Markovian General Bulk Service Queueing System with Additional Server and Vacation

By

K. Jeevarathinam

A DISSERTATION SUBMITTED TO THE AVINASHILINGAM INSTITUTE FOR HOME SCIENCE AND HIGHER EDUCATION FOR WOMEN - DEEMED UNIVERSITY, COIMBATORE - 641 043

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN MATHEMATICS

APRIL- 1998

MARKOVIAN GENERAL BULK SERVICE QUEUEING SYSTEM WITH ADDITIONAL SERVER AND VACATION

BY

K. JEEVARATHINAM

A DISSERTATION SUBMITTED TO THE AVINASHILINGAM INSTITUTE FOR HOME SCIENCE AND HIGHER EDUCATION FOR WOMEN, (DEEMED UNIVERSITY), COIMBATORE - 641 043

In partial fulfilment of the requirements for the Degree of

MASTER OF SCIENCE IN MATHEMATICS

APRIL 1998.

CERTIFIED AS BONAFIDE RESEARCH WORK

K. N. Meenakshi

SIGNATURE OF THE HEAD OF THE DEPARTMENT

11.5.98

Afthab Begum 11/5/98

SIGNATURE OF THE GUIDE

THIS DISSERTATION

IS

DEDICATED TO MY MOTHER

K. MEENAMMAL

Acknowledgement

ACKNOWLEDGEMENT

The author is grateful to **PADMASHREE COLONEL Mrs. RAJAMMAL P. DEVADAS, M.A., M.Sc., Ph.D., (Ohio State), D.Sc., (Madras), Hon. D.H.L. (Oregon State), Hon. D.H.L. (Ohio State), Hon. D.Sc. (C.A. Agri., Kanpur), Hon. D.Sc., (University of Vister, Northern Ireland),** *Chancellor, Avinashilingam Institute for Home Science and Higher Education for Women (Deemed University), Coimbatore, for providing the opportunity to conduct her research in this much esteemed University.*

The author expresses her gratitude to **Mrs. LAKSHMI SANTHA RAJAGOPAL, M.S., (Tennessee), Ph.D., (Madras),** *Vice Chancellor, Avinashilingam Institute for Home Science and Higher Education for Women (Deemed University), Coimbatore, for providing facilities to carry out the study.*

The author extends her deep sense of gratitude to **Mrs. SAROJA PRABHAKARAN, M.A., Dip. in Ed., (Madras), Ph.D., (Mother Terasa),** *Registrar and* **Selvi. C.V.R. INDIRA, M.Sc., Ph.D., (Madras),** *Dean, Faculty of Science, Avinashilingam Institute for Home Science and Higher Education for Women (Deemed University), Coimbatore, for the keen interest shown by them.*

The author wishes to express her gratitude to **Selvi. K.N. MEENAKSHI. M.Sc., Ph.D.,** *Professor and Head of the Department of Mathematics, Avinashilingam Institute for Home Science and Higher Education for Women (Deemed University), Coimbatore, for the encouragement extended by her, during the entire course of study.*

*The author wishes to place on record her deeply felt debt to **Mrs. M.I. AFTHAB BEGUM, M.Sc., M.Phil., Ph.D.,** Reader in Mathematics, Avinashilingam Institute for Home Science and Higher Education for Women (Deemed University), Coimbatore, for her constant benevolent help, invaluable advice and all illuminating guidance. She gratefully acknowledges her persistent efforts and affectionate endeavours.*

*The author would like to extend her thanks to other **MEMBERS OF THE STAFF** and also to her **DEAR FRIENDS** who were responsible for the completion of this dissertation.*

*The author is greatly indebted to her **LOVING PARENTS AND ALL THE FAMILY MEMBERS** for their motivation and moral support rendered for the achievement of the dissertation.*

Contents

CONTENTS

Chapter I

CHAPTER - I

INTRODUCTION

The queueing theory had its origin in 1909, when **A.K. Erlang [9]** published his fundamental paper relating to the study of congestion in telephone traffic. Erlang is called the Father of Queueing theory. The queueing theory provides predictions about waiting times, the average number of waiting customers, the length of the busy period and so forth. These predictions help us to anticipate situations and to take appropriate measures to shorten the queues.

Some of the prominent applications of queueing theory are telephone conversation, machine repair, toll booths, taxi stands, inventory control, the loading and unloading of ships, scheduling patients in the hospital, clinics and applications in the computer field with respect to program scheduling etc.

A queueing system may be described as one having a service facility, at which units of some kind called customers arrive for service and whenever there are more units in the system than the service facility can handle simultaneously, a queue or waiting line develops. The waiting units take their turn for service according to a preassigned rule and they leave system after service.

1.1 BASIC CHARACTERISTICS OF QUEUEING SYSTEM

The following characteristics provide an adequate description of any queueing system.

1.1.1 ARRIVAL PATTERN OF CUSTOMERS

The arrival process describes the manner in which units arrive and join the system. In most of the cases, the arrival pattern is random and hence characterized by a probability distribution. The interval between two consecutive arrivals is called the probability distribution. The interval between two consecutive arrivals is called the inter arrival time. Arrivals may occur in batches instead of one at a time. In such case, the input is said to occur in bulk or batch.

1.1.2 SERVICE PATTERN

The service mechanism describes the manner in which service is rendered. The units may be served either singly or in batches. The time required for serving a unit or a batch is called service time. Service time is also a random variable in most of the cases, and represented by a probability distribution.

In case of batch service, the service system is termed as bulk service system. There are number of policies or rules, according to which batches for bulk service may be formed. **Neuts [20]** introduced a more general bulk service rule.

1.1.3 QUEUE DISCIPLINE

Queue discipline refers to the manner by which customers are selected for service when the queue has formed. The most common discipline that can be observed in every day life is First in First out (FIFS). Some others in common usage are Last in First out (LIFO); Selection for service In Random Order independent of the time of arrival to the queue (SIRO) etc. In some cases, priority discipline is followed. This discipline allows priority in service to some customers in relation to other customers waiting in queue.

1.1.4 SYSTEM CAPACITY

The system may have either a limited or an unlimited capacity for holding customers. The source from which the customers come may be finite or infinite.

1.1.5 SERVICE CHANNELS

Queueing system may have several service channels to provide service. These service channels may be arranged in parallel or in series or as a more complex combination of both, depending on the design of the system's service mechanism.

In parallel channels, a number of channels provide identical service facilities, so that several customers may be served simultaneously. In case of series channels, a customer must pass successively through the ordered channels, before service is completed.

1.1.6 MARKOVIAN QUEUEING MODELS

Queueing models with exponential inter-arrival times and exponential service time are called Markovian queueing models. Markovian queueing models are ususally solved by :

1. Difference - differential equations method using Rouche's theorem and generating functions and
2. Neuts Matrix - geometric Algorithm

The first method is disscussed in detail by **Gross and Harris [12]** and **Saaty [26]**. **Neuts [21]** developed the Matrix-geometric algorithmic approach for studying the steady-state queueing models. Matrix-geometric approach involves only real arithmetic and avoids the calculations of complex roots based on Rouche's theorem.

1.1.7 SERVER'S VACATION

From practical consideration, it may not always be worth while to keep server unnecessarily idle. In such situations, the server may utilize his idle time in a useful and optimal way to perform additional jobs. If the server is a machine, the idle time can be used for some preventive maintenance work. This utilization of idle time is termed as the server's vacation.

There are differrent types of vacation policies. The policies that we have used in this thesis are the following :

1. ***REPEATED VACATION***

In bulk service queueing models, a server on completion of a service will start serving again, if the system has the minimum number of customers required to start the service. Otherwise, the server will withdraw from the system for a vacation. On returning from a vacation , if the server finds less than the required number of customers, he may immediately take another vacation. He will continue in this manner, until he finds, upon returning from a vacation, the required number of waiting customers.

2. ***SINGLE VACATION***

The assumptions are same as those of repeated vacations except that, even if the server finds less than the minimum number of customers required for service, when he returns from a vacation, he stays in the system awaiting the queue length to reach the minimum number for starting his next service.

The other types of vacation in the literature are exceptional first vacation, gated vacation, random vacation, limited service vacation etc.

Relevant Literature Survey

1.2 RELEVANT LITERATURE SURVEY

1.2.1 BULK SERVICE QUEUEING MODELS

The models studied during the earlier years were confined to single arrival and individual service systems. The study of bulk queues in the real sense, originated in 1954, with the paper by **Bailey [4]**,who studied a bulk service queue and applied the results to a practial problem. During the last thirty years, bulk queues have received considerable attention both from researchers, interested in theoretical developments and professionals and practitioners interested in practical applications. The literature on bulk queues have grown tremendously. The publication of the excellent book entitiled "A first course in bulk queues" by **M.L. Chaudry and J. G. C. Templeton [7]** has been very much welcomed by all interested in the area.

Borthakur and Medhi [6] have obtained the steady state probabilities for the number of customers in the queue and the waiting time distribution for the $M / M_{(a,\, b)} / 1$ queue. **Arora [2] and Ghare [11]** have studied the multi server Markovian queue with bulk service. Later **Medhi [16]** derived the waiting time distribution for the $M / M_{(a,\, b)} / c$ queue by analytic methods. Following Neuts algorithmic approach, **Neuts and Nadarajan [22]** have studied the $M / M_{(a,\, b)} / c$ queue and obtained the stationary waiting time distribution. **Sim and Templeton [27]** used yet another recursive algorithm to obtain the steady - state numerical results for the same model. **Chaudry, Medhi, Sim, Templeton *et al.,* [8]** have discussed a two - heterogeneous - server Markovian queueing model with general bulk service rule such that channel i $(i = 1, 2)$ will serve customers in batches of size atleast $\mathbf{a_i}$ and atmost $\mathbf{b_i}$. The results obtained by them are the generalization of those given by **Arora [2], Medhi and Borthakur [14]**.

1.2.2 BULK SERVICE QUEUEING MODELS WITH VACATION

A queueing system with single server who serves customers according to general bullk service rule and leaves the system for vacation has been analysed by **Nadarajan and Subramanian [19]**.

Nadarajan and Audsin Mohana Dhas [18] have analysed, "Multi Server general bulk service queue with vacation". In this model both repeated vacation and single vacation of servers are considered. The steady - state probability vector of the number of customers in the queue has been obtained using matrix-geometric approach.

Audsin Mohana Dhas [3] has also obtained the steady state probability vector and the stability condition for the queueing model $M^N / M_{(a,\, b)} / c$, $1 \leq N \leq b$, with vacation. Bulk service queueing models with servers vacation have also been discussed by **Kandaswamy [13]**.

1.2.3 QUEUEING SYSTEMS WITH ADDITIONAL SERVER

In many queueing situations normally the number of servers is dependent on the queue length. For instance in a bank, more and more windows are opened for service when the queues in front of the existing windows are observed to become too long. This procedure is sometimes used even in the case of airlines, buses etc. The basic advantage in such a system is the inherent flexibility of service.

Romani [25] and **Philips [24]** have discussed the single service queueing model with variable number of service channels, assuming that when the waiting line size increases to some preassigned fixed number **N**, then with each arriving unit a new channel is made available . The new channel is cancelled at the termination of service, if there is no unit waiting, with the exception of one channel, which remains open at all times. **Murari [17]** has modified the result with the assumption that when the queue length increases to some undesirable number $\mathbf{m_1}$, another channel is called for its help. If the queue length increases to some undesirable number $\mathbf{m_2} > \mathbf{m_1}$, the third channel is called for their help and so on.

In all these cases, the additional server is made available instantaneously . **Bidhi Singh [5]** considered the situation when on reaching the queue length the maximum capacity of the waiting space, a search for additional server is started and the availability time of an additional server is a random variable. **Neuts [23]** analysed a similar situations for a bulk service queueing model and obtained the steady-state results using matrix-geometric method.

Garg and Khanna [10] have considered the steady state behaviour of a queueing system with additional server facility wherein arrivals occur in batches of variable size. Whenever the queue infront of the first server reaches certain length, the system adds another server. Steady - state probabilities and expected queue lengths in single server system and additional server system are calculated explicitly.

WORK DONE IN THIS THESIS

Bulk service seems to be realistic for many practical situations. For example, an elevator serves a group of people at the same time, passengers at a bus stand are served in groups etc.

There are number of rules according to which batches for bulk service may be formed. Throughout our discussion we followed the general bulk service rule introduced by **Neuts [20]**. According to this service rule, a server starts service only if atleast ***a*** customers are present in the queue, the maximum capacity being ***b*** customers.

In otherwords, if there are *X* customers waiting at the completion of a service, the following rule is considered:

For $0 \leq X < a$	No service
For $a \leq X \leq b$	Service is done for a batch of **X** customers
For $X > b$	Service is done for a batch of first **b** customers and the remaining **X-b** customers continue to wait in the queue.

In the case of multi server (c) bulk service queueing models, each free server starts his service as soon as the queue length reaches the lower limit ***a***. Therefore at certain instances the number of customers served by all the *C* servers may happen to be only ***Ca*** which does not satisfy the purpose of the batch service. In other words, the bulk service queueing system with *C* servers is effective only when the number of customers waiting in the queue is greater than ***Cb***. Sometimes it may also happen that some of the servers may stay idle in the system for more percentage of time. Thus, a question arises, whether the system can be handled effectively with less number of servers providing with additional service facility, whenever required? In this thesis, we have analysed this question.

In model I of chapter - II, we have discussed bulk service queueing model $M / M_{(a, b)} / 1$ with overloading.

The server serves the units according to the usual general bulk srevice rule of capacity maximum **(a, b)** as long as the queue length is less than or equal to ***2b.***

When the queue length exceeds ***2b,*** the batch capacity is increased to b' $(b \leq b' \leq 2b)$ from ***b***.

Whenever the queue length shorten to a number less than **2b+1** the original batch size is followed.

In model II of chapter - II we have analysed the following additional server model :

There is one regular server serving the units according to the usual bulk service rule as long as the queue length is less than **2b+1**.

An additional server is employed instantaneously if the queue length exceeds ***2b***. The service distribution for both the servers are assumed to be exponential with same srevice rates.

The free server is dropped when the queue length is less than **2b+1**.

The expressions for steady-state probabilities and some of the effective measures are obtained in a closed form for both the models.

In chapter - III we have considered the generalizations of model II of chapter -II.

In the models of chapter - III, an additional server is employed instantaneously whenever the queue length exceeds a preassigned value ***Nb***. The service rate of the regular server is μ and that of additional server is μ_1.

The regular server is allowed to go on a vacation if he finds less than a customer in the system.

Multiple vacation is considered in model 3.1 and single vacation is followed in model 3.2.

Steady-state probability vectors are calculated using matrix-geometric method. The expected queue length is calculated for various parametric values for both the models.

Chapter II

CHAPTER - II

$M / M_{(a,b)} / 1$ QUEUEING MODEL WITH ADDITIONAL SERVICE FACILITY

In section 1 of this chapter we have first discussed the two homogeneous servers bulk service queueing model $M / M_{(a, b)} / 2$ and listed the expressions for

1. Expected queue length.
2. Average number of batches waiting in the queue.
3. Probability that only one server is busy.
4. Probability that both the servers are busy.

In section 2 of this chapter we have analysed $M / M_{(a, b)} / 1$ queueing model with over loading.

In section 3 we have considered the model in which there is single server, as long as the queue length is less than 2b+1 and an additional server is appointed instantaneously when the queue length exceeds 2b.

