DIGITAL SYSTEMS LABORATORY

STANFORD ELECTRONICS LABORATORIES
DEPARTMENT OF ELECTRICAL ENGINEERING
STANFORD UNIVERSITY · STANFORD, CA 94305

SEL-77-017

Passage Time Distributions for a Class of

Queueing Networks: Closed, Open, or Mixed,

with Different Classes of Customers

with Applications to Computer System Modeling

by

Philip S. Yu

March 1977

Technical Report No. 135

PASSAGE TIME DISTRIBUTIONS FOR A CLASS OF QUEUEING NETWORKS: CLOSED, OPEN, OR MIXED, WITH DIFFERENT CLASSES OF CUSTOMERS WITH APPLICATIONS TO COMPUTER SYSTEM MODELING

by
Philip S. Yu

March 1977

Technical Report No. 135

Digital Systems Laboratory
Departments of Electrical Engineering and Computer Science
Stanford University
Stanford, CA 94305

PASSAGE TIME DISTRIBUTIONS FOR A CLASS OF QUEUEING NETWORKS:
CLOSED, OPEN, OR MIXED, WITH DIFFERENT CLASSES OF CUSTOMERS
WITH APPLICATIONS TO COMPUTER SYSTEM MODELING

by

Philip S. Yu

ABSTRACT

Networks of queues are important models of multiprogrammed time-shared computer systems and computer communication networks. Although equilibrium state probabilities of a broad class of network models have been derived in the past, analytic or approximate solutions for response time distributions or more general passage time distributions are still open problems. In this paper we formulate the passage time problem as a "hitting time" or "first passage time" problem in a Markov system and derive the analytic solution to passage time distributions of closed queueing networks. Efficient numerical approximation is also proposed. The result for closed queueing networks is further extended to obtain approximate passage time distributions for open queueing networks. Finally, we employ the techniques derived in this paper to study the interfault time and response time distribution and density functions of multiprogrammed computer systems. The effects of program behavior, degree of multiprogramming, size of main memory, service time of paging devices and rate of file I/O requests on the shape of distribution functions and density functions have been examined.

CONTENTS

1. INTRODUCTION

Queueing network models have important applications in computer system and computer communication network modeling. Work on this application in the last several years has produced a variety of models to capture important aspects of computer systems and computer communication networks. A broad class of queueing network models has been put together under a uniform framework by Baskett, Chandy, Muntz and Palacios [2] and further extended by Kobayashi and Reiser [18] [26]. Obtainable from such analyses are stationary queue length distributions and measures of system performance derivable from them, such as mean response time of each user and throughput of the system. Although the average response time or waiting time can be obtained easily from the mean queue length based on Little's formula, "$L = \lambda W$" , the analytic or approximate solution for response time or waiting time distribution is still an open problem. Clearly, the distribution function is much more informative than the mean because it can provide us with higher moments, quantiles, percentiles, etc., i.e. fluctuations around the mean.

Using simulations via regenerative techniques to obtain interval estimations of passage time distributions has been studied by Iglehart and Shedler [14] for closed queueing networks, where passage time is defined to be the time for a job to traverse a portion of the network [14], hence, a more general term than response time or waiting time.

In section 2, we briefly survey the history on the development of the theory of queueing network models. In section 3, we summarize the queueing network models where analytic results are available and the results on steady state probabilities of the queueing networks. In section 4, we derive the

passage time distributions of the closed queueing networks specified in section 3 using the concept of "hitting time" or "first passage time" of Markov processes. In section 5, we use a multiprogrammed computer system model which is a two stage cyclic queueing network as an example to go through all the details involved in calculating a passage time distribution. In section 6, an efficient numerical technique to obtain approximate passage time distributions of the closed queueing networks is proposed. The approximate solutions are very satisfactory. In section 7, we examine the open queueing networks specified in section 3 and extend the results on the closed queueing networks to obtain approximate passage time distributions for the open networks. The major difficulty encountered in handling the open queueing networks is the infiniteness of its state space as we shall see later. In section 8, we employ the derived techniques to study interfault time distributions and response time distributions of multiprogrammed computer systems. Finally, in section 9 we draw the conclusion.

2. SURVEY

A good survey on the development of the theory of queueing network models up to 1972 is contained in Muntz and Baskett [22]. Here, we excerpt the contents from [22] and fill in the development thereafter. For some ten years, the most general class of queueing networks for which an analytic solution was known is treated by Jackson [15]. Jackson develops the equilibrium distributions of the states of the following class of networks:

(1) All the service time distributions are exponential

(2) All the customers are identical

(3) A customer leaving a service center can choose the next service center according to a fixed set of branching probabilities for the service center being left

(4) External arrivals and departures are allowed. The external arrival process is assumed to be Poisson.

Gordon and Newell [13] treat the similar kind of queueing networks and make clear the product form of the solution of the balance equations describing the steady state of the model. Properties (1) and (2) of this class of networks are clearly limitations of its applications. In the past few years, applications of such queueing networks to the modeling of computer systems have been drawing increasing attention, and the extensions of the above results to more general queueing networks have been attempted by various authors. We summarize their results in the following:

Ferdinand [12] analyzed a particular queueing model called the finite source model or the machine repairman model. The system was a cyclic model with two service centers and consists of a fixed number of customers. One

3

service center which can be viewed as the CPU is shared by all waiting customers simultaneously at a service rate inversely proportional to the number of customer-being served. The other service center which can be viewed as the terminals consists of a sufficient number of servers so that no queueing occurs. These two types of service centers will be referred to as processor sharing and infinite server stations, respectively. Different classes of customers are allowed, i.e. each customer has its own pair of exponentially distributed service times, one for each service center. Posner and Bernholz [14] consider the more general network model of Gordon and Newell where each customer has its own set of branching probabilities, and exponentially distributed service times. The network is closed and two kinds of service models are allowed: FCFS (first-come first-served) and processor sharing. Only under the processor sharing discipline, different customers may have different service time distributions.

Sakata, Noguchi, and Oizumi [27] discovered that when processor sharing scheduling was applied to the classical infinite source queueing model (M/G/1), the equilibrium distribution of queue sizes for the model was the same as that for a similar model with exponentially distributed service times with the same mean as the original general distribution. Baskett [1] derived a similar result for a finite source model in which the service time distribution at both service centers have rational Laplace transforms and Baskett and Palacios [4] extended that result to the central server network model which Buzen [5] has studied. The equilibrium solutions have the product form The model includes FCFS, processor sharing and infinite server types of service centers and the service time distributions can be any distributions with rational Laplace trans-forms for the two latter types of service centers. But the queueing system is closed and only a single class of customers is allowed.

4

Whittle [28] [29] showed that the balance equations describing the under-lying birth and death processes could be replaced by sets of "independent" balance equations. Chandy [8] extended this technique to more complex models and extended the range of networks for which product form solutions can be found. Chandy developed the solution for networks in which the service center is of FCFS, processor sharing or LCFS type and in which all customers are the same. Palacios [23] independently developed solutions for a particular network with "types" of customers. In [9], Chandy, Keller and Browne further extended the concept of customer "type" and added the concept of customer "mode" for general networks.

The recent most noteworthy progress in extending the class of analytically solvable queueing networks has been done by Baskett, Chandy, Muntz and Palacios [2]. These authors have succeeded in casting into a unified theory of previously known but unconnected results such as queue size distributions for M/M/1 with FCFS discipline, general service time distribution for processor sharing, infinite server discipline and pre-emptive --resume LCFS discipline and queueing systems with various classes of customers. Reiser and Kobayashi [26] generalize the result of [2] to the case in which customer transitions are characterized by more than one closed Markov chain. The technique of generating function has been applied to obtain closed form solutions. Kobayashi and Reiser [18] further extend the job routing behavior to high order Markov chain, i.e. the transition probability of a job from one station to the other can depend on, at least, the last two stations it has visited and not just the last one. In Lam [20], the class of queueing networks with a product form solution is extended to include state dependent lost arrivals and trigger arrivals. Such queueing network models can be used to model store and forward packet switching nodes and multiprogramming computer systems with storage constraint.

3. EQUILIBRIUM STATE PROBABILITIES OF QUEUEING NETWORKS

In this section, we examine the class of queueing network models whose equilibrium state probabilities have been obtained by Baskett, Chandy, Muntz and Palacios [2] under a uniform framework. Their results will serve as the basic foundation for our subsequent analysis of passage time distributions.

The queueing network can have any kind of topologies and any number of service centers. The customers may have different classes. We will assume that the queueing network consists of M service centers and R classes of customers. At any time, each job can only be in one job class, but it may change class as it traverses through the network. Upon completing service at center i, a job of class j is routed to center k and changes to class ℓ with probability $P_{ij,k\ell}^R$. Furthermore, the routing matrix $P^R = \{P_{ij,k\ell}^R ; 1 \leq k \leq M, 1 \leq j, \ell \leq R\}$ can be considered as defining a Markov chain whose state space is $\{(i,j), 1 \leq i \leq M, 1 \leq j \leq R\}$ and transition matrix is P^R.

The most general service time distribution considered is the one which has a rational Laplace transform. All exponential, hyperexponential and hypo-exponential distributions have rational Laplace transforms. Cox [10] has shown that any distribution with a rational Laplace transform can be represented by a network of exponential stages of the form shown in Fig 3.1.
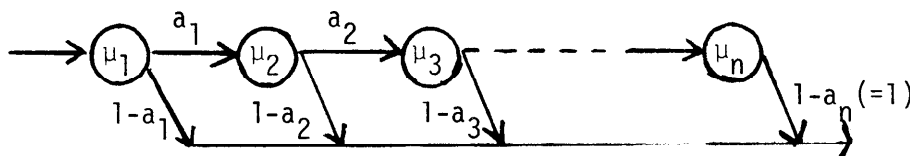

Fig 3.1 Representation of service time distributions by the method of stages

In Fig 3.1, $1-a_i$ is the probability that the customer leaves the service center after the i-th state and a_i is the probability that the customer goes to the next stage. The service time at stage i has an exponential distribution with mean $\frac{1}{\mu_i}$.

The service centers of the network can be any of the following four types.

Type 1: The service discipline is first-come first-served (FCFS). All customers must have the same service time distribution regardless of its job class.

Type 2: There is a single server at the service center, the service discipline is processor sharing. The processing rate is reduced to 1/n if n is the number of customers sharing the service. Each class of customers may have a distinct service time distribution. The service time can have any distribution with a rational Laplace transform.

Type 3: The number of servers in the service center is greater than or equal to the maximum number of customers that can be queued at this center. This is call the infinite server model. Each class of customers may have a distinct service time distribution. The service time can have any distribution with a rational Laplace transform.

Type 4: The station contains a single server, the queueing discipline is pre-emptive-resume last-come first-served (LCFS). Each class of customers may have a distinct service time distribution. The service time can have any distribution with a rational Laplace transform.

Notice a type 2 service center is, in many cases, a reasonable representation of the central processing unit allocating small quanta to jobs requiring service in round robin. Actually, it is the imiting case of round robin scheduling

where the size of the quanta goes to zero and no overhead is associated with job switching. A type 3 center is often used to represent the terminals in a time sharing system. Type 1 service centers are most commonly encountered, e.g. in modeling secondary storage I/O devices, channels in point to point computer communication networks, etc. A type 4 service center may also be used to model CPU since it is an efficient preemptive scheduling algorithm.

We now proceed to examine the state of the network model. The state can be represented by a vector

$$\tilde{s} = (\tilde{x}_1, \ldots, \tilde{x}_M)$$

where $\tilde{x}_i$ represents the state at service center i. The exact form of $\tilde{x}_i$ depends on the type of service center i. We summarize the forms of $\tilde{x}_i$'s below:

Type 1 service center:

$$\tilde{x}_i = (x_{i1}, \ldots x_{in_i})$$

x_{ij} is the class of the j-th customer at station i in FCFS order and n_i is the number of customers at station i.

Type 2 or 3 service center:

$$\tilde{x}_i = (\tilde{v}_{i1}, \tilde{v}_{i2}, \ldots, \tilde{v}_{iR})$$

$\tilde{v}_{ir}$ is a vector $(m_{ir1}, m_{ir2}, \ldots, m_{irU_{ir}})$ and the ℓ-th component of v_{ir} is the number of customers of class r in center i and in the ℓ-th stage of service. U_{ir} is the number of stages of the service time for a class r customer at service center i.

Type 4 service center:

$$\tilde{x}_i = ((r_{i1}, m_{i1}), (r_{i2}, m_{i2}) \cdots (r_{in_i}, m_{in_i}))$$

r_{ij} is the class of the j-th customer in LCFS order and m_{ij} is its stage of

service.

Let $\tilde{S}(t)$ be the stochastic process describing the state of the network model and E be the collection of all permissible state $\tilde{s}$'s described above.

Propositon 3.1

$\tilde{S}(t) = (\overset{\sim}{S_1}(t),\ldots, \overset{\sim}{S_M}(t))$ forms a continuous time Markov chain with state space $E = \{\tilde{s}\}$, where $\overset{\sim}{S_i}(t)$ describes the state of the i-th service center at time t.

Before presenting the solution to the class of networks described above, we need to define a set of terms that appear in the solution. Recall $P^R = \{P^R_{ir,js}\}$ is the routing matrix and defines a Markov chain. This Markov chain is assumed to be decomposable into m ergodic subchains. Let $E_1, E_2, \ldots, E_m$ be the sets of states in each of these subchains. For each ergodic subchain E_k, we define the following set of equations

$$\underset{(i,r)\in E_k}{\Sigma} e_{ir} P^R_{ir,js} + q_{js} = e_{js} \quad (j,s)\in E_k$$

The value of q_{js} is determined by the rate of exogenous arrivals of class s customer to service center j. Notice if $q_{js} = 0 \quad \forall (j,s)\in E_k$, is closed with respect to E_k. In this case, e_{ir} can only be determined to within a multiplicative constant and interpreted as the relative arrival rate of class r customer to service center i. If not all $q_{js} = 0$ for $(j,s)\in E_k$, then we assume a unique solution for e_{ir}. In this case, e_{ir} is the absolute arrival rate of class r customers to service center i.

One further definition is required. If at the i-th service center the customers of class r have a service time distribution that is represented as a network of stages, then it will be represented as shown in Fig 3.2.

9
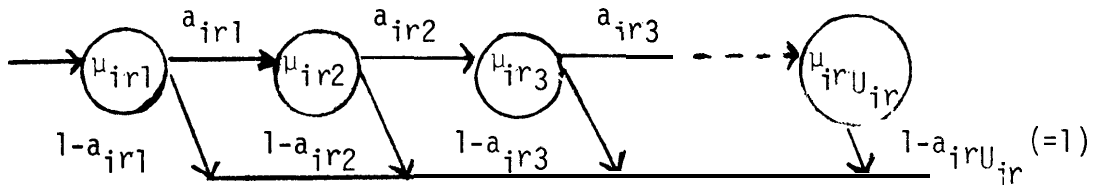
Note:

(1) the first subscript denotes the service center

(2) the second subscript denotes the class of the customer

(3) the third subscript denotes the stage of service

Fig 3.2 Representation of the service time distribution of a class r customer at service center i

Let $A_{ir\ell} = \prod_{j=1}^{\ell} a_{irj}$, i.e. $A_{ir\ell}$ is the probability that a class r customer at station i will reach its j-th stage of service. Finally, we state the equilibrium state probabilities in the following theorem which can be proved by checking that the independent balance equations are satisfied.

Theorem 3.1:

For a network of service stations which is open, closed or mixed in which each service center is of type 1, 2, 3 or 4, the equilibrium state probabilities are given by

$$P(\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_M) = c\, d(\tilde{s})\, f_1(\tilde{x}_1)\, f_2(\tilde{x}_2) \cdots f_M(\tilde{x}_M)$$

where c is a normalizing constant chosen to make the equilibrium probabilities sume to 1,

$\tilde{s}$ is an abbr. of $(\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_M)$,

$d(\tilde{s})$ is a function of the number of customers in the system,

and each f_i is a function that depends on the type of the service center i.

To be more specific, for

Type 1 service center:

$$f_i(\tilde{x}_i) = (\frac{1}{\mu_i})^{n_i} \prod_{j=1}^{n_i} e_{ix_{ij}}$$

Type 2 service center:

$$f_i(\tilde{x}_i) = n_i! \prod_{r=1}^{R} \prod_{\ell=1}^{u_{ir}} \{ [\frac{e_{ir} A_{ir\ell}}{\mu_{ir\ell}}]^{m_{ir\ell}} \frac{1}{m_{ir\ell}!} \}$$

Type 3 service center:

$$f_i(\tilde{x}_i) = \prod_{r=1}^{R} \prod_{\ell=1}^{u_{ir}} \{ [\frac{e_{ir} A_{ir\ell}}{\mu_{ir\ell}}]^{m_{ir\ell}} \frac{1}{m_{ir\ell}!} \}$$

Type 4 service center:

$$f_i(\tilde{x}_i) = \prod_{j=1}^{n_i} [\frac{e_{ir_{ij}} A_{ir_{ij}}^{m_{ij}}}{\mu_{ir_{ij}}^{m_{ij}}}]$$

If the arrivals to the system depend on the total number of customers in the system, denoted by $M(\tilde{s})$, and the arrivals of class r customers to center i follow fixed probability P_{ir}^A, then

$$d(\tilde{s}) = \prod_{i=0}^{M(\tilde{s})-1} \lambda(i)$$

where $\lambda(i)$ is the arrival rate when the total number of customers is equal to i.

Another case of interest will be the case where the arrival process consists of m Poisson arrival streams corresponding to the m ergodic subchains mentioned before. Let $\lambda_j(i)$ be the instantaneous mean arrival rate for the j-th stream when the total number of customers in the j-th subchain is equal to i and $M(\tilde{s}/E_j)$ denote the number of customers in the j-th subchain when the state of the system

11

is $\tilde{s}$, then

$$d(\tilde{s}) = \prod_{j=1}^{m} \prod_{i=0}^{M(\tilde{s}/E_j)-1} \lambda_j(i)$$

If the network is closed, then

$$d(\tilde{s}) = 1$$

Now let us consider the marginal distribution of queue lengths. Define an aggregate system state as the number of customers of each class in each center. More formally, an aggregate state $\tilde{W}$ of the system is given by $(\tilde{y}_1, \tilde{y}_2, \ldots \tilde{y}_M)$ where $\tilde{y}_i = (n_{i1}, n_{i2}, \cdots, n_{iR})$ with n_{ir} denoting the number of customers of class r in service center i. Let $\frac{1}{\mu_{ir}}$ be the mean service time of a class r customer at service center i.

Theorem 3.2

The equilibrium distribution of the aggregate state $\tilde{W} = (\tilde{y}_1, \cdots, \tilde{y}_M)$ is given by

$$P(\tilde{W} = (\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_M)) = c\, d(\tilde{W})\, g_1(\tilde{y}_1)\, g_2(\tilde{y}_2)\, \cdots\, g_M(\tilde{y}_M)$$

where

for Type 1 service center: $g_i(\tilde{y}_i) = n_i! \{\prod_{r=1}^{R} \frac{1}{n_{ir}!} (e_{ir})^{n_{ir}}\} (\frac{1}{\mu_i})^{n_i}$

Type 2 or 4 service center: $g_i(\tilde{y}_i) = n_i! \prod_{r=1}^{R} \frac{1}{n_{ir}!} (\frac{e_{ir}}{\mu_{ir}})^{n_{ir}}$

Type 3 service center: $g_i(\tilde{y}_i) = \prod_{r=1}^{R} \frac{1}{n_{ir}!} (\frac{e_{ir}}{\mu_{ir}})^{n_{ir}}$

The most obvious implication of this theorem is that the equilibrium distribution of the aggregate state $\tilde{W}$ depends only on the means of the service time distributions.

A further simplification is possible if the network is open and the arrival process does not depend on the state of the model. Let $\tilde{h} = (n_1, n_2, \ldots, n_M)$

12

be the aggregate state which represents the total number of customers in each service station. Let R_i = {r: class r customers may require service at service center i}. Furthermore, let

$$\rho_i = \Sigma_{r \in R_i} \lambda \left(\frac{e_{ir}}{\mu_i} \right) \quad \text{if service center i is type 1}$$

$$\rho_i = \Sigma_{r \in R_i} \lambda \left(\frac{e_{ir}}{\mu_{ir}} \right) \quad \text{if service center i is type 2,3 or 4.}$$

Theorem 3.3

The equilibrium distribution of the aggregate state $\tilde{n} = (n_1, \ldots, n_M)$ of an open network with state independent arrival rate is given by

$$P(\tilde{n} = (n_1, \ldots, n_M)) = \prod_{i=1}^{M} P_i(n_i)$$

where

$$P_i(n_i) = (1 - \rho_i) \, \rho_i^{n_i} \quad \text{if service center is type 1, 2, or 4}$$

$$P_i(n_i) = e^{-\rho_i} \left(\frac{\rho_i^{n_i}}{n_i!} \right) \quad \text{if service center is type 3}$$

Furthermore, various forms of state dependent service rates can easily be incorporated into the network models. We examine the following three cases:

Case 1: The service rate at a service center depends on the total number of customers at that service center. Let $v_i(n_i)$ be the rate of service at the i-th service center when there are n_i customers at that service center relative to the service rate when $n_i = 1$. Then $f_i(x_i)$ in theorem 3.1 becomes $f_i(x_i) \left(1 / \prod_{a=1}^{n_i} v_i(a) \right)$. This form of state dependent service rate is very useful. Consider the case when the i-th service center contains k_i multiple servers, we can let

13

$$v_i(n_i) = \begin{cases} n_i &, \; 1 \le n_i \le k_i \\ k_i &, \; n_i > k_i \end{cases}$$

Cases where $v_i(n_i)$ is a general function of n_i can be found in Kobayashi [19].

