52820

0-315-04926-X

National Library
of CanadaBibliothèque nationale
du CanadaCANADIAN THESES
ON MICROFICHETHÈSES CANADIENNES
SUR MICROFICHE

NAME OF AUTHOR/NOM DE L'AUTEUR

David Andrew Stanford

TITLE OF THESIS/TITRE DE LA THÈSE

Prediction of Times and
Queue Lengths in Queueing Systems

UNIVERSITY/UNIVERSITÉ

Carleton University

DEGREE FOR WHICH THESIS WAS PRESENTED/

GRADE POUR LEQUEL CETTE THÈSE FUT PRÉSENTÉE

Ph.D.

YEAR THIS DEGREE CONFERRED/ANNÉE D'OBTENTION DE CE DEGRÉ

1981

NAME OF SUPERVISOR/NOM DU DIRECTEUR DE THÈSE

Dr. B. Pagnurek, Dr. C. M. Woodside

Permission is hereby granted to the NATIONAL LIBRARY OF
CANADA to microfilm this thesis and to lend or sell copies
of the film.

L'autorisation est, par la présente, accordée à la BIBLIOTHÈ-
QUE NATIONALE DU CANADA de microfilmer cette thèse et
de prêter ou de vendre des exemplaires du film.

The author reserves other publication rights, and neither the
thesis nor extensive extracts from it may be printed or other-
wise reproduced without the author's written permission.

L'auteur se réserve les autres droits de publication; ni la
thèse ni de longs extraits de celle-ci ne doivent être imprimés
ou autrement reproduits sans l'autorisation écrite de l'auteur.

DATED/DATE

August 31st 1981

SIGNED/SIGNÉ

David A. Stanford

PERMANENT ADDRESS/RÉSIDENCE FIXE

2115-195 Clearview Avenue
Ottawa K1Z 6S1

NOTICE

The quality of this microfiche is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us a poor photocopy.

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30. Please read the authorization forms which accompany this thesis.

**THIS DISSERTATION
HAS BEEN MICROFILMED
EXACTLY AS RECEIVED**

AVIS

La qualité de cette microfiche dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de mauvaise qualité.

Les documents qui font déjà l'objet d'un droit d'auteur (articles de revue, examens publiés, etc.) ne sont pas microfilmés.

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30. Veuillez prendre connaissance des formules d'autorisation qui accompagnent cette thèse.

**LA THÈSE A ÉTÉ
MICROFILMÉE TELLE QUE
NOUS L'AVONS REÇUE**

PREDICTION OF TIMES AND QUEUE LENGTHS

IN QUEUEING SYSTEMS

by

David Andrew Stanford, B.Sc., M.Eng.

A thesis submitted to

the Faculty of Graduate Studies and Research

in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Department of Systems and Computer Engineering

Faculty of Engineering

Carleton University

Ottawa, Ontario

July 1981

David Andrew Stanford, 1981

The undersigned recommend to the
Faculty of Graduate Studies and Research
acceptance of the thesis

"PREDICTION OF TIMES AND QUEUE LENGTHS
IN QUEUEING SYSTEMS"

submitted by David Andrew Stanford, B.Sc., M.Eng.
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

Thesis Supervisor

Thesis Supervisor,
Chairman,
Department of Systems
and Computer Engineering

Carleton University

August 1981

Dedication

This thesis is dedicated to my parents, Ronald and Veronica, for the love they have always shown for their children; to my uncle, Fr. Lionel Stanford, S.J., for the profound impact he had on my life, and to Terry Fox, for the inspiration he has given to all Canadians.

Abstract

This research concerns the optimal mean square (OMS) prediction of the queue length, waiting time and system (waiting plus service) time processes in a variety of stationary queueing systems, and specifically addresses the question of using the queue length to provide an indication of current and future delays. Values of the mean squared error (MSE) are computed as a performance measure. In each of the queueing systems considered, the delay predictions and their associated MSE's are found to be closely related to the underlying predictor for the imbedded queue length process.

In the case of GI/M/1 queues, both OMS and optimal linear mean square (OLMS) prediction are considered. The OLMS predictors are shown to depend on more than the most recent available queue length, however extra earlier data produces only a small improvement (less than 1%) in the MSE. At the same time, the error associated with the OMS predictor is typically only 2-5% smaller than the one-observation linear predictor. For both the OMS and OLMS cases, the system and waiting time predictors use the queue length predictors as an intermediate result.

A study is carried out to assess the relative merits of using previous queue lengths or system times for system time prediction. While the predictor based on delays is noticeably better, the delay

data itself is available so much later than the corresponding queue length as to be irrelevant for prediction purposes.

The OMS predictors for delay in GI/M/M queues make use of a restricted summation of terms of the queue length predictor. As a result, the delay predictions and the associated MSE's do not use the queue length predictor as an intermediate result. While the normalized MSE for waiting times is a monotonically decreasing function of the number of servers, the queue length error behaves in a more complex manner.

Whereas the imbedded queue length is observed at arrival instants in GI/M/1 and GI/M/M queues, it is observed at departure instants in M/G/1 queues. As a result, expressions for the OMS predictors for delay in the M/G/1 queue are shown to depend upon the relative values of the prediction horizon n and initial length i . For $n > i$, it is a scalar multiple of the queue length trajectory.

The prediction trajectories of all 3 queues display a short term linear decay for a sufficiently large initial queue length. The duration of this behaviour is dependent upon the probability of the queue emptying.

Expressions for the expected delay in a series of tandem M/M/1 queues conditioned upon queue length data are presented. For two queues in tandem, an approach similar to that used for GI/M/1 queues applies. For 3 or more queues, the expected delay is given as the

time to absorption of a Markov chain with 1 absorbing state modelled from the queue length data.

The thesis concludes with a discussion of the applicability of predictors to adaptive routing schemes in data networks.

ACKNOWLEDGEMENTS

I would like to express my deepest thanks to my supervisors, Dr. Bernard Pagurek and Dr. C. Murray Woodside, for their constant help and guidance in the direction of this thesis. Their frequent comments turned many a stumbling block into fruitful research, and gave direction to the writing of the thesis.

I would also like to thank Dr. Raj Pandya for his comments on potential applications. I am indebted to Diane Dodds for her frequent help in cutting through departmental red tape and meeting a variety of deadlines.

I would also like to thank the following students for the frequent theological and political discussions which are System Engineering's trademark: Claude Laferriere, Sherin Kamal, Arthur Guibord, Mike Spyrtos and Mohsen Rashwan. Extreme thanks are also due to my good friends Morteza Niktash (for his "technical matters") and Mustafa Mehmet Ali (for a daily rendition of the theme to "The Godfather").

I am deeply indebted to Dr. Carl M. Harris for his helpful comments on the dissertation, and for agreeing to act as external examiner.

Finally, I would like to extend my sincere gratitude to Mrs. Elena Keen for typing this thesis so well and so quickly.

TABLE OF CONTENTS

Dedication	iii
Abstract	iv
Acknowledgements	vii
Table of Contents	
List of Tables	
List of Figures	
CHAPTER 1 INTRODUCTION	1.1
CHAPTER 2 PREVIOUS RESEARCH AND MATHEMATICAL PRELIMINARIES	2.1
2.1 Mathematical Preliminaries	2.1
2.1.1 GI/M/1 Queues	2.2
2.1.2 GI/M/m Queues	2.4
2.1.3 M/G/1 Queues	2.5
2.2 Related Research	2.7
CHAPTER 3 OPTIMAL LINEAR AND NONLINEAR PREDICTORS FOR N, W, AND S IN STATIONARY GI/M/1 QUEUES	3.1
3.1 Introduction	3.1
3.2 A Study of OMS Prediction	3.2
3.2.1 Queue Length Prediction	3.3
3.2.2 System Time Prediction	3.6
3.2.3 Waiting Time Prediction	3.7
3.2.4 System and Waiting Time Results	3.8
3.2.5 Algorithm for the OMS Predictors	3.8
3.3 A Study of OLMS Prediction	3.9
3.3.1 Queue Length Linear Prediction	3.10
3.3.2 System Time Linear Prediction	3.12

3.3.3	Waiting Time Linear Prediction	3.14
3.3.4	Algorithm for the OLMS Predictors	3.14
3.4	A Comparison of OLMS and OLMS Queue Length Predictors	3.15
3.5	A Comparison of $\{S_n/N_k\}$ and $\{S_n/S_k\}$	
3.6	Predictions Based on Current Queue Length	3.19
3.7	Conclusions	3.20
CHAPTER 4	NONLINEAR PREDICTION IN GI/M/m QUEUES	4.1
4.1	Introduction	4.1
4.2	Queue Length Prediction	4.2
4.3	Waiting Time Prediction	4.4
4.4	System Time Prediction	4.6
4.5	Algorithm for the OM $\hat{S}$ Predictors	4.7
4.6	Numerical Examples	4.8
4.7	Conclusions	4.11
CHAPTER 5	M/G/1 PREDICTION	5.1
5.1	Introduction	5.1
5.2	Queue Length Prediction	5.2
5.3	System Time Prediction	5.3
5.4	Waiting Time Prediction	5.7
5.5	Efficient Computation of the Queue Length Predictions	5.8
5.6	Numerical Examples	5.12
5.7	Conclusions	5.15
CHAPTER 6	SYSTEM TIMES IN TANDEM M/M/1 QUEUES	6.1
6.1	Reich's Analysis of Tandem M/M/1 Queues	6.2
6.2	System Times in Two Tandem M/M/1 Queues	6.4
6.3	The Extension to Three or More Tandem Queues	6.6
6.4	Finite Markov Chain Approaches for $\{S_n/N_{1,n}, \dots, N_{k,n}\}$	6.9
6.4.1	Imbedded Markov Chain Model	6.9
6.4.2	Continuous Parameter Markov Chain Model	6.14

6.5	A Comparison of the Two Markov Chains	6.16
6.6	Conclusions	6.17
CHAPTER 7	THE APPLICABILITY OF PREDICTORS TO ADAPTIVE ROUTING SCHEMES	7.1
7.1	Introduction	7.1
7.2	Adaptive Routing Model	7.1
7.3	Heuristic Predictors	7.5
7.4	ARPANET Routing Algorithms	7.6
7.5	Implementation of the Predictors	7.9
7.6	Conclusions	7.10
CHAPTER 8	CONCLUSIONS AND FURTHER WORK	8.1
8.1	Conclusions	8.1
8.2	Recommendations for Further Work	8.3
APPENDIX 1	CORRELATION COEFFICIENTS FOR QUEUE LENGTH, SYSTEM TIME, AND WAITING TIME IN GI/M/1 QUEUES.	A1.1
A1.2	Algorithms for $r_W(n)$ and $r_S(n)$	A1.7
A1.3	Results for $r_{NW}(n)$ and $r_{NS}(n)$	A1.9
A1.4	Conclusions	A1.10
APPENDIX 2	THE SERIAL CORRELATION COEFFICIENTS FOR WAITING TIMES IN THE GI/M/m QUEUE	A2.1
A2.1	Introduction	A2.1
A2.2	An Expression for $E\{e^{-\theta W_0}; W_1 < x\}$	A2.2
A2.3	An Expression for $E\{W_0; W_1 < x\}$	A2.3
A2.4	An Expression for $E\{W_0 W_n\}$	A2.5
A2.5	An Expression for $r_W(n)$	A2.6
APPENDIX 3	PROGRAM LISTINGS	A3.1

Bibliography

LIST OF TABLES

	Page
Table 3.1 Prediction of N_n and S_n	3.21
Table 3.2 Normalized Errors for Queue Length Predictions in M/M/1, ρ Varying	3.22
Table 3.3 Normalized Errors for System Time Predictions in M/M/1, ρ Varying	3.22
Table 3.4 Normalized Errors for Queue Length Predictions in Erlang Queues $E_R/M/1$, $\rho = 0.7$	3.23
Table 3.5 Normalized Errors for Queue Length Predictions with a Varying Number of Observations	3.23
Table 3.6 Predictor Coefficients and Errors for the Linear System Time Predictor	3.24
Table 4.1 Values of $E\{N_n/N_0=i\}$ in M/M/m, $\rho = 0.7$	4.13
Table 4.2 Values of $E\{W_n/N_0=i\}$ in M/M/m, $\rho = 0.7$	4.14
Table 4.3 Values of $E\{N_n/N_0=i\}$ in $E_R/M/2$, $\rho = 0.7$	4.15
Table 4.4 Values of $E\{W_n/N_0=i\}$ in $E_R/M/2$, $\rho = 0.7$	4.16
Table 4.5 Normalized Errors for Queue Length and Waiting Time Predictions in M/M/1, ρ Varying	4.17
Table 4.6 Normalized Errors for Queue Length and Waiting Time Predictions in M/M/2, ρ Varying	4.17
Table 4.7 Normalized Errors for Queue Length and Waiting Time Predictions in M/M/5, ρ Varying	4.18
Table 4.8 Normalized Errors for Queue Length and Waiting Time Predictions in $E_R/M/2$, R Varying, $\rho = 0.7$	4.18

Table 5.1	Queue Length Predictions	5.16
Table 5.2	System Time Predictions	5.17
Table 5.3	Normalized Errors for Queue Length Predictions ρ Varying	5.18
Table 5.4	Normalized Errors for System Time Predictions M/M/1 ρ Varying	5.18

LIST OF FIGURES

	Page
Figure 3.1 Linear and Nonlinear Predictions for the M/M/1 Queue with $\rho = 0.9$	3.25
Figure 3.2 Normalized Prediction Error for System Times in an M/M/1 Queue	3.26
Figure 4.1 Queue Length Predictions Trajectories M/M/m, $\rho = 0.7$	4.19
Figure 4.2 Queue Length Prediction Trajectories E _R /M/1, $\rho = 0.7$	4.20
Figure 5.1 The Relationship Between D_0 and S_n	5.19
Figure 5.2 M/G/1 Queue Length Prediction Trajectories $\rho = 0.7$	5.20
Figure 5.3 M/G/1 System Time Prediction Trajectories	5.21
Figure 6.1 A Series of Queues in Tandem	6.18
Figure 7.1 A Data Network	7.11

CHAPTER 1

INTRODUCTION

One of the most fundamental results of the theory of stationary queueing systems is Little's result [8], which relates the average queue length (i.e. number in system) to the average time a customer spends in the system. Furthermore, this result applies to queues having even the most general arrival and service distributions.

This fact reflects an intrinsic relationship between the queue length and delay processes in queueing systems. The current work addresses itself to a study of this relationship, and specifically to the question of using the queue length history to provide an indication of current and future delays customers will experience. The delays to be considered are the waiting time and the system (waiting plus service) time.

A potential application for this study is in the adaptive routing of messages through a computer communications network. The queue length at a given node can be used as a measure of future delay which messages would undergo on a link. This matter is discussed at greater length in chapter 7.

This work brings together two important areas of stochastic processes which have not, until recently, been explored in common. Prediction and estimation have been powerful, practical tools in many areas of stochastic processes, such as signal processing. The Wiener-Kalman filter is but one such tool with applications in the area of adaptive control of stochastic systems. At the same time, there is a great wealth of knowledge available regarding single and multiple server queueing systems, yet little attention has been paid to prediction of queueing processes. Hence it is a logical step to apply these tools which have proven so useful elsewhere in stochastic processes to the unexplored area of queueing systems.

When one considers predictors, the usual starting point is the conceptually simpler case of unbiased linear predictors. That is, the prediction $\hat{X}$ is based on a linear combination of the observations Y_i , $i=1, \dots, N$ in the following way:

$$\hat{X} = \mu_X + \sum_{i=1}^N \alpha_i (Y_i - \mu_Y)$$

Clearly, some criterion must be used to select the "best" coefficients α_i . In the case of optimal mean square predictors, the criterion is the minimization of the mean squared error Q given by

$$Q = E\{(\hat{X} - X)^2\}$$

As will be shown, this criterion leads to a set of linear equations

for the α_i 's which yield a unique solution. The coefficients of the equations and the right hand side are the various covariances of the queue length, system time, and waiting time processes under study. The resulting form is called the optimal linear mean square (OLMS) predictor.

Optimal (nonlinear) mean square (OMS) prediction uses the same optimality criterion as the linear predictor; however, the prediction is not restricted to being a linear function of the data. As a result, the mean squared error for OMS predictors is generally smaller than the OLMS mean squared error. The bulk of the thesis deals with OMS prediction; OLMS predictors are considered in the case of GI/M/1 queues so as to provide a comparison of the two methods, and can be easily extended to many of the other queueing systems to be considered.

Markov processes such as the queue length are defined by their dependence upon the most recent state. This is frequently misinterpreted as implying that a linear predictor based on data from such a process will necessarily depend only upon the most recent observation available. However, only "wide-sense" Markov processes, whose correlation coefficients are given by

$$R(n) = r^n$$

for some $0 < r < 1$ satisfy this property. Since none of the processes to be studied in the queues to be considered have such correlation coefficients, a linear predictor based on any of them would not depend solely upon the latest available data, except in the trivial cases of predicting a customer's waiting or system time from the queue length he sees upon arrival in

GI/M/1 and GI/M/m queues.

The optimal nonlinear predictor for X based on $Y_1, \dots, Y_n$ selects as its estimate the conditional mean

$$\hat{X} = E\{X/Y_1, \dots, Y_n\} \quad (1.2)$$

For all OMS predictors to be presented, the data will be drawn from the imbedded queue length process of the queue under study. As a result, the conditional expectations of the processes under study are found to be functions of the latest observation Y_n only.

As a starting point, Chapter 3 deals with prediction in stationary GI/M/1 queues. That is to say, prediction in a single server queue where service times are distributed exponentially and the interarrival times are independently drawn from a common arbitrary distribution. In this queue, the processes $\{N_n\}$ of queue lengths at arrival epochs, $\{S_n\}$ of system times, and $\{W_n\}$ of waiting times are very closely related.

This close relationship is due in great measure to the fact that residual service times are exponentially distributed with the same mean as total service times in the GI/M/1 queue. Hence, by knowing the queue length upon arrival, one immediately knows how many independent, identically distributed exponential service times are comprised in the arriving customer's wait and system time. Furthermore, the queue length at arrival instants is a natural, meaningful, easily observed process, which lends itself easily to being used as data in a predictor.

Both OLMS and OMS predictors for N_n , S_n , and W_n are considered in Chapter 3. As mentioned earlier, the computation of the OLMS predictor requires several covariances to be determined. The pertinent algorithms by Pakes [14], [15] in the case of serial auto-correlations, and new results in the case of cross-correlations, appear in Appendix I. In both the OLMS and OMS cases, the system and waiting time predictors use the queue length predictor as an intermediate result. Typical prediction trajectories for OLMS and OMS predictors are compared. Studies are undertaken which use the mean squared error as a performance measure to investigate the usefulness of additional earlier data in the linear predictor and to observe how close the OLMS predictor comes to the optimal OMS predictor.

Finally, the OLMS predictors for system times based on earlier system times and on queue lengths are compared in order to evaluate the utility of using previous delays for predicting future delays.

In Chapter 4, OMS prediction in GI/M/m (i.e. multiple server, exponential service) queues is considered. The usefulness of multiple server prediction results stems from the fact that single server models frequently prove to be poor approximations for queueing applications (in telephony, for example). These results are of further interest because they provide a contrast to the single server predictions.

One difficulty in dealing with multi-server queues is that customers do not necessarily leave in the order that they arrive. However, they enter service in order. This fact facilitates calculation of the OMS predictor for waiting times based on queue lengths.

Through a series of numerical examples, the effect of the number of servers on the predictions and mean squared errors of both the queue length and waiting time processes is considered. Contrary to the single server case, the waiting time predictor does not use the queue length predictor as an intermediate result; the examples presented serve to illustrate this point.

Chapter 5 deals with OMS prediction in $M/G/1$ queues; namely, queues having Poisson arrivals and general i.i.d. service times. Since Poisson arrivals are a more common phenomenon than exponential service times, $M/G/1$ prediction results have a greater potential applicability and provide an interesting dual to the $GI/M/1$ results.

In $M/G/1$ queues, the queue length process is observed at departure epochs. Furthermore, the mean residual service time is generally not equal to the average service time. These facts render an arrival instant approach to $M/G/1$ prediction impractical if not intractable.

Prediction results equally powerful to those for $GI/M/1$ queues exist for predictors based on queue lengths observed at departure instants. These predictors are presented in Chapter 5 along with a series of numerical examples. Depending upon the relative values of the prediction horizon and the observed queue length, one of two separate sets of results applies. The numerical examples presented demonstrate the effects of this dichotomy on the prediction trajectories.

For sufficiently large initial queue length conditions, the queue

length prediction trajectories of chapters 3, 4, and 5 all display a short-term linear decay. This empirical behaviour can be explained by the following heuristic: For each customer to arrive to the system, we expect that $(1/\rho)$ customers will finish service so long as the queue does not empty. In several cases prediction trajectories overshoot the mean queue length and then approach it from below as the prediction horizon approaches infinity. This behaviour closely parallels that observed by Roth [21] in the study of transient behaviour of the expected queue length.

Chapter 6 is devoted to the prediction of the time spent by a customer in a series of tandem M/M/1 queues. In the case of two queues, an extension of the methods of Chapter 3 is described. The reason why the method fails to extend to 3 or more queues is explained and a solution method employing the absorption time analysis of discrete parameter Markov chains is given.

In Chapter 7, the application of prediction techniques to adaptive routing schemes in data networks is considered. The adaptive routing model is described; and the problems in obtaining analytical results are discussed. A description of the current and former ARPANET routing algorithms is then given as a specific example, and several advantages and disadvantages in implementing the heuristic predictors are considered.

The thesis concludes with a summary of the results obtained and suggestions for further work.

CHAPTER 2

PREVIOUS RESEARCH AND MATHEMATICAL PRELIMINARIES

2.1 Mathematical Preliminaries

In this section, the properties of the queue length, system time and waiting time will be investigated in each of the GI/M/1, GI/M/m and M/G/1 queues at stationarity. Letting C_n denote "the n th customer", we define

N_n - queue length seen by C_n upon arrival

D_n - queue length left behind by C_n

W_n - C_n 's waiting time

S_n - C_n 's system time

X_n - C_n 's service time

T_n - time between arrival of C_{n-1} and C_n .

Since we are considering stationary queues, we define for each Y_n above.

$$E\{Y_n\} = \mu_Y$$

$$\text{Var}\{Y_n\} = \sigma_Y^2$$

We also define for $Y=W, S, X$ and T

$$F_Y(t) = \Pr\{Y_n \leq t\}$$

$$F_Y^{(n)}(t) = \text{nth convolution of } F_Y(t)$$

$$\phi_Y(s) = \int_0^{\infty} e^{-st} dF_Y(t)$$

In both GI/M/1 and GI/M/m queues, the ~~imbedded~~ Markov chain is the sequence of queue lengths observed by arrivals. Hence we define

$$\pi_i = \Pr\{N_n = i\}$$

$$p_{ij}^k = \Pr\{N_{n+k} = j / N_n = i\}$$

while in M/G/1 queues these probabilities shall be understood to refer to the D_n^c process. We first consider GI/M/1 queues.

2.1.1 GI/M/1 Queues

Both Gross and Harris [5] and Kleinrock [6] provide good descriptions of the standard analysis of GI/M/1 queues. Several pertinent results are stated here.

The stationary queue length as seen by an arrival is geometrically distributed, that is

$$\pi_i = (1-d)d^i; \quad i \geq 0 \quad (2.1)$$

where $x=d$ is the unique solution in $(0,1)$ of

$$x = \phi_T(\mu(1-x)) \quad (2.2)$$

Hence

$$\mu_N = d/(1-d); \quad \sigma_N^2 = d/(1-d)^2 \quad (2.3)$$

The one-step transition probabilities are

$$p_{ij} = \begin{cases} 0 & j > i+1 \\ k_{i+1-j} & 1 \leq j \leq i+1 \\ 1 - \sum_{\ell=1}^{i+1} p_{i\ell} & j=0 \end{cases} \quad (2.4)$$

where

$$k_m = \int_0^{\infty} e^{-\mu t} \frac{(\mu t)^m}{m!} dF_T(t) \quad (2.5)$$

The stationary system time distribution is exponential with mean $\frac{1}{\mu(1-d)}$ where μ is the rate of the exponential service mechanism. Hence

$$F_S(t) = 1 - e^{-\mu(1-d)t}; \quad t \geq 0 \quad (2.6)$$

$$\mu_S = \frac{1}{\mu(1-d)}; \quad \sigma_S^2 = \frac{1}{\mu^2(1-d)^2} \quad (2.7)$$

The waiting time distribution is a mixed distribution satisfying

$$F_W(t) = 1 - d e^{-\mu(1-d)t}; \quad t \geq 0 \quad (2.8)$$

Therefore

$$\mu_W = \frac{d}{\mu(1-d)}; \quad \sigma_W^2 = \frac{2d-d^2}{\mu^2(1-d)^2} \quad (2.9)$$

The preceding results are well-known. We now derive the first original result. We seek the conditional expectation $E\{S_n/N_k=i\}$ which will be needed for several prediction and correlation results. We have

$$E\{S_n/N_k=i\} = \int_0^{\infty} t d \Pr\{S_n \leq t/N_k=i\} \quad (2.10)$$

Conditioning upon N_n one obtains

$$E\{S_n/N_k=i\} = \sum_{j=0}^{\infty} p_{ij}^{n-k} \int_0^{\infty} t d\Pr\{S_n \leq t/N_k=i; N_n=j\} \quad (2.11)$$

If $N_n=j$, then S_n consists of $(j+1)$ i.i.d. exponential random variables, each with mean $(\frac{1}{\mu})$; thus regardless of the value of N_k we have

$$\begin{aligned} E\{S_n/N_k=i\} &= \\ &= \sum_{j=0}^{\infty} p_{ij}^{n-k} (j+1) \left(\frac{1}{\mu}\right) = \left(\frac{1}{\mu}\right) [E\{N_n/N_k=i\} + 1] \end{aligned} \quad (2.12)$$

N.B. A similar approach shows that additional knowledge of earlier queue lengths $N_0, N_1, \dots, N_{k-1}$ has no effect upon the expectation. Furthermore, we can show in like fashion that

$$E\{W_n/N_k=i\} = \left(\frac{1}{\mu}\right) E\{N_n/N_k=i\} \quad (2.13)$$

2.1.2 GI/M/m Queues

Let $c_m \triangleq \Pr\{N_n \geq m\}$. Takacs [24] has shown that

$$F_W(t) = 1 - c_m e^{-m\mu(1-d_m)t}; \quad t \geq 0 \quad (2.14)$$

where $x=d_m$ is the unique solution in $(0,1)$ of

$$x = \phi_T(m\mu(1-x)) \quad (2.15)$$

Therefore

$$\mu_W = \frac{c_m}{m\mu(1-d_m)}; \quad \sigma_W^2 = \frac{2c_m - c_m^2}{(m\mu(1-d_m))^2} \quad (2.16)$$

The stationary queue length is geometrically distributed for $i \geq (m-1)$, with parameter d_m as defined in (2.15). That is:

$$\pi_i = c_m (1-d_m) d_m^{i-m} \quad i \geq m-1 \quad (2.17)$$

The remaining probabilities are found using

$$\pi_i = \sum_{r=1}^{m-1} (-1)^{r-i} \binom{r}{i} U_r \quad i=0, \dots, m-2 \quad (2.18)$$

where

$$U_r = A C_r \sum_{j=r+1}^m \binom{m}{j} \frac{(1-\phi_j)^{-j}}{(1-d_m)^{-j}} \frac{1}{C_j (1-\phi_j)}$$

$$A = \left[\frac{1}{1-d_m} + \sum_{j=1}^m \binom{m}{j} \frac{1}{C_j (1-\phi_j)} \frac{(1-\phi_j)^{-j}}{(1-d_m)^{-j}} \right]^{-1}$$

$$C_j = \prod_{v=1}^j \left(\frac{\phi_v}{1-\phi_v} \right)$$

for $\phi_v \triangleq \phi(v\mu)$. The value of c_m is given by

$$c_m = A / (1-d_m) \quad (2.19)$$

2.1.3 M/G/1 Queues

Since it is the arrivals which are Markovian in M/G/1, the imbedded Markov chain is the sequence of queue lengths immediately following departures, D_n . The stationary distribution for D_n has been shown to be equal to the stationary queue length observed randomly. The stationary distributions of the waiting and system times cannot be specified without knowing the particular service distribution; nor, in fact can the stationary queue length be given. Nonetheless, several relationships involving

the transforms of these processes have been obtained; for example, see Gross and Harris [5]. Define

$$K_i \triangleq \int_0^{\infty} e^{-\lambda t} \frac{(\lambda t)^i}{i!} dF_X(t) \quad i=0,1,\dots$$

$$G_Y(z) \triangleq \sum_{i=0}^{\infty} \gamma_i z^i$$

Then

$$G_{\pi}(z) = (1-\rho) \frac{(1-z)G_K(z)}{G_K(z)-z} \quad (2.20)$$

$$G_{\pi}(z) = \phi_S(\lambda(1-z)) \quad (2.21)$$

$$G_K(z) = \phi_X(\lambda(1-z)) \quad (2.22)$$

and

$$\phi_S(s) = \frac{(1-\rho)s\phi_X(s)}{s-\lambda(1-\phi_X(s))} \quad (2.23)$$

where $\rho = \lambda \bar{X}$. Furthermore,

$$\mu_D = \rho + \frac{\rho^2 + \lambda^2 \sigma_X^2}{2(1-\rho)} \quad (2.24)$$

2.2 Related Research

While little work has been done in the area of prediction and time series analysis of queueing processes, the importance of the topic was recognized in 1968 by Daley [3]. However, a great deal of work has been done in the area of serial autocorrelation of the queue length, waiting time, and system time processes in both the GI/M/1 and M/G/1 queues. These correlation coefficients are needed to calculate the predictor coefficients of the optimal linear mean square predictors to be considered in Chapter 3.