The expressions (1) to (4) mentioned above are also derived for the overloading model and the additional server model.

The expected queue size are compared for all the three models.

A cost analysis is also presented, by considering the different cost associated with the servers and the waiting of the customers.

SECTION 2.1

M / $M_{(a,\,b)}$ / 2 QUEUEING SYSTEM

There are two homogeneous servers in the service facility and the service time of each server has an exponential distribution with service rate μ. The servers serve the customers in the queue according to the general bulk service rule.

Each server will start his service if there exist atleast **a** units in the queue. If a free server finds more than **a** units but less than **b** units in the queue he takes all the customers into a batch together, for service. If he finds more than **b** units he takes only **b** members in a batch for service. The service distribution is independent of the batch size.

Steady-State Results

Let $\mathbf{P}_{i\,n}$ denote the steady-state probability that there are **n** units in the queue and

both servers are busy, if $i = 2$ $(n \geq 0)$

one server is busy, if $i = 1$ $(0 \leq n \leq a\text{-}1)$

both servers are idle, if $i = 0$ $(0 \leq n \leq a\text{-}1)$.

The steady-state equations satisfied by $\mathbf{P}_{i\,n}$ are :

$$(\lambda + 2\mu)\, P_{2\,o} = \lambda\, P_{1\,a-1} + 2\,\mu \sum_{k=a}^{b} P_{2\,K} \qquad (2.1.1)$$

$$(\lambda + 2\mu)\, P_{2\,n} = \lambda\, P_{2\,n-1} + 2\,\mu\, P_{2\,n+b} \qquad (n \geq 1) \qquad (2.1.2)$$

$$(\lambda + \mu)\, P_{1\,o} = \lambda\, P_{0\,a-1} + 2\,\mu\, P_{2\,o} \qquad (2.1.3)$$

$$(\lambda + \mu)\, P_{1\,n} = \lambda\, P_{1\,n-1} + 2\,\mu\, P_{2\,n} \qquad (1 \leq n \leq a\text{-}1) \qquad (2.1.4)$$

$$\lambda P_{0\,n} = \lambda\, P_{0\,n-1} + \mu P_{1\,n} \qquad (1 \leq n \leq a\text{-}1) \qquad (2.1.5)$$

$$\lambda P_{0\,0} = \lambda P_{1\,o} \qquad (2.1.6)$$

Equation (2.1.2) implies,

$$[2\mu\, E^{b+1} - (\lambda + 2\mu)\, E + \lambda]\, P_{2\,n} = 0$$

$$P_{2\,n} = r_1^{\,n}\, P_{2\,0} \qquad (n \geq 0)$$

where r_1 in the real root of $2\mu\, Z^{b+1} - (\lambda + 2\mu)\, Z + \lambda = 0$, which lies in (0, 1). Such a real root exists uniquely, if $\frac{\lambda}{2b\mu} < 1$.

Equation (2.1.4) implies,

$$[(\lambda + \mu)\, E - \lambda]\, P_{1\,n} = 2\,\mu\, P_{2\,n+1}$$

solving we get,

$$P_{1\,n} = \left[AR^{\,n} + \frac{2\mu\, r_1^{\,n+1}}{(\lambda + \mu)\, r_1 - \lambda} \right] P_{2\,0}$$

(ie) $$P_{1\,n} = (AR^{n} + kr_1^{\,n})\, P_{2\,0} \qquad (0 \leq n \leq a\text{-}1) \qquad (2.1.7)$$

where $R = \frac{\lambda}{\lambda + \mu}$, $k = \left[\frac{2\mu r_1}{(\lambda + \mu)\, r_1 - \lambda} \right]$

Equation (2.1.5) implies,

$$P_{0\,n} = P_{0\,0} + \frac{\mu}{\lambda} \sum_{1}^{n} P_{1\,K} \qquad (1 \leq n \leq a\text{-}1)$$

$$= \frac{\mu}{\lambda} \sum_{0}^{n} P_{1\,K}$$

Substituting for $P_{1\,K}$, We have

$$P_{0\,n} = \frac{\mu}{\lambda} \left[\frac{A(1 - R^{n+1})}{1 - R} + \frac{k(1 - r_1^{\,n+1})}{1 - r_1} \right] P_{2\,0}$$

$$(0 \leq n \leq a\text{-}1) \qquad (2.1.8)$$

Equation (2.1.3) gives

$$(\lambda + \mu) P_{1\,0} = \lambda P_{0\,a-1} + 2\mu P_{2\,0}$$

Substituting for $P_{i\,j}$ we get,

$$(\lambda + \mu)(A + k) = \mu \left[\frac{A(1 - R^a)}{1 - R} + \frac{k(1 - r_1^a)}{1 - r_1} \right] \mu + 2\mu$$

Simplifying we have,

$$AR^a (\lambda + \mu) = - \frac{k \mu r_1^a}{1 - r_1} + \frac{2\mu}{1 - r_1}$$

$$\text{(i.e)} \quad A\lambda R^{a-1} = \frac{\mu}{1 - r_1}(2 - kr_1^a)$$

$$A = \frac{\mu(2 - kr_1^a)}{\lambda R^{a-1}(1 - r_1)}$$

All the expressions are obtained interms of $P_{2\,0}$

$P_{2\,0}$ is obtained from the normalizing equation

$$\sum_{n=0}^{a-1} (P_{1\,n} + P_{0\,n}) + \sum_{n=0}^{\infty} P_{2\,n} = 1$$

$$\frac{\mu}{\lambda} \left[\frac{A}{1 - R} + \frac{k}{1 - r_1} \right] a - \frac{2\mu}{\lambda (1 - r_1)} \sum_{0}^{a-1} r_1^{n+1} + \sum_{0}^{\infty} r_1^n \Bigg] P_{2\,0} = 1$$

$$\left[\frac{\mu}{\lambda} \left[\frac{A}{1 - R} + \frac{k}{1 - r_1} \right] a + \frac{1}{1 - r_1} - \frac{2\mu}{\lambda (1 - r_1)} \frac{(r_1 - r^{a+1})}{(1 - r_1)} \right] P_{2\,0} = 1$$

(i.e) $P_{2\,0}$ is obtained as

$$P_{2\,0}^{-1} = \left[\frac{1}{1 - r_1} + \frac{\mu}{\lambda} a \left[\frac{A}{1 - R} + \frac{k}{1 - r_1} \right] - \frac{2\mu r_1}{\lambda (1 - r_1)^2} (1 - r_1^a) \right] \qquad (2.1.9)$$

Next we shall calculate the expected queue length for this model.

$$L_q = \sum_{n=1}^{a-1} n(P_{0n} + P_{1n}) + \sum_{n=1}^{\infty} n P_{2n}$$

substituting for P_{in}, i = 0, 1, 2 from (2.1.7) and (2.1.8) we get,

$$L_q = \left[\sum_{n=1}^{a-1} n \left(\frac{\mu}{\lambda}\left[\frac{A}{1-R} + \frac{k}{1-r_1}\right] - \frac{2\mu r_1^{n+1}}{\lambda(1-r_1)} \right) + \sum_{n=1}^{\infty} n r_1^{n} \right] P_{20}$$

$$= \left[\frac{r_1}{(1-r_2)} + \frac{\mu}{\lambda}\left[\frac{A}{1-R} + \frac{k}{1-r_1}\right] \frac{a(a-1)}{2} - \frac{2\mu r_1}{\lambda(1-r_1)} \left(\frac{r_1(1-r_1^{a}) - a\, r_1^{a}(1-r_1)}{(1-r_1)^2} \right) \right] P_{20} \qquad (2.1.10)$$

The other measures obtained are the following :

(i) P_{2B} - The probability that both the servers are busy is $\sum_{n=1}^{\infty} P_{2n}$

i.e., $$P_{2B} = \frac{1}{1-r_1} P_{20}$$

(ii) P_{1B} - The probability that only one server is busy is $\sum_{n=0}^{a-1} P_{1n}$

i.e., $$P_{1B} = \left(A\left(\frac{1-R^{a}}{1-R}\right) + k\left(\frac{1-r_1^{a}}{1-r_1}\right) \right) P_{20}$$

(iii) P_I - The probability that both the servers are idle is $\sum_{n=0}^{a-1} P_{0n}$

i.e., $$P_I = \left[a\left(\frac{A}{1-R} + \frac{k}{1-r_1}\right) - \left(\frac{AR(1-R^{a})}{(1-R)^2} + \frac{k\, r_1(1-r_1^{a})}{(1-r_1)^2} \right) \right] \frac{\mu}{\lambda} P_{20}$$

SECTION 2.2

MODEL 2.1 (OVER LOADING)

Assume that units arrive into a service facility according to a Poisson process with rate λ. The units are served by a single server in batches of size **n** varying from **a** to **b′**. The service times, irrespective of the batch size, are assumed to have independent identical exponential distribution with parameter μ.

The service rule is assumed to operate as follows: The server starts service only when a minimum number **a** is reached. If the server, on completing his service finds

(i) $n(a \leq n \leq b)$ units in the queue, he takes all the units into the batch of service.

(ii) $n(b \leq n \leq 2b)$ units in the queue, he takes only **b** units in his batch of service.

(iii) $n(n > 2b)$ units, he takes **b′** units $(b \leq b' \leq 2b)$ in the batch of service.

2.2.1. Steady-State Equations

Assuming the steady-state exists, let $P_{0\,n}$ $(0 \leq n \leq a\text{-}1)$ denote the probability that the server is idle and **n** units are in the queue.

$P_{1\,n}$ $(n \geq 0)$ refers to the situations when the server is busy. The standard arguements lead to the following equations:

$$\lambda P_{0\,0} = \mu P_{1\,0} \qquad (2.2.1)$$

$$\lambda P_{0\,n} = \lambda P_{0\,n-1} + \mu P_{1\,n} \qquad (0 \leq n \leq a\text{-}1) \qquad (2.2.2)$$

$$(\lambda + \mu) P_{1\,n} = \lambda P_{1\,n-1} + \mu P_{1\,n+b'} \qquad (n > b) \qquad (2.2.3)$$

$$(\lambda + \mu) P_{1\,n} = \lambda P_{1\,n-1} + \mu (P_{1\,n+b} + P_{1\,n+b'}) \qquad (2b\text{-}\,b' < n \leq b) \qquad (2.2.4)$$

$$(\lambda + \mu) P_{1\,n} = \lambda P_{1\,n-1} + \mu P_{1\,n+b} \qquad (1 \leq n \leq 2b\text{-}\,b') \qquad (2.2.5)$$

$$(\lambda + \mu) P_{1\,0} = \lambda P_{0\,a-1} + \mu \sum_{n=a}^{b} P_{1\,n} \qquad (2.2.6)$$

2.2.2. Solution of Steady-state Probabilities

Equation (2.2.3) implies

$$[\mu E^{b'+1} - (\lambda + \mu) E + \lambda]P_{1\,n} = 0 \qquad (n \geq b)$$

$$P_{1\,n} = r_0^{\,n-b} P_{1\,b} \qquad (n \geq b) \qquad (2.2.7)$$

where r_0 is the unique root of the characteristic equation $(\mu Z^{b'+1} - (\lambda+\mu) Z + \lambda) = 0$ that lies inside the interval (0, 1). Such root exists uniquely whenever $\frac{\lambda}{b'\mu} < 1$.

Equation (2.2.4) implies

$$[(\lambda + \mu) E - \lambda] P_{1\,n} = \mu [P_{1\,n+b+1} + P_{1\,n+b'+1}] \qquad (2b - b' \leq n \leq b - 1)$$

substituting for $P_{i\,n}$, $n \geq b$ from (2.2.7) and solving for $P_{1\,n}$ we get,

$$P_{1\,n} = [AR^{\,n} + (r_0^{-b'} + r_0^{-b})\, r_0^{\,n}]\, P_{1\,b} \qquad (2b - b' \leq n \leq b - 1) \qquad (2.2.8)$$

where $R = \frac{\lambda}{\lambda + \mu}$ and A is obtained from equation (2.2.4) at n = b.

$$(\lambda + \mu) P_{1\,b} = \lambda P_{1\,b-1} + \mu[P_{1\,2b} + P_{1\,b+b'}]$$

Substituting for $P_{1\,n}$ from (2.2.7) and (2.2.8) we have

$$(\lambda + \mu) = \lambda\,(AR^{\,b-1} + (r_0^{-b'} + r_0^{-b})\, r_0^{\,b-1}) + \mu(r_0^{\,b} + r_0^{\,b'})$$

$$\lambda AR^{\,b-1} = (\lambda + \mu) - \lambda\,(r_0^{\,b-b'-1} + r_0^{-1}) - \mu(r_0^{\,b} + r_0^{\,b'})$$

Simplifying we have,

$$A = -\left(\frac{1}{R^{b} r_0^{\,b'-b}}\right) \qquad (2.2.9)$$

Similarly, equation (2.2.5) gives

$$[(\lambda + \mu) E - \lambda] P_{1\,n} = \mu P_{1\,n+b+1} \qquad (0 \leq n \leq 2b-b'-1)$$

Solving the difference equation, we get,

$$P_{1\,n} = \left(BR^n + \frac{\mu\, r_0^{n+1}}{(\lambda+\mu)r_0 - \lambda}\right) P_{1\,b}$$

Since r_0 is the root of the $\mu z^{b'+1} - (\lambda+\mu) z + \lambda = 0$, we have,

$$P_{1\,n} = (BR^n + r_0^{\,n-b'})\, P_{1\,b} \qquad (0 \le n \le 2b-b'-1) \qquad (2.2.10)$$

The constant B is evaluted from equation (2.2.5) at n = 2b - b′

$$(\lambda+\mu)\, P_{1\,2b-b'} = [\lambda\,(BR^{2b-b'-1} + (r_0^{2b-2b'-1}) + \mu r_0^{2b-b'}]\, P_{1\,b}$$

(From (2.2.7 and 2.2.10)

Subsituting for $P_{1\,2b-b'}$ from (2.2.8.) and simplifying, we have

$$(\lambda+\mu)\, AR^{2b-b'} = \lambda\, BR^{2b-b'-1} - (\lambda+\mu)\, r_0^{b-b'}$$

$$AR^{2b-b'} = BR^{2b-b'} + AR^{b}$$

$$B = A(1 - R^{b'-b}) \qquad (2.2.11)$$

Adding the equation (2.2.2) over n = 1 to k ($0 \le k \le a-1$) and changing k to n, we get

$$P_{0\,n} = \frac{\mu}{\lambda} \sum_{j=0}^{n} P_{1\,j} \qquad (0 \le n \le a-1)$$

The expression for $P_{0\,n}$ depends on whether $(a-1) \lesseqgtr (2b - b')$.