Case 2: The service rate of a class r customer at service center i depends on the number n_{ir} of class r customers at service center i. This form of state dependent service rate can not be modeled for type 1 service centers. Let $y_{ir}(n_{ir})$ be the service rate of class r customers at service center i relative to the service rate when there is only one class r customer at service center i. In this case, $f_i(x_i)$ is replaced by $f_i(x_i) \prod_{r=1}^{R} \prod_{a=1}^{n_{ir}} (1/y_{ir}(a))$

Case 3: The state dependent service rate involves the number of customers in several service centers. Let $I = \{i_1, i_2, \ldots, i_m\}$ be a subset of the service centers. Let $n_I = \sum_{i \; \in \; d} n_i$ and let $Z_I(n_I)$ be the relative service rate to customers in the subset I of service centers relative to the service rates when n_I is one. In this case $\prod_{i \; \in \; d} f_i(x_i)$ becomes $\prod_{i \in I} f_i(x_i) \prod_{a=1}^{n_I} (1/Z_I(a))$

Finally, we note that these various forms of state dependent service rates can be combined.

4. ANALYTIC SOLUTION ON PASSAGE TIME DISTRIBUTIONS FOR THE CLOSED QUEUEING NETWORKS

In this section, we obtain the distribution functions of passage times for the closed queueing networks specified in the previous section. That is to say the closed queueing networks can have any of the four types of service centers and different classes of customers whose transitions follow some routing chain. The passage time considered is the time required for a job to reach a specified destination from a given start where certain restrictions may be put on the passage. A formal definition will be given later. Response time is taken to be the time measured between the arrival instant of a customer and its departure instant for open queueing networks and the time required for a job to go through a complete circuit or a loop for closed queueing networks. As we shall see that response time is actually a special kind of passage time. The reason why we restrict ourselves to closed networks in this section is simply because the state space of a closed queueing network is finite. The advantage of the finite state space will be aparent later on.

We will first introduce the concept of a tag job. Concentrating on the behavior of the tag job as it traverses through the network provides us a means to evaluate the passage time distribution. Each job is assumed to have equal probability of being tagged. In order to keep track on the position of the tag job, we need to augment the state variable described in section 3 by an extra component which describes the position of the tag job in the network. Examining the state variable, we see that in the FCFS and LCFS service station, there is a one to one correspondence between the state components and the jobs in the system. This component specifies both the position (implicitly) and the class of the corresponding job. It also specifies the stage of service if the center is LCFS. By letting $K(t)$ denote the index of the state component

15

describing the tag job at time t, we can identify the tag job position and status unambiguously. But, for the processor sharing and infinite server service centers, a one to one correspondence between state components and jobs does not exist. All jobs in the same job class and stage of service will be described by a single state component, i.e. they are indistinguishable under the state description. Since we are only interested in the stochastic behavior of the tag job, this ambiguity really doesn't matter. This ambiguity only elaborates a little bit the specification of the transition matrix as we shall see. We will still let $K(t)$ indicate the index of the corresponding state component in the state variable. Let $\tilde{S}(t)$ be the stochastic process defined in proposition 3.1 and $(\tilde{S}(t), K(t))$ be the new stochastic process just described, and let E and E' be the corresponding state spaces, respectively. Recall $\{\tilde{S}(t)\}$ forms a continuous time Markov chain. Let $\{T_n\}$ be the jump times of the Markov chain, with $0 = T_0 < T_1 < \ldots$. Clearly $\tilde{S}(T_k)$ is related to $\tilde{S}(T_{k-1})$ through the transition matrix, $P^0 = \{P^0_{ij}\}$, of $\tilde{S}(t)$. If the tag job is a FCFS or LCFS station, at the service completion instant, i.e. the jump time of the underlying Markov chain, $K(T_k)$ will be uniquely determined by the $\tilde{S}(T_k)$ and $(\tilde{S}(T_{k-1}), K(T_{k-1}))$. That is to say there is a one to one correspondence between the possible next states of $\tilde{S}(T_{k-1})$ and $(\tilde{S}(T_{k-1}), K(T_{k-1}))$ under the same transition probabilities. If the tag job is in a processor sharing or infinite server station, this is no more true. For example, **assume** q jobs of the same class are in the same stage of a processor sharing or infinite server station and one of them is the tag job. At the next service completion instant of this stage of the service center, one of the q jobs changes its status, either moving to a different station or a different stage of the current service center. $K(T_k)$ will depend upon not only the value of $\tilde{S}(T_k)$ and $(\tilde{S}(T_{k-1}), K(T_{k-1}))$ but also the fact whether the tag job is the job just

16

completing its service. Nevertheless, the service completion of the tag job can be determined stochastically. Since the probability that the tag job completes its service at this instant is known which is equal to $1/q$, we can incorporate the uncertainty on $K(T_k)$ into the transition probabilities of the stochastic process $(\tilde{S}(t), K(t))$, and these transition probabilities only depend upon the current state $(\tilde{S}(T_{k-1}), K(T_{k\ 1}))$. Therefore, we conclude:

Proposition 4.1

$$\tilde{Z}(t) = (\tilde{S}(t), K(t)) \text{ forms a continuous time Markov chain}$$

In order to define the passage time considered formally, we need to introduce four subsets of E', (A_1, A_2, B_1, B_2) which will tell us in effect the start time and stop time of a particular passage time of the tag job as we observe the sample path of the stochastic process. A_2 and B_2 are the sets of state where passage times may start or stop, respectively. Knowing that the state of the stochastic process is in A_2 or B_2 is not sufficient to conclude that a passage time starts or stops in general under path restrictions. Path restrictions will be allowed if appropriate A_1 and B_1 can be defined such that the start or stop of a passage time can be determined from the additional fact that the process has also passed through some state in **A,** **or** B_1, respectively. In the case where A_2 and B_2 are disjoint and the passage time terminates at . the first time the stochastic process hits some state in B_2, the set B_1 will be redundant. Similar remark holds for A_1. We next define two sequences of random times, $\{S_j : j \geq 0\}$ and $\{\Gamma_j : j \geq 1\}$, where S_{j-1} is the start time of the j-th passage time and Γ_j is the termination time of the j-th passage time. Assume that the initial state of the Markov chain $\{\tilde{Z}(t) ; t \geq 0\}$ is such that a passage time for the tag job begins at t=0. Formally,

$$S_0 = 0$$

$$S_j = \inf \quad \{ T_n : \tilde{Z}(T_n) \in A_2, \ \tilde{Z}(T_k) \in A_1 \quad \text{for}$$
$$\text{some } T_k > S_{j-1} \text{ and } k < n\} , \ j \geq 1$$

17

$$t_j = \inf \{ T_n : \tilde{Z}(T_n) \in B_2, \; \tilde{Z}(T_k) \in B_1$$
$$\text{for some } T_k > S_{j-1} \text{ and } k < n\}, \; j \geq 1$$

Then the j-th passage time is simply $PS_j = \Gamma_j - S_{j-1}$, $j \geq 1$. In the case of response times, we will have $A_1 = B_1$, $A_2 = B_2$ and consequently $S_j = \Gamma_j$ for all $j \geq 1$. This formal definition of passage time is similar to but somewhat more restrictive than that in Iglehart and Shedler [14].

Now we'll try to convert the above passage time problem into the hitting time or first passage time problem of another stochastic process, $\tilde{Z}{}^{*}(t)$. The $\tilde{Z}{}^{*}(t)$ should have the following properties

(1) The set of possible initial states, A_2^{*}, should be isomorphic to the set A_2 which may start a passage time of $\tilde{Z}(t)$. Furthermore, the initial distribution at each state in A_2^{*} should equal to the stationary probability that $\tilde{Z}(t)$ may start the passage time from its corresponding state in A_2.

(2) The process $\tilde{Z}{}^{*}(t)$ has an absorbing state. When the passage of $\tilde{Z}(t)$ terminates, $\tilde{Z}{}^{*}(t)$ should hit the absorbing state simultaneously.

First of all, let us add an extra component, $I(t)$, to the state variable $(\tilde{S}(t), K(t))$. That is to say the new state variable will be the vector
$$\tilde{Y}(t) = (\tilde{S}(t), K(t), I(t)), \; t \geq 0$$
where $I(t)$ is used to indicate whether during the previous state transitions, any state in B_1^{*} has been passed through. Here we let B_1^{*} be the direct extensions of B_1 with the extra component setting to zero, i.e. $B_1^{*} = \{(\tilde{Z},0) : \tilde{Z} \in B_1\}$. We will further assume the process $\tilde{Y}(t)$ starts with some state $(\tilde{Z},0)$ at t=0 where $\tilde{Z} \in A_2$. Since $I(T_k)$ can be completely determined by $\tilde{Y}(T_{k-1})$, we conclude that

Proposition 4.2

$\tilde{Y}(t)$ is a continuous time Markov chain with the following properties:

18

(1) $\tilde{Y}(0) \in A_2^*$

where $A_2^* = \{(\tilde{z},0) : \tilde{z} \in A_2\}$

(2) $P_{(\tilde{z}_1,1)(\tilde{z}_2,i)} = \begin{cases} P_{\tilde{z}_1,\tilde{z}_2} & \text{for } i=1 \\ 0 & \text{for } i=0 \end{cases}$

$P_{(\tilde{z}_1,0)(\tilde{z}_2,0)} = \begin{cases} P_{\tilde{z}_1,\tilde{z}_2} & \text{for } \tilde{z}_1 \notin B_1 \\ 0 & \text{for } \tilde{z}_1 \in B_1 \end{cases}$

$P_{(\tilde{z}_1,0)(\tilde{z}_2,1)} = \begin{cases} P_{\tilde{z}_1,\tilde{z}_2} & \text{for } \tilde{z}_1 \in B_1 \\ 0 & \text{for } \tilde{z}_1 \notin B_1 \end{cases}$

where

$\{P_{(\tilde{z}_1,i_1)(\tilde{z}_2,i_2)}\}$ is the transition matrix of $\tilde{Y}(t)$

and

$\{P_{\tilde{z}_1,\tilde{z}_2}\}$ is the transition of matrix of $\tilde{Z}(t)$

From properties (1) and (2), it is apparent

$Y(t) = \begin{cases} (\tilde{Z}(t),0), & \text{if } \tilde{Z}(T_k) \notin B_1^* & \text{for all } 0 < T_k < T_{n-1} < t \leq T_n \\ (\tilde{Z}(t),1), & \text{if } \tilde{Z}(T_k) \in B_1^* & \text{for some } 0 < T_k < T_{n-1} < t \leq T_n \end{cases}$

If B_2 and B_1 are disjoint, the number of states reachable from A_2^* may be reduced if we modify the definition of the transition matrix on the state $(\tilde{z}_1,0)$ to be

$P_{(\tilde{z}_1,0)(\tilde{z}_2,1)} = \begin{cases} P_{\tilde{z}_1,\tilde{z}_2} & \text{for } \tilde{z}_1 \in B_1, \ \tilde{z}_2 \notin B_1 \\ 0 & \text{otherwise} \end{cases}$

and

$P_{(\tilde{z}_1,0)(\tilde{z}_2,0)} = \begin{cases} P_{\tilde{z}_1,\tilde{z}_2} & \text{for } \tilde{z}_1 \notin B_1 \text{ or } \tilde{z}_1 \text{ and } \tilde{z}_2 \in B_1 \\ 0 & \text{otherwise} \end{cases}$

for every $\tilde{z}_1 \in E$.

19

The difference between the two definitions is that the component $I(t)$ will not be set to one until the process $\tilde{Z}(t)$ leaves the set B_1^* for the first time-under the modified definitions.

Let us define B_2^* to be the direct augmentation of B_2 by setting the extra component to 1, i.e. $B_2^* = \{(\tilde{z},1): \tilde{z} \in B_2\}$. We further modify the state space of the stochastic process $Y(t)$ by lumping all the states in B_2^* into a single absorbing state $\tilde{r}$. Clearly the new stochastic process satisfies property (2) of $\tilde{Z}^*(t)$ and for each state $\tilde{z}_1$ in A_2 of $\tilde{Z}(t)$, there is a state $(\tilde{z}_1, 0)$ in A_2^* corresponding to it. That is to say the passage time of $Z(t)$ will have the same distribution as that of the hitting time to state $\tilde{r}$ of this new stochastic process under an appropriate initial distribution. We will call the new stochastic process $\tilde{Z}^*(t)$ and the corresponding state space E^*.

As pointed out earlier, if A_2 and B_2 are disjoint and the passage time terminates at the first time that the stochastic process hits some state in B_2, the set B_1 is redundant. In this case, the extra component $I(t)$ is also redundant. Lumping the states in B_2 into an absrobing state, we get the desired stochastic process $\tilde{Z}^*(t)$. Even if A_2 and B_2 are not disjoint, we may sometimes still be able to uniquely identify each state of $\tilde{Z}^*(t)$ after dropping the last component $I(t)$. In this case, $I(t)$ is a conceptual tool to help us define $\tilde{Z}^*(t)$. For example, if the last stage of the passage contains a critical service center which can not be reached by the tag job until the end of the passage time, $I(t)$ can be dropped at the final step even if the passage is a loop. Nevertheless, in the general case, $I(t)$ is required.

Fruthermore, the definition of passage time can be generalized in the following way. We can modify the definition of $I(t)$ so that it will not be

20

set to 1 until the underlying process $\tilde{Z}(t)$ not only passes through some state in B_1 but also transits to appropriate next states when it departs from that state in B_1. The setting of $I(t)$ can also be nondeterministic. The rule can be that when the underlying process $\tilde{Z}(t)$ jumping from state b in B_1 to state d, $I(t)$ will be set to 1 with probability g. That is to say $P_{(b,0),(d,1)} = gP_{bd}$ and $P_{(b,0),(d,0)} = (1-g) P_{bd}$. An example of this type will be that a passage time terminates at the time it transits from server i to server j with probability g. In this case, we can choose B_1 to be the set of all states corresponding to the tag job being served at station i. At the next state transition of $\tilde{Z}^*(t)$, the component $I(t)$ will be set to 1 with probability g if the state transition is due to the transition of the tag job to station j. Similar generalization holds for A_1 and A_2.

We now proceed to evaluate the appropriate initial distribution for $\tilde{Z}^*(t)$. First we need to find the steady state distribution of $\tilde{Z}(t) = (\tilde{S}(t), \tilde{K}(t))$. The steady state distribution of $\tilde{S}(t)$ is given in the previous section. Let $\{P(\tilde{s})\}$ be the steady state distribution of $\tilde{S}(t)$ and $\{P(\tilde{s},k)\}$ be that of $\tilde{Z}(t)$. Assume the total number of jobs in the network is N. Clearly, each job has probability $1/N$ being taged. If the tag job is in a FCFS or LCFS service center, the steady state distribution at state $(\tilde{s},k)$ will be

$$P(\tilde{s},k) = \frac{1}{N} P(\tilde{s}) \qquad (4.1)$$

and if the tag job is in a processor sharing on infinite server center, the steady distribution at state $(\tilde{s},k)$ will be

$$P(\tilde{s},k) = \frac{N_1(\tilde{s},k)}{N} P(\tilde{s}) \qquad (4.2)$$

where $N_1(\tilde{s},k)$ is the number of jobs in the same class and stage of service as the tag job.

To evaluate the initial distribution of $\tilde{Z}^*(t)$, we first consider the common case where each state $(\tilde{u},i)$ in A_1 can transit directly to some states in A_2 to start a new passage time[+]. The infinitesimal transition rate of the continuous time Markov chain $\tilde{Z}(t)$ from state $(\tilde{u},i)$ to state $(\tilde{s},k)$ is denoted by $q_{(\tilde{u},i)(\tilde{s},k)}$. For each state $(\tilde{s},k)$ in A_2^*, we will let $H(\tilde{s},k)$ denote the set of states in A_1^* which can transit directly to the state $(\tilde{s},k)$. The appropriate initial distribution will be

$$\Pi(\tilde{s},k,0) = \begin{cases} \dfrac{\displaystyle\sum_{(\tilde{v},j) \in H(\tilde{s},k)} P(\tilde{v},j) \, q_{(\tilde{v},j)(\tilde{s},k)}}{\displaystyle\sum_{\substack{(\tilde{u},i) \in H(\tilde{w},h) \\ (\tilde{w},h) \in A_2}} P(\tilde{u},i) \, q_{(\tilde{u},i)(\tilde{w},h)}} & \text{for } (\tilde{s},k,0) \in A_2^* \\[4ex] 0 & \text{otherwise} \end{cases} \tag{4.3}$$

$$\Pi(\tilde{s},k,1) = 0$$

For the general case, we need to calculate the taboo probability $_{A_1,A_2}f^*{}_{(\tilde{u},i)(\tilde{s},k)}$, which is defined to be the probability that starting from state $(\tilde{u},i)$ we can reach state $(\tilde{s},k)$ without passing through any states in A_1 and A_2 during the intermediate steps, from the transition maxtrix of the inbedded Markov chain of $\tilde{Z}(t)$. Furthermore, we define $_{A_1,A_2}f^*{}_{(\tilde{u},i),(\tilde{s},k)}$ to be $\delta_{(\tilde{u},i)(\tilde{s},k)}$ when both $(\tilde{u},i)$ and $(\tilde{s},k)$ are in A_2. Let

$$\tilde{c}(\tilde{s},k) = \sum_{\substack{(\tilde{v},j) \in A_1 \\ (\tilde{u},i) \notin A_1-A_2}} P(\tilde{v},j) \, q_{(\tilde{w},j)(\tilde{u},i)} \, {}_{A_1,A_2}f^*{}_{(\tilde{u},i)(\tilde{s},k)}$$

$$\text{for } (\tilde{s},k) \in A_2$$

The appropriate initial distribution will be

[+]Consider the case where the passage time terminates upon the tag job transiting from server x to server y.

$$\Pi(\tilde{s},k,0) = \begin{cases} \dfrac{\tilde{c}(\tilde{s},k)}{\displaystyle\sum_{(\tilde{u},i)\,\epsilon\,A_2} \tilde{c}(\tilde{u},i)} & \text{for } (\tilde{s},k,0)\,\epsilon\,A_2^{*} \\[2em] 0 & \text{otherwise} \end{cases}$$

(4.4)

$$\Pi(\tilde{s},k,1) = 0$$

After obtaining the appropriate initial distribution of $\tilde{Z}^{*}(t)$, let us start to solve for the distribution of the hitting time to the absorbing state $\tilde{r}$ for each state of $\tilde{Z}^{*}(t)$. Let

$F;(t)$ be the holding time distribution function at state $\tilde{w}$, and

$G_{\tilde{w}}(t)$ be the distribution function of the hitting time to absorbing state $\tilde{r}$ when starting from state $\tilde{w}$. Then decomposing over the holding time and possible next states of $\tilde{w}$, we get

$$G_{\tilde{w}}(t) = F_{\tilde{w}}(t) * \left(\sum_{\tilde{v} \neq \tilde{r}} P_{\tilde{w}\tilde{v}}\, G_{\tilde{v}}(t) + P_{\tilde{w}\tilde{r}} \right)$$

$$= \int_0^t \sum_{\tilde{v} \neq \tilde{r}} P_{\tilde{w}\tilde{v}}\, G_{\tilde{v}}(t-S)\, dF_{\tilde{w}}(S) + P_{\tilde{w}\tilde{r}}\, F_{\tilde{w}}(t) \qquad (4.5)$$

where $\displaystyle\sum_{\tilde{v} \neq \tilde{r}}$ means summation over all states in the state space of the Markov chain except $\tilde{r}$, the absorbing state.

Assume

$$F_{\tilde{w}}^{*}(S) = \mathcal{L}\{ F_{\tilde{w}}(t) \}$$

$$= \int_0^{\infty} e^{-st}\, dF_{\tilde{w}}(t)$$

$$G_{\tilde{w}}^{*}(S) = \mathcal{L}\{ G_{\tilde{w}}(t) \}$$

$$= \int_0^{\infty} e^{-st}\, dG_{\tilde{w}}(t)$$

i.e. $F^*_{\underset{w}{\sim}}(S)$ and $G^*_{\underset{w}{\sim}}(S)$ are the Laplace-Stieltjes transforms of $F_{\underset{w}{\sim}}(t)$ and $G^*_{\underset{w}{\sim}}(t)$, respectively.

