PSNO1510
PSNO1520
PSNO1530
PSNO1540
PSNO1550
PSNO1560
PSNO1570
PSNO1580
PSNO1590
PSNO1600
PSNO1610
PSNO1620
PSNO1630
PSNO1640
PSNO1650
PSNO1660
PSNO1670
PSNO1680
PSNO1690
PSNO1700
PSNO1710
PSNO1720
PSNO1730
PSNO1740
PSNO1750
PSNO1760
PSNO1770
PSNO1780
PSNO1790
PSNO1800
PSNO1810
PSNO1820
PSNO1830
PSNO1840
PSNO1850
PSNO1860

```



```

C      DO 48 K=1,7
C      PRJ(I,K) = FLOAT(KO(K))/(FLOAT(KT))
C      CONTINUE
C      WRITE(6,604) I,PR
C      CONTINUE
C      DO 60 K=1,7
C      PQT(K) = 0.
C      DO 59 I=1,N
C      PQT(I,K) = PQT(K)+PRJ(I,K)*QJ(I)
C      CONTINUE
C      WRITE(6,605) PQT
C      DO 62 J=1,7
C      CDF(LM,J) = PQT(J)
C      CONTINUE
C      DO 65 K=1,7
C      VARP(K) = 0.
C      DO 64 I=1,N
C      VARP(I,K) = VARP(K)+(PQT(K)*(1-PQT(K))*QJ(I)**2)/(FLOAT(KT))
C      CONTINUE
C      SDP(K) = VARP(K)**0.5
C      CONTINUE
C      WRITE(6,606) SDP
C      GO TO 70
C APPROX. BY EDGEWORTH EXPANSION
C 300 IF (LM.NE.1) GO TO 305
C      SKY = SKN
C      KTY = KTS
C      WRITE(6,666)
C      GO TO 307
C      I/PSI
C      MCLT = T*SMSQ/PSI**3
C      VCLT = T*VCLT**0.5
C      WRITE(6,664) MCLT,SDLT
C      BIAS = SKN+(3*BBIAS**E2R+6*BBIAS**E3R+6*BBIAS**E4R)/E2R**1.5
C      SKY = KTS+(4*BBIAS**E2R+6*BBIAS**E3R+6*BBIAS**E4R)/E2R**2
C      DO 310 I=1,7
C      DM1(I) = (Q(I)**2-1)*SKY/6.
C      DM2(I) = (Q(I)**3-3*Q(I))*KTY/24.
C      DM3(I) = (Q(I)**5-10*Q(I)**3+15*Q(I))*SKY**2/72.
C      FX(I) = (DM1(I)+DM2(I)+DM3(I))*EXP(-(Q(I)**2/2.)) / (SQRT(2*PI))
C      CONTINUE
C      FX(1) = 0.10 - FX(1)
C      FX(2) = 0.25 - FX(2)
C      FX(3) = 0.50 - FX(3)

```

```

PSNO1870
PSNO1880
PSNO1890
PSNO1900
PSNO1910
PSNO1920
PSNO1930
PSNO1940
PSNO1950
PSNO1960
PSNO1970
PSNO1980
PSNO1990
PSNO2000
PSNO2010
PSNO2020
PSNO2030
PSNO2040
PSNO2050
PSNO2060
PSNO2070
PSNO2080
PSNO2090
PSNO2100
PSNO2110
PSNO2120
PSNO2130
PSNO2140
PSNO2150
PSNO2160
PSNO2170
PSNO2180
PSNO2190
PSNO2200
PSNO2210
PSNO2220
PSNO2230
PSNO2240
PSNO2250
PSNO2260
PSNO2270
PSNO2280
PSNO2290
PSNO2300
PSNO2310
PSNO2320
PSNO2330
PSNO2340

```


PSN0 2350
PSN0 2360
PSN0 2370
PSN0 2380
PSN0 2390
PSN0 2400
PSN0 2410
PSN0 2420
PSN0 2430
PSN0 2440
PSN0 2450
PSN0 2460
PSN0 2470
PSN0 2480
PSN0 2490
PSN0 2500
PSN0 2510
PSN0 2520
PSN0 2530
PSN0 2540
PSN0 2550
PSN0 2560
PSN0 2570
PSN0 2580
PSN0 2590
PSN0 2600
PSN0 2610
PSN0 2620
PSN0 2630
PSN0 2640
PSN0 2650
PSN0 2660
PSN0 2670
PSN0 2680
PSN0 2690
PSN0 2700
PSN0 2710
PSN0 2720
PSN0 2730
PSN0 2740
PSN0 2750
PSN0 2760
PSN0 2770
PSN0 2780
PSN0 2790
PSN0 2800
PSN0 2810
PSN0 2820

```
FX(4) = 0.75  
FX(5) = 0.90  
FX(6) = 0.95  
FX(7) = 0.99  
WRITE(6,665) FX  
DO 68 J=1,7  
  CDF(LM,J) = FX(J)  
68 CONTINUE  
70 CONTINUE  
  I=1,7  
  DO 75 J=1,5  
    CF(I,J) = CDF(J,I)  
  CONTINUE  
74 CONTINUE  
75 WRITE(6,670) CF  
  STOP  
C FORMATS  
500 FORMAT(2I5,3F10.4) OF TERMINALS : , I4,5X,  
600 *NO. OF SIMULATION : , F10.4, I4,5X,  
  *REQUIRED WORK : , T = , F10.4, I4,5X,  
  *//3X, LAMDA = , F10.4, I4,5X, STANDARD DEV. : , F13.6,  
601 FORMAT(//10X, MEAN : , F13.6, I4,5X, DISTRIBUTION : , /7X, : , 10 Q, : , 5X,  
602 *//3X, QUANTILES OF NORMAL : , F13.6, I4,5X, : , 95 Q, : , 5X, : , 99 Q,  
  *//3X, Q, : , 5X, : , 75 Q, : , 5X, : , 90 Q, : , 5X, : , 99 Q,  
C 603 *//3X,7F10.4) CONDITIONAL RELATIVE FREQUENCIES FROM SIMULATION  
C 604 *//3X,5.5X, P .25,5X, P .50,5X, P .75,5X, P .90,5X,  
C 605 *//3X,5.5X, P .99,5X, P .50,5X, P .75,5X, P .90,5X, P .99,5X,  
  *//3X,5.5X, P .50,5X, P .75,5X, P .90,5X, P .99,5X,  
606 *//3X,7F10.4) STANDARD DEVIATION OF RESPONSE TIME : , F13.6  
660 FORMAT(//3X, APPROX. BY CENTRAL LIMIT THEOREM. : ,  
661 *//3X, APPROX. BY HEAVY TRAFFIC ANALYSIS I. : ,  
662 *//3X, APPROX. BY HEAVY TRAFFIC ANALYSIS II. : ,  
663 *//3X, SIMULATION MEAN RESPONSE : , F13.6  
  *//3X, ERROR OF THE MEAN RESPONSE : , F13.6  
  *//3X, STANDARD DEVIATION OF RESPONSE TIME : , F13.6  
  *//3X, STANDARD DEVIATION OF RESPONSE TIME : , F13.6  
  *//3X, THIRD CENTRAL MOMENT OF RESPONSE TIME : , F13.6  
  *//3X, SKWNESS OF RESPONSE TIME : , F13.6  
  *//3X, FCURTOSIS OF RESPONSE TIME : , F13.6  
664 *//3X, KURTOSIS OF RESPONSE TIME : , F13.6  
665 *//3X, MEAN : , F13.6, I4,5X, BY EDGEWORTH EXPANSION //3X,  
  *//3X, CLT. : , F13.6, I4,5X, STD. DEV. : , F13.6, I4,5X, : , 50, : , 6X, : , 50, : , 6X, : , 41
```



