STATE DEPENDENT SERVER SCHEDULING RULES IN POLLING SYSTEMS

By
YAVUZ GÜNALAY

A Dissertation
Submitted to the School of Graduate Studies in
Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy

McMaster University

STATE DEPENDENT SERVER SCHEDULING RULES IN POLLING SYSTEMS

DOCTOR OF PHILOSOPHY (1996) McMaster University
(Management Science and Information Systems) Hamilton, Ontario

TITLE: State Dependent Server Scheduling Rules in Polling Systems

AUTHOR: Yavuz Günalay
M.Sc. Bilkent University
B.Sc. Middle-East Technical University

SUPERVISORY COMMITTEE: Diwakar Gupta (Chairman)
Robert F. Love
Terry D. Todd

NUMBER OF PAGES: xi. 139

Abstract

A polling system is a cyclic queueing model with multiple customer classes and a single server. Each customer class has its own queue (station). After the server switches to a station, it serves customers waiting at that station according to a specified service regime, e.g., exhaustive, gated or globally gated. It then moves to the next station, following a strict cyclic order. These models have several application areas including computer and communication networks and multi-item production systems. For example, a Local Area Network (LAN) can be modeled as a polling system by defining the central processing unit as the server and the data transmission requests from each terminal as customers. Similarly, a multi-item production system can be modeled as a polling system by considering the flexible machining cell as the server and each product type as a different customer class. In most systems that polling models are used to represent, the server requires time to switch and/or setup before it may start serving a different customer class. These processes (switching/setup) may take considerable amounts of time, and when that happens, it is undesirable to setup for a product type if there are no (or only a few) jobs of that type in the system. Therefore, a server scheduling policy that ignores system state information can easily lead to suboptimal performance.

Whereas most previous studies on polling models have assumed that the server behaves independently of the state of the system, we discuss two kinds of state-dependent server scheduling rules: i) the threshold setups model, and ii) the threshold start-up model. In the former model, the server does not setup (and does not serve any customers) at a station at which it finds less than a critical number

of waiting customers, called the setup threshold. In the latter model, the server starts idling each time the system becomes empty, and it stays idle until arrivals to the system reach a critical number, called the start-up threshold. The server then resumes service from the station where it had stopped. Our analysis makes it possible to compare system performance under these state-dependent server scheduling rules and state-independent rules.

In this dissertation the following results are achieved. We develop an exact analysis for the one-threshold setup model with two stations, and an efficient approximation for the same model with any number of stations. For the general threshold setups model, we construct a numerical solution technique which is near-exact for calculating queue length distributions and station mean waiting times. The threshold start-up model is analyzed in detail, and mathematically exact expressions for mean station waiting times are obtained for both exhaustive and globally gated service regimes. For each model, the extension to the gated service regime is also discussed.

Acknowledgements

I would like to express my gratitude to those who helped and encouraged me to complete this dissertation. I am grateful to:

- My supervisor, Professor Diwakar Gupta, who generously devoted his time to me, contributed a lot to my knowledge and also, financially helped me during my graduate study;
- Professor Robert F. Love for his careful review of the manuscript and helpful suggestions;
- Professor Terry Todd for his comments and his interest in the dissertation;
- The external examiner, Professor Robert B. Cooper, for his interest in the dissertation and his guidance;
- Professor George O. Wesolowsky and Dr. David G. Jones for accepting to read the dissertation at such short notice and providing their valuable comments;
- Ph.D. students of the business faculty for creating an exciting and joyful environment to study and to *live*;
- Friendly administrative staff of Michael G. DeGroote School of Business for their technical assistance and their smiling faces.

I am also indebted to McMaster University for providing financial assistance during my doctoral study.

My sincere thanks are also due to my family and friends for their faith in me, and years of encouragement.

to my family,

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Polling systems are cyclic queueing models where multiple customer classes are attended by a single server. Each customer class (type) lines up in its own queue (station) to wait for the server to arrive, and the server visits stations in a predetermined cyclic order, e.g. $1, 2, \ldots, N, 1, 2 \ldots$. Each station has the M/G/1 characteristics, i.e., the arrival process is Poisson, service time has an independent general distribution and the buffer size is infinite. There are several application areas for these models, including:

i) computer and telecommunication networks such as token ring protocols, communication switch boards (see e.g., Takagi 1990, 1994).

ii) loop transportation and material handling systems such as automated guided vehicles (AGV) (see e.g., Bozer and Srinivasan 1991).

iii) multi-item production systems (see e.g., Federgruen and Katalan 1996).

