CURRENT STATUS OF
QUEUEING NETWORK THEORY

by

CHI-JIUNN JOU
B.S. (INDUSTRIAL ENGINEERING)
Tunghai University, Taichung, Taiwan
Republic of China, 1979

A MASTER'S REPORT

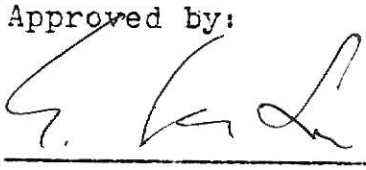
Submitted in partial filfillment of the
requirement for the degree

MASTER OF SCIENCE

Department of Industrial Engineering
Kansas State University
Manhattan, Kansas

1981

Approved by:



Major Professor

SPEC
COLL
LD
2668
.R4
1981
J58
c.2

A11200 067134

ACKNOWLEDGEMENTS

The author wishes to express his sincere appreciation to his major advisor, Dr. E. S. Lee, for his inspiring guidance and constant encouragement during the preparation of this report.

The author also wishes to thank Dr. C. A. Bennett and Dr. Chi-lung Huang for serving on his committee.

The author would like to dedicate this work to his parents for their continuous encouragement and assistance.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

A queueing network is a mathematical model applicable to problems involving a network of nodes with a queue formed in each node. A single queueing system in which there is only one node is just a special case of the queueing network model. Queueing network models have been applied to such important and diverse areas as computer time-sharing and multiprogramming systems, communications networks, air traffic control, production, assembly and inspection operations, maintenance and repair facilities, and medical car delivery systems. The purpose of this report is to survey the results available for queueing networks to date.

1-1 Queueing Network Models

The formation of queues is a common phenomenon which occurs whenever the current demand for a service exceeds the current capacity to provide that service. For example, customers await service at the products wait to be assembled in the assembly line. The basic model for a single queueing system is as follows: "customers" requiring service come from an "input source"; they enter the system and join a queue; at certain times a member of the queue is selected for service by some rule known as the queueing discipline; the required service is then performed for the customer by the server in the service station, after which the customer leaves the queueing system. This model is depicted is Figure 1.1.

THIS BOOK CONTAINS NUMEROUS PAGES WITH DIAGRAMS THAT ARE CROOKED COMPARED TO THE REST OF THE INFORMATION ON THE PAGE.

THIS IS AS RECEIVED FROM CUSTOMER.

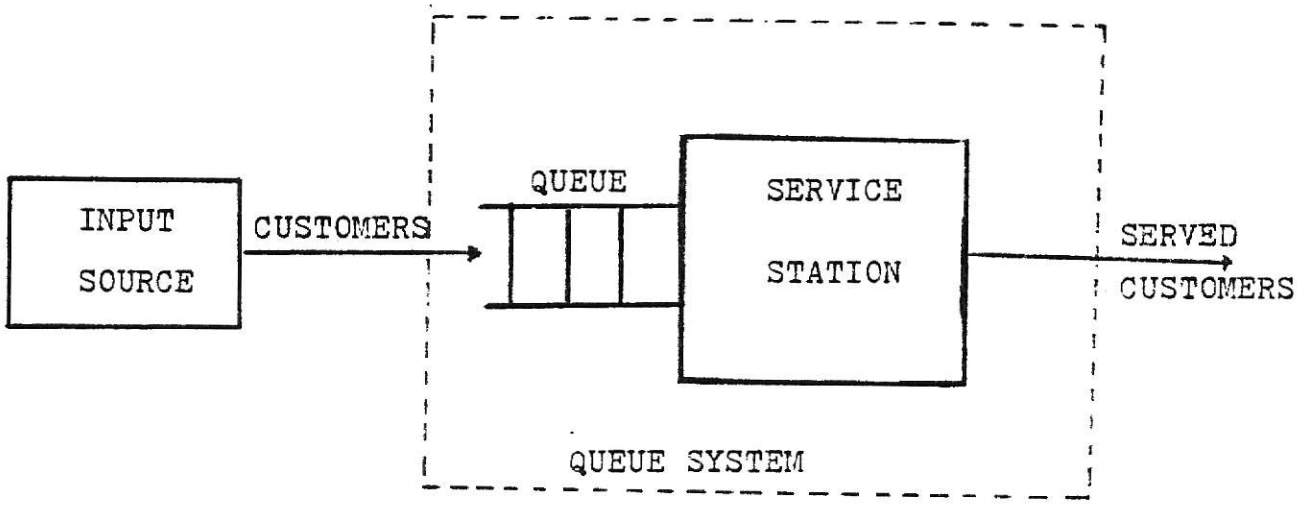


Figure 1.1 A Single Queueing System Model

A queueing network is a network of service stations where customers must receive service at some or all of the stations. Registration for classes in many universities is an example. The students must queue to see an advisor for course planning, then queue for registration, for fee payment, and so on. Another example is job-shop manufacturing. A casting may go to the grinding area, from there to a lathe, then to a milling machine, to a drilling machine, to the inspection area, then perhaps be fed back for more mill work, and so forth. At every atation a queue of work pieces is formed.

A possible queueing network model is shown in Figure 1.2. Customers arriving at service station 1 may come from outside the network with mean arrival rate $P_{0,1}$, or they may transfer from service station 3 with transfer probability $P_{3,1}$. After being serviced in each station, they will transfer to another atation with transfer probability such as $P_{1,4}$, or will leave the system with certain probability, such as $P_{2,0}$.

1-2 Classification of Queueing Network Models

Queueing network model applications abound in recent years. It is therefore useful to classify such models. Kienzle and Sevcik [2] proposed a classification scheme for computer system models based on six characteristics: model structure, customer classification, arrival process, queueing disciplines, and server characteristics. Below, each characteristic will be discussed in detail.

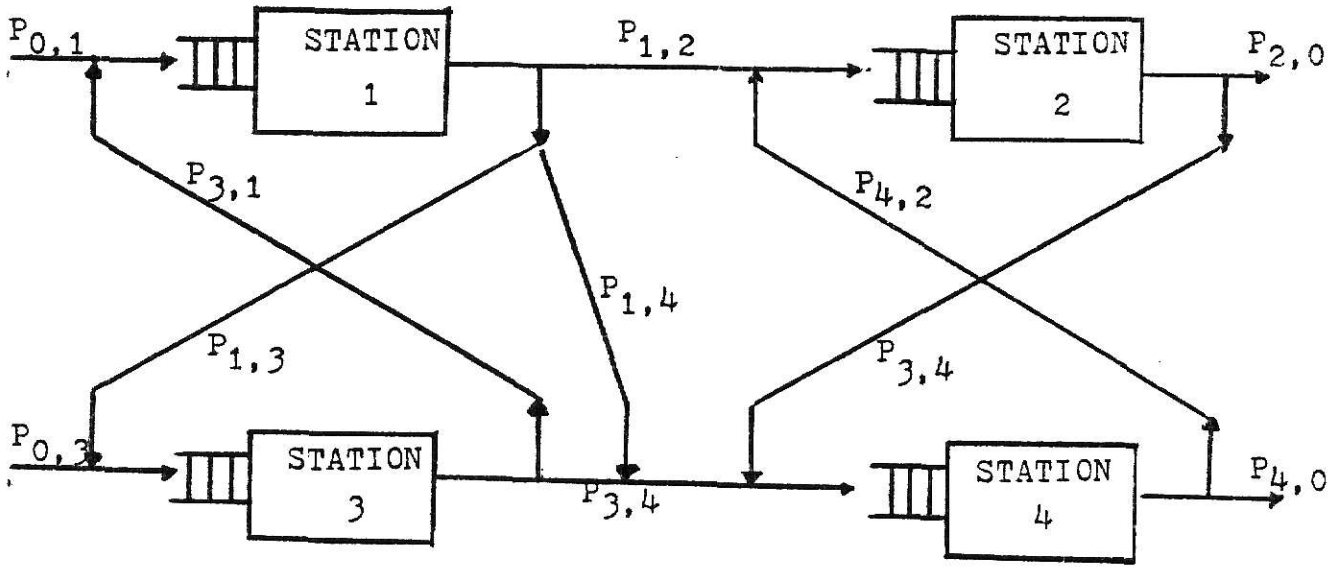


Figure 1.2 A General Queueing Network Model

1. Model structure: the manner in which customers flow among the service stations. A general queueing network structure is the one which has arbitrary routing among service stations; Figure 1.2 is one example. Some particular and useful model structures include;

(a) cyclic queueing model: a fixed number of customers cycle among the service stations. The model is shown in Figure 1.3.

(b) central server model: customers move to other stations from the central service station, but after receiving service they return to the central service station. The model is shown in Figure 1.4.

2. Customer classification: grouping customers classes which have different sets of service rates or routing probabilities. A multiple class model is shown in Figure 1.5. There are three groups:

(a) single class model: all the customers have the same set of service rates and routing probabilities.

(b) multiple class model with no class changes: some customers have different sets of service rates and routing probabilities from other customers. A customer will never change his class while he is in the system.

(c) multiple class model with class changes: customers have different sets of service rates and routing probabilities. A customer may change his present class to other one at a certain time.

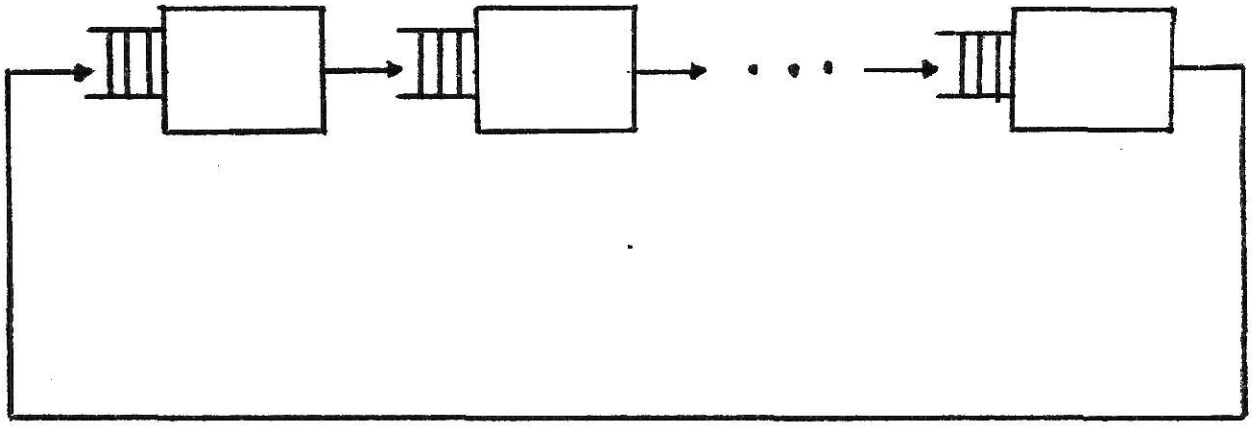


Figure 1.3 A Cyclic Queueing Network Model

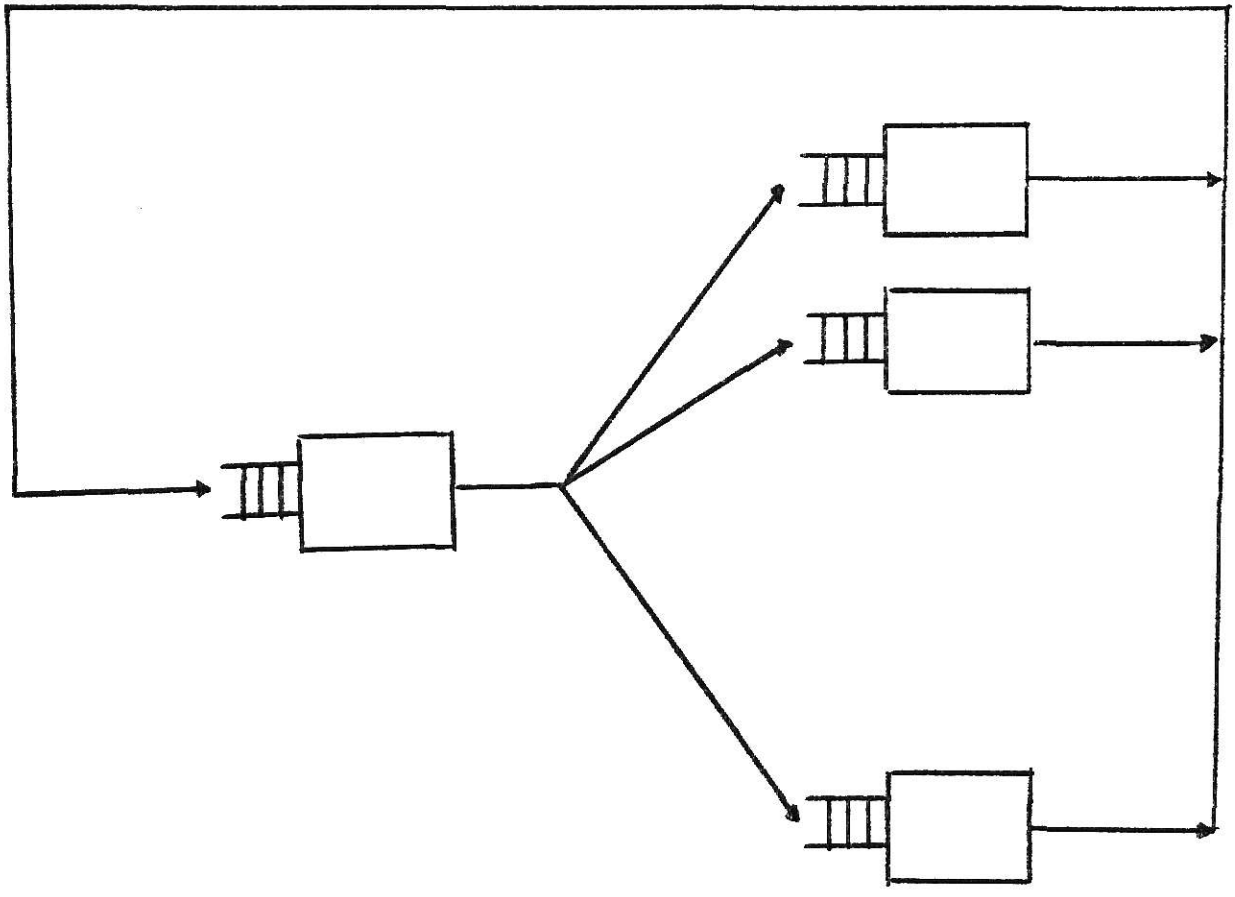
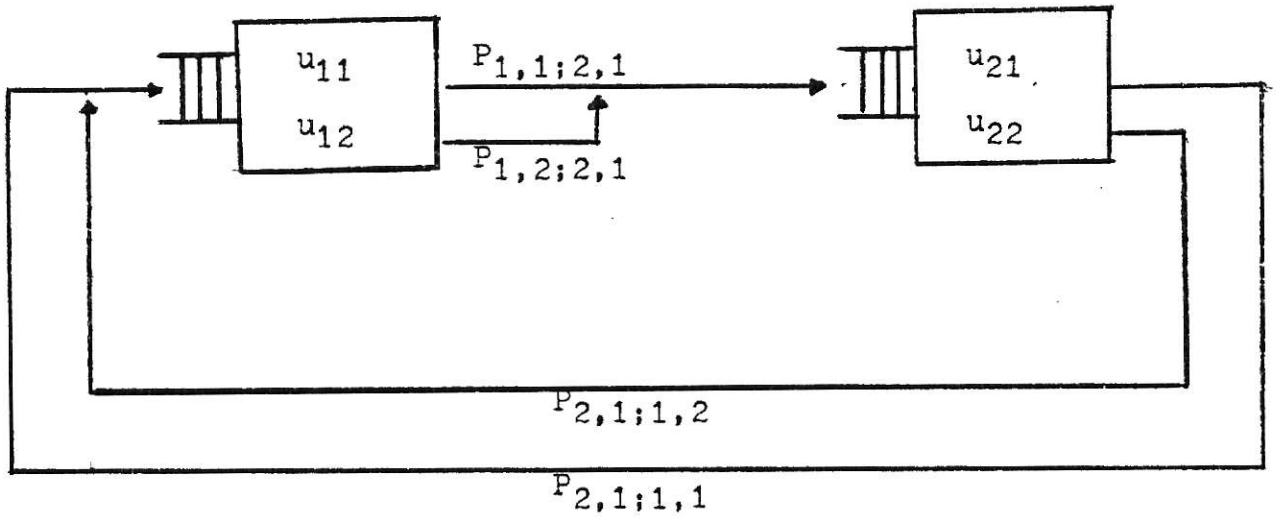


Figure 1.4 A Central Server Model



u_{ij} : service rate at station i for customer class j.

$P_{i,r;j,s}$: routing probability for customer class r at station i transfer to customer class s at station j.

Figure 1.5 A Multiple Class Model

3. Arrival process: the manner in which new customers come into existence. A model can be one of the following three types:

(a) closed network: a fixed number of customers circulate in the network at all times.

(b) open network: all customers are permitted to enter or leave the system. The arrival rate from outside may be constant rate Poisson, load dependent Poisson, or non-Poisson.

(c) mixed network: the system is closed with respect to some classes of customers and open with respect to others. An example is shown in Figure 1.6.

4. Queueing disciplines: the rules for selecting members of the queue for service. Some possible queueing disciplines include:

(a) FCFS: first-come-first-served.

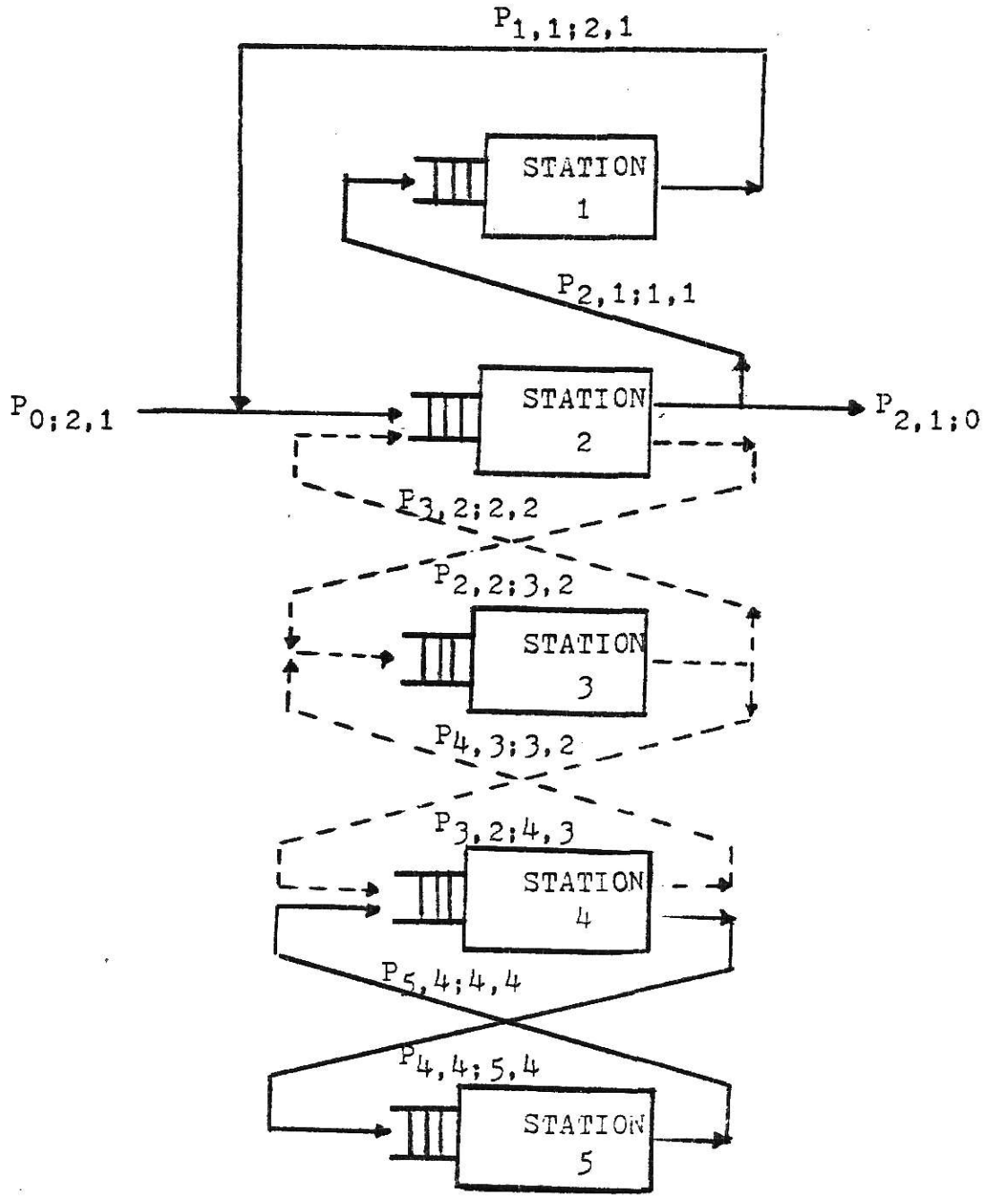
(b) PS: processor sharing, i.e. when there are n customers in the service station each is receiving service at a rate of 1/n sec/sec.