The previous work done in the area of correlation in GI/M/1 queues is due, in great measure, to Pakes [14], [15], and Daley [3]. In [15], Pakes derives an algorithm for the computation of serial correlation coefficients for waiting times $r_W(n)$. He then relates the system time coefficients $r_S(n)$ to the work in [15] on the queue length coefficients $r_N(n)$.

That work started with an approach by Daley [3] for the calculation of $\sum_{n=0}^{\infty} r_N(n)$, which had been based upon the z-transform of the correlation coefficients. Pakes inverted the z-transform, and showed how this inversion involved the distribution of number served in a busy period. Alternatively, the first $(n-1)$ derivatives of an expression involving the n -fold convolution of the interarrival distribution are needed to find $r_N(n)$.

Stanford in [6] obtained an expression for $\sum_{n=0}^{\infty} r_N(n)$ in a GI/M/1 queue where arrivals occurred in bulks of fixed size m . For bulks of size 1, the results reduce to those of Daley [3].

Related work in the M/G/1 queue is due primarily to Daley [2], Blomqvist [1]

Pakes [4], and Pagurek and Woodside [11]. The closed form expressions for the correlations in [14] and [4] suffer from the same requirements to differentiate complex expressions as those by Pakes [14] do in the case of the GI/M/1 queue.

Reynolds [20] provides an excellent survey of the correlation work in M/G/1 queues, considering both continuous time correlation functions and serial correlation coefficients for several processes in these queues.

Previous work in the area of prediction itself is limited to Stanford [23] where linear predictors for the waiting time process $\{W_n\}$ of the GI/M/1 queue are considered.

A related study has been recently performed by Roth [21], where the transient behaviour of the expected queue length of several stationary queueing systems based on initial queue length conditions is considered. She demonstrates empirically that the transients decay in an approximately exponential manner after a certain point. For appropriate combinations of traffic intensity and initial queue length the trajectories display an initial linear behaviour similar to those which will be presented in the present work.

The pertinent research in the area of system times in networks of queues consists of papers by Reich and Melamed. Reich shows in [19] that the system times at successive tandem M/M/1 queues are independent and exponentially distributed. Melamed [9] proves that Reich's analysis

extends to networks having no overtake free paths. Essentially, this requirement prohibits a customer's past from influencing his future. When overtaking is permitted, the analysis becomes complex and the solution is frequently unattainable.

CHAPTER 3

OPTIMAL LINEAR AND NONLINEAR PREDICTORS FOR N, W, and S IN STATIONARY GI/M/1 QUEUES

3.1 Introduction

How long will a customer spend in system knowing that the previous customer to arrive found 5 people ahead of him? How long does he expect to wait if a particular earlier customer spent 7 minutes in the queue? What do we expect the queue length to be when the next customer arrives given that the most recent customer found 10 in system?

In this chapter, questions of this nature will be considered as we study prediction and estimation in GI/M/1 queues. The answers to the preceding questions clearly depend upon such parameters as mean service time, utilization, and the interarrival density. However, even with this information, two questions remain to be resolved. According to what criterion is our answer "best"? Are there any restrictions as to how to make use of the data available?

In answer to the first question, we seek the estimate $\hat{Y}$ of Y which minimizes the mean squared error $Q(\hat{Y})$ given by

$$Q = E\{(\hat{Y} - Y)^2\}$$

The first predictors we shall consider will have no other restrictions as to how the available data is used. The resulting forms will be called

the optimal nonlinear mean square (OMS) predictors. These will be considered in the next section. In the following section only predictions which are linear functions of the data will be considered; these are denoted the optimal linear mean square (OLMS) predictors. Clearly, the latter group forms a subset of the former, hence the (OLMS) results are generally suboptimal to the OMS ones. In these sections, the only data to be considered will be the imbedded queue lengths. In a later section, linear predictors for system times based on earlier system times will be considered.

3.2 A Study of OMS Prediction

The following development is well-known; for example, see Papoulis [16], pp. 216-218.

We seek a suitable function $g(X)$ to estimate Y in terms of X . The criterion that we use is the minimization of the mean squared error

$$E\{(Y-g(X))^2\} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (y-g(x))^2 f(x,y) dy dx$$

$$= \int_{-\infty}^{\infty} f(x) \left[\int_{-\infty}^{\infty} (y-g(x))^2 f(y/x) dy \right] dx$$

Since the inner integrand is non-negative, it suffices to minimize it for each x in order to minimize the error. But

$$\int_{-\infty}^{\infty} (y-g(x))^2 f(y/x) dy = g(x)^2 - 2g(x)E\{Y/x\} + E\{Y^2/x\}$$

Differentiation of the above w.r.t. $g(x)$ produces the minimum

$$g(x) = E\{Y/x\}$$

and so we find

$$g(X) = E\{Y/X\} \quad (3.1)$$

Therefore, the optimal mean square predictor for Y in terms of the data X is the conditional expectation $E\{Y/X\}$.

We now consider the OMS predictors for N_n , W_n , and S_n in terms of a sequence of earlier queue lengths $N_K = \{N_0, N_1, \dots, N_K\}$.

3.2.1 Queue Length Prediction

First consider the prediction of the queue length N_n . Since we are dealing exclusively with the imbedded Markov chain, the optimal nonlinear estimate will depend solely upon the most recent available queue length N_K , so we seek $E\{N_n/N_K\}$. We are also interested in the conditional variance and the mean squared error. These latter quantities are given by

$$\text{Var}\{N_n/N_K=i\} = E\{N_n^2/N_K=i\} - E\{N_n/N_K=i\}^2 \quad (3.2)$$

$$\text{MSE}(N_n) = \sum_{i=0}^{\infty} \pi_i \text{Var}\{N_n/N_K=i\} \quad (3.3)$$

The conditional moments are given by

$$E\{N_n/N_K=i\} = \sum_{j=0}^{\infty} j p_{ij}^{n-k} \quad (3.4)$$

$$E\{N_n^2/N_K=i\} = \sum_{j=0}^{\infty} j^2 p_{ij}^{n-k} \quad (3.5)$$

Efficient recursive computation of the desired quantities can be done

as follows. The Kolmogorov-Chapman backward equation for the transition probabilities is

$$p_{ij}^{n-k} = \sum_{\ell=0}^{i+1} p_{i\ell} p_{\ell j}^{n-k-1} \quad (3.6)$$

since $p_{i\ell}=0$ for $\ell>i+1$ in this process. Substitution of (3.6) into (3.4) and (3.5) yields

$$E\{N_n/N_k=i\} = \sum_{\ell=0}^{i+1} p_{i\ell} E\{N_{n-1}/N_k=\ell\} \quad (3.7)$$

$$E\{N_n^2/N_k=i\} = \sum_{\ell=0}^{i+1} p_{i\ell} E\{N_{n-1}^2/N_k=\ell\} \quad (3.8)$$

Starting with $E\{N_k/N_k=\ell\}=\ell$, $E\{N_k^2/N_k=\ell\}=\ell^2$, (3.7) and (3.8) give the moments recursively without the need to calculate the multistep transition probability p_{ij}^{n-k} .

Now define

$$MSE_{L(I)}(N_n) = \sum_{i=0}^I \pi_i \text{Var}\{N_n/N_k=i\} \quad (3.9)$$

$$MSE_{U(I)}(N_n) = E\{N_n^2\} - \sum_{i=0}^I \pi_i E\{N_n/N_k=i\}^2 \quad (3.10)$$

Clearly (3.9) and (3.10) are, respectively, monotonically increasing and decreasing functions of I , with common limit $MSE(N_n)$. Therefore, they provide lower and upper bounds respectively for the true mean squared error.

The queue length prediction algorithm then, consists of (3.7) for the estimates and (3.8), (3.2), (3.9) and (3.10) for the mean squared error and variance. The necessary probabilities are given by (2.4).

Results: Table 3.1 and Figure 3.1 contain selected OMS prediction curves as functions of the prediction horizon. Several of the results are surprisingly counter-intuitive.

One would expect a predictor to behave in a monotonic fashion; for given data greater than (less than) the mean, the successive estimates would decrease (increase) smoothly to the mean. Yet, as the results show, this is not always the case. Consider the M/M/1 queue with $\rho=.7$, whose mean queue length is $\mu_N=2.33$. Yet based on $N_0=3$, we estimate that 20 arrivals later, the queue length would be 2.24. Consider also the M/M/1 queue with $\rho=.9$, for which $\mu_N=9$ (see Figure 3.1). Based on $N_0=10$, we expect that the queue length will be 8.02 some 50 arrivals later. For $N_0=5$, the expected queue length 10 lags later is 4.80.

The limiting behaviour of all predictors for every queue as the prediction horizon approaches infinity is to converge to the mean. In fact, for $\rho \geq .5$, the situation is similar to that shown in Figure 3.1. For certain initial queue lengths $N_0 > \mu_N$ the predictions drop below the mean at some point and then increase monotonically to the mean. Several queue lengths $N_0 < \mu_N$ have an initial dip as well, and hence their prediction curves are non-monotonic.

What causes the non-monotonic behaviour in the OMS predictors? The best explanation is that there are two conflicting forces in these situations. The initial one-step displacement is generally negative. Note that

$$E\{N_1/N_0=i\} = \sum_{j=1}^{i+1} j k_{i+1-j}$$

After rearranging terms, one obtains

$$E\{N_1/N_0=i\} = (i+1)(1-p_{i0}) - \sum_{k=0}^i k p_{ik}$$

Hence for i sufficiently large¹

$$E\{N_1/N_0=i\} \approx i + (1 - \frac{1}{\rho}) \quad (3.11)$$

which is less than i . Thus initially we expect a decrease in the queue length for each successive lag that we predict. At the same time, the probability of the queue emptying increases the further that we predict. This causes the predictions to increase, since each time the queue empties, the service mechanism stops. In the limit these two balance each other as the steady state tendencies predominate.

3.2.2 System Time Prediction

In this section, the OMS predictor for system time based on queue lengths will be developed. Recall that from (2.14) we have

$$E\{S_n/N_k=i\} = (\frac{1}{\mu})[E\{N_n/N_k=i\}+i] \quad (3.12)$$

This equation provides the predictions. We now seek an expression for $E\{S_n^2/N_k=i\}$, in order to find the predictor variance and mean squared error. Due to the development for (2.14) we may write

$$\begin{aligned} E\{S_n^2/N_k=i\} &= \sum_{j=0}^{\infty} p_{ij}^{n-k} (j+1)(j+2)(\frac{1}{\mu})^2 \\ &= (\frac{1}{\mu})^2 [E\{N_n^2/N_k=i\} + 3E\{N_n/N_k=i\} + 2] \end{aligned}$$

Hence

$$\text{Var}\{S_n/N_k=i\} = E\{S_n^2/N_k=i\} - E\{S_n/N_k=i\}^2$$

¹ In Chapter 5, we show that this behaviour occurs for all $i > 0$ in M/G/1 queues.

$$\begin{aligned}
&= \left(\frac{1}{\mu}\right)^2 [E\{N_n^2/N_k=i\} - E\{N_n/N_k=i\}^2 + E\{N_n/N_k=i\} + 1] \\
&= \left(\frac{1}{\mu}\right)^2 [\text{Var}\{N_n/N_k=i\} + E\{N_n/N_k=i\} + 1] \quad (3.13)
\end{aligned}$$

Furthermore

$$\begin{aligned}
\text{MSE}(S_n) &= \sum_{i=0}^{\infty} \pi_i \text{Var}\{S_n/N_k=i\} \\
&= \left(\frac{1}{\mu}\right)^2 \sum_{i=0}^{\infty} \pi_i \text{Var}\{N_n/N_k=i\} + \left(\frac{1}{\mu}\right)^2 \left[\sum_{i=0}^{\infty} \pi_i (E\{N_n/N_k=i\} + 1) \right] \\
&= \left(\frac{1}{\mu}\right)^2 \text{MSE}(N_n) + \left(\frac{1}{\mu}\right)^2 (\mu_N + 1) \quad (3.14)
\end{aligned}$$

Due to (3.12), we may also write

$$\text{MSE}(S_n) = \left(\frac{1}{\mu}\right)^2 \text{MSE}(N_n) + \left(\frac{1}{\mu}\right) \mu_S \quad (3.15)$$

Equations (3.12) and (3.15) provide a direct link for the predictions and errors between the system time process and the queue length process.

The sequel for the waiting time process follows.

3.2.3 Waiting Time Prediction

The waiting time results are found in the same manner and are summarized here.

$$E\{W_n/N_k=i\} = \left(\frac{1}{\mu}\right) E\{N_n/N_k=i\} \quad (3.16)$$

$$\text{Var}\{W_n/N_k=i\} = \left(\frac{1}{\mu}\right)^2 [\text{Var}\{N_n/N_k=i\} + E\{N_n/N_k=i\}] \quad (3.17)$$

and

$$\text{MSE}(W_n) = \left(\frac{1}{\mu}\right)^2 [\text{MSE}(N_n) + \mu_N] \quad (3.18)$$

3.2.4 System and Waiting Time Results

The preceding derivations indicate a significant relationship between the queue length predictor and the system and waiting time predictors. From equations (3.12) and (3.16) we see that the shape of the waiting time curves is the same as the queue length ones, only scaled by $(\frac{1}{\mu})$; the system time curves are similarly related, only in addition, there is a positive shift of $(\frac{1}{\mu})$. Hence, the non-monotonicities of several queue length curves also apply to the corresponding system and waiting time curves.

The mean squared errors $MSE(S_n)$ and $MSE(W_n)$ contain two components: the former is equal to $(\frac{1}{\mu})^2 MSE(N_n)$ and is a function of the prediction horizon. The latter is constant and reflects the uncertainty between the queue length seen by a customer upon arrival and his ensuing system (or waiting) time. From (3.14), we see that for system times this error is $(\frac{1}{\mu})^2(\mu_N+1)$, while for waiting times, (3.18) shows it to be $(\frac{1}{\mu})^2(\mu_N)$.

3.2.5 Algorithm for the OMS Predictors

Let us now summarize the calculation of the various predictions in the following algorithm.

Step 1: Select $N=\max(n)$ and $I=\max(i)$ for which $E\{Y_n/N_k=i\}$ is sought.

Step 2: For $i=0, \dots, N+I$; $\ell=0, \dots, i+1$ calculate $p_{i\ell}$ using (2.4)

Step 3: For $n=1, \dots, N$ and $i=0, \dots, N+I-n$ calculate $E\{N_n/N_k=i\}$, $E\{S_n/N_k=i\}$ and $E\{W_n/N_k=i\}$ using (3.7), (3.12) and (3.16). These are the predictors.

Step 4: For $n=1, \dots, N$ and $i=0, \dots, N+I-n$ calculate $E\{N_n^2/N_k=i\}$ and $\text{Var}\{N_n/N_k=i\}$ using (3.8) and (3.2), and the other respective variances using (3.13) and (3.17).

Step 5: For $n=1, \dots, N$ calculate the mean squared error bounds $MSE_{L(I)}(N_n)$

and $MSE_{U(I)}(N_n)$ using (3.9) and (3.10).

Step 6: If the discrepancy between the bounds is sufficiently small, set $MSE(N_n)$ to be the upper bound. If the bounds are not sufficiently close, increase I and extend the calculations of steps 2 through 4; then return to step 5.

Step 7: For $n=1, \dots, N$ calculate $MSE(S_n)$ and $MSE(W_n)$ using (3.14) and (3.18).

This concludes the analysis of nonlinear OMS predictors. In the next section, the OLMS predictors are considered, following which, several examples comparing the two types are considered.

3.3 A Study of OLMS Prediction

We now consider the unbiased optimal linear mean squared (OLMS) predictor for N_n , S_n , and W_n based on a sequence of previous queue lengths $N_k = \{N_0, \dots, N_k\}$. In the first appendix the correlation coefficients for the queue length, system time, and waiting time processes are derived. In addition, the cross-correlation coefficients between the queue length process and each of the other processes are found. It is the purpose of this section to apply the knowledge gained about these coefficients to the study of the 3 predictors under consideration.

It is well-known (e.g. Lewis & Odell [7], p. 49) that the OLMS predictor for any random variable Y in terms of a stationary time series such as $N_k = \{N_0, \dots, N_k\}$ is

$$L\{Y/N_k\} = \mu_Y + \sum_{i=0}^K \alpha_i (Y/N_k)(N_i - \mu_N) \quad (3.19)$$

The predictor coefficients $\alpha_i(Y/N_k)$, $i=0, \dots, k$, are obtained from the set of consistent linear equations $A\alpha=b$ where the matrix A and vector b are defined by

$$\begin{aligned} A_{ij} &= \text{Cov}(N_i, N_j) & i, j=0, \dots, k \\ b_i &= \text{Cov}(Y, N_i) & i=0, \dots, k \end{aligned} \quad (3.20)$$

The mean squared error of the estimate is denoted by $\text{LMSE}(Y)$, and is given by

$$\text{LMSE}(Y) = \sigma_Y^2 - \sum_{i=0}^k \alpha_i(Y/N_k) \text{Cov}\{Y, N_i\} \quad (3.21)$$

The determination of the linear predictor reduces to determining the covariances needed for A and b . It is in this regard that the correlation work is applied.

We start by considering $L\{N_n/N_k\}$.

3.3.1 Queue Length Linear Prediction

The matrix equation $A\alpha=b$ can be rewritten as

$$\sum_{j=0}^k \text{Cov}\{N_i, N_j\} \alpha_j(Y/N_k) = \text{Cov}\{Y, N_i\} \quad i=0, \dots, k \quad (3.22)$$

In this case $Y=N_n$, so substituting for Y yields

$$\sum_{j=0}^k \text{Cov}\{N_i, N_j\} \alpha_j(N_n/N_k) = \text{Cov}\{N_n, N_i\} \quad i=0, \dots, k \quad (3.23)$$

The OLMS queue length predictor and error are then given by

$$L\{N_n/N_k\} = \mu_N + \sum_{i=0}^k \alpha_i(N_n/N_k) (N_i - \mu_N) \quad (3.24)$$

$$\text{LMSE}(N_n) = \sigma_N^2 - \sum_{i=0}^k \alpha_i (N_n/N_k) \text{Cov}\{N_n, N_i\} \quad (3.25)$$

Although the OMS predictor was shown to be solely dependent upon the last queue length N_k of the time series $N_k = \{N_0, \dots, N_k\}$, the OLMS predictor depends on more than one, and in general all, of the available data. A solution of the form

$$\alpha_i(N_n/N_k) = 0 \quad i=0, \dots, k-1$$

$$\alpha_k(N_n/N_k) > 0$$

would imply that

$$\alpha_k(N_n/N_k) \text{Cov}\{N_i, N_k\} = \text{Cov}\{N_n, N_i\} \quad i=0, \dots, k$$

Dividing by σ_N^2 leads to

$$\alpha_k(N_n/N_k) r_{N_i}(k-i) = r_{N_i}(n-i) \quad i=0, \dots, k$$

This last condition would require that $r_N(n)$ be a geometric sequence. Only wide-sense Markov processes have such correlation properties. As this is not the case with $r_N(n)$, the predictor is affected by earlier data.

Results: Let us now reconsider Figure 3.1, which represents an M/M/1 queue with $\rho=0.9, \mu_N=9.0$. In contrast to the nonlinear predictors, the OLMS predictors are all monotonic functions of the prediction horizon. Returning also to table 3.1, we see that this is equally true when $\rho=0.7, \mu_N=2.33$. Furthermore, we see that the linear mean squared errors are all slightly larger than the nonlinear errors, which reflects the constraining effect of the linearity restriction.

Several more examples will be presented in section 3.4. Let us now consider the prediction of system times.

3.3.2 System Time Linear Prediction

Based on (3.22), the coefficients $\alpha_j(S_n/N_k)$ are the solution to

$$\sum_{j=0}^k \text{Cov}\{N_i, N_j\} \alpha_j(S_n/N_k) = \text{Cov}\{S_n, N_i\} \quad i=0, \dots, k \quad (3.26)$$

Since $i \leq n$, (A1.20) implies that

$$\text{Cov}\{S_n, N_i\} = \left(\frac{1}{\mu}\right) \text{Cov}\{N_n, N_i\} \quad i=0, \dots, k$$

so that (3.26) becomes

$$\sum_{j=0}^k \text{Cov}\{N_i, N_j\} \alpha_j(S_n/N_k) = \text{Cov}\{N_n, N_i\} \left(\frac{1}{\mu}\right) \quad i=0, \dots, k \quad (3.27)$$

Comparing (3.27) with (3.23) we observe that

$$\alpha_j(S_n/N_k) = \alpha_j(N_n/N_k) \left(\frac{1}{\mu}\right) \quad j=0, \dots, k \quad (3.28)$$

The system time equations are therefore

$$L\{S_n/N_k\} = \mu_S + \sum_{i=0}^k \alpha_i(S_n/N_k) (N_i - \mu_N) \quad (3.29)$$

$$\text{LMSE}(S_n) = \sigma_S^2 - \sum_{i=0}^k \alpha_i (S_n/N_k) \text{Cov}\{S_n, N_i\} \quad (3.30)$$

Noting that in GI/M/1

$$\mu_S = \left(\frac{1}{\mu}\right)(\mu_N + 1) \quad (3.31)$$

and substituting for $\alpha_j(S_n/N_k)$ in (3.29) using (3.28) we obtain

$$L\{S_n/N_k\} = \left(\frac{1}{\mu}\right)[L\{N_n/N_k\} + 1] \quad (3.32)$$

Substituting for $\alpha_i(S_n/N_k)$ and $\text{Cov}\{S_n, N_i\}$ in (3.30) yields

$$\begin{aligned} \text{LMSE}(S_n) &= \sigma_S^2 - \left(\frac{1}{\mu}\right)^2 \sum_{i=0}^k \alpha_i (N_n/N_k) \text{Cov}\{N_n, N_i\} \\ &= \sigma_S^2 - \left(\frac{1}{\mu}\right)^2 (\sigma_N^2 - \text{LMSE}(N_n)) \\ &= \left(\frac{1}{\mu}\right)^2 \text{LMSE}(N_n) + \left(\frac{1}{\mu}\right)^2 (\mu_N + 1) \end{aligned} \quad (3.33)$$

Notice at this point that (3.32) and (3.33) closely resemble the nonlinear equations (3.11) and (3.14). In general, then, when using queue length data, the system time estimates are obtained from the queue length estimates by a shift and a scaling, both in the linear and nonlinear cases. As well, the mean squared error of both the linear and nonlinear system time predictors contains a constant component $\left(\frac{1}{\mu}\right)^2 (\mu_N + 1)$, whose cause was explained in section 3.2.4. In the case of the linear predictor, (3.32) shows that $L\{S_n/N_k\}$ is dependent upon the earlier data $N_0, \dots, N_{k-1}$, since $L\{N_n/N_k\}$ depends on them. These statements apply equally well to the waiting time OLMS predictor, which we consider next.

3.3.3 Waiting Time Linear Prediction

By following the same approach as the last section, the following results are obtained:

$$L\{W_n/N_k\} = \left(\frac{1}{\mu}\right) L\{N_n/N_k\} \quad (3.34)$$

$$LMSE(W_n) = \left(\frac{1}{\mu}\right)^2 LMSE(N_n) + \left(\frac{1}{\mu}\right)^2 \mu_N \quad (3.35)$$

3.3.4 Algorithm for the OLMS predictors

The linear equivalent to the algorithm of section 3.2.5 consists of:

Step 1: Select $N = \max(n)$ for which $L\{Y_n/N_k\}$ is sought; $Y = S, N, W$. Select $0 \leq k < n$.

Step 2: Calculate $r_N(l)$, $l = 0, \dots, N$ using Pakes algorithm as described by (A1.11).

Step 3: Solve the set of equations (3.23), where $\text{Cov}\{N_i, N_j\} = \sigma_N^2 r_N(i-j)$.

Step 4: Compute $L\{N_n/N_k\}$, $L\{S_n/N_k\}$ and $L\{W_n/N_k\}$ using (3.24), (3.32) and (3.34) respectively, for all desired combinations $N_k = \{N_0, \dots, N_k\}$. These are the predictors.

Step 5: Compute $LMSE(N_n)$, $LMSE(S_n)$, and $LMSE(W_n)$ using (3.25), (3.33) and (3.35) respectively. These are the linear mean squared errors.

This completes the study of OLMS predictors based on previous queue lengths. In the next section, several comparisons are made between the OMS predictors of section 3.3 and the OLMS predictors of this section.

3.4 A Comparison of OMS and OLMS Queue Length Predictors

Computations were undertaken to show the predictive power of linear and nonlinear predictors, as indicated by the difference between the mean squared error and its limiting value, which is equal to the variance of the process being predicted. Various values of the traffic intensity ρ , prediction horizon $(n-k)$, and different arrival distributions were used.

In the Tables of this section most of the data are normalized by the variance of the corresponding variable, to avoid scaling problems and to facilitate comparisons between cases.

Table 3.2 provides a general perspective. It presents the normalized mean squared errors of the OMS and single observation OLMS predictors for N_n over a variety of utilizations and distances predicted. As expected, the error increases with the distance (i.e. number of lags) predicted. However, as ρ increases to 1, the error decreases to a limit of zero.

Table 3.3 presents the normalized errors for S_n with the same parameters as Table 3.2. Qualitatively, the results are the same. However, the system time error is consistently larger, reflecting the uncertainty between a customer's queue length upon arrival and his ensuing system time.

Table 3.4 compares the normalized MSE's for N_n in several Erlang queues. The effect of an increase in the order R of the Erlang density is to increase the prediction error. This follows from the fact that as R increases, each of the processes under study becomes less correlated, and hence the accuracy of the predictor decreases.

Table 3.5 deals with the usefulness of additional, earlier data in the linear predictor. As can be seen, the normalized mean squared error of the queue length predictor drops only slightly as more data is used and rarely exceeds a 1% reduction when 5 data values are used. By way of contrast, the optimal nonlinear predictor offers improvements over the single observation linear predictor of 2-5%. This improvement depends upon the utilization, the interarrival density, and the prediction horizon.

Table 3.6 contains typical predictor coefficients for a 5-observation linear predictor in the M/M/1 queue with $\rho=0.7$. The mean service time ($\frac{1}{\mu}$) has been chosen to be 1.0 so that the predictor coefficients are equally valid for queue length, system and waiting time predictions. Several characteristics are apparent:

- 1) The latest observation is the most heavily weighted; the earlier coefficients are only slightly greater than 0. Recalling that the queue length process is not a wide-sense Markov process, it is not surprising that the coefficients are non-zero. Since they are very small we can conclude that their effect upon the prediction is minimal.
- 2) In general $\alpha_i < \alpha_{i+1}$ for $i=1, \dots, n-1$; but $\alpha_0 > \alpha_1$. Older data is given less weight, but α_0 is a surrogate for the infinity of even earlier data values, and receives extra weight for this.
- 3) The value of a particular earlier coefficient rises and then falls as the distance predicted increases. This "peaking" effect is due to two opposing tendencies. The former is towards a more balanced weighting of the data in the overall prediction, causing the coefficients to rise. The latter is to predict a value closer to the mean as the prediction horizon goes to infinity, which causes all coefficients to decrease.

In addition to the predictor coefficients, Table 3.6 lists the normalized errors of the two predictors $L\{S_n/S_k\}$ and $L\{S_n/N_k\}$. The question of the viability of the queue-length based predictions for system times is discussed in the next section.