Typically, such systems have a single server, which is either a *token* (in case of local area networks using token passing protocol to control access), or a machining center,

or a robot, or perhaps an automated guided vehicle (AGV). This server attends to several types of customers (stations) in a cyclic fashion. Customers are either jobs or messages that queue up for service (machining, transmission) at either geographically or logically distinct locations in the system. There is a vast literature dealing with various applications and analyses of polling models. Comprehensive surveys of previous work can be found in two review articles by Takagi (1990, 1994).

In this dissertation we discuss a special class of polling systems, in which the server behavior depends on the state of the system. In contrast to models with state-independent server behavior, analysis of these models is known to be very hard (Ferguson, 1986, and Hofri and Ross, 1987), and hence, there are very few articles in polling literature that deal with them. We present two general models of this class, and discuss them in detail in this thesis.

This chapter is organized in the following way. First we provide the basic definitions used in the thesis. Next, a brief literature review of polling systems is presented in section 1.2. Finally, in section 1.3, we present the motivation of the dissertation and give a brief tour of the thesis. Readers should note that portions of the thesis have been written as journal articles. Thus, although the common background of these models is explained in this chapter, detailed model description and a survey of literature that is pertinent to each model are presented separately in each chapter.

1.1 Basic Definitions

- A *patient server (PS)* refers to the server behavior according to which it stops upon finding the system empty.
- A *continuously roving (CR) server*, on the other hand, continues to switch among stations even when there are no customers waiting for service in the system.
- The *polling instant* is the time epoch at which the server polls (checks the status of a queue) a station. It marks the end of a switchover period from the previous station.
- The *station beginning instant* is the time epoch at which the server is ready to start serving customers (if any) in the queue.
- The *station completion epoch* is the instant at which the server finishes its work at a station, as dictated by the service strategy at that station, and is ready to switch to the next station. We also use the term *server departure* to refer to this instant.
- The *switch point* is the time epoch at which the server (physically) begins to move to the next station[1]
- Some widely used service disciplines are:

[1]Note that, this definition is slightly different from the definition of *switch points* given by Cooper and Murray (1969) for the zero switchover times model. However, in either case, there is a single switch point per station during each cycle. This is by far the most important distinction between switch points and station completion epochs.

i) Exhaustive service (E); when the server leaves a station only after exhausting the queue at that station.

ii) Gated service (G); when the server leaves the station after serving only those customers that were present at the moment of its arrival at the station.