Taking Laplace-Stieltjes transform on both sides of (4.5) we get

$$G^*_{\underset{w}{\sim}}(S) = F^*_{\underset{w}{\sim}}(S) \left(\sum_{\substack{\underset{v}{\sim} \\ v \neq r}} P_{\underset{w}{\sim}\underset{v}{\sim}} G^*_{\underset{v}{\sim}}(S) + P_{\underset{w}{\sim}r} \right) \tag{4.6}$$

$$\sum_{\substack{\underset{v}{\sim} \\ v \neq r}} (P_{\underset{w}{\sim}\underset{v}{\sim}} F^*_{\underset{w}{\sim}}(S) - \delta_{\underset{ww}{\sim\sim}}) G^*_{\underset{v}{\sim}}(S) = -P_{\underset{wr}{\sim}} F^*_{\underset{w}{\sim}}(S) \tag{4.7}$$

where $\delta_{\underset{ww}{\sim\sim}} = \begin{cases} 1 & \text{if } \underset{w}{\sim} = \underset{v}{\sim} \\ 0 & \text{otherwise} \end{cases}$

Equation (4.7) represents a system of linear equations with finite dimension. After solving for $G^*_{\underset{w}{\sim}}(S)$, we can get $G_{\underset{w}{\sim}}(t)$ by taking the inverser Laplace-Stieltjes transform

Finally, we get the passage time distribution function

$$G(t) = \sum_{\underset{w}{\sim} \, \varepsilon \, A^*_2} \Pi(\underset{w}{\sim}) G_{\underset{w}{\sim}}(t)$$

Before closing the section, we would like to point out the fact that if we only tag the job from a specific class, we can obtain the passage time distribtuion for a specific class of jobs directly by following the same procedure.

24

5.   ILLUSTRATIVE **EXAMPLE**

Consider the simple queueing network model shown in Fig. 5.1. This
model is often encountered in modeling multiprogrammed computer systems.
Under this interpretation the two service stations will be referred to as
the CPU and I/O unit and the implication of a closed queueing network is that
the degree of multiprogramming is fixed.   Upon completion of service at the
CPU, the job rejoins the tail of the queue in the CPU with probability ψ
and that in the I/O unit with probability $1-\psi$. Neither the CPU nor the
I/O unit is subject to preemption and both follows the first-come first-
served discipline.   The passage time of interest will be the passage time
denoted by T which is measured from entrance into the CPU queue until
completion of the service at the I/O unit and rejoining the queue at the
CPU,. Also possible of interest in the model will be the passage time denoted
byT*defined as the time measured from entrance to the queue at the CPU after
I/O completion until entrance to the queue at the I/O unit.

For ease of illustration, we will assume there is only one class of jobs
in the system   The state variable $\tilde{Z}^*(t)$ can now be simplified into

$$\tilde{Z}^*(t) = (Q_0(t), Q_1(t), K(t), I(t))$$

where

$Q_0(t)$ = the number of jobs waiting or being served at the CPU

$Q_1(t)$ = the number of jobs waiting or being served at the I/O unit

$K(t)$ = position of the tag job in the network counting from the
tail of the CPU queue toward the head of the I/O queue

$I(t)$ = indication of whether the system ever passing though the
states in B_1^* as defined below

We can further simplify the state variable using the fact that the total number of jobs are fixed. After simplification, the state variable $\tilde{Z}^*(t)$ will be defined as

$$\tilde{Z}^*(t) = (Q_0(t), K(t), I(t))$$

with state space E^* and forms a continuous time Markov chain. For this model the state space E^* is given by

$$E^* = \{(j, k, i) : 0 \le j \le N; \ 1 \le k \le N; \ 0 \le i \le 1\}$$

where N is the number of jobs in the network.

Recall the sequence of refinements needed to obtain $\tilde{Z}^*(t)$ described in section 4. The sets A_1, A_2, B_1, B_2 defining the start point and end point of the passage time T of the tag job in the stochastic process $\tilde{Z}(t) = (Q_0(t), K(t))$ are given by

$$A_1 = B_1 = \{(i,N) : 0 \le i < N\}$$
$$= \{(0,2), (2,1)\}$$
$$A_2 = B_2 = \{(i,1) : 0 < i \le N\}$$
$$= \{(1,1), (2,1)\}$$

and the corresponding A_2^*, B_1^* and B_2^* in $\tilde{Z}^*(t)$ are

$$A_2^* = \{(i,1,0) : 0 < i \le N\}$$
$$= \{(1,1,0), (2,1,0)\}$$

$$B_2^* = \{(i,1,1) : 0 < i \le N\}$$
$$= \{(1,1,1), (2,1,1)\}$$

$$B_1^* = \{(i,2,0) : 0 \le i < N\}$$
$$= \{(0,2,1), (1,2,0)\}$$

The absorbing state is formed by lumping the states in B_2^*, i.e. (1,1,1) and (2,1,1), together.
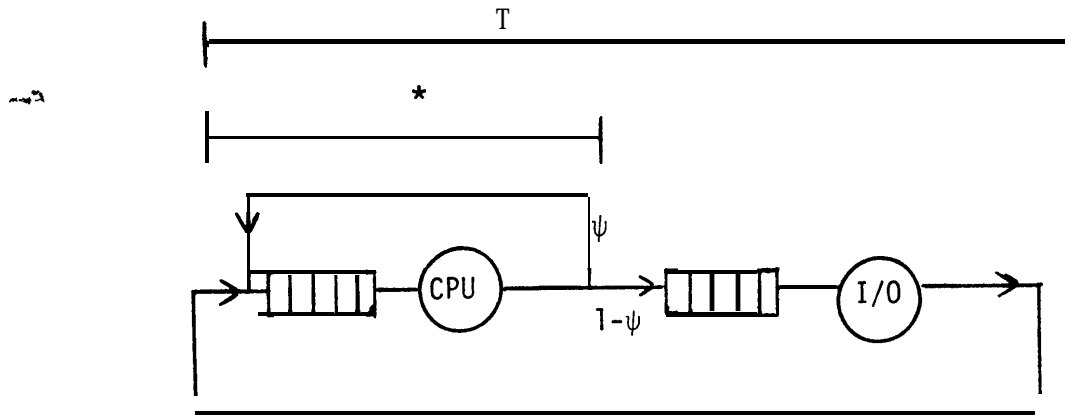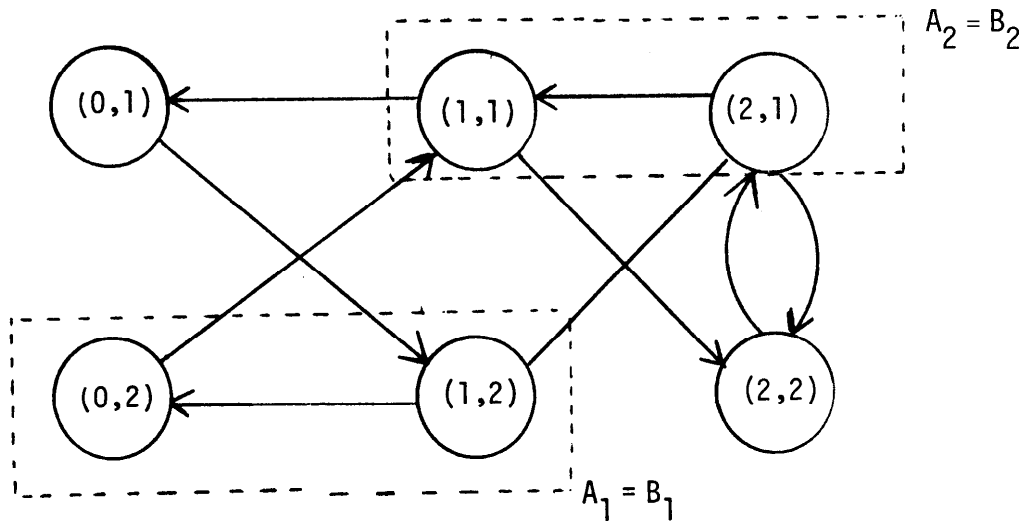
Fig. 5.1:   A Closed Two Server Queueing Model

Fig. 5.2:   State transition diagram of $x(t) = (Q_0(t), K(t))$ and subsets A_i's and B_i's for passage time T

Now we go through the refinement procedure step by step using a very simple case where N= 2 jobs. In Fig. 5.2, we show the state transition diagram of $\tilde{Z}(t) = (Q_r(t), K(t))$. There are six states in total. After adding the component $I(t)$ to the state variable, the state transition diagram is shown in Fig. 5.3. There are 12 states in total. Only the 11 states reachable from the permissible initial states which are the states in A_2^* are shown. Finally after merging the states in B_2^* into an absorbing state $\tilde{r}$, the transition diagram of the Markov chain is shown in Fig. 5.4. The corresponding A_i's and B_i's or A_i^*'s and B_i^*'s are also indicated in each figure. For comparison, in Fig. 5.5 and 5.6, we indicate the A_i's and B_i's of $\tilde{Z}(t)$ and A_i^*'s and B_i^*'s of the corresponding $\tilde{Z}^*(t)$ on their transition diagrams for passage time T^*, respectively.
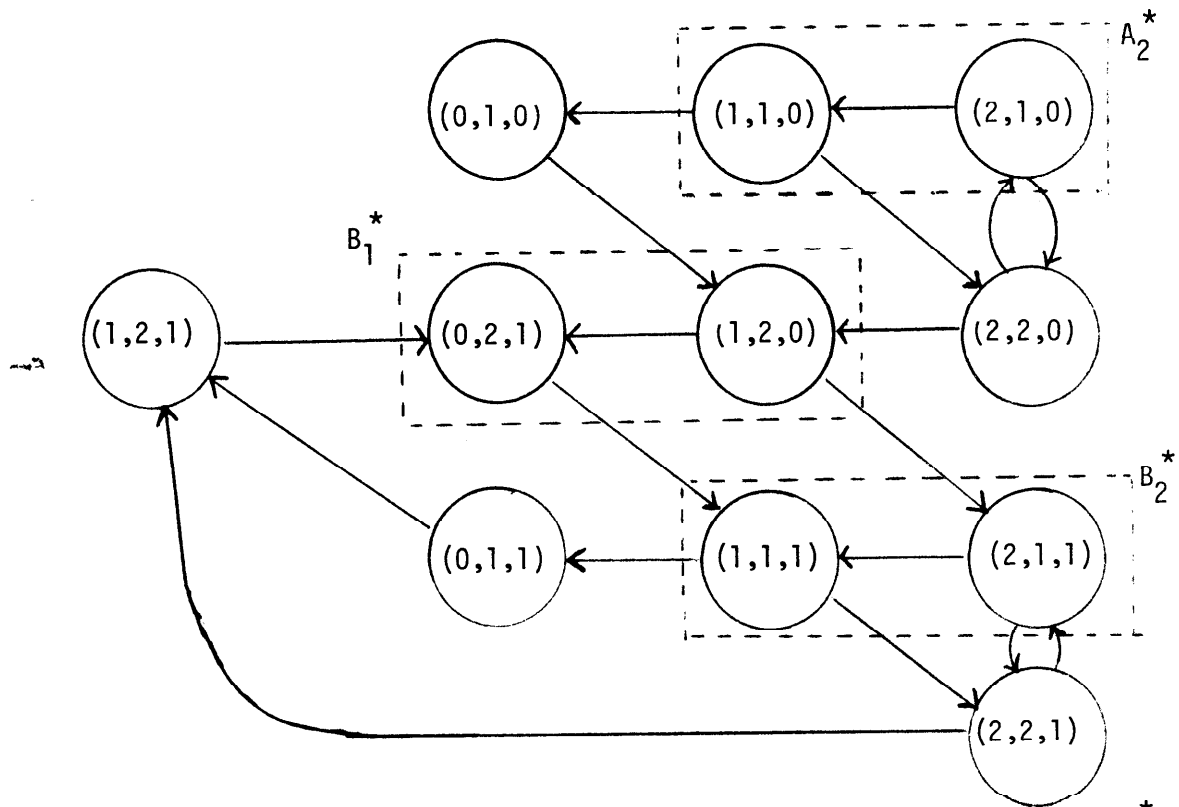
Fig. 5.3: State transition diagram of $(Q_0(t), K(t), I(t))$ before B_2^* is lumped into $\tilde{r}$ and subsets A_2^*, B_1^* and B_2^* for passage time T

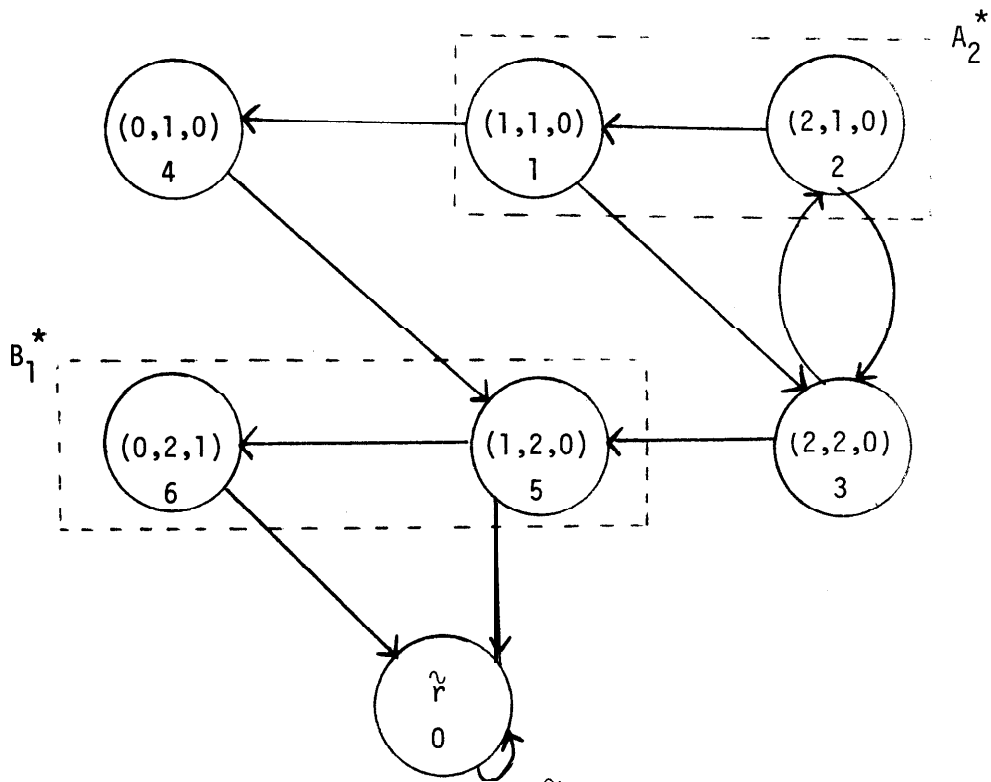Fig. 5.4: State transition diagram of $\tilde{Z}^*(t)$ and subsets A_2^* and B_1^* for passage time T
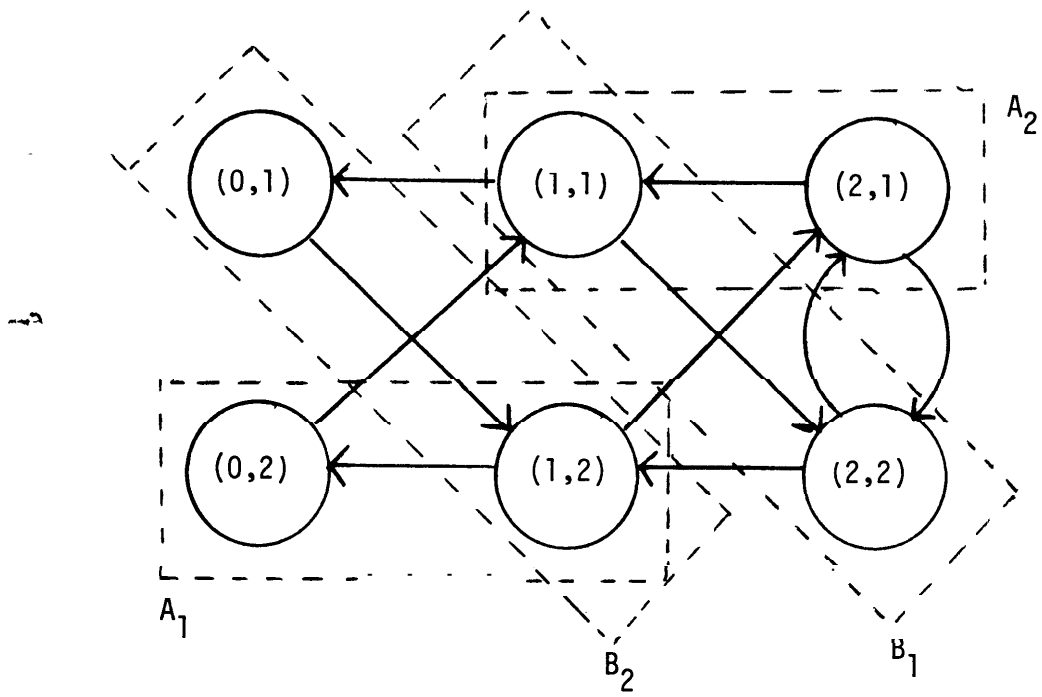
Fig. 5.5: State transition diagram of $\tilde{Z}(t) = (Q_0(t), K(t))$ and subsets A_i and B_i for passage time T^*
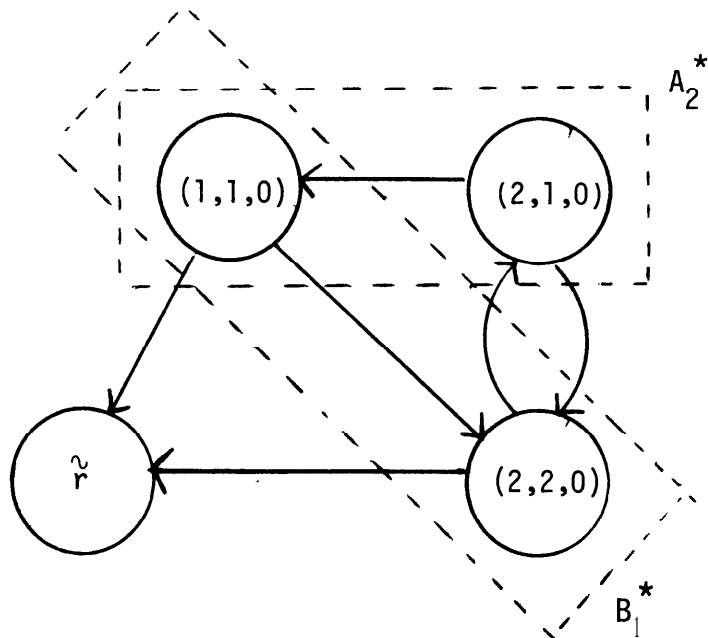

Fig. 5.6: State transition diagram of $\tilde{Z}^*(t)$ and subsets A_2^* and B_1^* for passage time T^*

30

To simplify the notation we will relabel the states as shown in Table 5.1

Relabelled state	Original state
0	$\tilde{r}$
1	(1,1,0)
2	(2,1,0)
3	(2,2,0)
4	(0,1,0)
5	(1,2,0)
6	(0,2,1)

Table 5.1:   Relabelling Table

Assume the means of the exponential service time distributions at the CPU and I/O unit are $\frac{1}{\lambda}$ and $\frac{1}{\mu}$, respectively. We get the following infinitesimal generator [11], Q, for the continuous time Markov chain $\tilde{Z}^*(t)$.

$$
Q=[q_{ij}] =
\begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -(\mu+(1-\psi)\lambda) & 0 & \mu & (1-\psi)\lambda & 0 & 0 \\
0 & (1-\psi)\lambda & -\lambda & \psi\lambda & 0 & 0 & 0 \\
0 & 0 & \psi\lambda & -\lambda & 0 & (1-\psi)\lambda & 0 \\
0 & 0 & 0 & 0 & -\mu & \mu & 0 \\
\mu & 0 & 0 & 0 & 0 & -(\mu+(1-\psi)\lambda) & (1-\psi)\lambda \\
\mu & 0 & 0 & 0 & 0 & 0 & -\mu
\end{bmatrix}
$$

and the transition matrix, **P**, of the imbedded Markov chain can be obtained from **Q** as [11].

$$
P = [P_{ij}] = \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \dfrac{\mu}{\mu+(1-\psi)\lambda} & \dfrac{(1-\psi)\lambda}{\mu+(1-\psi)\lambda} & 0 & 0 \\
0 & (1-\psi) & 0 & \psi & 0 & 0 & 0 \\
0 & 0 & \psi & 0 & 0 & (1-\psi) & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\dfrac{\mu}{\mu+(1-\psi)\lambda} & 0 & 0 & 0 & 0 & {}_{\circ} & \dfrac{(1-\psi)\lambda}{\mu+(1-\psi)\lambda} \\
1 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
$$

Let $F_i(t)$ be the holding time distribution of state i

$G_i(t)$ be the hitting time distribution to state 0 from state i,

and $F_i^*(S)$ and $Gi^*(S)$ be the corresponding Laplace-Stieltjes transforms, respectively, as before.