If $(a-1) < (2b - b')$, then

$$P_{0\,n} = \frac{\mu}{\lambda}\left[\left(\frac{B(1 - R^{n+1})}{1-R} + r_0^{-b'}\,\frac{(1 - r_0^{\,n+1})}{1 - r_0}\right) P_{1\,b}\right] \qquad (0 \le n \le a-1) \qquad (2.2.12)$$

If (a-1) ≥ (2b - b′), then

$$P_{0\,n} = \frac{\mu}{\lambda}\left[\sum_{j=0}^{2b-b'-1} P_{1\,j} + \sum_{2b-b'}^{n} P_{1\,j}\right] \qquad (0 \le n \le a-1)$$

substituting from (2.2.8) and (2.2.10), we get

$$P_{0\,n} = \frac{\mu}{\lambda}\left[\sum_{0}^{2b-b'-1}\left(BR^{n} + r_0^{\,n-b'}\right) + \sum_{2b-b'}^{n} AR^{n} + \left(r_0^{-b'} + r_0^{-b}\right) r_0^{\,n}\right] P_{1\,b}$$

$$= \frac{\mu}{\lambda}\left[\sum_{0}^{2b-b'-1} BR^{n} + \sum_{2b-b'}^{n} AR^{n} + \sum_{0}^{n} r_0^{\,n-b'} + \sum_{2b-b'}^{n} r_0^{\,n-b}\right] P_{1\,b}$$

Simplifying we have,

$$= \frac{\mu}{\lambda}\left[\frac{B(1-R^{n+1})}{1-R} + \frac{r_0^{-b'}(1-r_0^{\,n+1})}{1-r_0} + \frac{r_0^{-b}(r_0^{\,2b-b'} - r_0^{\,n+1})}{1-r_0} + AR^{b'-b}\left(\frac{R^{2b-b'} - R^{n+1})}{1-R}\right)\right] P_{1\,b}$$

$$P_{0\,n} = \frac{\mu}{\lambda}\left[\left(\frac{B(1-R^{n+1})}{1-R} + \frac{r_0^{-b'}(1-r_0^{\,n+1})}{1-r_0}\right) + C_n\right] P_{1\,b}$$

where $C_n = 0$ for $0 \le n \le 2b - b' - 1$

$$= \left(r_0^{-b}\,\frac{(r_0^{\,2b-b'} - r_0^{\,n+1})}{1-r_0} + AR^{b'-b}\,\frac{(R^{2b-b'} - R^{n+1})}{1-R}\right) \qquad (2.2.13)$$

for $2b - b' \le n \le a-1$

Thus, the steady-state probabilities are expressed in terms of $P_{1\,b}$ and $P_{1\,b}$ is calculated from the normalizing condition.

$$\sum_{n=0}^{a-1} P_{0\,n} + \sum_{n=0}^{\infty} P_{1\,n} = 1$$

substituting for $P_{0\,n}$ and $P_{1\,n}$, we get

$$\frac{\mu}{\lambda}\left[\sum_{0}^{a-1}\left[\frac{B(1-R^{n+1})}{1-R}+\frac{r_0^{-b'}(1-r_0^{\,n+1})}{1-r_0}+C_n\right]+\sum_{n=0}^{2b-b'-1}\left(BR^n+r_0^{\,n-b'}\right)+\sum_{2b-b'}^{b-1}\left(AR^n+\left(r_0^{-b'}+r_0^{-b}\right)r_0^{\,n}\right)+\sum_{b}^{\infty}r_1^{\,n-b}\right]P_{1\,b}=1$$

Simplifying,

$$\left[B\left[\frac{\mu}{\lambda}\left(\frac{1}{1-R}\right)a+\frac{1}{1-R}-\frac{R\mu}{(1-R)\lambda}\frac{(1-R^a)}{1-R}\right]+r_0^{-b'}\left[\frac{\mu\, a}{\lambda\,(1-r_0)}+\frac{1}{1-r_0}-\frac{r_0}{1-r_0}\frac{\mu}{\lambda}\frac{(1-r_0^{\,a})}{1-r_0}\right]+\frac{\mu}{\lambda}\sum_{n=0}^{a-1}C_n\right]P_{1\,b}=1$$

(i.e.) $\quad P_{1\,b} = [r_0^{-b'}\, g(r_0) + Bg(R) + C\,]^{-1}$ (2.2.14)

where $\quad C = 0 \qquad ,\text{if } (a-1) < (2b-b')$

$$= \frac{\mu}{\lambda}\sum_{n=2b-b'}^{a-1} C_n\ ,\ \text{if } (a-1) \geq (2b-b')$$

and $$g(X) = \frac{1+\dfrac{\mu}{\lambda}\left(a-\dfrac{X(1-X^a)}{1-X}\right)}{1-X}$$

Thus the expressions for the steady - state probabilities are given by (2.2.8), (2.2.10), (2.2.13) and (2.2.14).

2.2.2 The Mean Queue - Length

The mean queue length for the system is

$$L_q = \left(\sum_{n=1}^{a-1} n\, P_{0\,n} + \sum_{n=1}^{\infty} n\, P_{1\,n}\right)$$

substituting for $P_{0\,n}$ and $P_{1\,n}$ we have,

$$= \left[\sum_{n=0}^{a-1} n \left[\frac{\mu}{\lambda} \frac{\left(B(1-R^{n+1})\right)}{1-R} + \frac{r_0^{-b'}(1-r_0^{n+1})}{1-r_0} + Cn \right] \right.$$

$$\left. + \sum_{n=0}^{2b-b'-1} n\left(BR^n + r_0^{n-b'}\right) + \sum_{n=2b-b'}^{b-1} n\left(AR^n + r_0^{n-b'} + r_0^{n-b}\right) + \sum_{b}^{\infty} n\, r_0^{n-b} \right] P_{1\ b}$$

Simplifying we have,

$$L_q = [r_0^{-b'} H(r_0) + B\, H(R) - (b' - b) \frac{r_0^{b+1}}{1-r_0} + C']\, P_{1b} \qquad (2.2.15)$$

where

$$C' = 0 \qquad ,\text{if } (a-1) < 2b-b'$$

$$= \sum_{n=2b-b'}^{a-1} nC_n \qquad ,\text{if } (a-1) \geq 2b-b',$$

$$H(X) = \frac{\mu}{\lambda(1-X)} \left[\frac{\lambda X}{\mu[1-X]} + \frac{a(a-1)}{2} + \frac{aX^{a+1}(1-X) - X^2(1-X^a)}{(1-X)^2} \right] \qquad (2.2.16)$$

Remark

When $b' = b$, the queue length given by the equation (2.2.15) coincides with that of $M/M_{(a,\, b)}/1$ model.

SECTION 2.3

MODEL 2.2 (ADDITIONAL SERVER)

In this section we consider a bulk service queuing model with additional server and obtain the steady-state probability results by solving the difference - differential equations.

2.3.1. Problem Description

Customers arrive at the system in a Poisson stream with parameter λ. There is a single server serving the units in batches according to the general bulk service rule with max (a, b) in the batch as long as the queue length is less than or equal to 2b. When the queue size exceeds 2b, an additional server is employed instantaneously. The service time distributions of both the servers are exponential with the same parameter μ. When the length of the queue is less than (2b + 1), the server who terminates his service is withdrawn and the system continues with a single server.

2.3.2. State Space Of The Model

Let (0, n) denote the server is idle and there are **n** units in the queue ($0 \leq n \leq a-1$) and (1, n) denote only one server is available in the system and he is busy. Such state exists for $0 \leq n \leq 2b$.

(2, n) denotes that both the servers are busy and there are n customers in the queue. This state exists for $n \geq b+1$.

2.3.3. Steady-State Equations

Assuming that the steady-state exists, let $P_{i\,n}$ denote the steady-state probabilities when the system is in state (i, n). The difference-differential equations satisfied by $P_{i\,n}$ are :

$$\lambda P_{0\,0} = \mu P_{1\,0} \qquad (2.3.1)$$

$$\lambda P_{0\,n} = \lambda P_{0\,n-1} + \mu P_{1\,n} \qquad (1 \le n \le a-1) \qquad (2.3.2)$$

$$(\lambda + \mu) P_{1\,0} = \lambda P_{0\,a-1} + \mu \sum_{S=a}^{b} P_{1\,s} \qquad (2.3.3)$$

$$(\lambda + \mu) P_{1\,n} = \lambda P_{1\,n-1} + \mu P_{1\,n+b} \qquad (1 \le n \le b) \qquad (2.3.4)$$

$$(\lambda + \mu) P_{1\,n} = \lambda P_{1\,n-1} + 2\mu P_{2\,n} \qquad (b \le n \le 2b) \qquad (2.3.5)$$

$$(\lambda + 2\mu) P_{2\,b+1} = \lambda P_{1\,2b} + 2\mu P_{2\,2b+1} \qquad (2.3.6)$$

$$(\lambda + 2\mu) P_{2\,n} = \lambda P_{2\,n-1} + 2\mu P_{2\,b+1} \qquad (n \ge b+2) \qquad (2.3.7)$$

2.3.4. The Solution Of Steady-State Probabilities

Equation (2.3.7) implies

$$[2\mu E^{b+1} - (\lambda+2\mu) E+\lambda] P_{2\,n} = 0$$

$$P_{2\,n} = r_1^{n-(b+1)} P_{2\,b+1} \qquad (n \ge b+1)$$

where r_1 is the unique root, which lies in the interval of (0, 1), of the characteristic equation $[2\mu Z^{b+1} - (\lambda+2\mu) Z+\lambda] = 0$, when $\frac{\lambda}{2b\mu} < 1$.

Equation (2.3.6) implies

$$(\lambda+2\mu)P_{2\,b+1} = \lambda P_{1\,2b} + 2\mu r_1^{b} P_{2\,b+1}$$

(i.e.), $$P_{2\,b+1} \frac{(\lambda + 2\mu)r_1 - 2\mu r_1^{b+1}}{r_1} = \lambda P_{1\,2b}.$$

(i.e.) $$P_{1\,2b} = \frac{1}{r_1} P_{2\,b+1}$$

This implies

$$P_{2\,n} = r^{n-b} P_{1\,2b} \qquad (n \ge b+1) \qquad (2.3.8)$$

Equation (2.3.5) implies

$$[(\lambda+2\mu)E - \lambda\,]P_{1\ n} = 2\mu\, P_{2\ n+1} \qquad (b \le n \le 2b\text{-}1)$$

$$P_{1\ n} = \left[AR^{n} + \frac{2\mu\, r_1^{\,n\text{-}b+1}}{(\lambda + \mu)r_1 - \lambda}\right] P_{1\ 2b}$$

$$P_{1\ n} = \left[AR^{n} + kr_1^{\,n\text{-}b}\right] P_{1\ 2b} \qquad (b \le n \le 2b\text{-}1)$$

where $R = \dfrac{\lambda}{(\lambda + \mu)}$; $K = \dfrac{2\mu\, r_1}{(\lambda + \mu)r_1 - \lambda}$ and

A is obtained from equation (2.3.5) at n = 2b

$$(\lambda+\mu)P_{1\ 2b} = \lambda\, P_{1\ 2b\text{-}1} + 2\mu\, P_{2\ 2b}$$

$$(\lambda+\mu) = \lambda\left(AR^{2b\text{-}1} + kr_1^{\,b\text{-}1}\right) + 2\mu\,(r_1^{\,b})$$

$$\lambda AR^{2b\text{-}1} = (\lambda + \mu) - 2\mu\, r_1^{\,b} - \lambda\, kr_1^{\,b\text{-}1}$$

$$= \frac{\lambda - \mu r_1}{r_1} - \frac{\lambda(2\mu r_1^{\,b})}{(\lambda + \mu)r_1 - \lambda}$$

A further simplification implies,

$$AR^{2b} = \frac{-\mu r_1}{(\lambda + \mu)r_1 - \lambda}$$

$$A = \frac{-K}{2R^{2b}}$$

i.e., $$P_{1\ n} = K\left(r_1^{\,n-b} - \frac{R^{\,n\text{-}2b}}{2}\right) P_{1\ 2b} \qquad (b \le n \le 2b) \qquad (2.3.9)$$

Equation (2.3.4) impies

$$[(\lambda+\mu)E - \lambda\,]P_{1\ n} = 2\mu\, P_{1\ n+b+1}$$

By induction from the equation 2.3.4,we get ,

$$P_{1\ n} = [BR^{n-b} + Kr_1^{\ n} - (n\text{-}b+1)R^{n-b+1}\frac{\mu}{\lambda}]\ \frac{K}{2}P_{1\ 2b}$$

$(0 \le n \le b\text{-}1)$ (2.3.10)

where $B = 2 - R - \left(\frac{K}{2} + R^{-b}\right)$

Equations (2.3.1) and (2.3.2) together imply

$$P_{0\ n} = \frac{\mu}{\lambda}\left(\sum_{j=0}^{n} P_{1\ j}\right) \qquad (0 \le n \le a\text{-}1)$$

Adding equation (2.3.2) and (2.3.4) over n = 1 to k, we get

$$\lambda(P_{0\ k} + P_{1\ k}) = \lambda\ (P_{0\ 0} + P_{1\ 0}) + \mu \sum_{n=1}^{K} P_{1\ n+b}$$

$$= (\lambda+\mu)P_{1\ 0} + \mu \sum_{n=1}^{K} P_{1\ n+b}$$

On simplifying, we get,

$$(P_{0\ n} + P_{1\ n}) = \frac{P_{1\ 0}}{R} + \frac{K\mu}{\lambda}\left(\frac{r_0 - r_0^{\ n+1}}{1 - r_0} - \frac{R^{-b}}{2}\frac{(R - R^{n+1})}{1 - R}\right) P_{1\ 2b}$$

$(0 \le n \le a\text{-}1)$ (2.3.11)

$P_{1\ 2b}$ can be evaluated using the normalizing equation.

The mean queue length is obtained by substituting for $P_{i\ n}$ from (2.3.8), (2.3.9), (2.3.10) and (2.3.11) in Lq.

$$Lq = \sum_{n=b+1}^{\infty} nP_{2\ n} + \sum_{n=b}^{2b} nP_{1\ n} + \sum_{n=0}^{a-1} n(P_{0\ n} + P_{1\ n}) + \sum_{n=a}^{b-1} nP_{1\ n}$$

SECTION 2.4

COST ANALYSIS

In this section, we discuss the cost analysis for the three models analysed in this chapter, by considering different costs associated with the servers and the waiting of the customers. The mean queue length for all the three models are also presented in this section.

Let

(i) d denote the fixed cost per unit time for each server,

(ii) e denote the waiting cost per unit, per unit time and

(iii) f denote the cost per unit service by each server

If C denotes the total cost per unit time of the system, then

1. For $M / M_{(a, b)} / 2$ model,

$$C = 2d + e\,Lq + f\,\mu\,(2P_{2\ B} + P_{1\ B})$$

where Lq is the mean queue length and $P_{2\ B}$ ($P_{1\ B}$) denotes the probability that both the servers (one server) busy.

2. For model I (over loading),

$$C = d + e\,Lq + f\,\mu\,(P_B)$$

where P_B = Probability that the server is busy.

$$= \sum_{0}^{\infty} P_{1\ n}$$

$$= \left(\frac{r_0^{-b'}}{1 - r_0} + \frac{B}{1\ -\ R} \right) P_{1\ b}$$

3. For model II (Additional server),

$$C = d(1 + P_{2\,B}) + e\,Lq + f\,\mu\,(2P_{2\,B} + P_{1\,B}),$$

$$P_{2\,B} = \frac{r_1}{1 - r_1} P_{1\;2b}$$

and $P_{1\,B} = \sum_{n=0}^{2b} P_{1\;n}$ is obtained from (2.3.9) and (2.3.10)

NUMERICAL ANALYSIS

Tables 2.1a, 2.1b and 2.1c give the average queue length of the over loading model for various values of b' ($b \leq b' \leq 2b$). For given values of a, b, μ and λ, the table values show that the mean queue length (Lq) decreases as b' increases from b to 2b.