3.5 A Comparison of $L\{S_n/N_k\}$ and $L\{S_n/S_k\}$

The linear predictor for system times based on earlier system times has the same form as the foregoing ones, with the appropriate replacements for the time series $S_k = \{S_0, \dots, S_k\}$. We have

$$L\{S_n/S_k\} = \mu_S + \sum_{i=0}^k \alpha_i(S_n/S_k)(S_i - \mu_S) \quad (3.36)$$

$$\text{LMSE}(S_n; S_0) = \sigma_S^2 - \sum_{i=0}^k \alpha_i(S_n/S_k) \text{Cov}\{S_n, S_i\} \quad (3.37)$$

where the $\alpha_i(S_n/S_k)$, $i=0, \dots, k$ are the solution of

$$\sum_{j=0}^k \text{Cov}\{S_i, S_j\} \alpha_j(S_n/S_k) = \text{Cov}\{S_n, S_i\} \quad i=0, \dots, k \quad (3.38)$$

Recall that $r_S(n) = r_N(n) \forall n$ in the GI/M/1 queue. Thus, by multiplying (3.38) by σ_N^2/σ_S^2 , we get

$$\sum_{j=0}^k \text{Cov}\{N_i, N_j\} \alpha_j(S_n/S_k) = \text{Cov}\{N_n, S_i\} \quad i=0, \dots, k \quad (3.39)$$

But, due to (3.23), this implies that

$$\alpha_j(S_n/S_k) = \alpha_j(N_n/N_k) \quad j=0, \dots, k \quad (3.40)$$

Furthermore, substituting as appropriate, we find

$$\text{LMSE}(S_n; S_0) = \sigma_S^2 - \sum_{i=0}^k \alpha_i (N_n/N_k) \text{Cov}\{N_n, N_i\} \sigma_S^2 / \sigma_N^2 \quad (3.41)$$

and so

$$\frac{\text{LMSE}(S_n; S_0)}{\sigma_S^2} = \frac{\text{LMSE}(N_n)}{\sigma_N^2} \quad (3.42)$$

Equations (3.40) and (3.42) tell us two things:

- 1) The system times $S_0, \dots, S_k$ in (3.36) are weighted identically to the queue lengths $N_0, \dots, N_k$ in (3.23).
- 2) The normalized mean squared error for $L\{S_n/S_k\}$ is equal to the normalized mean squared error for $L\{N_n/N_k\}$.

The more realistic comparisons, however, are between $L\{S_n/N_k\}$ and $L\{S_n/S_k\}$. Due to (3.40) and (3.28) we can say

$$L\{S_n/N_k\} = \mu_S + \sum_{i=0}^k \alpha_i (S_n/S_k) ((N_i - \mu_N) (\frac{1}{\mu})) \quad (3.43)$$

while (3.33) and (3.42) imply

$$\text{LMSE}(S_n) = d \text{LMSE}(S_n; S_0) + (1-d) \sigma_S^2 \quad (3.44)$$

► Typical results are represented by Figure 3.2.

Thus, a greater degree of information is contained in S_k instead of N_k (compare 3.43 with 3.36). In addition, the error using N_k is a weighted average of the error using S_k and the (larger) error based on no conditioning (which is the variance). Therefore, it would seem as though predictions

based on S_k are preferable; however, several factors would imply otherwise.

The flaw in the error comparison lies in the time at which the information becomes available. The system times always become available later, perhaps so late as to be irrelevant.

Consider the single immediately previous system time as an example (predicting S_1 from S_0). S_0 is first known at the moment C_1 enters service if the queue is busy; by then W_1 is also known and the more appropriate action would be to predict S_1 as $W_1 + (\frac{1}{\mu})$, regardless of the value of S_0 . Linear prediction using S_0 is hence absurd, basically due to the delay in obtaining S_0 . This delay has not been accounted for in our accuracy calculations, and its length has an average value μ_W . In many applications this delay would be significant, and in every case it should enter the comparison. The result seems to tip in favour of using N_k . It is possible to perform a numerical comparison of $L\{W_n/N_k\}$ with the predictor $L\{W_n/W_k\}$. But since the correlation coefficients for the latter, $r_S(n)$, have no direct relationship to $r_{NW}(n)$, no equations similar to (3.43) and (3.44) exist. Qualitatively, the comments regarding delays in obtaining information apply here as well.

3.6 Predictions Based on Current Queue Length

Prior to leaving prediction in GI/M/1 queues, we consider the effect of knowing the current queue length on our predictions. The linear predictor for system time would have coefficients $\alpha_1(S_n/N_k)$ which were the solution to

$$\sum_{j=0}^n \text{Cov}\{N_i, N_j\} \alpha_j(N_n/N_n) = \text{Cov}\{N_n, N_i\} \left(\frac{1}{\mu}\right) \quad i=0, \dots, k$$

The solution is clearly

$$\alpha_j(N_n/N_n) = \begin{cases} 0 & j \neq n \\ \left(\frac{1}{\mu}\right) & j = n \end{cases} \quad (3.45)$$

When the right substitutions are made, the OLMS result is equal to the OMS result, (2.14). The errors are equal as well, and, as always, similar conclusions can be arrived at in the case of waiting times.

3.7 Conclusions

In the midst of the seemingly endless equations of this Chapter, five major points can be noted:

- 1) The system and waiting time predictors use the queue length predictors as an intermediate result.
- 2) The optimal prediction curve may be a non-monotonic function of the prediction horizon.
- 3) Linear predictors are almost optimal.
- 4) Earlier data gives only a tiny improvement in the linear predictor.
- 5) It seems preferable to use queue length data (rather than earlier system times) for predicting system times.

TABLE 3.1

Prediction of N_n and S_n M/M/1 $\rho=.7$ $\mu=1.0$ $\mu_N=2.33$

n	$E\{N_n/N_0=i\}$				$L\{N_n/N_0=i\}$			
	i=0	i=3	i=6	i=9	i=0	i=3	i=6	i=9
1	0.41	2.74	5.61	8.58	0.21	2.94	5.67	8.40
2	0.68	2.58	5.26	8.17	0.38	2.89	5.39	7.90
3	0.88	2.47	4.95	7.78	0.53	2.85	5.16	7.48
5	1.17	2.35	4.45	7.07	0.78	2.78	4.78	6.78
20	1.97	2.24	2.93	4.10	1.71	2.52	3.32	4.13
∞	2.33	2.33	2.33	2.33	2.33	2.33	2.33	2.33

 $\sigma_N^2=7.77$ $\text{Var}\{N_n/N_0=i\}$

n	i=0	i=3	i=6	i=9
1	0.24	1.90	2.95	3.32
2	0.56	3.04	5.28	6.36
3	0.89	3.78	7.07	9.04
5	1.54	4.68	9.42	13.28
20	4.81	6.55	11.00	18.43

 $\sigma_S^2=11.11$ $\text{Var}\{S_n/N_0=i\}$

i=0	i=3	i=6	i=9
1.65	5.64	9.56	12.90
2.24	6.62	11.54	15.53
2.77	7.25	12.02	17.82
3.71	8.03	14.87	21.35
7.78	9.79	14.93	22.53

TABLE 3.2

Normalized Errors for Queue Length Predictions
in M/M/1, ρ varying

(n-k)	$\rho=.5$		$\rho=.7$		$\rho=.9$	
	Linear	Nonlinear	Linear	Nonlinear	Linear	Nonlinear
1	.4375	.4286	.1719	.1677	.0199	.0196
2	.6597	.6448	.3013	.2910	.0386	.0377
3	.7857	.7696	.4038	.3882	.0563	.0548
5	.9084	.8957	.5561	.5335	.0897	.0865
10	.9862	.9821	.7709	.7441	.1636	.1558
20	.9995	.9992	.9280	.9101	.2842	.2680

TABLE 3.3

Normalized Errors for System Time Predictions
in M/M/1, ρ varying

(n-k)	$\rho=.5$		$\rho=.7$		$\rho=.9$	
	Linear	Nonlinear	Linear	Nonlinear	Linear	Nonlinear
1	.7188	.7143	.4203	.4174	.1179	.1177
2	.8299	.8224	.5109	.5037	.1347	.1340
3	.8928	.8848	.5827	.5717	.1507	.1493
5	.9542	.9478	.6893	.6734	.1807	.1778
10	.9931	.9911	.8396	.8208	.2472	.2403
20	.9998	.9996	.9496	.9371	.3558	.3412

TABLE 3.4

Normalized Errors for Queue Length Predictions
in Erlang Queues $E_R/M/1$, $\rho=.7$

(n-k)	R=1		R=2		R=5	
	Linear	Nonlinear	Linear	Nonlinear	Linear	Nonlinear
1	.1719	.1677	.2243	.2178	.2747	.2657
2	.3013	.2910	.3810	.3663	.4531	.4347
3	.4038	.3882	.4978	.4773	.5785	.5543
5	.5561	.5335	.6589	.6325	.7393	.7113
10	.7709	.7441	.8560	.8313	.9103	.8894
20	.9280	.9101	.9681	.9566	.9861	.9794

TABLE 3.5

Normalized Errors for Queue Length Predictions
with a Varying Number of Observations

(n-k)	M/M/1				
	$\rho=.7$				
	1 Obs.	2 Obs.	3 Obs.	5 Obs.	Nonlinear
1	.1719	.1716	.1714	.1713	.1677
2	.3013	.3005	.3002	.2999	.2910
3	.4038	.4027	.4022	.4018	.3882
5	.5561	.5548	.5541	.5535	.5335
10	.7709	.7697	.7690	.7682	.7441
20	.9280	.9274	.9270	.9266	.9101

TABLE 3.6

Predictor Coefficients and Errors
for the Linear System Time Predictor.

		M/M/1				$\rho = .7$	
(n-k)	α_0	α_1	α_2	α_3	α_4	MSE{S/S}	MSE{S/N}
1	.0148	.0067	.0104	.0207	.8667	.1713	.4199
2	.0246	.0104	.0154	.0282	.7717	.2999	.5100
3	.0313	.0126	.0180	.0312	.6966	.4018	.5813
5	.0389	.0145	.0198	.0321	.5815	.5535	.6874
10	.0425	.0142	.0182	.0272	.3974	.7682	.8378
20	.0331	.0099	.0120	.0169	.2116	.9266	.9486

Fig. 3.1 Linear and Nonlinear Predictions for the M/M/1 Queue with $\rho=.9$

— Nonlinear prediction $E\{N_n/N_0=i\}$
 - - - Linear prediction

Figure 3.2. Normalized Prediction Error for System Times in an M/M/1 Queue. (Prediction error/System time variance)

- Error for predictions based on queue length N_0
- - - Error for predictions based on system time S_0

CHAPTER 4

NONLINEAR PREDICTION IN GI/M/m QUEUES

4.1 Introduction

In the preceding chapter, we considered the optimal predictors (both linear and nonlinear) for the N_n , W_n , and S_n processes in the single server, exponential service queue. Numerous applications arise (in telephony, for example) when single-server results are not applicable, and where a multi-server queue would be more appropriate.

Each of the m servers is exponential at rate μ . Hence, to a waiting job, service completions occur exponentially at rate $m\mu$. This fact forms the cornerstone of our analysis, and leads to the nonlinear predictor for waiting times based on queue length, to be presented in section 4.3.

A major difference between the GI/M/1 and GI/M/m queues is that in the multi-server case, customers do not necessarily leave the system in the order they arrive. This phenomenon is called overtaking, and occurs due to the parallel servicing of customers. Due to this fact, several correlation results, especially those for the system time process, are not easily obtained. However, since the queue is FCFS, jobs enter service in the order they arrive, and so W_{n+1} follows W_n . This fact facilitates computation of the GI/M/m serial correlation coefficients for

waiting times which appear in Appendix 2, and which can be used in an algorithm for the OLMS predictor for W_n based on $W_k = \{W_0, \dots, W_k\}$. Therefore, we will only consider the OMS predictors for N_n , W_n and S_n based on N_0 in this chapter.

4.2 Queue Length Prediction

The quantities of interest are the same as those in the previous chapter, namely:

$$E\{N_n/N_0=i\} = \sum_{j=0}^{\infty} j p_{ij}^n \quad (4.1)$$

$$E\{N_n^2/N_0=i\} = \sum_{j=0}^{\infty} j^2 p_{ij}^n \quad (4.2)$$

$$\text{Var}\{N_n/N_0=i\} = E\{N_n^2/N_0=i\} - E\{N_n/N_0=i\}^2 \quad (4.3)$$

$$\text{MSE}(N_n/N_0) = \sum_{i=0}^{\infty} \pi_i \text{Var}\{N_n/N_0=i\} \quad (4.4)$$

However, the transition probabilities are different and are given by (4.12) below. In the waiting time prediction algorithm of the next section, only the sum over $j=m, \dots, \infty$ will be used. Therefore, in this section we further define the following restricted expectations:

$$E\{N_n; N_n \geq m/N_0=i\} = \sum_{j=m}^{\infty} j p_{ij}^n \quad (4.5)$$

$$E\{N_n; N_n < m/N_0=i\} = \sum_{j=0}^{m-1} j p_{ij}^n \quad (4.6)$$

The expressions $E\{N_n^2; N_n \geq m/N_0=i\}$ and $E\{N_n^2; N_n < m/N_0=i\}$ should be similarly understood. Now by paralleling the derivation of (3.7) and (3.8) we have

$$E\{N_n; N_n \geq m/N_0=i\} = \sum_{\ell=0}^{i+1} p_{i\ell} E\{N_{n-1}; N_{n-1} \geq m/N_0=\ell\} \quad (4.7)$$

$$E\{N_n; N_n < m/N_0=i\} = \sum_{\ell=0}^{i+1} p_{i\ell} E\{N_{n-1}; N_{n-1} < m/N_0=\ell\} \quad (4.8)$$

$$E\{N_n^2; N_n \geq m/N_0=i\} = \sum_{\ell=0}^{i+1} p_{i\ell} E\{N_{n-1}^2; N_{n-1} \geq m/N_0=\ell\} \quad (4.9)$$

$$E\{N_n^2; N_n < m/N_0=i\} = \sum_{\ell=0}^{i+1} p_{i\ell} E\{N_{n-1}^2; N_{n-1} < m/N_0=\ell\} \quad (4.10)$$

The following equation will be needed for the waiting and system time predictors:

$$\Pr\{N_n \geq m/N_0=i\} = \sum_{\ell=0}^{i+1} p_{i\ell} \Pr\{N_{n-1} \geq m/N_0=\ell\} \quad (4.11)$$

Finally, the one step transition probabilities are given by:

$$p_{i\ell} = \begin{cases} 0, & \ell > i+1 \\ \int_0^{\infty} \left[\begin{matrix} i+1 \\ i+1-\ell \end{matrix} \right] (1-e^{-\mu t})^{i+1-\ell} e^{-\mu t \ell} dF_T(t) & \ell \leq i+1 \leq m \\ \left[\begin{matrix} m \\ m-\ell \end{matrix} \right] \frac{(\mu t)^{i-m+1}}{(i-m)!} \int_0^{\infty} \int_0^t q(t-v)^{m-\ell} p(t-v)^{\ell} v^{i-m} e^{-\mu v} dv dF_T(t) & i+1 > m > \ell \\ \int_0^{\infty} e^{-\mu t} \frac{(\mu t)^{i+1-\ell}}{(i+1-\ell)!} dF_T(t) & i+1 \geq \ell \geq m \end{cases} \quad (4.12)$$

where $p(t) = e^{-\mu t}$; $q(t) = 1 - p(t)$

Due to the restriction on its summation,

$$E\{N_0; N_0 \geq m/N_0=i\} = \begin{cases} 0 & i < m \\ i & i \geq m \end{cases}$$

Each of the other terms is given by just as simple an expression. Now notice that

$$E\{N_n/N_0=i\} = E\{N_n; N_n < m/N_0=i\} + E\{N_n; N_n \geq m/N_0=i\} \quad (4.13)$$

$$E\{N_n^2/N_0=i\} = E\{N_n^2; N_n < m/N_0=i\} + E\{N_n^2; N_n \geq m/N_0=i\} \quad (4.14)$$

The prediction variance can now be obtained using (4.3). As before, the mean squared error is found according to a bounding approach, with bounds

$$MSE_{L(I)}(N_n) = \sum_{i=0}^I \pi_i \text{Var}\{N_n/N_0=i\} \quad (4.15)$$

$$MSE_{U(I)}(N_n) = E\{N_n^2\} - \sum_{i=0}^I \pi_i E\{N_n/N_0=i\}^2 \quad (4.16)$$

which converge on the true mean squared error as $I \rightarrow \infty$.

These equations will be summarized in an algorithm in section 4.5.

4.3 Waiting Time Prediction

We seek the predictor for W_n based upon knowing an earlier queue length $N_0=i$, as well as its variance and mean squared error. We will use

$$E\{W_n/N_0=i\} = \int_0^{\infty} t d\Pr\{W_n \leq t/N_0=i\}$$

By paralleling the derivation of (2.14) we obtain

$$\begin{aligned}
E\{W_n/N_0=i\} &= \sum_{j=0}^{\infty} p_{ij}^n E\{W_n/N_n=j\} \\
&= \sum_{j=0}^{m-1} p_{ij}^n \cdot 0 + \sum_{j=m}^{\infty} p_{ij}^n (j+1-m) \left(\frac{1}{m\mu}\right) \\
&= \left(\frac{1}{m\mu}\right) \left[\sum_{j=m}^{\infty} j p_{ij}^n - (m-1) \sum_{j=m}^{\infty} p_{ij}^n \right] \\
&= \left(\frac{1}{m\mu}\right) [E\{N_n; N_n \geq m/N_0=i\} - (m-1) \Pr\{N_n \geq m/N_0=i\}] \quad (4.17)
\end{aligned}$$

The second moment is

$$\begin{aligned}
E\{W_n^2/N_0=i\} &= \sum_{j=m}^{\infty} p_{ij}^n (j+1-m)(j+2-m) \left(\frac{1}{m\mu}\right)^2 \\
&= \left(\frac{1}{m\mu}\right)^2 \left[\sum_{j=m}^{\infty} j^2 p_{ij}^n - (2m-3) \sum_{j=m}^{\infty} j p_{ij}^n + (m-1)(m-2) \sum_{j=m}^{\infty} p_{ij}^n \right] \\
&= \left(\frac{1}{m\mu}\right)^2 [E\{N_n^2; N_n \geq m/N_0=i\} - (2m-3)E\{N_n; N_n \geq m/N_0=i\} \\
&\quad + (m-1)(m-2)\Pr\{N_n \geq m/N_0=i\}] \quad (4.18)
\end{aligned}$$

The variance is now found to be

$$\begin{aligned}
\text{Var}\{W_n/N_0=i\} &= \left(\frac{1}{m\mu}\right)^2 [E\{N_n^2; N_n \geq m/N_0=i\} \\
&\quad - (2m-3)E\{N_n; N_n \geq m/N_0=i\} + (m-1)(m-2)\Pr\{N_n \geq m/N_0=i\} \\
&\quad - (E\{N_n; N_n \geq m/N_0=i\} - (m-1)\Pr\{N_n \geq m/N_0=i\})^2] \\
&\quad (4.19)
\end{aligned}$$

The mean squared error is then

$$MSE(W_n) = \sum_{i=0}^{\infty} \pi_i \text{Var}\{W_n/N_0=i\} \quad (4.20)$$

However, no further simplification such as (3.14) is possible. Nonetheless, we can obtain the MSE from bounds similar to those for $MSE(N_n)$. These are

$$MSE_{L(I)}(W_n) = \sum_{i=0}^I \pi_i \text{Var}\{W_n/N_0=i\} \quad (4.21)$$

and

$$\begin{aligned} MSE_{U(I)}(W_n) &= \left(\frac{1}{m\mu}\right)^2 [E\{N_n^2; N_n \geq m\} - (2m-3)E\{N_n; N_n \geq m\} \\ &\quad + (m-1)(m-2)\Pr\{N_n \geq m\}] \\ &\quad - \left(\frac{1}{m\mu}\right)^2 \sum_{i=0}^I \pi_i [E\{N_n; N_n \geq m/N_0=i\} - (m-1)\Pr\{N_n \geq m/N_0=i\}]^2 \end{aligned} \quad (4.22)$$

4.4 System Time Prediction

The system time differs from the waiting time by one extra independent exponential component with mean $(\frac{1}{\mu})$. This leads to the result

$$E\{S_n/N_0=i\} = E\{W_n/N_0=i\} + \left(\frac{1}{\mu}\right) \quad (4.23)$$

Due to the independence of the waiting and service stages, we find

$$\text{Var}\{S_n/N_0=i\} = \text{Var}\{W_n/N_0=i\} + \left(\frac{1}{\mu}\right)^2 \quad (4.25)$$

Finally, the bounds on the mean squared error are

$$MSE_{L(I)}(S_n) = MSE_{L(I)}(W_n) + \left(\frac{1}{\mu}\right)^2 \quad (4.26)$$

and

$$\text{MSE}_{U(I)}(S_n) = \text{MSE}_{U(I)}(W_n) + \left(\frac{1}{\mu}\right)^2 \quad (4.27)$$

These results are now summarized in an algorithm.

4.5 Algorithm for the OMS Predictors

Step 1: Select $N=\max(n)$ and $I=\max(i)$ for which $E\{Y_n/N_0=i\}$ is sought.

Step 2: For $i=0, \dots, N+I, \ell=0, \dots, i+1$ calculate $p_{i\ell}$ using (4.12).

Step 3: For $n=1, \dots, N$ and $i=0, \dots, N+I-n$ calculate $E\{N_n; N_n \geq m/N_0=i\}$; $E\{N_n; N_n < m/N_0=i\}$ and $\text{Pr}\{N_n \geq m/N_0=i\}$ using (4.7), (4.8) and (4.11) and then $E\{N_n/N_0=i\}$, $E\{W_n/N_0=i\}$ and $E\{S_n/N_0=i\}$ using (4.13), (4.17), and (4.23). These are the predictors.

Step 4: For $n=1, \dots, N$ and $i=0, \dots, N+I-n$ calculate $E\{N_n^2; N_n \geq m/N_0=i\}$, $E\{N_n^2; N_n < m/N_0=i\}$ using (4.9) and (4.10), and then $E\{N_n^2/N_0=i\}$, $\text{Var}\{N_n/N_0=i\}$, $\text{Var}\{W_n/N_0=i\}$ and $\text{Var}\{S_n/N_0=i\}$ using (4.14), (4.3), (4.19) and (4.25).

Step 5: For $n=1, \dots, N$ calculate the mean squared error bound for each process: for N_n using (4.15) and (4.16); for W_n using (4.21) and (4.22); for S_n using (4.26) and (4.27).

Step 6: If the discrepancy between the bounds for each process is sufficiently small, set the respective errors to the respective upper bounds. If any of the bounds are not sufficiently close, increase I and extend the calculations of steps 2 through 4; then return to step 5.

4.6 Numerical Examples

A number of computational examples were undertaken to reveal the effect upon the predictions (and the predictive power) of such parameters as the traffic intensity ρ , arrival process variability C_a^2 , and number of servers m . Results were obtained for both the queue length and waiting time predictors. Furthermore, examples were done to observe the rate of convergence of the bounding process for $MSE(N_n)$ and $MSE(W_n)$.

In the last chapter, we observed that the prediction trajectories initially decreased linearly for i sufficiently large. A broader statement can be made: in ergodic GI/G/m queues, we expect $(\frac{1}{\rho})$ departures from the queue for every arrival, so long as all m servers remain busy. This can be stated mathematically as follows: given any combination of the initial queue length i , traffic intensity ρ , and prediction horizon n such that $\Pr\{N_k < m/N_0 = i\}$ is small for all $k \leq n$, we have

$$\begin{aligned} E\{N_n/N_0 = i\} &\approx i + n - n\left(\frac{1}{\rho}\right) \\ &\approx i - n\left(\frac{1-\rho}{\rho}\right) \end{aligned} \quad (4.29)$$

In the case where $n=1$, (4.29) reduces to (3.11).

Equation (4.29) shows that for as long as all servers are busy, the single server (s.s.) and multiserver queues handle the unfinished work similarly.

The key to this approximation is $\Pr\{N_k < m/N_0 = i\}$, which is a decreasing function of i and an increasing function of n (assuming $i > m$). Thus for n small, we anticipate a wide range of prediction trajectories to be given by (4.29). As n increases we expect the prediction trajectories to deviate

from this approximation and to approach the unconditional mean. Furthermore, the changeover in tendencies may produce overshoots similar to those observed in the s.s. queue.

Table 4.1 presents trajectories of $E\{N_n/N_0=i\}$ for $i=0, 3, 6$, and 9 in the case of one, two, three and five-server M/M/m queues with traffic intensity $\rho=0.7$. In the case $i=0$, the four trajectories are dissimilar; this is not surprising as $\Pr\{N_n < m/N_0=0\}$ is not small for any value of n . Next, with $i=3$, all servers are busy for $m=1, 2$, and 3 but not for $m=5$. Hence the 1-lag predictions of the first three decrease while the last one, not serving at full capacity, increases. In addition, we note the non-monotonic behaviour of the one, two and three-server prediction trajectories.

In the case of $i=6$, the 1-lag predictions are all reasonably similar, but as n increases, the predictions quickly diverge. In the case of $i=9$, the predictions are almost identical up to 5 lags, after which they start to diverge significantly.

The prediction trajectories of the two most dissimilar of these queues, the s.s. and five-server queues are shown in Figure 4.1, where the short-term linear behaviour of the trajectories for $i=9$ is apparent.

Table 4.2 contains the waiting time predictions for the same queues as Table 4.1. For the purpose of more meaningful comparisons, we have chosen the service rate $\mu=(\frac{1}{m})$, so that service completions from a busy queue occur at the rate of $m\mu=1.0$ in all cases. As expected, an increase in m leads to a decrease in the predicted wait.

In s.s. queues, each waiting time prediction trajectory had the same shape as the queue length trajectory with the same parameters. This is not the case in the multi-server queue. For example, as Tables 4.1 and 4.2 indicate the queue length trajectory for $i=3$ and $m=3$ is not monotonic, but the corresponding waiting time trajectory is.

Finally, we note that if $i+n < m$, C_n can never wait, and so $W_n=0$ exactly.

In order to study the effect of some non-Markovian arrivals, Erlang arrivals of degree $R=2, 3$ and 5 were considered. The results for queue lengths are given in Table 4.3. The effect of an increase in R is a decrease in the predicted value, due to the reduced variability of the arrival process ($C_a^2 = \frac{1}{R}$). It would appear as well that as R increases, the tendency of the prediction to overshoot disappears (as in the case of $i=3$), but this is actually a result of the changing mean queue length. The combination of $i=2$, $R=5$ leads to a non-monotonic prediction curve, as can be seen from Figure 4.2. These comments apply as well to the waiting time predictions, which are listed in Table 4.4.

Tables 4.5 to 4.8 deal with the predictive power of the multiserver predictors. As in the last chapter, the mean squared error has been normalized by dividing by the variance, so as to facilitate comparisons between cases. Tables 4.5 to 4.7 consider the normalized errors over a range of utilizations in the case of $m=1, 2$, and 5 servers. As one would expect, the error for both processes increases with n and decreases with ρ .

Unlike the s.s. results, the normalized errors for waiting time are not consistently larger than those for queue lengths. The M/M/5 queue with

$\rho=0.7$ reflects this best. Initially, the waiting time error is larger, but some 10 lags later, the situation has reversed.

Noteworthy is the effect of the number of servers upon the normalized prediction errors. For queue lengths, the errors peak at a particular value of m . For example, when $\rho=0.5$ and $n=1$ lag, the maximum error is found when $m=2$. As ρ or n increases, so does the value of m corresponding to the maximum error. However, the corresponding waiting time normalized prediction errors are decreasing functions of m .

The effect of the traffic intensity ρ on the bounding process is revealed in Tables 4.6 and 4.7 by the separation of the lower and upper bounds for a fixed summation limit of $I=40$. The bounds agree to three figures at $\rho=0.7$ but diverge widely at $\rho=0.9$ in both cases. Hence I should be increased to 100 or 200, and the necessary computations extended.

The last table (4.8) is designed to show the effect of an increase in R upon the errors in the case of a 2-server queue. The results here agree with those for single server queues: an increase in R leads to an increase in the prediction error, for both processes.

4.7 Conclusions

In this chapter, the special properties of the GI/M/m queue were exploited to produce an algorithm to compute the OMS predictors for N_n , W_n , and S_n based on earlier queue length information. In numerical examples, we observed the same non-monotonic behaviour of certain prediction trajectories.

as had been observed in single server queues. Several differences were observed between the s.s. and multiserver queues in a study of the predictive power of both predictors.

In the next chapter, we consider OMS prediction in the $M/G/1$ -queue.