Since the holding time is exponentially distributed,

$$
F_i^*(S) = \frac{q_i}{S+q_f}
$$

where $q._1 = -q._{11}$

Furthermore, let

$$
R = [r_{ij}, \ 1 \le i, \ j \le 6]
$$

where $r._{ij} = P_{ij} \, F_i^*(S) - \delta_{ij}$

then clearly

$$
R = \begin{bmatrix}
-1 & 0 & \dfrac{\mu}{S+\phi} & \dfrac{(1-\psi)\lambda}{S+\phi} & 0 & 0 \\[2mm]
\dfrac{(1-\psi)\lambda}{S+\lambda} & -1 & \dfrac{\psi\lambda}{S+\lambda} & 0 & & 0 \\[2mm]
0 & \dfrac{\psi\lambda}{S+\lambda} & -1 & 0 & \dfrac{(1-\psi)\lambda}{S+\lambda} & 0 \\[2mm]
0 & 0 & 0 & -1 & \dfrac{\mu}{S+\mu} & 0 \\[2mm]
0 & 0 & 0 & 0 & -1 & \dfrac{(1-\psi)\lambda}{S+\phi} \\[2mm]
0 & 0 & 0 & 0 & 0 & -1
\end{bmatrix}
$$

where $\phi = \mu + (1-S)X$

From equation (4.7), we get

$$
R \begin{bmatrix} G_1^*(S) \\[1mm] G_2^*(S) \\[1mm] G_3^*(S) \\[1mm] G_4^*(S) \\[1mm] G_5^*(S) \\[1mm] G_6^*(S) \end{bmatrix} = \begin{bmatrix} 0 \\[1mm] 0 \\[1mm] 0 \\[1mm] 0 \\[1mm] -g_1 \\[1mm] -g_2 \end{bmatrix}
$$

where
$$g_1 = \frac{\mu}{S+\phi}$$

and
$$g_2 = \frac{\mu}{S+\mu}$$

After simplifications, we get

$$G; (S) = \frac{(r_{23}\, r_{35} + r_{21}\, r_{13} r_{35} + r_{21}\, r_{14}\, r_{45})(g_1 + r_{56}\, g_2)}{1 - r_{23}\, r_{32} - r_{21}\, r_{13}\, r_{32}}$$

$$G_1^*(S) = r13\, r_{32}\, F_2^*(S) + (r_{13}\, r_{35} + r_{14}\, r_{45})(g_1 + r_{56}\, g_2)$$

33

Both $G_1^*(S)$ and $G_2^*(S)$ are rational functions of S. After simplification and taking inverse transform, we can get $G_1(t)$ and $G_2(t)$.

From theorem 3.1, we obtain the initial distribution

$$\Pi(1) = \frac{\lambda(1-\psi)}{\lambda(1-\psi)+\mu}$$

$$\Pi(2) = \frac{\mu}{\lambda(1-\psi)+\mu}$$

Hence, the distribution of passage time T is

$$G(t) = \frac{\lambda(1-\psi)}{\lambda(1-\psi)+\mu}\ G_1(t) + \frac{\mu}{\lambda(1-\psi)+\mu}\ G_2(t)$$

Let us look at the case where $\lambda = 1$, $\mu = 0.5$, and $\psi = 0.75$. The values of the parameters are the same as those used by Iglehart and Shedler [14] where the regenerative simulation is used to estimate the passage time distributions of the same system. After simplification, we get

$$G_1^*(S) = \frac{32S^2 + 56S + 15}{(64S^3 + 176S^2 + 124S + 15)(S+1)(2S+1)^2}$$

$$G_2^*(S) = \frac{(6S+5)\ (4S+3)}{(64S^3 + 176S^2 + 124S + 15)(2S+1)^2}$$

and

$$\Pi(1) = \frac{1}{3}$$

$$\Pi(2) = \frac{2}{3}$$

-- Finally, by taking the inverse Laplace transform, we get the passage time distribution.

$$G(t) = 1.61832(1-e^{-0.1519139t}) + 0.121284(1-e^{-0.9186508t})$$
$$+ 0.0234787(1-e^{-1.679435t}) - 0.0793388(1-e^{-0.5t})$$
$$+ 0.0303030(1-e^{-0.5t} - \frac{t}{2}\ e^{-0.5t})$$

which is a combination of exponential and Gamma distributions.

6. NUMERICAL APPROXIMATION ON PASSAGE TIME DISTRIBUTIONS FOR THE CLOSED QUEUEING NETWORKS

From the example in the previous section, we can see as the complexity of the queueing network increases, the number of states increases rapidly and the manipulations on transfer functions become very tedious. In this section, a numerical approximation method is proposed. The basic idea is to estimate the discrete approximation of the passage time distribution instead. Under the discrete approximation, all the convolution integrals become recurrence relations, so the enumeration on computers is straightforward.

Recall $F_{\tilde{W}}(t)$ is the holding time distribution function at state $\tilde{W}$ and is exponentially distributed. Assume that its mean is $\frac{1}{\lambda_{\tilde{W}}}$. If we discretize the density function $dF_{\tilde{W}}(t)/dt$ into a string of impulses separated by d as in fig. 6.1 and set the magnitude of each impulse equal to the area under the density function on its left hand interval, we get a geometric distribution $P_{F\tilde{W}}(k)$, where

$$P_{F\tilde{W}}(k) = \begin{cases} e^{-\lambda_{\tilde{W}}(k-1)d} - e^{-\lambda_{\tilde{W}}kd} & \text{for } k > 1 \\ 0 & \text{for } k=0 \end{cases}$$

As d decreases, the accuracy increases and the efficiency decreases. In Fig. 6.1a and 6.1b, we display the density function $dF_{\tilde{W}}(t)/dt$ and its discrete approximation $P_{F\tilde{W}}(k)$, respectively. Note the reason why the discretization has a delay or shift cl comes from the fact that no probability mass should concentrate on the origin for holding time distributions in general.
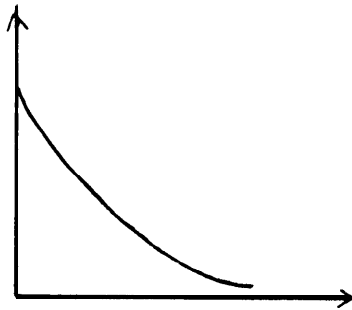
(a) density function $dF_{\tilde{W}}(t)/dt$

$1 - e^{-\lambda_{\tilde{W}}d}$
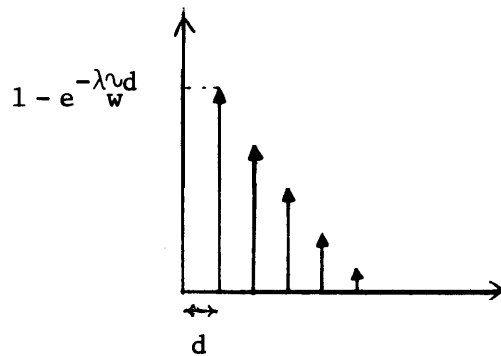
(b) discrete approximation $P_{F_{\tilde{W}}}(k)$

Fig. 6.1: Density function $dF_{\tilde{W}}(t)/dt$ and its discrete

approximation $P_{F_{\tilde{W}}}(k)$

Let $P_{G\tilde{W}}(j)$ be the discrete approximation of the hitting time density function $dG_{\tilde{W}}(t)/dt$.

The discrete version of (4.5) now becomes

$$P_{G\tilde{W}}(j) = \begin{cases} \sum\limits_{i=1}^{j} P_{F\tilde{W}}(i) \sum\limits_{\tilde{V}\in E}^{*} P_{\tilde{W}\tilde{V}} \; P_{G\tilde{V}}(j-i) & \text{for } \tilde{w} \neq \tilde{r} \\[2ex] 0 & \text{for } \tilde{w} = \tilde{r} \end{cases} \quad j \geq 1 \quad (6.1)$$

and

$$P_{G\tilde{W}}(0) = \begin{cases} 0 & \text{for } \tilde{w} \neq \tilde{r} \\[2ex] 1 & \text{for } \tilde{w} = \tilde{r} \end{cases}$$

Furthermore, using the fact that the holding time is geometrically distributed, the above equations for $P_{G\tilde{W}}(j)$ can be further simplified into

$$P_{G\tilde{W}}(j) = \begin{cases} \sum\limits_{\tilde{V}\in E}^{*} P_{\tilde{W}\tilde{V}}^{*} \; P_{G\tilde{V}}(j-1) & \text{for } \tilde{w} \neq \tilde{r} \\[2ex] 0 & \text{for } \tilde{w} = \tilde{r} \end{cases} \quad j \geq 1 \quad (6.2)$$

and

$$P_{G\tilde{W}}(0) = \begin{cases} 0 & \text{for } \tilde{w} \neq \tilde{r} \\[2ex] 1 & \text{for } \tilde{w} = \tilde{r} \end{cases}$$

where

$$P_{\tilde{W}\tilde{V}}^{*} = \begin{cases} (1-e^{-\lambda_{\tilde{W}}d}) \; P_{\tilde{W}\tilde{V}} & \text{for } \tilde{v} \neq \tilde{w} \\[2ex] e^{-\lambda_{\tilde{W}}d} & \text{for } \tilde{v} = \tilde{w} \end{cases}$$

Clearly, the $P_{G\tilde{W}}(j)$'s can now be solved recursively for $j \geq 1$, starting with $P_{G\tilde{W}}(0) = \delta_{\tilde{W}\tilde{r}}$ for all $\tilde{w} \in E^{*}$.

Hence, $P_G(i)$ the discrete approximation of the passage time density function, $dG(t)/dt$, can be expressed as

$$P_G(i) = \sum_{\widetilde{w} = \epsilon \; A_2^*} \Pi(\widetilde{w}) \; P_{G\widetilde{w}}(i) \qquad\qquad (6.3)$$

where the initial distribution $\Pi(\widetilde{w})$ is given in (4.3) or (4.4).

By linear interpolation, we get

$$G(t) \stackrel{.}{\underset{\sim}{\sim}} \sum_{j=1}^{i} P_G(j) + \frac{t-id}{d} P_G(i+1) \qquad \text{for } id \le t < (i+1)d$$

Finally, let us apply the approximation technique to estimate the response time distribution of the multiprogrammed computer system model in the previous section. In table 6.1, we tabulate not only the approximate percentile response time distributions under two different d values, 0.05 and 0.1, but also the analytic result obtained in the previous section and the simulation result obtained by Iglehart and shedler [14]. The approximation leads to very satisfactory results especially when the size of d is small.

38

Percentile	Analytic result	numerical approximations		simulation results	
		d=0.05	d=0.1	point estimate	confidence interval
P{R < 2.333}	0.07659	0.07211	0.06777	0.07314	(0.06671, 0.07757)
P{R ≤ 4.667}	0.26900	0.2610	0.2530	0.2572	(0.2482, 0.2662)
P{R ≤ 9.333}	0.61379	0.6045	0.5972	0.6174	(0.6071, 0.6277)
P{R ≤ 14.0}	0.80757	0.8002	0.7928	0.8084	(0.8001, 0.8251)
P{R ≤ 18.667}	0.90509	0.9000	0.8949	0.9012	(0.8948, 0.9076)

Table 6.1: Comparison of analytic result, numerical approximation and simulation result on percentile response time

7. APPROXIMATE PASSAGE TIME DISTRIBUTIONS FOR THE OPEN QUEUEING NETWORKS

The major problem encountered on tackling open queueing network is that the state space of the network is infinite. We first use a specific passage time which is the response time of a job to illustrate the similarities and differences encountered in handling open queueing networks and closed queueing networks. We assume that the network has already been in steady state at t = 0- and a job arrives at t = 0-. We will use this newly arrival job as our tag job. The state variable of the network will still take the form

$$\tilde{Z}{}^{*}(t) = (1;(t),\ K(t),\ I(t))\ t \geq 0$$

as that in the closed queueing network. The set A_2^{*}, B_1^{*}, and B_2^{*} will also be defined as before. Actually, the set B_1^{*} is redundant in this case, but for more general passage time it is indeed required. The states in A_2^{*} which correspond to all possible start points of response times have the form $(\tilde{s}, k, 0)$ where the k-th component of $\tilde{s}$ describes a job which may be considered to be a new arrival to the queueing network. To be more specific, a job can be considered as the newly arrival job to the queueing network, if it is the last job in the queue of a FCFS service center, or the job currently being served in the first stage of a LCFS service center, or any job in its first stage of service in a PS or IS service center. The set B_2^{*} which corresponds to all possible end points of response times and will eventually be lumped into an absorbing state consists of all states which can be the next states immediately after an exit of the tag job from the queueing network. The states in B_2^{*} have the form $(\tilde{s}, 0, 1)$. Notice we set K(t) equal to zero to indicate that the tag job has left the network by time t.

We now proceed to evaluate the initial distributions of the states in A_2^* under our previous assumptions. Let us for the moment return to the state description of $S(t)$ with state space E cited in section 3. Let $H(\tilde{s},k)$ be the state that leads to $\tilde{s}$ after an external arrival whose class and entering service center is described by the k-th component of $\tilde{s}$ where both $\tilde{s}$ and $H(\tilde{s},k) \in E$. Here we implicitly assume that the k-th component of $\tilde{s}$ can indeed represent a new arrival to the network. Notice given $\tilde{s}$ and k, the preceding state $H(\tilde{s},k)$ is unique. To be more precise, if the arrival enters from a FCFS or LCFS service center, $H(\tilde{s},k)$ is obtained by deleting the k-th component of $\tilde{s}$ and if the arrival enters from a PS or IS service center, $H(\tilde{s},k)$ is obtained by decreasing the k-th component of $\tilde{s}$ by 1 and deleting it if it drops to zero after decrementation. Let $C(\tilde{s},k)$ be the service class of the new arrival described by the k-th component of $\tilde{s}$ and $T(\tilde{s},k)$ be its entering service center. Recall that $P^A_{i,r}$ is defined in section 3 to be the probability that the new arrival will be of class r and entering from service center i. Clearly, the transition probability from $H(\tilde{s},k)$ to $\tilde{s}$ conditioning on a new arrival will be $P^A_{T(\tilde{s},k),C(\tilde{s},k)}$. Finally the probability that conditioning on a new arrival with class $C(\tilde{s},k)$ and entering service center $T(\tilde{s},k)$, the network will be in state $\tilde{s}$ is given by

$$P_0(\tilde{s},k) = P(H(\tilde{s},k)) \, P^A_{T(\tilde{s},k),C(\tilde{s},k)} \qquad (7.1)$$

where $P(H(\tilde{s},k))$ can be evaluated by Theorem 3.1. Furthermore, the above probability is in fact the initial probability distribution of the state $(\tilde{s},k,0)$ in A_2^* of the augmented state space E^*, i.e. $\pi(\tilde{s},k,0)=P_0(\tilde{s},k)$.

Following the same argument for closed queueing networks, we can derive the same equations as in section 4 for $G_{\tilde{w}}(t)$ and $G^*_{\tilde{w}}(S)$, the hitting time

41

distribution and its Laplace-Stieltjes transform for each state $\tilde{w}$ in E^*, respectively. That is in time domain

$$G_{\tilde{w}}(t) = \int_0^t \sum_{\substack{\tilde{v} \\ \tilde{v} \neq \tilde{r}}} P_{\tilde{w}\tilde{v}} \, G_{\tilde{v}}(t-S)dF_{\tilde{w}}(S) + P_{\tilde{w}\tilde{r}} \, F_{\tilde{w}}(t) \qquad (7.2)$$

in transform domain

$$\sum_{\substack{\tilde{v} \\ \tilde{v} \neq \tilde{r}}} (P_{\tilde{w}\tilde{v}} \, F_{\tilde{w}}^*(S) - \delta_{\tilde{w}\tilde{v}}) \, G_{\tilde{v}}^*(S) = -P_{\tilde{w}\tilde{r}} \, F_{\tilde{w}}^*(S) \qquad (7.3)$$

But the system of equations appeared in (7.3) now has infinite dimensions. Clearly, we only need to consider a finite subset, $D^* = \{(\hat{S}, \hat{K}, i) : S \varepsilon D\}$, in the state space of $\tilde{Z}^*(t)$ where D is the most frequently occurred states of $\tilde{Z}(t)$ such that the total steady state probability of $\tilde{Z}(t)$ in D, which can be derived from theorem 3.1, is close to 1. By neglecting the other states not in that subset, the system of equations reduces to a finite set of linear equations. We note that after the state reduction the transition probabilities should be normalized.

To simplify the problem, we can again apply the discretization technique introduced in section 6 to obtain recurrence relations for the discrete approximation of the density function of the hitting time for each state $\tilde{w}$. The recurrence relations are similar to (6.2) except that now we have infinite simultaneous recurrence relations due to the infinite state space. Using the same principle cited above to reduce the state space to D^*, the problem becomes solvable. The discrete approximation of the density function of the response time is again give by (6.3) except that $\sum_{(\tilde{s},k,0)\varepsilon A_2^*}$ becomes an infinite summation. By considering only the intersection of A_2^* and D^*,

the summation becomes a finite summation. As before, all the probabilities
should be normalized after the state reduction. As long as the traffic in
the network is not heavy, all the infinite summations appeared above will
converge very quickly and only a reasonable number of states need to be
considered.

Alternatively, we can approximate an open system by a closed system
and apply the result in section 6 directly. Since the queue length distribution
of the system is known, we can calculate the quantile of queue length
distribution with no difficulties. Let N be the 95% or 99% quantile of the
distribution of the total number of jobs in the system depending upon the
accuracy desired. That is to say 95% or 99% of the time the number of jobs
in the network is less than or equal to N. So if we consider a queueing
network which is identical to the original network in all respects except
that the arrival process is shut down when the number of customers in the
system is equal to N, the performance difference should be very minor. The
new queueing network can be viewed as a closed queueing network with N
customers as shown in Fig. 7.1. In fact, when N → ∞, the performance
the two network models becomes identical. As long as the traffic in the network
is not heavy, N will be reasonably small.

Let us consider an example. The example chosen is an open two server
queueing model. As in the closed two server queueing model, the servers can be
interpreted as CPU and I/O unit, respectively. Now the total number of jobs being
activated is assumed to be variable. The open queueing model and its approximate
closed queueing model are given in Fig. 7.2a and 7.2b respectively. The passage
time under interest is the response time which is the time between arrival and
departure of a job to the system. In Fig. 7.2b, the equivalent passage time in
the closed network is the time measured from entrance into the CPU queue from
the source until the entrance into the source after service completion at the
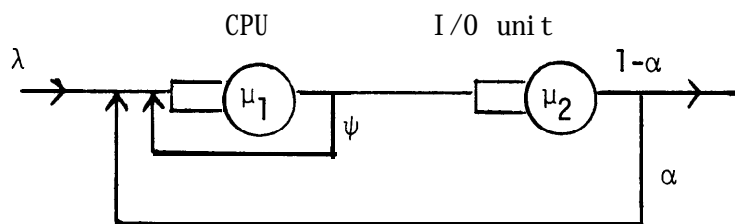I/O unit.