(c) LCFS: preemptive-last-come-first-served, i.e. coming customer will preempt the customer being served now.

(d) IS: infinite servers, i.e. there are enough server so that no queue will happen.

(e) priority disciplines: the priority of service may be based on customer class.

5. Service time description: some possible descriptions are:



Fifure 1.6 A Mixed Network with 1 Open Chain, 2 Closed Chains, 4 Classes, and 5 Stations.

(a) a workload vector: in which the mean total service time required by a customer of a class at each service station is stated.

(b) exponential distribution: the service time for each customer at each service station is assumed to be exponentially distributed.

(c) general distribution: the service time is an arbitrary distribution.

(d) mean and variance: the service time distribution is in terms of its mean and variance.

6. Server charcteristics: describes the reaction of the server to the load. These include:

(a) load independent servers: server has a constant service rate.

(b) load dependent servers: service rate of a server may depend on the number of customers in the same class at the service station, on the total number of customers in a subsystem.

In addition to the above, some other characteristics are needs to classify queueing network models which may not be restricted to computer systems. Three different characteristics are mentioned by Disney [1]. One is routing properties, by which customers may proceed through the network according to their needs, or be routed through the network by a routing scheme imposed on the network exogenously. Another is the waiting space at each service station. The number of spaces before a given server may be finite or infinite and may even be

zero such as in some telephone system models. The third is switching rules, which are used to determine the next path to be taken by a customer. In principle, there are two broad classes of switching rules. Decomposition rules partition the flow processes into substreams; recomposition rules accumulate substreams into a single stream.

1-3. Computer Systems and Queueing Network Models

The growing complexty of computer systems has motivated the development of analytic tools to investigate computer system performance. The queueing network model is an appropriate tool to analyze the modern computer system. Since we may picture jobs flowing from one device to another within the computer system as they place successive demands upon these devices, simultaneous conflicting demands on a device are solved by the formation of a queue in front of the device. Therefore, the application of queueing network models to computer system performance analysis has generated considerable interest. There have been notable advances in both the fundamental theory and practical experience for this area during the past ten years.

The first successful application of a queueing network model to a computer system came in 1965 when Scherr used the finite population model or machine repairman model to analyze the Compatible Time-Sharing System (CTSS) at MIT [205]; the finite population model is shown in Figure 1.7.

In 1971, Moore showed that a closed queueing network model

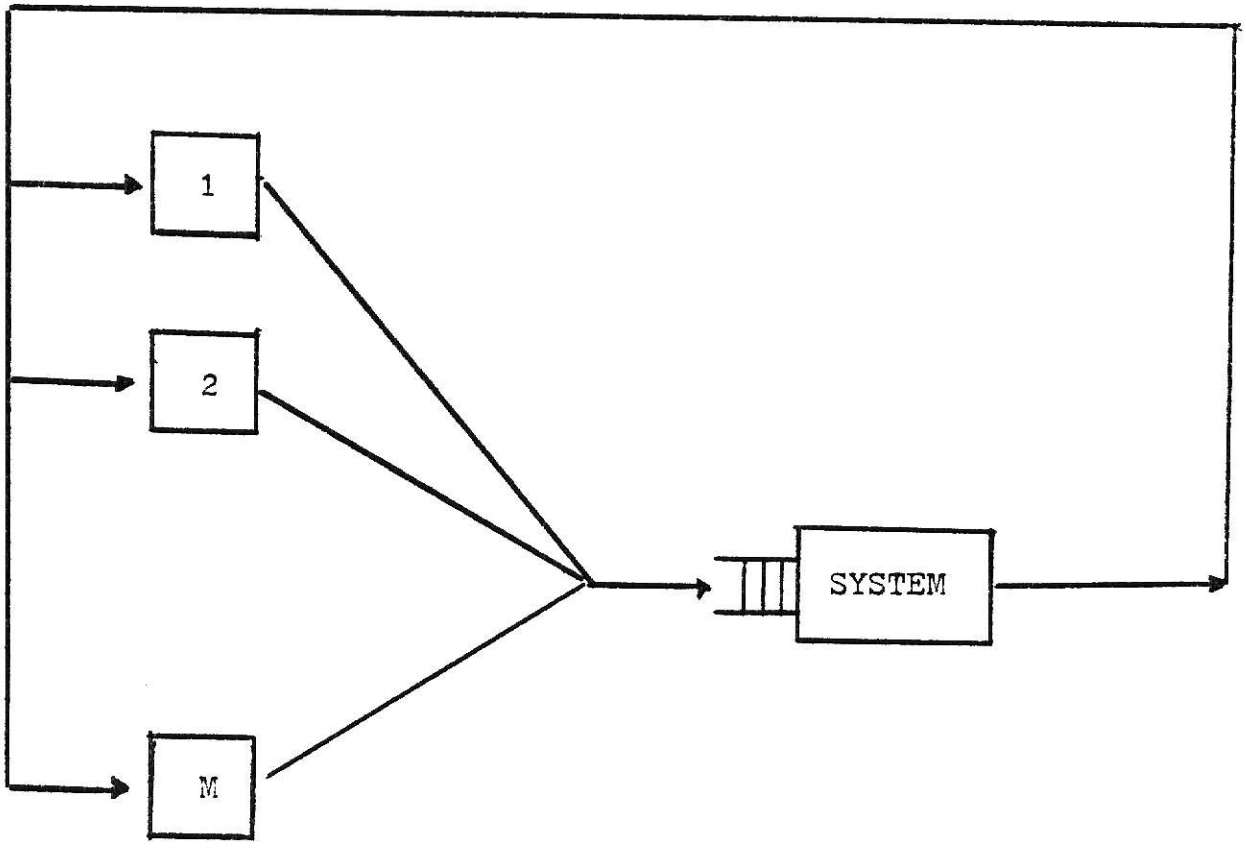


Figure 1.7 A Finite Population Model

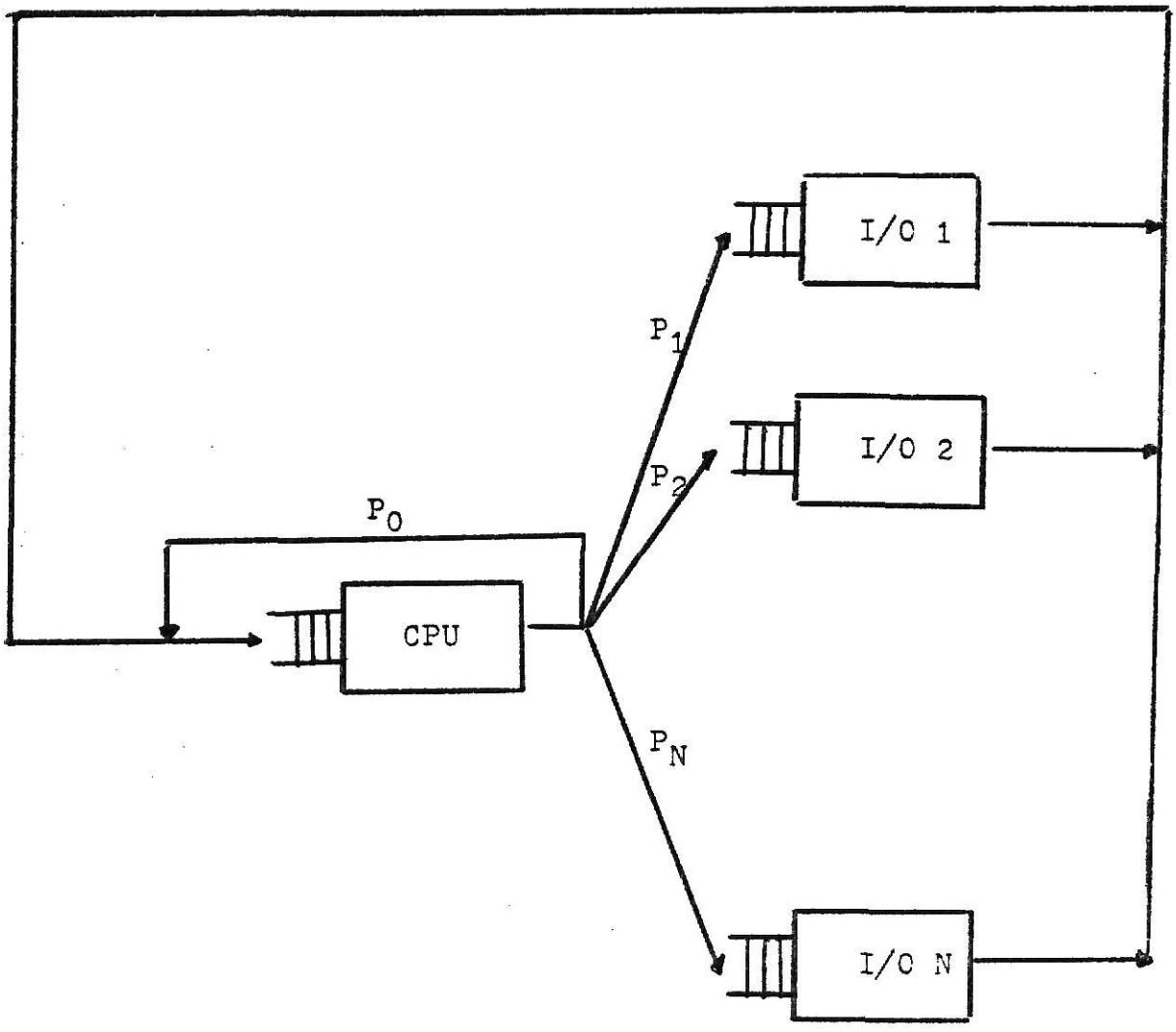
could predict the response time on the Michigan Terminal System (MTS) to within 10% [37]. At the same time, Buzen intrduced the central server model to represent the behavior of multiprogrammed computer systems [57]. This model is shown in Figure 1.8. A more complicated model is shown in Figure 1.9.

Besides the model structure, some queueing disciplines are also introduced to model computer systems. The FCFS scheduling is appropriate for modeling secondary storage input/output devices because preemptive scheduling is not possible or efficient for such devices. Processor-sharing scheduling and LCFS are appropriate models for central processing units (CPUs) since LCFS is an efficient preemptive scheduling method, and both methods have been found to improve the performance of CPUs. No queueing or IS are appropriate models for terminals and for routing delays in the network. Different customer **classes present varying service requirements, since they needs different kinds of compliers or input/output devices.**

1-4 Plan of this Report

Queueing network models are widely used to analyze the performance of modern computer systems. The intent of this report is to provide an overview of available models and methods for queueing networks of computer systems.

In Chapter 2, the development of queueing network models is reviewed. Then two different approaches to deriving those



P_0 : an old job leaves the system and a new one comes in.

P_1 : a job transfer to I/O i.

Figure 1.8 Central Server Model to Present Computer System

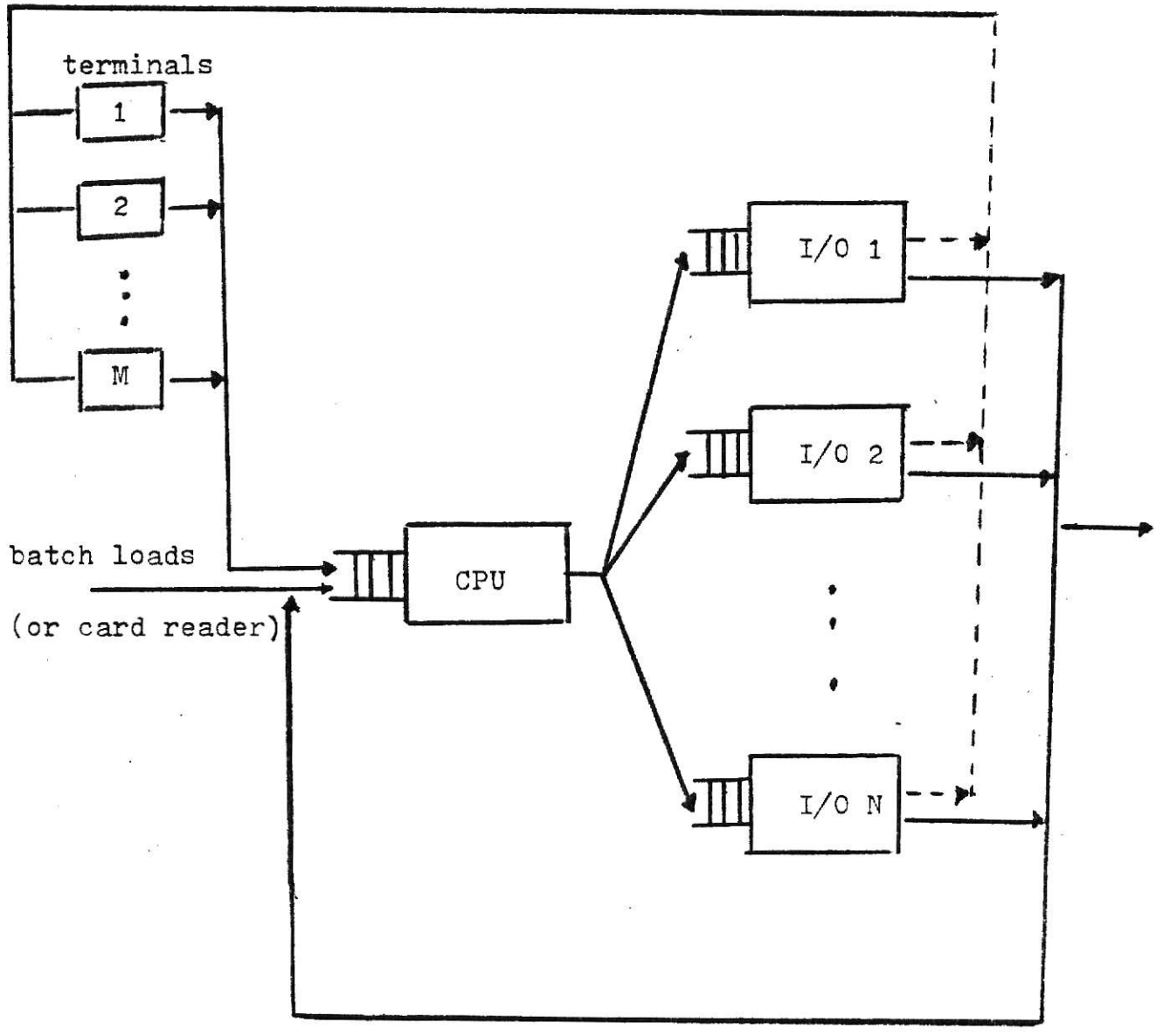


Figure 1.9 Central Server Model with Terminal and Batch Loads

models are discussed. The traditional approach is based on stochastic assumption; the other approach is operational analysis.

In Chapter 3, the efficient computational algorithms to get the performance values for the queueing networks with product form solutions are discussed. These algorithms include convolution algorithm, mean value analysis, and other approaches.

In Chapter 4, the methods for solving any general queueing network are discussed. These methods can solve the queueing networks which do not have product form solutions. Two different approaches are considered, namely numerical methods and approximation methods. Numerical methods include some matrix iteration methods and other recursive techniques. Two major methods for approximation are diffusion and aggregation.

Chapter 5 summarizes the contents of this report, and gives some areas for future research.

Lastly, a classified bibliography of research on queueing networks is placed at the end of this report. These papers are classified in an order similar to the contents of this report. The scheme of classification is as follows:

I. Survey Papers and Books.

II. Models

II.A Stochastic Analysis.

II.B Operational Analysis.

III. Methods

III.A Computational Algorithms for Product Form Solution

III.A.1 Convolutional Method.

III.A.2 Mean Value Analysis.

III.A.3 Other Methods.

III.B Numerical Methods.

III.C Approximation Methods

III.C.1 Diffusion Approximation.

III.C.2 Aggregration (or Decomposition).

III.D Simulation.

III.E Software Package.

IV. Application

IV.A Computer System.

IV.B Other Fields.

V. Miscellaneous Papers.

CHAPTER 2

QUEUEING NETWORK MODELS

In recent years the queueing network models have developed rapidly. The study of queueing networks began when Jackson studied an open network with only negative exponential service distribution and a single customer class. Presently, the most general queueing network models allow a variety of customer classes and some kinds of service stations with different queueing disciplines and service time distirbutions.

The traditional approach to derive these models is based on some stochastic assumptions. These assumptions have been studied in recent years. Analysts used the operational (i.e. directly measured) values in actual systems to validate the results of network models based on these assumptions. The repeated successed of validations led to a new approach - operational analysis - to derive the same results as before. The approach allows the analyst to test whether each assumption is met in a given sustem.

In this chapter, we first review the progress in queueing networks so far. In section 2-2, we discuss the analysis based on stochastic assumptions. In the last section, we discuss the operational analysis approach to network models.

2-1 Development of Queueing Network Models

The study of queueing network models dates back to the

1950's. In 1957, Jackson [29] studied an open network with a single class of customers. Operationally, this implied using only mean and one set of routing probabilities to describe all customers.

Jackson's model consists of an N station network, each station containing one or more parallel servers. Service time distribution at these stations. A state is a vector $\underline{n} = (n_1, n_2, \ldots, n_M)$. Other values such as station utilizations, mean queue lengths, mean waiting time and throughputs can be derived from these equilibrium state probabilities. We denote $P_i(n)$ as the marginal probability of finding n customers at the ith station, that is

$$P_i(n) = \sum_{\substack{\text{all feasible states} \\ \underline{n} \text{ such that } n_i = n}} P(n_1, n_2, \ldots, n_M)$$

Then the utilization, which measures the probability of being busy for a station, can be computed as

$$U_i = 1 - P_i(0)$$

Jackson showed that the equilibrium probability of state $\underline{n}$ in an open network is simply the product of the marginal probability for each station, i.e.

$$P(n_1, n_2, \ldots, n_M) = P_1(n_1)\ P_2(n_2) \ \ldots\ P_M(n_M)$$

This expression is termed a "product form" since it separates into a product of factors, one factor for each station in the network.

In contrast to the open network, Gordon and Newell [28] considered a closed network. In their model, there are a finite number of customers N circulating through the network, requesting and receiving service from a finite number of stations M. Each station i may have a_i parallel identical servers with negative exponential service time distribution. The state of the network is again described by a vector $\underline{n} = (n_1, n_2, \ldots, n_M)$, but the sum of all the n_i must be N.

Gordon and Newell obtained the equilibrium state probability, and proved it also satisfied a product form

$$P(\underline{n}) = P(n_1, n_2, \ldots, n_M) = \frac{1}{G(N)} \prod_{i=1}^{M} f_i(n_i)$$

where G(N) is a normalization constant chosed to make all the feasible state probabilities sum to one, that is,

$$G(N) = \sum_{\substack{\text{all feasible} \\ \text{states } \underline{n}}} \prod_{i=1}^{M} f_i(n_i)$$

The function $f_i(n_i)$ depends on the characteristics of the ith station. We defer a precise description of the $f_i(n)$ until chapter 3.

In 1963, Jackson [30] introduced a general network model which includes both open and closed networks. In this model, the external arrival is Poisson process and total arrival rate is allowed to depend upon the total number of customers in the systems. Each station may have load dependent servers with service rate depending on the total number of customers at the

service station and exponential service time distribution for the servers. Jackson demonstrated that the equilibrium state distribution of this network also has the "product form".

In 1968, Posner and Bernholtz [41] generalized the closed network in Gordon and Newells' model to permit different classes of customers with different sets of service rates and routing probabilities. Once again, it was demonstrated that the equilibrium state probability is of product form.

A more general queueing network model was developed by Baskett, Chandy, Muntz, and Palacios in 1975 [14]. This model allows multiple classes of customers and different kinds of service stations. Customers may change their classes through the system. This model describes either an open, closed, or mixed queueing network.

The queueing discipline is assumed FCFS at each station for earlier queueing network models. In the general model developed by Baskett et al., each service station can have any of the following four types of queueing disciplines:

Type 1: FCFS. Each customer has the same exponential service time distribution. The server may be load dependent with service rate depending on total number of customers at the station.

Type 2: PS. Only one server is involved. Each class of customer may have a distinct service time distribution. The distribution is arbitrary except that it must have a rational

Laplace transform.

Type 3: IS. There are sufficient servers so that no queueing will occur. Each class of customer may have a distinct, arbitrary service time distribution with rational Laplace transform.

Type 4: LCFS. There is only one server. Again, each class of customer may have a distinct arbitrary service time distribution with rational Laplace transform.

From the above disciplines, we can see this model can represent a wide range of computer systems. A state in this model represents a distribution of customers over classes and stations. For an M-station network, a state is denoted by a vector $\underline{n} = (\underline{n}_1, \underline{n}_2, \ldots, \underline{n}_M)$, where each component of $\underline{n}$ is also a vector. Let R be the total number of customers classes. The component of $\underline{n}$ can be denoted as $\underline{n}_i = (n_{i1}, n_{i2}, \ldots, n_{iR})$, where n_{ir} is the number of class r customers at station i. The equilibrium state probabilities are derived and also have product form.