TABLE 4.1

Values of $E\{N_n/N_0=1\}$ in M/M/m, $\rho=0.7$

Horizon	i=0				i=3			
	m=1	m=2	m=3	m=5	m=1	m=2	m=3	m=5
1	0.41	0.58	0.68	0.78	2.74	2.78	2.84	3.11
2	0.68	0.92	1.14	1.38	2.58	2.65	2.78	3.20
3	0.88	1.16	1.45	1.85	2.47	2.58	2.75	3.28
5	1.17	1.50	1.86	2.50	2.35	2.52	2.76	3.44
20	1.97	2.36	2.84	3.89	2.24	2.59	3.03	4.05
∞	2.33	2.75	3.24	4.38	2.33	2.75	3.24	4.38

Horizon	i=6				i=9			
	m=1	m=2	m=3	m=5	m=1	m=2	m=3	m=5
1	5.61	5.61	5.63	5.68	8.58	8.58	8.58	8.59
2	5.26	5.28	5.31	5.44	8.17	8.18	8.19	8.22
3	4.95	4.99	5.05	5.25	7.78	7.79	7.81	7.88
5	4.45	4.53	4.64	4.98	7.07	7.10	7.15	7.29
20	2.93	3.21	3.57	4.44	4.10	4.31	4.58	5.26
∞	2.33	2.75	3.24	4.38	2.33	2.75	3.24	4.38

Horizon	MSE(N_n)			
	m=1	m=2	m=3	m=5
1	1.30	1.39	1.46	1.57
2	2.26	2.39	2.52	2.74
3	3.02	3.18	3.34	3.66
5	4.15	4.34	4.55	5.01
20	7.07	7.33	7.63	8.36
∞	7.77	8.02	8.32	9.04

TABLE 4.2

Values of $E\{W_n/N_0=i\}$ in M/M/m, $\rho=0.7$

Horizon	i=0				i=3			
	m=1	m=2	m=3	m=5	m=1	m=2	m=3	m=5
1	0.41	0.00	0.00	0.00	2.74	1.86	1.07	0.00
2	0.68	0.24	0.00	0.00	2.58	1.78	1.10	0.19
3	0.88	0.44	0.14	0.00	2.47	1.73	1.12	0.31
5	1.17	0.73	0.42	0.05	2.35	1.69	1.16	0.47
20	1.97	1.55	1.26	0.84	2.24	1.77	1.43	0.97
∞	2.33	1.92	1.64	1.26	2.33	1.92	1.64	1.26

Horizon	i=6				i=9			
	m=1	m=2	m=3	m=5	m=1	m=2	m=3	m=5
1	5.61	4.63	3.67	1.86	8.58	7.58	6.59	4.63
2	5.26	4.32	3.41	1.77	8.17	7.19	6.21	4.31
3	4.95	4.05	3.19	1.70	7.78	6.81	5.86	4.04
5	4.45	3.62	2.86	1.59	7.07	6.14	5.25	3.59
20	2.93	2.37	1.94	1.30	4.10	3.45	2.90	2.02
∞	2.33	1.92	1.64	1.26	2.33	1.92	1.64	1.26

Horizon	MSE(W_n)			
	m=1	m=2	m=3	m=5
1	3.64	3.00	2.56	1.96
2	4.60	3.86	3.31	2.55
3	5.35	4.57	3.94	3.05
5	6.48	5.63	4.93	3.86
20	9.41	8.44	7.59	6.22
∞	10.11	9.12	8.25	6.81

TABLE 4.3

Values of $E\{N_n/N_0=i\}$ in $E_R/M/2$, $\rho=0.7$

Horizon	i=0				i=3			
	R=1	R=2	R=3	R=5	R=1	R=2	R=3	R=5
1	0.58	0.54	0.53	0.51	2.78	2.70	2.67	2.65
2	0.92	0.84	0.80	0.78	2.65	2.51	2.45	2.40
3	1.16	1.04	0.99	0.94	2.58	2.38	2.30	2.23
5	1.50	1.29	1.22	1.15	2.52	2.23	2.12	2.02
20	2.36	1.88	1.71	1.57	2.59	2.05	1.85	1.69
∞	2.75	2.06	1.83	1.65	2.75	2.06	1.83	1.65

Horizon	i=6				i=9			
	R=1	R=2	R=3	R=5	R=1	R=2	R=3	R=5
1	5.61	5.58	5.58	5.57	8.58	8.57	8.57	8.57
2	5.28	5.20	5.18	5.17	8.18	8.15	8.15	8.14
3	4.99	4.86	4.82	4.79	7.79	7.74	7.73	7.72
5	4.53	4.29	4.21	4.14	7.10	6.97	6.93	6.91
20	3.21	2.59	2.36	2.16	4.31	3.65	3.41	3.20
∞	2.75	2.06	1.83	1.65	2.75	2.06	1.83	1.65

Horizon	MSE(N_n)			
	R=1	R=2	R=3	R=5
1	1.39	1.01	0.88	0.78
2	2.39	1.67	1.43	1.25
3	3.18	2.16	1.83	1.58
5	4.34	2.83	2.36	2.00
20	7.33	4.23	3.35	2.72
∞	8.02	4.41	3.45	2.76

TABLE 4.4

Values of $E\{W_n/N_0=i\}$ in $E_R/M/2$, $\rho=0.7$

Horizon	i=0				i=3			
	R=1	R=2	R=3	R=5	R=1	R=2	R=3	R=5
1	0.00	0.00	0.00	0.00	1.86	1.77	1.73	1.70
2	0.24	0.18	0.16	0.15	1.78	1.62	1.56	1.51
3	0.44	0.34	0.30	0.27	1.73	1.53	1.45	1.38
5	0.73	0.56	0.49	0.44	1.69	1.41	1.30	1.21
20	1.55	1.10	0.94	0.81	1.77	1.25	1.07	0.92
∞	1.92	1.26	1.06	0.89	1.92	1.26	1.06	0.89

Horizon	i=6				i=9			
	R=1	R=2	R=3	R=5	R=1	R=2	R=3	R=5
1	4.63	4.59	4.58	4.58	7.58	7.57	7.57	7.57
2	4.32	4.23	4.20	4.18	7.19	7.15	7.15	7.15
3	4.05	3.90	3.85	3.82	6.81	6.75	6.73	6.72
5	3.62	3.37	3.28	3.21	6.14	6.00	5.95	5.92
20	2.37	1.77	1.55	1.37	3.45	2.81	2.56	2.36
∞	1.92	1.26	1.06	0.89	1.92	1.26	1.05	0.89

Horizon	MSE(W_n)			
	R=1	R=2	R=3	R=5
1	3.00	1.98	1.66	1.40
2	3.86	2.53	2.10	1.76
3	4.57	2.94	2.43	2.03
5	5.63	3.53	2.89	2.39
20	8.44	4.81	3.77	3.01
∞	9.12	4.98	3.86	3.06

TABLE 4.5

Normalized Errors for Queue Length and Waiting Time Predictions
in M/M/1, ρ varying*

Horizon	$\rho=.5$	$\rho=.7$	$\rho=.9$
1	.429(.561)	.168(.360)	.020(.246)
2	.645(.727)	.291(.455)	.038(.260)
3	.770(.823)	.388(.529)	.055(.273)
5	.896(.920)	.534(.641)	.087(.297)
10	.982(.986)	.744(.803)	.156(.351)
20	.999(.999)	.910(.931)	.268(.437)

TABLE 4.6

Normalized Errors for Queue Length and Waiting Time Predictions
in M/M/2, ρ varying. $I=40$ *

Horizon	$\rho=.5$	$\rho=.7$	$\rho=.9$	
			Upper Bound	Lower Bound
1	.445(.557)	.173(.329)	.022(.103)	.222(.284)
2	.659(.717)	.298(.424)	.042(.120)	.242(.302)
3	.780(.814)	.396(.501)	.061(.137)	.261(.319)
5	.901(.915)	.541(.618)	.096(.168)	.297(.351)
10	.983(.985)	.750(.790)	.173(.237)	.376(.421)
20	.999(.999)	.914(.927)	.287(.336)	.677(.688)

* Waiting time errors are in brackets.

TABLE 4.7

Normalized Errors for Queue Length and Waiting Time Predictions
in M/M/5, ρ varying. $I=40^*$

Horizon	$\rho=.9$			
	$\rho=.5$	$\rho=.7$	Lower Bound	Upper Bound
1	.402(.497)	.173(.288)	.023(.094)	.265(.303)
2	.630(.646)	.303(.374)	.043(.111)	.286(.319)
3	.764(.748)	.405(.448)	.063(.127)	.307(.335)
5	.897(.877)	.554(.567)	.100(.156)	.344(.365)
10	.983(.975)	.763(.755)	.179(.224)	.426(.432)
20	.999(.999)	.925(.913)	.296(.318)	.770(.711)

TABLE 4.8

Normalized Errors for Queue Length and Waiting Time Predictions
in $E_R/M/2$, R varying. $\rho=0.7^*$

Horizon	$R=1$	$R=2$	$R=5$
1	.174(.329)	.228(.398)	.282(.458)
2	.298(.424)	.378(.507)	.452(.576)
3	.396(.501)	.489(.590)	.571(.662)
5	.541(.618)	.642(.710)	.724(.779)
10	.750(.790)	.836(.866)	.894(.915)
20	.914(.927)	.958(.965)	.981(.984)

* Waiting Time Errors are in brackets.

Figure 4.1: Queue Length Predictions Trajectories

 $M/M/m, \rho=0.7$

Figure 4.2: Queue Length Prediction Trajectories
 $E_R/M/1$, $\rho=0.7$

CHAPTER 5

M/G/1 PREDICTION

5.1 Introduction

Optimal mean square predictors for queues with an exponential service mechanism have been considered in Chapters 3 and 4. In this chapter, we consider OMS prediction in the M/G/1 queue. This queue is a widely used model because its assumption of Poisson arrivals accommodates more situations than an exponential service assumption does. Hence M/G/1 results would have a greater range of applicability.

Derivation of an OMS prediction is affected by two points:

(i) Since the service mechanism is not exponential, the system time is not a sum of i.i.d. exponential random variables. An arriving customer who finds i in system will have to stay for 1 residual service interval and i complete service intervals. Therefore the moments of the residual service distribution must also be found.

(ii) The imbedded Markov chain consists of queue lengths observed at departure instants rather than arrival instants. Hence the transition probabilities cannot be used for a predictor based on queue lengths at arrival instants.

In fact, no way has been found to make use of predictions from the

arrival instant queue lengths. However, if the set of departure instants is used, OMS predictors comparable to those found for GI/M/1 can be found.

The key to this approach is that the queue length left behind by a customer is also the number of arrivals during his system time.

Hence, while the system time is a simple function of the queue length upon arrival in GI/M/1, in M/G/1 the queue length upon departure is simply dependent upon the system time. A similar relationship applies to waiting time predictions.

We consider first the case of prediction of queue lengths.

5.2. Queue Length Prediction

To emphasize that we are dealing with a different queue-length process in M/G/1, we shall denote the queue length left behind when C_n departs by D_n . As usual, the one-step transition probabilities will be called for. From Gross and Harris [5], we find that

$$p_{ij} = \Pr\{D_{n+1} = j | D_n = i\}$$

$$= \begin{cases} k_j & i=0 \\ k_{j-i+1} & i \geq 1; j \geq i-1 \\ 0 & \text{elsewhere} \end{cases} \quad (5.1)$$

for k_l defined by

$$k_l = \int_0^{\infty} e^{-\lambda t} \frac{(\lambda t)^l}{l!} dF_X(t) \quad (5.2)$$

The queue length predictor is now found in our usual way. We have the following summations:

$$E\{D_n/D_0=i\} = \sum_{\ell=i-1}^{\infty} p_{i\ell} E\{D_{n-1}/D_0=\ell\} \quad (5.3)$$

$$E\{D_n^2/D_0=i\} = \sum_{\ell=i-1}^{\infty} p_{i\ell} E\{D_{n-1}^2/D_0=\ell\} \quad (5.4)$$

The conditional and MSE are

$$\text{Var}\{D_n/D_0=i\} = E\{D_n^2/D_0=i\} - E\{D_n/D_0=i\}^2 \quad (5.5)$$

$$\text{MSE}(D_n) = \sum_{i=0}^{\infty} \pi_i \text{Var}\{D_n/D_0=i\} \quad (5.6)$$

where

$$\pi_i = \Pr\{D_0=i\} \quad \forall i \quad (5.7)$$

It would seem from (5.3) and (5.4) that not only are bounds required for the mean squared error, but also (due to the infinite range of summation) bounds are needed to approximate the conditional moments. This matter will be considered after the relationship between the queue length and system and waiting time predictors have been discussed.

5.3 System Time Prediction

We stated earlier that D_n is a function of S_n . Clearly S_n is affected by earlier queue lengths, and in particular D_0 (when $D_0 < n$). These facts will lead to the equations for the desired predictors. Let us define

$$S_i^n(t) = \Pr\{S_n \leq t / D_0 = i\} \quad (5.8)$$

$$D_i^n(z) = \sum_{j=0}^{\infty} p_{ij}^n z^j \quad (5.9)$$

$$S_i^h(s) = \int_0^{\infty} e^{-st} dS_i^n(t) \quad (5.10)$$

Assume $i < n$. Since the arrivals are from a Poisson process at rate λ , and independent of all earlier queue lengths.

$$E\{D_n / D_0 = i; S_n = t\} = \lambda t$$

Unconditioning, one obtains

$$E\{D_n / D_0 = i\} = \int_0^{\infty} \lambda t dS_i^n(t) = \lambda E\{S_n / D_0 = i\}$$

or equivalently

$$E\{S_n / D_0 = i\} = \left(\frac{1}{\lambda}\right) E\{D_n / D_0 = i\} \quad i < n \quad (5.11)$$

Higher moments can be generated in a variety of ways; the following method makes use of a transform approach.

$$\begin{aligned} p_{ij}^n &= \Pr\{D_n = j / D_0 = i\} \\ &= \int_0^{\infty} e^{-\lambda t} \frac{(\lambda t)^j}{j!} dS_i^n(t) \quad \text{for } i < n \end{aligned} \quad (5.12)$$

Therefore

$$D_i^n(z) = \sum_{j=0}^{\infty} p_{ij}^n z^j = \sum_{j=0}^{\infty} \int_0^{\infty} e^{-\lambda t} \frac{(\lambda t)^j}{j!} z^j dS_i^n(t)$$

$$= \int_0^{\infty} e^{-\lambda t(1-z)} dS_i^n(t)$$

$$= S_i^n(s) |_{s=\lambda(1-z)} \quad i < n$$

The transform equation is the conditional analogue to (2.21). By taking the appropriate derivatives w.r.t. z , one obtains (5.11). In a similar manner it can be shown that

$$E\{S_n^2/D_0=i\} = \left(\frac{1}{\lambda}\right)^2 [E\{D_n^2/D_0=i\} - E\{D_n/D_0=i\}], \quad i < n$$

(5.14)

$$\text{Var}\{S_n/D_0=i\} = \left(\frac{1}{\lambda}\right)^2 \text{Var}\{D_n/D_0=i\} - \left(\frac{1}{\lambda}\right)^2 E\{D_n/D_0=i\} \quad i < n$$

(5.15)

For $i \geq n$ a different approach is needed because D_0 is a queue length occurring during S_n (see Figure 5.1). Define $S_n = S_n^O + S_n^R$, where S_n^O is the time between C_n arriving and C_0 departing, and S_n^R is the time between C_0 departing and C_n departing. Since successive service times are independent, S_n^O and S_n^R are independent. In fact, the latter consists of the n i.i.d. service times $X_1, \dots, X_n$. Hence for $i \geq n$

$$E\{S_n^R/D_0=i\} = n\bar{x} \quad (5.16)$$

$$\text{Var}\{S_n^R/D_0=i\} = n\sigma_x^2 \quad (5.17)$$

Similarly, $E\{S_n^0/D_0=i\}$ can be found in a straightforward manner when $i \geq n$.

$$\begin{aligned} E\{S_n^0/D_0=i\} &= \int_0^\infty t d\Pr\{S_n^0 \leq t/D_0=i\} \\ &= \frac{1}{\pi_i} \int_0^\infty t d\Pr\{S_n^0 \leq t; D_0=i\} \\ &= \frac{1}{\pi_i} \int_0^\infty t \Pr\{D_0=i/S_n^0=t\} d\Pr\{S_n^0 \leq t\} \quad i \geq n \end{aligned}$$

Since C_n has arrived before C_0 departs, $D_0=i$ if and only if $(i-n)$ arrivals occur during t . Therefore

$$\begin{aligned} E\{S_n^0/D_0=i\} &= \frac{1}{\pi_i} \int_0^\infty t e^{-\lambda t} \frac{(\lambda t)^{i-n}}{(i-n)!} d\Pr\{S_n^0 \leq t\} \\ &= \frac{1}{\pi_i} (i-n+1) \left(\frac{1}{\lambda}\right) \int_0^\infty e^{-\lambda t} \frac{(\lambda t)^{i+1-n}}{(i+1-n)!} d\Pr\{S_n^0 \leq t\} \\ &= (i-n+1) \left(\frac{1}{\lambda}\right) \pi_{i+1}/\pi_i \quad i \geq n \end{aligned} \quad (5.18)$$

In like manner it can be shown that

$$E\{S_n^{02}/D_0=i\} = (i-n+1)(i-n+2) \left(\frac{1}{\lambda}\right)^2 \pi_{i+2}/\pi_i \quad i \geq n \quad (5.19)$$

Combining (5.18) and (5.19) leads to

$$\begin{aligned} \text{Var}\{S_n^0/D_0=i\} &= (i-n+1)(i-n+2) \left(\frac{1}{\lambda}\right)^2 \pi_{i+2}/\pi_i \\ &\quad - \left[(i-n+1) \left(\frac{1}{\lambda}\right) \pi_{i+1}/\pi_i \right]^2 \quad i \geq n \end{aligned} \quad (5.20)$$

Due to the independence of S_n^0 and S_n^R , (5.16) and (5.18) jointly imply

$$E\{S_n/D_0=i\} = (i-n+1)\left(\frac{1}{\lambda}\right)(\pi_{i+1}/\pi_i) + n\bar{x} \quad i \geq n \quad (5.21)$$

while (5.17) and (5.20) imply

$$\begin{aligned} \text{Var}\{S_n/D_0=i\} &= (i-n+1)(i-n+2)\left(\frac{1}{\lambda}\right)^2 \pi_{i+2}/\pi_i \\ &\quad - \left[(i-n+1)\left(\frac{1}{\lambda}\right)\pi_{i+1}/\pi_i\right]^2 + n\sigma_x^2 \quad i \geq n \quad (5.22) \end{aligned}$$

The mean squared prediction error is then

$$\text{MSE}(S_n) = \sum_{i=0}^{\infty} \text{Var}\{S_n/D_0=i\} \pi_i \quad (5.23)$$

for which bounds can be generated. Equations (5.21) and (5.22) are useful both for mathematical completeness and for computing $\text{MSE}(S_n)$. However, the typical prediction problem would deal with future arrivals to the system and hence would use (5.11) and (5.15).

5.4 Waiting Time Prediction

Consider an arrival C_n to a busy queue. Since C_n enters service when C_{n-1} departs, the number of arrivals during C_n 's waiting time is $(D_{n-1} - 1)$. Using an approach similar to that for (5.13), one obtains

$$E\{W_n/D_0=i\} = \left(\frac{1}{\lambda}\right)[E\{D_{n-1}/D_0=i\} - (1-p_{i0}^{n-1})] \quad i < n-1$$

(5.24)

$$E\{W_n^2/D_0=i\} = \left(\frac{1}{\lambda}\right)^2 [E\{D_{n-1}^2/D_0=i\} - 3E\{D_{n-1}/D_0=i\} + 2(1-p_{i0}^{n-1})] \quad i < n-1 \quad (5.25)$$

Expressions can be found for the case where $i \geq n-1$, but would only be of practical interest in order to compute $MSE(W_n)$.

5.5 Efficient Computation of the Queue Length Predictions

We now consider the problem in computing the queue length predictors posed by the infinite sums in equations (5.3), (5.4) and (5.6). Results will be derived which allow us to replace the infinite sums by equivalent finite sums, and which lead to a computationally tractable queue length predictor.

First let us consider for $i \geq n$

$$\begin{aligned} p_{ij}^n &= \Pr\{D_n=j/D_0=i\} \\ &= \Pr\{(n+j-i) \text{ arrivals during } n \text{ service times}\} \\ &= \begin{cases} \int_0^\infty e^{-\lambda t} \frac{(\lambda t)^{n+j-i}}{(n+j-i)!} dF_X^{(n)}(t) & j \geq i-n \\ 0 & j < i-n \end{cases} \quad (5.26) \end{aligned}$$

Then we have

$$D_i^n(z) = \sum_{j=0}^{\infty} p_{ij}^n z^j$$

$$= z^{i-n} (\phi_X(s)^n) \Big|_{s=\lambda(1-z)}; \quad i \geq n \quad (5.27)$$

Taking derivatives of both sides of (5.27) we get

$$\begin{aligned} D_i^{n'}(z) &= (i-n) z^{i-n-1} (\phi_X(s)^n) \Big|_{s=\lambda(1-z)} \\ &\quad - \lambda z^{i-n} [n \phi_X(s)^{n-1} \phi_X'(s)] \Big|_{s=\lambda(1-z)} \quad i \geq n \end{aligned} \quad (5.28)$$

Evaluating this expression at $z=1$ we get

$$E\{D_n/D_0=i\} = (i-n) + \lambda n \bar{x} = (i-n) + np \quad i \geq n \quad (5.29)$$

Taking second derivatives produces

$$\begin{aligned} D_i^{n''}(z) &= (i-n)(i-n-1) z^{i-n-2} (\phi_X(s)^n) \Big|_{s=\lambda(1-z)} \\ &\quad - 2\lambda(1-n) z^{i-n-1} [n \phi_X(s)^{n-1} \phi_X'(s)] \Big|_{s=\lambda(1-z)} \\ &\quad - \lambda z^{i-n} [n \phi_X(s)^{n-1} \phi_X''(s) (-\lambda) + n(n-1) \phi_X(s)^{n-2} (\phi_X'(s))^2 (-\lambda)] \Big|_{s=\lambda(1-z)} \end{aligned} \quad i \geq n \quad (5.30)$$

When (5.30) is evaluated at $z=1$ we obtain

$$\begin{aligned} E\{D_n^2/D_0=i\} - E\{D_n/D_0=i\} &= (i-n)(i-n-1) \\ &\quad + 2\lambda(i-n)n\bar{x} + \lambda^2 [n\bar{x}^2 + n(n-1)\bar{x}^2] \quad i \geq n \end{aligned} \quad (5.31)$$

Adding equations (5.29) and (5.31) we obtain

$$E\{D_n^2/D_0=i\} = (i-n)^2 + n[(2(i-n)+1)\rho + (n-1)\rho^2 + \lambda^2 \overline{x^2}] \quad i \geq n \quad (5.32)$$

Therefore, using (5.29), we can replace (5.3) by

$$E\{D_n/D_0=i\} = \begin{cases} \sum_{\ell=i-1}^{\infty} p_{i\ell} E\{D_{n-1}/D_0=\ell\} & i < n \\ (i-n) + n\rho & i \geq n \end{cases} \quad (5.33)$$

The former summation can be simplified as well. For $\ell \geq n-1$ we have

$$E\{D_{n-1}/D_0=\ell\} = (\ell - n + 1) + (n-1)\rho$$

hence

$$\begin{aligned} E\{D_n/D_0=i\} &= \sum_{\ell=i-1}^{n-2} p_{i\ell} E\{D_{n-1}/D_0=\ell\} + \sum_{\ell=n-1}^{\infty} \ell p_{i\ell} - (n-1)(1-\rho) \sum_{\ell=n-1}^{\infty} p_{i\ell} \\ &= (i-n) + n\rho + \sum_{\ell=i-1}^{n-2} p_{i\ell} [E\{D_{n-1}/D_0=\ell\} - \ell + (n-1)(1-\rho)] \quad i < n \end{aligned}$$

Therefore

$$E\{D_n/D_0=i\} = \begin{cases} (i-n) + n\rho + \sum_{\ell=i-1}^{n-2} p_{i\ell} [E\{D_{n-1}/D_0=\ell\} - \ell + (n-1)(1-\rho)] & 0 < i < n \\ (i-n) + n\rho & i \geq n \end{cases} \quad (5.34)$$

Thus we have replaced the infinite summation of equation (5.3) by the finite sum of equation (5.34). By the same technique, and using (5.32), we can replace (5.4) by

$$E\{D_n^2/D_0=i\} = \begin{cases} (i-n)^2 + n[(2(i-n)+1)\rho + (n-1)\rho^2 + \lambda^2 \overline{x^2}] \\ + \sum_{\ell=i-1}^{n-2} p_{i\ell} [E\{D_{n-1}^2/D_0=\ell\} - (n-1-\ell)^2 \\ + (n-1)\{[2(n-1-\ell)-1]\rho - (n-2)\rho^2 - \lambda^2 \overline{x^2}\}] & i < n \\ (i-n)^2 + n[(2(i-n)+1)\rho + (n-1)\rho^2 + \lambda^2 \overline{x^2}] & i \geq n \end{cases} \quad (5.35)$$

The mean squared error need not involve an infinite sum either. For $i \geq n$ we have

$$\begin{aligned} \text{Var}\{D_n/D_0=i\} &= E\{D_n^2/D_0=i\} - E\{D_n/D_0=i\}^2 \\ &= (i-n)^2 + n[(2(i-n)+1)\rho + (n-1)\rho^2 + \lambda^2 \overline{x^2}] - ((i-n) + n\rho)^2 \\ &= n(\rho + \lambda^2 \sigma_x^2) \quad i \geq n \end{aligned} \quad (5.36)$$

Substituting (5.36) into (5.6) yields

$$\begin{aligned} \text{MSE}(D_n) &= \sum_{i=0}^{n-1} \pi_i \text{Var}\{D_n/D_0=i\} + \sum_{i=n}^{\infty} \pi_i n(\rho + \lambda^2 \sigma_x^2) \\ &= \sum_{i=0}^{n-1} \pi_i \text{Var}\{D_n/D_0=i\} + n(\rho + \lambda^2 \sigma_x^2) \left(1 - \sum_{i=0}^{n-1} \pi_i\right) \end{aligned} \quad (5.37)$$

5.6 Numerical Examples

A variety of numerical examples were computed to study the effects of various service distributions upon the predictions and the predictive power. In addition comparisons were made in the case of the M/M/1 queue between the results for the departure instant approach of this chapter and those for the arrival instant approach of Chapter 3.

Table 5.1 and Figure 5.2 present the OMS queue length prediction trajectories for several M/G/1 queues with traffic intensity $\rho=0.7$. Based on (5.34), we see that the expected one-step displacement for all $i>0$ is negative and equal to $-(1-\rho)$. This is shown separately in Gross and Harris [5], p. 238.

In fact, for $n \leq i$, the prediction trajectories are parallel straight lines with common slope $-(1-\rho)$. At the same time $p_{i0}^n = 0$ for $n < i$. These facts lend further credence to the heuristic (4.29) developed in chapter 4. The additional factor $(\frac{1}{\rho})$ in (4.29) reflects the fact that in a busy queue, transitions of the $\{D_n\}$ process occur more quickly than those of the $\{N_n\}$ process.

For $n > i$, the trajectories diverge, as can best be seen in Figure 5.2. This clearly demonstrates that as the variability of the service process C_b^2 drops from 1.0 in M/M/1 to 0.5 in M/E₂/1 and then to 0.0 in M/D/1, the predicted value drops as well. As the prediction horizon goes to infinity, the predicted values approach their respective mean queue lengths, which have an explicit dependence upon the service process variability (see

Kleinrock [6] or Gross and Harris [5]):

$$E\{D\} = \rho + \rho^2 \frac{(1 + C_b^2)}{2(1 - \rho)} \quad (5.38)$$

Although it appears from Table 5.1 that the trajectories of the M/D/1 queue are monotonic, this is not the case. The trajectory for $i=2$ (not shown) produces an overshoot for $\rho=0.7$.

Table 5.2 and Figure 5.3 deal with system time predictions in M/M/1 and M/D/1 with $\rho=0.7$. In both cases, there is a jump discontinuity in the trajectories at $n=i$. As Figure 5.3 shows, the discontinuity at $n=i$ is a drop of one mean service time. For $n>i$, the trajectories are scaled versions of the queue length trajectories, and so they will display the same characteristics.

The case $n \leq i$ is noteworthy. Recall that

$$E\{S_n/D_0=i\} = (i-n+1)\left(\frac{1}{\lambda}\right)\pi_{i+1}/\pi_i + n\bar{x}; \quad n \leq i \quad (5.21)$$

In M/M/1 queues $\pi_{i+1}/\pi_i = \rho$, and so

$$E\{S_n/D_0=i\} = (i+1)\bar{x} \quad n \leq i \quad (5.39)$$

Hence, a departing customer from an M/M/1 queue "expects" all customers behind him to have equal total system times.

The corresponding M/D/1 trajectories are linear. By rewriting (5.21)

this slope can be seen to be $(\frac{1}{\lambda})(\rho - \pi_{i+1}/\pi_i)$. Furthermore, if both the M/D/1 and M/M/1 trajectories for $n \leq i$ were carried to $n = i+1$, they would intersect, as can be seen from (5.21).

Tables 5.3 and 5.4 deal with the predictive power of the queue length and system time predictions. Table 5.3 lists values of the normalized mean squared error of the queue length process for M/D/1 and M/M/1 queues. Not surprisingly, the errors increase as a function of the prediction horizon, and decrease with the traffic intensity. Furthermore, the normalized errors for the M/D/1 predictions are larger than those of the M/M/1 process, reflecting the reduced correlation between successive queue lengths.

It is possible to compare the results for prediction based on queue lengths at arrival instants which were developed in Chapter 3 with those for the predictors using the departure instant approach of this chapter by applying both methods to the analysis of an M/M/1 queue. The best measure of effectiveness in this comparison is the predictive power. Tables 3.2 and 5.3 contain the pertinent normalized mean squared errors for the queue length predictions, over a variety of traffic intensities. The errors for the latter method are all smaller than those for the former, sometimes by a factor of 10% or more when the traffic intensity is moderate and when the prediction horizon is small. A similar comparison of Tables 3.3 and 5.4 shows that the same is true for system time predictions, but to a greater degree. For example the normalized error using the method of this chapter, when $n=2$ and $\rho=0.7$, is almost 40% smaller than the equivalent value using the earlier method.