```

666 FORMAT(///3X, ' EDG E W O R T H E X P A N S I O N W I T H A C T U A L M E A N ' )
670 FORMAT(///3X, ' S U M M A R Y O F N O R M A L A P P R O X I M A T I O N S ' , / / 3 X ,
* . 4 X , 0 . 1 0 , 6 X , 0 . 2 5 , 6 X , 0 . 5 0 , 6 X , 0 . 7 5 , 6 X ,
* . 9 0 , 6 X , 0 . 9 9 , / / 3 X , ' E W . W / O . C L T : ' , / / 3 X ,
* . E W . W / C L T : ' , / / 3 X , ' C L T A P P R O X : ' , / / 3 X ,
* . H V Y T R A F . I : ' , / / 3 X , ' H V Y T R A F . I I : ' , / / 3 X ,
END
C ***** TO SIMULATE RESPONSE TIME *****
C SUBROUTINE *****
C ***** SUBROUTINE SIMPSN(KT,N,T,LAMJ,MUJ,RJ,RTJ) *****
C REAL *4 W0,C0,EXP1,LAMJ(50),MUJ(50),RJ(50),RTJ(50,1000),
* INTEGER IX1,IX2,N,KT
IX1 = 182394
IX2 = 328653
C C O N D I T I O N A L R E S P O N S E T I M E
DO 100 J=1,N
DO 99 I=1,KT
W0 = T
C0 = 0.
K = J
C C G E N E R A T E E X P ( L A M D A * K * ( N - J ) / R J ( J ) )
30 CALL LEXPN(IX1,EXP1,1,1,0)
C C G E N E R A T E E X P ( M U * ( J - 1 ) )
CALL LEXPN(IX2,EXP2,1,1,0)
C
IF (K.NE.1) GO TO 60
EXP1 = EXP1*RJ(K)/(K*LAMJ(K))
EXPJ = EXP1
GC TO 80
IF (K.NE.N) GO TO 70
EXP2 = EXP2*RJ(K)/(K*MUJ(K))
EXPJ = EXP2
GC TO 80
EXP1 = EXP1*RJ(K)/(K*LAMJ(K))
EXP2 = EXP2*RJ(K)/(K*MUJ(K))
C C G E T T H E M I N I M U M O F E X P 1 & E X P 2
EXPJ = AMINI(EXP1,EXP2)
80 IF (EXPJ.LT.W0) GO TO 81
RTJ(J,I) = C0+K*W0/RJ(K)
GC TO 90
81 C0 = C0+EXPJ*K/RJ(K)

```

PSNO 2830
PSNO 2840
PSNO 2850
PSNO 2860
PSNO 2870
PSNO 2880
PSNO 2890
PSNO 2900
PSNO 2910
PSNO 2920
PSNO 2930
PSNO 2940
PSNO 2950
PSNO 2960
PSNO 2970
PSNO 2980
PSNO 2990
PSNO 3000
PSNO 3010
PSNO 3020
PSNO 3030
PSNO 3040
PSNO 3050
PSNO 3060
PSNO 3070
PSNO 3080
PSNO 3090
PSNO 3100
PSNO 3110
PSNO 3120
PSNO 3130
PSNO 3140
PSNO 3150
PSNO 3160
PSNO 3170
PSNO 3180
PSNO 3190
PSNO 3200
PSNO 3210
PSNO 3220
PSNO 3230
PSNO 3240
PSNO 3250
PSNO 3260
PSNO 3270
PSNO 3280
PSNO 3290
PSNO 3300


```

      W0 = W0-EXPJ
      IF (EXPJ.EQ. EXP1) GO TO 82
      IF (EXPJ.EQ. EXP2) K=K-1
      GO TO 83
      K = K+1
      IX1 = IX1+1
      IX2 = IX2+1
      GO TO 30
      CONTINUE
90 CONTINUE
99 CONTINUE
100 RETURN
      END
C *****
C SUBROUTINE TO APPROX. RESPONSE TIME BY CENTRAL LIMIT THEOREM
C *****
      SUBROUTINE CLTRT(N,LAMDA,MU,RJ,PSI,SMSQ)
      REAL *4 LAMDA,MU,LAMJP(10),FJ(10),IDN(10),MA(10,10)
      *,PSI,NUJ(25),GAMMA,ML(10,10),MPPI(10,10),SINVS(10,10)
      *,MSUM(10,10),MZ(10,10),FMTJ(1,10),SMSQ,SINV(10,10)
      *,C(10,1),D(10,1),E(1,1),EE,PP(10),WKAREA(2650),RJ(10)
      *,IER1,IER2,IER3,IER4
      *,INTEGER N,NP
      COMPUTE F.(J)
      DO 200 I=1,N
      FJ(I) = RJ(I)/I
      200 CONTINUE
      COMPUTE LAMCA.(J) AND MU.(J)
      NP = N-1
      DO 205 J=1,N
      LAMJP(J) = LAMDA*(N-J)
      MUJP(J) = MU*(J-1)/J
      205 CONTINUE
      COMPUTE PI(J)_VECTOR
      DO 210 I=1,NP
      PP(I) = I
      DO 209 IJ=1,II
      PP(IJ) = PP(IJ)*LAMJP(IJ)/MUJP(IJ+1)
      209 CONTINUE
      PI0 = 0
      DO 212 IK=1,NP
      PI0 = PI0+PP(IK)
      212 CONTINUE

```


PSN03790
 PSN03800
 PSN03810
 PSN03820
 PSN03830
 PSN03840
 PSN03850
 PSN03860
 PSN03870
 PSN03880
 PSN03890
 PSN03900
 PSN03910
 PSN03920
 PSN03930
 PSN03940
 PSN03950
 PSN03960
 PSN03970
 PSN03980
 PSN03990
 PSN04000
 PSN04010
 PSN04020
 PSN04030
 PSN04040
 PSN04050
 PSN04060
 PSN04070
 PSN04080
 PSN04090
 PSN04100
 PSN04110
 PSN04120
 PSN04130
 PSN04140
 PSN04150
 PSN04160
 PSN04170
 PSN04180
 PSN04190
 PSN04200
 PSN04210
 PSN04220
 PSN04230
 PSN04240
 PSN04250
 PSN04260