43
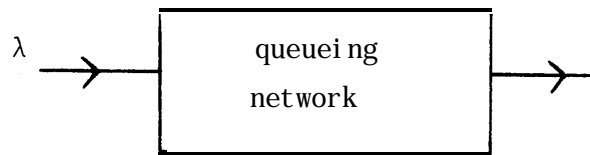
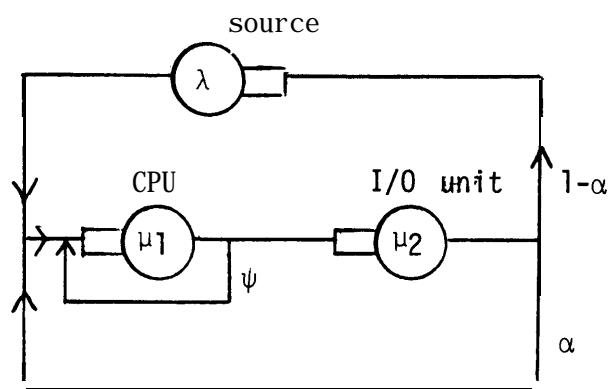(a) open queueing network model
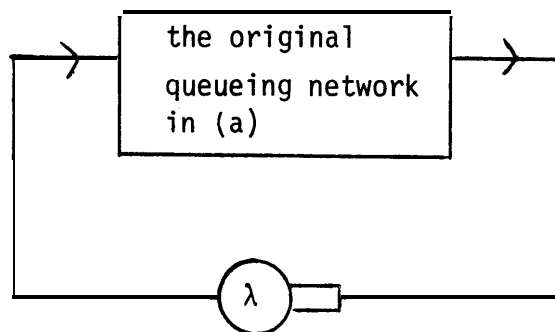
(b) approximate closed queueing network model

Fig. 7.1: An Open Queueing Network Model and Its Approximate Closed Queueing Network Model

(a) open queueing network model

(b) approximate closed queueing network model

Fig. 7.2: An Open Two Server Queueing Model and Its Approximate Closed Queueing Network Model

44

In Table 7.1, we compare the simulation result and the approximation results on percentile response time when $\psi = 0.3$, $a = 0.1$, $A = 0.1$, $\mu_1 = 1$ and $\mu_2 = 0.5$. The simulation result is obtained via regenerative simulation where not only the point estimation but also the 95% confidence interval are provided. We evaluate the approximate percentile response time under two different cut off values, i.e. $N = 3$ and 4. Furthermore, the interval of discrete approximation, d, is chosen to be 0.05. As we can see the approximations obtained under different N's have little difference since the system seldom has more than 3 jobs simultaneously. The approximation results are also very close to the simulation results. In Table 7.2, we show the approximation results for $N = 3$ and 4 when the value of d is doubled. The results are still very nice. In Table 7.3, we compare the simulation result and the approximation results on percentile response time when ψ increases to 0.75. The traffic intensity increases in this case, so does the appropriate cut off value on the arrival process. We consider two different approximatfons for $N = 7$ and 8, respectively. The value of d is chosen to be 0.05. Again, not only the approximate results under different cut off values have very little difference, but also they are close to the simulation result.

Percentile	Simulation	approximation	
		N=3	N=4
$\{R \leq 1\}$	0.0758 + 0.0059	0.0736	0.0723
$\{R \leq 2\}$	0.227 + 0.010	0.225	0.221
$\{R \leq 3\}$	0.384 + 0.012	0.386	0.379
$\{R \leq 4\}$	0.523 + 0.013	0.528	0.520
$\{R \leq 5\}$	0.640 + 0.013	0.644	0.634
$\{R \leq 6\}$	0.729 + 0.012	0.734	0.724
$\{R \leq 7\}$	0.798 + 0.011	0.802	0.793
$\{R \leq 8\}$	0.847 0.010	0.854	0.846
$\{R \leq 9\}$	0.884 + 0.009	0.892	0.885
$\{R \leq 11\}$	0.935 + 0.007	0.941	0.936

Table 7.1: Percentile Response Time When $+= 0.3$, $a = 0.1$, $\lambda = 0.1$
$\mu_1 = 1$, $\mu_2 = 0.5$, and $d = 0.05$

Percentile	approximation	
	N=3	N=4
P{R ≤ 1}	0.0695	0.0682
P{R ≤ 2}	0.219	0.215
P{R ≤ 3}	0.378	0.372
P{R ≤ 4}	0.520	0.512
P{R ≤ 5}	0.636	0.627
P{R ≤ 6}	0.727	0.718
P{R ≤ 7}	0.797	0.788
P{R ≤ 8}	0.849	0.841
P{R ≤ 9}	0.888	0.881
P{R ≤ 11}	0.939	0.933

Table 7.2: **Percentile Response Time** When ψ = **0.3,** α = **0.1**
λ = **0.1,** μ_1 = **1,** μ_2 = **0.5 and d = 0.1**

Percentile	Simulation	approximation	
		N=7	N=8
P{R ≤ 1}	0.0202 ± 0.0026	0.0208	0.0207
P{R ≤ 2}	0.0755 ± 0.0053	0.0752	0.0749
P{R ≤ 3}	0.148 ± 0.0076	0.148	0.147
P{R ≤ 5}	0.304 ± 0.011	0.306	0.305
P{R ≤ 7}	0.443 ± 0.013	0.448	0.447
P{R ≤ 9}	0.562 ± 0.013	0.564	0.562
P{R ≤ 11}	0.653 ± 0.13	0.653	0.652
P{R ≤ 13}	0.723 ± 0.013	0.723	0.721
P{R ≤ 16}	0.797 ± 0.012	0.799	0.797
P{R ≤ 20}	0.867 ± 0.010	0.865	0.864

Iable **7.3:** **Percentile Response Time When** ψ = **0.75, a = 0.1**
λ = **0.1,** μ_1 = **1,** μ_2 = **0.5, and d = 0.05**

8. A CASE STUDY ON INTERFAULT TIME DISTRIBUTIONS AND RESPONSE TIME DISTRIBUTIONS OF MULTIPROGRAMMED COMPUTER SYSTEMS

We now proceed to investigate the distribution and density functions of two important quantities encountered in analyzing the performance of a multi-programmed computer system, namely interfault time and response time. Both --quantities are measured in real time. The number of active processes will affect the interfault time and response time on two folds. It not only explicitly implies the contention level on the processors but also implicitly implies the contention level on main memory. The amount of memory allocated will have a drastic effect on the number of instructions executed between page faults, i.e. the virtual interfault time. In order to capture the memory effect on the performance, we will take the approach of hierarchical modeling. In the first level of modeling hierarchy we only consider the interactions between the CPU and paging device. The closed two server queueing model in Fig. 8.1a is used to represent the CPU-PGU subsystem The interfault time of a process is defined to be the time between two consecutive epochs that the process enters the CPU queue after receiving service from the paging device. The stoppage of CPU processing can either be due to the expiration of allocated time slice or a page fault. In the first case, the process rejoins the CPU queue and in the second case, it joins the queue of the paging device. In the second level of modeling hierarchy, we consider the interaction between the CPU-PGU subsystem and file I/O device. Another closed two server queueing model in Fig. 8.1b is used for this case. The response time of a process is defined to be the time between two consecutive time epochs that the process passes through the self loop of the CPU-PGU subsystem That is to say a transfer through the self loop can be viewed as a termination of a process and entering of a new process at the same instant.

47
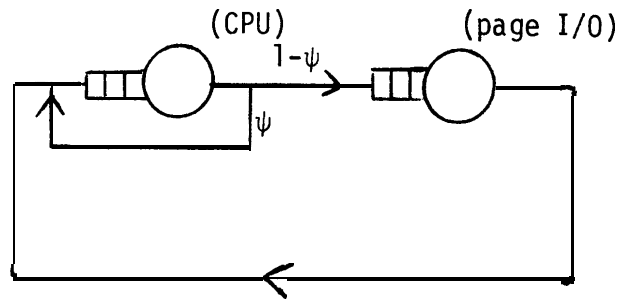
(a) CPU-PGU model (first level)
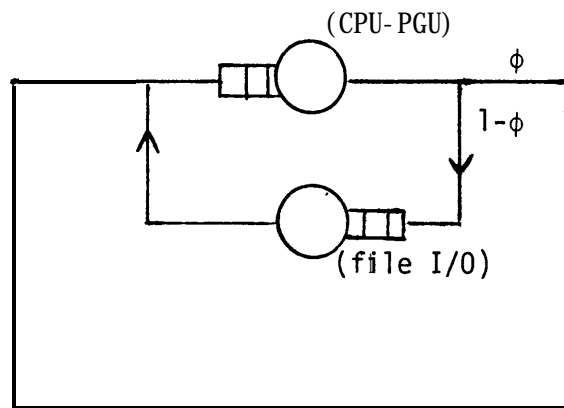
(b) Multiprogrammed computer system model
(second level)

Fig. 8.1:   Hierarchical Models of a Multiprogrammed
Computer System

48

The virtual interfault time depends on the program behavior. As studied by Chamberlin, Fuller and Lin [30], the mean virtual interfault time, q, can be expressed as

$$q = \frac{2S}{1 + (\frac{dn}{M})^2}$$

where

M: main memory size

d: the number of pages that provides the process with half of its largest possible life time

S: the expected virtual interfault time when the process is allocated a memory space d

n: degree of miltiprogramming

We will consider two types of program behavior as considered in [30], [31].

Type 1:   S = 25 ms
           d = 50 pages

Type 2:   S = 20 ms
           d = 60 pages

Clearly, Type 1 programs lead to better performance. We will let N denote the total number of processes being activated, i.e. the total number of processes in the CPU, page and file I/O devices. Furthermore, we assume that processes in the file I/O queue will be swapped out from memory. Note n denotes the total number of processes contending for memory, i.e. the total number of processes in the CPU and paging device which is usually referred to as the degree of multiprogramming.

We begin with the first level model in Fig. 8.1a. Let us assume that the total memory size is 128 pages and the time slice, t_s, for each job is 50 msec. The mean service time, t_{pg}, of the paging device is assumed to be 5 msec. The mean CPU overheads for a page fault and a process switching, referred to as 0_p

49

and O_s, are assumed to be 0.2 and 0.3 msec., respectively. The mean time
between service completions at CPU is approximately

$$t_{CPU} = \frac{t_s q}{t_s + q} + \psi \, O_s + (1-\psi)O_p$$

where

$$\psi = \frac{q}{t_s + q}$$

The numerical approximation technique in section 6 is employed in the following
study. In Fig. 8.2a and Fig. 8.3a, we plot the interfault time distribution
functions and density functions for type 1 programs when n=2, 4, 6 and 8,
respectively. The time is normalized with respect to mean virtual interfault time.
In Fig. 8.2b and 8.3b we plot the same curves for type 2 programs. As we can
see when the degree of multiprogramming increases, the changes in the shape of
distribution functions or density functions follow the same pattern for both
types of programs. Nevertheless, for type 2 programs, the performance deteriorates
further as n becomes large. In Table 8.1 and 8.2, we tabulate the CPU utilization
under various values of n for both types of programs. The CPU utilization for
type 2 programs decrease sharply as n goes to 8. The CPU utilizations given
in Table 8.1 and 8.2 will be used in the next level of the multiprogrammed
computer system model to determine the effective CPU execution time between file
I/O requests or process completions. The effective CPU execution time is defined
to be the CPU execution time devided by the true CPU utilization referred to as
U_{CPU}.

FIG 8.2.A INTERFAULT TIME DISTRIBUTION
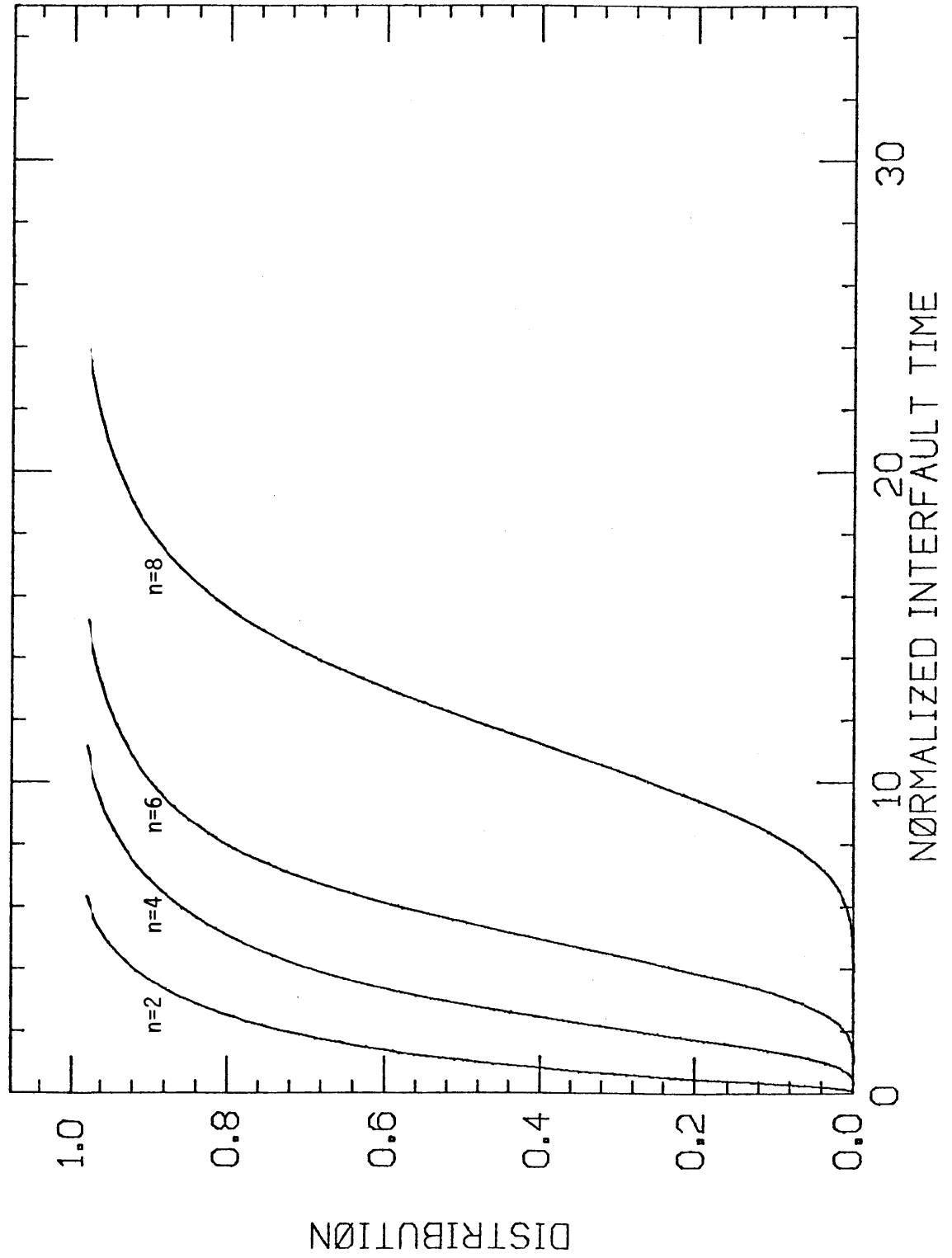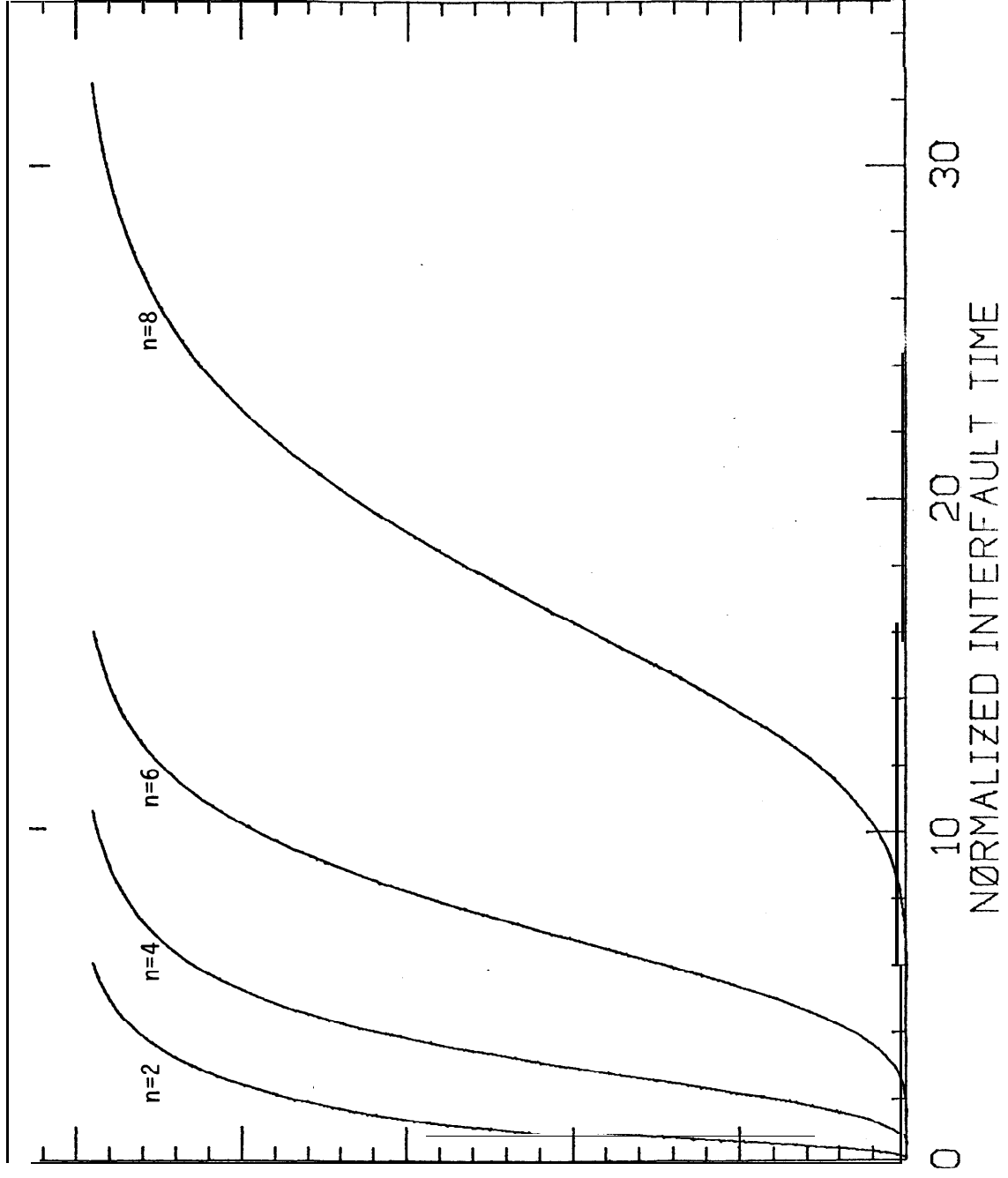
n=8