A computer program is written to calculate the following system measures for the three models : $M / M_{(a,b)} / 2$, overloading (Model I) and Additional server model (Model II) and the results are tabulated.

C	-	The average cost per unit time of the system
Lq	-	Mean queue length
Bs	-	The average number of batches waiting in the queue
P_1	-	Probability that only one server is busy.
P_2	-	Probability that two servers are busy.

Numerical results in table 2.2.2 show that the average cost per unit time of the over loading system and additional server system are lesser than that of the two-servers model. Considering the mean number of members waiting in the queue, we find that the size of the waiting batch is 0 for all the three models, when the arrival rate is less.

$a = 20$; $b = 30$; $\mu = 1$

b′	λ	Lq	λ	Lq
30		9.7448		23.4839
40	6	9.7441	18	20.0221
45		9.7439		19.1449
30		12.8708		60.7526
40	12	12.6466	24	32.8823
45		12.5648		29.1829

Table 2.1.a

$a = 10$; $b = 30$; $\mu = 1.2$

b′	λ	Lq	λ	Lq
40		9.3350		26.6087
50	12	9.2903	24	24.2439
60		9.2561		22.8871
40		16.3102		42.5442
50	18	15.7769	30	35.1048
60		15.4161		31.6270

Table 2.1.b

$a = 20$; $b = 50$; $\mu = 1.2$

b′	λ	Lq	λ	Lq
50		15.9431		54.0368
70	20	15.8324	40	42.4533
90		15.7645		38.6933
50		28.3098		130.9715
70	30	26.5019	50	66.5369
90		25.6423		54.9644

Table 2.1.c

d = 25, e = 3, f = 20

a=20, b=50, μ=1	b'	λ	C	L_q	B_s	P_1	P_2
$M / M_{(a,b)} / 2$ (Model 1)	-	10	68.8442	9.5171	0	0.4148	0.04230
Overloading (Model 2.1)	70	10	37.6186	10.9624	0	0.4665	-
	80	10	37.6184	10.9599	0	0.4665	-
Additional Server (Model 2.2)	-	10	37.6185	10.9587	0	0.4665	0.00002
Model 1	-	20	72.3939	10.2406	0	0.4933	0.2364
Model 2.1	70	20	45.6907	19.5096	0	0.7419	-
	80	20	45.6424	19.3818	0	0.7415	-
Model 2.2	-	20	45.6044	19.0511	0	0.7379	0.0038
Model 1	-	30	80.3724	13.0312	0	0.4066	0.4583
Model 2.1	70	30	53.1178	35.3536	0.7	0.8753	-
	80	30	52.6842	34.0625	0.7	0.8733	-
Model 2.2	-	30	52.4040	30.9047	0.6	0.8429	0.0307
Model 1	-	40	86.7689	18.3800	0	0.2975	0.6327
Model 2.1	70	40	61.9842	60.5699	1	0.9466	-
	80	40	60.2240	54.9488	1	0.9369	-
Model 2.2	-	40	58.6410	42.9545	0.8	0.8406	0.0959

Table 2.2.2

Chapter III

CHAPTER - III

$M / M_{(a,b)} / 1$ QUEUEING MODEL WITH ADDITIONAL SERVER AND VACATION

In this chapter, Markovian general bulk service queueing system $M / M_{(a, b)} / 1$ with servers vacation and additional server facility is considered. We have discussed two models in this chapter and these models are the generalizations of Model II of Chapter II.

In both the models we have assumed that if the number of customers in the queue exceeds a preassigned value ***Nb*** (where **N** is any positive integer), an additional server is employed instantaneously. At the end of his service if he finds the queue length is less than *Nb*, the additional server is dropped from the system.

Mutiple vacation is introduced in Model I and single vacation is considered in Model II.

The matrix-geometric method is used to obtain the steady-state probabilities and the expected queue length for both the models. The result of Model II of Chapter II are obtained as a special case.

SECTION 3.1
MODEL I
REPEATED VACATION AND ADDITIONAL SERVER EMPLOYED INSTANTANEOUSLY

3.1.1 PROBLEM DESCRIPTION

We consider $M / M_{(a, b)} / 1$ queueing system, in which the regular server, processes the customers in batches according to the general bulk service rule with maximum (a, b) in the batch as long as there are atleast ***a*** customers waiting in the queue.

If the server completes a service and finds less than the quorum of ***a*** customers in the queue he leaves the system for a random period of time called vacation, which is exponentially distributed with parameter α. On returning to the system if the server finds less than ***a*** waiting customers, he immediately takes another vacaton. The server continues his vacation in this manner until he finds ***a*** customers in the system.

The service time distribution of the regular server follows a negative exponential distribution with parameter μ.

When the queue size exceeds ***Nb***, an additional server is employed instantaneously and he also also serves the customers according to the general bulk service rule. The service time distribution for the additional server is also a negetive exponential distribution with parameter μ_1.

Customers arrive at the system in a Poisson stream with mean rate λ.

3.2.1 MATHEMATICAL MODEL

The above $M / M_{(a, b)} / 1$ queueing model can be studied by a continuous time Markov chain on the state space $\{\underline{i}, i \geq 0\}$ where

$$i = (i,j,k) \qquad 0 \le i \le Nb, \quad j,k = 0,1$$

$$= (i,j,k) \qquad i > Nb, \qquad k = 1,\ j = 0,1$$

The state (i,j,k) represents that

- i customers are in the waiting line
- the regular server is on vacation when $j = 0$
- the regular server is busy when $j = 1$
- the additional server is busy in the system when $k = 1$
- the additional server is not in the system when $k = 0$

Hence for example, (0,1,1) represents there are no customers in the queue, the regular server is busy with his customer and the additional server is also busy in the system.

To facilitate the representation of the infinitesimal generator Q_M of the continuous

Markov chain with the above state space, we define the following submatrices,

Ai, Bi; i = 0,1,2,3,4.

The matrices A_0, A_1, A_2 are square matrices of order 2, defined by,

$$A_0 = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} \qquad . \qquad A_1 = \begin{bmatrix} -\lambda - \mu_1 - \alpha & 0 \\ 0 & -\lambda - \mu - \mu_1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} \mu_1 & \alpha \\ 0 & \mu + \mu_1 \end{bmatrix}$$

The matrix A_3 of order 2 X 4 is defined by

$$A_3 = \begin{bmatrix} 0 & 0 & \mu_1 & \alpha \\ 0 & 0 & 0 & \mu+\mu_1 \end{bmatrix}$$

The matrix A_4 is a square matrix of order 4, defined by $A_4 = \lambda I$ where I is a unit matrix of order 4.

The matrix B_0, B_1, B_2 and B_3 are square matrices of order 4 defined by

$$B_0 = \begin{bmatrix} -\lambda & 0 & 0 & 0 \\ \mu & -\lambda-\mu & 0 & 0 \\ \mu_1 & 0 & -\lambda-\mu_1 & 0 \\ 0 & \mu_1 & \mu & -\lambda-\mu-\mu_1 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} -\lambda-\alpha & 0 & 0 & 0 \\ 0 & -\lambda-\mu & 0 & 0 \\ \mu_1 & 0 & -\lambda-\alpha-\mu_1 & 0 \\ 0 & \mu_1 & 0 & -\lambda-\mu-\mu_1 \end{bmatrix}$$

$$B_2 = \begin{bmatrix} 0 & \alpha & 0 & 0 \\ 0 & \mu & 0 & 0 \\ 0 & 0 & 0 & \alpha \\ 0 & 0 & 0 & \mu \end{bmatrix} \qquad B_3 = \begin{bmatrix} 0 & 0 & \lambda & 0 \\ 0 & 0 & 0 & \lambda \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

The matrix B_4 of order 4 X 2 is defined by

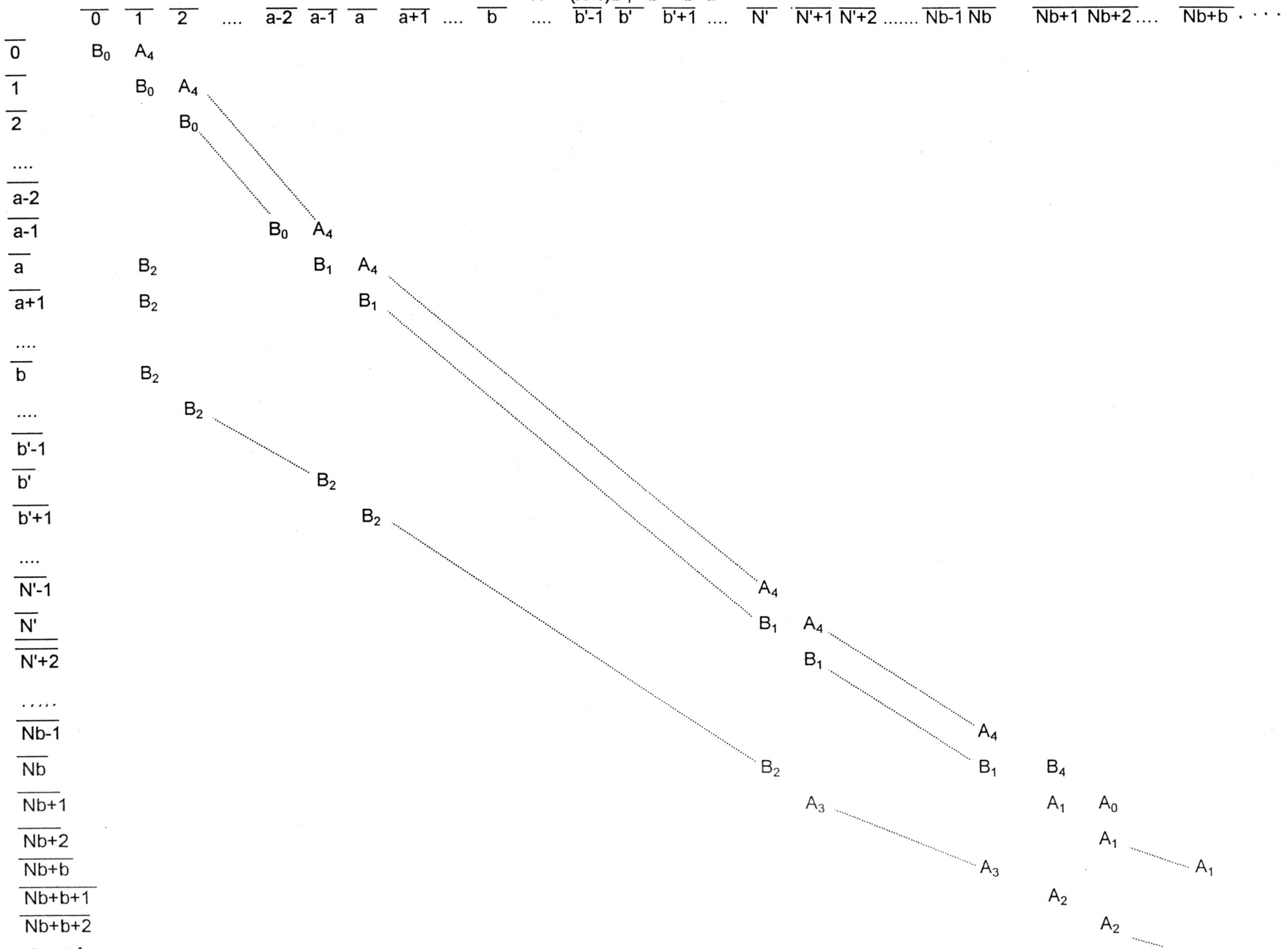
$$B_4 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \lambda & 0 \\ 0 & \lambda \end{bmatrix}$$

The infinitesimal generator Q_M of the continuous time Markov chain, with the state space defined by $\{\underline{i}\, ;\, i \geq 0\}$ has the block partitioned structure as shown below :

Some of the illustrations of the transition from $\underline{i}$ to $\underline{j}$ are explained in the Appendix I which is given at the end of this section.

QM - Multiple Vacation

$\overline{N'} = \overline{(N-1)b}$, $\overline{b'} = \overline{b+a}$



THE STEADY-STATE PROBABILITY VECTOR AND

THE STEABILITY CONDITION

Let $\underline{X}$ be the steady-state probabilities associated with Q_M such that

$$\underline{X}\, Q_M = 0$$

$$\underline{X}\, \underline{e} = 1$$

where $\underline{e}$ is the column vector of dimensions (Nb + 1) 4 + 2 with all elements equal to 1.

The vector $\underline{X}$ is partitioned as

$$\underline{X} = (\underline{X}_0, \underline{X}_1, \ldots\ldots\ldots\ldots\ldots \underline{X}_{Nb}, \underline{X}_{Nb+1}, \ldots\ldots)$$

where $\underline{X}_i = (P_{i\,00}, P_{i\,10}, P_{i\,01}, P_{i\,11}), \quad 0 \le i \le Nb$

$= (P_{i\,01}, P_{i\,11}), \quad i > Nb$

and P_{ijk} = the steady- state probability that the system is at the state (i, j, k).

Thus for $0 \le i \le Nb$, $\underline{X}_i$ is of type 1 X 4 and

for $i > Nb$, $\underline{X}_i$ is of type 1 X 2.

Following **Neuts [21]**, the solution of $\underline{X}_i$, for $i \ge Nb+1$ is of the form

$$\underline{X}_i = \underline{X}_{Nb+1} R^{i-(Nb+1)}, \quad i \ge Nb+1 \qquad (3.1.1)$$

where the matrix R of order 2 X 2 is the minimal non-negative solution of the matrix equation

$$A_0 + RA_1 + R^{b+1}A_2 = 0 \qquad (3.1.2)$$

The equation (3.1.2) is obtained by expanding the equation $\underline{X}Q_M = 0$ and substituting for $\underline{X}_i$, $i \ge Nb+1$ from (3.1.1).

Following theorem 1 of **Neuts [21]**, the matrix **R** can be computed numerically by using the recurrence relations,

$$R(0) = 0$$

$$R(n+1) = -(A_0A_1^{-1} + R^{b+1}(n)\, A_2A_1^{-1}), \qquad n \geq 0$$

(i.e), R is the limit of the monotonically increasing sequence of matrices $\{R_n, n \geq 0\}$

STEABILITY CONDITION

The condition under which the solution exists is discussed in this section.

Consider the infinitesimal generator

$$A = A_0 + A_1 + A_2$$

Then A is a square matrix of order 2 given by

$$A = \begin{bmatrix} -\alpha & \alpha \\ 0 & 0 \end{bmatrix}$$

We find

$$A\underline{e} = 0 \text{ where } \underline{e} = (1,1)^T$$

Thus A is irreducible.

Therefore there exists a unique row vector

$$\underline{\pi} = (\pi_1, \pi_2) \text{ such that}$$

$$\underline{\pi}\, A = 0 \text{ and } \underline{\pi}\, \underline{e} = 1$$

This implies $\pi_0 = 0$ and $\pi_1 = 1$

Following **Netus [21]** the system is stable if and only if $\varphi'(1)$ is greater than zero, where

$$\varphi(z) = \underline{\pi}\, (A_0 + ZA_1 + z^{b+1}A_2)\, \underline{e}$$

and $$\varphi'(1) = \underline{\pi}\, (A_1 + (b+1)\, A_2)\underline{e}$$

$$> 0$$

(i.e.) $\underline{\pi}\, b\, A_2\, \underline{e} - \underline{\pi}\, A_0\, \underline{e} > 0$ (Since $(A_0 + A_1 + A_2)\, \underline{e} = 0$)

(i.e.) $\underline{\pi}\, b\, A_2\, \underline{e} > \underline{\pi}\, A_0\, \underline{e}$

where $\underline{\pi} = (0, 1)$

Substituting for A_2 and A_0 we get,

$$b(\mu + \mu_1) > \lambda$$

Now the matrix R decides the vector $\underline{X}_i$, $i \geq Nb+1$ if $\dfrac{\lambda}{b(\mu + \mu_1)} < 1$.