In 1977, Lam [33] further extended the previous general queueing network models to include mechanisms of state dependent lost arrivals and triggered arrivals. That is, arrivals to the network from outside may be lost when the system is in certain states. Also, a new customer may be injected instantaneously into the network upon the departure of a customer from the network. This model can be used for computer communication networks with storage and flow constrints. The equilibrium

state probabilities are also proved to be product form.

2-2 Analysis Based on Stochastic Assumptions

The theory of stochastic procedded has traditionally been used as a framework for deriving queueing netwirk models. Most analysis of queueing systems begins with the stochastic hypothesis; that is, the behavior of the real system is characterized by the probability distributions of a stochastic process. Assumption used in the theory of stochastic processes include:

- The system is modeled by a stationary stochastic process;
- Customers are stochastically independent;
- Customers steps from station to station following a Markov chain;
- The service time distribution is exponential;

and so on. All these assumptions constitute a stochastic model, which produces many benefits. All the variables can be defined exactly, assumptions can be stated concisely, and a lot of known results of theory can be called on during analysis. The theory of queueing networks based on these assumptions is usually called "Markovian queueing network theory" [3].

The network models in which service time is exponentially distributed could be solved by deriving and solving the global balance equation for the network. The global balance equation will equate the transition rate of the network out of a state with the transition rate into this state; a transition out of

state $\underline{n}$ occurs when a customer at any service station i completes service. The rate of transition out of state $\underline{n}$ is

$$\sum_{j=1}^{M} P(\underline{n})\ u_j$$

Here u_j is the service rate of station j. A transition from another state into state $\underline{n} = (n_1, n_2, \ldots, n_M)$ can be written as

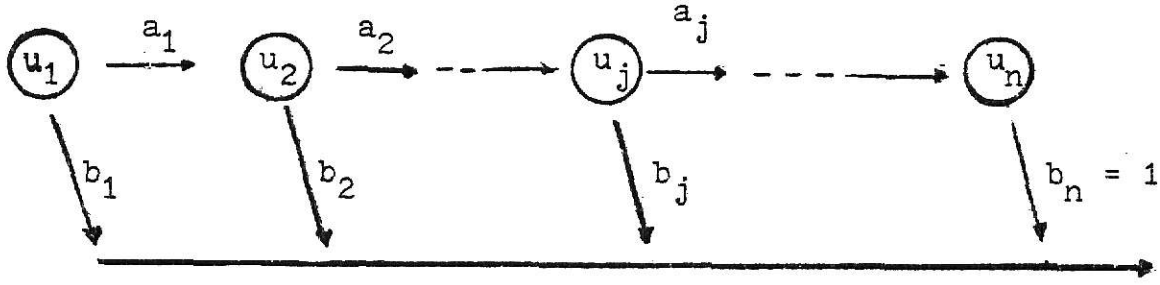
$$\sum_{i=1}^{M} \sum_{j=1}^{M} P(n_1, \ldots, n_i+1, \ldots, n_j-1, \ldots, n_M)\ u_i\ P_{ij}$$

Therefore, for the equilibrium probability in state $\underline{n} = (n_1, n_2, \ldots, n_M)$, we have

$$\sum_{j=1}^{M} P(n_1, \ldots, n_i, \ldots, n_j, \ldots n_M)\ u_j =$$
$$\sum_{i=1}^{M} \sum_{j=1}^{M} P(n_1, \ldots, n_i+1, \ldots, n_j-1, \ldots, n_M)\ u_i\ Pij$$

Solving these linear equations and applying the global balance equation to all feasible states will result in a product form solution.

For networks with general service time distribution, we can still preserve the Markovian nature. Cox [24] has shown that any service time distribution with a rational Laplace transform can be represented by a network of exponential stages of the form shown in the following figure:



In this figure, a station with general service time distribution is represented by a network of stages. The service time at each stage is exponentially distributed. In terms of the above figure, a customer proceeds to stage j + 1 of the station, after completing service at stage j with probabilty a_j, or exits from the service station with probability $b_j = 1 - a_j$. At each stage j the service rate is u_j. This representation of general service time distribution is called the "method of stages" [12].

The global balance equation becomes unmanageable when applied to some complex models, since it may consist of an enormous (yet finite) number of states. Chandy [19] discovered that the equilibrium probabilities for the system states with Markovian nature obey not only the global balance equations but also the "local balance equations", and proved that a solution to the local balance equations is also a solution of the global balance equation. The local balance equation equates the rate of entry to a given state caused by a customer entering a given queue with the rate of exit from that state caused by a customer exiting the given queue. A local balance equation exists for each queue of the network. For a queue at service station j of the network, the local balance equation would be

$$P(n_1,\ldots,n_i,\ldots,n_j,\ldots,n_M)\ u_j = \sum_{i=1}^{M} P(n_1,\ldots,n_i+1,\ldots,n_j-1,\ldots,n_M)\ u_i\ P_{ij}$$

Local balance equations have been used to derive the equilibrium state probability for the general network model developed by Baskett et al., and proved it to be product form [39].

Muntz [38] investigated the "M ⇒ M" property of Poisson arrivals implying Poisson departures. He showed that a network of queues with the "M ⇒ M" property has product form and also had the "M ⇒ M" property.

2-3 Operational Analysis of Queueing Networks

The Markovian assumptions used to derive the queueing network model are often violated by many computer systems. For example, service time is not exponential, station to station transitions do not follow Markov chains, parameters change over time. Therefore, many people doubt the accuracy of the queueing network model. In recent years some analysts studied the validation of the results of Markovian queueing networks theory with operational (i.e. directly measured) values of real systems. The repeated successes of validation led to an investigation of the relationship between operational values, and the use of different assumptions to derive the same results as before [45 - 51]. The new approach is called "operational analysis".

In operational analysis, the real system is observed for a finite period of time. All the quantities - such as utilizationa, completion rates, mean queue sizes - should be defined so as to be precisely testable. The precision of results should depend only on these assumptions which can be

tested by observing a real system for a finite period of time. All the equations in operational analysis depend on these four assumptions:

1. Job flow balance: the number of jobs (or customers) which are observed to arrive at a given station is (almost) the same as the number being observed to depart from the station for this finite period of time;

2. State transition balance: the number of transitions into a given system state is (almost) the same as the number out of the state for this finite period of time;

3. One step behavior: the only observable state changes result from single customers either entering the system, or moving between pairs of stations in the system, or exiting from the system. In other words, the simultaneous customer-move is negligible.

4. Homogeneity; for stations, the output rate of a station is determined completely by its queue length, and is independent of queue lengths of other stations; for routing, the routing of customers is independent of the system's state.

Using operational analysis, we can also get the product form solution for state probability. In operational analysis, the state probability $P(\underline{n})$ represents the proportion of time state $\underline{n}$ is occupied. Before discussing how to get product form solution, we need some notations to defined:

$\underline{k}$, $\underline{n}$, $\underline{m}$ denote distinct system states,

T : total observed time period,

$T(\underline{n})$: total time during which state $\underline{n}$ is occupied,

$T_i(n)$: total time during which the number of customers in station i is n,

$C(\underline{n},\underline{m})$: the number of one-step state transitions observed from $\underline{n}$ to $\underline{m}$,

$r(\underline{n},\underline{m})$: $C(\underline{n},\underline{m}) / T(\underline{n})$ the number of transitions per unit time when $\underline{n}$ is occupied,

$P(\underline{n})$: $T(\underline{n}) / T$ the proportion of time $\underline{n}$ is occupied,

$C_{ij}(n)$: the number of times at which a customer requests service at station j immediately after completing a service request at station i, when the number of customers in station i is n,

$C_i(n)$: $\sum_{n=1} C_i(n)$ the total number of times at which a customer moves out of station i,

C_i : $\sum_{n=1} C_i(n)$ the total number of times at which a customer moves out of station i,

C_{ij} : $\sum_{n=1} C_{ij}(n)$ the total number of times at which a customer moves to service j immediately after completing service at station i,

$S_i(n)$: $T_i(n) / C_i(n)$ the mean time between service completion (or mean service time) at station i when n customers are there,

q_{ij} : C_{ij} / C_i routing frequency, the fraction of customers moving to station j immediately after completing service at service i,

X_i : throughputs or output rate of station i.

From state transition balance assumption, we have

$$\sum_{\underline{k}} C(\underline{k},\underline{n}) = \sum_{\underline{m}} C(\underline{n},\underline{m}) \qquad \text{for all } \underline{n}.$$

Since transition rate $r(\underline{n},\underline{m}) = C(\underline{n},\underline{m}) / T(\underline{n})$ and state probability $P(\underline{n}) = T(\underline{n}) / T$, we can change state transition balance to

$$\sum_{\underline{k}} P(\underline{k})\, r(\underline{k},\underline{n}) = P(\underline{n}) \sum_{\underline{m}} r(\underline{n},\underline{m}) \qquad \text{for all } \underline{n}.$$

From these equations, we can express $P(\underline{n})$ in terms of $r(\underline{n},\underline{m})$. This form of expression is generally not useful since $r(\underline{n},\underline{m})$ needs too much computation.

To reduce the number of states involved, we introduce one step behavior assumption. This assumption asserts that the simultaneous customer-moves will not be observed, and that transitions are possible only between neighboring states. Let us denote $\underline{n} = (n_1,\ldots,n_i,\ldots,n_j,\ldots n_M)$ and $\underline{n}_{ij} = (n_1,\ldots,n_i-1,\ldots,n_j+1,\ldots,n_M)$ are two states neighboring each other. When a customer moves from station i to station j, the system will move from state $\underline{n}$ to $\underline{n}_{ij}$. After this one step behavior assumption, the state balance equations can be transformed into

$$\sum_{\substack{i=1 \\ i\neq j}}^{M} \sum_{j=1}^{M} P(\underline{n}_{ij})\, r(\underline{n}_{ij},\underline{n}) = P(\underline{n}) \sum_{\substack{i=1 \\ i\neq j}}^{M} \sum_{j=1}^{M} r(\underline{n},\underline{n}_{ji}) \qquad \text{for all } \underline{n}.$$

Using the homogeneity assumption, we can replace $r(\underline{n},\underline{m})$ with $S_i(n)$ and q_{ij} which can easily be obtained. For stations, the homogeneity assumption gives us

$$r(\underline{n},\underline{n}_{ij}) = \frac{C(\underline{n},\underline{n}_{ij})}{T(\underline{n})} = \frac{C_{ij}(n_j)}{T_i(n_i)}$$

For routing, the homogeneity assumption give us $q_{ij} = C_{ij} / C_i$

is the same for the system. Combining these two relationships, we have

$$r(\underline{n},\underline{n}_{ij}) = q_{ij} \,/\, S_i(n_i)$$

Replacing it, the previous balance equations will transform into

$$\sum_{\substack{i=1 \\ i\neq j}}^{M} \sum_{j=1}^{M} P(\underline{n}_{ij}) \frac{q_{ji}}{S_j(n_j+1)} = P(\underline{n}) \sum_{j=1}^{M} \frac{1 - q_{jj}}{S_j(n_j)} \quad \text{for all } \underline{n}.$$

$P(\underline{n})$ can be derived by solving these equations, and proved to be of product form [50,51].

To simplify these equations, we can replace routing frequencies with throughput. From job flow balance equations, we have

$$\text{output rate} = \text{input rate} \quad \text{for all stations, or}$$
$$X_j = \sum_{i=1}^{M} X_i \, q_{ij} \quad \text{for } j = 1, 2, \ldots, M.$$

X_i can be obtained by solving these linear equations.

Operational analysis can also be applied to queueing network models with multiple classes of customers [53]. Compared with Markovian queueing network theory, operational analysis is easily understood and applied since it does not involve advanced queueing theory. In addition, its results can be applied to more classes of networks. The drawback of operational analysis is that it cannot deal with transient behavior.

CHAPTER 3

COMPUTATIONAL ALGORITHMS FOR PRODUCT FORM SOLUTIONS

All the models of queueing networks previously described can result in product form solutions for equilibrium state probability. For the open network, this equilibrium state probability $P(\underline{n})$ is simply the product of the marginal probability for each station, which can be analyzed separately. For closed or mixed networks, we have to compute the normalization constant $G(N)$ before determining the equilibrium state probability.

In this chapter, we discuss some efficient computational algorithms to calculate the normalization constant $G(N)$, and other performance values for the closed queueing network model. In section 3-1, we describe the product form solution and the performance values needed when we analyze a real system. In addition, the computational problem is pointed out when we calculate the normalization constant. Next, two well known computational algorithms - convolution algorithm (or normalization constant method) and mean value analysis - are discussed in section 3-2 and section 3-3, respectively. In section 3-4 we mention some other efficient algorithms.

3-1 Product Form Solution and Performance Values

For a closed queueing network with a single customer class, the equilibrium state probability is [28,30]

$$P(\underline{n}) = P(n_1, n_2, \ldots, n_M) = \frac{1}{G(N)} \prod_{i=1}^{M} f_i(n_i)$$

In this model we have M stations and N customers; queueing disciplines are assumed FCFS for all stations, and all service time distributions are exponential. The service rate at each station may be dependent on the total number of customers at the service station. For this solution G(N) is a normalization constant, or

$$G(N) = \sum_{\substack{\text{all feacible} \\ \text{state } \underline{n}}} \prod_{i=1}^{M} f_i(n_i)$$

$f_i(n_i)$ is a function depending on station i, or

$$f_i(n_i) = e_i^n \Big/ \prod_{k=1}^{n_i} u_i(k)$$

where $u_i(n_i)$ is the service rate for station i when there are n customers at service station i. e_i is the relative throughput to station i, which can be calculated from flow balance equations:

$$e_j = \sum_{i=1}^{M} e_i P_{ij} \qquad j = 1, 2, \ldots, M.$$

where P_{ij} is the transition probability from station i to station j. Notice that the state probability is the same as the queue length distribution for a single class network.

The expression for the equilibrium state probability is a little complicated in the closed queueing network with multiple customer classes and different queueing disciplines [14]. In

addition to M stations and N customers, we have R distinct customer classes. This model permits four different queueing disciplines: FCFS (First Come First Served), PS (Processor Sharing), IS (Infinite Servers), and LCFS (Last Come First Served). Except that we need the exponential service time distribution at the stations with FCFS queueing discipline, the general service time distributions are assumed for the stations with the other three queueing disciplines. In this model, a state is a distribution of customers over classes and stations other than a queue length distribution. The state can be denoted by a vector $\underline{n} = (n_1, n_2, \ldots, n_M)$, where $\underline{n}_i = (n_{i1}, n_{i2}, \ldots, n_{iR})$ n_{ir} is the number of class r customers at station i. Let us restrict our attention to the multiple class networks which do not allow customers to change class membership. The equilibrium state probability is [56]

$$P(\underline{n}) = P(\underline{n}_1, \underline{n}_2, \ldots, \underline{n}_M) = \frac{1}{G(N)} \prod_{i=1}^{M} f_i(\underline{n}_i)$$

where

$$f_i(n_i) = \begin{cases} \dfrac{n_i!}{\prod_{k=1}^{n_i} u_i(k)} \prod_{r=1}^{R} \dfrac{1}{n_{ir}!} e_{ir}^{n_{ir}} & \text{if station i has FCFS PS, LCFS queueing disciplines.} \\ \prod_{i=1}^{R} \dfrac{1}{n_{ir}!} \left(\dfrac{e_{ir}}{u_{ir}} \right)^{n_{ir}} & \text{if station i has an IS queueing discipline.} \end{cases}$$

e_{ir} is the relative throughput for calss r customers at station i, and can be calculated from flow balance equations for r = 1, 2, ..., R

$$e_{jr} = \sum_{i=1}^{M} e_{ir} P_{ij}(r) \qquad j = 1, 2, \ldots, M.$$

where $P_{ij}(r)$ is the transition probability from station i to station j for class r customers. The equilibrium state probability for multiple calss networks with class changes is also a similar form [14].

In addition to equilibrium state probability, some other values such as station utilization, throughput, mean queue length, and mean waiting time are also needed when evaluating the performances of computer systems. After obtaining the equilibrium state probability $P(\underline{n})$ for a closed network with a single customer class, the marginal probability $p_i(n)$, which is the probability of finding n customers at the ith station, can be computed as

$$P_i(n) = \sum_{\substack{\text{all feasible states} \\ \underline{n} \text{ such that } n_i=n}} P(n_1, n_2, \ldots, n_i, \ldots, n_M)$$

Then the utilization U_i, which measures the probability of being busy for station i, can be calculated as

$$U_i = \sum_{i=1}^{N} P_i(n) = 1 - P_i(0)$$

The throughput X_i, which is the output rate of station i, can be calculated as

$$X_i = \sum_{i=1}^{N} P_i(n)\, u_i(n)$$

Here $u_i(n)$ is the load dependent service rate for station i. The mean queue length $\bar{n}_i$, which is the average number of

customers staying at station i, can be calculated as

$$\bar{n}_i = \sum_{n=1}^{N} n \, P_i(n)$$

The mean waiting time $\bar{w}_i$, which is the average time a customer spends at station i (including both waiting time and service time), can be calculated from Little's equation [35] which relates mean waiting itme with mean queue length and throughput for any queueing system, i.e.

$$\bar{w}_i = \frac{\bar{n}_i}{X_i}$$

From the above discussions, we can see the importance of G(N) in a closed queueing network with product form solution. The computation of G(N) is dependent on the number of feasible states. For a single class queueing network with total number of customers N, a feasible state $\underline{n} = (n_1, n_2, \ldots, n_M)$ is the state in which the sum of all the n_i is equal to N. i.e. $\sum_{i=1}^{M} n_i = N$ and $n_i \geq 0$ for i = 1, 2, ..., M. The number of feasible states is the number of ways one can place N customers among the M service stations, and is equal to $\binom{N + M - 1}{M - 1}$. For a relatively modest model in which M = 6 and N = 20, we have 53130 feasible states, and the calculation of G(N) requires the summation of 53130 terms, each of which is the product of 6 factors. For multiple class queueing networks, a state is a distribution of customer over classes and stations, so there is an even greater number of feasible states. Therefore, a straightforward evaluation of G(N) must be avoided.

3-2 Convolution Algorithm

The convolution algorithm is a recursive algorithm for computing G(N). This algorithm was introduced by Buzen [57]. Buzen used this algorithm only for closed single class qyeyeing networks. Adopting generating function method, Reiser and Kobayashi generalized the convolution method to systems with multiple class networks and mixed networks [64,66]. Below, we discuss the basic scheme of the convolution method for closed single class queueing networks.

3-2.1 The Computation of the Normalization Constant

In the closed single class queueing network models, we have M stations and N customers. Each station has only FCFS queueing discipline and exponential service time distribution. Service rate are dependent only on the total number of customers at these stations. Let $u_i(n)$ denote the service rate for station i when there are n customers, and let p_{ij} be the transition probability from station i to station j. The relative throughput for each station i, denoted by e_i, can be calculated from flow balance equations, or solving these N linear equations:

$$e_j = \sum_{i=1}^{M} e_i P_{ij} \qquad j = 1, 2, \ldots, M.$$

In this model, the equilibrium state probability is shown as

$$P(\underline{n}) = P(n_1, n_2, \ldots, n_M) = \frac{1}{G(N)} \prod_{i=1}^{M} f_i(n_i)$$

where

$$f_i(n_i) = \frac{e_i^{n}}{\prod_{k=1}^{n_i} u_i(k)}$$

G(N) is a normalization constant, or

$$G(N) = \sum_{\substack{\text{all feasible} \\ \text{states } \underline{n}}} \prod_{i=1}^{M} f_i(n_i)$$

Since a feasible state $\underline{n} = (n_1, n_2, \ldots, n_M)$ is the state in which the sum of all the n_i equals N, we can define the set of all feasible states as S(N,M), where

$$S(N,M) = \left\{(n_1, n_2, \ldots, n_M) \mid \sum_{i=1}^{M} n_i = N,\ n_i \geq 0,\ i = 1,2,\ldots,M\right\}$$

Hence the normalization constant can be given by

$$G(N) = \sum_{\underline{n} \in S(N,M)} \prod_{i=1}^{M} f_i(n_i)$$

To derive this algorithm, Buzen [58] introduced an auxiliary function g(n,m), where g(n,m) is defined as

$$g(n,m) = \sum_{\underline{n} \in S(n,m)} \prod_{i=1}^{m} f_i(n_i)$$

for n = 0, 1, 2, ..., N and m = 1, 2, ..., M. Note that G(N) is equal to g(N,M), and in fact,

$$g(n,M) = G(n) \qquad \text{for } n = 0, 1, 2, \ldots, N.$$

Now observe that for m > 1,

$$
\begin{aligned}
g(n,m) &= \sum_{\underline{n} \in S(n,m)} \prod_{i=1}^{m} f_i(n_i) \\
&= \sum_{k=0}^{n} \Big[\sum_{\underline{n} \in S(n,m) \;\&\; n_m = k} \prod_{i=1}^{m} f_i(n_i) \Big] \\
&= \sum_{k=0}^{n} f_m(k) \Big[\sum_{\underline{n} \in S(n-k,m-1)} \prod_{i=1}^{m-1} f_i(n_i) \Big] \\
&= \sum_{k=0}^{n} f_m(k)\, g(n-k,m-1)
\end{aligned}
$$

This expression is a recurrence relation, i.e. $g_{m-1}(n)$ is required before $g_m(n)$ can be computed. The initial condition for this recurrence relation can be obtained from observing that

$$
\begin{aligned}
g(n,1) &= \sum_{\underline{n} \in S(n,1)} \prod_{i=1}^{1} f_i(n_i) \\
&= \sum_{\underline{n} \in \{(n)\}} f_i(n) \\
&= f_i(n) \qquad \text{for } n = 0, 1, 2, \ldots, N.
\end{aligned}
$$

and $g(m.0) = 1$ for $m = 1, 2, \ldots, M$.