n=6

n=4

n=2

NORMALIZED INTERFAULT TIME

DISTRIBUTION

51

8.2.B INTERFAULT TIME DISTRIBUTION

n=8

n=6

n=4

n=2

NORMALIZED INTERFAULT TIME

0   10   20   30

FIG 8.3.A INTERFAULT TIME DENSITY

DENSITY

NORMALIZED INTERFAULT TIME

n=2
n=4
n=6
n=8

53

FIG 8.3.B INTERFAULT TIME DENSITY

n=2

n=4

n=6

n=8

DENSITY

NORMALIZED INTERFAULT TIME

0   10   20   30

0.0   0.2   0.4   0.6
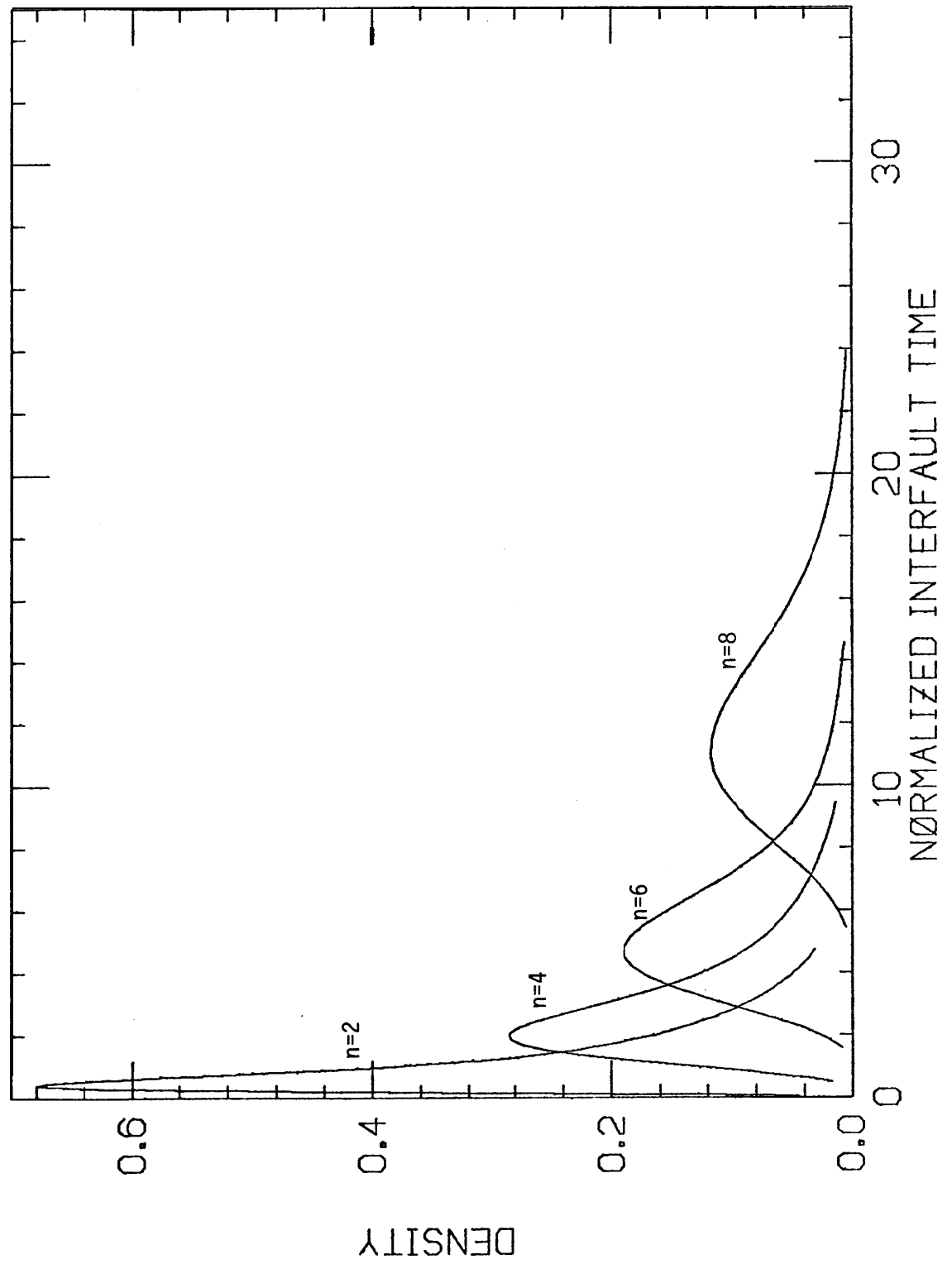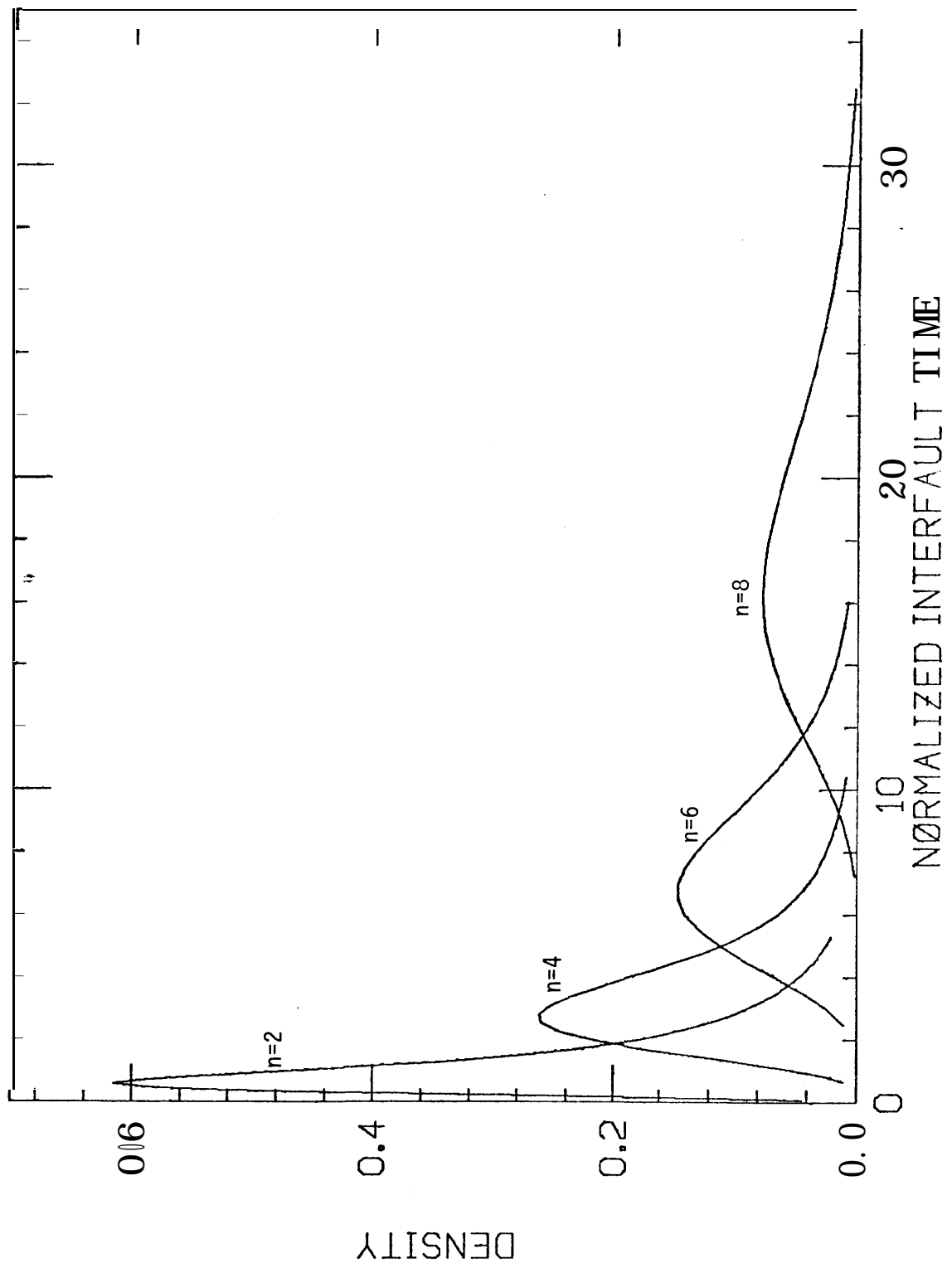
degree of multiprogramming	CPU utilization (including overhead)	CPU utilization (true), U_{CPU}
1	0.898	0.882
2	0.979	0.956
3	0.991	0.961
4	0.992	0.952
5	0.989	0.937
6	0.980	0.914
7	0.956	0.876
8	0.904	0.811

Table 8.1:   CPU Utilization Under Type 1 Programs

degree of multiprogramming	CPU utilization (including overhead)	CPU utilization (true), U_{CPU}
1	0.870	0.851
2	0.959	0.931
3	0.970	0.929
4	0.961	0.903
5	0.929	0.854
6	0.859	0.768
7	0.749	0.649
8	0.629	0.526

Table 8.2:   CPU Utilization Under Type 2 Programs

Now let us consider the second level model in Fig. 8.1b. We further assume that t_e, the mean CPU execution time of a process, is 100 ms and t_f, the mean execution time between file I/O requests of a process, is 50 ms. The service time of the file I/O device is assumed to be 50 ms. The CPU overheads for a file I/O fault and a process termination and initiation, referred to as O_f and O_c, are assumed to be 4 ms and 0.3 ms, respectively. The mean service completion time of the CPU-PGU subsystem is approximately

$$t_{CPU-PGU} = \frac{t_f t_c}{(t_f + t_c) U_{CPU}} + \phi O_c + (1 - \phi) O_f$$

where

$$\phi = \frac{t_f}{t_c + t_f}$$

In Fig. 8.4a and 8.5a, we plot the response time distribution functions and density functions for type 1 programs when N = 2, 4, 6 and 8. The response time is normalized with respect to the mean CPU execution time of a process. In Fig. 8.4b and 8.5b, we plot the same curves for type 2 programs. As we can see as N increases, the changes in the shape of response time distributions or densities follow the same pattern for both types of programs, respectively. In Table 8.3, we tabulate the mean response times for both cases. As we can see when N is less than 6, the response time distributions or densities under both types of programs are very close to each other. When N further increases, the performance under type 2 programs deteriorates faster than than under type 1 programs since the paging device is not fast enough to support paging requests.

Let us change the mean execution time between file I/O requests to 30 msec. and plot the same kind of curves in Fig. 8.4c, 8.4d and Fig. 8.5c, 8.5d as before. Now the file I/O request rate increases and the file I/O device becomes the bottleneck of the system. Comparing Fig. 8.5c with 8.5a or Fig. 8.5d with 8.5b,

we can see the forms of density functions change sharply from before. In Table 8.4, we tabulate the mean response times for both types of programs. Since processes contending for the file I/O device do not content **for memory resource**, the performance difference between type 1 and type 2 programs become smaller compared with the previous case.

To investigate the balancing of a computer system let us keep the degree of multiprogramming to be 8 and the processing rate of the CPU and paging device as before. Type 2 programs are used for illustration. In Fig. 8.6a and 8.6b, we plot the interfault time distribution functions and density functions under memory sizes of 128, 192, 256 and 320 pages, respectively. The time is normalized with respect to mean virtual interfault time. In Table 8.5, we tabulate the mean virtual interfault time for each case. **Although the** mean virtual interfault increases steadily according to almost linear rate, the closeness of the distribution curves after memory size exceeds 192 pages indicates that the system attains its balance after the memory size exceeds 192 pages. Further increasing the memory size only makes the paging device idle most of the time. Notice the crossing of the distribution functions is due to the time slice control which forces the long interfault time to be interrupted and increases the tail of the interfault time distribution. This phenomenon becomes more apparent as the memory size or the mean virtual interfault time further increases. Apparently, when the memory size is 128 pages, the system is not balanced and the paging device is overloaded. Instead of increasing memory size to make the system balance, we can increase the speed of the paging device. In Fig. 8.7a and 8.7b, we plot the interfault time distribution functions and density functions when service rate of the paging device, μ, is 0.2, 0.4, 0.6 and 0.8 processes/msec, respectively. The original service rate of the paging device is 0.2. After we double its service rate, the system gets close to balance. Further increasing the speed of the paging device has marginal effect on the performance.

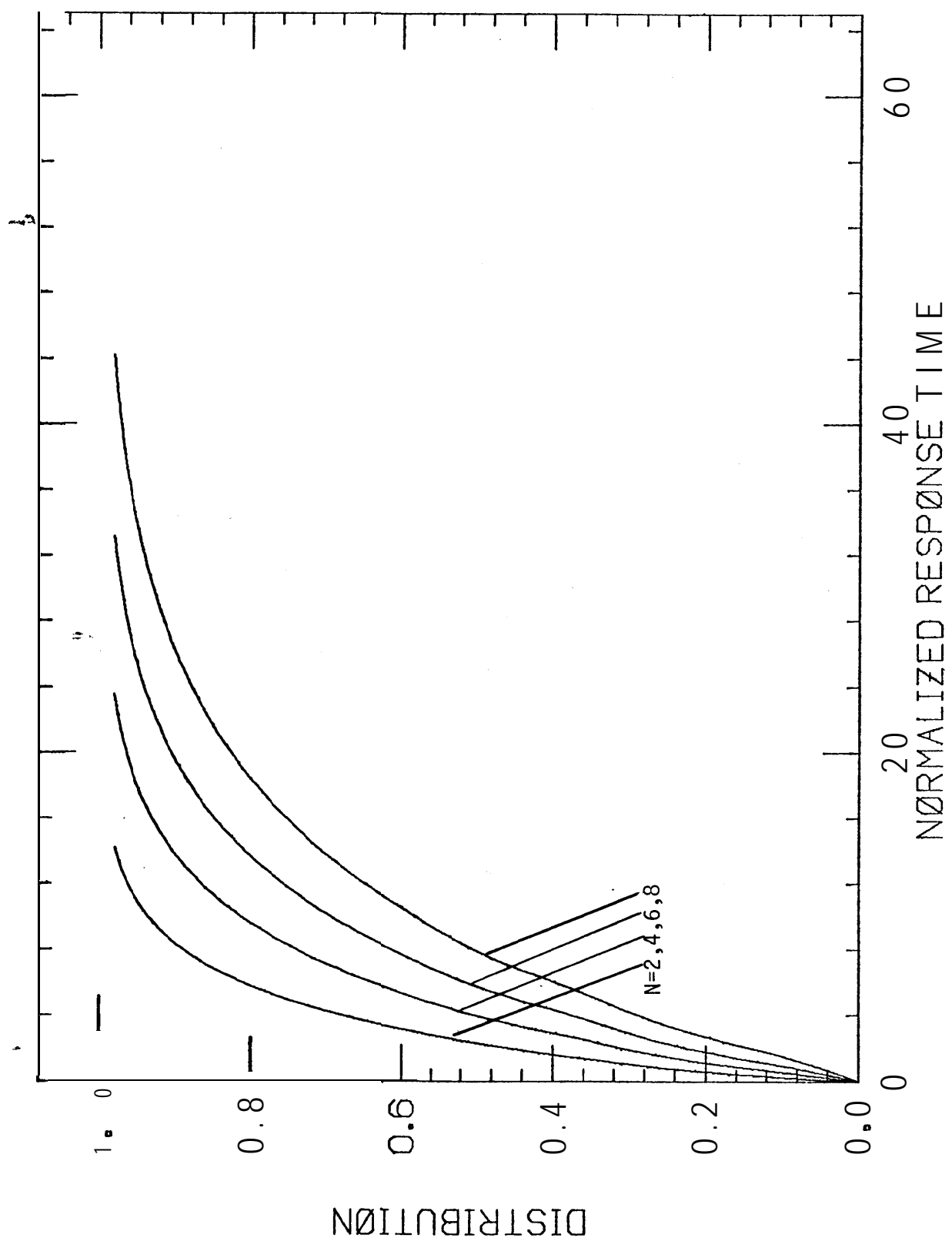FIG 8.4.A RESPØNSE TIME DISTRIBUTIØN
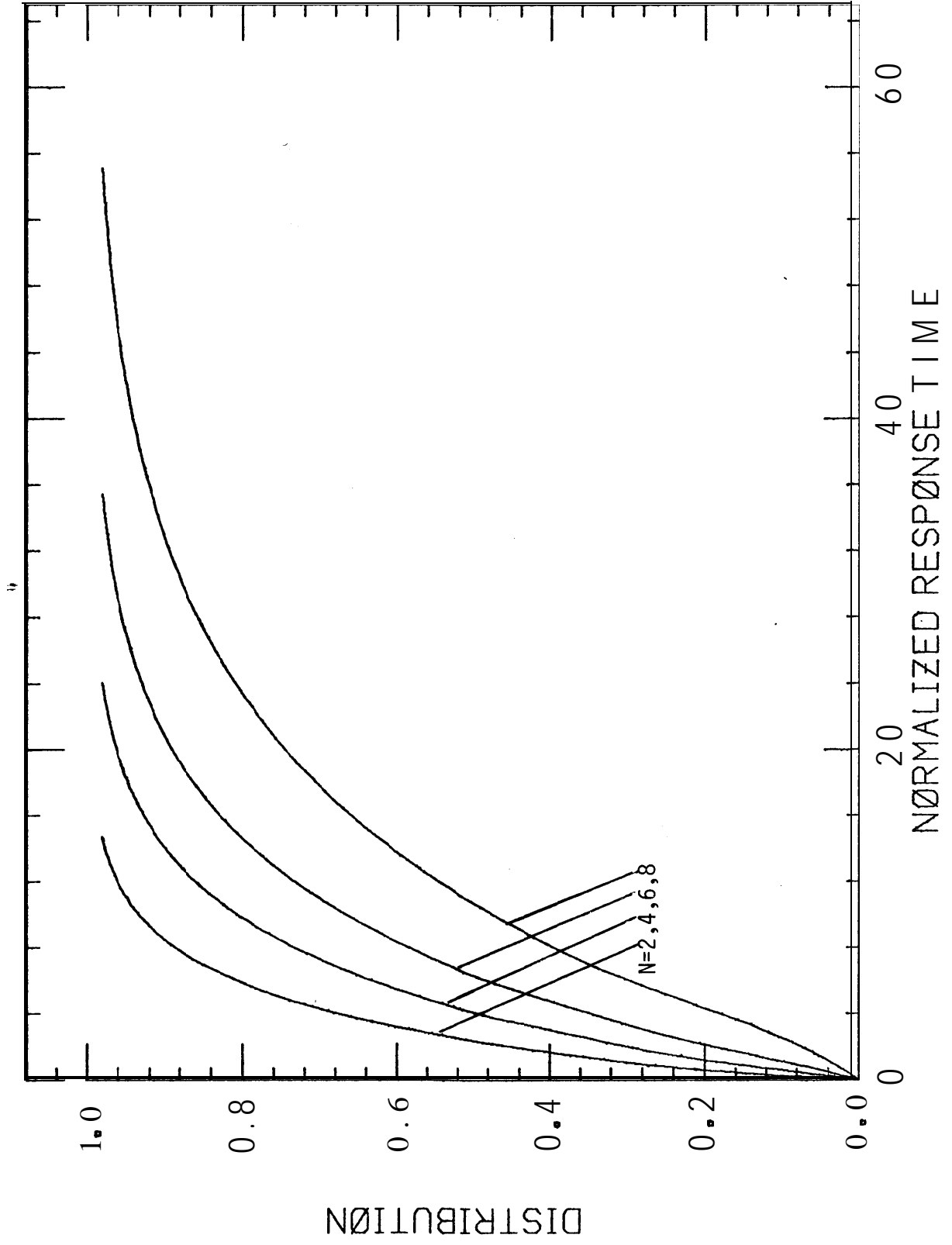
N=2,4,6,8