```

    PIO = 1/(1+PIO)
    DO 213 IL=1,NP
      PPI(IL) = PIO*PP(IL)
    213 CONTINUE
    DO 220 IM=1,N
      IF (IM.NE. 1) GO TO 215
      PPI(IM) = PIO
      GO TO 219
    215 PPI(IM) = PPI(IM-1)
    219 CONTINUE
    220 CONTINUE
  C COMPUTE PSI
    PSI = 0
    DO 230 IN=1,N
      PSI = PSI+FJ(IN)*PPI(IN)
    230 CONTINUE
  C COMPUTE SIGMA SQUARE
    DO 250 JI=1,N
      IF (JI.NE. 1) GO TO 240
      NUJ(JI) = LAMJP(JI)
      GO TO 249
    240 IF (JI.NE. N) GO TO 245
      NUJ(JI) = MUJP(JI)
      GO TO 249
    245 NUJ(JI) = LAMJP(JI)+MUJP(JI)
    249 CONTINUE
    250 CONTINUE
    GAMMA = 0
    DO 251 KK=1,N
      GAMMA = AMAX1(GAMMA,NUJ(KK))
    251 CONTINUE
    DO 270 I=1,N
      ML(I,J) = PPI(J)
      IF (I.NE. J) GO TO 255
      MPI(I,J) = PPI(J)
      IDN(I,J) = 1
      MA(I,J) = 1-NUJ(J)/GAMMA
      GC TO 260
    255 MPI(I,J) = 0
      IDN(I,J) = 0
      IF (J.NE. (I-1)) GO TO 256
      MA(I,J) = MUJP(I)/GAMMA
      GC TO 260
    256 IF (J.NE. (I+1)) GO TO 257
      MA(I,J) = LAMJP(I)/GAMMA
  
```



```

257      GC TO 260
260      MA(I,J) = 0.
270      CONTINUE
C
DO 280 K=1,N
DO 275 L=1,N
MSUM(K,L) = IDN(K,L)-MA(K,L)+ML(K,L)
275      CONTINUE
280      CONTINUE
C
CALL LINV2F(MSUM,N,N,SINV,O,WKAREA,IER1)
DO 290 II=1,N
DO 285 JJ=1,N
MZ(II,JJ) = (SINV(II,JJ)-ML(II,JJ))/GAMMA
285      CONTINUE
290      CONTINUE
C
DO 300 IJ=1,N
FMJ(IJ,IJ) = FJ(IJ)
FMTJ(I,IJ) = FJ(IJ)
300      CONTINUE
CALL VMULFF(MZ,FMJ,N,N,1,N,N,C,N,IER2)
CALL VMULFF(MP,I,C,N,I,N,I,N,D,N,IER3)
CALL VMULFF(FMTJ,D,I,N,I,I,N,E,I,IER4)
C
EE = E(1,1)
SM SQ = 2*EE
RETURN
END

```

```

PSN04270
PSN04280
PSN04290
PSN04300
PSN04310
PSN04320
PSN04330
PSN04340
PSN04350
PSN04360
PSN04370
PSN04380
PSN04390
PSN04400
PSN04410
PSN04420
PSN04430
PSN04440
PSN04450
PSN04460
PSN04470
PSN04480
PSN04490
PSN04500
PSN04510
PSN04520
PSN04530
PSN04540
PSN04550
PSN04560

```


APPENDIX B

SIMULATION PROGRAM FOR TWO-TYPE MODEL

```

C**PROGRAM TO SIMULATE RESPONSE TIME AND COMPARE TC NORMAL APPROX.*
C**FOR TWO-TYPE JOB MODEL*****
C**VARIABLES AND CONSTANTS*****
C**M = NUMBER OF TERMINALS OF WORK TYPE 1 TERMINAL
C**N = INITIALLY REQUIRED A JOB ARRIVES AT TYPE 2 TERMINAL
C**LAM1 = RATE AT WHICH A JOB ARRIVES AT TYPE 1 TERMINAL
C**LAM2 = RATE AT WHICH WAITING JOBS AT TYPE 2 TERMINAL ARE SERVED
C**MU1 = RATE AT WHICH WAITING JOBS AT TYPE 1 TERMINAL ARE SERVED
C**MU2 = RATE AT WHICH WAITING JOBS AT TYPE 2 TERMINAL ARE SERVED
C**REAL T, LAM1, LAM2, MU1, MU2, LJ1(6), LJ2(6), MJ1(6), MJ2(6), SS3
C**MRT, SE, X1(6,6), V1(6,6), A1(6,6), B1(6,6), QH(7), PHT(7), SKHT
C**RT3, RT4, Q1, NUJ1(6), C1(6,6), SS2, SS4, VAR, SDV, SKEW, KURT, KUHT
C**XBLT, SCLT, VARP(7), SDP(7), PHI(7), VPH(7), HTXB, HTSD
C**INTEGER M, N, M1, N1, KT
C**KT = 300
C**ENTER SUBROUTINE VARIABLES
C**READ(5,500) M, N, T, LAM1, LAM2, MU1, MU2
C**M1 = M+1
C**N1 = N+1
C**CALL SUBROUTINE COMPUTING STEADY STATE DISTRIBUTION
C**CALL SST(M, N, LAM1, LAM2, MU1, MU2, NU1,
C**SUBROUTINE SIMULATING CONDITIONAL RESPONSE TIME
C**CALL SIMRT(M, N, T, LAM1, LAM2, MU1, MU2, X1, A1, B1, C1
C**PRINT(6,601) M, LAM1, MU1, N, LAM2, MU2, T
C**WRITE(6,601) M, LAM1, MU1, N, LAM2, MU2, T
C**DO 270 I=1, M1
C**DO 260 J=1, N1
C**MRT = MRT+NU1(I,J)*X1(I,J)
C**CONTINUE
C**DO 270 I=1, M1
C**DO 272 J=1, N1
C**V1(I,J) = X1(I,J)-MRT
C**CONTINUE
C**DO 271 I=1, M1
C**DO 272 J=1, N1
C**V1(I,J) = X1(I,J)-MRT
C**CONTINUE
C**VAR = 0.

```

APP00010
APP00020
APP00030
APP00040
APP00050
APP00060
APP00070
APP00080
APP00090
APP00100
APP00110
APP00120
APP00130
APP00140
APP00150
APP00160
APP00170
APP00180
APP00190
APP00200
APP00210
APP00220
APP00230
APP00240
APP00250
APP00260
APP00270
APP00280
APP00290
APP00300
APP00310
APP00320
APP00330
APP00340
APP00350
APP00360
APP00370
APP00380
APP00390
APP00400
APP00410
APP00420