To determine the remaining vectors $\underline{X}_i$, $0 \leq i \leq Nb + 1$ we define Q_M^* by

Qm* - Multiple Vacation

$\overline{N'} = \overline{(N-1)b}$, $\overline{b'} = \overline{b+a}$

	$\overline{0}$	$\overline{1}$	$\overline{2}$	....	$\overline{a-2}$	$\overline{a-1}$	$\overline{a}$	$\overline{a+1}$	....	$\overline{b}$	....	$\overline{b'-1}$	$\overline{b'}$	$\overline{b'+1}$	....	$\overline{N'-1}$	$\overline{N'}$	$\overline{N'+1}$	$\overline{N'+2}$	....	$\overline{Nb-1}$	$\overline{Nb}$	$\overline{Nb+1}$
$\overline{0}$	B_0	A_4																					
$\overline{1}$		B_0	A_4																				
$\overline{2}$			B_0																				
....																							
$\overline{a-2}$						A_4																	
$\overline{a-1}$						B_0	A_4																
$\overline{a}$	B_2						B_1	A_4															
$\overline{a+1}$								B_1															
....																							
$\overline{b}$	B_2																						
....																							
$\overline{b'-1}$																							
$\overline{b'}$							B_2																
$\overline{b'+1}$								B_2															
....																							
$\overline{N'-1}$																	A_4						
$\overline{N'}$																	B_1	A_4					
$\overline{N'+1}$																		B_1	A_4				
$\overline{N'+2}$																			B_1				
....																							
$\overline{Nb-1}$																						A_4	
$\overline{Nb}$																	B_2					B_1	B_4
$\overline{Nb+1}$																		A_3	RA_3	....		$R^{b-1}A_3$	$A_1+R^bA_2$

Q_M^* is an infinitesimal generator.

(i.e) $Q_M^* \underline{e} = 0$

To prove $Q_M^* \underline{e} = 0$ it is enough to consider the (last row of Q^*)$\underline{e}$, since the other rows of Q^* are identical to that of Q_M and Q_M is an infinitesimal generator.

$$(\text{Last row of } Q^*)\,\underline{e} = \sum_{i=0}^{b-1} R^i A_3 \underline{e} + A_1 \underline{e} + R^b A_2 \underline{e}$$

where $(\underline{e})^T = (1,1)$

$$(\text{i.e.}) \quad = \frac{I - R^b}{I - R} A_3 \underline{e} + A_1 \underline{e} + R^b A_2 \underline{e}$$

$$= (I - R)^{-1}\left[(I - R^b) A_3 \underline{e} + (I - R)(A_1 + R^b A_2)\underline{e}\right]$$

$$= (I - R)^{-1}\left[(I - R^b) A_3 \underline{e} + (A_1 + R^b A_2)\underline{e} + A_0 \underline{e}\right]$$

(Since $A_0 \underline{e} = -R\, A_1 \underline{e} - R^{b+1} A_2 \underline{e}$)

By the definition of A_2 and A_3,

$A_2 \underline{e} = A_3 \underline{e}$

Substituting this we get,

(Last row of Q^*) $\underline{e}$

$= (I - R)^{-1} [(A_1 + A_2 + A_3)\, \underline{e}\,]$

$= 0$

Therefore Q^* is an infinitesimal generator and hence irreducible.

Now we have to calculate the remaining vectors $\underline{X}_i$ with $0 \leq i \leq Nb+1$

Let $\underline{X}^* = (\underline{X}_0, \underline{X}_1, \ldots\ldots\ldots \underline{X}_{Nb+1})$,

then $\underline{X}^*$ satisfies the equations

$$\underline{X}^* Q_M^* = 0$$

and $$\sum_{i=0}^{\infty} \underline{X}_i \, \underline{e} = 1$$

Expanding $\underline{X}^* Q_M^* = 0$ we get,

$$\left.\begin{array}{l}
\underline{X}_0 B_0 + \sum_{i=a}^{b} \underline{X}_i B_2 = 0 \\
\underline{X}_0 A_4 + \underline{X}_1 B_1 + \underline{X}_{b+1} B_2 = 0 \\
------- \\
------- \\
\underline{X}_{a-1} A_4 + \underline{X}_a B_1 + \underline{X}_{a+b} B_2 = 0 \\
\underline{X}_{(N-1)b} A_4 + \underline{X}_{(N-1)b+1} B_1 + \underline{X}_{Nb+1} A_3 = 0 \\
\underline{X}_{(N-1)b+1} A_4 + \underline{X}_{(N-1)b+2} B_1 + \underline{X}_{Nb+1} R A_3 = 0 \\
------- \\
------- \\
\underline{X}_{Nb-1} A_4 + \underline{X}_{Nb} B_1 + \underline{X}_{Nb+1} R^{b-1} A_3 = 0 \\
\underline{X}_{Nb} B_4 + \underline{X}_{Nb+1} (A_1 + R^b A_2) = 0
\end{array}\right\} \quad \text{(I)}$$

$$\sum_{i=0}^{\infty} \underline{X}_i \, \underline{e} = 1 \text{ implies}$$

$$\sum_{0}^{Nb} \underline{X}_i \, \underline{e} + \sum_{i=Nb+1}^{\infty} \underline{X}_{Nb+1} R^{i-(Nb+1)} \underline{e} = 1$$

$$\sum_{0}^{Nb} \underline{X}_i \, \underline{e} + \underline{X}_{Nb+1} (I - R)^{-1} \underline{e} = 1 \qquad \text{(II)}$$

Substituting for A_i, B_i, i = 0 to 4 in I we get,

4(Nb+1)+2 equations in 4(Nb+1)+2 unknowns,

P_{ijk}'s with $0 \le i \le Nb$, $0 \le j \le 1$, $0 \le k \le 1$; $i = Nb+1$, $j = 0,1$, $k = 1$.

Replacing the first 4(Nb+1) rows of the last column of Q_M^* by 1, the last two rows by $(I - R)^{-1}\underline{e}$ and correspondingly the right hand side by (0,0, ... 1) of Q_M^*, (I) become

$$\underline{X}^* Q_{M1}^* = (0,0, \ldots 1)$$

where Q_{M1}^* is the matrix obtained from Q_M^* by making the replacement mentioned above.

Now the last row of the inverse matrix (Q_{M1}^*) will give the required probability vectors X_i' s for $0 \leq i \leq Nb+1$.

MEAN QUEUE LENGTH

Next we shall calculate the expected number of customers waiting in the queue. Recalling the definition of P_{ijk} with $i \geq 0$; $j = 0,1$ and $k = 0,1$, the expected queue length Lq of this model is given by,

$$L_q = \sum_{k=0}^{1}\sum_{j=0}^{1}\sum_{i=0}^{Nb}(i\, P_{ijk}) + \sum_{j=0}^{1}\sum_{i=Nb+1}^{\infty}(i\, P_{ij1})$$

$$\underline{X}_i \underline{e} = \sum_{k=0}^{1}\sum_{j=0}^{1} P_{ijk}\ , \text{ for } 0 \leq i \leq Nb$$

$$= \sum_{j=0}^{1} P_{ij1}\ , \qquad \text{for } i > Nb$$

Thus

$$Lq = \sum_{i=0}^{Nb} i\, \underline{X}_i \underline{e} + \sum_{i=Nb+1}^{\infty} i\, \underline{X}_i \underline{e}$$

Substituting for

$$\underline{X}_i = \underline{X}_{Nb+1} R^{i-(Nb+1)} \quad \text{for } i > Nb,$$

We get,

$$Lq = \left[\sum_{i=0}^{Nb} i\, \underline{X}_i + \underline{X}_{Nb+1} \sum_{i=Nb+1}^{\infty} i\, R^{\,i-(Nb+1)} \right] \underline{e}$$

Simplifying,

$$Lq = \left[\sum_{i=0}^{Nb} i\, \underline{X}_i + \underline{X}_{Nb+1} \frac{(Nb(I-R)+I}{(I-R)^2} \right] \underline{e}$$

substituting for $\underline{X}_i$ for $0 \le i \le Nb$, we obtain the values for Lq.

Numerical values for the steady-state vectors $\underline{X}_i$, $i > 0$ and the value for the mean queue length are calculated.

It is verified that the normalizing condition

$$\sum_{i=0}^{\infty} \underline{X}_i \, \underline{e} = 1 \text{ is true.}$$

NUMERICAL ANALYSIS FOR MULTIPLE VACATION MODEL

The matrix R and the probablity vectors $\underline{X}i$ $(0 \le i \le Nb)$ for $M / M_{(a,b)} / 1$ model with additional server and multiple vacation when N=2, a=2, b=4, λ=5, μ=1, α=.5, μ_1=1.2 are given below:

$$\mathbf{R}=\begin{bmatrix} .8079212 & .0624070 \\ 0 & .7862624 \end{bmatrix}$$

X_0 = [.0051033 .0243235 .0009942 .0061639]

$\underline{X}_1$ = [.0116969 .0302888 .0022332 .0088752]

$\underline{X}_2$ = [.0109972 .0358264 .0016666 .0116807]

$\underline{X}_3$ = [.0102688 .0411146 .0012437 .0142692]

$\underline{X}_4$ = [.0095377 .0462016 .0009282 .0165110]

$\underline{X}_5$ = [.0088363 .0450418 .0007591 .0327023]

$\underline{X}_6$ = [.0081683 .0454178 .0006202 .0394148]

$\underline{X}_7$ = [.0075385 .046634 .0005166 .0440764]

$\underline{X}_8$ = [.006949 .0483489 .0004391 .0473135]

$\underline{X}_9$ = [.0003711 .0694163]

The values for expected queue lengh Lq are tabulated for various parametric values in tables 3.1.1 to 3.1.4 for the multiple vacation model.

In the table 3.1.1 it is found that the mean queue length Lq increases as the arrival rate λ increases and as $1/\alpha$ increases.

N=1, a=4, b=15, μ=1, μ_1=1.

	L_q			
λ \ $\frac{1}{\alpha}$	.33	2	10	$\rho = \frac{\lambda}{b(\mu + \mu_1)}$
10	2.0626	2.8803	4.3878	.33
15	6.8524	8.1340	12.8625	.5
20	16.7740	18.4408	31.5896	.66
25	43.5079	45.4384	70.8756	.83

3.1.1

The table 3.1.2 shows that the queue length decreases if we increase the service rate of the additional server.

N=1, a=4, b=15, α=3, μ=1.

λ \ μ_1	1	1.2	1.5
10	2.0626	1.7240	1.3706
15	6.8524	5.5267	4.2281
20	16.7740	12.7214	9.2057
25	43.5079	27.6800	17.8139

3.1.2

The conclusions made from tables 3.1.1 and 3.1.2 are verified for different values of a,b and N in the following tables 3.1.3, 3.1.4.

$N=2$, $a=4$, $b=7$, $\mu=1$, $\mu_1 = 1.2$.

	L_q			
λ \ $\frac{1}{\alpha}$	.33	2	10	$\rho = \frac{\lambda}{b(\mu + \mu_1)}$
5	0.8206	1.4875	2.008	.32
8	4.6339	5.428	6.1072	.51
10	9.6387	10.3275	11.4686	.64
13	27.586	28.08	30.688	.84

3.1.3

$N=3$, $a=2$, $b=4$, $\mu=1$, $\mu_1=1.2$.

	L_q	
λ \ $\frac{1}{\alpha}$	2	$\rho = \frac{\lambda}{b(\mu + \mu_1)}$
3	.97760	.34
5	5.0999	.56
7	15.20747	.79

3.1.4

ILLUSTRATION

APPENDIX - I

Transitions from 0, a-1, b to 0

		0			
		000	010	001	011
	000	$-\lambda$	0	0	0
0	010	μ	$-(\lambda+\mu)$	0	0
	001	μ_1	0	$-(\lambda+\mu_1)$	0
	011	0	μ_1	μ	$-(\lambda+\mu+\mu_1)$
	a-1 00	0	0	0	0
a-1	a-1 10	0	0	0	0
	a-1 01	0	0	0	0
	a-1 11	0	0	0	0
	b 00	0	α	0	0
b	b 10	0	μ	0	0
	b 01	0	0	0	α
	b 11	0	0	0	μ

Transitions from <u>a-1</u>, <u>a</u>, <u>b+a</u> to <u>a</u>

		a: 000	a: 010	a: 001	a: 011
	a-1 00	λ	0	0	0
a-1	a-1 10	0	λ	0	0
	a-1 01	0	0	λ	0
	a-1 11	0	0	0	λ
	a 00	$-\lambda-\alpha$	0	0	0
a	a 10	0	$-\lambda-\mu$	0	0
	a 01	μ_1	0	$-\lambda-\alpha-\mu_1$	0
	a 11	0	μ_1	0	$-\lambda-\mu-\mu_1$
	b+a 00	0	α	0	0
b+a	b+a 10	0	μ	0	0
	b+a 01	0	0	0	α
	b+a 11	0	0	0	μ

Transitions from <u>Nb</u>, <u>Nb+1</u>, <u>(N+1)b+1</u> to <u>Nb+1</u>

		Nb+1: Nb+1 01	Nb+1: Nb+1 11
	Nb 00	0	0
Nb	Nb 10	0	0
	Nb 01	λ	0
	Nb 11	0	λ
Nb+1	Nb+1 01	λ	0
	Nb+1 11	0	λ
(N+1)b+1	(N+1)b+1 01	μ_1	α
	(N+1)b+1 11	0	$\mu+\mu_1$

SECTION 3.2

MODEL II

SINGLE VACATION AND ADDITIONAL SERVER EMPLOYED INSTANTANEOUSLY

3.2.1 MODEL DESCRIPTION

This model differs from the model I in that on returning to the main system from vacation the regular server will join the system and remain idle in the system if the number of customers in the queue is less than **a**. He takes another vacation after having served at least one batch. That is only single vacation is taken each time. The other assumptions are exactly the same as in Model I.

3.2.2 MATHEMATICAL MODEL

The single vacation additional server queueing model $M / M_{(a,\ b)} / 1$ can be studied by a continuous time Markovian chain on the state space $\{\underline{i}, i \geq 0\}$

where $\underline{i} = (i, j, k)$, $j = 0,1,2$, $k = 0,1$ for $0 \leq i \leq a\text{-}1$

$= (i, j, k)$, $j = 0,2$, $k = 0,1$ for $a \leq i \leq Nb$

$= (i, j, k)$, $j = 0,2$, $k = 1$ for $i > Nb$

The state (i, j, k) represents that there are

- i customers are waiting in the queue
- the regular server is on vacation when $j = 0$
- the regular server is idle when $j = 1$
- the regular server is busy when $j = 2$
- the additional server is busy in the system when $k = 1$
- the additional server is not in the system when $k = 0$.