The reason for this behaviour is that while N_0 is always "a priori" information for S_n , D_0 is not necessarily so (this was discussed in the development of the system time predictor). As a result, the "a posteriori" information which D_0 may contain reduces the mean squared error and offers an improved estimate.

However, any practical prediction problem would make use of D_0 only to predict the time of customers who have not arrived. A much more practical estimate of the total time in system of customers who have already arrived is their elapsed time plus the mean residual time in system, which is easily found at departure instants.

5.7 Conclusions

In this chapter, efficient predictors for queue lengths and system times based on queue lengths at departure instants have been derived. These predictors were tested for a variety of service distributions and comparisons were drawn between the arrival and departure instant approaches to prediction.

TABLE 5.1

Queue Length Predictions, $\rho=0.7$

$\mu_N = 2.33$ $E\{N_n/N_0=i\}$					$\mu_N = 1.52$ $E\{N_n/N_0=i\}$				
Horizon	i=1	i=4	i=7	i=10	i=1	i=4	i=7	i=10	
1	0.70	3.70	6.70	9.70	0.70	3.70	6.70	9.70	
3	1.18	3.10	6.10	9.10	1.02	3.10	6.10	9.10	
5	1.43	2.62	5.50	8.50	1.16	2.56	5.50	8.50	
10	1.77	2.32	4.17	7.00	1.34	1.95	4.07	7.00	
20	2.06	2.27	3.05	4.68	1.45	1.63	2.48	4.36	
30	2.18	2.28	2.67	3.57	1.49	1.56	1.90	2.89	
$\sigma_N^2 = 7.77$ $\text{Var}\{N_n/N_0=i\}$					$\sigma_N^2 = 2.56$ $\text{Var}\{N_n/N_0=i\}$				
Horizon	i=1	i=4	i=7	i=10	i=1	i=4	i=7	i=10	
1	0.70	0.70	0.70	0.70	0.70	0.70	0.70	0.70	
3	2.27	2.10	2.10	2.10	1.15	2.10	2.10	2.10	
5	3.02	5.38	3.50	3.50	1.44	3.22	3.50	3.50	
10	4.32	6.38	10.55	11.90	1.88	3.48	6.44	7.00	
20	5.78	6.93	10.86	17.02	2.28	2.98	6.00	10.69	
30	6.57	7.23	9.76	15.10	2.44	2.74	4.35	8.44	

TABLE 5.2

System Time Predictions

M/M/1 $\rho=0.7$ $\mu_S=3.33$

Horizon	i=1	i=4	i=7	i=10
1	2.00	5.00	8.00	11.00
2	1.41	5.00	8.00	11.00
3	1.68	5.00	8.00	11.00
5	2.04	3.74	8.00	11.00
10	2.53	3.31	5.95	11.00
20	2.94	3.24	4.35	6.69
30	3.12	3.26	3.82	5.10

M/D/1 $\rho=0.7$ $\mu_S=2.17$

Horizon	i=1	i=4	i=7	i=10
1	1.89	3.96	6.09	8.27
2	1.28	4.18	6.36	8.54
3	1.45	4.45	6.64	8.82
5	1.66	3.66	7.18	9.36
10	1.91	2.79	5.81	10.73
20	2.08	2.34	3.54	6.22
*30	2.13	2.23	2.72	4.08

TABLE 5.3

Normalized Errors for Queue Length Predictions

 ρ varying

Horizon	M/D/1			M/M/1		
	$\rho=.5$	$\rho=.7$	$\rho=.9$	$\rho=.5$	$\rho=.7$	$\rho=.9$
1	.698	.410	.069	.375	.153	.019
2	.876	.596	.122	.584	.268	.037
3	.923	.680	.163	.711	.360	.053
5	.966	.765	.222	.855	.500	.084
10	.997	.893	.311	.969	.712	.151
20	1.000	.983	.432	.998	.891	.261

TABLE 5.4

Normalized Errors for System Time Predictions

M/M/1 - ρ varying

Horizon	$\rho=.5$	$\rho=.6$	$\rho=.7$	$\rho=.9$
1	.500	.400	.300	.100
2	.542	.424	.311	.100
3	.612	.471	.336	.102
5	.757	.589	.409	.107
10	.939	.818	.610	.131
20	.996	.966	.845	.205

Figure 5.1: The Relationship Between D_0 and S_n Case 1: $i < n$ Case 2: $i \geq n$

Figure 5.2: M/G/1 Queue Length Prediction Trajectories

 $\rho=0.7$

2 2
OF / DE

OF / DE

2

Figure 5.3: M/G/1 System Time Prediction Trajectories

 $\rho=0.7 \quad \bar{x}=1.0$

CHAPTER 6

SYSTEM TIMES IN TANDEM M/M/1 QUEUES

In this chapter, the time spent by C_n in a tandem series of stationary s.s. queues $Q_1, \dots, Q_k$, will be considered. Service at Q_ℓ is exponential at rate μ_ℓ ($\ell=1, \dots, k$), and the arrivals to Q_1 are Poisson at rate λ , where $\min(\mu_1, \dots, \mu_k) > \lambda$ to ensure stationarity.

The analysis of tandem M/M/1 queues may act as a first step towards using prediction for adaptive routing in general data networks. Tractable results can be found for the former, leading to heuristic and approximations for the latter.

The first major work in tandem queues was by Reich [19] in 1957, who showed that the system times at successive M/M/1 queues in tandem are mutually independent. Since system times in M/M/1 queues are exponentially distributed, the time C_n spends in the series of queues is the sum of K independent exponential random variables. An important element in the tractability of the system time results is that overtaking is not possible, and hence a customer's past cannot affect his future. Melamed [9] has shown that when overtaking is possible, the total time in system cannot be expressed as the sum of exponential system times at each queue. Hence tandem M/M/1 queues provide a greater likelihood for useful prediction

results than even Jackson networks.

The bulk of the chapter deals with the OMS prediction for the time spent in the sequence of queues conditioned upon the queue lengths at each queue at the time of C_n 's arrival to Q_1 . The analysis starts with a proof of Reich's results, both as an introduction to the analysis of tandem queues and to describe the independence of the time spent at Q_i from earlier occurrences at queues downstream. Hence only the first i queue lengths observed can have an effect upon the time C_n spends at Q_i . After Reich's analysis, results for two M/M/1 queues in tandem are derived. The difficulties in extending this approach to a series of three or more such queues are then discussed, along with two other methods of solution.

6.1 Reich's Analysis of Tandem M/M/1 Queues

Reich [19] used the reversibility of the queue length process $n(t)$ of a single stationary M/M/m queue to prove the following:

Theorem 6.1: (a) The sequence of departure times forms a Poisson process;
 (b) The value of $n(t)$ is independent of all past departure times;
 (c) If t_0 is a departure time, then $n(t_0+0)$ is independent of all past departure times.

He then used Theorem 6.1 to prove

Theorem 6.2: If $m=1$, (i.e. a s.s. queue) then the times T_1 spent at Q_1 and T_2 spent at Q_2 are independent.

Since the proof of the latter theorem is pertinent for later prediction results, it is repeated here.

Proof of Theorem 6.2: Let n_1 and n_2 be the queue lengths at Q_1 and Q_2 respectively when C_n arrives at Q_2 . Hence n_1 is the queue length immediately after C_n leaves Q_1 . From part (c) of Theorem 6.1, n_1 and n_2 are independent. Define

$$A(t; k) = \Pr\{T_1 \leq t / n_2 = k\}$$

Recalling that n_1 is also the number of arrivals to Q_1 during T_1 , we find

$$\Pr\{n_1 = j / n_2 = k\} = \int_0^{\infty} e^{-\lambda t} \frac{(\lambda t)^j}{j!} dA(t; k) \quad (6.1)$$

Since the left hand side of (6.1) is independent of k , it follows that $A(t; k)$ does not depend on k . Hence T_1 is independent of n_2 , and consequently T_2 (consisting of $(n_2 + 1)$ i.i.d. exponential random variables) is independent of T_1 . This completes the proof of the theorem.

Since T_1 is independent of n_2 , it is also independent of any previous queue length at Q_2 . By extending this to a series of k tandem queues it is clear that the time spent at a particular queue in the series Q_i is only affected by the queue lengths C_n observes at $Q_1, \dots, Q_i$ upon arrival to Q_i .

Theorem 6.2 shows that the time spent in k tandem M/M/1 queues, with no queue length conditions, is thus the sum of k independent exponential random variables. We now consider the question of predicting S_n based upon these conditions.

6.2 System Times in Two Tandem M/M/1 Queues

Define

$S_{i,n}$ = C_n 's system time at Q_i

$N_{i,n}$ = Queue length seen by C_n upon arrival to Q_i

$N_{i,n}^j$ = Queue length at Q_i when C_n arrives at Q_j

Clearly

$$S_n = \sum_{i=1}^k S_{i,n}$$

Since $S_{i,n}$ is independent of $N_{\ell,n}^1; \ell=i+1, \dots, k$, the OMS predictor takes the form

$$E\{S_n / N_{1,n}^1 = i_1, \dots, N_{k,n}^1 = i_k\} = \sum_{\ell=1}^k E\{S_{\ell,n} / N_{1,n}^1 = i_1, \dots, N_{\ell,n}^1 = i_{\ell}\} \quad (6.2)$$

The queue lengths $N_{1,n}^1, \dots, N_{k,n}^1$ can be viewed as initial conditions to the system. Hence initially the internodal flows are not necessarily Poisson at rate λ and hence the expectations in the sum cannot be simplified further.

Let us consider the case $k=2$. The OMS predictor for $S_n = S_{1,n} + S_{2,n}$ given $N_{1,n}^1$ and $N_{2,n}^1$ is

$$E\{S_n / N_{1,n}^1 = i; N_{2,n}^1 = \ell\} = E\{S_{1,n} / N_{1,n}^1 = i\} + E\{S_{2,n} / N_{1,n}^1 = i; N_{2,n}^1 = \ell\}$$

The first expression is clearly

$$E\{S_{1,n}/N_{1,n}=1\} = (i+1)\left(\frac{1}{\mu_1}\right) \quad (6.3)$$

Note that at least until C_n reaches Q_2 , C_1 will remain busy and hence the interdeparture times from Q_1 (which are the interarrival times to Q_2) will be exponential at rate μ_1 . So during the period of interest, Q_2 will behave as an M/M/1 queue with arrival rate μ_1 and service rate μ_2 , and C_n will be the $(i+1)$ th next arrival.

Therefore, C_{n-i-1} was the last arrival to Q_2 prior to C_n 's arrival to Q_1 . If $N_{2,n}^1 > 0$, C_{n-i-1} is still at Q_2 , and due to the memoryless property of the IAT at Q_2 we can assume that C_{n-i-1} arrived at Q_2 the instant before C_n reached Q_1 . Otherwise C_{n-i-1} has left and C_{n-1} will find Q_2 empty. Hence

$$\Pr\{N_{2,n}=j/N_{1,n}=i; N_{2,n}^1=l\} = \begin{cases} \Pr\{N_{2,n}=j/N_{2,n-i-1}=l-1\} & l > 0 \\ \Pr\{N_{2,n}=j/N_{2,n-i}=0\} & l=0 \end{cases} \quad (6.4)$$

Therefore for $l > 0$

$$E\{S_{2,n}/N_{1,n}=i; N_{2,n}^1=l\} = E\{S_{2,n}/N_{2,n-i-1}=l-1\}$$

$$= \sum_{m=0}^l q_{l-1,m} E\{S_{2,n}/N_{2,n-i}=m\} \quad (6.5)$$

while for $l=0$

$$E\{S_{2,n}/N_{1,n}=i; N_{2,n}^1=0\} = E\{S_{2,n}/N_{2,n-i}=0\}$$

$$= \sum_{m=0}^{\infty} q_{0,m} E\{S_{2,n}/N_{2,n-i+1}=m\} \quad (6.6)$$

Recalling that the arrival rate to Q_2 during the period of interest is μ_1 , the probabilities q_{ij} are given by

$$q_{ij} = \begin{cases} (1-p)^{i+1} & j=0 \\ p(1-p)^{i+1-j} & 0 < j \leq i+1 \end{cases} \quad (6.7)$$

$$\text{where } p = \left(\frac{\mu_1}{\mu_1 + \mu_2} \right)$$

The expectation can be calculated recursively starting with

$$E\{S_{2,n}/N_{2,n}=m\} = (m+1) \left(\frac{1}{\mu_2} \right) \quad (6.8)$$

Note that the recursion of (6.5) bears a strong resemblance to the GI/M/1 predictors of chapter 3:

In a similar manner, higher moments can be generated for the time C_n spends at Q_2 , and therefore the variance of the time C_n spends in the two queues can be obtained.

However, the methods of this section cannot be extended to 3 queues in tandem, as we shall see in the next section.

6.3 The Extension to Three or More Tandem Queues

Consider a series of tandem M/M/1 queues. As mentioned in the

preceding section, the quantity of interest is

$$\begin{aligned}
 & E\{S_n/N_{1,n}=i_1, \dots, N_{k,n}=i_k\} \\
 &= \sum_{\ell=1}^k E\{S_{\ell,n}/N_{1,n}=i_1, \dots, N_{\ell,n}=i_{\ell}\} \quad (6.9)
 \end{aligned}$$

The first two expectations in the preceding sum can be obtained as before. However, these methods cannot extend to three or more queues, as can be seen below.

Consider the case of $k=3$. The remaining expectation to be found is

$$\begin{aligned}
 & E\{S_{3,n}/N_{1,n}=l_1; N_{2,n}=l_2; N_{3,n}=l_3\} \\
 &= \sum_{m=0}^{\infty} \Pr\{N_{3,n}=m/N_{1,n}=l_1; N_{2,n}=l_2; N_{3,n}=l_3\} (m+1) \left(\frac{1}{\mu_3}\right) \quad (6.10)
 \end{aligned}$$

Hence the calculation of the expectation depends upon the tractability of an algorithm to find the conditional probability (or an equivalent strategy). Several approaches follow, along with the reasons why they fail to produce results.

The first approach consists of conditioning upon $N_{2,n}$ and $N_{3,n}^2$. The benefit of this method is that once these queue lengths are known the conditional expectation reduces to $E\{S_{3,n}/N_{2,n}=j_2; N_{3,n}^2=j_3\}$, which can be found using the methods for the second of two queues in tandem. Hence we seek

$$\begin{aligned}
& E\{S_{3,n}/N_{1,n}=\ell_1; N_{2,n}=\ell_2; N_{3,n}=\ell_3\} \\
&= \sum_{j_2=0}^{\infty} \sum_{j_3=0}^{\infty} E\{S_{3,n}/N_{2,n}=j_2; N_{3,n}^2=j_3\} \times \Pr\{N_{2,n}=j_2/N_{1,n}=\ell_1; N_{2,n}=\ell_2\} \\
&\quad \times \Pr\{N_{3,n}^2=j_3/N_{1,n}=\ell_1; N_{2,n}=\ell_2; N_{3,n}=\ell_3; N_{2,n}=j_2\}
\end{aligned}$$

The first probability is given by (6.4); however, the latter one cannot be easily obtained. Whereas arrivals to Q_2 , based on the initial conditions, are Poisson at rate μ_1 at least until C_n arrives, the conditional arrival process to Q_3 is not Poisson at a known rate. Indeed, if Q_2 empties, the arrival process to Q_3 is stopped until the next arrival to Q_2 occurs. Furthermore, this behaviour may happen several times prior to C_n reaching Q_2 .

A related approach would be to imbed the calculations for completions as successive queues. During one service at Q_1 , several Q_2 completions occur; during each Q_2 service, several Q_3 completions occur. However, the conditioning upon the length of each such service interval quickly makes this option intractable.

Since Q_2 can enter an idle state only at departure epochs, another alternative would be to inspect Q_2 at departure epochs (and simultaneously Q_3 for which these are arrival epochs) until C_n departs from Q_2 . This method suffers from the need to know the arrival process to Q_2 after C_n 's arrival there in order to calculate $S_{2,n}$; this process is not easily obtained due to the initial condition $N_{1,n}=\ell_1$.

Finally, we consider the certainty equivalence approach. In this situation, each exponential service mechanism is replaced by a deterministic server with the same mean, and no services underway. The problem now becomes one of scheduling as no variations occur. However, this method can at best be described as a crude approximation.

In the next section, two finite Markov chains are presented, both of which lead to tractable algorithms for predicting the time in a sequence of three or more tandem queues.

6.4 Finite Markov Chain Approaches for $E\{S_n/N_1, n, \dots, N_k, n\}$

In the preceding section it was shown that there is no easy extension of our prediction results for two M/M/1 queues in tandem to a more general series of k queues, since intermediate queues may empty prior to the arrival of C_n . However, two alternate approaches which view the behaviour of the queues as finite Markov chains with a single absorbing state produce results. In each, the conditional expectation of C_n 's time in system can be regarded as the time to absorption of a Markov chain conditioned upon an initial state. The first approach is an imbedded discrete parameter Markov chain, while the latter is a continuous parameter Markov chain. The former will be presented in detail, while the latter will be considered primarily for comparative purposes.

6.4.1 Imbedded Markov Chain Model

The behaviour of the series of tandem queues is modelled as a discrete parameter Markov chain imbedded at service completion epochs. The state

description consists of a vector $\underline{n}(\ell) = (n_1(\ell), n_2(\ell), \dots, n_k(\ell))$ indicating the number of customers $n_i(\ell)$ at Q_i immediately following the ℓ th service completion from any of the k queues. Arrivals to Q_1 following C_n have no effect upon S_n , and are assumed to stop once he has arrived. Hence all states are transient except for $(0, \dots, 0)$ which is an absorbing state; and $E\{S_n / N_{1,n} = \ell_1, \dots, N_{k,n} = \ell_k\}$ is then the expected time to absorption from the state $(\ell_1 + 1, \ell_2, \dots, \ell_k)$.

Parzen [17], pp 238-241 considers the mean time to absorption of a discrete parameter Markov chain. The following work parallels this treatment for the case of an imbedded Markov chain.

Define

m_j = mean time to absorption from state j

t_j = mean time to next transition from state j .

r_{jh} = $\Pr\{\underline{n}(\ell+1) = h / \underline{n}(\ell) = j\}$

T = set of transient states

R = $\{r_{jh} : j, h \in T\}$

τ = $\{t_j : j \in T\}$

M = $\{m_j : j \in T\}$

Upon arrival to Q_1 , C_n observes state j , and undergoes a state-dependent amount of time until the next transition (service completion), at which time he is either absorbed, or (with probability r_{jh}) has entered a new transient state h . Hence taking expectations we obtain

$$m_j = t_j + \sum_{h \in T} r_{jh} m_h \quad j \in T \quad (6.12)$$

or equivalently

$$IM = \tau + RM$$

which may be rewritten as

$$(I - R)M = \tau \quad (6.13)$$

The matrix $(I-R)$ possesses an inverse given by

$$(I - R)^{-1} = I + R + R^2 + \dots$$

which can be shown to converge. Hence the mean absorption times are

$$M = (I - R)^{-1}\tau \quad (6.14)$$

Therefore in order to obtain the mean time to absorption all that is required is the mean times to the next transition t_j , $j \in T$ and the transition probabilities r_{jh} , $j, h \in T$.

Both quantities are easily found. Consider an arbitrary state $j = (j_1, \dots, j_k)$. Define

$$Z_j = \{m: j_m = 0\} \quad (6.15)$$

That is, Z_j is the set of indices for all idle queues. All remaining queues Q_i are working exponentially at rate μ_i . The time until the next service completion is then exponentially distributed at rate $\sum_{i \in Z_j} \mu_i$, and hence

$$t_j = \frac{1}{\sum_{i \in Z_j} \mu_i} \quad (6.16)$$

The transition probabilities are given by

$$r_{jh} = \begin{cases} \frac{\mu_p}{\sum_{i \in Z_j} \mu_i} & p \notin Z_j, \quad h = (j_1, \dots, j_{p-1}, j_{p+1}, \dots, j_k) \\ \frac{\mu_k}{\sum_{i \in Z_j} \mu_i} & p \notin Z_j, \quad h = (j_1, \dots, j_{k-1}, j_k) \\ 1 & j = h = (0, \dots, 0) \\ 0 & \text{elsewhere} \end{cases}$$

(6.17)

Given the form of r_{jh} , it is always possible to arrange the states $j=1, \dots, N$ so that the transition matrix $P=\{r_{jh}\}$ is upper triangular. This ordering always has as its last state the absorbing state $(0, \dots, 0)$. Hence R , which is obtained from P by deleting the last row and column, is upper triangular, as is $(I-R)$. This in turn implies that the m_j 's can be obtained by back-substitution, and the inverse $(I-R)^{-1}$ need never actually be calculated. This amounts to a recursion on the state observed on entry. This recursion simultaneously computes the expected system times for all other states which can be reached from the initial state $(l_1+1, l_2, \dots, l_k)$.

Example:

To illustrate the imbedded Markov chain approach, a 2-queue example is considered. Two customers are present at the first queue upon arrival, and the second queue is empty. Hence the initial state is $(3,0)$, and becomes state 1. The remaining states in order are $(2,1)$; $(1,2)$; $(0,3)$;

$(2,0); (1,1); (0,2); (1,0); (0,1); (0,0)$. The service rates are $\mu_1=1.0$; $\mu_2=.25$. The matrix R and vector τ are given by

$$R = \begin{bmatrix} 0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.8 & 0 & 0.2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.8 & 0 & 0.2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.8 & 0.2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \quad \tau = \begin{bmatrix} 1.00 \\ 0.8 \\ 0.8 \\ 4.0 \\ 1.0 \\ 0.8 \\ 4.0 \\ 1.0 \\ 4.0 \end{bmatrix}$$

Directly using (6.13) and back-substitution one obtains

$$m_9 = 4.0$$

$$m_9 = 4.0$$

$$m_8 - m_9 = 1.0$$

$$m_8 = 5.0$$

$$m_7 - m_9 = 4.0$$

$$m_7 = 8.0$$

$$m_6 - .8m_7 - .2m_8 = 0.8$$

$$m_6 = 8.2$$

$$m_5 - m_6 = 1.0$$

$$m_5 = 9.2$$

$$m_4 - m_7 = 4.0$$

$$m_4 = 12.0$$

$$m_3 - .8m_4 - .2m_6 = 0.8$$

$$m_3 = 12.04$$

$$m_2 - .8m_3 - .2m_5 = 0.8$$

$$m_2 = 12.272$$

$$m_1 - m_2 = 1.0$$

$$m_1 = 13.272$$

Hence $E\{S_n/N_{1,n}=2; N_{2,n}=0\} = 13.272$. Note in conclusion that we may simultaneously compute $E\{S_n/N_{1,n}=i_1; N_{2,n}=i_2\}$ for all $i_1+i_2 \leq 2$.

Ordering: Suppose that the queue lengths observed upon arrival are

$N_{1,n} = l_1, \dots, N_{k,n} = l_k$. The initial state of the imbedded Markov chain is $(n_1, n_2, \dots, n_k) = (l_1 + 1, l_2, \dots, l_k)$. On the basis of (6.17), we know that a particular state can only lead in one step to a state where one entry in the n -tuple has decreased by one and the next entry to the right has increased by one, or where the last entry has decreased by one. Hence the sequence of states which ensures that R is upper triangular is

$(n_1, n_1, \dots, n_k); (n_1 - 1, n_2 + 1, \dots, n_k); (n_1 - 2, n_2 + 2, \dots, n_k); \dots; (0, n_2 + n_1, \dots, n_k);$
 $(n_1 - 1, n_2, n_3 + 1, \dots, n_k); \dots; (0, n_1 + n_2 - 1, n_3 + 1, \dots, n_k); (n_1 - 2, n_2, n_3 + 2, \dots, n_k); \dots;$
 $(0, n_2, n_3 + n_1, \dots, n_k); \dots; (0, \dots, 0, n_1 + \dots + n_k); (n_1 - 1, n_2, \dots, n_k); \dots;$
 $\dots; (0, \dots, 0, n_1 + \dots + n_k - 1); (n_1 - 2, n_2, \dots, n_k); \dots; (0, \dots, 0).$

Computational tractability: A discussion of the absorbing time approach would be incomplete without some mention of the effect of the number of customers and number of queues on the size of the problem.

The primary consideration is how many states are required to compute the prediction. Given a total of N customers (C_n included) distributed over k queues, there are at most $\binom{N+k}{k}$ possible states to consider. The maximum occurs when all customers are at Q_1 when C_n arrives. For example, if $N=10$ and $k=3$, a maximum of 286 states must be considered. Hence if k is moderate, the number of states required is $O(N^k)$.

6.4.2 Continuous Parameter Markov Chain Model

The behaviour of the series of queues can equally be modelled as a continuous parameter Markov chain. This method is included primarily for

$$= \begin{bmatrix} -1.0 & 1.0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1.25 & 1.0 & 0 & .25 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1.25 & 1.0 & 0 & .25 & 0 & 0 & 0 \\ 0 & 0 & 0 & -.25 & 0 & 0 & .25 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1.0 & 1.0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1.25 & 1.0 & .25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -.25 & 0 & .25 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1.0 & 1.0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -.25 \end{bmatrix}$$

Solution of $-Q_1 M = e$ produces

$$.25m_9 = 1.0$$

$$m_9 = 4.0$$

$$m_8 - m_9 = 1.0$$

$$m_8 = 5.0$$

$$.25m_7 - .25m_9 = 1.0$$

$$m_7 = 8.0$$

$$1.25m_6 - m_7 - .25m_8 = 1.0$$

$$m_6 = 8.2$$

$$m_5 - m_6 = 1.0$$

$$m_5 = 9.2$$

$$.25m_4 - .25m_7 = 1.0$$

$$m_4 = 12.0$$

$$1.25m_3 - m_4 - .25m_6 = 1.0$$

$$m_3 = 12.04$$

$$1.25m_2 - m_3 - .25m_5 = 1.0$$

$$m_2 = 12.272$$

$$m_1 - m_2 = 1.0$$

$$m_1 = 13.272$$

6.5 A Comparison of the Two Markov Chains

Essentially, both the imbedded discrete parameter and the continuous parameter Markov chain approaches are equivalent. They have the same number of states, the same state space, and both rely on solving an upper triangular system of equations. Hence for any given series of queues,

either both methods are tractable or neither is. The continuous approach offers a minor benefit in that the vector τ of mean transition times need not be specified. However, the saving is negligible when considered with other more important factors, such as the computation of the R and Q matrices.

6.6 Conclusions

A method for analyzing system times in two tandem queues was presented, and a model for analyzing more lengthy series of tandem queues was described. In the next chapter, we consider the problem of using predictions in general data networks and discuss heuristic solutions.

Figure 6.1: A Series of Queues in Tandem

CHAPTER 7

THE APPLICABILITY OF PREDICTORS TO ADAPTIVE ROUTING SCHEMES

7.1 Introduction

In preceding chapters, a variety of exact prediction results for stationary queueing systems were derived which satisfied specific arrival and/or service distribution restrictions. A logical application for queueing system predictors is in an adaptive routing scheme for data networks. In this chapter, we examine the applicability and shortcomings of such application, along with an analysis of how the predictors might be implemented and the potential benefits to be obtained. In section 7.2 the problems defined and the tractability of an analytical solution is considered. Possible heuristic predictors are developed in section 7.3. In section 7.4, two adaptive routing schemes which have been used in the ARPANET packet-switched network are presented. The ways in which the heuristic predictors could be applied to these schemes is discussed in section 7.5, and conclusions regarding the utility of further investigation of this area are given in section 7.6.

7.2 Adaptive Routing Model

The data network model to be considered consists of N nodes and L internodal links (see Figure 7.1). Link i has capacity C_i and external traffic arriving at source node j destined for node k arrives at rate λ_{jk} . The distributions for the interarrival times and the message lengths are assumed to be general and independent.

In an adaptive routing scheme, some optimization criterion is used to evaluate the performance of the message flow between source and destination. Many possible criteria exist, among them the maximization of the throughput between the nodes or minimization of the cost. However, in the current work, the optimization criterion shall be the minimization of the average delay between source and destination nodes. An adaptive routing scheme based on the delay criterion reacts favourably to topological changes such as link failure. A periodic measurement of local or global information is performed to detect any significant change in the value of the optimization criterion. Typical quantities measured are the queue lengths of output buffers and the average delay for all messages used by a particular link during a period of time. The measurements are then disseminated globally or locally, following which each source node evaluates values of the optimization criterion for a variety of routes to each destination node j . The optimal route is then selected, and all messages between nodes i and j are routed along this path until the next evaluation (update).

Chou [2] among others has considered the question of which traffic situations favour adaptive routing over a fixed routing scheme. He shows that while a fixed routing scheme tends to be preferable when the traffic load is typically balanced and stationary, adaptive routing is generally desirable when the network is subject to surges in traffic load or when network behaviour is unbalanced or chaotic. Yet the dynamic behaviour of queueing processes has not previously been looked at from a predictive viewpoint; such an application seems worthy of consideration.

First, however, let us consider the tractability of an exact analytical solution for the average message delay. Such a solution appears to be extremely unlikely. The foremost reason is the non-stationary behaviour mentioned earlier. The predictors which have been developed until now have been designed for the stationary fluctuations of a queue. While the non-stationarities produce the fluctuations which are a suitable environment for an adaptive routing scheme, they also complicate the exact analysis of even the most simple queues. Hence the presence of non-stationary behaviour acts as a two-edged sword. Furthermore, the periodic changes in routing of an adaptive routing scheme typically contribute to the non-stationary nature of the problem.