Therefore, G(N) can be computed iteratively. Table 3.1 provides a schematic representation of the algorithm for a single class closed queueing network. Buzen originally required two vectors of size N + 1 for storage of computation in this algorithm [58]. But observe that if the elements in the mth column are computed starting with the last element g(N,m) and proceeding to the first element g(1,m), then we can reduce the need of storage to only one vector of size N + 1. For the network with load independent servers, the computation will be simpler and easier. The computational algorithm for the load

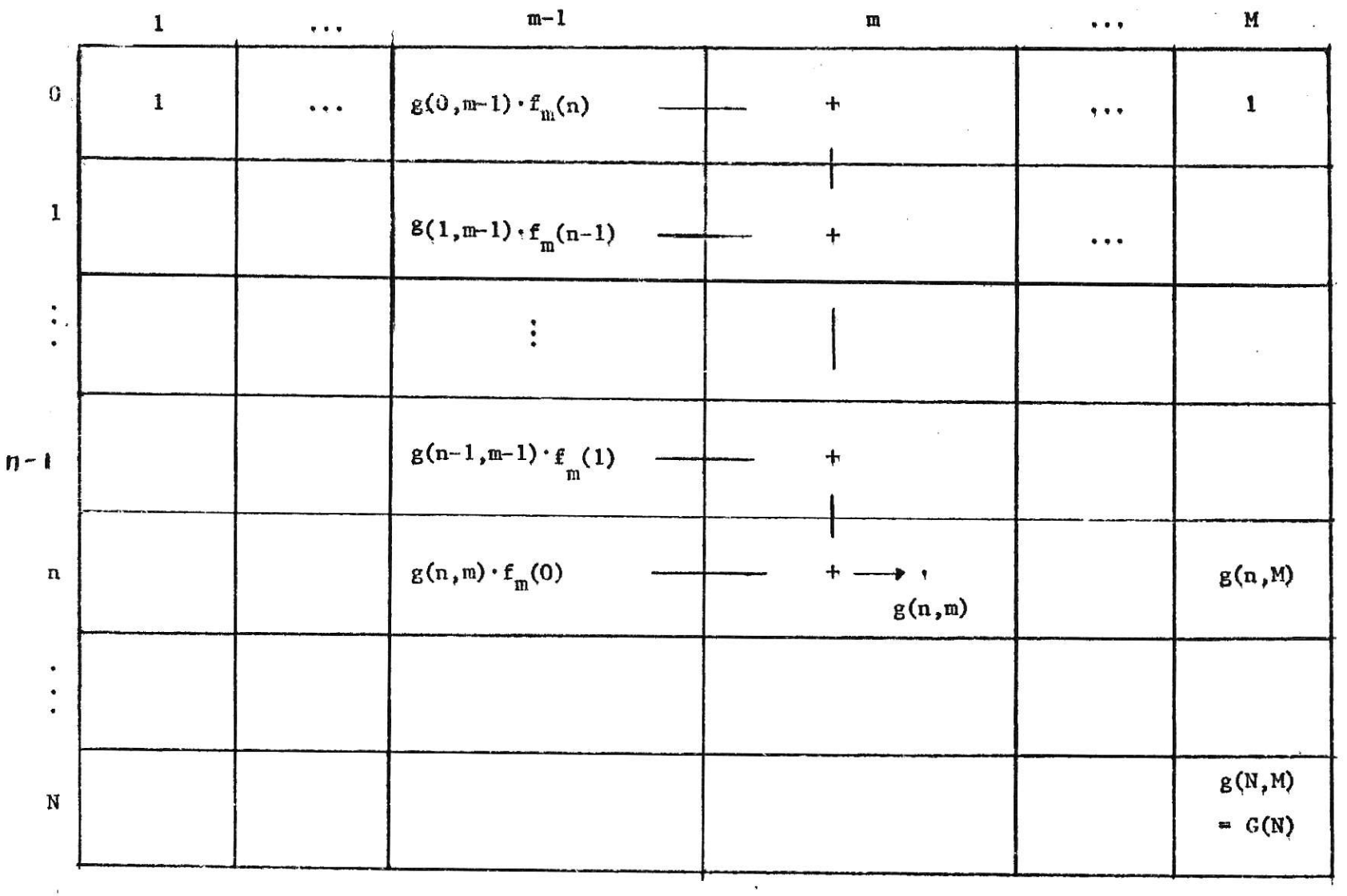


Table 3.1 Algorithms for Computing Normalization Constant of a Single Class Closed Queueing Network

independent case is presented in [56,58].

3-2.2 The Marginal Probability

Besides equilibrium stat probability, we sometimes want to know the probability of finding n customers at each station i, i.e. marginal probability $P_i(n)$. On the other hand, the marginal probability $P_i(n)$ is needed when we want to compute some performance values such as utilization, throughput, and mean queue length.

The marginal probability $P_i(n)$ can be expressed by

$$P_i(n) = \sum_{\underline{n} \in S(N,M) \ \& \ n_i = n} P(n_1,\ldots,n_{i-1},n,n_{i+1},\ldots,n_M)$$

One can get $P_i(n)$ directly from summation of all feasible states with $n_i = n$. This computational scheme is not efficient since there are too many states involved. Alternatively, we discuss a more efficient algorithm.

The above expression for $P_i(n)$ can be simplified as

$$P_i(n) = \frac{1}{G(N)} \sum_{\underline{n} \in S(N,M) \ \& \ n_i = n} \prod_{j=1}^{M} f_j(n_j)$$

$$= \frac{f_i(n)}{G(N)} \sum_{\underline{n} \in S(N,M) \ \& \ n_i = n \ \& j \neq i} \prod_{j \neq i}^{M} f_j(n_j)$$

To derive an algorithm for computing $P_i(n)$, we need to define another auxiliary function $g^i(n,m)$ as

$$g^i(n,m) = \sum_{\underline{n} \in S(n,m) \ \& \ n_i = n} \prod_{i=1}^{m} f_i(n_i)$$

This term can be viewed as a normalization constant of a network which is the same network as the original one except removing station i and having only n customers. With this new auxiliary function definition, the marginal probability becomes

$$P_i(n) = \frac{f_i(n)}{G(N)} g^i(N-n,M)$$

An algorithm is then needed to evaluate this new auxiliary function $g^i(N-n,M)$. The derivation of this algorithm is based on the fact that the marginal probabilities must sum to 1. Therefore, we have

$$1 = \sum_{k=0}^{N} P_i(k) \qquad \text{for each station i}$$

$$= \sum_{k=0}^{N} \frac{f_i(k)}{G(N)} g^i(N-k,M)$$

Hence,

$$G(N) = \sum_{k=0}^{N} f_i(k) \, g^i(N-k,M)$$

and

$$g^i(N,M) = G(N) - \sum_{k=1}^{N} f_i(k) \, g^i(N-k,M)$$

Values of this auxiliary function can be computed by this recurrence relation with the initial condition

$$g^i(0,M) = G(0) = 1$$

3-2.3 Other Performance Values and Considerations

In this section, we represent the algorithm for computing some performance values. These performance values are throughput, utilization, mean queue length, and mean waiting time at each station. Most of these performance values can be simply computed by means of the normalization constant. Therefore, some people use the term "normalization constant method" instead of "convolution method" [55,56].

The throughput of a station measures the rate at which customers leave that station. Hence, the throughput X_i can be expressed as

$$X_i = \sum_{n=1}^{N} P_i(n)\, u_i(n)$$

When $P_i(n)$ is replaced by the expression derived in the previous section, X_i can be transformed as [56]

$$X_i = e_i \frac{G(N-1)}{G(N)}$$

Thus the value of throughput can be simply obtained by computing this expression.

The utilization denoted by U_i is the measurement of the fraction of time station i is busy. By definition, we have

$$\begin{aligned} U_i &= \sum_{n=1}^{N} P_i(n) \\ &= 1 - P_i(0) \\ &= 1 - \frac{g^i(N,M)}{G(N)} \end{aligned}$$

The mean queue length $\bar{n}_i$ is the average number of customers staying at station i and needs to be evaluated from its definition, that is

$$\bar{n}_i = \sum_{n=1}^{N} n \, P_i(n)$$

The mean waiting time for station i is the average time a customer spends at it. The mean waiting time $\bar{w}_i$ can be obtained directly by Little's equation [35]. Little's equation means that the average number of customers in the queueing system is equal to the average arrival rate of customers to that system times the average time spent in that system. In a steady-state, the average arrival rate in a system is the same as the average output rate (or throughput). Hence, Little's equation can be expressed as

$$\bar{n} = X \, \bar{w}$$

When applying Little's equation to the queueing system at station i, we have

$$\bar{w}_i = \frac{\bar{n}_i}{X_i}$$

The convolution algorithm described above is restricted to the single class closed queueing network. Extensions of this algorithm to more general queueing networks can be seen in [56,64,66].

The problem of exceeding the floating point range may sometimes occur during the computation of the normalization constant [59]. This problem has been discussed and can be avoided by some methods [55,56,59,67,68].

Some computer program packages have been developed to solve large queueing networks with product form solutions. A typical one is QNET [170], which is now one component of another package RESQ (RESearch Queueing analyzer) [174]. Another typical one is PNET (Purdue NETwork of queues evaluator) [55,56].

3-3 Mean Value Analysis

Mean value analysis is another recursive algorithm for solving the closed queueing networks with product form solutions.

In this algorithm, mean queue lengths, mean waiting time, and throughputs can be computed iteratively without computing product terms and normalization constants. This new technique was developed by Reiser and Lavenberg in [70,71]. Bard extended this technique to the more general case [69].

The mean value analysis is based on intuitively appealing principles, namely;

1. The queue length distribution seen by a customer upon his arrival at a given station is the same as the overall distribution seen by an outside observer when one less customer is in the system.

2. Little's equation can be applied to the entire system and to each station individually.

The first principle has been proven to hold in all closed queueing networks with product form solutions [245,267]. For a network with load independent servers, the following equation can be obtained from the first principle,

$$\bar{w}_i(n) = S_i + S_i \bar{n}_i(n-1)$$

where $S_i = 1 / u_i$ is the mean service time at each station i, $\bar{w}_i(n)$ and $\bar{n}_i(n)$ denotes $\bar{w}_i$ and $\bar{n}_i$ with only n customers in the system. This equation states that the mean waiting time of a customer at each station i ia equivalent to its average service time plus the mean time spent by the previously queued customers at this station. For load dependent cases, we have the similar equation from the first principle, namely,

$$\bar{w}_i(n) = \sum_{k=1}^{n} k \, S_i(k) \, P_i(k-1,n-1)$$

where $P_i(k,n)$ denotes the marginal probability $P_i(k)$ when only n customers are in the system. The above equation implies that the mean waiting time of a customer at each station is the average of the backlog that exists at this station. The average backlog at each station is the average service time spent by all customers when the station has different loads. In agreement with principle 1, the weighting factor used in computing the average is the marginal probabilitydistribution at this station when n - 1 customers circulate throughout the system.

The second priciple states that Little's equation can be used to transform mean queue lengths and mean waiting times throughout the network. When Little's equation is applied to the entire queueing network, the system throughput X_s can be obtained, i.e.

$$X_s = \frac{n}{\sum_{i=1}^{M} e_i w_i(n)}$$

Note that the average number of customers in the system is n, and the mean time a customer spends in the system is the sum of the time it spends at each of the individual service stations. The average time a customer spends at each station can be expressed as the mean waiting time a customer spends per visit times the mean number of visits to this station, which is the relative throughput at this station. Then the throughput at each station i can be computed as the relative throughput at this station times system throughput, i.e.

$$X_i(n) = e_i \, X_s(n)$$

When Little's equation is applied to each station i, we can get the mean queue length, namely

$$\bar{n}_i = X_i(n)\ \bar{w}_i(n)$$

Combining the above discussions, we have the following recursive equations for i = 1, 2, ..., M and n = 1, 2, ..., N in independent case:

$$\bar{w}_i(n) = S_i \qquad \text{if station i is IS,}$$
$$= S_i(1+\bar{n}_i(n-1)) \qquad \text{otherwise.}$$

$$X_s(n) = \frac{n}{\sum_{i=1}^{M} e_i \bar{w}_i(n)}$$

$$X_i(n) = e_i\ X_s(n)$$

$$U_i(n) = S_i\ X_i(n)$$

$$\bar{n}_i(n) = X_i(n)\ \bar{w}_i(n)$$

with initial condition $\bar{n}_i(0) = 0$ for i = 1, 2, ..., M. The above equation for utilization $U_i(n)$ is proved to be true in the load independent case [56].

In load dependent cases, the computation of mean waiting time involves the marginal probability. The recursive relation exists for the marginal probability [71], that is

$$P_i(k,n) = S_i(k)\ X_i(n)\ P_i(k-1,n-1) \qquad \text{for i = 1, 2, ..., M}$$

In addition, we have

$$\sum_{k=0}^{n} P_i(k,n) = 1 \qquad \text{for } i = 1, 2, \ldots, M$$

or

$$P_i(0,n) = 1 - \sum_{k=1}^{n} P_i(k,n) \qquad \text{for } i = 1, 2, \ldots, M$$

Therefore, the recursive equations for load dependent cases for $i = 1, 2, \ldots, M$ and $n = 0, 1, 2, \ldots, N$ will be

$$P_i(k,n) = S_i(k)\, X_i(n)\, P_i(k-1,n-1)$$

$$P_i(0,n) = 1 - \sum_{k=1}^{n} P_i(k,n)$$

$$\bar{w}_i(n) = S_i \qquad \text{if station } i \text{ is IS,}$$

$$= \sum_{k=1}^{\bar{n}} k\, S_i(k)\, P_i(k-1,n-1) \qquad \text{otherwise.}$$

$$X_s(n) = \frac{n}{\sum_{i=1}^{M} e_i\, \bar{w}_i(n)}$$

$$X_i(n) = e_i\, X_s(n)$$

$$U_i(n) = 1 - P_i(0,n)$$

$$\bar{n}_i(n) = X_i(n)\, \bar{w}_i(n)$$

with initial condition $P_i(0,0) = 1 \quad$ for $i = 1, 2, \ldots, M$

Compared with the convolution algorithm, mean value analysis is easy to intuit and simple to implement or program. It will never exceed the floating point range during computation. But the recursive expressions for marginal probability may fail for

relatively small populations since $P_i(0,n)$ may turn out to be negative [59]. Hence, mean value analysis may not be able to handle some networks with small populations and variable service rates. Reiser proposed a modification of mean value analysis which avoids this problem [62]. But the modification requires additional computations.

3-4 Other Efficient Algorithms

In addition to the two algorithms described, there are other algorithms which give still less computation. One of them is partial fraction approach. This approach is based on partial fraction expansion of the generating function, and gives an explicit expression for the normalization constant G(N). This approach was developed by Moore for closed queueing networks with single class and exponential service time distribution [77]. Reiser and Kobayashi [64], and Lam [76] extended this approach to general queueing networks. This approach has the same computer efficiency as the convolution method, but it will become numerically unstable since it requires the summation of terms with alternating signs which may be subject to round-off errore in some cases [64]. Hence, the partial fraction approach is clearly inferior when compared with the convolution method.

In recent years Kobayashi [75] proposed a new computational algorithm for calculating normalization constant and other performance values of a queueing network. This algorithm is based on the Polya theory of counting, which is an application of group theory to combinatorial problems. This algorithm also

deirves some recursive equations to compute G(N). The amount of computations required in this algorithm is the order of N^2, where N is the total number of customers. Thus the algorithm is preferable to the convolution algorithm when there are many service stations and N is small. But this algorithm is now restricted to a network with exponential serivce time distribution and constant service rates.

Two new algorithms were proposed by Chandy and Sauer in 1980 [59]. One is similar to mean value analysis and the other one is closely related to the convolution algorithm. The former is called the Local Balance Algorithm for Normalizing Constant (LBANC), and the latter is the algorithm to Coalesce Computation of Normalizing Constant (CCNC). LBANC depends on Little's equation, but it also requires the normalization constant G(N). We use the unnormalized mean queue length to keep the equations recursive. The normalization constant can easily be calcuated from these unnormalized mean ququq lengths. For a closed queueing network with load dependent service rates, the unnormalized mean queue length, denoted by $\overline{LBn_i}(n)$, can be expressed as

$$\overline{LBn_i}(n) = \sum_{k=1}^{n} k \, LBP_i(k,n)$$

where $LBP_i(k,n)$ is the unnormalized queue length distribution for station i, or unnormalized marginal probability, we have

$$\sum_{k=0}^{n} LBP_i(k,n) = G(n)$$

or

$$LBP_i(0,n) = G(n) - \sum_{k=1}^{n} LBP_i(k,n)$$

The relation between unnormalized mean queue length $\overline{LBn}_i(n)$ and mean queue length $\bar{n}_i(n)$ is

$$\overline{LBn}_i(n) = \bar{n}_i(n)\ G(n)$$

To get the normalized constant G(n), wenotice the fact that

$$\sum_{i=1}^{M} \bar{N}_i(n) = n$$

Thus, G(n) can be obtained from the above two equations, namely

$$G(n) = \sum_{i=1}^{M} LBn_i(n) \ / \ n$$

Therefore, we have the recursive equations for i = 1, 2, ..., M and n = 0, 1, 2, ..., N

$$LBP_i(k,n) = S_i(k)\ X_i(n)\ LBP_i(k-1,n-1)$$

$$\overline{LBn}_i(n) = \sum_{k=1}^{n} k\ LBP_i(k,n)$$

$$G(n) = \sum_{i=1}^{M} \overline{LBn}_i(n) \ / \ n$$

$$LBP_i(0,n) = G(n) - \sum_{k=1}^{n} LBP_i(k,n)$$

$$\bar{n}_i(n) = \overline{LBn}_i(n) \ / \ G(n)$$

$$X_i(n) = e_i \ \frac{G(N-1)}{G(N)}$$

$$w_i(n) = \bar{n}_i(n) \ / \ X_i(n)$$

$$U_i(n) = 1 - P_i(0,n) = 1 - LBP_i(0,n) \ / \ G(n)$$

with initial condition $LBP_i(0,0) = 1$ for i = 1, 2, ..., M

This new algorithm has the same computational efficiency as mean value analysis and convolution algorithm. LBANC is also very simple to program, and can be applied to mixed networks which

cannot be applied in mean value analysis. But this algorithm has the numerical problems which happen in both convolution algorithm and mean value analysis. That is, it may fail because the normalization conatant exceeds the floating point range for some populations, and may fail for relatively small populations.

Another algorithm CCNC is intended for use with programmable calculators. This algorithm gives an explicit expression for normalization constant. Then it computes the normalization constant by taking advantage of exponentiation and factorial operations which are usually provided as machine instructions in calculators. Other performance values can be computed in the same manner as in LBANC. But this algorithm applies only to the queueing networks with constant serivce rates.

Zahorjan [79] also proposed a convolution algorithm for queueing networks with product form solutions. This algorithm leads to a different way to determine the normalization constant and other performance values. Not only can it be applied to all the queueing network models as before, this new algorithm is the first efficient algorithm to deal with Lam-type networks [33] which allow external arrivals to be triggered by a departing customer, and arriving customers to be lost depending on the current state.

CHAPTER 4

METHODS FOR SOLVING GENERAL QUEUEING NETWORKS

The computation algorithms discussed in Chapter 3 are applied only to queueing networks with product form solutions. This kind of network exists only when it has "M $\Rightarrow$ M", or local balance property [22,38], or when it satisfies the homogeneous assumption [51]. A typical netwrok with local balance property is the one developed by Baskett et al. [14[, which includes four different types of queueing disciplines: FCFS, PS, IS, and LCFS. There are some queueing networks which do not have the "local balance" property. One example is the network with general service time distribution at the service station and FCFS queueing discipline. Since a general service time distribution is represented by a series of stages at this station, and if the queueing discipline is FCFS, a customer waiting at the head of the line is not allowed to enter the first stage until the customer currently in service completes its last stage and departs from this service station. The entrance stage is considered blocked when a customer is still in some stage of this service station.

We need other approaches to solve the queueing networks which do not have product form solutions. Three different approaches can be considered:

1. Some numerical methods.
2. Some approximation methods.
3. Simulation.

Numerical methods may give an exact solution for a general queueing network, but they will have difficukty in handling a very large system. Approximation methods and simulation can handle any large and comples queueing networks. But since simulation is not an anaytical method, we do not want to discuss it. A discussion of simulation methods can be seen in [5]. Regenerative method has recently been developed for estimating confidence intervals when simulating a queueing system. An introduction to regenerative method is given by Lavenberg and Slutz [147,156]. Iglehart gives a thorough survey of this method [151]. One computer program package, APLOMB [69], is used for simulation of general networks. Written in FORTRAN, it is the simulation component of RESQ [174].