Thus (0,1,1) represents that there are no customers in the queue, the regular server is idle and the additional server is busy in the system.

The infinitesimal generator Qs of the continuous time Markov chain is obtained by adopting the similar argument of Model I.

The matrices A_i, i = 0 to 4 and B_j, i = 1 to 4 are the same as the matrices A_i and B_i's given in Model I.

The other matrices A_i, i = 5,6,7 and B_j, j = 5,6 involved in matrix Qs are listed here

$A_5 = \lambda I$ where I is an identity matrix of order 6.

The matrix A_6 of order 6 X 6 is defined by

$$A_6 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \lambda \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The matrix A_7 of order 6 X 4 is defined by

$$A_7 = \begin{bmatrix} \lambda & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \lambda & 0 & 0 \\ 0 & 0 & \lambda & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda \end{bmatrix}$$

The matrix B_5 of order 6 X 6 is defined by

$$B_5 = \begin{bmatrix} -\lambda-\alpha & \alpha & 0 & 0 & 0 & 0 \\ 0 & -\lambda & 0 & 0 & 0 & 0 \\ \mu & 0 & -\lambda-\mu & 0 & 0 & 0 \\ \mu_1 & 0 & 0 & -\lambda-\mu_1-\alpha & \alpha & 0 \\ 0 & \mu_1 & 0 & 0 & -\lambda-\mu_1 & 0 \\ 0 & 0 & \mu_1 & \mu & 0 & -\lambda-\mu-\mu_1 \end{bmatrix}$$

The matrix B_6 of order 4 X 6 is defined by

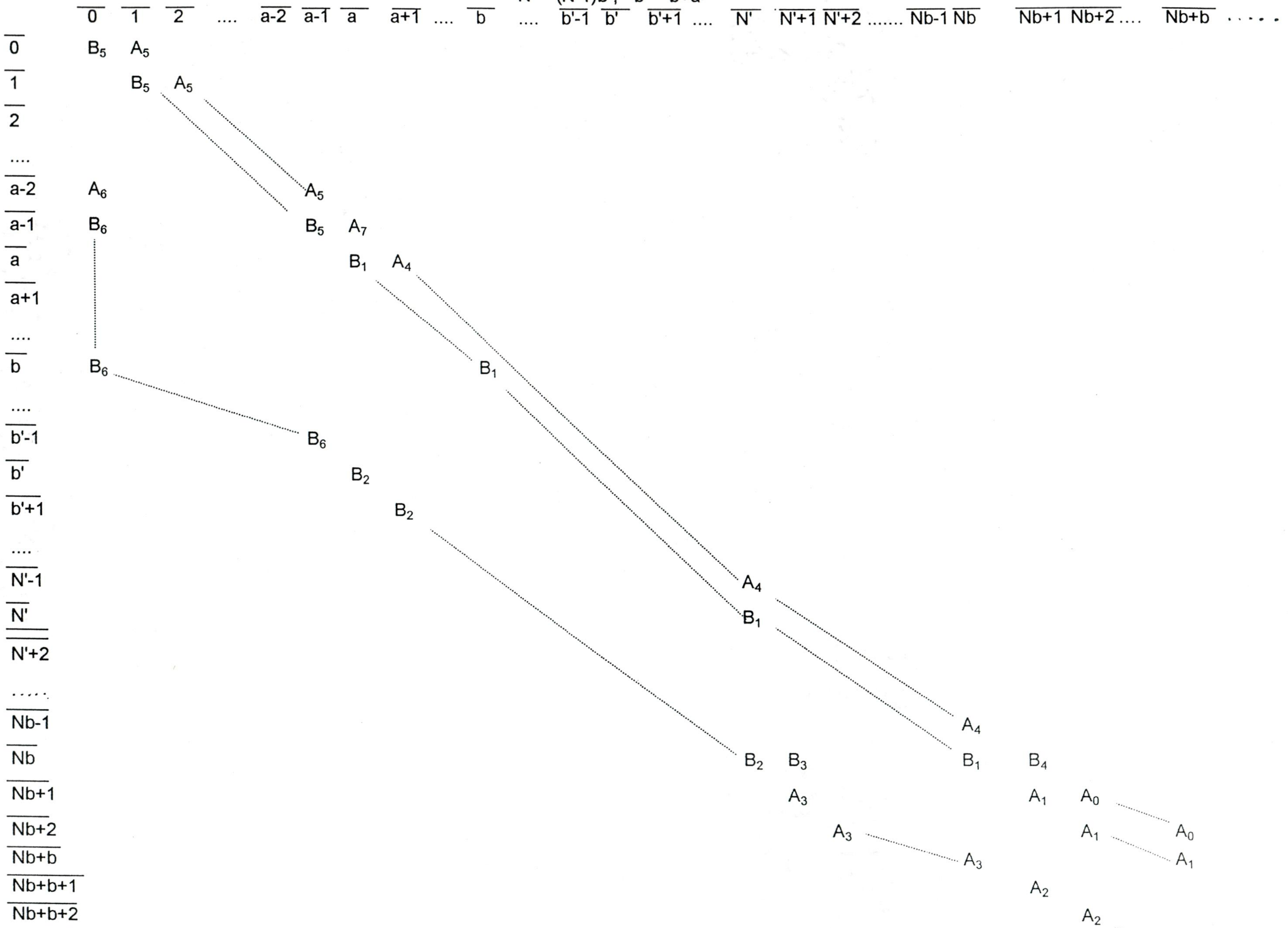
$$B_6 = \begin{bmatrix} 0 & 0 & \alpha & 0 & 0 & 0 \\ 0 & 0 & \mu & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha \\ 0 & 0 & 0 & 0 & 0 & \mu \end{bmatrix}$$

Now, the infinitesimal generator Qs of the continuous time Markov chain, with state space defined in the beginning of this section, has the block partitioned structure, as shown below.

Some of the illustrations of the transitions from i to j are explained in the Appendix II which is given at the end of this section.

Qs - Single Vacation

$\overline{N'} = \overline{(N-1)b}$, $\overline{b'} = \overline{b+a}$



THE STEADY-STATE PROBABILITY VECTOR AND THE STEABILITY CONDITION

Let P_{ijk} denote the steady state probability that the system is at the state (i, j, k).

Let $\underline{X}_i = (P_{i00}, P_{i10}, P_{i20}, P_{i01}, P_{i11}, P_{i21})$ for $0 \le i \le a - 1$

$= (P_{i00}, P_{i20}, P_{i01}, P_{i21})$ for $a \le i \le Nb$

$= (P_{i01}, P_{i21})$ for $i > Nb$.

Let $\underline{X} = (\underline{X}_0, \underline{X}_1, \ldots\ldots, \underline{X}_{a-1}, \underline{X}_a, \ldots\ldots\ldots \underline{X}_{Nb}, \underline{X}_{Nb+1}, \ldots\ldots\ldots\ldots)$

Thus the vector $\underline{X}$ satifies the equations

$$\underline{X}\,Qs = 0$$

and $\underline{X}\,\underline{e} = 1$

where $\underline{e}$ is a column vactor of suitable dimension with all the elements equal to one.

As in the case of model I, the solution of the form

$$\underline{X}_i = \underline{X}_{Nb+1} R^{i-(Nb+1)} \quad \text{for } i \ge Nb+1$$

exists (**Neuts [21]**) where R is the unique non negative solution of the matrix equation,

$$A_0 + RA_1 + R^{b+1}A_2 = 0$$

with spectral radius is less than one.

Since this equation is exactly the same as in the case of model I the condition under which the matrix R satisfying the equation

$$A_0 + RA_1 + R^{b+1}A_2 = 0 \quad \text{exists},$$

~~We have Qs* is irreducible~~

If $\dfrac{\lambda}{b(\mu+\mu_1)} < 1$

Defining Qs^*,

Qs* - Single Vacation

$\overline{N'} = \overline{(N-1)b}$, $\overline{b'} = \overline{b+a}$

	$\overline{0}$	$\overline{1}$	$\overline{2}$	....	$\overline{a-2}$	$\overline{a-1}$	$\overline{a}$	$\overline{a+1}$	....	$\overline{b}$	....	$\overline{b'-1}$	$\overline{b'}$	$\overline{b'+1}$	....	$\overline{N'-1}$	$\overline{N'}$	$\overline{N'+1}$	$\overline{N'+2}$	....	$\overline{Nb-1}$	$\overline{Nb}$	$\overline{Nb+1}$
$\overline{0}$	B_5	A_5																					
$\overline{1}$		B_5	A_5																				
$\overline{2}$																							
....																							
$\overline{a-2}$	A_6					A_5																	
$\overline{a-1}$	B_6					B_5	A_7																
$\overline{a}$							B_1	A_4															
$\overline{a+1}$																							
....																							
$\overline{b}$	B_6									B_1													
....																							
$\overline{b'-1}$						B_6																	
$\overline{b'}$							B_2																
$\overline{b'+1}$								B_2															
....																							
$\overline{N'-1}$																B_1	A_4						
$\overline{N'}$																	B_1	A_4					
$\overline{N'+1}$																		B_1	A_4				
$\overline{N'+2}$																			B_1				
....																							
$\overline{Nb-1}$																B_2						A_4	
$\overline{Nb}$																	B_2	B_3				B_1	B_4
$\overline{Nb+1}$																		A_3	RA_3	....		$R^{b-1} A_3$	$A_1 + R^b A_2$

We have Q_s^* is irreducible

Now we have to calculate the vector $\underline{X}_i$, $0 \le i \le Nb+1$ satisfying the equations

$$\underline{X}^* Q_s^* = 0 \qquad (3.2.1)$$

$$\underline{X}^* \underline{e} = 1 \qquad (3.2.2)$$

Expanding (3.2.1) we get,

$$\underline{X}_0 B_5 + \underline{X}_{a-1} A_6 + \underline{X}_a B_6 + \ldots + \underline{X}_b B_6 = 0$$
$$\underline{X}_0 A_5 + \underline{X}_1 B_5 + \underline{X}_{b+1} B_6 = 0$$

$$\underline{X}_{a-2} A_5 + \underline{X}_{a-1} B_5 + \underline{X}_{b+a-1} B_6 = 0$$
$$\underline{X}_{a-1} A_7 + \underline{X}_a B_1 + \underline{X}_{b+a} B_2 = 0$$
$$\underline{X}_a A_4 + \underline{X}_{a+1} B_1 + \underline{X}_{a+b+1} B_2 = 0$$

$$\underline{X}_{(N-1)b-1} A_4 + \underline{X}_{(N-1)b} B_1 + \underline{X}_{Nb} B_2 = 0$$
$$\underline{X}_{(N-1)b} A_4 + \underline{X}_{(N-1)b+1} B_1 + \underline{X}_{Nb} B_3 + \underline{X}_{Nb+1} A_3 = 0$$
$$\underline{X}_{(N-1)b+1} A_4 + \underline{X}_{(N-1)b+2} B_1 + \underline{X}_{Nb+1} R A_3 = 0$$

$$\underline{X}_{Nb-1} A_4 + \underline{X}_{Nb} B_1 + \underline{X}_{Nb+1} R^{b-1} A_3 = 0$$
$$\underline{X}_{Nb} B_4 + \underline{X}_{Nb+1} (A_1 + R^b A_2) = 0$$

Substituting for Ai's and bi's we find that the above set of equations are

$6a+4(Nb+1-a)+2$ in number and $6a+4(Nb-a+1)+2$ unknowns are involved in it.

As in the case of model I replacing the first $6a+4(Nb+1-a)$ rows of the last column of Qs^* by 1, the last two rows by $(I - R)^{-1}\underline{e}$ and the right hand side by (0,0,.....1) we get,

$$\underline{X}^* Q_{S1}^* = (0,0,\ldots..1)$$

where Q_{S1}^* is the replaced matrix.

Thus the values of $\underline{X}_i$, $0 \le i \le Nb+1$ are obtained from the last row of inverse of (Q_{S1}*).

Mean queue Length

Let Lqs denote the expected queue length for this model.

Then,

$$Lq = \sum_{k=0}^{1}\sum_{j=0}^{2}\sum_{i=0}^{a-1} i\, P_{ijk} + \sum_{k=0}^{1}\sum_{j=0,2}\sum_{i=a}^{Nb} i\, P_{ijk} + \sum_{j=0,2}\sum_{i=Nb+1}^{\infty} i\, P_{ijk}$$

$$= \sum_{i=0}^{a-1} i\, \underline{X}_i \underline{e} + \sum_{i=a}^{Nb} i\, \underline{X}_i \underline{e} + \sum_{i=Nb+1}^{\infty} i\, \underline{X}_i \underline{e}$$

where the vector $\underline{X}_i$ of summation 1,2,3 are of dimension 1X6, 1X 4, 1X 2, respectively.

Thus after simplification we find,

$$L_{qs} = \left[\sum_{i=0}^{Nb} i\, \underline{X}_i + \underline{X}_{Nb+1} \frac{(Nb(I-R)+I}{(I-R)^2} \right] \underline{e}$$

NUMERICAL ANALYSIS FOR SINGLE VACATION MODEL

The values for the mean queue length **Lq** are tabulated for various parametric values in table 3.2.1. to 3.2.4 for the single vacation.

In the table 3.2.1 it is found that for the single vacation ~~ean~~ also the mean queue length **Lq** increases as λ increases and as **1/** α increases.

N=1, a=4, b=10, μ=1, μ_1=1.5

	L_q		
λ \ $\frac{1}{\alpha}$	2	10	$\rho = \frac{\lambda}{b(\mu + \mu_1)}$
12	.20455	4.3720	.48
15	3.2494	17.0824	.6
20	20.2245	68.8555	.8

3.2.1

The table 3.2.2 shows that the queue length decreases if we increase the service rate of the additional server.

N=1, a=4, b=10, α=.5, μ=1.

λ \ μ_1	1	1.2	1.5
12	3.4330	1.5160	.20455
15	14.1059	7.7500	3.2494
20	397.462	57.6417	20.2245

3.2.2

The conclusion made from table 3.2.1 and 3.2.2 are verified for different values of a, b when N = 2, 3

N = 2, a=4, b=7, μ=1, μ_1=1.5

	L_q			
λ \ $\frac{1}{\alpha}$	.33	2	10	$\rho = \frac{\lambda}{b(\mu + \mu_1)}$
8	1.5448	2.6285	5.7412	.45
9	2.8996	3.8740	7.9769	.51
12	9.3104	10.0737	17.6922	.68

3.2.3

N=3, a=2, b=4, μ=1, μ_1=1.2.

	L_q	
λ \ $\frac{1}{\alpha}$	2	$\rho = \frac{\lambda}{b(\mu + \mu_1)}$
3	.348851	.34
5	2.5923	.56
7	10.8440	.79

3.2.4

For given parameter values of λ, a, b, μ, μ_1 and N , the expected queue lengh in the single vacation is lesser than that of the multiple vacation for certen values of α's. This is justified in the following table.