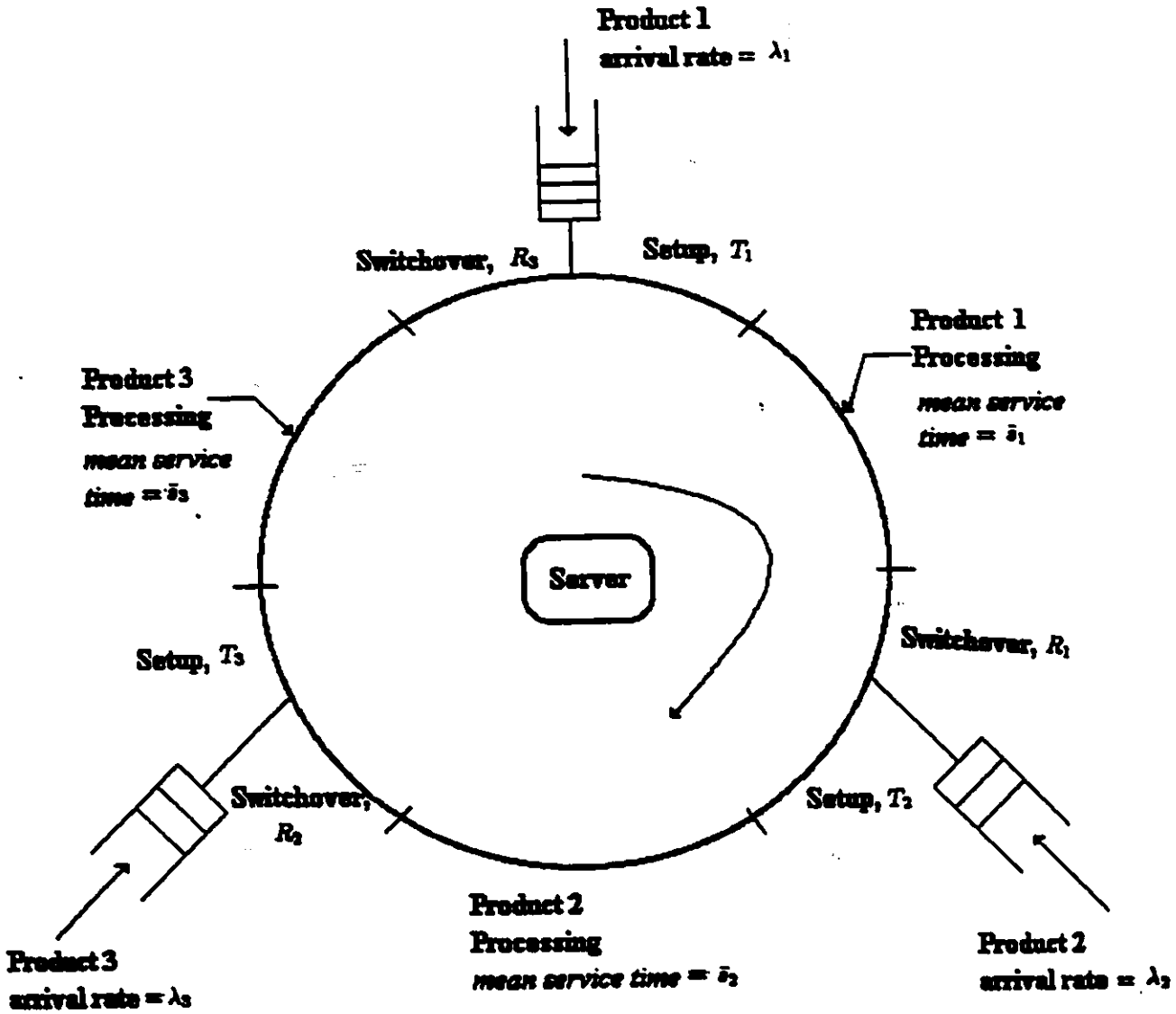
iii) Globally gated service (GG); when the server serves only those customers that are in the system at the polling instant of station 1 (where station 1 is chosen arbitrarily, without loss of generality) during its next visit to each station.

iv) K-limited service (L,K); when the server serves up to K customers at each visit to station i. Further specializations of this service regime are K-gated and K-exhaustive. The former strategy causes the server to process either K or the number present in the queue when the server arrives at the station, whichever is smaller. The latter, on the other hand, causes the server to either process exactly K customers or exhaust the queue before moving to the next station. Unless otherwise mentioned, K-limited service regime, and also the notation (L,K), refer to K-exhaustive discipline.

v) Decrementing service (D,K); when the server leaves the station with K less customers in the queue than what it found at the polling instant, unless the number of customers at the polling instant was less than K. In that case, the server leaves after exhausting the queue.

The basic definitions and notation for a three station polling system are illustrated in Figure 1.1. The system parameters and definitions of frequently used

Figure 1.1: A Three Station Polling System.



time variables for station i are:

- The arrival process for each station is independent and Poisson with rate λ_i.
- The *switchover (reply) time*, R_i, is a random amount of time needed by the server to physically move from station i to station $i+1$. This time interval may be zero.
- The *setup time*, T_i, is a random amount of time that the server spends getting ready to serve waiting type-i customers. It takes place after the server reaches station i, i.e., the switching process is over. The setup time may also be zero. If the setups are performed independently, i.e., at every visit of the server to station i, then we can treat the setup time of station i as a part of the switchover time from station $(i-1)$ to i, and define a new random variable which is the sum of the switchover and setup times. Thus, so long as setups are state independent, we may use the terms switchover and setup interchangeably. However, having a common mathematical treatment, does not change the fact that these time intervals are physically distinct.
- The *cycle time*, C_i, is the time elapsed between two consecutive station i polling instants.
- The *intervisit time*, I_i, is the time period, which starts with a type i switch point and ends at the next station i polling instant.
- The *vacation time*, V_i, is the period of time that the server is away from station-i. Thus, it starts at a station-i completion epoch and ends at the next station-i beginning instant.

1.2 Brief Literature Review

Polling systems have been studied in the queueing literature since the late 1960's. First, two-station queueing models were studied (see e.g., Eisenberg 1968, 1971, Takács 1968, and Sykes 1970) and then the models generalized to systems with any number of stations. However, two-station models continue to get special attention, because they are usually much simpler to analyze (e.g., Hofri and Ross 1987, Gupta and Srinivasan 1996).