NØRMALIZED RESPØNSE TIME

DISTRIBUTIØN

58

FIG 8.4.B RESPØNSE TIME DISTRIBUTIØN

DISTRIBUTIØN

NØRMALIZED RESPØNSE TIME

N=2,4,6,8

FIG 8.4.C. RESPONSE TIME DISTRIBUTION

N=2,4,6,8

NORMALIZED RESPONSE TIME

DISTRIBUTION

FIG 8.4.D RESPONSE TIME DISTRIBUTION

N=2,4,6,8

NORMALIZED RESPONSE TIME
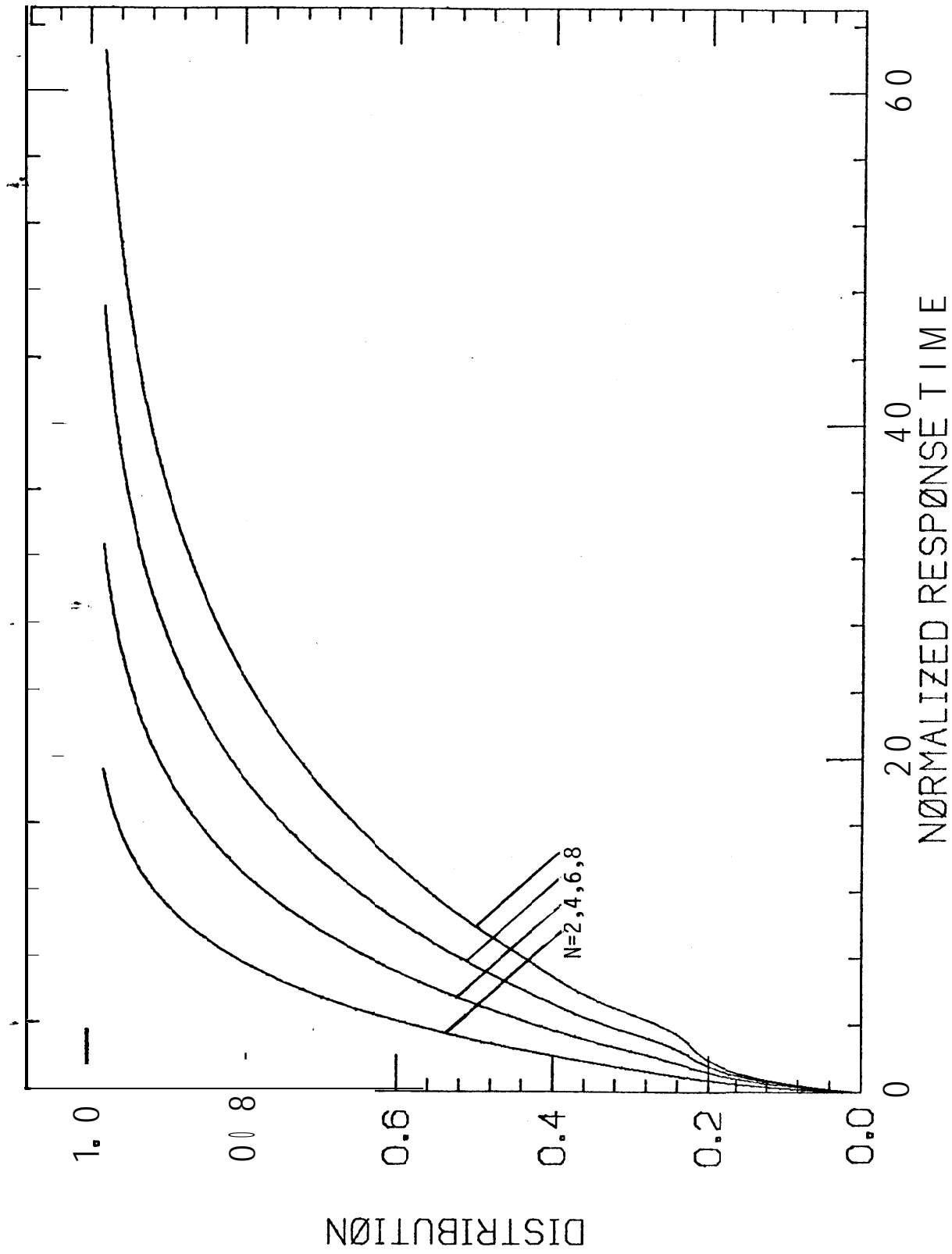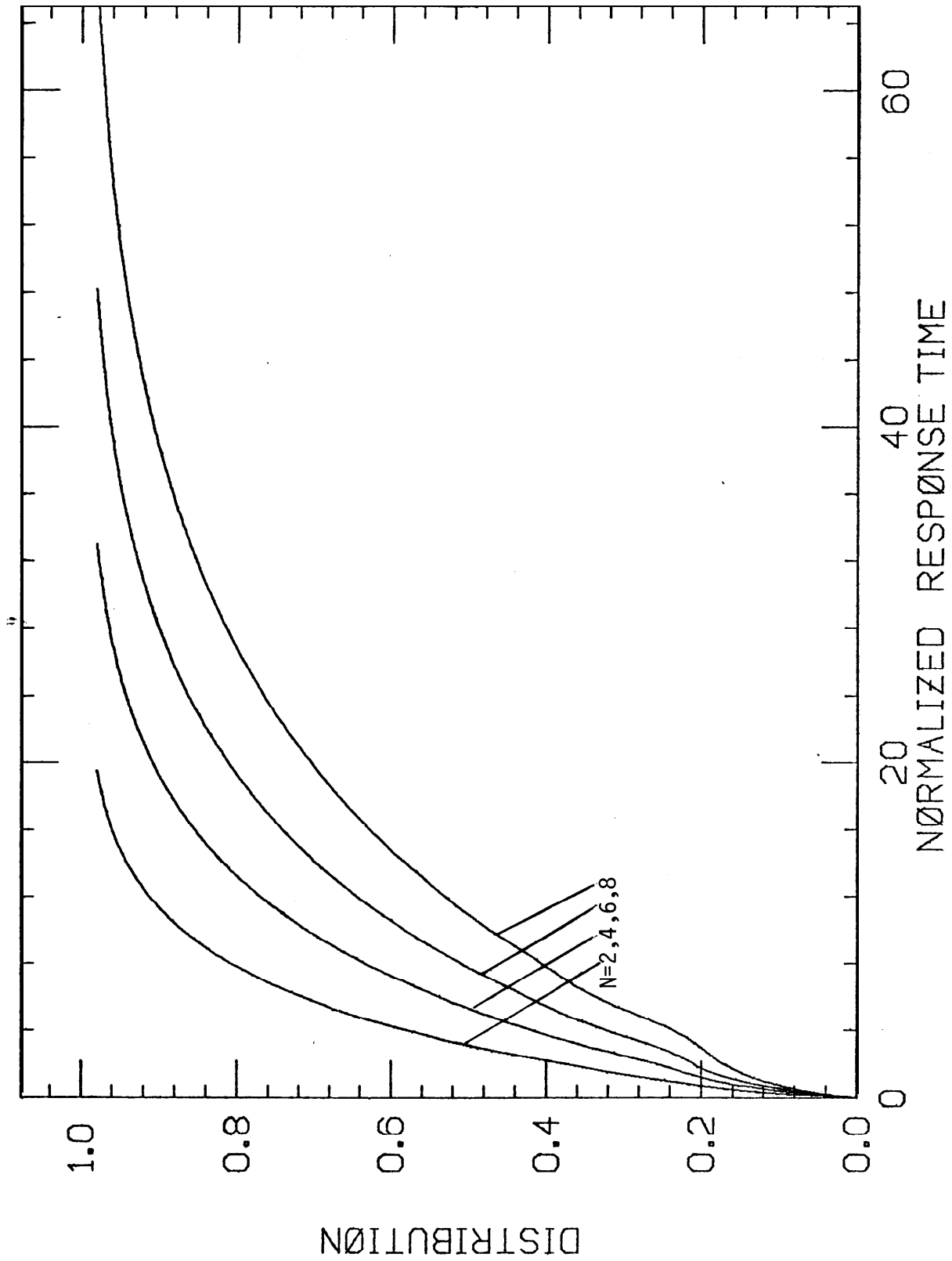
DISTRIBUTION

FIG 8.5.A RESPØNSE TIME DENSITY

N=2

N=4

N=6

N=8

DENSITY

NØRMALIZED RESPØNSE TIME

62
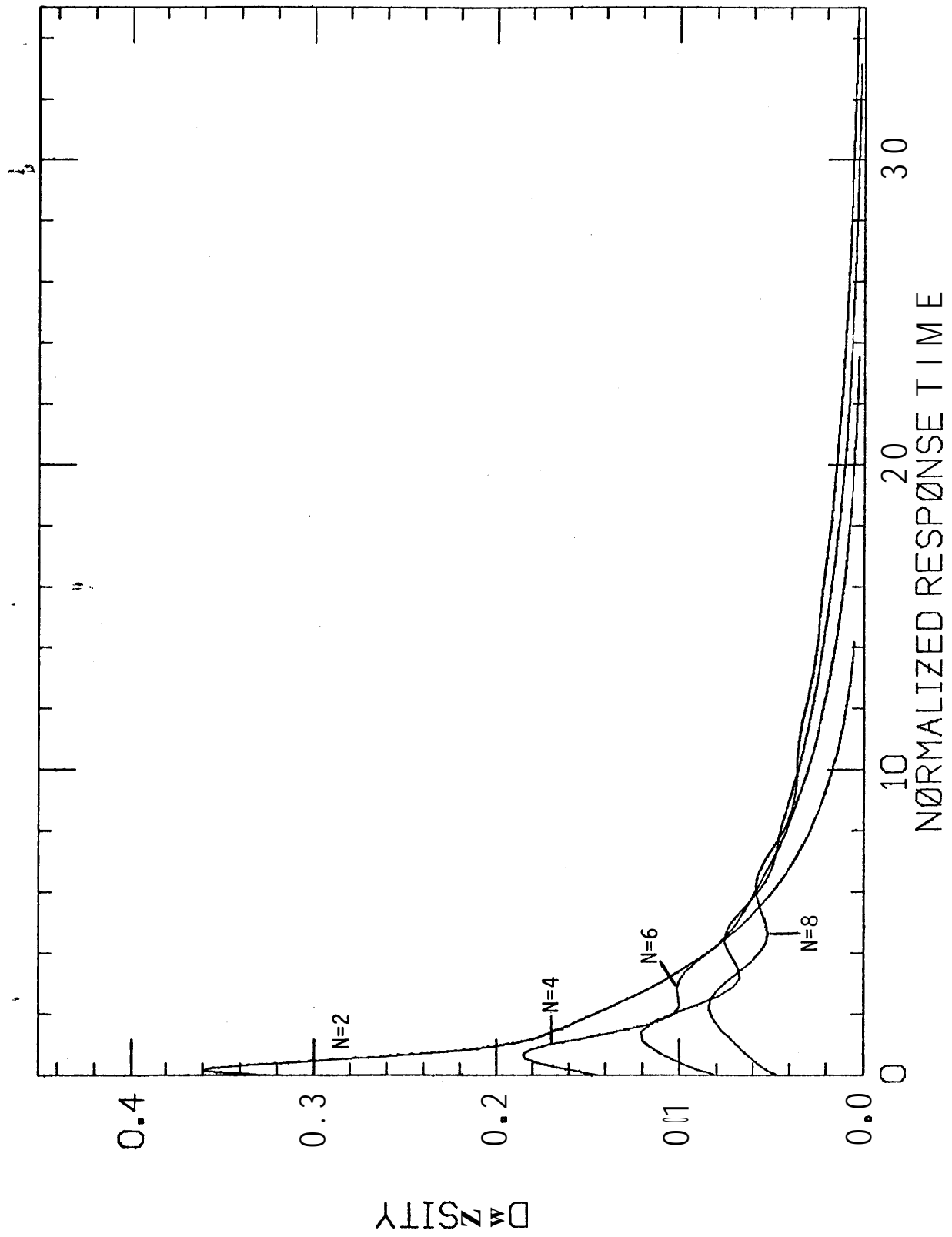
FIG 8.5.B RESPØNSE TIME DENSITY
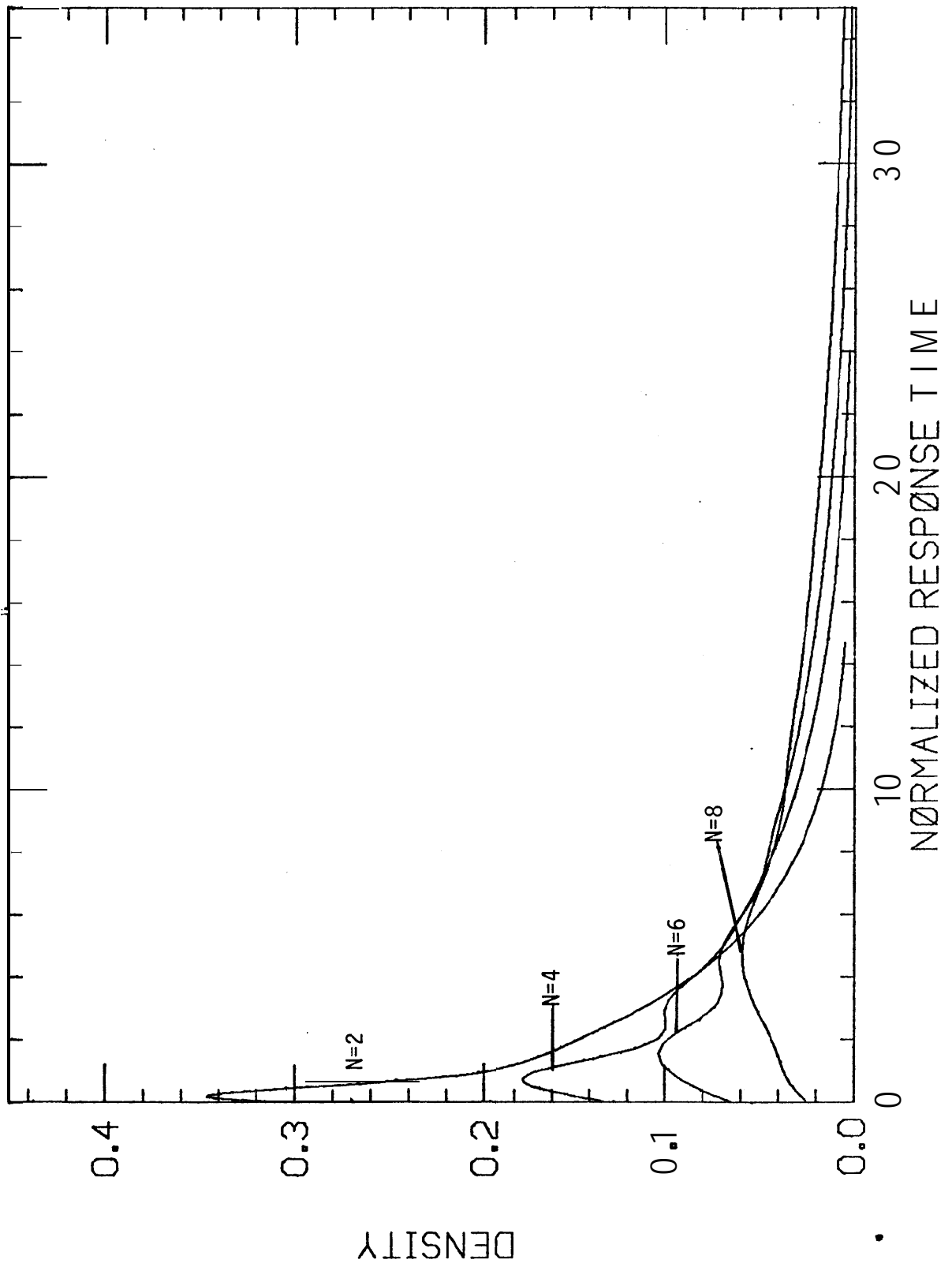
FIG 8.5.C RESPONSE TIME DENSITY

N=2
N=4
N=6
N=8

NORMALIZED RESPONSE TIME
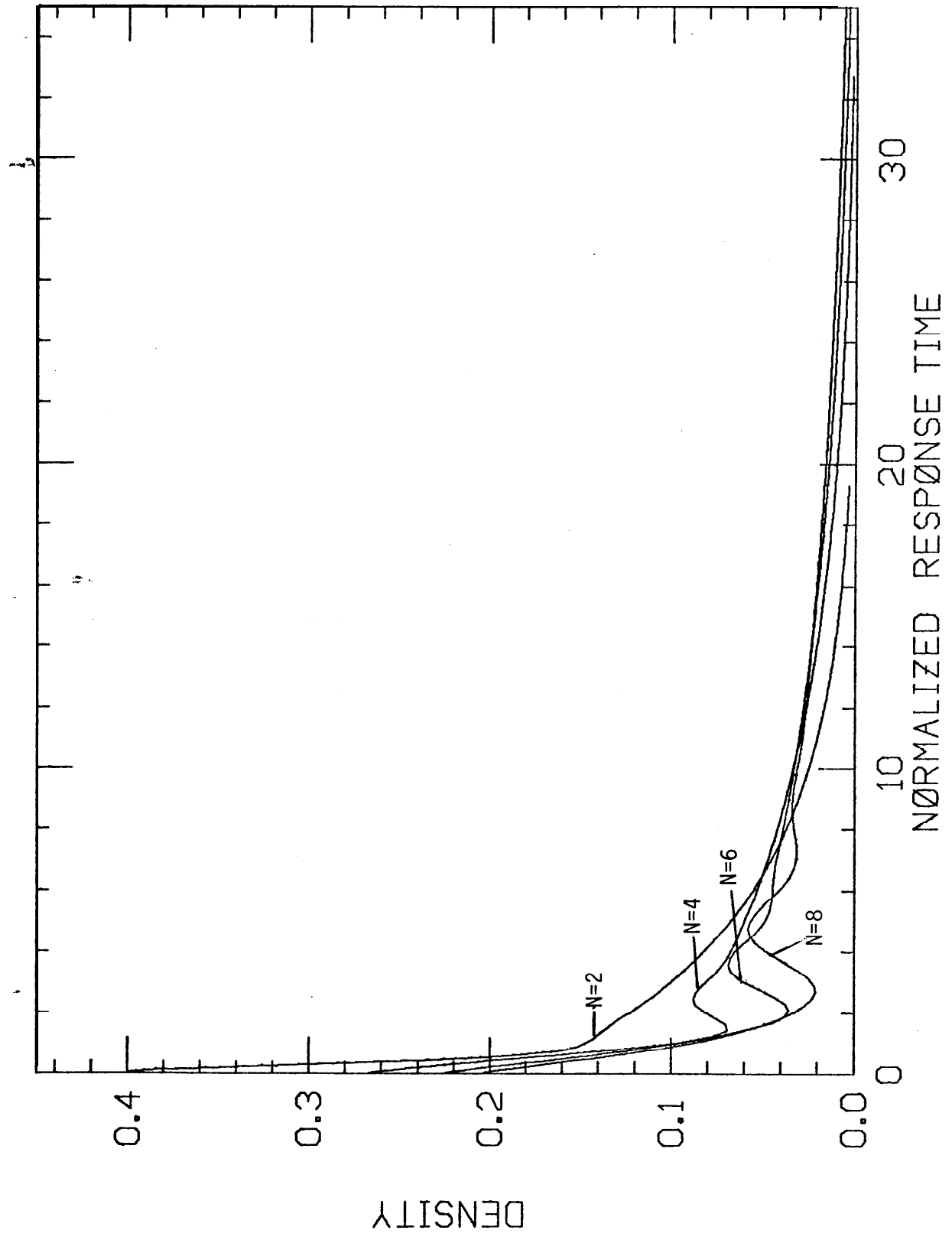
DENSITY

FIG 8.5.D RESPONSE TIME DENSITY

FIG 8.6.A INTERFAULT TIME DISTRIBUTION

DISTRIBUTION

NORMALIZED INTERFAULT TIME

M=192,128

M=320,256

66
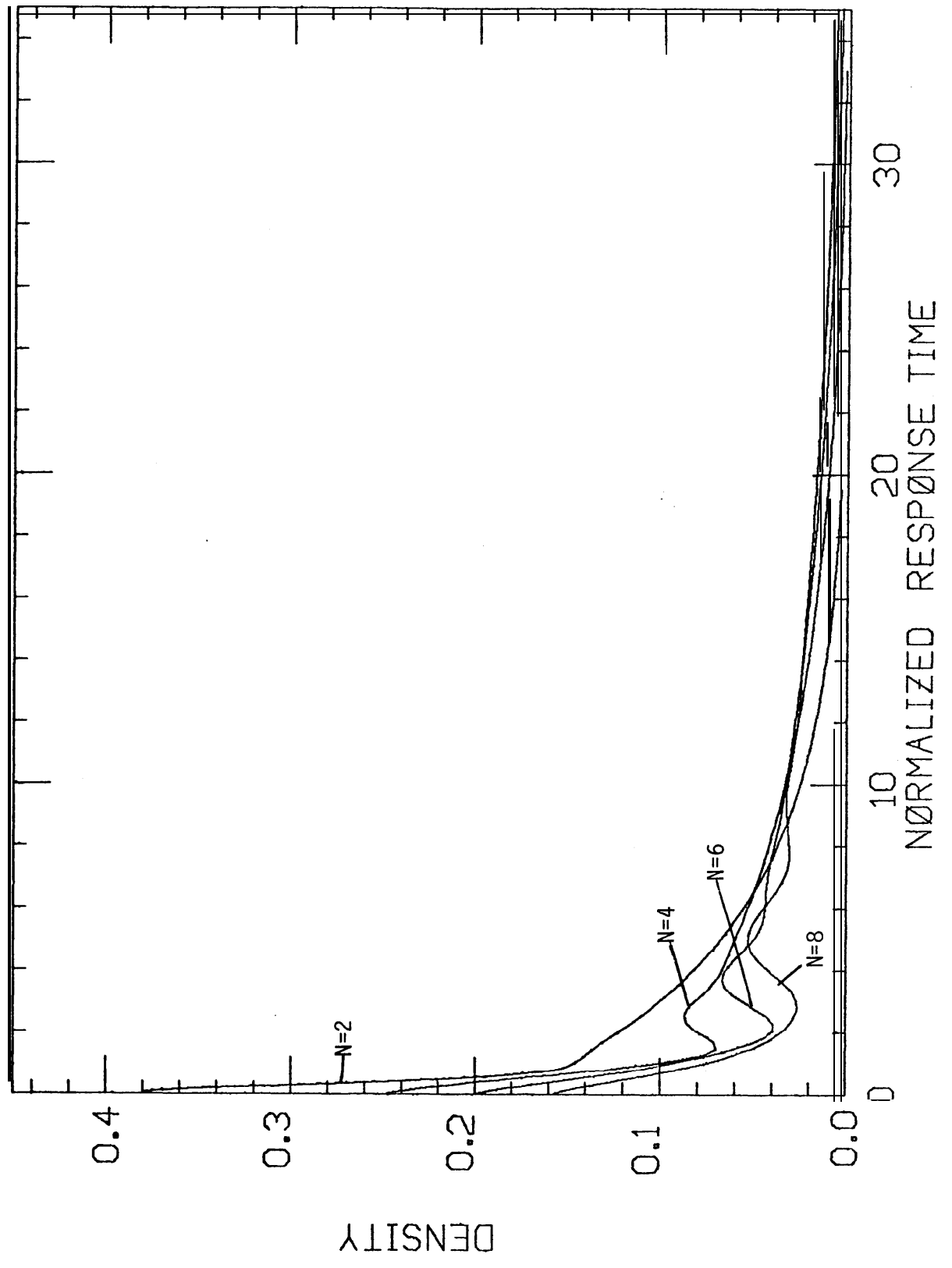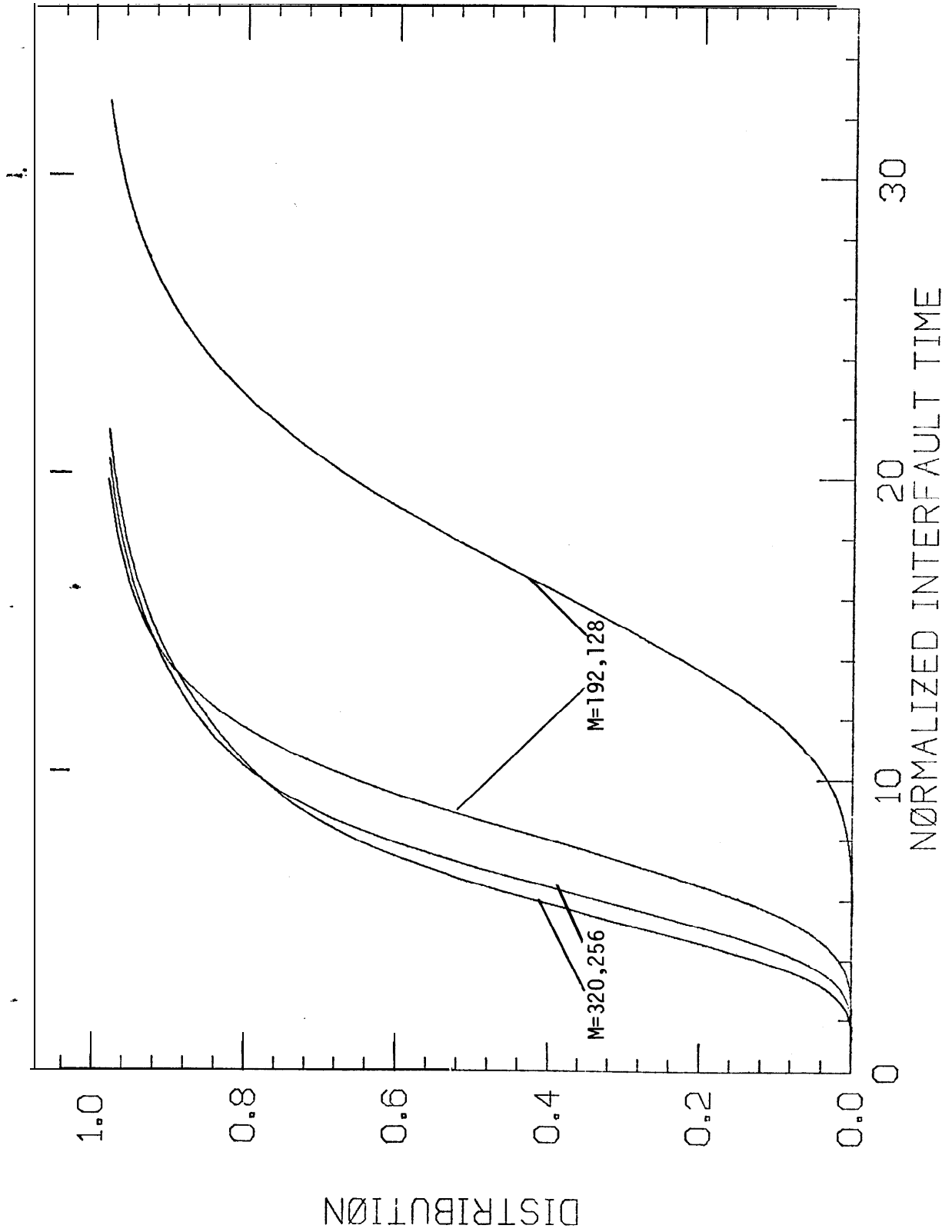
FIG 8.6.B INTERFAULT TIME DENSITY