Secondly, in general data networks, there is the possibility of messages between the same source and destination nodes overtaking one another, thereby further complicating the analysis. Hence, even if rigid assumptions were made regarding the arrival patterns and message lengths, a tractable solution would likely not be possible.

Define the staleness of the measurement data as the time from the measurement evaluation until the start of the delay which a potential message would incur on the link. This time has three components. The first is the time from the data becoming available until it is transmitted. This delay may occur due to the nature of the measurement; for example, when considering all delays on a link over a 10 second period, the average delay is typically 5 seconds old. The delay may also be due to the time needed to make a decision as to whether the measurement is significantly different

from the previous value to warrant transmission. In any case, this period of time can be considered as fixed for each adaptive routing scheme, and is typically a small component of the staleness.

The second component corresponds to the time from transmission of the update until it arrives at the source node where the routing decision is to be made; this delay is due to queueing and propagation delays in the network. It could be measured by "stamping" the update with its transmission time. The final component represents the delay from the receipt of the update until the message to be transmitted reaches the node in question. This component is likely to be the most significant.

There are essentially two ways in which predictors can be applied to an adaptive routing scheme. In the first, the predictor accounts for the time that has elapsed from measurement until the update reaches the source node. This method would be designed to provide the source node with a "snapshot" of the delays after the time that the optimal routes are to be selected, and would only consider the first two components of the staleness of the data.

The second way in which predictors can be applied is to account for all three components of the staleness. Thus essentially, the predictor would be used to predict the delay that a message would undergo once it had arrived at the link in question.

In the next section heuristic predictors are developed to provide an indication of how queue length or delays would behave during the staleness

time.

7.3 Heuristic Predictors

From the analysis of the staleness time, it is clear that continuous time heuristic predictors are needed for an adaptive routing application. The predictors presented here are based on the non-monotonic trajectories of the imbedded queue length predictors of chapters 3 through 5.

In these chapters we observed that the short-term behaviour of the OMS queue length predictors was approximately linear for sufficiently large initial queue lengths. The linear segment decayed by a factor of $(1-\rho)$ per departure (equivalently, $(1-\rho)/\rho$ per arrival). The continuous time equivalent for the expected behaviour of $Q(t)$, the queue length at time t , is

$$E\{Q(t)/Q(0)=i\} = i - \frac{(1-\rho)}{\bar{x}} t; \quad i \text{ large} \quad (7.1)$$

Thus the corresponding expected delay for an arrival at time t is

$$E\{S(t)/Q(0)=i\} = \bar{x}(i+1) - (1-\rho)t; \quad i \text{ large} \quad (7.2)$$

An equivalent expression for the expected delay based upon an initial (sufficiently large) delay s_0 is

$$E\{S(t)/S(0)=s_0\} = s_0 - (1-\rho)t; \quad s_0 \text{ large} \quad (7.3)$$

Recall that the criterion for i being "significantly large" in the earlier discussions was that the probability of the queue emptying within n transitions of the chain should be very small. Therefore, the criterion

for i and s_0 being sufficiently large in the current discussion is that the likelihood of the queue emptying by time t should be negligible.

In situations where i or s_0 are moderate or small, the preceding heuristic would not be appropriate and a careful weighing of options must be made. In the case of a light load (and hence a small delay), large fluctuations are not likely to occur and hence the changing behaviour of the link during the staleness time is not crucial to the operation of the routing scheme. Hence one possible course of action would be not to use a predictor for lightly loaded links at all. In the case of a moderate load, the predicted delay could be the maximum of the value given by the appropriate equation ((7.2) or (7.3)) or some other value, such as the average delay on the link. A similar approach could handle the case of a heavily loaded system where t is large.

Notice that as the utilization exceeds 1.0 the predicted delay increases with time, and hence the suggested predictor would react quickly to sudden surges in traffic load.

Next, we present two specific adaptive routing algorithms which have been used in the ARPANET network. A discussion of the applicability of this section's predictors follows.

7.4 ARPANET Routing Algorithms

The best-known adaptive routing strategy is the original ARPANET algorithm, as described, for instance, by Schwartz [22]. This strategy, known as "shortest time plus bias", used a set of tables at each node which

represented the minimum times required to reach the various destinations and also indicated the adjacent node which corresponded to the first stop along the optimal path. Every 128 ms neighbouring nodes exchanged this information, and added the time required to reach the various adjacent nodes along each of the outgoing paths. The smallest of these was then selected as the routing path until the next update.

The old ARPANET algorithm estimated the delay between adjacent nodes by inspecting the queue lengths on outgoing lines periodically. To reduce looping, a bias constant was added to the queue length estimate.

McQuillan, Richer and Rosen [10] recently discussed the problems inherent in the old ARPANET algorithm and indicated how these problems have been ameliorated in the new algorithm. Essentially, whereas the old algorithm dispatched global information locally, the new one sends local information (in the form of delay averages on adjacent links) globally. Each node maintains a tree of shortest paths to all other nodes. An incoming packet dealing with the status of a particular link may change part or all of the tree depending on whether the new estimate represents an increase or decrease in delay and whether or not the link was formerly in the tree.

The authors felt that the "shortest-time plus bias" delay calculation reflected the effect of a particular queue length on delay too heavily, because:

- 1) Different line speeds make a comparison between queue lengths on out-

going trunks questionable.

- 2) Software constraints impose a short maximum queue length, hence a packet may spend a great deal of time acquiring resources prior to being queued for transmission while the queue length is short.
- 3) Significant fluctuations in queue lengths (and hence the queue length at an arbitrary instant) render them a poor indicator of average delay.

As a result, the new ARPANET algorithm measures all delays incurred by packets using a particular link, and computes a new average delay every 10 seconds. If the change in delay is significant, a new delay estimate floods the network, usually reaching all nodes within 100 ms.

The flaws in the use of queue length data in the original ARPANET algorithm reflect more on the way in which the data was used than on any inherent lack of relevance of queue lengths. In fact several of the problems mentioned in McQuillan et al [10] can be overcome by predictors:

- 1) A difference in line speeds can be accounted for by adjusting the appropriate utilization. The predictor would indicate that the line with the greater line speed would reduce its load more quickly, as the user would hope.
- 2) Instead of computing the average of all delays during each 10 second period, the average queue length at departure or arrival instants during the same period could be computed. In addition, the queue lengths have the advantage of being less stale, as they provide an indication of future delays. This question was dealt with at greater length in Chapter 3 where the utility of using previous system times to predict later system times was discussed.

In any case, heuristic predictors were suggested in the preceding section which would use either previous delays or previous queue lengths. In the next section, possible means for implementing these predictors are considered.

7.5 Implementation of the predictors

The predictors which have been developed cannot be implemented in the old ARPANET algorithm. They have been geared to reflect the changing delay on a particular link over time: the further away the link is, the greater the effect of the predictor. Since the old algorithm provides estimates of total delay which cannot be broken down into their component delays, and since the entire route to the destination node is not known, the predictors cannot be used.

By way of contrast, the form of the new ARPANET algorithm is amenable to the implementation of the predictors. The tree of shortest paths allows us to identify the entire route to a destination node; furthermore, the individual delays along particular links are available at each node.

In implementing the "snapshot" model, an estimate of the first two components of the staleness time are needed. The first, a function of the adaptive routing strategy, is assumed to be (more or less) constant and known. The second component, namely time that the update packet spends in the network, can be calculated by encoding the transmission time in the update packet. Upon arrival at the source node, the elapsed time can be found. The sum of these two times is then used to produce an estimate of the current link delay using the heuristic predictors of section 7.3. The estimate thus produced acts as the delay until the next update is received.

In most situations, the delay that a message undergoes in reaching the link in question, is the largest component of the staleness time. Hence the latter method of prediction, which accounts for the entire staleness time, appears to be a more appropriate method. The time through the network to reach the link in question can be estimated by the sums of the delays on the links that precede it. This component of the staleness would then be added to the first two and the result would be used to generate the predictions as before.

The use of a prediction approach necessitates that the updates be performed frequently enough so that the time from the receipt of an update until the message is transmitted is kept small. If not, this becomes a significant further component of the staleness, and should be accounted for.

7.6 Conclusions

This chapter has taken an introductory look at the use of prediction in an adaptive routing scheme. The various components of the staleness of the data have been identified, and simple heuristic predictors for the delay have been suggested. Further refinement of these predictors as well as testing in a variety of adaptive routing environments are required before a conclusive evaluation of their merit can be assessed.

Figure 7.1: A Data Network

CHAPTER 8

CONCLUSIONS AND FURTHER WORK

8.1 Conclusions

This thesis has presented a variety of new results for the prediction of queue length and delay processes in several queues. In each case, the predictions and the mean squared errors of the delay processes were found to be closely related to those for the queue length. Furthermore, for appropriate initial values of the queue length, the OMS prediction trajectories displayed an initial linear decay. The duration of the linear decay was found to be related to the probability of the queue emptying. Once this probability became sufficiently large, the prediction trajectories ceased to decay linearly.

In the case of GI/M/1 queues, comparisons were made between the OMS and OLMS predictors. In both cases the delay predictions used the queue length prediction as an intermediate result; the same situation applied for the respective mean squared errors. It was observed that the linear predictors were not solely dependent upon the most recent of the available data; however earlier data had a marginal effect in terms of the prediction and in reducing the mean squared error. Furthermore, the OLMS error was typically only slightly larger than its OMS counterpart. Lastly, OLMS system time predictors based on previous system times were shown to be more accurate than those based on queue lengths; however it was

observed that the system time data is only available much later than the corresponding queue length data.

In the case of GI/M/m predictions, it was shown that the waiting time predictor used a restricted summation of terms for the queue length predictor as an intermediate result. As a consequence, the shapes of the waiting time and queue length trajectories were not identical, whereas the single server trajectories had been. An increase in the number of servers produced a decrease in the normalized waiting time prediction errors; however, the behaviour of the queue length error was more complex.

Two sets of expressions for the system time predictor in M/G/1 queues were derived. The relative values of the prediction horizon n and initial queue length i determined which set of expressions was applicable. For $n > i$, the results were similar to those for GI/M/1 in that they used the queue length predictions as intermediate results. For $n \leq i$, the queue length prediction trajectories were identical straight lines irrespective of the service distribution; at the same time, the corresponding probability of the queue emptying was 0. These facts provide the clearest example of the duration of the linear decay. Finally, in the case of M/M/1 queues, the queue length predictors were shown to have a smaller predictor error using the M/G/1 approach than by using the GI/M/1 approach.

Predictions results for the total time spent in a series of tandem M/M/1 queues conditioned upon the queue lengths observed upon arrival to the first queue were developed. In the case of two queues in tandem, a predictor similar to those for single GI/M/1 queues was developed. This

approach failed to extend to three or more queues in tandem due to the possibility of intermediate queues falling idle prior to the arrival of the customer in question. In this case, an absorption time approach was used.

Several aspects of the applicability of predictors to adaptive routing schemes in data networks were discussed in Chapter 7. The value of further analysis of the area was demonstrated.

8.2 Recommendations for Further Work

This thesis has presented new analytical results for the prediction of delay times based upon values of the imbedded queue length process in each of the GI/M/1, GI/M/m, and M/G/1 queues. There appears to be little likelihood of further extension of these results to the GI/G/1 queue; however, significant work could be done using approximations and bounds for the delay based upon the queue length at arrival instants in M/G/1 and GI/M/1 queues. Extensions for the virtual waiting time are also possible in GI/M/1 queues.

In the area of tandem queues, an appropriate extension would be to consider external arrivals to intermediate nodes. Possible extensions to queues with feedback and Jackson networks merit further consideration.

A great deal of further work is called for in the application of predictors to adaptive routing schemes. These investigations should include further refinement of the proposed predictors, and analysis of the relative size of the components of the staleness time. An assessment of the overhead associated with various implementations of these predictors should be carried out. Finally, further simulation studies should be

carried out [✓]to identify those traffic environments favourable to a prospective implementation of predictors.

APPENDIX 1

CORRELATION COEFFICIENTS FOR QUEUE LENGTH, SYSTEM TIME, AND WAITING TIME IN GI/M/1 QUEUES

The purpose of this appendix is to review the serial correlation coefficients for queue length at arrival instants $r_N(n)$, system time $r_S(n)$, and waiting time $r_W(n)$ in the stationary GI/M/1 queue. These results are needed in order to calculate the coefficients for the corresponding linear predictors which are considered in Chapter 3.

In addition, new results relating to the cross-correlation coefficients between queue length and each of the other processes will be presented. These results will enable us to calculate the coefficients for the linear predictors for system time and waiting time, based upon queue length information.

A1.1 Pakes Algorithm for $r_N(n)$

By definition, the n -step covariance of the queue length γ_n is

$$\begin{aligned}\gamma_n &= \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} (i - \mu_N)(j - \mu_N) \Pr\{N_k = i; N_{(k+n)} = j\} \\ &= \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} ij \pi_i p_{ij}^n - \mu_N^2\end{aligned}\tag{A1.1}$$

Therefore, the z-transform for the correlation coefficients $r_N(n)$ is

$$\begin{aligned}
 F(z) &= \sum_{n=0}^{\infty} r_N(n) z^n \\
 &= \frac{1}{\sigma_N^2} \left\{ \sum_{n=0}^{\infty} \left(\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} i j \pi_i p_{ij}^n - (\mu_N^2) z^n \right) \right\} \\
 &= \frac{1}{\sigma_N^2} \left\{ \sum_{n=0}^{\infty} \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \left[i j (1-d) d^i p_{ij}^n z^n \right] - \frac{(\mu_N^2)}{(1-z)} \right\} \quad (A1.2)
 \end{aligned}$$

This latest step follows from the results for the steady state queue length distribution (at arrival instants) which were stated in Chapter 2.

Pakes [14] strategy to find $r_N(n)$ consists of

- Evaluating $F(z)$ using available z -transforms
- Inverting $F(z)$ to obtain the coefficients.

Part (a): Takacs [24], page 115, provides an expression for the transition probabilities p_{ij}^n transformed over i and n :

$$\begin{aligned}
 H_j(w, z) &= \sum_{n=0}^{\infty} \sum_{i=0}^{\infty} p_{ij}^n w^i z^n \\
 &= \frac{(1-z)(1-w)^{j+1} + z[w - \phi(\mu(1-w))](1-d(z))d(z)^j}{(1-z)(1-w)(w - z\phi(\mu(1-w)))} \quad (A1.3)
 \end{aligned}$$

Since

$$\left. \frac{\partial H_j(w, z)}{\partial w} \right|_{w=d} = \sum_{n=0}^{\infty} \sum_{i=0}^{\infty} i p_{ij}^n d^{i-1} z^n$$

we see that

$$F(z) = \frac{1}{\sigma_N^2} \left\{ \sum_{j=0}^{\infty} j(1-d)d \left[\left. \frac{\partial H_j(w, z)}{\partial w} \right|_{w=d} \right] - \frac{(\mu_N^2)}{(1-z)} \right\} \quad (A1.4)$$

After having performed the necessary differentiation, evaluation, and substitution, we arrive at the following expression for $F(z)$:

$$F(z) = \frac{1}{(1-z)} - \frac{z}{(1-z)^2} \cdot \frac{d(1-d)}{d'(1)} + \frac{(1-d)^2}{d'(1)} \left[\frac{d(z)}{(1-d(z))} \cdot \frac{z}{(1-z)^2} \right] \quad (A1.5)$$

In fact, several substitutions have taken place between (A1.4) and (A1.5) such as

$$\sigma_N^2 = \frac{d}{(1-d)^2}; \quad \bar{N} = \frac{d}{(1-d)}; \quad d'(1) = \frac{d}{1+\mu\phi'(\mu(1-d))}$$

Part (b): We now proceed to invert $F(z)$ to obtain the correlation coefficients $r_N(n)$. We will use the following property of z-transforms, and two z-transform pairs:

Convolution Property:

Let $\{a_n\}, \{b_n\}$ be two well defined discrete sequences, and define

$$A(z) = \sum_{n=0}^{\infty} a_n z^n; \quad B(z) = \sum_{n=0}^{\infty} b_n z^n$$

provided that they exist. Then the discrete sequence

$$c_n = \sum_{\ell=0}^n a_{n-\ell} b_{\ell}$$

has z-transform $C(z)$ given by

$$C(z) = A(z)B(z).$$

2 z-transform pairs:

$$\text{Let } C(z) = \sum_{n=0}^{\infty} g_n z^n$$

If $g_n = 1 \forall n$, then $C(z) = \frac{1}{1-z}$

If $g_n = n$, then $C(z) = \frac{z}{(1-z)^2}$

We are now ready to invert (A1.5). Using the convolution property for the bracketed term, we find

$$r_N(n) = 1 - n \cdot \frac{d(1-d)}{d'(1)} + \frac{(1-d)^2}{d'(1)} \left\{ \sum_{\ell=0}^n (n-\ell) \beta_\ell \right\} \quad (A1.6)$$

$$\text{where } \sum_{n=0}^{\infty} \beta_n z^n = \frac{d(z)}{(1-d(z))} \quad (A1.7)$$

The algorithm defined by (A1.6) will be complete once an explicit expression for $\{\beta_n\}$ can be found. Pakes, in [14], shows

$$\beta_j = \frac{1}{j!} \frac{d^{j-1}}{dx^{j-1}} \{F^j(x)/(1-x)^2\} \Big|_{x=0} \quad j \geq 1.$$

where

$$F(x) = \int_0^{\infty} e^{-\mu(1-x)t} dA(t)$$

but clearly this method would require a great deal of computation to find even the first few terms of $r_N(n)$.

There is an alternate approach, however. Pakes refers to an expression derived by Takacs. Define

$$f_0^{(n)} = \Pr\{N_n=0; N_m>0 \ (m=1, \dots, n-1) / N_0=0\}$$

In conceptual terms, $f_0^{(n)}$ is the probability that n customers are served during a busy period. Takacs [24], p. 116, has shown that

$$\sum_{n=1}^{\infty} f_0^{(n)} z^n = \frac{(z-d(z))}{1-d(z)} \quad (A1.8)$$

The relationship between the right hand sides of (A1.7) and (A1.8) is such that $\{\beta_n\}$ can be expressed as (Pakes, []).

$$\begin{aligned} \beta_0 &= 0 \\ \beta_n &= 1 - \sum_{k=1}^n f_0^{(k)} \end{aligned} \quad (\text{A1.9})$$

Given the definition of $f_0^{(n)}$, this is equivalent to saying

$$\beta_n = \Pr\{N_n > 0, N_m > 0 \ (m=1, \dots, n-1) / N_0 = 0\}$$

Several means of calculating $\{\beta_n\}$ follow from this probabilistic interpretation. One of the simplest ways makes use of results by Prabhu [], p. 117, in order to arrive at

$$\beta_n = \frac{1}{n} \sum_j j k_{n-j}^{(n)} \quad (\text{A1.10})$$

where

$$k_i^{(j)} = \Pr\{i \text{ Poisson events at rate } \mu \text{ during } j \text{ interval periods}\}$$

Since the $k_i^{(j)}$ terms can be recursively calculated, this proves to be a manageable way to find β_n .

Therefore, the algorithm can be recursively stated (following from (2.6) and (2.10)) as:

$$\begin{aligned} r_N(0) &= 1 \\ r_N(n) &= r_N(n-1) - \frac{(1-d)d}{d^*(1)} + \frac{(1-d)^2}{d^*(1)} \sum_{j=1}^{n-1} \beta_j \end{aligned} \quad (\text{A1.11})$$

where

$$\beta_n = \frac{1}{n} \sum_{j=1}^n j k_{n-j}^{(n)} ; \quad n \geq 1$$

It can be seen that the form given by (A1.11) does not represent a geometric sequence of the form

$$h_n = (r)^n ; \quad 0 < r < 1$$

As a result, we know immediately that the queue length process $\{N_n\}$ is not a wide-sense Markov process. Hence, as we shall see in Chapter III, a linear predictor based upon a certain number (greater than one) of observations will not depend solely upon the last of these observations.

Examples of some typical correlation coefficients can be found in Tables A1.1 and A1.2. The cases considered are:

- a) Markovian arrivals with intensities of .5, .7, and .9.
- b) Erlang-5 arrivals with intensities of .5, .7, and .9.

These examples reveal several characteristics:

- i) An increase in ρ leads to greater correlation, as expected.
- ii) An increase in the order of the Erlang density (recall that Markovian arrivals are Erlang-1) leads to a reduced correlation.
- iii) Successive ratios $r_N(n+1)/r_N(n)$ increase, although the sequence is convex.

Since the linear predictor coefficient based on a single observation will be shown to be the appropriate correlation coefficient, these tables also list the single observation coefficients. More will be said about this in Chapter 3.

A1.2 Algorithms for $r_W(n)$ and $r_S(n)$ Waiting times:

Pagurek and Woodside [11] have developed two algorithms for the computation of $r_W(n)$ and $r_S(n)$, the serial correlation coefficients for waiting and system time, respectively, for GI/G/1 queues having rational arrival processes. On a more limited scale, Pakes [15] derived an algorithm for $r_W(n)$ in GI/M/1 queues. However, the algorithm was computationally impractical. Pagurek [13] developed an expression found in the proof of Pakes [15] algorithm, and thereby obtained a more efficient means to calculate the coefficients. This was presented and discussed extensively in Stanford [23].

The version of the algorithm quoted in [23] was

$$\begin{aligned}
 r_W(0) &= 1, \\
 r_W(n) &= r_W(n-1) + \frac{(1-d)}{\mu d \sigma_W^2} [E\{W_n/W_0=0\} - E\{W_{n-1}/W_0=0\}] \\
 &\quad + \frac{1}{\mu d^2(1)\sigma_W^2} [E\{W_{n-1}/W_0=0\} - E\{W\}]
 \end{aligned} \tag{A1.12}$$

while the version stated in Pakes [15] was

$$\begin{aligned}
 r_W(0) &= 1 \\
 r_W(n) &= r_W(n-1) - \frac{(1-d)}{(2-d)d^2(1)} + \frac{(1-d)^3}{d^2(2-d)} \beta_n + \frac{(1-d)^2}{d(2-d)d^2(1)} \sum_{k=1}^{n-1} \beta_k
 \end{aligned} \tag{A1.13}$$

where the β_k terms are defined as

$$\beta_k = \Pr\{W_m \neq 0 \ (m=1, \dots, n)/W_0=0\}$$

A comparison of Table 2.1 and Stanford [23], Tables 5.1 and 5.2 reveals that the waiting times are more highly correlated than the queue lengths, but that this difference decreases as ρ increases.

System times:

Pakes [15] then makes use of the relation

$$\text{Cov}\{S_0, S_n\} = \text{Cov}\{W_0, W_n\} + \text{Cov}\{X_0, X_n\} \quad (\text{A1.14})$$

in order to develop an algorithm for $r_S(n)$. He arrives at the following expression for the z-transform for $r_S(n)$:

$$\begin{aligned} G(z) &= \sum_{n=0}^{\infty} r_S(n) z^n \\ &= \frac{1}{1-z} - \frac{z}{(1-z)^2} \frac{d(1-d)}{d'(1)} + \frac{zd(z)}{(1-z)^2(1-d(z))} \cdot \frac{(1-d)^2}{d'(1)} \end{aligned}$$

But since this is the same as (A1.5), it follows that $\forall n$:

$$r_S(n) = r_N(n) \quad (\text{A1.15})$$

This rather interesting result which is independent of the interarrival distribution will prove to be of great use in the study of linear predictors in Chapter 3.

For the moment, however, it is interesting to note that the comments on the properties of $r_N(n)$ in GI/M/1 queues also extend to $r_S(n)$. In particular, we observe that waiting times are more highly correlated than system times:

$$r_W(n) > r_S(n) \quad \forall n$$

A1.3 Results for $r_{NW}(n)$ and $r_{NS}(n)$:

Let us define the serial cross-correlation coefficients

$$r_{NW}(n) = \frac{\text{Cov}\{N_k, W_{k+n}\}}{\sqrt{\sigma_N^2 \sigma_W^2}} \quad n \geq 0 \quad (\text{A1.16})$$

and

$$r_{NS}(n) = \frac{\text{Cov}\{N_k, S_{k+n}\}}{\sqrt{\sigma_N^2 \sigma_S^2}} \quad n \geq 0 \quad (\text{A1.17})$$

It is our intention to derive expressions for $r_{NW}(n)$ and $r_{NS}(n)$. We start by observing that

$$\text{Cov}\{N_k, S_{k+n}\} = \text{Cov}\{N_k, W_{k+n}\} + \text{Cov}\{N_k, X_{k+n}\}$$

The latter term vanishes since X_{k+n} is independent of N_k for all $n \geq 0$, and so it follows that

$$r_{NW}(n) = \frac{\sqrt{\sigma_S^2}}{\sqrt{\sigma_W^2}} r_{NS}(n) = \frac{1}{2d-d^2} r_{NS}(n) \quad (\text{A1.18})$$

Since $2d-d^2 = d(2-d)$ and $d < 1$, it follows that

$$r_{NW}(n) > r_{NS}(n)$$

In the development which follows, it will be shown that

$$r_{NS}(n) = \sqrt{d} r_N(n) \quad \forall n \geq 0 \quad (\text{A1.19})$$

and so $r_{NS}(n)$ and $r_{NW}(n)$ can be found using (A1.11), (A1.18) and (A1.19).

Proof of (A1.19): The proof that follows will establish that $\forall n \geq 0$,

$$\text{Cov}\{N_k, S_{k+n}\} = \left(\frac{1}{\mu}\right) \text{Cov}\{N_k, N_{k+n}\} \quad (\text{A1.20})$$

from which (A1.19) follows directly. We start with

$$\text{Cov}\{N_k, S_{k+n}\} = E\{N_k, S_{k+n}\} - \mu_N \mu_S \quad (\text{A1.21})$$

However,

$$E\{N_k, S_{k+n}\} = \sum_{i=0}^{\infty} \int_0^{\infty} i t d\text{Pr}\{N_k = i; S_{k+n} < t\}$$

$$= \sum_{i=0}^{\infty} i \pi_i \int_0^{\infty} t d\Pr \{S_{k+n} \leq t / N_k = i\} = \sum_{i=0}^{\infty} i \pi_i E\{S_{k+n} / N_k = i\} \quad (A1.22)$$

But in the mathematical preliminaries we found

$$E\{S_{k+n} / N_k = i\} = \left(\frac{1}{\mu}\right) [E\{N_{k+n} / N_k = i\} + 1]$$

Therefore

$$\begin{aligned} E\{N_k S_{k+n}\} &= \left(\frac{1}{\mu}\right) \sum_{i=0}^{\infty} i \pi_i [E\{N_{k+n} / N_k = i\} + 1] \\ &= \left(\frac{1}{\mu}\right) [E\{N_k N_{k+n}\} + \mu] \end{aligned} \quad (A1.23)$$

Substitution into (A1.21) using (A1.23), (2.3) and (2.7) yields

$$\begin{aligned} \text{Cov}\{N_k, S_{k+n}\} &= \left(\frac{1}{\mu}\right) [E\{N_k N_{k+n}\} + \frac{d}{1-d}] - \frac{d}{1-d} \frac{1}{\mu(1-d)} \\ &= \left(\frac{1}{\mu}\right) [E\{N_k N_{k+n}\} - \frac{d^2}{(1-d)^2}] \\ &= \left(\frac{1}{\mu}\right) \text{Cov}\{N_k, N_{k+n}\} \end{aligned} \quad (A1.24)$$

This proves (A1.20), and (A1.19) follows directly.

A1.4 Conclusions

We have stated an algorithm by Pakes for the serial correlation coefficients $r_N(n)$. Through a proof by Pakes, we showed that

$$r_S(n) = r_N(n)$$

and by deriving new results we found that

$$r_{NS}(n) = \sqrt{d} r_N(n) \quad (A1.25)$$

$$r_{NW}(n) = \frac{1}{\sqrt{2-d}} r_N(n) \quad (A1.26)$$

As a result, the algorithm for $r_N(n)$ can be used to find the other coefficients as well.

In addition, we stated Pakes' algorithm for the waiting time serial correlation coefficients $r_W(n)$. By inspecting typical results, we observed that

$$r_W(n) > r_N(n) \quad \forall n$$

and that the discrepancy between them decreases as ρ approaches 1. On account of the equalities already established, this relationship regarding $r_W(n)$ extends to the cross correlations as well.