In this chapter, we discuss some numerical methods and approximation methods. The section 4-1, we discuss some numerical methods which are used to obtain the stationary probability vector of a Markovian model. These numerical methods include some matrix iteration methods: power method, lopsided metnod, Jacobi method, Gauss-Seidel method, and successive overrelaxation method, and two other recursive methods: one developed by Herzog, Woo, Chandy [87], the other by Brandwajn [81]. In section 4-2, we discuss two major approximation methods: diffusion and aggregation.

4-1 Numerical Methods

If a queueing system can be modeled by a continuous time Markov chain, then numerical methods may be used to obtain the stationary (or long-run) probability vector of the system, the

vector whose length is equal to the number of states which the system can occupy, and whose ith component denotes the probability of the system being in state i after a long period of time which is independent of the initial state. From such stationary probabilities, the performance values may be derived.

Consider a queueing network which is modeled by a continuous time Markov chain with discrete state space. Let $P_i(t)$ be the probability that the system is in state i at time t, then $P_i(n)$ can be expressed in the form of a Chapman-Kolmogorov equation [250]

$$P_i(t+\Delta t) = P_i(t) \left\{ 1 - \sum_{\substack{j=1 \\ j \neq i}}^{n} s_{ij} \Delta t + \sum_{\substack{k=1 \\ k \neq i}}^{n} s_{ki} P_k(t) \Delta t \right\}$$

where n is the total number of states, and s_{ij} is the rate of transition from state i to station j. Let $s_{ii} = - \sum_{\substack{j=1 \\ j \neq i}}^{n} s_{ij}$, then

$$P_i(t+\Delta t) = P_i(t) + \sum_{k=1}^{n} s_{ki} P_k(t) \Delta t$$

$$\lim_{\Delta t \to 0} \frac{P_i(t+\Delta t) - P_i(t)}{\Delta t} = \frac{d}{dt} P_i(t) = \sum_{k=1}^{n} s_{ki} P_k(t)$$

In matrix notation

$$\frac{d}{dt} \underline{P}(t) = \underline{s}^T \underline{P}(t)$$

At steady state, the rate of change of $\underline{P}(t)$ is zero, and therefore

$$\underline{s}^T \underline{P} = \underline{0}$$

where $\underline{P}(t)$ is now written as $\underline{P}$. Therefore, some numerical

methods can be applied to solve this matrix equation in order to get $\underline{P}$.

Direct methods and iterative methods are two different kinds of numerical methods for solving simultaneous linear equations. When a direct method like Gauss elimination method is used , the solution will be yielded after an amount of computation that can be specified in adveance. In contrast, an iterative method is one in which we start from an approximate solution iteratively until the required accuracy is achieved.

When numerical methods are used to solve the above matrix equation, iterative methods are generally preferred to a direct method because the matrix involved is usually large and sparse, i.e. having few nonzero elements. The only operation in which the matrix $\underline{s}$ is involved is a multiplication with one or more vectors in a iterative method. The operation does not change the the form of the matrix, and thus compact storage schemes may be used. Such schemes store only the nonzero elements of the matrix and the position of these elements in the array, and consequently a considerable saving in core requirements is effected. On the other hand, direct methods will cause the form of the matrix to be changed, creating nonzero elements in positions which were previously empty. However, iterative methods also have a major disadvantage in that they often require a very long time to converge to the desired solution compared with direct methods.

In this section, we first discuss some iterative methods [92,97] for solving this matrix equation. Then a direct method [94] developed by Stewart is also described. A comparison of these numerical methods can be seen in [93]. Finally, two numerical methods with other approaches are described.

Power method and lopsided method

The power is used to solve the eigenvalue problem. This method is the first numerical approach to analyze the queueing network models [97]. The computer program package RQA-1 (Recursive Queue Analyzer) [176] was developed based on the power method. This method involves repeatedly premultiplying an arbitrary trial vector by the matrix until the results obtained from consecutive iterations become proportional to one another.

Power method can be investigated as follows: the equation

$$\underline{s}^T \underline{P} = \underline{0}$$

can be transformed to

$$\underline{P} = \delta \underline{s}^T \underline{P} + \underline{P} = (\delta \underline{s}^T + \underline{I}) \underline{P}$$

where δ is a scalar multiplier. This in turn suggests the iteration

$$\underline{P}_{n+1} = (\delta \underline{s}^T + \underline{I}) \underline{P}_n$$

It has been shown that $\underline{P}$ will be assured to converge onto unique vector when δ is chosen such that $\delta \leq (\max_i |s_{ii}|)^{-1}$ where s_{ii} is a diagonal element of $\underline{s}$ [98]. This can also be considered

as an eigenvalue problem in which $\underline{P}$ is presented as an eigenvector [92]. There is only one dominant eigenvector in the power method. Thus, sometimes the rate of convergence is very slow [93].

This problem can be overcome to a certain extent by using simultaneous iterative methods. Such methods are an extension of the power method in which iterative is carried out with a eigenvectors and yield the dominant eigenvalues [93]. The simultaneous iterative methods have been most highly developed for the real symmetric eigenvalue problem. There are also some simultaneous iterative methods for unsymmetric cases [239]. The lopsided method developed by Jennings and Stewart [89] is a simultaneous iterative method for unsymmetric cases. This method determines only one set of eigenvectors for a real unsymmetric matrix, thus considerably reducing core requirements [93]. Therefore the lopsided method is well suited to analyzing the Markovian problem. One computer program package, MARCA (MARkov Chain Analyzer), has been developed based on this method [91,175].

Jacobi, Gauss-Seidel, and successive overrelaxation methods

In the above section, either the power method or lopsided method is used to determine the stationary probability vector $\underline{P}$ from the equation $\underline{P} = (\underline{s}^T + \underline{I})\ \underline{P}$. Stewart [92] proposed an alternative approach to obtain $\underline{P}$ from the homogeneous system of the linear equation: $\underline{s}^T\ \underline{P} = \underline{0}$. This approach lets us use some standard and well known iterative methods are Jacobi method, Gauss-Seidel method, and successive overrelaxation method.

Consider the equation: $\underline{s}^T \underline{P} = \underline{0}$; let

$$\underline{s}^T = (\underline{L} + \underline{D} + \underline{U})$$

where $\underline{D}$ is a diagonal matrix and $\underline{L}$ and $\underline{U}$ are, respectively, lower and upper triangular matrices. Then

$$(\underline{L} + \underline{D} + \underline{U})\ \underline{P} = \underline{0}$$

$$\underline{D}\ \underline{P} = -(\underline{L} + \underline{U})\ \underline{P}$$

Assume that $\underline{D}$ is nonsingular; this yields

$$\underline{P} = -\underline{D}^{-1}(\underline{L} + \underline{U})\ \underline{P}$$

The matrix $-\underline{D}^{-1}(\underline{L} + \underline{U})$ is just the iterative matrix for the Jacobi method. Then $\underline{P}$ can be obtained from this iterative method.

Similar results may be derived for the Gauss-Seidel method. From

$$(\underline{L} + \underline{D} + \underline{U})\ \underline{P} = \underline{0}$$

we have

$$(\underline{L} + \underline{D})\ \underline{P} = -\underline{U}\ \underline{P}$$

Assume that $(L + D)$ is nonsingular; it yields

$$\underline{P} = -(\underline{L} + \underline{D})^{-1}\ \underline{U}\ \underline{P}$$

This matrix $-(\underline{L} + \underline{D})^{-1}\ \underline{U}$ is just the iterative matrix for the Gauss-Seidel method, and $\underline{P}$ can be also be obtained.

Finally, for the successive overrelaxation method, we begin

with

$$w\,(\underline{L} + \underline{D} + \underline{U})\,\underline{P} = \underline{0}$$

where w is the relaxation factor. Then

$$(w\,\underline{U} + w\,\underline{D})\,\underline{P} = -\,w\,\underline{L}\,\underline{P}$$

$$(w\,\underline{U} + w\,\underline{D})\,\underline{P} - \underline{D}\,\underline{P} = -\,w\,\underline{L}\,\underline{P} - \underline{D}\,\underline{P}$$

$$[w\,\underline{U} + (w - 1)\,\underline{D}]\,\underline{P} = -\,(w\,\underline{L} + \underline{D})\,\underline{P}$$

$$\underline{P} = -\,(w\,\underline{L} + \underline{D})^{-1}\,[w\,\underline{U} + (w - 1)\,\underline{D}]\,\underline{P}$$

This is the iterative equation used in the successive overrelaxation iterative method. Note that when w = 1, it will be the same as Gauss-Seidel method.

It has been shown that the Jacobi iterative method will give poor results compared with power method, while the Gauss-Seidel and successive overrelaxation methods permit numerical solutions to be obtained very rapidly and should be used whenever possible [92].

Direct method

A direct method was proposed by Stewart [94]. Instead of the usual direct methods, this approach used the method of inverse iteration; the latter requires less numerical computation. A fixed bandwidth storage scheme is also recommended to handle the core memory problem. Note that although it requires more memory for storing arrays, the direct method obtains much more accurate results in a considerably shorter time period.

Other numerical methods

Two other numerical methods which may efficiently solve queueing problems have been developed. One is a recursive method proposed by Herzog, Woo, and Chandy [87]. Taking advantage of the special recursive structure of the systems of equations, this method is easy to program and has less computing time and/or memory compared with some numerical methods. All those systems of equations which are described by means of Chapman-Kolmogoroff stationary equations have the folowing typical feature: there exists a subset of the state probabilities, which we define as boundaries, and if the values of the boundaries are known, the recursive of the total system of equations can be carried out efficiently [87]. Therefore, using this feature, this recursive method first determines the boundaries and derives expressions for all remaining state probabilities as functions of the boundary values. Then it solves a reduced system of equations for these boundaries. Finally, it determines all interesting state probabilities and performance values by means of the boundaries. When solving the reduced system of equations, some common techniques such as matrix inversion can be applied.

Another numerical method was proposed by Brandwajn [81]. This method is based on a systematic use of the notion of equivalence, and of conditional probability distributions. In most cases, it implies an iterative scheme. The computation involved at each iteration is simple and does not require any matrix operation; sufficient convergence conditions have been

obtained, and the rate of convergence appears to be good.

4-2 Approximation Methods

There is a limitation for numerical methods which can give an exact solution for a queueing network. Since the numerical methods are trying to solve the steady-state balance equations, they often require a prohibitively large core memory to store the transition matrix, or a long time to obtain the solution. This is due to the fact that some numerical methods need to solve the balance equations directly, and some need a large number of iterations to converge. Therefore, instead of numerical methods, we usually use approximation methods to handle more complex queueing networks. Approximation methods approximately at relatively low cost, i.e. taking less compute core memory or/ and computation time. Two major approximation methods, diffusion and aggregation, are discussed in this section.

4-2.1 Diffusion Approximation

The diffusion approximation is based on the assumption that queues are almost always nonempty (i.e. heavy traffic conditions). The central limit theorem is applied to characterize the fluctuations in the queue lengths, and then the discrete-state queueing process is replaced by a continuous time Markov process (also called a diffusion process). The probability distribution of this continuous process is described by a diffusion equation, which is in the form of partial differential equations. These equations can be solved with appropriate boundary conditions. The diffusion approximation was originally used to solve a single

queueing system with G/G/1 (i.e. general input process, general service time distribution, and one service) [117]. Gaver and Shelder [106] used diffusion approximation to evaluate the CPU utilization in a multiprogramming system which is represented by a cyclic queueing model. Kobayashi [118] extended the diffusion approximation to solve the general queueing networks. Furthermore, a transient solution of a queueing network can also be obtained via diffusion approximation [119]. Gelenbe [107] proposed different boundary conditions to improve diffusion approximation. Reiser and Kobayashi [121], and Badel and Shum [102] have assessed and evaluated the accuracy of diffusion approximation to queueing systems. Some application of diffusion approximation can be seen in [105,109,110]. In this section, we discuss some basic ideas in diffusion approximation.

The basic problems involved in diffusion approximation are (1) the choice of the mean and variance which are the parameters for characterizing the diffusion process, (2) the choice of the appropriate boundary conditions, and (3) the selection of intervals needed to approximate queue length distribution from continuous-path process. Consider a single queueing system with G/G/1. Let Q(t) be the queue length at time t. The interarrival time and service time are both represented by their means and variances, namely

$\frac{1}{u_a}$ = mean interarrival time

σ_a^2 = variance of interarrival time

$\frac{1}{u_s}$ = mean service time

σ_s^2 = variance of service time

Q(t) will not become zero since it is a heavy traffic condition. Then on the basis of the "cantral limit theorem", the change in queue length $\Delta Q(t) = Q(t+\Delta t) - Q(t)$ can be approximately normally distributed [118] with mean

$$E[\Delta Q(t)] = (u_a - u_s)\,\Delta t = \beta\,\Delta t$$

with variance

$$Var[\Delta Q(t)] = (C_a\,u_a + C_s\,u_s)\,\Delta t = \alpha\,\Delta t$$

where $C_a = u_a^2\,\sigma_a^2$ and $C_s = u_s^2\,\sigma_s^2$, i.e. the squared coeffients of variation.

Therefore, we can approximate a discrete-state process Q(t) by a continuous path process X(t), whose incremental change

$$dX(t) = X(t+dt) - X(t)$$

is normally distributed with mean dt and variance dt. Let $P(X_0, X, t)$ be the probability density function of X(t) given that its initial value $X(t) = X_0$. It can be shown that $P(X_0, X, t)$ satisfies the equation [118]

$$\frac{\partial}{\partial t} P(X_0, X, t) = \frac{\alpha}{2} \frac{\partial}{\partial X^2} P(X_0, X, t) - \beta \frac{\partial}{\partial x} P(X_0, X, t)$$

This equation is called the Kolmogorov diffusion equation or Fokker-Planck equation.

The diffusion equation is now solved with appropriate boundary conditions. Since the queue length cannot be negative, the solution must satisfy the boundary condition

$$P(X_0,X,t) = 0 \quad \text{for } X < 0$$

The natural way to handle this condition will be to treat $X = 0$ as a reflecting barrier. Since a general distribution has a different coefficient, a simple reflecting barrier tends to bring considerable error in the solution of queue length. Therefore, there are several different ways to modify the boundary condition [106,107,118,121].

After obtaining $P(X_0,X,t)$, we need to approximate the queue length distribution $P(n_0,n,t)$ by integrating $P(X_0,X,t)$ over a selected interval. A reasonable heuristic one is a unit interval $n \leq X < n + 1$. In steady state, we use the equilibrium queue length distribution $P(n)$ instead of $P(n_0,n,t)$, or

$$P(n) = P(n_0,n,t)$$

From the integration, we can get

$$P(n) = \int_n^{n+1} P(X_0,X,t) = (1 - \hat{\rho}) \hat{\rho}^n \quad \text{for } n = 0, 1, 2, \ldots$$

where $\hat{\rho} = \exp[- 2 (1 - \rho) / (C_s + C_a \rho)]$, and ρ is utilization.

For a queueing network, the queueing processes can be approximated by a vector valued diffusion process. The interactions among different queueing procedded are explicitly

considered in the diffusion equations in terms of the variance-covariance matrix. Kobayshi [118] has derived the joint queue length distribution, which is expressed in a product form of the marginal queue length distributions. That is,

$$P(n_1, n_2, \ldots, n_M) = \frac{1}{\hat{G}} \prod_{i=1}^{M} \hat{P}_i(n_i)$$

where M is the total number of service stations, and $\hat{G}$ is the normalization constant, i.e.

$$G = \sum_{\substack{\text{all feasible} \\ \text{states}}} \prod_{i=1}^{M} \hat{P}_i(n_i)$$

where

$$\hat{P}_i(n) = (1 - \hat{\rho}_i) \hat{\rho}_i^n$$

and

$$\rho_m = \exp(2 \beta_m / \alpha_m)$$

β_m depends on the mean interarrival and service time, and routing probabilities, and α_m depends on the variances of interarrival and service time.

The diffusion approximation is used not only to obtain the equilibrium state solution P(n), but also to get the nonequilibrium solution $P(n_0, n, t)$. Kobayashi [119] showed that the transient solution for a cyclic queueing model can be obtained via diffusion approximation. However, there are cercertain restrictions for diffusion approximation. It requires a nonpreemptive queueing discipline and a heavy load in each station of the network. The latter condition is usually

met in open networks under heavy traffic conditions, but is not the case in closed queueing networks where several stations are often likely to be underutilized. Hence, the diffusion approximation is seldom used for analyzing complex closed queueing networks.

4-2.2 Aggregation

The key concept in aggregation is that one solves portions of the networks in isolation, and replaces each portion by a single composite queue, then analyzes these composite queues to produce a solution of the whole system. This approach has also been referred to as decomposition, since one can view the strategy alternatively as decomposition of the whole system or as aggregation of portions of the system. Aggregation can be shown to give exact solutions and to do parametric analysis for queueing networks with product form solutions [20]. But in general queueing networks, the aggregation will cause some error because it dose not totally capture the interaction between the individual portions. Courtois [129] studied the conditions for determining how and when decomposition is viable, and derived the error bound for some queueing network models. Vantilborgh [141] did a similar study in which aggregation yields exact results for some queueing networks with an exponential, load dependent server. On the other hand, aggregations are also studied by means of flow-equivalent approximation. Flow-equivalent methods rely mostly on intuitive heuristics, but they have proved to be usefully accurate in a number of particular applications. An introduction of flow-

equivalent methods is given by Chandy and Sauer [104]. To reduce the error in aggregation, a lot of methods which will improve the flow-equivalence are developed. Iteration [125] and product form [137] are two such methods. Comparison among these methods can be seen in [101] or in [123]. Some other approximation methods using different approaches for open queueing networks may be seen in [134,138]. In this section, we discuss some ideas about flow-equivalent, iteration, and product form methods.

Flow-equivalent method

The equivalent method is a basic approach to replcing a subnetwork of queues by a single composite queue. In this method, the customer flow through the composite queue is equal to the customer flow through the network. For a queueing network with product form solution, flow-equivalent method will give an exact result for a composite queue [20]. To determine the composite queue service time, we first analyze the subnetwork isolately. By considering the subnetwork with the output fed back to the input, we can determine the throughput, X(n), along this feedback path for each possible customer population size n in the subnetwork. Then the mean service time of the composite queue given n customers in the queue, S(n), is set just equal to 1 / X(n) . To illustrate this method, we consider a central server model in Figure 4.1 There are M stations and N customers in this model. Except for station 1, all stations satisfy local balance. First we consider the subsystem in Figure 4.2, and calculate the throughput X(n) for n = 1, 2, ..., N. Then the original network is reduced to a simple network which contains

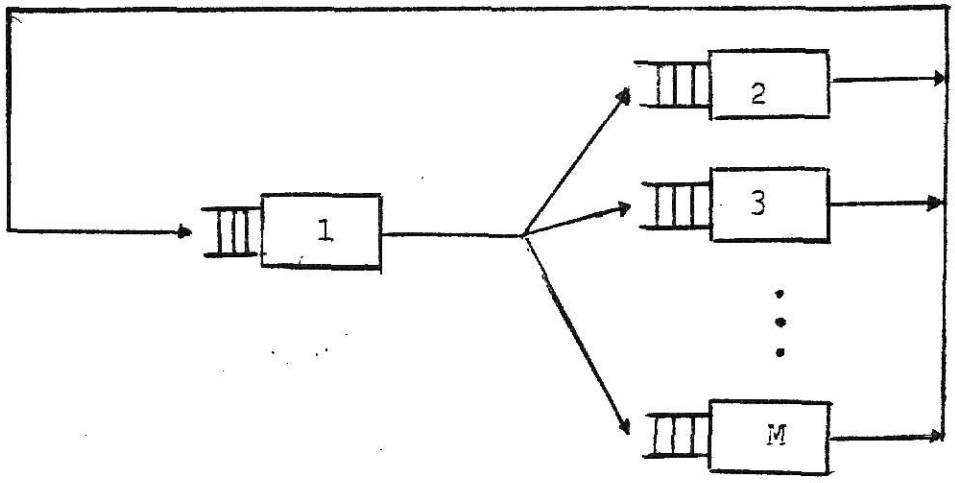


Figure 4.1 A Central Server Model

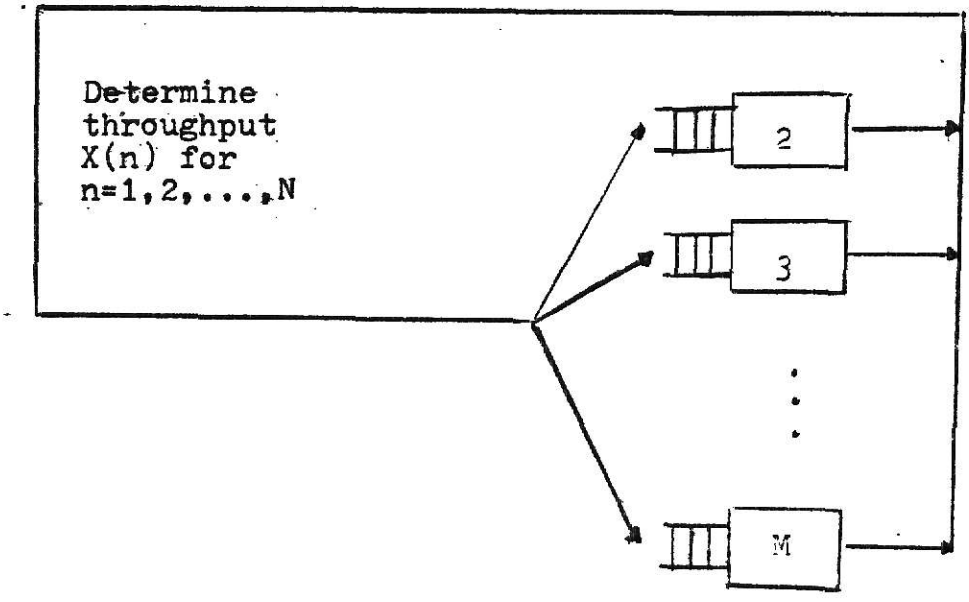


Figure 4.2 A Subsystem of Central Server Model

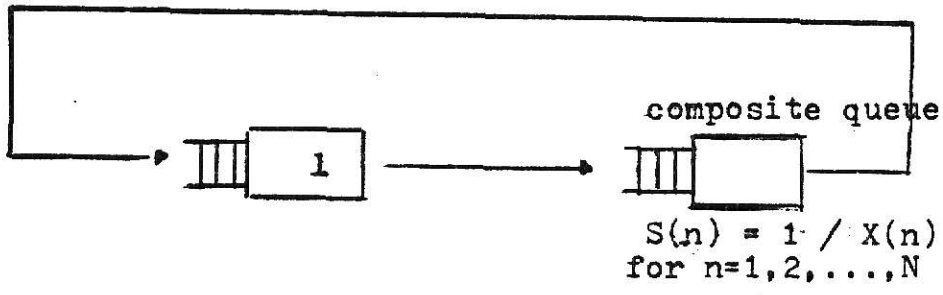


Figure 4.3 A Reduced Network by Replacing Subsystem with a Single Composite Queue

only two queues as shown in Figure 4.3. The mean service time of the composite queue S(n) is equal to 1 / X(n) for n = 1, 2, ..., N. The reduced network can be easily analyzed by any numerical method.