```

RT3 = 0.
RT4 = 0.
SE = 0.
DO 274 J=1,MI
  SE=SE+A1(I,J)*NU1(I,J)**2/FLOAT(KT)
  VAR = VAR+(A1(I,J)+V1(I,J)**2)*NU1(I,J)
  RT3 = RT3+(A1(I,J)+3*A1(I,J)*V1(I,J)+V1(I,J)**3)*NU1(I,J)**4
  D5=C1(I,J)+4*B1(I,J)*V1(I,J)+6*A1(I,J)*V1(I,J)**2+V1(I,J)**4
  RT4 = RT4+D5*NU1(I,J)
274 CONTINUE
275 CONTINUE
  SE=SQR(T(VAR))
  SDV = SE
  SKEW = SE*RT3/VAR**1.5
  KURT = SE*RT4/VAR**2
  DO 50 K=2,7
    QH(K) = (QH(K)+QH(K-1))/2.
    PHT(K) = PHI(K)-PHI(K-1)
50 CONTINUE
  PH1(1) = PHT(1)
  SS2 = 0.
  SS3 = 0.
  SS4 = 0.
  DO 60 I=1,7
    SS2 = SS2+(QH(I)-HTXB)**2*PHT(I)
    SS3 = SS3+(QH(I)-HTXB)**2*PHT(I)
    SS4 = SS4+(QH(I)-HTXB)**4*PHT(I)
60 CONTINUE
  SS3/SS2**1.5
  SS4/SS2**2 - 3.
  SDV,RT3, SKEW,RT4,KURT
C CALL SUBROUTINE TO READ CLT MEAN AND VARIANCE
  CALL APPROX(XBLT,SDLT)
  XBLT = XBLT*PI
  SDLT = SDLT*PI
  WRITTE(6,602) M,RT,SE,SDV,RT3, SKEW,RT4,KURT
  DO 276 I=1,7
    SDP(I) = SQRT(VARP(I))
    VPH(I) = SQRT(VPH(I))
276 CONTINUE
  WRITTE(6,604) PR1,SDP
  WRITTE(6,605) HTXB,HTSD,SKHT,KUHT
  WRITTE(6,606) PH1,VPH
C
C STOP
C FORMAT S
C 500 FORMAT(2I5,5F10.4)

```

```

APP000430
APP000440
APP000450
APP000460
APP000470
APP000480
APP000490
APP000500
APP000510
APP000520
APP000530
APP000540
APP000550
APP000560
APP000570
APP000580
APP000590
APP000600
APP000610
APP000620
APP000630
APP000640
APP000650
APP000660
APP000670
APP000680
APP000690
APP000700
APP000710
APP000720
APP000730
APP000740
APP000750
APP000760
APP000770
APP000780
APP000790
APP000800
APP000810
APP000820
APP000830
APP000840
APP000850
APP000860
APP000870
APP000880
APP000890
APP000900

```



```

601 FORMAT(1, //5X, 'NG. OF TYPE I TERMINALS : ', I5, 5X,
* LAMDA I = ', F10.4//5X,
* NO. OF TYPE I TERMINALS : ', I5, 5X, 'LAMDA II = ', F10.4,
* 5X, 'MU I = ', F10.4//5X,
* REQUIRED WORK TIME = ', F13.6)
602 * STD. ERROR OF MEAN RESPONSE TIME : ', F13.6//5X,
* STANDARD DEVIATION OF MEAN RESPONSE TIME : ', F13.6//5X,
* THIRD CENTRAL MOMENT OF RESPONSE TIME : ', F13.6//5X,
* SKEWNESS OF RESPONSE TIME : ', F13.6//5X,
* FOURTH CENTRAL MOMENT OF RESPONSE TIME : ', F13.6//5X,
* KURTOSIS OF RESPONSE TIME : ', F13.6)
603 * FORMAT(//3X, 'APPROXIMATION BY CENTRAL LIMIT THEOREM',
//3X, 'MEAN : ', F13.6, 5X, 'CLT. DEV. : ', F13.6)
604 * FORMAT(//3X, 'NORMAL CDF : ', F13.6, 5X, 'HTA. KURTOSIS : ', F13.6)
* 6X, '0.95 : ', F13.6, 5X, '0.99 : ', F13.6, 5X, '0.50',
* 7F10.4//3X, 'STD. ERROR : ', F13.6, 5X, 'HTA APPROX. : '
605 * FORMAT(//3X, 'NORMAL MEAN : ', F13.6, 5X, 'HTA. KURTOSIS : ', F13.6)
//3X, 'SKEWNESS : ', F13.6, 5X, 'HTA. KURTOSIS : ', F13.6)
606 * FORMAT(//3X, 'NORMAL CDF : ', F13.6, 5X, 'HTA APPROX. : '
* 6X, '0.95 : ', F13.6, 5X, '0.99 : ', F13.6, 5X, '0.50',
* 7F10.4//3X, 'STD. ERROR : ', F13.6, 5X, 'HTA APPROX. : '
* END
C * SUBROUTINE TO COMPUTE STEADY-STATE DISTRIBUTION FOR TYPE I JOB
C * * * * *
SUBROUTINE SST(M, N, LAM1, LAM2, MU1, MU2, NU1, NU2, PI(6,6), TUI(6,6), Q1
INTEGER+1 I=1, N
NI = N+1
DO 100 FI = 1, NI
DO 100 FJ = 1, NI
FM = FJ - FI
FN = FJ + FI
PP(I, J) = PP(I, J) * GAMMA(FM+1.) * GAMMA(FN+1.) * (LAMI/MUI) ** I
PP(I, J) = PP(I, J) / (GAMMA(FM+1.) * GAMMA(FN+1.) * GAMMA(FJ+1.))
PP(I, J) = PP(I, J) * GAMMA(FM-FI+1.) * GAMMA(FN-FJ+1.)
PP(I, J) = PP(I, J) * GAMMA(FI+FJ+1.)
CONTINUE
DO 102 I=2, NI
DO 101 J=2, NI
PI(I, J) = PP(I, J)
CONTINUE
END

```