The moments of the customer waiting times have been widely used performance measures for polling systems. Most early studies, however, calculated the mean waiting times only. Later, "pseudo-conservation" laws, which can often be derived without tedious mathematics, attracted the attention of researchers. Finally, after fast algorithms for calculating various performance measures were found, recent studies have concentrated on optimization of the polling systems. Therefore, we summarize the polling systems literature under three headings: i) calculating the mean waiting times, ii) "pseudo-conservation" laws, and iii) optimization.

1.2.1 Calculating the Mean Waiting Times

Cooper (1970) first calculated the mean waiting times in a polling system with zero switchover times, where the server must be assumed to be patient for the purpose of the mathematical analysis, since otherwise it leads to a singularity (the server performs an infinite number of cycles in a finite time). A complete analysis of queue lengths in the system can be found in Cooper and Murray (1969). They showed that in order to calculate mean customer waiting times in zero switchover polling

models, when the server stops it does not matter at which station it stops. This can also be explained by intuitive arguments. Since the switchover times are zero, after observing an arrival to an idle system the server moves to the arriving customer's station immediately. Thus, the station at which the server is idling is immaterial. Cooper and Murray (1969) also defined the *switch points*, which later become the main observation epochs for models with a patient server. In a later study, Cooper (1970) completed the analysis for zero switchover polling models by discussing the waiting times via the vacation model, which was introduced here. Cooper (1970) also presented a method to calculate the moments of waiting times. However, some recent methods are computationally more efficient.

Eisenberg (1971, 1972) presented a similar analysis for nonzero switchover polling systems. Eisenberg (1972) showed that observing a system at four sets of time instants is enough to analyze (all) polling models completely. These time instants are: i) polling instants, ii) station beginning epochs, iii) station completion epochs and iv) switch points. However in most cases, especially when the server behavior is state independent, it is enough to imbed a single Markov chain at any one of these four time epochs to analyze the system, e.g. for the continuously roving server polling systems with non-zero setup times (Takagi 1990) the polling instants, and for the zero switchover times model (Cooper 1970) the switch points, are used as the only observation instants.

Since the arrival processes are Poisson, the number of customers left behind by a departing customer is equal to the number of arrivals during the departing customer's sojourn time (see e.g., Kleinrock 1975, section 5.3) if the service discipline in the queue is first-in-first-out (FIFO), and thus the waiting time distribution of

type-i customers is a function of the distribution of the number of type-i customers in the system. Most of the previous studies (e.g., Cooper and Murray 1969, Cooper 1970, Eisenberg 1972, Ferguson 1985, Konheim *et al.* 1994, Srinivasan *et al.* 1996, and Takagi 1990) express the queue length distribution of station-i at arbitrary time epochs by relating it to the queue length distribution at polling instants or switch points. These methods are classified as *buffer occupancy* techniques in Konheim, Levy and Srinivasan (1994). Others, e.g., Ferguson and Aminetzah (1985), and Sarkar and Zangwill (1989) further obtain these queue length distributions at specific time instants using the cycle time or the intervisit time of the station (depending on the service discipline). We refer to these approaches as the *station time* methods.

Takagi (1987, 1990, 1994) summarized the previous studies on zero and nonzero switchover time models and describes a different method of calculating the waiting time moments. He obtains the moments of queue lengths upon solving a system of linear equations. For example, it requires the simultaneous solution of linear equations of size equal to the third power of the number of stations to obtain the mean waiting times. The number of equations to be solved increases exponentially if we wish to calculate the higher moments of the waiting time. Numerous variations of the zero switchover time models are discussed by Takagi (1987).

In order to be able to compute mean waiting times rapidly, many approximate methods have been proposed for different polling models. Among these the *individual stations* technique by Srinivasan, Levy and Konheim (1996) is very efficient, and provides the following important result. Srinivasan *et al.* prove that for the non-zero setup (switchover) times model the probability generating function (PGF) of the queue length of a station at its polling instant is affected mainly by the six or seven

neighboring stations, i.e., ignoring the effects of other stations does not reduce the accuracy of the approximation by much. Therefore, in order to find the mean waiting time at a single station, say i, it is enough to solve a seven by seven system of linear equations instead of solving a system of N^2 equations, which was the case for most of the exact methods at that time.