For the subnetwork in which some stations do not satisfy local balance, we need to consider two things: service time distributions and queueing disciplines for the composite queue. Very little work has been done in the area of selecting queueing disciplines, though it does affect the aggregation's results. Usually the queueing disciplines are selected more to reduce computational complexity than to better model the composite system. For service time distribution, we need to determine the mean service time and variance. Mean service time can be determined by throughputs. There are several ways of estimating the throughput in the subnetwork. The better the estimation, the more expensive the method. The most accurate solution is to model this subnetwork as a discrete-state Markov process and then to determine steady-state probabilities numerically, and thus compute the throughputs or mean service time of the flow-equivalent. One easy but not very accurate method is by assuming all stations in the subnetwork have local balance and determine the mean service time directly by the flow-equivalent. The simplest way to determine variances is to assume that the service time of the composite queues are exponential. Sauer and Chandy [135], and Sevick, Levy, Tripathi, and Zahorjan [136] used different approaches to determine the coefficient of variation of the service time for composite queue. More detailed

discussion can be found in [104,136].

Iteration and product form methods

Iteration and product form methods are the methods which try to reduce the error of aggregation from flow-equivalent approximations. Rather than attempting to represent a subnetwork accurately, these two methods carry out the computation assuming simplistic subnetwork representation and later attempt to correct for the inaccuracy in subnetwork representation. In the iteration method [125], the queueing network is first assumed to satisfy local balance. Then in each iteration, we determine the complement of queue for each queue i in the netwok. The complement of queue is a composite queue of subnetwork which is the network excluding queue i. Since the subnetwork satisfies local balance, we can determine service station directly by using flow-equivalence. Then we determine queue length distribution for each queue i by analyzing the two queue network consisting of this original queue and its complement queue. We can use only numerical methods to solve this two queue network. Then we make consistency tests which check whether flow is balanced balanced and whether the mean queue lengths in the queues sum to the total number of customers in the network. If the tests are not satisfied, we adjust the mean service time of the queues, and do the next iteration. Thus queue length distribution for each queue i can be improved iteratively until the tests are satisfied.

In product form method [137], we first assume the queue

length distribution $P(n_1,n_2,\ldots,n_M)$ has a product form which is the products of factors $P_i(n_i)$ for each queue, i.e.

$$P(n_1,n_2,\ldots,n_M) = \frac{1}{G(n)} \prod_{i=1}^{M} P_i(n_i)$$

Each factor $P_i(n_i)$ is analyzed as an M/G/1/N queue isolately. The input rate for each queue i is $e_i X_s$, where e_i is relative throughput to queue i, and X_s is the system throughput. Beginning with an arbitrary X_s, we can find each factor $P_i(n_i)$ independently, and then get the joint queue length distribution $P(n_1,n_2,\ldots,n_M)$. A set of throughputs can be computed from the joint queue length distribution. Then we check whether these throughputs satisfy flow balance. If they do not satisfy, we adjust system throughput X_s, and do the next iteration. The joint queue length distribution can be improved iteratively until a set of throughputs satisfies flow balance.

CHAPTER 5

CONCLUSION

This report gives an overview of queueing networks for modeling computer systems. First, an introduction and calssification of queueing network models is given. The development and derivation of queueing networks which will result in product form solutions is then represented. Finally, computational algorithms and methods for solving queueing networks are discussed. In addition to describing some basic ideas about those different approaches, computational algorithms, and methods, we also give a classified bibliography of research in queueing networks at the end of this chapter.

In summary, there are three different approaches for solving queueing network problems. One is to make some assumptions about the system so as to get a queueing network model which has a product form solution, and then use some computational algorithms to solve it. Although this approach will give an exact solution and can be solved efficiently, it needs some strong assumptions which may severely affect credibility. A second approach is by numerical methods. These methods can also give an exact solution, but they have restrictions about core memory or/and time of convergence. The third approach is by approximation methods. Diffusion approximation will give nearly exact results when applied to a system with heavy traffic conditions. Aggregation approximation is an attractive approach for analyzing complex systems, since

it allows the mixture of various techniques (e.g. product form solution, numerical methods, and simulation).

There are a lot of areas for future research. For product form solutions, one may develop some other computational algorithm with more efficiency, or extend existing algorithms such as Polya enumeration to more general models. For numerical methods, one may find other methods with the fast rate of a direct method. For approximation methods, one may find some ways to improve flow-equivalent methods, or conditions in which aggregation will yield exact results for any general queueing network.

REFERENCES

I. Survey Papers and Books

(1) Disney, R.L., "Dandom Flow in Queueing Networks: A Review and Critique", AIIE Transactions, 7, 3, 1975, pp. 268-288.

(2) Kienzle, M.G., and Sevcik, K.C., "Survey of Analysis Queueing Network Models of Computer Systems", Performance Evaluation Review, 8, 3, 1979, pp. 113-130.

(3) Kleinrock, L., Queueing Systems: Vol. I, Wiley-Interscie Interscience, New York, 1975.

(4) Kleinrock, L., Queueing Systems: Vol. II, Wiley-Interscience, New york, 1976.

(5) Kobayashi, H., Modeling and Analysis: An Introduction to System Performance Evaluation Methodology, Addison-Wesley, Mass., 1978.

(6) Kobayshi, H., "System Design and Performance Analysis Using Analytic Models", in Current Trends in Programming Methodology Vol. III: Software Modeling and its Impact on Performance, K.M. Chandy, and R.T. Yeh (Eds.), Prentice-Hall, Englewood Cliffs, N.J., 1978, pp. 72-114.

(7) Lemoine, A.J., "Networks of Queues - A Survey of Equilibrium Analysis", Management Science, 24, 4, 1977, pp. 464-481.

(8) Lemoine, A.J., "Networks of Queues - A Survey of Weak Convergence Results", Management Science, 24, 11, 1978, pp. 1175-1193.

(9) Makinney, J.M., "A Survey of Analytical Time-Sharing Models", Computing Surveys, 1, 1969, pp. 105-116.

(10) Muntz, R.R., "Queueing Networks: A Critique of the State of the Art and Directions for the Future", Computing Surveys, 10, 3, 1978, pp. 353-359.

(11) Wyszewianki, R.J., and Disney, R.L., "Feedback Queues in The Modeling of Computer Systems: A Survey", Technical Report No. 74-1, Department of Industrial and Operations Engineering, University of Michigan, 1975.

II. Models

II.A Stochastic Analysis

(12) Barbour, A.D., "Networks of Queues and the Method of Stages", Advances in Applied Probability, 8, 1976, pp. 584-591.

(13) Baskett, F., "The Dependence of Computer System Queues upon Proceeding Time Distribution and Central Processor Scheduling", Proceedings of the ACM SIGOPS Third Symposium on Operating System Principles, Stanford University, 1971, pp. 109-113.

(14) Baskett, F., Chandy, K.M., Muntz, R.R., and Papacios, F.G., "Open, Closed, and Mixed Networks of Queues with Different Classes of Customers", Journal of the Association for Computing Machinery, 22, 2, 1975, pp. 248-260.

(15) Baskett, F., and Gomez, F.P., "Processor Sharing in a Central Server Queueing Model", Proceedings of the Sixth Annual Princeton Conference on Information Sciences and Systems, Priceton University, 1972.

(16) Baskett, F., and Muntz, R.R., "Queueing Network Models with Different Classes of Customers", Proceedings of Sixth Annual IEEE Computer Soceity International Conference, 1972, pp. 428-434.

(17) Baskett, F., and Muntz, R.R., "Networks of Queues", Proceedings of the Seventh Annual Princeton Conference on Information Sciences and Systems, Princeton University, 1973, pp. 428-434.

(18) Burke, P.J., "The Output of a Queueing System", Operations Research, 4, 1956, pp. 699-704.

(19) Chandy, K.M., "The Analysis and Solutions for General Queueing Networks", Proceedings of the Sixth Annual Princeton Conference on Information Sciences and Systems, Princeton University, 1972, pp. 224-228.

(20.a) Chandy, K.M., Herzog, U., and Woo, L., "Parametric Analysis of Queuing Networks", IBM Journal of Research and Development, 19, 1975, pp. 36-42.

(21) Chandy, K.M., Howard, J., Keller, T.W., and Towsley, D.J., "Local Balance, Robustness, Poisson Departures and the Product Form in Queuing Networks", Department of Computer Sciences, TR-15, the University of Texas at Austin, 1973.

(22) Chandy, K.M., Howard, J.H., and Towsley, D.F., "Product Form and Local Balance in Queueing Networks", Journal of the Association for Computing Machinery, 24, 2, 1977, pp. 250-263.

(23) Chang, A., and Lavenberg, S., "Work rates in Closed Queueing Networks with General Independent Servers", Operations Research, 22, 4, 1974, pp. 838-847.

(24) Cox, D.R., "A Use of Complex Probabilities in the Theory of Stochastic Processes", Proceedings of Cambridge Philosophical Society, 51, 1955, pp. 313-319.

(25) Daley, D.J., "Queueing Output Processes", Advances in Applied Probability, 8, 1976, pp. 395-415.

(26) Daley, D.J., and Vere-Jones, D., "A Summary of the Theory of Point Processes", in Stochastic Point Processes: Statistical Analysis, Theory, and Applications, John Wiley and Sons, New York, 1972.

(27) Gordon, W.J., and Newell, G.F., "Cyclic Queueing Systems with Restricted Queue Lengths", Operations Research, 15, 2, 1967, pp. 266-277.

(28) Gordon, W. J., and Newell, G.F., "Closed Queuing Systems with Exponential Servers", Operations Research, 15, 1967, pp. 254-265.

(29) Jackson, J.R., "Networks of Waiting Lines", Operations Research, 5, 1957, pp. 518-521.

(30) Jackson, J.R., "Jobshop-Like Queueing Systems", Management Science, 10, 1, 1963, pp. 131-142.

(31) Kelly, F.P., "Networks of Queues with Customers of Different Classes", Journal of Applied Probability, 12, 1975, pp. 542-554.

(32) Kelly, F.P., "Networks of Queues", Advances in Applied Probability, 8, 1976, pp. 416-432.

(33) Lam, S.S., "Queueing Networks with Population Size Constraints", IBM Journal of Research and Development, 21, 1977, pp. 370-378.

(34) Lemoine, A.J., "On total Sojourn Time in Networks of Queues", Management Science, 25, 10, 1979, pp. 1034-1035.

(35) Little, J.D.C., "A Proof for the Queuing Formula: $L = \lambda W$", Operations Research, 9, 1961, pp. 383-387.

(36) Mitran, I., "A Critical Note on a Result by Lemoine", Management Science, 25, 10, 1979, pp. 1026-1027.

(37.a) Moore, C.G., III, "Network Models for Large Scale Time Sharing Systems", Technical Report No. 71-1, Department of Industrical Engineering, University of Michigan, Ann Arbor, Michigan, 1971.

(38) Muntz, R.R., "Poisson Departure Processes and Queueing Networks", IBM Research Report RC-4145, 1972.

(39) Muntz, R.R., and Baskett, F., "Open, Closed, and Mixed Networks of Queues with Different Classes of Customers", Technical Report No. 33, Stanford Electronics Laboratories, Stanford University, 1972.

(40) Posner, M., and Bernholtz, B., "Closed Finite Queueing Networks with Time Lags", Operations Research, 16, 1968, pp. 962-976.

(41) Posner, M., and Bernholtz, B., "Closed Finite Queueing Networks with Time Lags and with Several Classes of Unites", Operations Research, 16, 1968, pp. 977-985.

(42) Samelson, C.L., "Product Form Solution for Queuing Networks with Poisson Arrivals and General Service Time Distribution with Finite Mean", Ph.D. Thesis, Mathematical Department, University of Kansas, 1978.

(43.a) Sevcik, K.C., and Klawe, M.M., "Operational Analysis versus Stochastic Modeling of Computer Systems", Proceedings of Computer Science and Statistics: 12th Annual Symposium on the Interface, University of Waterloo, 1979.

(44) Simon, B., and Foley, R.D., "Some Results on Sojourn Times in Acyclic Tackson Networks", Management Science, 25, 10, 1979, pp. 1027-1034.

II.B Operational Analysis

(45) Buzen, J.P., "Fundamental Laws of Computer System Performance", in International Symposium on Computer Performance Modeling, Measurement, and Evaluation, Cambridge, Mass., 1976, pp. 200-210.

(46) Buzen, J.P., "Fundamental Operational Laws of Computer System Performance", Acta Informtrica, 7, 2, 1976, pp.167-182.

(47) Buzen, J.P., "Operational Analysis: The key to the New Generation of performance prediction Tools", Proceedings of IEEE Compcon., 1976.

(48) Buzen, J.P., "Operational Analysis: An Alternative to Stochastic Modeling", Technical Report, BGS Systems, Inc., Box 128, Lincoln, MA 01773, 1976.

(49) Denning, P.J., and Buzen, J.P., "An Operational Overview of Queueing Networks", in Infotech State of the Art Report on Performance Modeling and Prediction, Infotech Int. Ltd., Maidenhead, UK, 1977, pp. 75-108.

(50) Denning, P.J., and Buzen, J.P., "Operational Analysis of Queueing Networks", in Measuring, Modeling, and Evaluating Computer Systems, H. Beilner, and E. Gelenbe (Eds.), North-Holland, New York, 1977, pp. 151-172.

(51) Denning, P.J., and Buzen, J.P., "The Operational Analysis of Queueing Network Models", Computing Surveys, 10, 3, 1978, pp. 225-261.

(52.a) Denning, P.J., and Buzen, J.P., "An Operational Treatment of Queueing Distributions and Mean Value Analysis", Technique Report CSD-TR 309, Purdue University, 1979.

(53) Roode, J.D., "Multiclass Operational Analysis of Queueing Networks", in Performance of Computer Systems, M. Arato, A. Butrimenko, and E. Gelenbe (Eds.), North-Holland, New York, 1979, pp. 339-352.

(43.b) Sevcik, K.C., and Klawe, M.M., "Operational Analysis versus Stochastic Modeling of Computer Systems", Proceeding of Computer Science and Statistics: 12th Annual Symposium on the Interface, University of Waterloo, 1979.

III. Methods

III.A Computational Algorithms for Product Form Solution

III.A.1 Convolution Method

(54) Balbo, G., Bruell, S.C., and Schwetman, H.D., "Customer Classes and Closed Netwrok Models - A Solution Technique", in Proceedings IFIP Congress '77, North-Holland, New York. pp. 559-564.

(55.a) Bruell, S.C., "On Single and Multiple Job Class Queueing Network Models of Computing Systems", Ph.D.Thesis, Computer Science Department, Purdue University, 1978.

(56.a) Bruell, S.C., and Balbo, G., Computational Algorithms for Closed Queueing Networks, North-Holland, New York, 1980.

(57.a) Buzen, J.P., "Queueing Netwrok Models of Multiprogramming", Ph.D. Thesis, Division of Engineering and Applied Science, Harvard University, Cambridge, Mass., 1971.

(58) Buzen, J.P., "Computational Algorithms for Closed Queueing Netwroks with Exponential Servers", Communications of the Association for Computing Machinery, 10, 9, 1973, pp. 527-531.

(59.a) Chandy, K.M., and Sauer, C.H., "Computational Algorithms for Product Form Queueing Networks", Communications of the Association for Computing Machinery, 23, 10, 1980, pp. 573-583.

(60) Muntz, R.R., and Wong, J., "Efficient Computational Procedures for Closed Queueing Network Models", Proceedings of the Seventh Hawaii International Conference on System Sciences, Honolulu, Hawaii, 1974, pp. 33-36.

(61) Reiser, M., "Numerical Methods in Separable Queueing Network", IBM Research Report, Yorktown Heights, N.Y., 1976

(62.a) Reiser, M., "Mean Value Analysis and Convolution Method for Queue-dependent Servers in Closed queueing Networks", Rep. RZ-1009, IBM Zurich Research Center, Zurich, Switzerland, 1980.

(63) Reiser, M., and Kobayashi, H., "Recursive Algorithm for General Queueing Networks with Exponential Servers", IBM Research Report RC-4254, Yorktown Heights, New York, 1973.

(64) Reiser, M., and Kobayashi, H., "Queuing Networks with Multiple Closed Chains: Theory and Computational Algorithms", IBM Journal of Research and Development, 19, 1975, pp. 283-294.

(65) Reiser, M., and Kobayashi, H., "Numerical Methods in Queueing Networks", Proceedings of Computer Science and Statistics, Eighth Annual Symposium on the Interface, UCLA, 1975.

(66) Reiser, M., and Kobayashi, H., "Horner's Rule for the Evaluation of General Closed Queuing Networks", Communications of the Association for Computing Machinery, 18, 10, 1975, pp. 592-593.

(67) Reiser, M., and Kobayashi, H., "On the Convolution Algorithm for Separable Queuing Networks", in International Symposium on Computer Performance Modeling, Measurement, and Evaluation, Cambridge, Mass., 1976, pp. 109-117.

(68) Reiser, M., and Sauer, C.H., "Queueing Network Models: Methods of Solution and their Program Implementation", in Current Trends in Programming Methodology Vol. III: Software Modeling and its Impact on Performance, K.M. Chandy, and R.T. Yeh (Eds.), Prentice-Hall, Englewood Cliffs, N.J., 1978, pp. 115-167.

III.A.2 Mean Value Analysis

(69) Bard, Y., "Some Extensions to Multiclass Queueing Network Analysis", in Performance of Computer Systems, M. Arato, A. Butrimenko, and E. Gelenbe (Eds.), Norht-Holland, New York, 1979, pp. 51-62.

(56.b) Bruell, S.C., and Balbo, G., Computational Algorithms for Closed Queueing Network, North-Holland, New York, 1980.

(59.b) Chandy, K.M., and Sauer, C.H., "Computational Algorithms for Procuct Form Queueing Networks", Communications of the Association for Computing Machinery, 23, 10, 1980, pp. 573-583.

(52.b) Denning, P.J., and Buzen, J.P., "An Operational Treatment of Queueing Distributions and Mean Value Analysis", Technique Report CSD-TR 309, Purdue University, 1979.

(70) Reiser, M., "Mean Value Analysis of Queuing Networks. A New Look at an Old Problem", in Performance of Computer Systems, M. Arato, A. Butrimanko, and E. Gelenbe (Eds.), North-Holland, New York, 1979, pp. 63-77.

(62.b) Reiser, M., "Mean Value Analysis and Convolution Method for Queue-dependent Servers in Closed Queueing Networks", Report RZ-1009, IBM Zurich Research Center, Zurich, Switzerland, 1980.

(71) Reiser, M., and Lavenberg, S.S., "Mean Value Analysis of Closed Multichain Queuing Networks", Journal of the Association for Computing Machinery, 27, 2, 1980, pp. 313-322.