```

APP000510
APP000920
APP000530
APP000940
APP000950
APP000960
APP000970
APP000580
APP000990
APP010000
APP010010
APP010020
APP010030
APP010040
APP010050
APP010060
APP010070
APP010080
APP010090
APP010100
APP010110
APP010120
APP010130
APP010140
APP010150
APP010160
APP010170
APP010180
APP010190
APP010200
APP010210
APP010220
APP010230
APP010240
APP010250
APP010260
APP010270
APP010280
APP010290
APP010300
APP010310
APP010320
APP010330
APP010340
APP010350
APP010360
APP010370
APP010380

```


APP01390
 APP01400
 APP01410
 APP01420
 APP01430
 APP01440
 APP01450
 APP01460
 APP01470
 APP01480
 APP01490
 APP01500
 APP01510
 APP01520
 APP01530
 APP01540
 APP01550
 APP01560
 APP01570
 APP01580
 APP01590
 APP01600
 APP01610
 APP01620
 APP01630
 APP01640
 APP01650
 APP01660
 APP01670
 APP01680
 APP01690
 APP01700
 APP01710
 APP01720
 APP01730
 APP01740
 APP01750
 APP01760
 APP01770
 APP01780
 APP01790
 APP01800
 APP01810
 APP01820
 APP01830
 APP01840
 APP01850
 APP01860

```

101 CONTINUE
102 CONTINUE
DO 103 J=1,N
  PI(I,(J+1))=GAMMA((FLOAT(N+1)))/GAMMA((FLOAT(N-J+1)))
  PI(I,(J+1))=PI(I,(J+1))*((LAM2/MU2)**J)
103 CONTINUE
DO 104 I=1,M
  PI((I+1),1)=GAMMA((FLOAT(M+1)))/GAMMA((FLOAT(M-I+1)))
  PI((I+1),1)=PI((I+1),1)*((LAM1/MU1)**I)
104 CONTINUE
PI(1,1)=0.
DO 120 I=1,M1
  DO 110 J=1,N1
    PI(I,1)=PI(1,1)+PI(I,J)
110 CONTINUE
120 CONTINUE
PI(1,1)=1./((PI(1,1)+1.))
DO 140 I=1,M1
  DO 130 J=1,N1
    IF (I.EQ.1) .AND. (J.EQ.1) GO TO 130
    PI(I,J)=PI(I,1)*PI(I,J)
130 CONTINUE
140 CONTINUE
DO 180 I=1,M1
  DO 170 J=1,N1
    TUI(I,J)=PI(I,J)*LAM1*(M1-I)
170 CONTINUE
180 CONTINUE
Q1=0.
DO 200 I=1,M1
  DO 190 J=1,N1
    Q1=Q1+TUI(I,J)
190 CONTINUE
200 CONTINUE
Q1=1./Q1
DO 220 I=1,M1
  DO 210 J=1,N1
    TUI(I,J)=TUI(I,J)*Q1
210 CONTINUE
220 CONTINUE
DO 225 J=1,N1
  NUI(1,J)=0.
225 CONTINUE
DO 232 I=2,M1
  DO 231 J=1,N1
    NUI(I,J)=TUI((I-1),J)
231 CONTINUE
232 CONTINUE
  
```


235 CONTINUE
RETURN
END

```

C *** TO SIMULATE RESPONSE TIME AND COMPARISON TO CLT ***
C SUBROUTINE SIMRT(M,N,T,LAM1,LAM2,MU1,MU2,X1,A1,B1,C1
* ,P,VAR,PHI,VPH,HTXB,HTSD,MU1,MU2,MU1,MU2,R1(5,6),XR(1000),AR1(6,6)
* ,LJ1(6),LJ2(6),MJ1(6),MJ2(6),XP(6),XJ,XL,XM,BRI(5,6),CRI(5,6)
* ,EXPI,EXP2,EXP3,EXP4,X1(6,6),A1(6,6),B1(6,6),C1(6,6)
* ,QU(7),QI(7),PK1(7),PK3(7),PRI(7),PHI(7),VARP(7),DMY(7),QH(7)
* ,VPH(7),DNY(7),NUI(6,6),XBLLI,SDLI,HTXB,HTSD
* ,INTEGER IX1,IX2,M,N,M1,N1,KT,K1(7),K3(7)
IX1 = 253766
IX2 = 344921
M1 = M+1
N1 = N+1
KT = 300
DO 10 I=1,M1
LJ1(I) = (M1-I)*LAM1
MJ1(I) = MU1*(I-1)
10 CONTINUE
DO 11 J=1,N1
LJ2(J) = (N1-J)*LAM2
MJ2(J) = MU2*(J-1)
11 CONTINUE
C CALL SUBROUTINE TO COMPUTE STEADY-STATE DISTRIBUTION
C STD. NORMAL QUANTILES (.10,.25,.50,.75,.90,.95,.99)
QU(1) = -1.2816
QU(2) = -0.6745
QU(3) = 0.
QU(4) = 0.6745
QU(5) = 1.2816
QU(6) = 1.6449
QU(7) = 2.3263
C CALL SUBROUTINE TO READ CLT MEAN AND VARIANCE
C XBLT = XBLT*HT
C SDLT = SDLT*HT
C CALL SUBROUTINE TO COMPUTE HEAVY TRAFFIC MEAN AND VARIANCE
C HTSD = HTSD
DO 13 I=1,XBLT+SDLT*QU(I)
PRI(I) = 0.

```

APP01870
APP01880
APP01890
APP01900
APP01910
APP01920
APP01930
APP01940
APP01950
APP01960
APP01970
APP01980
APP01990
APP02000
APP02010
APP02020
APP02030
APP02040
APP02050
APP02060
APP02070
APP02080
APP02090
APP02100
APP02110
APP02120
APP02130
APP02140
APP02150
APP02160
APP02170
APP02180
APP02190
APP02200
APP02210
APP02220
APP02230
APP02240
APP02250
APP02260
APP02270
APP02280
APP02290
APP02300
APP02310
APP02320
APP02330
APP02340

APP02350
 APP02360
 APP02370
 APP02380
 APP02390
 APP02400
 APP02410
 APP02420
 APP02430
 APP02440
 APP02450
 APP02460
 APP02470
 APP02480
 APP02490
 APP02500
 APP02510
 APP02520
 APP02530
 APP02540
 APP02550
 APP02560
 APP02570
 APP02580
 APP02590
 APP02600
 APP02610
 APP02620
 APP02630
 APP02640
 APP02650
 APP02660
 APP02670
 APP02680
 APP02690
 APP02700
 APP02710
 APP02720
 APP02730
 APP02740
 APP02750
 APP02760
 APP02770
 APP02780
 APP02790
 APP02800
 APP02810
 APP02820