Although some of the approximate solutions are reported to perform very well, the need for faster exact algorithms to calculate the mean waiting times continues. In the 1980's, several efficient exact approaches for mean waiting times are presented, e.g. Ferguson and Aminetzah (1985), Sarkar and Zangwill (1989) and Konheim, Levy and Srinivasan (1994). Konheim *et al.* (1994) used the *descendant sets* (DS) method to calculate the PGF of the queue length at observation instants, e.g. polling instants. The DS method defines the successors of a customer, i.e., the arrivals during the service time of a customer and the arrivals during their service periods, as its "descendants". Then each customer in the system at an observation epoch (or at any time instant) has a single originator, where a customer is defined as an originator, if it does not arrive during the service period of another customer. Therefore, the PGF of queue lengths at an observation epoch can be calculated by counting their originators in a novel manner. Günalay and Gupta (1993) show that the DS method is the fastest among the methods mentioned above provided the number of stations is at least twenty and $\rho \leq 0.9$. For the remaining problem instants, the computation times are so small that run time differences of algorithms become unimportant. The same study shows that in addition to being fast, the DS technique has the following advantages over other algorithms:

i) Its run time is proportional to the number of stations for which the user wants to calculate the mean waiting times. Thus, if the waiting times at all stations are not required, the algorithm runs even faster.

ii) Calculating the higher moments of the waiting times require about the same effort as the mean waiting time calculations need.

iii) The technique is applicable to many polling models.

iv) The memory requirement is minimal, for both the code and the data.

As a fast and efficient technique, the DS method has been successfully implemented for various polling models, e.g., Gupta and Srinivasan (1996). Also, Srinivasan, Niu and Cooper (1995) use the DS method to generate a fast algorithm to calculate the waiting time moments in a non-zero switchover model, given the same performance measures in a zero-switchover model with the same system parameters. Their algorithm is based on the fact that the effect of switchover times on the waiting time distributions is additive. This relationship was first presented by Fuhrmann (1992), when the switchover times are constant. Later, Cooper, Niu and Srinivasan (1996) proved that for general switchover times the mean waiting time at each station decomposes into sum of two terms: 1) the mean waiting time in a "corresponding" model of the "original' system with switchover times set to zero, and the service time variances are modified appropriately, and 2) a simple function of moments of the switchover times. this relationship holds, i.e., the switchover times affect waiting time distributions in an additive manner. Therefore, the zero switchover time model can be viewed as a special case of the nonzero switchover time model,

with $R_i = 0$, $i = 1, \ldots, N$. This important result shows that categorizing the polling systems with respect to the server behavior (PS versus CR) is more appropriate than the zero vs. nonzero switchover times classification. The zero switchover time models form a subclass of the PS models (Eisenberg, 1995). Recently, in concurrent studies Eisenberg (1995) and Srinivasan and Gupta (1996) obtain the mean waiting times for the patient server polling systems. Eisenberg's study is a generalization of his earlier work on a two queue polling model with stopping server (Eisenberg 1971). Furthermore, the detailed study by Eisenberg (1995) contains variety of server stopping and start-up policies; but not the threshold start-up rule. The latter study, Srinivasan and Gupta (1996) utilizes the DS technique and thus it is highly efficient in calculating the mean waiting times of the customers. Furthermore, Srinivasan and Gupta (1996) present the relationship between CR server and PS polling models, which complements Srinivasan, Niu and Cooper's (1995) results on decomposition of polling systems.

Polling models with state dependent setup times are suitable for systems where unnecessary setups are undesirable. However, they are known to be very hard problems (see e.g., Ferguson 1986) and thus not much research has been done related to these models. Ferguson (1986), Bradlow and Byrd (1987) and Gupta and Srinivasan (1996) are among the few who study state dependent setup models. Among these, Bradlow and Byrd consider the patient server model and the other two assume a continuously roving server. Except for the two stations model studied by Gupta and Srinivasan (1996) all other reported methods are approximations. The approximations presented by Gupta and Srinivasan (1996) perform comparatively better than the others. Furthermore, Gupta and Srinivasan (1996) mentioned that

finding bounds to these models is not easy and present examples which show instants when all the upper bounds suggested by Ferguson (1986) fail to bound the mean customer waiting times for these polling systems.