III.A.3 Other Methods

(59.c) Chandy, K.M., and Sauer, C.H., "Computational Algorithms for Porduct Form Queueing Networks", Communications of the Association for Computing Machinery, 23, 10, 1980, pp. 573-583.

(72) Gelenbe, E., and Muntz, R.R., "Probability Models of Computer Systems I: Exact Results", Acta Informatrica, 7, 1, 1976. pp. 35-60.

(73.a) Grillo, D., "Implemantation of Algorithms for Performance Analysis of a Class of Multiprogrammed Computers", in Measuring, Modeling, and Evaluating Computer Systems, H. Beilner, and E. Gelenbe (Eds.), North-Holland, New York, 1977, pp. 131-150.

(74) Grillo, D., "Use of Generating Functions for Performance Analysis of a Class of Multiprogrammed Computers", Foudazione Ugo Bordoni Monograph, IIIEz1, 1977.

(75) Kobayashi, H., "A Computational Algorithm for Queue Distributions via the Polya Theory of Enumeration", in Performance of Computer Systems, M. Arato, A. Butrimenko, E. Gelenbe (Eds.), North-Holland, New York, 1979, pp. 79-88.

(76) Lam, S.S., "An Extension of Moore's Result for Closed Queuing Networks", IBM Journal of Research and Development, 21, 1977, pp. 384-387.

(77) Moore, F.R., "Computational Model of a Closed Queuing Network wtih Exponential Servers", IBM Journal of Research and Development, 21, 1972, pp. 567-572.

(78) Reynolds, P.F., "Queueing Network Algorithms on Programmable Pocket Calculators", Technical Report, Department of Computer Science University of Texas at Austin.

(79) Zahorjan, J., "An Exact Solution Method for the General Class of Closed Separable Queueing Networks", Performance Evaluation Review, 8, 3, 1979, pp. 107-112.

(80) Zahorjan, J.L., "Computational Algorithms for Queueing Networks with Product Form Solutions", in Topics in Performance Evaluation, G.S. Graham (Eds.), CSRG-100, Computer Systems Research Group, University of Toronto, Canada, 1979.

III.B Numerical Methods

(81) Brandwajn, A., "A Model of a Time Sharing System Solved Using Equivalence and Decomposition Methods", Acta Informatrica, 4, 1, 1974, pp. 11-47.

(82) Carrol, J.L., "A Study of Closed Queueing Networks with Population Size Constraints", Ph.D. Thesis, Department of Computer Science, University of Nebrasks at Lincoln, 1979.

(83) Carrol, J.L., and Lispsky, L., "A Recursive Method for Solving Closed Queueing Systems with Population Size Constraints", ACM 79 Computer Science Conference, Dayton, Ohio, 1979.

(84) Disney, R.L., "A Matrix Solution for the Two Server Queue with Overflows", Management Science, 19, 1972, pp. 254-265.

(85) Gaver, D.P., and Humfeld, G., "Multiprogramming: Probability Models and Numerical Procedures", in International Symposium on Computer Performance Modeling, Measurement, and Evaluation, Cambridge, Mass., 1976, pp. 38-43.

(86) Georgunas, N., "Numerical Solution of Queueing Networks with Multiple Semiclosed Chains", Proceedings of the Institution of Electrical Engineers, 126, 1979, pp. 229-231.

(87) Herzog, U., Woo, L., and Chandy, K.M., "Solution of Queuing Problems by a Recursive Technique", IBM Journal of Research and Development, 19, 1975, pp. 295-300.

(88) Irani, K.B., and Wallace, V.L., "A System for the Solution of Simple Stochastic Networks", Technical Report No. 31, Systems Engineering Laboratory, University of Michigan, Ann Arbor, 1969.

(89) Jennings, A., and Stewart, W.J., "Simultaneous Iteration for Partial Engen Solution of Real Matrices", Journal of the Institute of Mathematics and its Applications, 15, 1975, pp. 351-361.

(90.a) Marie, R., and Stewart, W.J., "A Hybrid Iterative-Numerical Method for the Solution of a General Queueing Network", in Measuring, Modeling, and Evaluating Computer Systems, H. Beilner, and E. Gelenbe (Eds.), Norht-Holland, New York, 1977, pp. 173-188.

(91.a) Stewart, W.J., "Practical Considerations in the Numerical Analysis of Markovian Models", in Measuring, Modeling, and Evaluating Computer Systems, H. Beilner, and E. Gelenbe (Eds.), North-Holland, New York, 1976, pp. 363-376.

(92) Stewart, W.J., "A New Approach to the Numerical Analysis of Markovian Models", in Computer Performance, K.M. Chandy, and M. Reiser (Eds.), North-Holland, New York, 1977, pp. 279-295.

(93) Stewart, W.J., "A Comparison of Numerical Techniques in Markov Modeling", Communications of the Association for Computing Machinery, 21, 2, 1978, pp. 144-152.

(94) Stewart, W.J., "A Direct Numerical Method for Queueing Networks", in Performance of Computer Systems, M. Arato, A. Butrimanko, and E. Gelenbe (Eds.), North-Holland, New York, 1979, pp. 89-104.

(95) Wallace, V.L., "The Solution of Quasi-Birth and -Death Processed Arising from Multiple Access Computer-Systems", Ph.D. Thesis, Department of Electrical Engineering, University of Michigan, Ann Arbor, 1969.

(96) Wallace, V.L., "Toward an Algebraic Theory of Markovian Networks", Proceedings of Symposium Computer-Communications Networks and Teletraffic, Polytechnic Press, New York, 1973.

(97) Wallace, V.L., "Algebraic Techniques for Numerical Solution of Queueing Networks", in Mathematical Methods in Queueing Theory, Lecture Notes in Economics and Mathematical Systems No. 98, Springer-Verlag, Berlin-New York, 1974, pp. 295-306.

(98) Wallace, V.L., and Rosenberg, R.S., "Markovian Models and Numerical Analysis of Computer Computer System Behavior", Proceedings AFIPS Spring Joint Computer Conference, 28, 1966, pp. 141-148.

(99) Williams, A.C., and Bhandiwad, R.A., "A Generating Function Approach to Queueing Network Analysis of Multiprogrammed Computers", Network, 6, 1, 1976, pp. 1-22.

(100.a) Zarling, R.L., "Numerical Solution of Nearly Decomposable Queueing Networks", Ph.D. Thesis, University of North Carolina, 1976.

III.C Approximation Methods

III.C.1 Diffusion Approximation

(101.a) Balbo, B., "Approximate Solutions of Queueing Network Models of Computer Systems", Ph.D. Thesis, Computer Science Department, Purdue University, 1979.

(102) Badel, M., and Shum, A.V.Y., "Accuracy of an Approximate Computer System Model", in Modeling and Performance Evaluation of Computer System, H. Beilner, and E. Gelenbe (Eds.), North-Holland, New York, 1976, pp. 11-33.

(103) Brice, R.S., and Franta, W.R., "A Closed Cyclic, Two-Stage Multiprogrammed System Model and its Diffusion Approximation Solution", Proceedings of the Second Annual ACM Sigmetrics Symposium on Measurement and Evaluation, Montreal, Canada, 1974, pp. 54-64.

(104.a) Chandy, K.M., and Sauer, C.H., "Approximate Methods for Analyzing Queueing Network Models of Computer Systems", Computing Surveys, 10, 3, 1978, pp. 281-317.

(105) Foschini, G.J., "On heavy Traffic Diffusion Analysis and Dynamic Routing in Packet Switched Networks", in Computer Performance, K.M. Chandy, and M. Reiser (Eds.), North-Holland, New York, 1977, pp. 499-513.

(106) Grave, D.P., and Shedler, G.S., "Processor Utilization in Multiprogramming System via Diffusion Approximations", Operations Research, 21, 1973, pp. 569-571.

(107) Gelenbe, E., "On Approximate Computer System Models", Journal of the Association for Computing Machinery, 22, 2, 1975, pp. 261-269.

(108) Gelenbe, E., and Pujolle, G., "The Behaviour of a Single Queue in a General Queueing Network", Acta Informatrica, 7, 2, 1976, pp. 123-136.

(109) Gelenbe, E., and Pujolle, G., "Probabilistic Models of Computer Systems", in International Symposium on Computer Performance Modeling, Measurement, and Evaluation, Cambridge, Mass., 1976, pp. 118-125.

(110) Gelenbe, E., and Pujolle, G., "A Diffusion Model for Multiple Class Queueing Networks", in Measuring, Modeling, and Evaluating Computer Systems, H. Beilner, and E. Gelenbe (Eds.), North-Holland, New York, 1977, pp. 189-200.

(111) Harrison, J.M., "The Heavy Traffic Approximation for Single Server Queues in Series", Journal of Applied Probability, 10, 1973, pp. 613-629.

(112) Harrison, J.M., "The Diffusion Approximation for Tandem Queues in Heavy Traffic", Technical Report, Department of Operations Research, Stanford University, 1977.

(113) Iglehart, D.L., and Whitt, W., "Multiple Channel Queues in Heavy Traffic I", Advances in Applied Probability, 2, 1970, pp. 150-177.

(114) Iglehart, D.L., and Whitt, W., "Multiple Channel Queues in Heavy Traffic II: Sequences, Networks and Batches", Advances in Applied Probability, 2, 1970, pp. 355-369.

(115) Kennedy, D.P., "Rates of Convergence for Queues in Heavy Traffic I", Advances in Applied Probability, 4, 1972, pp. 357-381.

(116) Kennedy, D.P., "Rates of Convergence for Queues in Heavy Traffic II: Sequences of Queueing Systems", Advances in Applied Probability, 4, 1972, pp. 382-391.

(117) Kingman, J.F.C., "The Heavy Traffic Approximation in the Theory of Queues", in Proceedings of the Symposium on Congestion Theory, W.L. Smith, and W.E. Wilkinson (Eds.), University of North Carnolina, 1965, pp. 137-169.

(118) Kobayashi, H., "Application of the Diffusion Approximation to Queueing Networks, I - Equilibrium Queue Distributions", Journal of the Association for Computing Machinery, 21, 2, 1974, pp. 316-328.

(119) Kobayashi, H,, "Application of the Diffusion Approximation to Queueing Networks II: Nonequilibrium Distributions and Application to Computer Modeling", Journal of the Association for Computing Machinery, 21, 3, 1974, pp. 459-469.

(120) Reiman, M.I., "Queuing Networks in Heavy Traffic", Ph.D. Thesis, Department of Operations Research, Stanford University, 1977.

(121) Reiser, M., and Kobayashi, H., "Accuracy of the Diffusion Approximation for Some Queuing Systems", IBM Journal of Research and Development, 18, 1974, pp. 110-124.

(122.a) Sauer, C.H., and Chandy, K.M., "Approximate Solution of Queueing Models", Computer, 13, 4, 1980, pp. 25-32.

III.C.2 Aggregation

(101.b) Balbo, G., "Approximate Solutions of Queueing Network Models of Computer Systems", Ph.D. Thesis, Computer Science Department, Purdue University, 1979.

(123) Balbo, G., and Denning, P.J., "Homogeneous Approximations of General Queueing Networks", in Performance of Computer Systems, M. Arato, A. Butrimenko, and E. Gelenbe (Eds.), North-Holland, New York, 1979, pp. 353-374.

(124) Brandwajn, A., "An Approach to the Numerical Solution of Some Queueing Problems", in Computer Performance, K.M. Chandy, and M. Reiser (Eds.), North-Holland, New York, 1977, pp. 83-112.

(20.b) Chandy, K.M., Herzog, U., and Woo, L., "Parametric Analysis of Queuing Networks", IBM Journal of Research and Development, 19, 1975, pp. 36-42.

(125) Chandy, K.M., Herzog, U., and Woo, L., "Approximate Analysis of Genaral Queuing Networks", IBM Journal of Research and Development, 19, 1975, pp. 43-49.

(104.b) Chandy, K.M., and Sauer, C.H., "Approximate Methods for Analyzing Queueing Network Models of Computer Systems", Computing Surveys, 10, 3, 1978, pp. 281-317.

(126) Courtois, P.J., "On the Near-Complete-Decomposability of Networks of Queues and of Stochastic Models of Multiprogrammed Computer Systems", Science Report CMU-CS-71-11, Carnegie-Mellon University, 1971.

(127) Courtois, P.J., "Error Analysis in Nearly-Completely Decomposable Stochastic Systems", Econometrica, 43, 1975, pp. 691-709.

(128) Courtois, P.J., "Decomposability, Instabilities, and Saturation in Multiprogramming Systems", Communication of the Computing Machinery, 18, 7, 1975, pp. 371-377.

(129) Courtois, P.J., Decomposability: Queueing and Computer System Applications, Academic Press, New York, 1977.

(130) Courtois, P.J., and Georges, J., "On a Single-Server Finite Queuing Model with State-Dependent Arrival and Service Processes", Operations Research, 19, 1971, pp. 424-435.

(131) Kuehn, P.J., "Approximate Analysis of General Queueing Networks by Decomposition", IEEE Transactions on Communications, Vol. COM-27, 1, 1979, pp. 113-126.

(132) Kuhn, P., "Analysis of Complex Queueing Networks by Decomposition", Proceedings of Eighth International Teletraffic Congress, Melbourne, Australis, 1976.

(133) Marie, R., "Approximate Analytical Method for General Queueing Network", IEEE Transactions on Software Engineering, Vol. SE-5, 1979, pp. 530-537.

(90.b) Marie, R., and Stewart, W.J., "A Hybrid Iterative-Numerical Method for the Solution of a General Queueing Network", in Measuring, Modeling, and Evalusting Computer Systems, H. Beilner, and E. Gelenbe (Eds.), North-Holland, New York, 1977, pp. 173-188.

(134) Pujolle, G., and Soula, C., "A Study of Flows in Queueing Networks and an Approximate Method for Solution", in Performance of Computer Systems, M. Arato, A. Butrimenko, E. Gelenbe (Eds.), North-Holland, New York, 1979, pp. 375-389.

(135) Sauer, C.H., and Chandy, K.M., "Approximate Analysis of Central Server Models", IBM Journal of Research and Development, 19, 1975, pp. 301-313.

(122.b) Sauer, C.H., and Chandy, K.M., "Approximate Solution of Queueing Models", Computer, 13, 4, 1980, pp. 25-32.

(136) Sevcik, K.C., Levy, A.I., Tripathi, S.K., and Zahorjan, J.L., "Improving Approximations of Aggregated Queuing Network Subsystems", in Computer Performance, K.M. Chandy, and M. Reiser (Eds.), North-Holland, New York, 1977, pp. 1-22.

(137) Shum, A.M., and Buzen, J.P., "A Method for Obtaining Approximate Solutions to Closed Queueing Networks with General Service Times", in Measuring, Modeling, and Evaluating Computer Systems, H. Beilner, and E. Gelenbe (Eds.), North-Holland, New York, 1977, pp. 201-220.

(138) Takahashi, Y., Miyahara, H., and Hasegawa, T., "An Approximation Method for Open Restricted Queueing Networks", Operations Research, 28, 3, 1980, pp. 594-602.

(139) Towsly, D., "Queuing Network Models with State-Dependent Routing", Journal of the Association for Computing Machinery, 27, 2, 1980, pp. 323-337.

(140) Tripathi, S.K., "An Approximate Solution Technique for Queueing Network Models of Computer Systems", Ph.D. Thesis, Department of Computer Science, University of Toronto, Canada, 1979.

(141) Vantilborgh, H., "Exact Aggregation in Exponential Queueing Networks", Journal of the Association for Computing Machinery, 25,4, 1978, pp. 620-629.

(100.b) Zarling, R.L., "Numerical Solution of Nearly Decomposable Queueing Networks", Ph.D. Thesis, University of North Carolina, 1976.

III.D Simulation

(142) Crane, M.A., and Iglehart, D.L., "A New Approach to Simulating Stable Stochastic Systems", Proceedings of the 1973 Winter-Simulation Conference, San Francisco, 1973, pp. 264-272.

(143) Crane, M.A., and Iglehart, D.L., "Simulating Stable Stochastic Systems, I: General Multi-Server Queues", Journal of the Association for Computing Machinery, 21, 1, 1974, pp. 103-113.

(144) Crane, M.A., and Iglehart, D.L., "Simulating Stable Stochastic Systems, II: Markov Chains", Journal of the Association for Computing Machinery, 21, 1, 1974, pp. 114-123.

(145) Crane, M.A., and Iglehart, D.L., "Simulating Stable Stochastic Systems, III: Regenerative Processes and Discrete-Event Simulations", Operations Research, 23, 1975, pp. 33-45.

(146) Crane, M.A., and Iglehart, D.L., "Simulating Stable Stochastic Systems, IV: Approximation Techniques", Management Science, 21, 1975, pp. 1215-1224.

(147) Crane, M.A., and Lemoine, A.J., An Introduction to the Regenerative Method for Simulation Analysis, Springer-Verlag, New York, 1977.

(148) Gunther, F.L., and Wolff, R.W., "The Almost Regenerative Method for Stochastic System Simulations", Operations Research, 28, 2, 1980, pp. 375-386.

(149) Iglehart, D.L., "Simulating Stable Stochastic Systems, V: Comparison of Ratio Estimators", Naval Research Logistics Quartly, 22, 1975, pp. 553-565.

(150) Iglehart, D.L., "Simulating Stable Stochastic Systems, VI: Quantile Estimation", Journal of the Association for Computing Machinery, 23, 2, 1976, pp. 347-360.

(151) Iglehart, D.L., "The Regenerative Method for Simulation Analysis", in Current Trends in Programming Methodology Vol. III: Software Modeling and its Impact on Performance, K.M. Chandy, and R.T. Yeh (Eds.), Prentice-Hall, Englewood Cliffs, N.J., 1978, pp. 52-71.

(152) Iglehart, D.L., and Shedler, G.S., "Regenerative Simulation of Response Times in Networks of Queues", Journal of the Association for Computing Machinery, 25, 3, 1978, pp. 449-460.

(153) Iglehart, D.L., and Shedler, G.S., "Simulation of Response Time in Finite-Capacity Open Networks of Queues", Operations Research, 26, 5, 1978, pp. 896-914.

(154) Iglehart, D.L., and Shedler, G.S., Regenerative Simulation of Response Times in Networks of Queues, Springer-Verlag, Berlin-New York, 1980.

(155) Lavenberg, S.S., and Sauer, C.H., "Sequential Stopping Rules for the Regenrative Method of Simulation", IBM Journal of Research and Development, 21, 1977, pp. 545-558.

(156) Lavenberg, S.S., and Slutz, D.R., "Introduction to Regenrative Simulation", IBM Journal of Research and Development, 19, 1975, pp.458-462.

(157) Lavenberg, S.S., and Slutz, D.R., "Regenrative Simulation of a Queuing Model of an Automated Tape Library", IBM Journal of Research and Development, 19, 1975, pp. 463-475.

(158) Moeller, T.L., and Sauer, C.H., "Control Variables in Regenrative Simulation of Queueing Models", IBM Research Report, Yorktown Heights, New York, 1977.

(159) Sauer, C.H., "Simulation Analysis of Generalized Queueing Networks", Proceedings of 1975 Summer Computer Simulation Conference, 1975, pp. 75-81.

(160) Sauer, C.H., "Characterization and Simulation of Queueing Networks", IBM Research Report, Yorktown Heights, New York, 1977.

(161) Sauer, C.H., "Confidence Intervals for Queueing Simulations of Computer Systems, IBM Research Report RC-6669, Yorktown Heights, New York, 1977.

(162) Wang, K.G., "Continuous Simulation of Queueing Network with Transient and Time Varying Conditions", Ph,D. Thesis, Department of Electrical Engineering, State University of New York at Stony Brook, 1979.

III.E Software Packages

(55.b) Bruell, S.C., "On Single and Multiple Job Class Queueing Network Models of Computing Systems", Ph.D. Thesis, Computer Science Department, Purdue University, 1978.

(56.c) Bruell, S.C., and Balbo, G., Computational Algorithms for Closed Queueing Networks, North-Holland, New York, 1980.

(163) Buzen, J.P., et al., "BEST/1 - Design of a Tool for Computer System Capacity Planning", Proceedings of 1978 AFIPS National Computer Conference 47, AFIPS press, Montvale, N.J., pp. 447-455.

(164) Chandy, K.M., Keller, T.W., and Browne, J.C., "Design Automation and Queueing Networks", Proceedings of Ninth Annual Design Automation Conference, 1972, pp. 357-367.

(165) Foster, D.V., McGehearty, P.F., Sauer, C.H., and Waggoner, C.N., "A Language for Analysis of Queueing Models", Department of Computer Science, TR-33, University of Texas at Austin, 1974.

(166) Grillo, D., "CQNA-1, A Package for Analyzing Closed Queueing Networks. Description and User's Guide", Fondazione Ugo Brodoni Monograph, IIIIEz2, 1977.

(74.b) Grillo, D., "Implementation of Algorithms for Performance Analysis of a Class of Multiprogrammed Computers", in Measuring, Modeling, and Evaluating Computer Systems, H. Beilner, and E. Gelenbe (Eds.), North-Holland, New York, 1977, pp. 131-150.