The mean queue length for the Multiple Vacation and Single Vacation A Comparison

N = 1, a=4, b=10, μ=1

	Multiple Vacation						Single vacation					
μ_1	1		1.2		1.5		1		1.2		1.5	
λ \ $\frac{1}{\alpha}$	2	ρ	2	ρ	2	ρ	2	ρ	2	ρ	2	ρ
12	9.82	.6	7.70	.54	5.75	.48	3.43	.6	1.51	.54	.204	.48
15	19.7	.75	14.2	.68	9.97	.6	14.1	.75	7.75	.68	3.24	.6
N=2, a=4, b= 7, μ = 1												
5	1.74	.35	1.48	.32	1.21	.28	.510	.35	.365	.32	.238	.28
8	6.69	.57	5.42	.51	4.21	.45	5.06	.57	3.78	.51	2.62	.45
9	9.57	.64	7.57	.58	5.75	.51	7.74	.64	5.68	.58	3.87	.51
12	30.2	.85	19.2	.77	12.8	.68	27.9	.85	16.7	.77	10.0	.68

ILLUSTRATION

APPENDIX - II

Transitions from 0, a-1, a, b to 0 are explained

		000	010	0 020	001	011	021
	000	$-(\lambda+\alpha)$	α	0	0	0	0
	010	0	$-\lambda$	0	0	0	0
	020	μ	0	$-(\lambda+\mu)$	0	0	0
0	001	μ_1	0	0	$-(\lambda+\mu_1+\alpha)$	α	0
	011	0	μ_1	0	0	$-(\lambda+\mu_1)$	0
	021	0	0	μ_1	μ	0	$-(\lambda+\mu+\mu_1)$
	a-1 00	0	0	0	0	0	0
	a-1 10	0	0	λ	0	0	0
a-1	a-1 20	0	0	0	0	0	0
	a-1 01	0	0	0	0	0	0
	a-1 11	0	0	0	0	0	λ
	a-1 21	0	0	0	0	0	0
	a 00	0	0	α	0	0	0
a	a 20	0	0	μ	0	0	0
	a 01	0	0	0	0	0	α
	a 21	0	0	0	0	0	μ
	b 00	0	0	α	0	0	0
b	b 20	0	0	μ	0	0	0
	b 01	0	0	0	0	0	α
	b 21	0	0	0	0	0	μ

Transitions from (N-1)b, (N-1)b+1, Nb, Nb+1 to (N-1)b+1

		(N-1)b+1			
		(N-1)b+1 00	(N-1)b+1 20	(N-1)b+1 01	(N-1)b+1 21
(N-1)b	(N-1)b 00	λ	0	0	0
	(N-1)b 20	0	λ	0	0
	(N-1)b 01	0	0	λ	0
	(N-1)b 21	0	0	0	λ
(N-1)b+1	(N-1)b+1 00	$-\lambda-\alpha$	0	0	0
	(N-1)b+1 20	0	$-\lambda-\mu$	0	0
	(N-1)b+1 01	μ_1	0	$-\lambda-\alpha-\mu_1$	0
	(N-1)b+1 21	0	μ_1	0	$-\lambda-\mu-\mu_1$
Nb	Nb 00	0	0	λ	0
	Nb 20	0	0	0	λ
	Nb 01	0	0	0	0
	Nb 21	0	0	0	0
Nb+1	Nb+1 00	0	0	μ_1	α
	Nb+1 21	0	0	0	$\mu+\mu_1$

Transitions from 0, 1, b+1 to 1

		1					
		100	110	120	101	111	121
0	000	λ	0	0	0	0	0
	010	0	λ	0	0	0	0
	020	0	0	λ	0	0	0
	001	0	0	0	λ	0	0
	011	0	0	0	0	λ	0
	021	0	0	0	0	0	λ
1	100	$-\lambda-\alpha$	α	0	0	0	0
	110	0	$-\lambda$	0	0	0	0
	120	μ	0	$-\lambda-\mu$	0	0	0
	101	μ_1	0	0	$-\lambda-\mu_1-\alpha$	α	0
	111	0	μ_1	0	0	$-\lambda-\mu_1$	0
	121	0	0	μ_1	μ	0	$-\lambda-\mu-\mu_1$
b+1	b+1 00	0	0	α	0	0	0
	b+1 20	0	0	μ	0	0	0
	b+1 01	0	0	0	0	0	α
	b+1 21	0	0	0	0	0	μ

Transitions from a, a+1, a+b+1 to a+1

		a+1			
		a+1 00	a+1 20	a+1 01	a+1 21
a	a 00	λ	0	0	0
	a 20	0	λ	0	0
	a 01	0	0	λ	0
	a 11	0	0	0	λ
a+1	a+1 00	$-\lambda-\alpha$	0	0	0
	a+1 20	0	$-\lambda-\mu$	0	0
	a+1 01	μ_1	0	$-\lambda-\alpha-\mu_1$	0
	a+1 21	0	μ_1	0	$-\lambda-\mu-\mu_1$
a+b+1	a+b+1 00	0	α	0	0
	a+b+1 20	0	μ	0	0
	a+b+1 01	0	0	0	α
	a+b+1 21	0	0	0	μ

Summary and Conclusion

SUMMARY AND CONCLUSION

Single service queueing models with additionl server have been discussed by many authors. ***Kandasamy,P.R. [13]***, has considered the bulk service queueing model in which the regular server serves the customers according to the general bulk service rule. An additional server is appointed instantaneously whenever the queue length exceeds Nb and dropped if the server is free and the queue length is less than Nb. The regular server is allowed to go on vacation whenever he is free and the queue length is less than a. Kandasamy in his thesis has obtained the steady-state condition for the model using the matrix geometric approach. But the expression for the expected queue length **Lq** and the numerical values for it were not obtained for the models in his thesis.

In our work we have discussed the model in detail for both single and multiple vacation and derived the expression for Lq. The numerical values for steady state probability vectors and the expected queue length are calculated for various parameters by writing a Fortran program. The following conclusions are made from the table values.

1. The expected queue length Lq increases as the arrival rate increases.
2. If the server is on vacation for more time the mean queue length increases.
3. By increasing the service rate of the additional server, the queue length can be minimized.
4. The mean queue length for the single vacation model is less than that of the multiple vacation model for certain values of α's.

Aftahb Begam M.I. [1] has considered the additional server queueing model without vacation when $N=2$. The expression for the expected queue length is obtained using the analytical method. The model is compared with the single server model in which batch capacity is increased to b' from b whenever required. A comparision is made by considering the cost occured to maintain the system. We have also discussed this model in detail.

The author suggests for future work that these models can be generalised by appointing more additional servers whenever required. The waiting time distribution for each server can also be calculated.

Bibliography

BIBLIOGRAPHY

1. **Afthab Begum, .M.I.**

   Queueing models with bulk service and vacation, Ph.D., Dissertation, Bharathiar University (1996).

2. **Arora, K.L.**

   Two-server bulk service queueing process, Opns. Res. 12, 286 - 294, (1964).

3. **Audsin Mohana Dhas, D.**

   Markovian general bulk service queueing models, Ph.D. thesis, Bharathiar University, Coimbatore, India. 16 (1989).

4. **Bailey, N.T.J.**

   On queueing process with bulk service, J. Roy Statist. Soc. B16, 80 - 87 (1954).

5. **Bidhi Singh, K.**

   A queueing system with random additional service facility, R.A. I.R.O. Operations Research, Vol. 12, No.3, 311 - 318 (1978).

6. **Borthakur, A. and Medhi, J.**

   A Poisson queue with a general bulk service rule .J. Assam Sci. Soc. XIV, 162 - 167, (1971).

7. **Chaudry, M.L. and J.G.C. Templeton**

   A first course in bulk queues, Wiley, New york (1983).

8. **Chaudry, M.L., Medhi , J., Sim. S.H. and Templeton, J.G.C.**

   On a two-heterogeneous-server Markovian queue with general bulk service rules, Sankhya; The Indian Journal of statistics 49, B. Pt-1, 36 -50, (1987).

9. **Erlang, A.K.**

The theory of probabilities and telephone conversation, Nyt Tidsslerft Matemalik, B.20, 33-39 (1909).

10. **Garg, R.L. and Khanna, S.K.**

Queue dependent additional server queueing problem with arrivals. R.A.I.R.O. Ops. Res. Vol 17 No. 4 P 391 - 398 (1983).

11. **Ghare, P.M.**

Multichannel queueing system with bulk service, Opns, Res, 16, 189 - 192, (1968).

12. **Gross and Harris, C.M.**

Fundamentals of Queueing theory, John Wiley, New york, (2nd ed:) (1985)

13. **Kandasamy, P.R.**

Matrix - Geometric Queueing Algorithmic Approach to some Markovian Queueing and inventory models, Ph.D., Dissertation, Bharathiar University (1990).

14. **Medhi, J. and Borthakur, A.**

On a two-server Markovian queue with a general bulk service rule, Cahiers du C.E.R.O. 14, 151 - 158, (1972).

15. **Medhi, J.**

Waiting time distribution in a Poisson queue with a general bulk service rule, Mgmt. Sci. 21, 777 - 782, (1975).

16. **Medhi, J.**

Further results on waiting time distribution in a Poisson queue under a general bulk service rule, Cahiers du C.E.R.O. 21, 183 - 189 (1979).

17. **Murari, K.**

Some C additional service channels limited space queuing problem, ZAMM, Vol. 51, 157 - 162 (1971)

18. **Nadarajan, R. and Audsio Mohana Dhas, D.**

Multi server general bulk service queue with vacation APJOR Vol. 7 No.2 (1990).

19. **Nadarajan, R. and Subramanian, A.**

A general bulk service queue with server's vacation. Operational research in management system Academic Pub., 127-135 (1984).

20. **Neuts, M.F.**

A general class of bulk queues with Poisson input. Ann Math. Statist 38, 759 - 770, (1967).

21. **Neuts, M.F.**

Matrix-Geometric solutions in stochastic models - an algorithm approach, The John Hopkins, University Press, Baltimore (1981).

22. **Neuts, M.F. and Nadarajan, R.**

A multi-server queue with thresholds for the acceptance of customers into service, Opns. Res 30, 948 - 960, (1982).

23. **Neuts, M.F. and Ramalhoto, M.F.**

A service model in which the server is required to search for customers J.Appl. Prob. 21, 157 - 166. (1984).

24. Phillips, C.R.

A variable channel queueing model with limited number of channels. Thesis, Georgia Institute of Technology (1960).

25. Romani

A queueing model with variable number of channels, Tarbajos de Estadistics, Vol. 18, No.3, 175 - 189 (1957).

26. Saaty, T.

Elements of Queueing theory with applications, McGraw Hill, New York (1961).

27. Sim, S.H. and Templeton, J.G.C.

Computational procedures for steady-state characteristics of a unscheduled multicarrier shuttle systems, Euro. J. Opnl. Res. 12, 190 - 202, (1983).

Program to find Queue length (L_q) for multiple vacation

```
      REAL LA,MU,AL,MU1
      REAL A0(20,20),A1(20,20),A2(20,20),E(20,20),U(20,20)
      REAL C5I(20,20),ZMI(75,75),F(75,75),G(75,75),F1(75,75)
      REAL A1I(20,20),X(20,20),RB(20,20),R1(20,20),RXY(20,20)
      REAL Z(20,20),Y(20,20),RN(20,20),B0(20,20),B1(20,20)
      REAL B2(20,20),B3(20,20),B4(20,20),A3(20,20),A4(20,20)
      REAL D1(20,20),ZM(75,75),C0(20,20),C1(20,20),C23(20,20)
      REAL C3(20,20),C4(20,20),P(20,20),C5(20,20),C6(20,20),Q(20,20)
      REAL C7(20,20),C17(20,20),C18(20,20),C19(20,20),C21(20,20)
      REAL XB(20,20),E4(20,20),LQ(20,20),C20(20,20)
      INTEGER N,B,A,NI,K1,L,L1,L2,T,L3,L4,J1,T1
      PRINT *,'ENTER THE VALUES FOR LA,MU,AL,MU1,A,B,N'
      READ (*,*) LA,MU,AL,MU1,A,B,N
      DO 10 I=1,2
      DO 20 J=1,2
      READ (*,*) A0(I,J),A1(I,J),A2(I,J)
20    CONTINUE
10    CONTINUE
      CALL MATINV(A1,A1I,2)
      CALL MATMUL(A0,A1I,X,2,2,2)
      DO 290 I1=1,2
      DO 290 J1=1,2
290   R1(I1,J1)=-X(I1,J1)
      NI=0
295   CALL MATCHA(R1,RB,2,2)
      DO 300 L=1,B
      CALL MATMUL(RB,R1,P,2,2,2)
300   CALL MATCHA(P,RB,2,2)
      CALL MATMUL(A2,A1I,Z,2,2,2)
      CALL MATMUL(RB,Z,Y,2,2,2)
      CALL MATADD(X,Y,RXY,2)
      DO 310 I1=1,2
      DO 310 J1=1,2
310   RN(I1,J1)=-RXY(I1,J1)
      DO 320 I1=1,2
      DO 320 J1=1,2
      D1(I1,J1)=RN(I1,J1)-R1(I1,J1)
      DIF=ABS(D1(I1,J1))
      IF(DIF.LT.0.001) GO TO 320
      GO TO 340
320   CONTINUE
```

```
340     CALL MATCHA(RN,R1,2,2)
        NI=NI+1
        IF (NI.EQ.75) GO TO 350
        GO TO 295
350     WRITE(*,*) ((RN(I1,J1),J1=1,2),I1=1,2)
        PRINT *,'ENTER THE VALUES FOR MATRICES B0,B1,B2,B3'
        DO 351 I=1,4
        DO 351 J=1,4
351     READ(*,*) B0(I,J),B1(I,J),B2(I,J)
        PRINT *,'ENTER THE VALUES FOR MATRIX B4'
        DO 352 I=1,4
        DO 352 J=1,2
352     READ(*,*) B4(I,J)
        PRINT *,'ENTER THE MATRIX A3'
        DO 353 I=1,2
        DO 353 J=1,4
353     READ (*,*) A3(I,J)
        PRINT *,'ENTER THE MATRIX A4'
        DO 354 I=1,4
        DO 354 J=1,4
354     READ (*,*) A4(I,J)
        K1=(N*B+1)*4+2
        DO 360 I=1,K1
        DO 360 J=1,K1
360     ZM(I,J)=0.0
        DO 365 K=0,A-1
        DO 365 I=1,4
        DO 365 J=1,4
        I1=4*K+I
        J1=4*K+J
365     ZM(I1,J1)=B0(I,J)
        DO 370 K=A,B
        DO 370 I=1,4
        DO 370 J=1,4
        I1=4*K+I
370     ZM(I1,J)=B2(I,J)
        L=N*B
        DO 380 K=0,L-1
        DO 380 I=1,4
        DO 380 J=1,4
        I1=4*K+I
        J1=4*(K+1)+J
380     ZM(I1,J1)=A4(I,J)
```

```
      DO 390 K=A,L
      DO 390 I=1,4
      DO 390 J=1,4
      I1=4*K+I
      J1=4*K+J
390   ZM(I1,J1)=B1(I,J)
      DO 400 I=1,2
      DO 400 J=1,4
      I1=(N*B+1)*4+I
      J1=((N-1)*B+1)*4+J
400   ZM(I1,J1)=A3(I,J)
      DO 410 I=1,4
      DO 410 J=1,2
      I1=(N*B)*4+I
      J1=((N*B)+1)*4+J
410   ZM(I1,J1)=B4(I,J)
      L1=(N-1)*B
      DO 412 K=1,L1
      DO 412 I=1,4
      DO 412 J=1,4
      I1=4*(K+B)+I
      J1=4*K+J
412   ZM(I1,J1)=B2(I,J)
      DO 415 L=1,B-1
      CALL MATCHA(RN,R1,2,2)
      CALL MATMUL(R1,A3,C0,2,2,4)
      CALL MATMUL(R1,R1,C1,2,2,2)
      CALL MATCHA(C1,R1,2,2)
      DO 415 I=1,2
      DO 415 J=1,4
      I1=(N*B+1)*4+I
      J1=((N-1)*B+1+L)*4+J
      ZM(I1,J1)=C0(I,J)
415   CONTINUE
      CALL MATMUL(R1,A2,C3,2,2,2)
      CALL MATADD(A1,C3,C4,2)
      DO 420 I=1,2
      DO 420 J=1,2
      I1=(N*B+1)*4+I
      J1=(N*B+1)*4+J
420   ZM(I1,J1)=C4(I,J)
      DO 425 I=1,2
      DO 425 J=1,2
```

```
          U(I,J)=0.0
425       U(I,I)=1
          DO 426 I=1,2
          DO 426 J=1,2
426       C5(I,J)=U(I,J)-RN(I,J)
          CALL MATINV(C5,C5I,2)
          DO 427 I=1,2
427       E(I,1)=1
          CALL MATMUL(C5I,E,C7,2,2,1)
          L2=(N*B+1)*4
          DO 430 I=1,L2
          T=(N*B+1)*4+2
430       ZM(I,T)=1
          DO 435 I=1,2
          I1=(N*B+1)*4+I
          T=(N*B+1)*4+2
435       ZM(I1,T)=C7(I,1)
          CALL MATIN(ZM,ZMI,T)
          T1=T-2
          DO 436 J=1,T1
436       F(1,J)=ZMI(T,J)
          DO 437 I=1,T1
437       G(I,1)=1
          CALL MATMULT(F,G,F1,1,T1,1)
          L3=N*B+1
          DO 438 O=1,L3
          DO 438 J=1,4
          L4=(O-1)*4+J
          SUM=0
438       SUM=SUM+(O-1)*ZMI(T,L4)
          DO 439 I=1,1
          DO 439 J = 1,2
          J1=(N*B+1)*4+J
          XB(I,J)=ZMI(T,J1)
439       CONTINUE
          DO 440 I=1,2
          DO 440 J=1,2
440       Q(I,J)=(N*B)*C5I(I,J)
          CALL MATMUL(C5I,C5I,C6,2,2,2)
          DO 441 I=1,2
441       E4(I,1)=1
          DO 442 I=1,1
          DO 442 J=1,1
```

```
      CALL MATADD(Q,C6,C17,2)
      CALL MATMUL(XB,C17,C18,1,2,2)
      CALL MATMUL(C18,E4,C19,1,2,1)
442   LQ(I,J)=SUM+C19(I,J)
      CALL MATMUL(XB,C5I,C20,1,2,2)
      CALL MATMUL(C20,E4,C21,1,2,1)
      CALL MATADD(F1,C21,C23,1)
      WRITE(*,*)((C23(I,J),J=1,1),I=1,1)
      WRITE(*,*)((LQ(I,J),J=1,1),I=1,1)
      END
```