FIG 8.7.A INTERFAULT TIME DISTRIBUTION

μ = 0.4, 0.2

μ = 0.8, 0.6

NORMALIZED INTERFAULT TIME
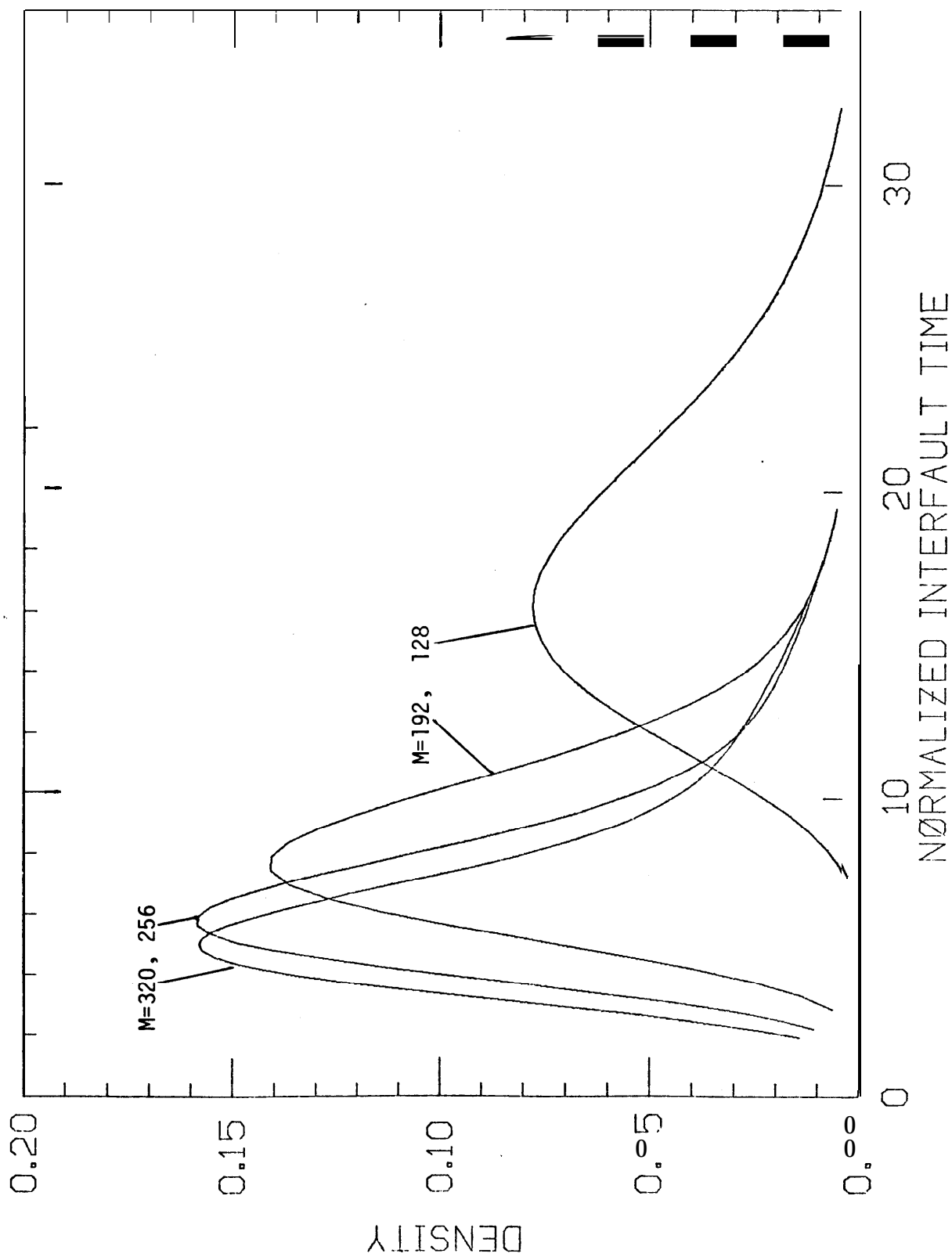
DISTRIBUTION
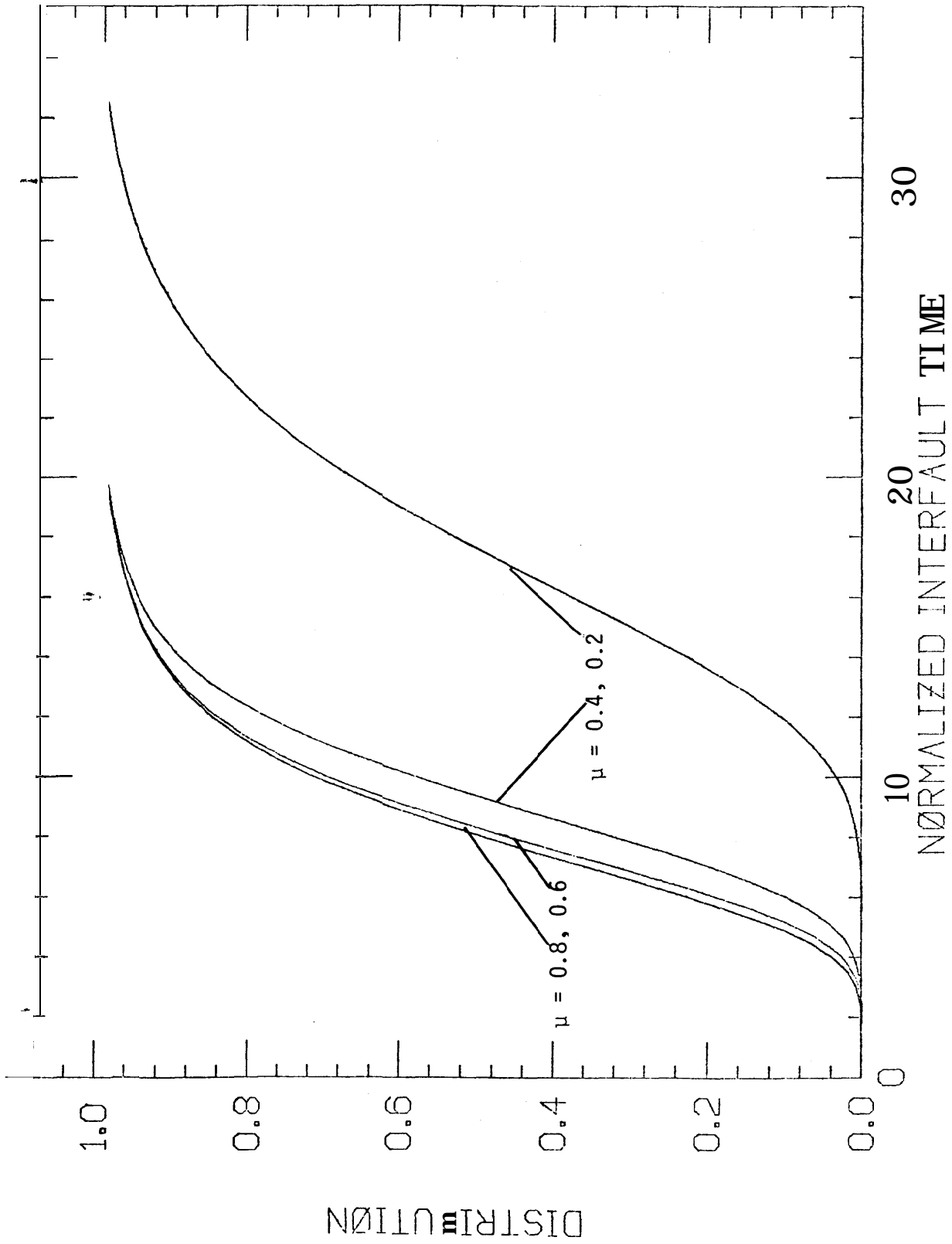
FIG 8.7.B INTERFAULT TIME DENSITY

69

N	type 1 program	type 2 program
2	326	332
4	540	553
6	766	823
8	1028	1318

Table 8.3:   **Mean Response Time When the Mean Time Between File I/O Requests is 50** msec
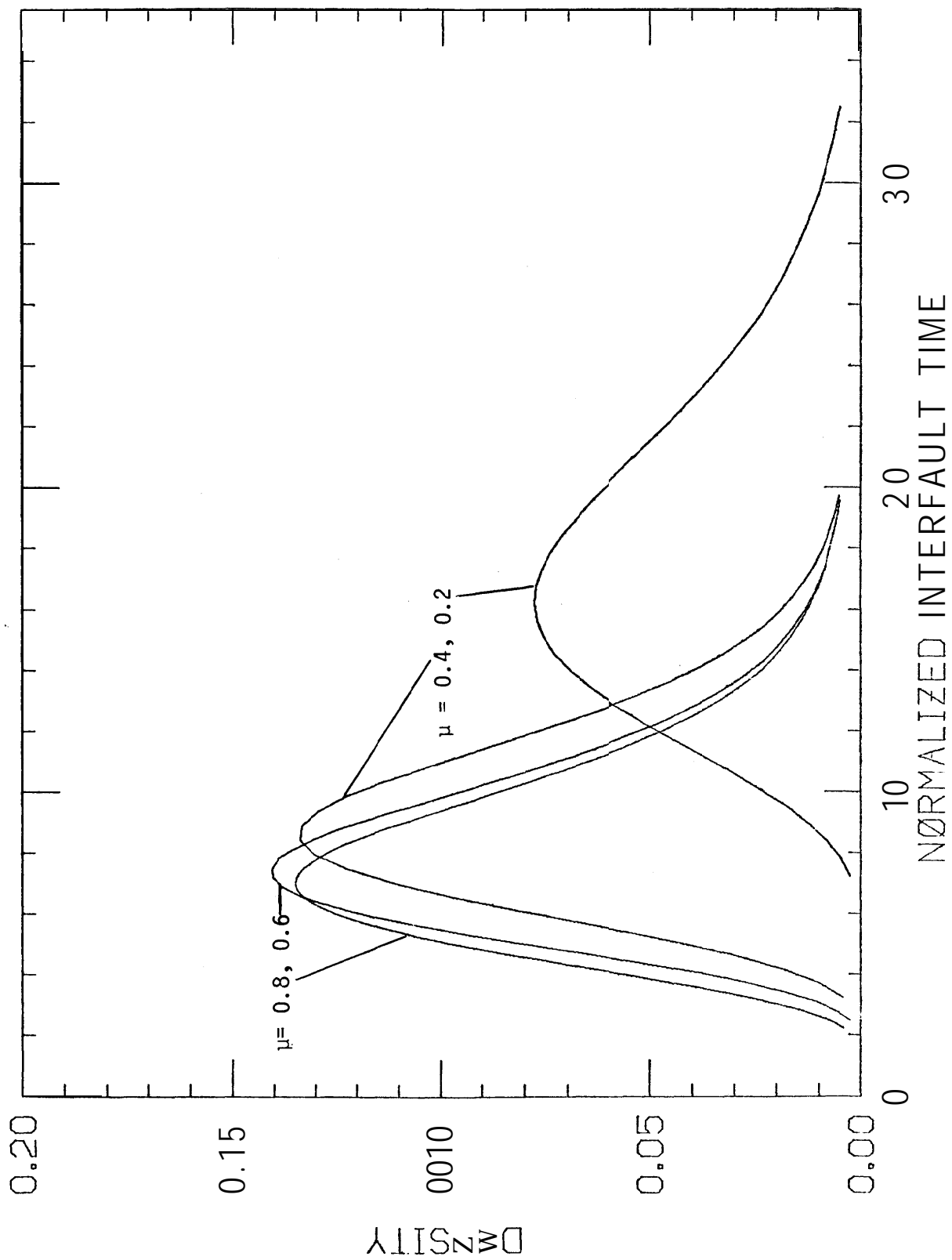
N	type 1 program	type 2 program
2	438	443
4	737	745
6	1048	1070
8	1373	1445

Table 8.4:   **Mean Response Time When the Mean Time Between File I/O Requests is 30** msec

memory size	mean virtual interfault time
128	2.66
192	5.52
256	8.86
320	12.31

Table 8.5:   **Mean Virtual Interfault Time Under Various Memory Sizes.**

9. CONCLUSION

Queueing network models have been used extensively in modeling computer systems and computer communication networks. The situations where we can get analytic solutions for stationary state probabilitieshave been studies in the past few years. Although their average values can be obtained through Little's formula, the response time distributions or the more general passage time distributions have not yet been solved. The distribution function can provide us with a lot of useful information, such as variance or other higher moments, percentile, quantile, etc. By transforming passage times into hitting times of appropriate Markov systems, we derive an analytical solution for passage time distributions for the same class of closed queueing networks specified in section 3, where the analytic solution on stationary state probabilities is available. Avoiding matrix inversion in transform domain required by the exact analysis, efficient numerical approximation replacing convolutions by recurrence relations is also proposed using the concept of discretization of the distribution functions. Then we consider passage time distributions for the class of open queueing networks specified in section 3. The result for closed queueing networks is extended to obtain approximate passage time distributions for open queueing networks. Finally, we employ the techniques derived in this paper to study the interfault time distributions and response time distributions of multiprogrammed computer systems. The effects of program behavior, degree of multiprogramming, size of main memory, service time of paging devices, and rate of file I/O requests on the shape of distribution functions and density functions have been examined.

71

REFERENCES

1. Baskett, F., "The Dependence of Computer System Queues Upon Processing Time Distribution and Central Processor Scheduling", Proc. of the ACM SIGOPS Third Symposium on Operating System Principles, 1971, pp. 109-113.

2. Baskett, F., Chandy, K.M, Muntz, R.R. and Palacios, F.G., "Open, Closed and Mixed Networks of Queues With Different Classes of Customers", J. ACM 22, April 1975, pp. 248-260.

3. Baskett, F., and Muntz, R.R., "Queueing Network Models With Different Classes of Customers", Proc. Sixth Annual IEEE Computer Society Int'l Conf., Sept. 1972, pp. 205-209.

4. Baskett, F. and Palacios, F.G., "Processor Sharing in a Central Server Queueing Model of Multiprogramming With Applications", Proc. of the Sixth Annual Princeton Conference on Information Sciences and Systems, March 1972.

5. Buzen, J., "Queueing Network Models of Multiprogramming", Ph.D. Thesis, Harvard University, 1971.

6. Buzen, J., "Computational Algorithms for Closed Queueing Networks With Exponential Servers", CACM 16, Sept. 1973, pp. 527-531.

7. Browne, J.C., et. al., "Hierarchical Techniques for the Development of Realistic Models of Complex Computer Systems", Proc. IEEE, 63.6, June 1975, pp. 966-975.

8. Chandy, K.M, "The Analysis and Solutions for General Queueing Networks", Proc. of the Sixth Annual Princeton Conf. on Information Sciences and Systems, March 1972, pp. 224-228.

9. Chandy, K.M, Keller, T.W and Browne, J.C., "Design Automation and Queueing Networks: An Interactive System for the Evaluation of Computer Queueing Models", Proc. of the Ninth Annual Design Automation Workshop, June 1972.

10. Cox, D.R., "A Use of Complex Probabilities in Theory of Stochastic Processes", Proc. Camb. Phil, Soc. 51 (1955), pp. 313-319.

11. Feller, W, "An Introduction to Probability Theory and Its Application", Vol.1, 3rd ed., Wiley, 1968.

12. Ferdinand, A.E., "An Analysis of the Machine Interference Model", IBM Systems Journal, 10. 2, 1971, pp. 129-142.

13. Gordon, W.J. and Newell, G.F., "Closed Queueing Systems With Exponential Servers", O.R., 15, 1967, pp. 254-265.

14. Iglehart, **D. L. and Shedler, G. S., "Estimation Via Regenerative Simulation of Response Times in Networks of Queues"**, IBM **Res. Report.** RJ1740, 1976.

15. Jackson, J.R., **"Jobshop-like Queueing Systems", Man. Sci. 10, 1, 1973, pp. 131-142.**

16. **Kleinrock, L., "Queueing Systems", Vol. I and** II, **Wiley** Interscience, 1976.

17. **Kobayashi, H. and Konheim, A. G., "Queueing Models for Computer Communications System Analysis",** IBM **Res. Report,** RC5922, 1976.

18. **Kobayashi, H. and Reiser, M, "On Generalization of** Job **Routing Behavior in a Queueing Network Model",** IBM **Res.** Report RC5252, 1975.

19. **Kobayashi, H., "System Design and Performance Analysis Using Analytic Models",** IBM **Res. Report** RA75, 1975.

20. **Lam, S. S., "Queueing Networks With Lost and Triggered Arrivals",** IBM **Res. Report RC5540, 1975.**

21. **Moore,** C.G. III, **"Network Models for Large Scale Time Sharing Systems", TR 71-1, Dep. of** I.E., **U. of Michigan at Ann Arbor, 1971.**

22. **Muntz, R. R. and Baskett, F., "Open, Closed and Mixed Networks of Queues With Different Classes of Customers", TR 23, Stanford University, 1972.**

23. **Palacios, F. G., "An Analytic Model of a Multiprogramming System Including a** Job **Mix", TR-4, Dept. of C. S., U. of Texas at Austin, 1972.**

24. **Posner, M and Bernholtz, B., "Closed Finite Queueing Networks With Time Lags and With Several Classes of Units", O. R. 16, 1968, pp. 977-985.**

25. **Reiser, M and Kobayashi, H., "Recursive Algorithm for General Queueing Networks With Exponential Servers",** IBM **Res. Report** RC4254, **1973.**

26. **Reiser, M and Kobayashi, H., "Queueing Networks With Multiple Closed Chains: Theory and Computational Algorithms",** IBM J. **of Res. and Dev. 19. 3, 1975, pp. 283-294.**

27. **Sakata, M, Noguchi, S. and Oizumi,** J., **"Analysis of a Processor Shared Queueing Model for Time-Sharing Systems",** Proc. **of the Second Hawaii International Conf. on System Sciences,** Jan. 1969.

28. **Whittle, P., "Nonlinear Migration Processes",** Proc. **36th Session of the** International **Statistical** Institute, 1969, pp. 642-647.

29. Whittle, P., "Equilibrium Distributions for Open Migration Process",
    J. Appl. Prob. 5, 1968, PP. 567-571.

30. Chamberlin, D., Fuller, S. and Liu, L. "An Analysis of Page Allocation
    Stratagies for Multiprogramming Systems With Virtual Memory",
    IBM Journal R & D, 17, 1973, pp. 404-412.

31. Brandwajn, A., "A Model of a Virtual Memory System", Acta Informatica 6,
    1976, pp. 365-386.