```

VARP(I) = 0.
QH(I) = HTXB + HTSD * QU(I)
PHI(I) = 0.
VPH(I) = 0.
13 CONTINUE

```

C CONDITIONAL RESPONSE TIME

```

DO 25 NI=1,M
D O 24 NJ=1,N1
RI(NI,NJ) = 0.
ARI(NI,NJ) = 0.
BRI(NI,NJ) = 0.
CRI(NI,NJ) = 0.
24 CONTINUE
25 CONTINUE
DO 250 I=1,M
DO 240 J=1,N1
DO 230 K=1,KT
WO = T
CO = 0.
IO = I
JO = J

```

C GENERATE EXPONENTIAL ARRIVAL AND DEPARTURE TIME

```

30 CALL LEXPN(IX1,EXP,6,1,0)
C
IF ( IO .NE. M ) GO TO 50
EXP1 = 9999.99
IF ( IO .NE. 1 ) GO TO 38
EXP3 = 9999.99
GO TO 39
EXP3 = EXP(3) / ((MUI*(IO-1)))
IF ( JO .NE. 1 ) GO TO 40
EXP2 = EXP(2) / ((LJ2(JO)) * (IO+JO-1))
XL = EXP(5) / ((LJ1(IO+1) + LJ2(JO)) * (IO+JO-1))
EXP4 = 9999.99
XM = EXP3
GO TO 110
IF ( JO .NE. N1 ) GO TO 41
EXP2 = 9999.99
XL = EXP1
EXP4 = EXP(4) / ((MUI*(IO-1)) + MUI2*(JO-1))
XM = EXP(6) / ((MUI*(IO-1)) + MUI2*(JO-1))
GO TO 110
EXP2 = EXP(2) / ((LJ2(JO)) * (IO+JO-1))
XL = EXP(5) / ((LJ1(IO+1) + LJ2(JO)) * (IO+JO-1))
EXP4 = EXP(4) / ((MUI2*(JO-1))
XM = EXP(6) / ((MUI*(IO-1)) + MUI2*(JO-1))

```



```

50  GO TO 110
    IF ( IO .NE. 1 ) GO TO 55
    EXP1 = EXP(1)/(LJ1(I0+1))*(I0+J0-1)
    EXP3 = 9999.99
    GO TO 39
55  EXP1 = EXP(1)/(LJ1(I0+1))*(I0+J0-1)
    EXP3 = EXP(3)/(MUI*(I0-1))
    GO TO 39
110  XJ = AMINI(XL, XM)
    IF ( XJ .LT. W0 ) GO TO 120
    RI(I, J) = RI(I, J)+(C0+W0*(I0+J0-1))/(FLOAT(KT))
    XR(K) = C0+W0*(I0+J0-1)
    GC TO 220
120  CO = CO+XJ*(I0+J0-1)
    W0 = W0-XJ
    IF ( XJ .NE. XL ) GO TO 140
    EXP1 .GT. EXP2 ) GO TO 130
    IF ( IO .EQ. M ) GO TO 160
    IF ( IO = I0+1
        GO TO 160
    IF ( JO .EQ. N1 ) GO TO 160
    JO = JO+1
    GO TO 160
140  IF ( EXP3 .GT. EXP4 ) GO TO 150
    IF ( IO .EQ. 1 ) GO TO 160
    IF ( IO = I0-1
        GO TO 160
    IF ( JO .EQ. 1 ) GO TO 160
    JO = JO-1
    GO TO 30
220  CONTINUE
230  CONTINUE
    DO 231 LI=1, 7
    K1(LI) = 0
    K3(LI) = 0
    CONTINUE
    DO 235 K=1, KT
    DO 235 LI=1, 7
    AR1(I, J)+(XR(K)-RI(I, J))**2/(FLOAT(KT))
    BR1(I, J) = BR1(I, J)-RI(I, J)**3/(FLOAT(KT))
    CR1(I, J) = CR1(I, J)+(XR(K)-RI(I, J))**4/(FLOAT(KT))
    IF ( XR(K) .GT. QI(1) ) GO TO 355
    DO 350 LI=1, 7
    K1(LI) = K1(LI)+1
    CCNTINUE
    GO TO 400
350  IF ( XR(K) .GT. QI(2) ) GO TO 360
355  DC 357 LI=2, 7

```

```

APP02830
APP02840
APP02850
APP02860
APP02870
APP02880
APP02890
APP02900
APP02910
APP02920
APP02930
APP02940
APP02950
APP02960
APP02970
APP02980
APP02990
APP03000
APP03010
APP03020
APP03030
APP03040
APP03050
APP03060
APP03070
APP03080
APP03090
APP03100
APP03110
APP03120
APP03130
APP03140
APP03150
APP03160
APP03170
APP03180
APP03190
APP03200
APP03210
APP03220
APP03230
APP03240
APP03250
APP03260
APP03270
APP03280
APP03290
APP03300

```


APP03310
 APP03320
 APP03330
 APP03340
 APP03350
 APP03360
 APP03370
 APP03380
 APP03390
 APP03400
 APP03410
 APP03420
 APP03430
 APP03440
 APP03450
 APP03460
 APP03470
 APP03480
 APP03490
 APP03500
 APP03510
 APP03520
 APP03530
 APP03540
 APP03550
 APP03560
 APP03570
 APP03580
 APP03590
 APP03600
 APP03610
 APP03620
 APP03630
 APP03640
 APP03650
 APP03660
 APP03670
 APP03680
 APP03690
 APP03700
 APP03710
 APP03720
 APP03730
 APP03740
 APP03750
 APP03760
 APP03770
 APP03780