TABLE A1.1

M/M/1 Serial Correlation Coefficients $r_N(n)$

n	ρ		
	.5	.7	.9
1	.75000	.91000	.99000
2	.58333	.83588	.98053
3	.46296	.77215	.97143
4	.37243	.71615	.96265
5	.30258	.66625	.95413
6	.24774	.62138	.94583
7	.20410	.58073	.93775
8	.16902	.54370	.92985
9	.14059	.50981	.92213
10	.11738	.47868	.91457
11	.09832	.45000	.90716
12	.08260	.42349	.89989

TABLE A1.2

 $E_5/M/1$ Serial Correlation Coefficients $r_N(n)$

n	ρ		
	.5	.7	.9
1	.59973	.85162	.98335
2	.38731	.73952	.96789
3	.25971	.64921	.95325
4	.17837	.57417	.93928
5	.12460	.51064	.92587
6	.08816	.45612	.91294
7	.06302	.40889	.90044
8	.04542	.36766	.88833
9	.03296	.33144	.87657
10	.02406	.29946	.86515
11	.01765	.27110	.85403
12	.01301	.24587	.84319

APPENDIX 2

THE SERIAL CORRELATION COEFFICIENTS FOR WAITING TIMES IN THE GI/M/m QUEUE

A2.1 Introduction.

The serial correlation coefficients for a variety of processes in the GI/M/1 and M/G/1 queues are known; however, little is known about the correlation properties of multi-server queues.

In this appendix we consider the serial correlation coefficients for the waiting time process in GI/M/m queues, thereby extending the work of Pakes [15] for GI/M/1 queues. The usefulness of these coefficients has been discussed elsewhere in the thesis, and includes OLMS prediction of waiting times based on earlier waiting times.

The distribution of the stationary waiting time process in GI/M/m is given by

$$F_W(t) = 1 - ce^{-m\mu(1-d)t} \quad ; \quad t \geq 0 \quad (A2.1)$$

where $z=d$ is the unique root in $0 < z < 1$ of

$$z = \phi(m\mu(1-z)) \quad (A2.2)$$

and where

$$c = \Pr\{N \geq m\}$$

Hence

$$\mu_w = \frac{c}{m\mu(1-d)} \quad ; \quad \sigma_w^2 = \frac{c(2-c)}{(m\mu(1-d))^2} \quad (A2.3)$$

We use these results to obtain

$$r_W(n) = \frac{\text{Cov}\{W_0, W_n\}}{\sigma_W^2} = \frac{E\{W_0 W_n\}}{\sigma_W^2} - \frac{(W)^2}{\sigma_W^2} \quad (\text{A2.4})$$

The approach follows that of Pakes [15], namely:

- 1) Obtain an expression for $E\{e^{-\theta W_0}; W_1 \leq x\}$
- 2) Use this to get $E\{W_0; W_1 \leq x\}$ in terms of $E\{W_1; W_1 \leq x\}$
- 3) Obtain $E\{W_0 W_n\}$ in terms of $E\{W_0 W_{n-1}\}$ using $E\{W_0; W_1 \leq x\}$
- 4) Use (A2.4) to obtain $r_W(n)$.

A2.2 An expression for $E\{e^{-\theta W_0}; W_1 \leq x\}$

In the GI/M/1 queue, we have

$$W_1 = (W_0 + X_0 - T_1)^+ \quad (\text{A2.5})$$

There is an equivalent expression in GI/M/m. Define V_0 as the time from C_0 's entry into service until the next customer departs. Then we have

$$W_1 = (W_0 + V_0 - T_1)^+ \quad (\text{A2.6})$$

Either C_0 does not wait, or V_0 is distributed by

$$F_V(t) = 1 - e^{-m\mu t}; \quad t \geq 0 \quad (\text{A2.7})$$

and so we have

$$\begin{aligned} E\{e^{-\theta W_0}; W_1 \leq x\} &= E\{e^{-\theta W_0}; (W_0 + V_0 - T_1)^+ \leq x\} \\ &= \int_0^\infty \int_0^{x+y} \int_0^{x+y-z} e^{-\theta u} dF_W(u) dF_V(z) dF_T(y) \end{aligned} \quad (\text{A2.8})$$

Now from (A2.1) we have

$$\int_0^{x+y-z} e^{-\theta u} dF_W(u) = (1-c) + \frac{cv(1-e^{-(\theta+v)(x+y-z)})}{(\theta+v)} \quad (\text{A2.9})$$

for $v=m(1-d)$. Similarly using (A2.7) we find

$$\begin{aligned}
& \int_0^{x+y} (\cdot) dF_V(z) \\
&= \int_0^{x+y} \left[(1-c) + \frac{cv(1-e^{-(\theta+v)(x+y-z)})}{(\theta+v)} \right] m_\mu e^{-m_\mu z} dz \\
&= \frac{\theta(1-c)+v}{v+\theta} F_V(x+y) + \frac{cve^{-(\theta+v)(x+y)}}{\theta+v} \frac{m_\mu (1-e^{-(m_\mu-\theta-v)(x+y)})}{(m_\mu-\theta-v)}
\end{aligned} \tag{A2.10}$$

Now integrating

$$\begin{aligned}
\int_0^\infty (\cdot) dF_T(y) &= \frac{\theta(1-c)+v}{(v+\theta)} \int_0^\infty (1-e^{-m_\mu(x+y)}) dF_T(y) \\
&\quad - \frac{m_\mu cv}{(v+\theta)(m_\mu-\theta-v)} \int_0^\infty \left[e^{-(\theta+v)(x+y)} - e^{-m_\mu(x+y)} \right] dF_T(y)
\end{aligned}$$

and so

$$\begin{aligned}
E\{e^{-\theta W_0}; W_1 \leq x\} &= \frac{\theta(1-c)+v}{v+\theta} (1 - e^{-m_\mu x} \phi(m_\mu)) \\
&\quad - \frac{m_\mu vc}{(v+\theta)(m_\mu-\theta-v)} \left[e^{-(\theta+v)x} \phi(\theta+v) - e^{-m_\mu x} \phi(m_\mu) \right]
\end{aligned} \tag{A2.11}$$

A2.3 An expression for $E\{W_0; W_1 \leq x\}$

After differentiating (A2.11) w.r.t. θ , evaluating at $\theta=0$, and changing signs we get

$$\begin{aligned}
E\{W_0; W_1 \leq x\} &= \frac{c}{v} [1 - e^{-m_\mu x} \phi(m_\mu)] + \frac{m_\mu c}{(m_\mu-v)} [\phi'(v) e^{-vx} - x \phi(v) e^{-vx}] \\
&= \frac{m_\mu c}{v} \frac{(m_\mu-2v)}{(m_\mu-v)^2} [e^{-vx} \phi(v) - e^{-m_\mu x} \phi(m_\mu)]
\end{aligned} \tag{A2.12}$$

Based on the equation

$$m_\mu - v(t) = m_\mu t \phi(v(t)) \tag{A2.13}$$

we get

$$\phi'(v) = \frac{-(v' + m_\mu - v)}{m_\mu v} \tag{A2.14}$$

Based on (A2.2), we have

$$\phi(v) = \frac{m\mu - v}{m\mu} \quad (A2.15)$$

Substitution of (A2.14) and (A2.15) into (A2.12) yields

$$\begin{aligned} E\{W_0; W_1 \leq x\} &= \frac{c}{v} [1 - e^{-m\mu x} \phi(m\mu)] - \frac{m\mu c}{(m\mu - v)} \left[\frac{(v' + m\mu - v)}{m\mu v'} e^{-vx} + x \frac{(m\mu - v)}{m\mu} e^{-vx} \right] \\ &\quad - \frac{m\mu c}{v} \frac{(m\mu - 2v)}{(m\mu - v)^2} \left[e^{-vx} \frac{(m\mu - v)}{m\mu} - e^{-m\mu x} \phi(m\mu) \right] \end{aligned} \quad (A2.16)$$

We now hope to identify some expressions in (A2.16). Note that from (A2.1) we have

$$\Pr\{W \leq x\} = 1 - ce^{-vx} \quad (A2.17)$$

and also

$$E\{W_1; W_1 \leq x\} = c \left[\frac{1 - e^{-vx}}{v} - xe^{-vx} \right] \quad (A2.18)$$

Rearranging (A2.16) and substituting for (A2.18) gives

$$\begin{aligned} E\{W_0; W_1 \leq x\} &= E\{W_1; W_1 \leq x\} - \frac{ce^{-vx}}{v} \\ &\quad + \frac{c}{v} \left[\frac{m\mu(m\mu - 2v)}{(m\mu - v)^2} - 1 \right] \phi(m\mu) e^{-m\mu x} \end{aligned}$$

By adding and subtracting $\frac{1}{v} U(x)$, simplifying $\frac{m\mu(m\mu - 2v)}{(m\mu - v)^2} - 1$, and then substituting for (A2.17) we find

$$\begin{aligned} E\{W_0; W_1 \leq x\} &= E\{W_1; W_1 \leq x\} + \frac{1}{v} \Pr\{W_1 \leq x\} \\ &\quad - \frac{c}{v} \left(\frac{v^2}{(m\mu - v)^2} \right) \phi(m\mu) e^{-m\mu x} - \frac{1}{v} U(x) \end{aligned} \quad (A2.19)$$

Now consider

$$\phi(m\mu) e^{-m\mu x} = e^{-m\mu x} \int_0^{\infty} e^{-m\mu t} dF_T(t) = \int_0^{\infty} e^{-m\mu(x+t)} dF_T(t)$$

Similarly

$$\begin{aligned} \Pr\{W_1 > x / N_0 = m-1\} &= \int_0^{\infty} d\Pr\{\text{no completions in } (x+t) / N_0 = m-1; T_1 = t\} \\ &= \int_0^{\infty} e^{-m\mu(x+t)} dF_T(t) \end{aligned}$$

Therefore

$$\Pr\{W_1 > x / N_0 = m-1\} = \phi(m\mu) e^{-m\mu x}$$

and hence

$$\Pr\{W_1 \leq x / N_0 = m-1\} = 1 - \phi(m\mu) e^{-m\mu x} \quad (\text{A2.20})$$

By adding and subtracting $\frac{cv}{(m\mu-v)^2} U(x)$ to (A2.19) and then substituting for (A2.20) we have

$$\begin{aligned} E\{W_0; W_1 \leq x\} &= E\{W_1; W_1 \leq x\} + \frac{1}{v} \Pr\{W_1 \leq x\} \\ &\quad + \frac{cv}{(m\mu-v)^2} \Pr\{W_1 \leq x / N_0 = m-1\} - \left[\frac{cv}{(m\mu-v)^2} + \frac{1}{v} \right] U(x) \end{aligned} \quad (\text{A2.21})$$

A2.4 An expression for $E\{W_0 W_n\}$

Since

$$E\{W_0 W_n\} = E\{W_n / W_1 = x\} dE\{W_0; W_1 \leq x\} \quad (\text{A2.22})$$

substitution of (A2.21) into (A2.22) produces

$$\begin{aligned} E\{W_0 W_n\} &= \int_0^{\infty} E\{W_n / W_1 = x\} dE\{W_1; W_1 \leq x\} \\ &\quad + \frac{1}{v} \int_0^{\infty} E\{W_n / W_1 = x\} d\Pr\{W_1 \leq x\} + \frac{cv}{(m\mu-v)^2} \int_0^{\infty} E\{W_n / W_1 = x\} d\Pr\{W_1 \leq x / N_0 = m-1\} \\ &\quad - \left[\frac{cv}{(m\mu-v)^2} + \frac{1}{v} \right] \int_0^{\infty} E\{W_n / W_1 = x\} dU(x) \end{aligned}$$

Since $U(x)$ is the unit step function at $x=0$, this last integral is simply $E\{W_n / W_1 = 0\}$. Thus we have

$$\begin{aligned} E\{W_0 W_n\} &= E\{W_1 W_n\} + \frac{1}{v} E\{W_n\} + \frac{cv}{(m\mu-v)^2} E\{W_n / N_0 = m-1\} \\ &\quad - \left[\frac{cv}{(m\mu-v)^2} + \frac{1}{v} \right] E\{W_n / W_1 = 0\} \end{aligned}$$

and since the process is stationary we may write

$$\begin{aligned}
E\{W_0 W_n\} &= E\{W_0 W_{n-1}\} + \frac{1}{v'} E\{W\} + \frac{cv}{(m\mu - v)^2} E\{W_n / N_0 = m-1\} \\
&\quad - \left(\frac{cv}{(m\mu - v)^2} + \frac{1}{v'} \right) E\{W_{n-1} / W_0 = 0\}
\end{aligned} \tag{A2.23}$$

A2.5 An expression for $r_W(n)$

By subtracting $\bar{W}^2$ from both sides of (A2.23), dividing this result by σ_W^2 and substituting for (A2.4) we get

$$\begin{aligned}
r_W(n) &= r_W(n-1) + \frac{v^2}{cv'} E\{W\} + \frac{v^3}{(m\mu - v)^2} E\{W_n / N_0 = m-1\} \\
&\quad - \left(\frac{v^3}{(m\mu - v)^2} + \frac{v^2}{cv'} \right) E\{W_{n-1} / W_0 = 0\}
\end{aligned} \tag{A2.24}$$

Since $v = m\mu(1-d)$, $v' = -m\mu d$, we can also write

$$\begin{aligned}
r_W(n) &= r_W(n-1) - \frac{(m\mu)(1-d)^2}{cd'} E\{W\} + \frac{m\mu(1-d)^3}{d} E\{W_n / N_0 = m-1\} \\
&\quad - \left(\frac{m\mu(1-d)^2}{cd'} + \frac{m\mu(1-d)^3}{d} \right) E\{W_{n-1} / W_0 = 0\}
\end{aligned} \tag{A2.25}$$

The values of $E\{W_n / N_0 = m-1\}$ can be found from the QMS waiting time predictor based on queue lengths; see equation (4.17). Similarly

$$E\{W_{n-1} / W_0 = 0\} = \left(\sum_{i=0}^{m-1} \pi_i E\{W_{n-1} / N_0 = i\} \right) / \left(\sum_{i=0}^{m-1} \pi_i \right) \tag{A2.26}$$

and so the same predictors can be used for both conditional expectations. Since $E\{W\}$ is given by (A2.3), this completely specifies the recursive algorithm (A2.25).

APPENDIX 3

PROGRAM LISTINGS

This appendix contains the program listings for the OMS predictors which are considered in chapters 3 through 5. They are titled as appropriate.

GI/M/1 PREDICTION PROGRAM

IMPLICIT DOUBLE PRECISION(A-H,K,O-Z)

INTEGER R

THE PURPOSE OF THIS PROGRAM IS TO CALCULATE
 $E(N(K)/N(0)=1)$ FOR VARYING VALUES OF K AND I.

DIMENSION EODD(500),EVEN(500),EODD2(500),EVEN2(500)
 DIMENSION K(500),PROB1(500),PI(500)
 DIMENSION VODD(500),VEVEN(500),VRIANL(250),VRIANU(250)

5 CONTINUE

READ 100,NQUEUE,NLAG,IN

NLIM = NQUEUE + NLAG

READ 100,R,RHO

IF(R.EQ.0) STOP

IF(R.EQ.1) D=RHO;GO TO 77

CALL DFIND(R,RHO,D)

77 PRINT 107,D,RHO,R

AVE = D/(1.-D)

VRIAN = AVE/(1.-D)

E2MEAN = VRIAN+AVE**2

PRINT 108,AVE,VRIAN

PRINT 102

PRINT 109

PRINT 102

THE PURPOSE OF THE NEXT LOOP IS TO CALCULATE
 $K(J)$,THE PROBABILITY OF J DEPARTURES BETWEEN
 SUCCESSIVE ARRIVALS.

RHOR = R*RHO

RHORP1 = RHOR + 1.

KO = (RHOR/RHORP1)**R

PIO = 1.-D

PROB10 = 1.-KO

IF(PROB10.LT.0) STOP

K(1) = KO*R/RHORP1

PI(1) = PIO*D

PROB1(1) = PROB10-K(1)

IF(PROB1(1).LT.0) STOP

LIM = NLIM-1

DO 10 L=1,LIM

K(L+1) = K(L)*(R+L)/((L+1)*RHORP1)

PI(L+1)=PI(L)*D

PROB1(L+1) = PROB1(L)-K(L+1)

IF(PROB1(L+1).LT.0) PROB1(L+1)=0

10 CONTINUE

THE NEXT STEP IS TO CALCULATE $E(N(1)/N(0)=1)$.

EODDO = KO

```

EODD20 = K0
VODD0 = EODD20 - EODD0**2
VRIANU(1) = P10*EODD0**2
VRIANL(1) = P10*VODD0
DO 20 L = 1,LIM
    EODD(L)=K0*(L+1)
    EODD2(L) = K0*(L+1)**2
    DO 30 M = 1,L
        EODD(L)=EODD(L)+K(L+1-M)*M
        EODD2(L)=EODD2(L)+K(L+1-M)*M**2
30    CONTINUE
    VODD(L)=EODD2(L)-EODD(L)**2
20    CONTINUE
DO 22 I = 1,NQUEUE
    VRIANL(1) = VRIANL(1)+VODD(I)*PI(I)
    VRIANU(1) = VRIANU(1)+PI(I)*EODD(I)**2
22    CONTINUE
    VRIANU(1) = E2MEAN-VRIANU(1)
    N=1
    IF(N/IN*IN.EQ.N) PRINT 101,N,EODD0,(EODD(L),L=1,20)
C-
C    WE NOW ENCODE THE RECURSIVE RELATIONSHIP
C
DO 40 N = 2,NLAG
    LIM = NLIM-N
    IF(N/2*2.NE.N) GO TO 44
    EVEN0 = EODD0*PROB10+EODD(1)*K0
    EVEN20=EODD20*PROB10+EODD2(1)*K0
    VEVEN0=EVEN20-EVEN0**2
    VRIANL(N)=P10*VEVEN0
    VRIANU(N)=P10*EVEN0**2
    IF(LIM.EQ.0) GO TO 40
    DO 51 L = 1,LIM
        EVEN(L)=K0*EODD(L+1)
        EVEN2(L)=K0*EODD2(L+1)
        DO 61 M = 1,L
            EVEN(L)=EVEN(L)+K(L+1-M)*EODD(M)
            EVEN2(L)=EVEN2(L)+K(L+1-M)*EODD2(M)
61    CONTINUE
        EVEN(L)=EVEN(L)+PROB1(L)*EODD0
        EVEN2(L)=EVEN2(L)+PROB1(L)*EODD20
        VEVEN(L)=EVEN2(L)-EVEN(L)**2
51    CONTINUE
    DO 56 I = 1,NQUEUE
        VRIANL(N)=VRIANL(N)+VEVEN(I)*PI(I)
        VRIANU(N)=VRIANU(N)+PI(I)*EVEN(I)**2
56    CONTINUE
    VRIANU(N)=E2MEAN-VRIANU(N)
    IF(N/IN*IN.EQ.N) PRINT 101,N,EVEN0,(EVEN(L),L=1,20)
    GO TO 40
44    EODD0 = EVEN0*PROB10+EVEN(1)*K0

```

```

EODD20=EVEN20*PROB10+EVEN2(1)*K0
VODD0=EODD20-EODD0**2
VRANL(N)=PI0*VODD0
VRANU(N)=PI0*EODD0**2
IF(LIM.EQ.0) GO TO 40.
DO 50 L = 1,LIM
  EODD(L)=K0*EVEN(L+1)
  EODD2(L)=K0*EVEN2(L+1)
DO 60 M=1,L
  EODD(L)=EODD(L)+K(L+1-M)*EVEN(M)
  EODD2(L)=EODD2(L)+K(L+1-M)*EVEN2(M)
60 CONTINUE
  EODD(L)=EODD(L)+PROB1(L)*EVEN0
  EODD2(L)=EODD2(L)+PROB1(L)*EVEN20
  VODD(L)=EODD2(L)-EODD(L)**2
50 CONTINUE
DO 55 I = 1,NQUEUE
  VRANL(N)=VRANL(N)+VODD(I)*PI(I)
  VRANU(N)=VRANU(N)+PI(I)*EODD(I)**2
55 CONTINUE
  VRANU(N)=E2MEAN-VRANU(N)
IF(N/IN*IN.EQ.N) PRINT 101,N,EODD0,(EODD(L),L=1,20)
40 CONTINUE

C PRINT 102
C PRINT 103
DO 70 NN=1,NLAG
  SYSTEM=(VRANU(NN)+AVE+1.)*(1-D)**2
  WAIT=(VRANU(NN)+AVE)*(1-D)**2/(D*(2-D))
  VRANL(NN)=VRANL(NN)/VRAN
  VRANU(NN)=VRANU(NN)/VRAN
  PRINT 104, NN,VRANL(NN),VRANU(NN),WAIT,SYSTEM
70 CONTINUE
PRINT 102
PRINT 102
GO TO 5
100 FORMAT(3G)
101 FORMAT(1X,13,21(1X,F5.2))
102 FORMAT(5X)
C103 FORMAT('LAG NO.',3X,'LOWER',5X,'UPPER',5X,'WAITING',5X
C *,'SYSTEM',/10X,'BOUND',5X,'BOUND',5X,'TIME BD.',4X,'TIME BD.'/)
104 FORMAT(13,5X,F7.5,3X,F7.5,5X,F7.5,5X,F7.5)
105 FORMAT(1X,'SUPPLY NQUEUE AND NLAG')
106 FORMAT(1X,'SUPPLY ERLANG ORDER AND UTILIZATION')
107 FORMAT(1X,'D = ',1G,' RHO = ',1G,' AND R = ',1G)
108 FORMAT(1X,'MEAN QUEUE = ',1G,' AND VARIANCE = ',1G)
109 FORMAT(2X,'N',5X,'0',5X,'1',5X,'2',5X,'3',5X,'4',5X,'5',5X,
C '6',5X,'7',5X,'8',5X,'9',4X,'10',4X,'11',4X,'12',4X,'13',4X,
C '14',4X,'15',4X,'16',4X,'17',4X,'18',4X,'19',4X,'20')
END

```

C

```

SUBROUTINE DFIND(R,RHO,D)
DOUBLE PRECISION XLOW,XHI,FLOW,FD,D,RHO,PROD,EPS
INTEGER R
EPS = .0000000000000001
D = 0.5
XLOW = 0
FLOW = -1
XHI = 1
77 CALL FEVAL(R,RHO,D,FD)
IF(DABS(FD).LT.EPS) RETURN
PROD = FLOW*FD
IF(PROD.GT.0) GO TO 78
XHI = D
GO TO 79
78 XLOW = D
FLOW = FD
79 D = (XLOW+XHI)/2.
GO TO 77
END

```

C

```

SUBROUTINE FEVAL(R,RHO,D,FD)
DOUBLE PRECISION RHO,D,FD,MESS
INTEGER R
MESS = (R*RHO/(R*RHO+1.-D))**R
FD = D-MESS
RETURN
END

```

GI/M/m PREDICTION PROGRAM

```

IMPLICIT DOUBLE PRECISION(A-H,O-Z)
INTEGER R,V
DIMENSION P(61,61),PO(61),FA(61),PSUM(61)
DIMENSION EODD(61),EODDG(61),EODDS(61),PROBO(61)
DIMENSION EVEN(61),EVENG(61),EVENS(61),PROBE(61)
DIMENSION EODDW(61),EVENW(61)
DIMENSION E2OG(61),E2OS(61),E2EG(61),E2ES(61)
DIMENSION E2O(61),E2E(61),PI(200)
DIMENSION PHI(10),PHIC(10),C(10),TU(10)
TOL = 0.0001

C
C   THE MAXIMUM QUEUE LENGTH, MAXIMUM PREDICTION
C   HORIZON, AND THE NUMBER OF SERVERS ARE SPECIFIED,
C   AND RELATED CONSTANTS ARE CALCULATED.
C

READ 100,NQUEUE,NLIM,M
ML1 = M-1
Z=M
MM1=M-1
C1=2*M-3
C2=MM1*(M-2)

C
C   THE ORDER OF THE ERLANG DISTRIBUTION FOR I.A.T.'S,
C   THE UTILIZATION, AND THE SERVICE RATE ARE
C   SPECIFIED. SEVERAL RELATED TERMS TO BE USED LATER
C   ARE CALCULATED. AN ARRAY OF FACTORIALS IS SET UP.
C

999 READ 100,R,RHO,SERV
IND = NQUEUE+NLIM
IF(RHO.EQ.0) STOP
RHOM = M*RHO
RHR = R*RHOM
Q = 1./(R*RHO+1)
QC = 1.-Q
W=M*SERV
N = 0
FA(1) = 1
DO 7 I = 2,53
  FA(I) = FA(I-1)*I
7 CONTINUE

C
C   THIS SECTION COMPUTES THE TRANSITION PROBABILITIES P(I,J)
C   FOR I=0,...,M-1; J=0,...,I+1 IN ACCORDANCE WITH (4.28).
C

P01=(RHR/(1.+RHR))*R
P00=1.-P01
I=0
PSUM0 = P00+P01
PRINT 103,I,PSUM0,P00,P01
DO 1 I = 1,ML1
  IP1 = I+1

```

```

SUM=1./ (FA(IP1)*RHR**R)+(-1)**IP1/(FA(IP1)*(IP1+RHR)**R)
DO 2 K = 1,I
  SUM=SUM+(-1)**K/(FA(IP1-K)*FA(K)*(K+RHR)**R)
2 CONTINUE
PO(I) = FA(IP1)*RHR**R*SUM
PSUM(I) = PO(I)
DO 3 L = 1,IP1
  IP1L=IP1-L
  SUM=1./ (IP1+RHR)**R
  IF (IP1L.GT.0) SUM=SUM*(-1)**IP1L/FA(IP1L)
C  +1./ (FA(IP1L)*(L+RHR)**R)
  IML = I-L
  IF (IML.LE.0) GO TO 44
  DO 4 K = 1,IML
    SUM=SUM+(-1)**K/(FA(IP1L-K)*FA(K)*(K+L+RHR)**R)
4 CONTINUE
44 CONTINUE
P(I,L)=FA(IP1)*RHR**R*SUM/FA(L)
PSUM(I)=PSUM(I)+P(I,L)
3 CONTINUE
IP2=IP1+1
DO 61 L=IP2,50
  P(I,L)=0.
61 CONTINUE
PRINT 103,I,PSUM(I),PO(I),(P(I,L),L=1,9)
1 CONTINUE
C
C THE NEXT SECTION FINDS P(I,J) FOR I=M,...;J=0,...,M-1 IN
C ACCORDANCE WITH (4.28).
C
DO 91 I = M,IND
  IMM=I-M
  IMM1=IMM+1
  IF (R.EQ.1) PO(I)=(-1)**M*QC*Q**IMM1;GO TO 901
  PO(I)=(-1)**M*(QC)**R*Q**IMM1*FA(IMM1+R-1)/(FA(IMM1)*FA(R-1))
901 CONTINUE
  SUMK=0
  DO 92 K=1,M
    SUMJ=1.0
    IF (IMM.EQ.0) GO TO 93
    RATJ=K/(M+RHR)
    TERMJ=1.0
    DO 94 J=1,IMM
      TERMJ=TERMJ*RATJ*(J+R-1)/J
      SUMJ=SUMJ+TERMJ
94 CONTINUE
93 CONTINUE
    TERMK=((RHR/(RHR+M-K))**R-(QC)**R*SUMJ)/FA(K)
    IF (K.EQ.M) GO TO 81
    TERMK=TERMK*(-1)**(M-K)*(Z/K)**IMM1/FA(M-K)
81 CONTINUE

```

```

SUMK=SUMK+TERMK
92  CONTINUE
    IF( IMM.EQ.0) PO(I)=PO(I)+SUMK*FA(M); GO TO 83
    PO(I)=PO(I)+SUMK*FA(M)
83  CONTINUE
    IF( PO(I).GT.PO(I-1).AND.PO(I-1).LT.TOL) PO(I)=0.
    IF( PO(I).LT.TOL) PO(I)=0.
    PSUM(I)=PO(I)
    DO 95 L = 1, MM1
      MML=M-L
      IF( R.EQ.1) P(I,L)=FA(M)*(-1)**MML*QC*Q**IMM1/
        (FA(L)*FA(MML)); GO TO 900
      P(I,L)=FA(M)*FA(IMM1+R-1)*(-1)**MML*(QC)**R*Q**IMM1/
        (FA(L)*FA(MML)*FA(IMM1)*FA(R-1))
900 CONTINUE
      SUMK=0
      DO 96 K = 1, MML
        SUMJ=1.0
        IF( IMM.EQ.0) GO TO 97
        RATJ=K/(M+RHR)
        TERMJ=1.0
        DO 98 J=1, IMM
          TERMJ=TERMJ*RATJ*(J+R-1)/J
          SUMJ=SUMJ+TERMJ
98  CONTINUE
97  CONTINUE
        TERMK=((RHR/(RHR+M-K))**R-(QC)**R*SUMJ)/FA(K)
        TERMK=TERMK*(Z/K)**IMM1
        IF( K.EQ.MML) GO TO 82
        TERMK=TERMK*(-1)**(MML-K)/FA(MML-K)
82  CONTINUE
      SUMK=SUMK+TERMK
96  CONTINUE
      IF( IMM.EQ.0) P(I,L)=P(I,L)+SUMK*FA(M)/FA(L);
      GO TO 84
      P(I,L)=P(I,L)+SUMK*(FA(M)/FA(L))
84  CONTINUE
      IF( P(I,L).GT.P(I-1,L).AND.P(I-1,L).LT.TOL) P(I,L)=0.
      IF( P(I,L).LT.TOL) P(I,L)=0.
      PSUM(I)=PSUM(I)+P(I,L)
95  CONTINUE
C
C
C
C
THE FOLLOWING SEGMENT COMPUTES P(I,J) FOR I=M,...;
J=M,...,I+1 IN ACCORDANCE WITH (4.28).
P(I,I+1)=(QC)**R
PSUM(I)=PSUM(I)+P(I,I+1)
DO 85 L = 1, IMM1
  I1L=I+1-L
  P(I,I1L)=P(I,I1L+1)*Q*(R-1+L)/L
C
  IF( P(I,I1L).LT.TOL) P(I,I1L)=0.