1.2.2 The Pseudo Conservation Laws

Work conservation principle was first defined on single server priority queueing systems by Kleinrock (1976). If the arrival processes are independent and Poisson, and a non-preemptive service discipline is used, then the *conservation law* holds, i.e.,

$$\sum_{i=1}^{N} \rho_i E[W_i] = \frac{\rho \sum_{i=1}^{N} \lambda_i E[S_i^2]}{2(1-\rho)}. \tag{1.1}$$

Many service regimes, as long as they satisfy the above conditions, do not affect the expected total work in the system and therefore equation (1.1) holds. In polling models, the server may spend time on switching and/or setups while there are customers waiting in the system. Therefore, the implementation of a work conservation principle to polling systems leads to *pseudo-conservation* laws, which possess the following form:

$$\sum_{i=1}^{N} \rho_i E[W_i] = \frac{\rho \sum_{i=1}^{N} \lambda_i E[S_i^2]}{2(1-\rho)} + E[Y], \tag{1.2}$$

where $E[Y]$ depends on the system characteristics, i.e., the server scheduling policy and the service strategy at each station. This system dependent term, $E[Y]$, is first obtained for the non-zero switchover times model with limited service regime, by Watson (1984) and with exhaustive and gated service regimes by Ferguson and Aminetzah (1985). Later it has been shown that similar pseudo-conversation laws apply to many other polling models by different researchers, for example Boxma and

Groenendijk (1987), Everitt (1989), and Srinivasan and Gupta (1996) are a few among them.

The pseudo-conservation laws are obtained usually using intuitive arguments and they present explicit relations. In some cases they can be obtained, even when there is no known method to calculate the individual mean waiting times. Therefore, for such models the expected customer waiting times are obtained from the pseudo-conservation law, if the system is symmetric. Since these relationships are exact and explicit, some of their uses are as follows:

i) to develop approximations for individual mean waiting times in asymmetric polling models for which either there is no exact solution or solution algorithm is computationally inefficient, e.g., Everitt 1986.

ii) to test approximations and validate simulation results, e.g., Leung 1991.

The pseudo-conservation laws are available for many service strategies, and the models with state independent server behavior. However, the only state dependent server behavior model that is known to have a pseudo-conservation law is the patient server model (see Srinivasan and Gupta 1996).

1.2.3 Optimization

Optimization of polling systems is a multi-criteria problem, since many different objective functions can be chosen. The widely used one is to minimize the average total number in the system (work-in-process), which is equivalent to minimizing the mean waiting time of an arbitrary customer, i.e., $\sum_{i=1}^{N}(\lambda_i/\Lambda)E[W_i]$. In this section, we review some studies that use polling models to optimize different functions (see

e.g., Browne and Yechiali 1989, Boxma 1993, Boxma 1991, Federgruen 1993, Hofri 1986, Hofri and Ross 1987, Levy *et al.* 1990, and Liu *et al.* 1992).

Hofri (1986) studied a two station model with threshold setups and patient server, and tried to minimize total work in process in the system. Later, Hofri and Ross (1987) proved that the exhaustive service regime is optimal for minimizing total work in process in a two queue polling system with patient server and threshold setups, if service times are identically distributed. This study is important as it proves the optimality of a service regime to a class of polling models. The optimal service discipline is exhaustive, which means that at a station completion epoch the station queue is empty. Thus, for a two-station system, the "two stage threshold" switching rule is the best policy. This policy allows the server to switch to the next station, only if the present queue is empty and the other station has at least h_i, $i = 1, 2$, customers waiting at that queue. These threshold values can be found numerically for the two-station model. Since there are two stations and the service regime is E, the two stage threshold policy also dictates to the server when to stop. However, the analysis cannot be extended to models with more than two stations and thus, it is not clear if this rule is optimum for systems with more than two stations.

Browne and Yechiali (1989) present a dynamic server scheduling scheme for a cyclic queueing system to optimize the total work-in-process. They use the relationship between two consecutive cycles. At the beginning of each cycle, the best order for visiting stations is generated, so that the unnecessary switches are reduced and thus the cycle length is minimized. To implement these rules, the GG service discipline is ideal, because in the GG service regime the server checks the status of all the queues at cycle beginnings.

Boxma (1991) proved that the periodic, but not necessarily cyclic, server scheduling minimizes the total work in process in non-zero switchover polling models with 1-Limited service regime. He used a linear programming (LP) formulation to get the optimum station visit frequencies. Then the optimum station scheduling, also called *polling table* (Eisenberg 1972), is obtained by rounding these frequencies. This is a static optimization policy. Furthermore, Boxma mentioned that with respect to a "fairness" criterion, this policy is also better than G and (L,1) service disciplines under cyclic service scheduling. Fairness is measured by the degree to which customers are served in the order of their arrival, regardless of their type.