Program to find Queue length (L_q) for single vacation (After finding RN)

```
      PRINT *,'ENTER THE VALUES FOR THE MATRICES B1,B2,B3'
      DO 351 I=1,4
      DO 351 J=1,4
351   READ (*,*) B1(I,J),B2(I,J),B3(I,J)
      PRINT *,'ENTER THE VALUES FOR THE MATRIX B4'
      DO 352 I=1,4
      DO 352 J=1,2
352   READ (*,*) B4(I,J)
      PRINT *,'ENTER THE VALUES FOR THE MATRIX A3'
      DO 353 I=1,2
      DO 353 J=1,4
353   READ (*,*) A3(I,J)
      PRINT *,'ENTER THE VALUES FOR THE MATRIX A4'
      DO 354 I=1,4
      DO 354 J=1,4
354   READ (*,*) A4(I,I)
      PRINT *,'ENTER THE VALUES FOR THE MATRICES A6,B5,A5 '
      DO 451 I=1,6
      DO 451 J=1,6
451   READ (*,*) A6(I,J),B5(I,J),A5(I,J)
      PRINT *,'ENTER THE VALUES FOR THE MATRIX A7'
      DO 153 I=1,6
      DO 153 J=1,4
153   READ (*,*) A7(I,J)
      PRINT *,'ENTER THE VALUES FOR THE MATRIX B6'
      DO 154 I=1,4
      DO 154 J=1,6
154   READ (*,*) B6(I,J)
      L5=6*A+4*(N*B-A+1)+2
      DO 355 I=1,L5
      DO 355 J=1,L5
355   Z(I,J)=0.0
      DO 365 K=0,A-1
      DO 365 I=1,6
      DO 365 J=1,6
      I1=6*K+I
      J1=6*K+J
365   Z(I1,J1)=B5(I,J)
      DO 370 K=0,A-2
      DO 370 I=1,6
      DO 370 J=1,6
```

```
      I1=6*K+I
      J1=6*(K+1)+J
370   Z(I1,J1)=A5(I,J)
      DO 375 I=1,6
      DO 375 J=1,6
      I1=6*(A-1)+I
375   Z(I1,J)=A6(I,J)
      DO 380 K=0,B-A
      DO 380 I=1,4
      DO 380 J=1,6
      I1=6*A+I+4*K
380   Z(I1,J)=B6(I,J)
      DO 385 K=0,A-2
      DO 385 I=1,4
      DO 385 J=1,6
      I1=6*A+4((B-A)+1)+I+4*K
      J1=J+6*(K+1)
385   Z(I1,J1)=B6(I,J)
      DO 390 I=1,6
      DO 390 J=1,4
      I1=6*(A-1)+I
      J1=6*A+J
390   Z(I1,J1)=A7(I,J)
      L1=B*(N-1)-A
      DO 395 K=0,L1
      DO 395 I=1,4
      DO 395 J=1,4
      I1=4*B+6*A+I+4*K
      J1=6*A+4*K+J
395   Z(I1;J1)=B2(I,J)
      L2=N*B-A
      DO 400 I=1,4
      DO 400 J=1,4
      DO 400 K=0,L2
      I1=6*A+4*K+I
      J1=6*A+4*K+J
400   Z(I1,J1)=B1(I,J)
      L3=N*B-A-1
      DO 415 K=0,L3
      DO 415 I=1,4
      DO 415 J=1,4
      I1=6*A+4*K+I
      J1=6*A+4*(K+1)+J
```

```
415   Z(I1,J1)=A4(I,J)
      DO 420 I=1,2
      DO 420 J=1,4
      I1=6*A+(N*B-A+1)*4+I
      J1=6*A+4*(((N-1)*B)-A+1)+J
420   Z(I1,J1)=A3(I,J)
      DO 130 I=1,4
      DO 130 J=1,2
      I1=6*A+(N*B-A)*4+I
      J1=6*A+4*((N*B)-A+1)+J
130   Z(I1,J1)=B4(I,J)
      DO 131 I=1,4
      DO 131 J=1,4
      I1=6*A+(N*B-A)*4+I
      J1=6*A+4*(((N-1)*B)-A+1)+J
131   Z(I1,J1)=B3(I,J)
      DO 430 I=1,2
      DO 430 J=1,2
      D(I,J)=0.0
430   D(I,I)=1
      DO 431 L=1,B-1
      CALL MATMUL(D,RN,R1,2,2,2)
      CALL MATMUL(D,A3,R2,2,2,4)
      CALL MATCHA(R1,D,2,2)
      DO 431 I=1,2
      DO 431 J=1,4
      I1=(N*B+1-A)*4+6*A+I
      J1=((N-1)*B+1-A)*4+6*A+J+4*L
      Z(I1,J1)=R2(I,J)
431   CONTINUE
      CALL MATMUL(D,RN,R3,2,2,2)
      CALL MATMUL(R3,A2,D2,2,2,2)
      CALL MATADD(A1,D2,D3,2)
      DO 432 I=1,2
      DO 432 J=1,2
      I1=(N*B+1-A)*4+6*A+I
      J1=(N*B+1-A)*4+6*A+J
432   Z(I1,J1)=D3(I,J)
      L4=(N*B+1-A)*4+6*A
      L5=L4+2
      DO 433 I=1,L4
433   Z(I,L5)=1
      DO 434 I=1,2
```

```
      DO 434 J=1,2
      E(I,J)=0.0
434   E(I,I)=1
      DO 435 I=1,2
      DO 435 J=1,2
435   F(I,J)=E(I,J)-RN(I,J)
      CALL MATINV(F,FI,2)
      DO 135 I=1,2
135   E1(I,1)=1
      CALL MATMUL(FI,E1,F2,2,2,1)
      DO 536 I=1,2
      I1=(N*B+1-A)*4+6*A+I
536   Z(I1,L5)=F2(I,1)
      CALL MATIN(Z,ZI,L5)
      M1=6*A
      DO 537 I=1,M1
537   XA(1,I)=ZI(L5,I)
      M2=(N*B+1-A)*4
      DO 538 I=1,M2
      I1=6*A+I
538   XB(1,I)=ZI(L5,I1)
      DO 539 I=1,2
      I1=6*A+4*(N*B+1-A)+I
539   XB1(1,I)=ZI(L5,I1)
      DO 540 I=1,M1
540   E3(I,1)=1
      DO 541 I=1,M2
541   E4(I,1)=1
      CALL MATMULT(XA,E3,E6,1,M1,1)
      CALL MATMULT(XB,E4,E7,1,M2,1)
      CALL MATMUL(XB1,FI,E9,1,2,2)
      CALL MATMUL(E9,E1,E10,1,2,1)
      CALL MATADD(E6,E7,E11,1)
      CALL MATADD(E11,E10,E12,1)
      WRITE(*,*) ((E12(I,J),J=1,1),I=1,1)
      Q1=A-1
      DO 542 O=0,Q1
      DO 542 J=1,6
      I1=(O)*6+J
      SUM=0
542   SUM=SUM+O*ZI(L5,I1)
      S1=SUM
      Q2=N*B
```

```
      DO 544 O=A,Q2
      DO 544 J=1,4
      I1=(O-A)*4+6*A+J
      SUM=0
544   SUM=SUM+O*ZI(L5,I1)
      S2=SUM
      DO 545 I=1,2
      DO 545 J=1,2
545   Q(I,J)=(N*B)*FI(I,J)
      CALL MATMUL(FI,FI,C13,2,2,2)
      DO 546 I=1,2
546   E14(I,1)=1
      CALL MATADD(Q,C13,C17,2)
      CALL MATMUL(XB1,C17,C18,1,2,2)
      CALL MATMUL(C18,E14,C19,1,2,1)
      DO 547 I=1,1
      DO 547 J=1,1
547   LQ(I,J)=S1+S2+C19(I,J)
      WRITE(*,*) ((LQ(I,J),J=1,1),I=1,1)
      END
```

Program for subroutines

```
      SUBROUTINE MATMUL(EE,H,U,N,L,M)
      REAL EE(20,20),H(20,20),U(20,20)
      INTEGER N,L,M
      DO 445 I=1,N
      DO 445 J=1,M
      U(I,J)=0.0
      DO 445 K=1,L
      U(I,J)=U(I,J)+EE(I,K)*H(K,J)
445   CONTINUE
      RETURN
      END
      SUBROUTINE MATMULT(EE,H,U,N,L,M)
      REAL EE(75,75),H(75,75),U(75,75)
      INTEGER N,L,M
      DO 445 I=1,N
      DO 445 J=1,M
      U(I,J)=0.0
      DO 445 K=1,L
      U(I,J)=U(I,J)+EE(I,K)*H(K,J)
445   CONTINUE
      RETURN
      END
      SUBROUTINE MATADD(T,G,W,N)
      REAL T(20,20),G(20,20),W(20,20)
      INTEGER N
      DO 777 I=1,N
      DO 777 J=1,N
      W(I,J)=T(I,J)+G(I,J)
777   CONTINUE
      RETURN
      END
      SUBROUTINE MATCHA(EB,E,M,N)
      REAL EB(20,20),E(20,20)
      INTEGER N,M
      DO 666 I=1,M
      DO 666 J=1,N
      E(I,J)=EB(I,J)
666   CONTINUE
      RETURN
      END
      SUBROUTINE MATIN(A,AI,N)
```

```
      REAL A(75,75),AI(75,75),AA(75,150)
      INTEGER N
      M=N+N
      M2=N+1
      DO 10 I=1,N
      DO 10 J=1,N
10    AA(I,J)=A(I,J)
      DO 20 I=1,N
      DO 20 J=M2,M
20    AA(I,J)=0.0
      DO 30 I=1,N
      J=I+N
30    AA(I,J)=1.0
      DO 280 LJ=1,N
      K=LJ
      IF (K.EQ.N) GO TO 100
      JJ=K
      BIG=ABS(AA(K,K))
      KP1=K+1
      DO 70 I=KP1,N
      AB=ABS(AA(I,K))
      IF (BIG-AB)60,70,70
60    BIG=AB
      JJ=I
70    CONTINUE
      IF (JJ-K)80,100,80
80    DO 90 J=K,M
      T=AA(JJ,J)
      AA(JJ,J)=AA(K,J)
      AA(K,J)=T
90    CONTINUE
100   P=AA(LJ,LJ)
      DO 250 I=LJ,M
250   AA(LJ,I)=AA(LJ,I)/P
      DO 270 LK=1,N
      T=AA(LK,LJ)
      DO 270 LI=LJ,M
      IF (LK-LJ)260,270,260
260   AA(LK,LI)=AA(LK,LI)-AA(LJ,LI)*T
270   CONTINUE
280   CONTINUE
      DO 290 I=1,N
      DO 290 J=M2,M
```

```
       L3=J-N
       AI(I,L3)=AA(I,J)
290    CONTINUE
       RETURN
       END
       SUBROUTINE MATINV(A,AI,N)
       REAL A(20,20),AI(20,20),AA(20,40)
       INTEGER N
       M=N+N
       M2=N+1
       DO 10 I=1,N
       DO 10 J=1,N
10     AA(I,J)=A(I,J)
       DO 20 I=1,N
       DO 20 J=M2,M
20     AA(I,J)=0.0
       DO 30 I=1,N
       J=I+N
30     AA(I,J)=1.0
       DO 280 LJ=1,N
       K=LJ
       IF (K.EQ.N) GO TO 100
       JJ=K
       BIG=ABS(AA(K,K))
       KP1=K+1
       DO 70 I=KP1,N
       AB=ABS(AA(I,K))
       IF (BIG-AB)60,70,70
60     BIG=AB
       JJ=I
70     CONTINUE
       IF (JJ-K)80,100,80
80     DO 90 J=K,M
       T=AA(JJ,J)
       AA(JJ,J)=AA(K,J)
       AA(K,J)=T
90     CONTINUE
100    P=AA(LJ,LJ)
       DO 250 I=LJ,M
250    AA(LJ,I)=AA(LJ,I)/P
       DO 270 LK=1,N
       T=AA(LK,LJ)
       DO 270 LI=LJ,M
```

```
      IF (LK-LJ)260,270,260
260   AA(LK,LI)=AA(LK,LI)-AA(LJ,LI)*T
270   CONTINUE
280   CONTINUE
      DO 290 I=1,N
      DO 290 J=M2,M
      L3=J-N
      AI(I,L3)=AA(I,J)
290   CONTINUE
      RETURN
      END
```