```

357      KI(LI) = KI(LI)+1
        CCNTINUE
        GO TO 400
360      IF ( XR(K) .GT. QI(3) ) GO TO 365
        DO 363 LI=3,7
          KI(LI) = KI(LI)+1
        CCNTINUE
        GO TO 400
365      IF ( XR(K) .GT. QI(4) ) GO TO 370
        DO 367 IL=4,7
          KI(IL) = KI(IL)+1
        CCNTINUE
        GO TO 400
367      IF ( XR(K) .GT. QI(5) ) GO TO 375
        DO 372 KI=5,7
          KI(KI) = KI(KI)+1
        CCNTINUE
        GO TO 400
370      IF ( XR(K) .GT. QI(6) ) GO TO 380
        DO 377 LK=6,7
          KI(LK) = KI(LK)+1
        CCNTINUE
        GO TO 400
377      IF ( XR(K) .GT. QI(7) ) GO TO 400
        KI(7) = KI(7)+1
        CCNTINUE
        GO TO 400
400      CCNTINUE
235      DO 800 MI=1, KT
        IF ( XR(MI) .GT. QH(1) ) GO TO 710
        DO 705 MK=1,7
          K3(MK) = K3(MK)+1
        CCNTINUE
        GO TO 780
705      IF ( XR(MI) .GT. QH(2) ) GO TO 720
        DO 715 MKK=2,7
          K3(MKK) = K3(MKK)+1
        CCNTINUE
        GO TO 780
710      IF ( XR(MI) .GT. QH(3) ) GO TO 730
        DO 725 MKI=3,7
          K3(MKI) = K3(MKI)+1
        CCNTINUE
        GO TO 780
715      IF ( XR(MI) .GT. QH(4) ) GO TO 740
        DO 735 MKJ=4,7
          K3(MKJ) = K3(MKJ)+1
        CCNTINUE
        GO TO 780
720      IF ( XR(MI) .GT. QH(4) ) GO TO 740
        DO 735 MKJ=4,7
          K3(MKJ) = K3(MKJ)+1
        CCNTINUE
        GO TO 780
725      IF ( XR(MI) .GT. QH(4) ) GO TO 740
        DO 735 MKJ=4,7
          K3(MKJ) = K3(MKJ)+1
        CCNTINUE
        GO TO 780
730      IF ( XR(MI) .GT. QH(4) ) GO TO 740
        DO 735 MKJ=4,7
          K3(MKJ) = K3(MKJ)+1
        CCNTINUE
        GO TO 780
735      IF ( XR(MI) .GT. QH(4) ) GO TO 740
        DO 735 MKJ=4,7
          K3(MKJ) = K3(MKJ)+1
        CCNTINUE
        GO TO 780

```



```

74C  IF ( XR(MI) ,GT, QH(5) ) GO TO 750
      DO 745 MKL=5,7
      K3(MKL) = K3(MKL)+1
      CONTINUE
745  GC TO 780
      IF ( XR(MI) ,GT, QH(6) ) GO TO 760
      DO 755 MKM=6,7
      K3(MKM) = K3(MKM)+1
      CONTINUE
755  GC TO 780
      IF ( XR(MI) ,GT, QH(7) ) GO TO 780
      K3(7) = K3(7)+1
      CONTINUE
800  CONTINUE
      DO 410 KK=1,7
      PK1(KK) = FLOAT(K1(KK))/FLOAT(KT)
      PK3(KK) = FLOAT(K3(KK))/FLOAT(KT)
410  CONTINUE
      DO 420 LM=1,7
      PR1(LM)+PK1(LM)*NUI((I+1),J)
      PH1(LM) = PH1(LM)+PK3(LM)*NUI((I+1),J)
      CONTINUE
240  CONTINUE
250  CONTINUE
      DO 425 LL=1,7
      IF (PR1(LL) ,GT, 1.0) GO TO 200
      GO TO 201
      PR1(LL) = 1.0
      IF (PH1(LL) ,GT, 1.0) PH1(LL) = 1.0
      DO 424 II=1,M
      DO 423 JJ=1,N1
      DMY(LL) = PR1(LL)*(1-PR1(LL))*NUI((II+1),JJ)**2/FLOAT(KT)
      VARP(LL) = VARP(LL)+DMY(LL)
      DNY(LL) = PH1(LL)*(1-PH1(LL))*NUI((II+1),JJ)**2/FLOAT(KT)
      VPH(LL) = VPH(LL)+DNY(LL)
      CONTINUE
423  CONTINUE
424  CONTINUE
425  CONTINUE
490  DO 310 JJJ=1,N1
      XI(1,JJJ) = 0.
      AI(1,JJJ) = 0.
      BI(1,JJJ) = 0.
      CI(1,JJJ) = 0.
      CONTINUE
310  DO 320 IK=1,M
      DO 315 JK=1,N1
      XI((IK+1),JK) = R1(IK,JK)

```

```

APP03790
APP03800
APP03810
APP03820
APP03830
APP03840
APP03850
APP03860
APP03870
APP03880
APP03890
APP03900
APP03910
APP03920
APP03930
APP03940
APP03950
APP03960
APP03970
APP03980
APP03990
APP04000
APP04010
APP04020
APP04030
APP04040
APP04050
APP04060
APP04070
APP04080
APP04090
APP04100
APP04110
APP04120
APP04130
APP04140
APP04150
APP04160
APP04170
APP04180
APP04190
APP04200
APP04210
APP04220
APP04230
APP04240
APP04250
APP04260

```