Levy, Sidi and Boxma (1990) compare several service disciplines commonly used in polling systems. Their analysis is distinct from the earlier studies, since they compare the service regimes with respect to *unfinished work*[2] at any time, whereas the previous comparisons have been based on the expected value of the performance measures summed over all stations. They prove that the exhaustive service regime dominates many other service strategies, including gated and limited, in minimizing the unfinished work in the system at any time.

Liu, Nain and Towsley (1992), further generalize the work by Levy *et al.* (1990) for symmetric stations[3]. Liu *et al.* consider not only the service policies, but also the server scheduling policies to stochastically minimize the unfinished work in the system. They show that an exhaustive service policy is optimal for polling

[2]Unfinished work is defined as the amount of time required to serve all customers in the system at that time. Furthermore, the mean of unfinished work corresponds to $\sum_{i=1}^{N} \rho_i E[W_i]$, when the queue discipline is FIFO, and it is called "virtual waiting time" by Takács (1962).

[3]We call polling systems with identical stations, i.e., all stations have the same arrival rate and identical service time distribution, as *symmetric stations*. Furthermore, if switchover times also have identical distribution, i.e., $R_i = R$, $i = 1, \ldots, N$, then the system will be called *symmetric systems*.

systems. This result is consistent with the study by Hofri and Ross (1987). Also, Liu *et al.* prove that for symmetric systems patient server policy is optimal. Furthermore, when dynamic server scheduling is considered the optimum policy belongs to the class of *stochastically largest queue* (SQL) policies. An SQL policy is such that it never schedules the server to a queue which is known to have a queue length that is stochastically smaller than that of another queue.

Federgruen and Katalan (1996) use the non-zero switchover times model to analyze multi-item produce-to-stock production systems. Their aim is to calculate the economic lot size for each item, in a stochastic environment. First, they note that the "variance effect" (Sarkar and Zangwill 1991) can be eliminated by increasing the mean cycle length. Therefore, they force the server to idle even when the system is not empty. Furthermore utilizing the results of Cooper, Niu and Srinivasan (1996), Federgruen and Katalan (1996) show that mean waiting times are affected through the sum of the mean switchover times. Thus, when the idle period is seen as a part of the switchover time, its effect is the same regardless of where the server is idling. This way, they investigate the influence of idle times on the economic lot scheduling problem in a stochastic environment.

1.3 Motivation of The Thesis

Research on polling systems is still growing at a rapid pace, and more and more application areas are being discovered. In order to adapt polling models to suit these areas, new variations are being studied. However, models with state-dependent server behavior have not received the attention that they deserve. Only, a few simple state-

dependent server scheduling rules, such as stopping the server when the system is empty (see Eisenberg 1995, and Srinivasan and Gupta 1996) or skipping the setup at empty stations (see Ferguson 1986, and Gupta and Srinivasan 1996) have been studied so far. The word "simple" in the above sentence refers to the server behavior, which shows a 0/1 (i.e., empty state vs. non-empty state) dependence, and not to the analysis, which is quite complicated. Indeed, the latter model has not been solved exactly for more than two stations. Hofri and Ross (1987) studied a two-station queueing model with threshold setups. Their study is one of the few exact results that are available for the state-dependent server scheduling rule models. In this dissertation, we consider more general state-dependent server scheduling rules for polling systems, which also include the simple rules mentioned above. We categorize these rules as *local* and *global*. The former type depends on the status of queue at the current station. However, for the latter rules one must consider all queue lengths before making server scheduling decisions. We present examples from both groups in the thesis.

In most polling systems there are setups. For example, in a multi-product manufacturing facility each customer class generally requires a different service and the server has to be set-up for each operation. Another case is a computer network, where all customer types might have the same service distribution, but a link between the terminal and the server computer has to be established before any transfer can occur. In such systems, although a setup is a must for the individual customer class, it is seen as an overhead for the whole system. Therefore, setups are undesirable and in practice their occurrences should be minimized. A state-dependent setup model employs an intuitive control mechanism: skip the setup of a station whose queue is

empty at its polling instant. Although the implementation of this rule is very simple, its analysis is very difficult. A few researchers have proposed approximate algorithms for this model. For example, Ferguson (1986) and Gupta and Srinivasan (1996) work with the continuous roving server variation, and Bradlow and Byrd (1987) and (in a recent study) Olsen (1996) work with the patient server variation of this model. In chapter 2 we discuss a patient server model with state-dependent setup times, and present two approximations and one numerical algorithm to calculate the mean waiting times of this model.