```



```

      PSUM(I)=PSUM(I)+P(I,I1L)
85  CONTINUE
      IP2=I+2
      IF(IP2.GT.50) GO TO 63
      DO 62 L=IP2,50
        P(I,L)=0.
62  CONTINUE
63  CONTINUE
      IF(I.LE.9)PRINT 103,I,PSUM(I),PO(I),(P(I,L),L=1,9)
91  CONTINUE
C
C  THIS SECTION FINDS THE ROOT D(M) OF EQUATION (2.17)
C  WHICH IS USED FOR THE CALCULATION OF THE STEADY STATE
C  PROBABILITIES PI(I).
C
      EPS=0.00000001
      D = 0.5
      XLOW = 0.
      FLOW = -1.
      XHI = 1.
777  FUN = (R*RHO/(R*RHO+1.-D))**R
      FD = D-FUN
      IF(DABS(FD).LT.EPS) GO TO 770
      PROD = FLOW*FD
      IF(PROD.GT.0) GO TO 778
      XHI = D
      GO TO 779
778  XLOW = D
      FLOW = FD
779  D = (XLOW+XHI)/2.
      GO TO 777
770  OUTPUT D
C
C  THE STEADY STATE PROBABILITIES ARE NOW CALCULATED USING
C  EQUATIONS (2.19) AND (2.20).
C
      PHI(1)=(RHR/(RHR+1))**R
      PHIC(1)=1.-PHI(1)
      C(1)=PHI(1)/PHIC(1)
      DO 201 I=2,M
        PHI(I)=(RHR/(RHR+I))**R
        PHIC(I)=1.-PHI(I)
        C(I)=C(I-1)*PHI(I)/PHIC(I)
201  CONTINUE
      TU(M-1)=PHI(M)/(C(M)*PHIC(M)*D)
      DO 202 J=2,MM1
        V=M-J+1
        TU(V-1)=TU(V)+FA(M)*(M*PHIC(V)-V)/
          (FA(M-V)*FA(V)*C(V)*(M*(1.-D)-V)*PHIC(V))
202  CONTINUE
      TU0=TU(1)+M*(M*PHIC(1)-1)/(C(1)*PHIC(1)*(M*(1.-D)-1))

```

```

      A = (1./((TU0+1./(1.-D)))
      PI(M) = A
      PIO = A*TU0
      DO 203 I=1,MM1
        PIO=PIO+(-1)**I*A*C(I)*TU(I)
203    CONTINUE
      PISUM=PIO+PI(M)
      DO 204 I=1,MM1
        PI(I)=A*C(I)*TU(I)
        IF(I.EQ.MM1) GO TO 207
        IP1=I+1
        DO 205 J=IP1,MM1
          PI(I)=PI(I)+(-1)**(J-I)*FA(J)*A*C(J)*TU(J)/
C      (FA(J-I)*FA(I))
205    CONTINUE
207    PISUM=PISUM+PI(I)
204    CONTINUE
      MP1=M+1
      DO 206 I=MP1,200
        PI(I)=A*D**(I-M)
        PISUM=PISUM+PI(I)
206    CONTINUE
      OUTPUT PISUM

C
C   THE NEXT SECTION CALCULATES THE MEAN AND VARIANCE OF THE
C   QUEUE LENGTH AND WAITING TIME PROCESSES. AS WELL, SEVERAL
C   RESTRICTED MOMENTS AND THE PROBABILITY OF ALL SERVERS
C   BEING BUSY ARE COMPUTED.
C
      EN=0
      EN2=0
      ENG=0
      EN2G=0
      DO 208 I=1,MM1
        EN=EN+PI(I)*I
        EN2=EN2+PI(I)*I*I
208    CONTINUE
      DO 209 I=M,200
        ENG=ENG+PI(I)*I
        EN2G=EN2G+PI(I)*I*I
209    CONTINUE
      EN=EN+ENG
      EN2=EN2+EN2G
      VARN=EN2-EN**2
      PROBG=A/(1.-D)
      EW=PROBG/(W*(1.-D))
      VARW=PROBG*(2.-PROBG)/(W*(1.-D))**2
      OUTPUT EN, VARN, PROBG, EW, VARW
      WUP=(EN2G-C1*ENG+C2*PROBG)/W**2
      PRINT 102

```

C

C THE NEXT TASK IS THE INITIALIZATION OF THE CONDITIONAL
C MOMENTS IN ACCORDANCE WITH THE REMARKS PRECEDING (4.13).
C

```

EVENGO=0
E2EGO=0
EVENSO=0
E2ESO=0
PROBEO=0
DO 86 I=1,MM1
    EVENG(I)=0
    E2EG(I)=0
    EVENS(I)=I
    E2ES(I)=I*I
    PROBE(I)=0
86 CONTINUE
DO 87 I=M,IND
    EVENG(I)=I
    E2EG(I)=I*I
    EVENS(I)=0
    E2ES(I)=0
    PROBE(I)=1.0
87 CONTINUE

```

C THE ALGORITHM NOW PROCEEDS BY CALCULATING THE NEW
C EXPECTATIONS IN THE 'ODD' ARRAYS FROM THE OLD ONES IN
C THE 'EVEN' ARRAYS. THE EQUATIONS USED HERE ARE (4.7) TO
C (4.11), (4.13), (4.14) AND (4.17) TO FIND THE MOMENTS, AND
C (4.15), (4.16), (4.21) AND (4.22) TO FIND THE BOUNDS. THE
C NORMALIZED BOUNDS RATL, RATU, RATLW, AND RATUW ARE FOUND AS
C WELL.
C

```

99 N = N+1
IND = IND-1
EODDGO = P00*EVENGO+P01*EVENG(1)
E2OGO = P00*E2EGO+P01*E2EG(1)
EODDSO = P00*EVENSO+P01*EVENS(1)
E2OSO = P00*E2ESO+P01*E2ES(1)
PROB00 = P00*PROBEO+P01*PROBE(1)
EODDO = EODDGO+EODDSO
E2OO = E2OGO+E2OSO
BDL = P10*(E2OO-EODDO**2)
BDU = EN2-P10*EODDO**2
EODDWO=(EODDGO-MM1*PROB00)/W
BDLW=P10*(E2OGO-C1*EODDGO+C2*PROB00)/W**2
BDLW=BDLW-P10*EODDWO**2
BDUW=WUP-P10*EODDWO**2
DO 11 I = 1,IND
    EODDG(I)=P0(I)*EVENGO
    E2OG(I)=P0(I)*E2EGO
    EODDS(I)=P0(I)*EVENSO
    E2OS(I)=P0(I)*E2ESO

```

```

PROBO(I)=PO(I)*PROBE0
IP1=I+1
DO 12 L = 1, IP1
  EODDG(I)=EODDG(I)+P(I,L)*EVENG(L)
  E2OG(I)=E2OG(I)+P(I,L)*E2EG(L)
  EODDS(I)=EODDS(I)+P(I,L)*EVEN(S(L)
  E2OS(I)=E2OS(I)+P(I,L)*E2ES(L)
  PROBO(I)=PROBO(I)+P(I,L)*PROBE(L)
12 CONTINUE
EODD(I) = EODDG(I)+EODDS(I)
E2O(I)=E2OG(I)+E2OS(I)
BDL=BDL+PI(I)*(E2O(I)-EODD(I)**2)
BDU=BDU-PI(I)*EODD(I)**2
EODDW(I)=(EODDG(I)-MM1*PROBO(I))/W
BDLW=BDLW+PI(I)*(E2OG(I)-C1*EODDG(I)+C2*PROBO(I))/W**2
BDLW=BDLW-PI(I)*EODDW(I)**2
BDUW=BDUW-PI(I)*EODDW(I)**2
11 CONTINUE
RATL=BDL/VARN
RATU=BDU/VARN
RATLW=BDLW/VARW
RATUW=BDUW/VARW
PRINT 101,N,EODDO,(EODD(L),L=1,9)
PRINT 101,N,EODDWO,(EODDW(L),L=1,9)
PRINT 106,N,BDL,BDU,RATL,RATU
PRINT 106,N,BDLW,BDUW,RATLW,RATUW
PRINT 102

```

C
C
C
C
C
C
THE ROLES OF THE ODD AND EVEN ARRAYS ARE NOW REVERSED:
NEW EXPECTATIONS ARE FOUND IN THE 'EVEN' ARRAY FROM
THE PREVIOUS VALUES IN THE 'ODD' ARRAY. THE SAME
EQUATIONS ARE USED.

```

N = N+1
IND = IND-1
EVENG0 = P00*EODDGO+P01*EODDG(1)
E2EGO = P00*E2OGO+P01*E2OG(1)
EVENSO = P00*EODDSO+P01*EODDS(1)
E2ESO = P00*E2OSO+P01*E2OS(1)
PROBE0 = P00*PROBO0+P01*PROBO(1)
EVENQ = EVENG0+EVENSO
E2EQ = E2EGO+E2ESO
BDL=PIO*(E2EQ-EVENQ**2)
BDU=EN2-PIO*EVENQ**2
EVENWO=(EVENG0-MM1*PROBE0)/W
BDLW=PIO*(E2EGO-C1*EVENG0+C2*PROBE0)/W**2
BDLW=BDLW-PIO*EVENWO**2
BDUW=WUP-PIO*EVENWO**2
DO 21 I = 1, IND
  EVENG(I)=PO(I)*EODDGO
  E2EG(I)=PO(I)*E2OGO

```

```

EVEN(I)=PO(I)*EODDSO
E2ES(I)=PO(I)*E2OSO
PROBE(I)=PO(I)*PROB00
IP1=I+1
DO 22 L = 1, IP1
    EVENG(I)=EVENG(I)+P(I,L)*EODDG(L)
    E2EG(I)=E2EG(I)+P(I,L)*E2OG(L)
    EVENS(I)=EVENS(I)+P(I,L)*EODDS(L)
    E2ES(I)=E2ES(I)+P(I,L)*E2OS(L)
    PROBE(I)=PROBE(I)+P(I,L)*PROB0(L)
22 CONTINUE
EVEN(I) = EVENG(I)+EVENS(I)
E2E(I)=E2EG(I)+E2ES(I)
BDL=BDL+PI(I)*(E2E(I)-EVEN(I)**2)
BDU=BDU-PI(I)*EVEN(I)**2
EVENW(I)=(EVENG(I)-MM1*PROBE(I))/W
BDLW=BDLW+PI(I)*(E2EG(I)-C1*EVENG(I)+C2*PROBE(I))/W**2
BDLW=BDLW-PI(I)*EVENW(I)**2
BDUW=BDUW-PI(I)*EVENW(I)**2
21 CONTINUE
RATL=BDL/VARN
RATU=BDU/VARN
RATLW=BDLW/VARW
RATUW=BDUW/VARW
PRINT 101,N,EVENO,(EVEN(L),L=1,9)
PRINT 101,N,EVENWO,(EVENW(L),L=1,9)
PRINT 106,N,BDL,BDU,RATL,RATU
PRINT 106,N,BDLW,BDUW,RATLW,RATUW
PRINT 102

IF(N.GE.NLIM) GO TO 999
GO TO 99
100 FORMAT(3G)
101 FORMAT(1X,I3,11(1X,F5.2))
102 FORMAT(5X)
103 FORMAT(1X,I3,1X,F5.3,10(1X,F5.4))
C104 FORMAT(1X,I3,11(1X,F5.2),/,4X,11(1X,F5.2))
106 FORMAT(1X,I3,2(1X,F10.5),2(1X,F7.5))
END

```

M/G/1 PREDICTION PROGRAM

```

IMPLICIT DOUBLE PRECISION(A-H,K,O-Z)
DIMENSION K(50),KSUM(50),EVEN(50),EODD(50)
DIMENSION ERR(50),ES(50),FA(50)
DIMENSION PI(50),PISUM(50),EODD2(50),EVEN2(50)

```

```

C
C THE PREDICTION HORIZON AND MAXIMUM QUEUE LENGTH ARE
C CALLED FOR. A FACTORIAL ARRAY IS SET UP.
C

```

```

NQUEUE = 50
FA(1) = 1.
DO 7 I = 2,50
    FA(I) = FA(I-1)*I
7 CONTINUE

```

```

C
C EXP(RHO) IS NOW FOUND, FOLLOWED BY THE MEAN AND VARIANCE
C OF THE QUEUE LENGTH PROCESS. THE AVERAGE SYSTEM TIME IS
C FOUND USING LITTLE'S FORMULA, AND EXPRESSED AS A MULTIPLE
C OF THE MEAN SERVICE TIME.
C

```

```

6 READ 100,NLIM,RHO,TOL
TERM = 1.
ERHO = 1.
DO 1 I = 1,50
    TERM = TERM*RHO/I
    ERHO = ERHO + TERM

```

```

1 CONTINUE
OUTPUT ERHO
IF(NLIM.LE.0) STOP
RHOC = 1.-RHO
RHOSQ = RHO**2
EN = RHO + 0.5*RHOSQ/(1.-RHO)
KEW = (RHO**3/3 - RHOSQ)/RHOC + 2*((RHO - RHOSQ/2)/RHOC)**2
VARN = KEW + EN*(1.-EN)
ESYS = EN/RHO
OUTPUT EN,VARN,ESYS
X2L2 = RHOSQ

```

```

C
C THE STATIONARY AND TRANSITION PROBABILITIES PI AND K
C ARE NOW CALCULATED. DUE TO NUMERICAL PROBLEMS THE
C DISTRIBUTION FOR THE FORMER IS TRUNCATED ONCE THE
C PROBABILITY HAS FALLEN BELOW A SPECIFIED TOLERANCE.
C THE INDEX AT WHICH THIS OCCURS IS PRINTED.
C

```

```

PIO = 1.-RHO
PI(1) = PIO*(ERHO-1.)
PISUM(1) = PIO + PI(1)
KO = 1./ERHO
K(1) = KO*RHO
KSUM(1) = KO + K(1)
DO 10 I = 2,50

```

```

      IM1=I-1
      SUMJ=1.0
      DO 11 J = 1, IM1
        SUMJ=SUMJ+(-J*RHO/ERHO)**(I-J)/
C      FA(I-J)*(1.+(I-J)/(J*RHO))
11    CONTINUE
      PI(I) = PIO*SUMJ*ERHO**I
      IF(DABS(PI(I)).LT.TOL) NO=I;GO TO 2
      PISUM(I) = PISUM(IM1)+PI(I)
      K(I)=K(I-1)*RHO/I
      KSUM(I)=KSUM(I-1)+K(I)
10    CONTINUE
      GO TO 4
      2    OUTPUT NO
      DO 3 I=NO,NQUEUE
        PI(I)=0.
        K(I)=K(I-1)*RHO/I
        KSUM(I)=KSUM(I-1)+K(I)
        PISUM(I)=PISUM(I-1)
      3    CONTINUE
      4    CONTINUE
      OUTPUT PISUM(50),KSUM(50)

C
C   THE ALGORITHM NOW COMPUTES THE ONE-LAG MOMENTS USING
C   (5.30) AND (5.31) FOR THE QUEUE LENGTH PROCESS.
C   EQUATION (5.35) IS USED TO PRODUCE THE SYSTEM TIME
C   PREDICTIONS,WHICH ARE EXPRESSED AS MULTIPLES
C   OF THE MEAN SERVICE TIME.
C

      N = 1
      EODDO = RHO
      EODD(1) = RHO
      ES(1) = PI(2)/PI(1)+RHO
      DO 20 I = 2,50
        EODD(I) = I-1+RHO
        EODD2(I)=(I-1)**2+(2*I-1)*RHO+X2L2
        IF(I.GT.12)GO TO 20
        ES(I)=I*PI(I+1)/PI(I)+RHO
20    CONTINUE
      EODD2(1)=RHO+RHOSQ-
      EODD20 = EODD2(1)
      ERR(1)=(RHO+RHOSQ)*(1.-(PIO)+PIO*(EODD20-EODDO**2))
      DO 51 I=1,NQUEUE
        ES(I)=ES(I)/RHO
51    CONTINUE
      PRINT 101,N,(EODD(L),L=1,10)
      PRINT 101,N,(ES(L),L=1,10)
      PRINT 102

C
C   THE TWO-LAG VALUES ARE NOW COMPUTED.

```

```

C
N = 2
EVENO = N*RHO-1+KO
EVEN(1) = EVENO
ES(1) = EVENO
EVEN2(1)=1+2*(-RHO+RHOSQ+X2L2)+KO*(EODD20-1+RHO-X2L2)
EVEN20 = EVEN2(1)
DO 30 I = N,50
  EVEN(I) = (I-N)+RHO*N
  EVEN2(I)=(I-N)**2+N*((2*(I-N)+1)*RHO+(N-1)*RHOSQ+X2L2)
  IF(I.GT.12) GO TO 30
  ES(I)=(I-1)*PI(I+1)/PI(I)+2*RHO
30 CONTINUE
ERR(N)=N*(RHO+RHOSQ)*(1.-PISUM(N-1))
ERR(N)=ERR(N)+PIO*(EVEN20-EVENO**2)
ERR(N)=ERR(N)+PI(1)*(EVEN2(1)-EVEN(1)**2)
DO 52 I=1,NQUEUE
  ES(I)=ES(I)/RHO
52 CONTINUE
PRINT 101,N,(EVEN(L),L=1,10)
PRINT 101,N,(ES(L),L=1,10)
PRINT 102

```

C
C
C
C
THE RECURSION NOW STARTS BY COMPUTING THE NEW MOMENTS IN
THE ODD ARRAYS FROM THE OLD MOMENTS IN THE EVEN ARRAYS.

```

N=3
99 NM1 = N-1
   NM2 = N-2
   SUM = KO*EVENO
   SUM2=KO*(EVEN20-(1-N)**2
C   +NM1*((2*N-3)*RHO-NM2*RHOSQ-X2L2))
   DO 40 L = 1,NM2
     SUM=SUM+K(L)*(EVEN(L)-L)
     SUM2=SUM2+K(L)*(EVEN2(L)-(N-L-1)**2
C   +NM1*((2*(N-L-1)-1)*RHO-NM2*RHOSQ-X2L2))
40 CONTINUE
EODD0=RHO-NM1*(1.-RHO)*(1.-KSUM(NM2)) +SUM
EODD(1) = EODD0
ES(1) = EODD(1)
EODD2(1)=(1-N)**2+N*((2*(1-N)+1)*RHO+NM1*RHOSQ+X2L2)
EODD2(1)=EODD2(1)+SUM2
EODD20=EODD2(1)
IF(N.EQ.3) GO TO 44
DO 41 I = 2,NM2
  SUM = KO*(EVEN(I-1)-I+1)
  SUM2=KO*(EVEN2(I-1)-(I-N)**2
C   +NM1*((2*(N-I)-1)*RHO-NM2*RHOSQ-X2L2))
  EODD(I)=I-1+RHO-NM1*(1.-RHO)*(1.-KSUM(NM1-I))
  EODD2(I)=(I-N)**2+N*((2*(I-N)+1)*RHO+NM1*RHOSQ+X2L2)

```



```

      NMI1 = NM1-I
      DO 42 L = 1, NMI1
        SUM = SUM + K(L) * (EVEN(L+I-1) - (L+I-1))
        SUM2 = SUM2 + K(L) * (EVEN2(L+I-1) - (N-L-I)**2
C      + NM1 * ((2*(N-L-I)-1)*RHO - NM2*RHOSQ - X2L2))
42    CONTINUE
      EODD(I) = EODD(I) + SUM
      ES(I) = EODD(I)
      EODD2(I) = EODD2(I) + SUM2
41    CONTINUE
44    EODD(NM1) = NM2 + RHO - NM1 * (1.-RHO) * (1.-KO) + KO * (EVEN(NM2) - NM2)
      ES(NM1) = EODD(NM1)
      EODD2(NM1) = 1 + N * (-RHO + NM1*RHOSQ + X2L2)
C    + KO * (EVEN2(NM2) - 1 - NM1 * (-RHO + NM2*RHOSQ + X2L2))
      DO 43 I = N, NQUEUE
        EODD(I) = I - N + N*RHO
        EODD2(I) = (I-N)**2 + N * ((2*(I-N)+1)*RHO + NM1*RHOSQ + X2L2)
        IF(I.GT.12) GO TO 43
        ES(I) = (I-N+1)*PI(I+1)/PI(I) + N*RHO
43    CONTINUE
      ERR(N) = N * (RHO + RHOSQ) * (1.-PI(SUM(NM1)))
      ERR(N) = ERR(N) + PIO * (EODD20 - EODD0**2)
      DO 90 I = 1, NM1
        ERR(N) = ERR(N) + PI(I) * (EODD2(I) - EODD(I)**2)
90    CONTINUE
      PRINT 101, N, (EODD(L), L=1, 10)
      DO 53 I = 1, NQUEUE
        ES(I) = ES(I)/RHO
53    CONTINUE
      PRINT 101, N, (ES(L), L=1, 10)
      PRINT 102

```

C THE ROLES ARE NOW REVERSED AND THE NEW MOMENTS ARE NOW
 C COMPUTED IN THE EVEN ARRAYS FROM THE PREVIOUS DATA IN
 C THE ODD ARRAYS.
 C

```

      N = N+1
      NM1 = N-1
      NM2 = N-2
      SUM = KO * EODD0
      SUM2 = KO * (EODD20 - (1-N)**2
C    + NM1 * ((2*N-3)*RHO - NM2*RHOSQ - X2L2))
      DO 60 L = 1, NM2
        SUM = SUM + K(L) * (EODD(L) - L)
        SUM2 = SUM2 + K(L) * (EODD2(L) - (N-L-1)**2
C    + NM1 * ((2*(N-L-1)-1)*RHO - NM2*RHOSQ - X2L2))
60    CONTINUE
      EVEN0 = RHO - NM1 * (1.-RHO) * (1.-KSUM(NM2)) + SUM
      EVEN(1) = EVEN0
      ES(1) = EVEN(1)

```

```

EVEN2(I) = (1-N)**2 + N*((2*(1-N)+1)*RHO + NM1*RHOSQ + X2L2)
EVEN2(I) = EVEN2(I) + SUM2
EVEN20 = EVEN2(I)

```

```

DO 61 I = 2, NM2

```

```

    SUM = K0*(EODD(I-1) - I + 1)

```

```

    SUM2 = K0*(EODD2(I-1) - (I-N)**2

```

```

    + NM1*((2*(I-1)-1)*RHO - NM2*RHOSQ - X2L2))

```

```

    EVEN(I) = I - 1 + RHO - NM1*(1 - RHO)*(1 - KSUM(NM1-I))

```

```

    EVEN2(I) = (I-N)**2 + N*((2*(I-N)+1)*RHO + NM1*RHOSQ + X2L2)

```

```

    NM1I = NM1 - I

```

```

DO 62 L = 1, NM1I

```

```

    SUM = SUM + K(L)*(EODD(L+I-1) - (L+I-1))

```

```

    SUM2 = SUM2 + K(L)*(EODD2(L+I-1) - (N-L-I)**2

```

```

    + NM1*((2*(N-L-I)-1)*RHO - NM2*RHOSQ - X2L2))

```

```

62 CONTINUE

```

```

    EVEN(I) = -EVEN(I) + SUM

```

```

    ES(I) = EVEN(I)

```

```

    EVEN2(I) = EVEN2(I) + SUM2

```

```

61 CONTINUE

```

```

    EVEN(NM1) = NM2 + RHO - NM1*(1 - RHO)*(1 - K0) + K0*(EODD(NM2) - NM2)

```

```

    ES(NM1) = EVEN(NM1)

```

```

    EVEN2(NM1) = 1 + N*(-RHO + NM1*RHOSQ + X2L2)

```

```

    + K0*(EODD2(NM2) - 1 - NM1*(-RHO + NM2*RHOSQ + X2L2))

```

```

DO 63 I = N, NQUEUE

```

```

    EVEN(I) = I - N + N*RHO

```

```

    EVEN2(I) = (I-N)**2 + N*((2*(I-N)+1)*RHO + NM1*RHOSQ + X2L2)

```

```

    IF(I.GT.12) GO TO 63

```

```

    ES(I) = (I-N+1)*PI(I+1)/PI(I) + N*RHO

```

```

63 CONTINUE

```

```

    ERR(N) = N*(RHO + RHOSQ)*(1 - PISUM(NM1))

```

```

    ERR(N) = ERR(N) + PIO*(EVEN20 - EVEN0**2)

```

```

DO 91 I = 1, NM1

```

```

    ERR(N) = ERR(N) + PI(I)*(EVEN2(I) - EVEN(I)**2)

```

```

91 CONTINUE

```

```

    PRINT 101, N, (EVEN(L), L=1, 10)

```

```

DO 54 I = 1, NQUEUE

```

```

    ES(I) = ES(I)/RHO

```

```

54 CONTINUE

```

```

    PRINT 101, N, (ES(L), L=1, 10)

```

```

    PRINT 102

```

```

    IF(N.GE.NLIM) GO TO 6

```

```

    N = N+1

```

```

    GO TO 99

```

```

100 FORMAT(3G)

```

```

101 FORMAT(1X, I3, 10(1X, F5.2))

```

```

102 FORMAT(5X)

```

```

END

```

- [14] A.G. Pakes, "The Correlation Coefficients of the Queue Lengths of Some Stationary Single Server Queues". Journal of the Australian Mathematical Society, Vol. 12, pp. 35-46 (1971).
- [15] A.G. Pakes, "The Serial Correlation Coefficients of Waiting Times in the Stationary GI/M/1 Queue". Annals of Mathematical Statistics, Vol. 42, pp. 1727-1734 (1971).
- [16] A. Papoulis, "Probability Random Variables, and Stochastic Processes". McGraw-Hill (1965).
- [17] E. Parzen, "Stochastic Processes". Holden-Day (1962).
- [18] N.V. Brabhu, "Queues and Inventories". Wiley (N.Y.) (1965).
- [19] E. Reich, "Waiting Times When Queues Are In Tandem", Ann. Math. Stat., Vol. 28, pp. 768-773 (1957).
- [20] J. F. Reynolds, "The covariance structure of queues and related processes - a survey of recent work". Adv. Appl. Prob., v. 7, pp. 383-415 (1975).
- [21] E. Roth, "An Investigation of the Transient Behaviour of Stationary Queueing Systems". Doctoral Thesis, Sloan School of Management, M.I.T. (1981).
- [22] M. Schwartz, "Computer Communication Network Design and Analysis". Prentice-Hall (1977).
- [23] D.A. Staford, "A Study of Correlation in GI/M/1 Queues", Master's Thesis, Carleton University (1978).
- [24] L.J. Takacs, "Introduction to the Theory of Queues", Oxford University Press (1962).

Bibliography

- [1] N. Blomqvist, "The Covariance Function of the M/G/1 Queueing System". Skand. Akt., pp. 157-174 (1967).
- [2] W. Chou, A. Bragg, & A. Nilsson, "The Need for Adaptive Routing in the Chaotic and Unbalanced Traffic Environment". IEEE Trans. on Comm., Vol. COM-29, pp.481-490 (1981).
- [3] D.J. Daley, "Monte Carlo Estimation of the Mean Queue Size in a Stationary GI/M/1 Queue". Operations Research, vol. 16, pp.1002-1005 (1968).
- [4] D.J. Daley, "The Serial Correlation Coefficients of Waiting Times in a Stationary Single Server Queue". Journal of the Australian Mathematical Society, vol. 8, pp. 683-699 (1968).
- [5] D. Gross & C. Harris, "Fundamentals of Queueing Theory". Wiley (1974).
- [6] L. Kleinrock, "Queueing Systems, Volume 1: Theory". J. Wiley and Sons (1975).
- [7] T. Lewis and P. Odell, "Estimation in Linear Models". Prentice-Hall (1971).
- [8] J. Little, "A Proof for the Queueing Formula $L=\lambda W$ ". Oper. Res., vol.9, pp. 383-387 (1961).
- [9] B. Melamed, "Sojourn Times in Queueing Networks". Research Report, Dept. of Ind. Eng. & Man. Sci., Northwestern University (1979).
- [10] J. McQuillan, I. Richer, & E. Rosen, "The New Routing Algorithm for the ARPANET". IEEE Trans. on Comm., Vol. COM-28, pp. 711-719 (1980).
- [11] B. Pagurek and C. M. Woodside, "A Recursive Algorithm for Computing Serial Correlations of Times in a M/G/1 Queue". TIMS Studies in the Man. Sciences, #3, North Holland Pub. Co., pp. 161-175 (1977).
- [12] B. Pagurek and C.M. Woodside, "An Algorithm for Computing Serial Correlations of Times in GI/G/1 Queues with Rational Arrival Processes", Management Science, Vol. 25, pp. 54-63 (1979).
- [13] B. Pagurek, personal communication (1977).

END

07H09H82

FIN