(167) Keller, T.W., "ASQ Use's Manual", TR-27, Computer Science Department, University of Texas at Austine, Texas, 1973.

(168) Levy, A.I., "QSOLVE, A Queueing Network Solution System", Technical Note 6, Computer Systems Research Group, University of Toronto, 1977.

(169) MaNair, E.A., and Sauer, C.H., "Multiple Language (APL and PL/I) Interfaces for Queueing Network Software", IBM Research Report RC-7535, Yorktown Heights, New York, 1979.

(170) Reiser, M., "QNET4 Use's Guide", IBM Research Report RA71, Yorktown Heights, New York, 1975.

(171) Reiser, M., "Modeling of Computer Systems with QNET4", IBM System Journal , 15, 4, 1976, pp. 309-327.

(69.b) Reiser, M., and Sauer, C.H., "Queueing Network Models Methods of Solution and their Program Implementation", in Current Trends in Programming Methodology Vol. III: Software Modeling and its Impact on Performance, K.M. Chandy, and R.T. Yeh (Eds.), Prentice-Hall, Englewood Cliffs, N.J., 1978, pp. 115-167.

(172) Sauer, C.H., and MacNair, E.A., "Queueing Network Soteware for Systems Modeling", Software: Practical and Experience, 9, 5, 1979, pp. 369-380.

(173) Sauer, C.H., and MacNair, E.A., "A Language for Extended Queuing Network Models", IBM Journal of Research and Development, 24, 6, 1980, pp. 747-755.

(175) Stewart, W.J., "MARCA: Markov Chain Analyzer", IRISA Publication Interne N 45, Universite de Rennes, 35031, France.

(91.b) Stewart, W.J., "Practical Considerations in the Numerical Analysis of Markovian Models", in Measuring, Modeling, and Evaluating Computer Systems, H. Beilner, and E. Gelenbe (Eds.), North-Holland, New York, 1976, pp. 363-376.

(176) Wallace, V.L., and Rosenberg, R.S., "The Recursive Queue Analyzer", System Engineering Department Technical Report No. 2, University of Michigan, Ann Arbor, 1960.

IV. Applications

IV.A Computer System

(177) Adiri, I., "Queueing Models for Multiprogrammed Computers", Proceedings of the International Symposium on Computer-Communication Networks and Teletraffic, Polytech Press, Brooklyn, New York, 1972, pp. 441-448.

(178) Allen, A.O., "Queuing Models of Computer Systems", Computer, 13, 3, 1980, pp. 13-24.

(179) Avi-Itzhak, B., and Heyman, D.P., "Approximate Queueing Models for Multiprogramming Computer Systems", Operations Research, 21, 6, 1973, pp. 1212-1230.

(180) Bard, Y., "A Characterization of VM/370 Workloads", in Modeling and Performance Evaluation of Computer Systems, E. Beilner, and E. Gelenbe (Eds.), North-Holland, New York, !976, pp. 35-56.

(181) Bard, Y., "The VM/370 Performance Predictor", Computing Surveys, 10, 3, 1978, pp. 333-342.

(182) Baskett, F., and Palacios, F.G., "Processor Sharing in a Central Server Queueing Model of Multiprogramming with Applications", Proceedings of the Sixth Annual Princeton University, 1972, pp. 598-603.

(183) Bhandiwad, R., and Williams, A., "Queueing Network Models of Computer Systems", Third Texas Conference on Computing Systems, 1974.

(57.b) Buzen, J.P., "Queueing Network Models of Multiprogramming", Ph.D. Thesis, Division of Engineering and Applied Science, Harvard University, Cambridge, Mass., 1971.

(184) Buzen, J.P., "A Queueing Network Model of MVS", Computing Surveys, 10, 3, 1978, pp. 320-331.

(185) Chen, P.P., "Queueing Network Models of Interactive Systems", Proceedings of the IEEE, 63, 5, 1975, pp. 954-957.

(186) Chow, W.M., "Central Server Model for Multiprogrammed Computer System with Different Classes of Jobs", IBM Journal of Research and Development, 19, 1975, pp. 314-320.

(187) Denning, P.J., "Optimal Multiprogrammed Memory Management", in Current Trends in Programming Methodology Vol. III: Software Modeling and its Impact on Performance, K.M. Chandy, and R.T. Yeh (Eds.), Prentice-Hall, Englewood Cliffs, N.J., 1978, pp.298-322.

(188) Grag, U.K., "A Queuing Network Model of Multiprogrammed Time Sharing Virtual Memory System for Performance Evaluation", Ph.D. Thesis, Department of Operations Research, Cornell University, 1977.

(189) Gaver, D.P., "Probability Models for Multiprogramming Computer Systems", Journal of the Association for Computing Machinery, 14, 3, 1976, pp. 423-438.

(190) Gomma, H., "A Modeling Approach to the Evaluation of Computer System Performance", in Measuring, Modeling, and Evaluating Computer Systems, H. Beilner, and E. Gelenbe (Eds.), North-Holland, New York, 1976, pp. 171-200 .

(191) Graham, G.S., "Queueing Network Models of Computer Systems Performance", Computing Surveys, 10, 3, 1978, pp. 219-224.

(192) Hine, J.H., "Generalizations of Queueing Network Models for Multiprogrammed Computer Systems", Ph.D. Thesis, Computer Science Department, Universoty of Wisconsin-Madison, 1973.

(193) Kienzle, M., "Measurement of Computer Systems for Queueing Network Models", Technical Report CSRG-86, Computer Systems Group, University of Toronto, 1977.

(194) Krzesinski, A., Gerber, S., and Teunissen, P., "A Multicalss Network Model of a Multiprogramming Time Sharing Computer System", Proceedings IFIP Congress'77, Toronto, 1977.

(195) Kuck, D.J., and Kumar, B., "A System Model for Computer Performance Evaluation", in International Symposium on Computer Performance Modeling, Measurement, and Evaluation, Cambridge, Mass., 1976, pp. 187-199.

(196) Kurinckx, A., and Pujolle, G., "Analytic Methods for Multiprocessor System Modeling", in Performance of Computer Systems, M. Arato, A. Butrimenko, E. Gelenbe (Eds.), North-Holland, New York, 1979, pp. 305-318.

(197) Lewis, P.A.W., and Shedler, G.S., "A Cyclic-Queue Model of System Overhead in Multiprogrammed Computer Systems", Journal of the Association for Computing Machinery, 18, 1971, pp. 199-220.

(198) Lipsky, L., and Church, J.D., "Applications of Queueing Network Model for a Computer System", Computing Surveys, 9, 3, 1977, pp. 205-221.

(37.b) Moore, C.G., III, "Network Models for Large-Scale Time Sharing Systems", Technical Report No. 71-1, Department of Industrial Engineering, University of Michigan, Ann Arbor, Michigan, 1971.

(199) Muntz, R.R., "Analytic Models for Computer System Performance Analysis", Proceedings of the NTG/G1 Conference on Computer Architecture and Operation Systems Braunshweig, Germany, 1974.

(200) Neilson, J.E., "An Analytical Performance Model of a Multiprogrammed Batch-Time Shared Computer", in International Symposium on Computer Performance Modeling, Measurement, and Evaluation, Cambridgs, Mass., 1976, pp. 59-70.

(201) Price, T.G., "A Comparison of Queueing Network Models and Measurements of a Multiprogrammed Computer System", Performance Evaluation Review, 5, 4, 1976.

(202) Rose, C.A., "A Measurement Procedure for Queueing Network Models of Computer Systems", Computing Surveys, 10, 3, 1978, pp. 263-280.

(203) Sauer, C.H., "Configuration of Computing Systems: An Approach Using Queueing Network Models", Ph.D. Thesis, University of Texas at Austin, 1975.

(204) Sauer, C.H., and MacNair, E.A., "Computer/Communication System Modeling with Extended Queueing Networks", IBM Research Report RC-6654, Yorktown Heights, New York, 1977.

(205) Scherr, A.A., An Analysis of Time-Shared Computer Systems, MIT Press, Mass. 1967.

(206) Shum, W.C., "Queueing Models for Computer Systems with General Service Time Distributions", Ph.D. Thesis, Division of Engineering and Applied Science, Harvard University, Cambridge, Mass., 1976.

(207) Spirn, J.R., "Multi-Queue Scheduling of Two Tasks", in International Symposium on Computer Performance, Modeling, Measurement, and Evaluation, Cambridge, Mass., 1976, pp. 102-108.

(208) Spragins, J., "Guest Editor's Introduction : Analytical Queueing Models", Computer, 13, 4, 1980, pp. 9-12.

(209) Wong, J.W., "Queueing Network Models for Computer Systems", Ph.D. Thesis, Computer Science Department, UCLA, 1975.

IV.B Other Fields

(210) Hershey, J.C., Weiss, E.N., and Cohen, M.A., "A Stochastic Service Network Model with Application to Hospital Facilities", Operations Research, 29, 1, 1981, pp. 1-22.

(211) Jones, A.T., "Queueing Networks in Large Scale Transportation Systems", Ph.D. Thesis, Department of Industrial Engineering, Purdue University, 1978.

(212) Kobayashi, H., and Konheim, A.G., "Queueing Models for Computer Communications System Analysis", IEEE Transactions on Communications, 25, 1, 1977, pp. 2-28.

(213) Marakami, K., "Queueing Network Model for Poly-Processor System and Analysis", Systems, Computers, Controls, 9, 2, 1978, pp. 78-87.

(214) Smith, J.M., and Rouse, W.B., "Application of Queueing Network Models to Optimization of Resource Allocation within Libraries", Journal of American Society for Information Science, 30, 5, 1979, pp. 250-263.

(215) Steudel, H.J., Pandit, S.M., and Wu, S.M., "A Multiple Time Series Approach to Modeling the Manufacturing Job-Shop as a Network of Queues", Management Science, 24, 4, 1977, pp. 456-463.

(216) Wong, J.W., "Queueing Network Modeling of Computer Communication Networks", Computing Surveys, 10, 3, 1978, pp. 343-351.

V. Miscellaneous Papers

(217) Agnew, C.E., "On Quqdratic Adaptive Routing Algorithms", Communications of the Association for Computing Machinery. 19, 1, 1976, pp. 18-22.

(218) Beutler, F.J., and Melamed, B., "Decomposition and Customer Streams of Feedback Queueing Networks in Equilibrium", Technical Report 77-1, Computer, Information, and Control Engineering Program, University of Michigan, 1977.

(219) Beutler, F.J., Melamed, B., and Zeigler, B.P., "Equilibrium Properties of Arbitrarily Introconnected Queueing Networks", Technical Report 75-4, Computer, Information, and Engineering Program, University of Michigan, 1975.

(220) Bharath Kumar, K.P., "Discrete Time Queueing Networks: Modeling Analysis and Design", Ph.D. Thesis, Department of Electrical Engineering, University of Hawaii, 1979.

(221) Bharath Kumar, K.P., "Discrete Time Queueing Systems and their Networks", IEEE Transactions on Communications, Vol. COM-28, 2, 1980, pp. 260-263.

(222) Bouhana, J., "Operational Aspects of Centralized Queueing Networks", Ph.D. Thesis, Computer Science Department, University of Wisconsin at Madison, 1976.

(223) Brandwajn, A., "Control Schemes in Queueing Networks", Management Science, 22, 7, 1976, pp. 810-822.

(224) Buzen, J.P., "Analysis of System Bottlenecks Using a Queueing Network Model", Proceedings of ACM Workshop on System Performance Evaluation, Harvard University, 1971, pp. 82-102.

(225) Buzen, J.P., and Denning, P.J., "Measuring and Calculating Queue Length Distributions", Computer, 13, 4, 1980, pp. 33-46.

(226) Cheng, S., "Design for Priority in Queuing Networks", Proceedings of the IEEE, 65, 9, 1977, pp. 1420-1421.

(227) Cherry, W.P., "The Superposition of Two Independent Markov Renewal Processes", Ph.D. Thesis, Department of Industrial and Operations Engineering, University of Michigan, 1972.

(228) Cooper, R.B., "INtroduction to Queueing Theory", The MacMillan Company, New York, 1972.

(229) Davignon, G.R., "Queues with Dependent Feedback", Department of Industrial and Operations Engineering Technical Report No. 72-11, University of Michigan, 1972.

(230) Disney, P.L., and Cherry, W.P., "Some Topics in Queueing Network Theory", in Mathematical Methods in Queueing Theory, Lecture Notes in Economics and Mathematical Systems No. 98, Springer-Verlag, Berlin-New York, 1974, pp. 23-33.

(231) Disney, R.L., and Morais, P.R., "Some Properties of Departure Processes from M/G/1/N Queues", Department of Industrial and Operations Engineering Technical Report, University of Michigan, 1971.

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

(233) Gallager, R.G., "A Minimum Delay Routing Algorithm Using Distributed Computation", IEEE Transactions on Communications, 25, 1, 1977, pp. 73-85.

(234) Giammo, T.P., "Validation of Computer Performance Model of the Exponantial Queueing Network Family", in International Symposium on Computer Performance Modeling, Measurement, and Evaluation, Cambridge, Mass., 1976, pp. 44-58.

(235) Gonnet, G.H., and Morgan, D.E., "Analysis of Closed Queueing Networks with Periodic Servers", IEEE Transactions on Software Engineering, Vol. SE-5, 6, 1979, pp. 653-658.

(236) Gonsalves, T.A., and Kumar, B., "Analysis of Interconnection Structures for Distributed Computer Systems", Performance Evaluation Review, 8, 3, 1979, pp. 89-98.

(237) Gonzalez, C., "Using Covariance Analysis as an Aid to Interpret the Results of a Performance Measurements", in International Symposium on Computer Performance Modeling, Measurement, and Evaluation, Cambridge, Mass., 1976, pp. 179-186.

(138) Irani, K.B., and Wallace, V.L., "On Network Linguistics and the Conversational Design of Queueing Networks", Journal of the Association for Computing Machinery, 18, 4, 1971.

(239) Jennings, A., Matrix Computation for Engineers and Scientists, Wiley-Interscience, New York, 1977.

(240) John, H., "Optimization and Analysis of Queueing Network", Ph.D. Thesis, Computer Science Department, University of Texas at Austin, 1975.

(241) Kawashima, T., "Turnaround Time Equations in Queueing Networks", Journal of the Operations Research Society of Japan, 21, 4, 1978, pp. 477-484.

(242) Kritzinger, P.S., Krzesinski, A.E., and Teunissen, P., "Incorporating System Overhead in Queueing Network Model", IEEE Transactions on Software Engineering, Vol. SE-6, 4, 1980, pp. 381-390.

(243) Lam, S.S., "Store-and- Forward Buffer Requirements in Packet Switching Network", IEEE Transactions on Communications, 24, 4, 1976, pp. 394-403.

(244) Lavenberg, S.S., "Stability and Maximum DepartureRate of Certain Opern Queueing Networks Having Finite Capacity Constraints", IBM Research Report RJ-1625, Yorktown Heights, New York, 1975.

(245) Lavenberg, S.S., and Reiser, M., "Stationary State Probabilities at Arrival Instants for Closed Queueing Networks with Multiple Types of Customers", IBM Research Report RC-7592, Yorktown Heights, New York, 1979.

(246) Lazowska, E., "Characterization of Service Time and Response Time Distributions in Queueing Networks of Computer Systems", Technical Report CSRG-85, Computer Systems Research Group, University of Toronto, 1977.

(247) Leech, R.L., "EQN Models for the Analysis and Design of a Computer Network of Functionalized Processes", Ph.D. Thesis, Department of Electrical Engineering, Duke University, 1977.

(248) Lipsky, L., and Carroll, J., "A Closed Queueing Loop with One Non-Exponential Server", Performance Evaluation Review, 8, 3, 1979, pp. 99-106.

(249) Maher, M.J., and Cabreira, J.G., "A Multi-Stage Cyclic Queueing Model", the International Journal of Production Research, 13, 3, 1975.

(250) Marlow, W.H., Mathematics for Operations Research, Wiley-Interscience, New York, 1978.

(251) Melamed, B., "Analysis and Simplifications of Discrete Event Systems and Jackson Queueing Networks", Technical Report 76-6, Department of Industrial and Operations Engineering, University of Michigan, 1976.

(252) Melamed, B., "Characterizations of Poisson Traffic Streams in Jackson Queueing Networks", Advances in Applied Probability, 11, 2, 1979, pp. 422-438.

(253) Melamed, B., Ziegler, B.P., and Beutler, F.J., " Simplifications of Jackson Queueing Networks", Technical Report 163, Department of Computer and Communications Services, University of Michigan, 1975.

(254) Michel, J.A., and Coffman, K.G., "Synthesis of a Feedback Queueing Discipline for Computer Operation", Journal of the Association for Computing Machinery, 21, 1974, pp. 329-339.

(255) Muntz, R.R., and Wong, J., "Asymptotic Properties of Closed Queueing Network Models", Proceedings of the Eighth Annual Princeton Conference on Information Sciences and Systems, Princeton University, 1964, pp. 348-352.

(256) Newell, G.F., Approximate Behavior of Tandem Queues, Springer-Verlag, New York, 1979,

(257) Noetzel, A.S., "Thoughput in Locally Balanced Computer System Models", in Computer Performance, K.M. Chandy, and M. Reiser (Eds.), North-Holland, New York, 1977, pp. 67-82.

(258) Noetzel, A.S., "Generalized Queueing Discipline for Product Form Network Solution", Journal of the Association for Computing Machinery, 21, 4, 1979, pp. 779-793.

(259) Pittel, B., "Closed Exponential Networks of Queues with Blocking: the Jackson-Type Stationary Distribution and its Asymptotic Analysis", IBM Research Report No. 26548, Yorktown Heights, New York, 1976.

(260) Pollard, J.M., "Combinatorial Properties of a Queueing Systems with Limited Availability", Advances in Applied Probability, 7, 1975, pp.844-877.

(261) Pollard, J.M., "Ergodicity Conditions and Congestion Control in Computer Networks", in Measuring, Modeling, and Evaluating Computer Systems, H. Beilner, and E. Gelenbe (Eds.), North-Holland, New York, 1976, pp. 287-318.

(262) Reiser, M., "A Queueing Network Analysis of Computer Communication Networks with Window Flow Control", IEEE Transactions on Communications, 27, 8, 1979, pp. 1199-1209.

(263) Reynolds, J.F., "The Covariance Structure of Queues and Related Processes - A Survey of Recent Work", Advances in Applied Probability, 7, 1975, pp. 383-415.

(264) Rose, C.A., "Validation of a Queuing Model with Classes of Customers", in International Symposium on Computer Performance Modeling, Measurement, and Evaluation, Cambridge, Mass., 1976, pp. 318-325.

(265) Schassberger, R., "Insensitivity of Steady State Distributions of Generalized Semi-Markov Process, Part I", Annals of Probability, 5, 1, 1977, pp. 87-99.

(266) Sevcik, K.C., "Priority Scheduling Disciplines in Queueing Network Models of Computer Systems", Proceedings IFIP Congress '77, Toronto, 1977.

(267) Sevcik, K.C., and Mitrani, I., "The Distribution of Queueing Network States at Input and Output Instants", in Performance of Computer Systems, M. Arato, A. Butrimenko, and E. Gelenbe (Eds.), North-Holland, New York, 1979, pp. 319-335.

(268) Solberg, J.J., "A Graph Theoretic Approach to the Study of Networks of Queues", Ph.D. Thesis, Department of Industrial Engineering, University of Michigan, 1969.

(269) Spirn, J.P., "Queueing Networks with Random Selection for Service", IEEE Transactions on Software Engineering, Vol. SE-5, 3, 1979, pp. 287-290.

(270) Trivedis, K.S., and Wagner, R.A., "Decision Model for Closed Queueing Networks", IEEE Transactions on Software Engineering, Vol. SE-5, 4, 1979, pp. 328-332.

CURRENT STATUS OF
QUEUEING NETWORK THEORY

by

CHI-JIUNN JOU
B.S. (INDUSTRIAL ENGINEERING)
Tunghai University, Taichung, Taiwan
Republic of China, 1979

AN ABSTRACT OF A MASTER'S THESIS

Sumitted in partial filfillment of the
requirement for the degree

MASTER OF SCIENCE

Department of Industrial Engineering
Kansas State University
Manhattan, Kansas

1981

Abstract

This research gives an overview of queueing networks for modeling computer systems. First, an introduction and calssif classification of queueing network models is given. Then the development of queueing network models is reviewed, and two different approaches to derive these network models are discussed. The traditional approach is based on stochastic assumption; another approach is operational analysis.

Finally, three different approaches for solving queueing network problems are described. One is to make some assumption about the system so as to get a queueing network model which has product form solution, then using some computational algorithm to solve it efficiently. The second approach is by the numerical methods which can give an exact solution but are restricted in terms of core memory or/and time of convergence. The third approach is by the approximation methods. Diffusion approximation will give nearly exact results when the system has heavy traffic conditions. Aggregation is an attractive approach for analyzing complex systems, since it allows the mixture of various techniques (e.g. product form solution, numerical methods, and simulation).

In addition to describing some basic ideas about those different approaches, computational algorithms, and methods, we also give a classified bibliography of research in queueing networks at the end of this report.