```

= (SQR((BS2+AS1)**2-4*(AS1*BS2-AS2*BS1))-BS2-AS1)/2.
S0 = (BS2+AS1)/(S0-S1)
S1 = (S0+BS2)/(S0-S1)
B11 = BS1/(S1-S0)
B12 = AS2/(S1-S0)
B21 = (S0+AS1)/(S0-S1)
B22 = (S1+BS2)/(S1-S0)
K11 = (S1+AS1)/(S1-S0)
K12 = AS2/(S0-S1)
K21 = BS1/(S0-S1)
K22 = (AS1-BS2)/(S0*S1)
G31 = (AS1-AS2)/(S0*S1)
G32 = (AS1+1)/(S0-S1)
B31 = (G31*S1+1)/(S0-S1)
B32 = (G31*S0+1)/(S1-S0)
G33 = (G32*S0+1)/(S1-S0)
G34 = (-SM1**2*(B11**2/(2*S0)+2*B11*K11/(S0+SI))+K11**2/(2*S1))
Y10 = -SM1**2*(B21**2/(2*S0)+2*B21*K21/(S0+SI))+K21**2/(2*S1)
Y11 = -SM1**2*(B22**2/(2*S0)+2*B22*K22/(S0+SI))+K22**2/(2*S1)
Y12 = B11*B21/(2*S0)+(K11*B22)/(S0+SI)+K11*K22/(2*S1)
D10 = -SM1**2*(DI0-SM2**2*D20)
D11 = B31**2*(EXP(2*S0)+K31**2*(EXP(2*S1*T)-1))/S1
D12 = 2*G31*B31*(EXP(S0*T)-1)/S0+2*G31*K31*(EXP(S1*T)-1)/S1
D13 = G31**2*(B31**2*(EXP(S0*T)-1)/(2*S0)+K32**2*(EXP(2*S1*T)-1)/S1)
ZT1 = DS1+DS2+DS3
ZT2 = B32**2*(EXP(2*S0*T)-1)/(2*S0)+K32**2*(EXP(2*S1*T)-1)/S1
ZT3 = 2*G32*B32*(EXP(S0+SI)*T)-1)/(S0+SI)
ZT4 = DT1+DT2+DT3
DZ1 = (G31+B31*EXP(S1*T))**2*Y10
DZ2 = (G32+B32*EXP(S0*T))**2*Y20
DZ3 = 2*SQR(DZ1/Y10)*SQR(DZ2/Y20)*Y12
C COMPUTE THE VARIANCE
HTSD = (DZ1+DZ2+DZ3+ZT1*SM1**2+ZT2*SM2**2)*M
C
RETURN
END
APP04750
APP04760
APP04770
APP04780
APP04790
APP04800
APP04810
APP04820
APP04830
APP04840
APP04850
APP04860
APP04870
APP04880
APP04890
APP04900
APP04910
APP04920
APP04930
APP04940
APP04950
APP04960
APP04970
APP04980
APP04990
APP05000
APP05010
APP05020
APP05030
APP05040
APP05050
APP05060
APP05070
APP05080
APP05090
APP05100
APP05110
APP05120
APP05130

```


LIST OF REFERENCES

1. Coffman, E.G., Muntz, R.R. and Trotter, H., "Waiting time distribution for processor-sharing system," Journal of the Association for Computing Machinery, 17, pp. 120-130, 1970.
2. Mitra, D., Waiting time distributions from closed queueing network models of shared processor systems, Bell Laboratories Report, 1981.
3. Gaver, D., Jacobs, P. and Latouche, G., The Normal Approximation and Queue Control for Response Time in a Processor-Shared Computer System Model, Naval Postgraduate School Technical Report, 1984.
4. Lavenberg, S.S. and Reiser, M., "Stationary state probabilities at arrival instants for closed queueing network with multiple types of customers," Journal of Applied Probability, 17, pp. 1048-1061, 1980.
5. Cohen, J. W., "The multiple phase service network with generalized Processor Sharing," Acta Informatica, 12, pp. 245-284, 1979.
6. Cinlar, E., Introduction to Stochastic Processes, pp. 269-271, Prentice-Hall, Englewood Cliffs, N.J., 1975.
7. Kelly, P. P., Reversibility and Stochastic Networks, p. 12, John Wiley and Sons, New York, 1979.
8. Keilson, J., Markov Chain Models-Rarity and Exponentiality, Springer-Verlag, New York, 1979.
9. Cox, D. R., Renewal Theory, Methuen Monograph, 1962.
10. Iglehart, D. L., "Limiting diffusion approximations for the many server queue and the repairman problem," Journal of Applied Probability, 2, pp. 429-441, 1965.
11. Karlin, S. and Taylor, H. M., A first course in Stochastic Process, (second edition), Academic Press, New York, 1975.
12. Gaver, D., Jacobs, P., Processor-shared time-sharing model in heavy traffic, in preparation.
13. Arnold, L., Stochastic Differential Equations : Theory and Applications, John Wiley and Sons, 1974.

14. Berman, D., An analytical approach to Diffusion Approximation in Queueing, Unpublished Doctoral Dissertation, New York University, 1979.
15. Cochran, W. G., Sampling Techniques, (second edition), John Wiley and Sons, 1963.
16. Gaver, D. P., and Lehoczky, J. P., "Gaussian approximation to service problem: a communications system example," Journal of Applied Probability, 13, pp. 768-780, 1976.
17. Gaver, D., Jaccbs, P. and Latouche, G., "Finite birth and death models in randomly changing environments," to appear Journal of Applied Probability, 1984.
18. Kleinrock, I., Queueing System, Vol. II, Wiley-Interscience, 1976.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
3.	Department Chairman, Code 55 Department of Operation Analysis Naval Postgraduate School Monterey, California 93943	1
4.	Professor P. A. Jacobs, code 55Jc Department of Operation Analysis Naval Postgraduate School Monterey, California 93943	1
5.	Professor D. P. Gaver, code 55Gv Department of Operation Analysis Naval Postgraduate School Monterey, California 93943	1
6.	Department of Educations The Headquarters of Royal Thai Navy Tanch Arun Amarin, Bangkok Yai Bangkok 10600, Thailand	2
7.	Ens. Suriya Pornsuriya 155 Moc 4, Tambon Bangyapraek Amphoe Muang, Samut Sakhon 74000 Thailand	2

207404

Thesis
P7416
c.1

Pornsuriya

Normal approximation
for response time in a
processor-shared compu-
ter system model.

MAY 13 85

50281

207404

Thesis
P7416
c.1

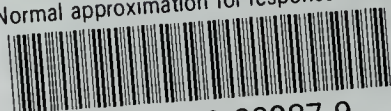
Pornsuriya

Normal approximation
for response time in a
processor-shared compu-
ter system model.



thesP7416

Normal approximation for response time i



3 2768 000 99287 9

DUDLEY KNOX LIBRARY