The threshold setups model is a generalization of the state-dependent setups model and it is more challenging. Due to its applications in manufacturing environments, this problem has attracted many researchers. However, because of its complexity, no exact solutions to the problem of computing mean waiting times for systems with more than two stations have been reported, yet. Hofri and Ross (1987) presented an exact solution for the two-station case, and showed that the E service regime is optimum if the service time distribution of both customer classes are identical. Markowitz *et al.* (1995) used a threshold setups model to analyze the stochastic economic lot scheduling problem (ELSP). They stated the analogy between the discrete space S-ESLP for a make-to-stock system and a polling system with setups. Then solving the dynamic server scheduling problem for the polling model with setups generates an ELSP policy which they compare to other policies. Markowitz *et al.* (1995) use the heavy traffic approximation (see Coffman *et al.* 1995) in their analysis which is an efficient method under certain conditions. Also, Federgruen and Katalan (1994) developed approximations for the queue length distributions in polling systems. They use these (approximate) queue

length distributions at polling instants to analyze the performance of a class of periodic base-stock policies for the ESLP (Federgruen and Katalan, 1996).

We examined the threshold setups model by considering the mean waiting times as the performance measure. We modified the numerical algorithm, which is introduced in chapter 2, to calculate the performance measure. Although the algorithm requires a considerable amount of computer resources, it is important since it is the only known solution that is nearly exact. Therefore, it can be used as a benchmark for testing approximations in the future.

Due to recent developments in the information technology, the cost of collecting information has reduced a lot. In many polling systems the electronic data (about queue status) travels much faster than the server itself. Most manufacturing systems use continuous monitoring of their inventory and machine status, i.e., the whole system state. Thus, one important disadvantage of centralized control policies, that is, the high cost of information, has virtually disappeared. Therefore, using global state-dependent server scheduling rules makes sense. Several global rules can be constructed. We present an example of these, which is the N-threshold start-up rule. We assume the server is patient and that when it becomes idle, it stays dormant until $N \geq 0$ arrivals (of any type) occur. On one hand, this rule is an extension of the "classical" patient server model (see e.g., Srinivasan and Gupta, 1996) and on the other hand, it generalizes the N-policy (see e.g., Heyman and Sobel, 1982) in M/G/1 queueing models to multi-product systems. We calculate the mean waiting times for E and GG service regimes. Patient server models were proven to be superior to the continuously roving server models for the GG service discipline by Borst (1995). Srinivasan and Gupta (1996) showed that a similar relationship is true for two queue

systems with E service regime, under certain conditions. In the lights of these studies, we have attempted to address the question "when the server stops, how patient should it be?" We have showed that the optimum threshold level, i.e., one that minimizes the mean unfinished work in system, can be obtained by a finite enumeration of the threshold level.

Besides its optimizing aspect, our model helps us to unify literature on polling systems. By setting $N = 0$ and $N = 1$ in this model (with either E or GG service regime), the results for the continuously roving model and the patient server model can be obtained, respectively. Similarly, setting the number of customer types to one reduces the model to an M/G/1 queueing system with N-policy. Such relationships (see e.g., Srinivasan, Niu and Cooper 1995) form bridges among different models and provide new insights to the problem.

The remainder of this thesis is organized in the following manner. In chapter 2, we discuss the patient server polling system with state-dependent setup times. The analysis of the model is accomplished by two approximations and a numerical algorithm to calculate the mean waiting times. A generalization of the state-dependent setups model is considered in chapter 3. We consider polling systems with threshold set-ups in this chapter. Chapter 4 is reserved for the discussion of a generalized patient server model: "Threshold Start-Up Control Policy for Polling Systems." In that chapter an exact analysis of the N-threshold start-up policy is presented for E and GG service regimes, and numerical examples are given to discuss some of the properties of such models. Finally, concluding remarks and future directions of this dissertation are presented in chapter 5. Appendices A, B and C are used to provide details of the analysis reported in chapter 4.

Chapter 2

Polling Systems with a Patient Server and State-Dependent Setup Times

In this chapter, we analyze a class of polling systems in which the server stops cycling upon finding the entire system empty and initiates a setup only when the polled station has at least one customer in the queue. Interest in such systems is fueled by applications in design and performance analysis of manufacturing as well as telecommunication systems.

We develop a Discrete Fourier Transform (DFT) based "near-exact" numerical technique and two approximate methods for systems with any number of stations. The DFT-based algorithm is accurate but computationally demanding when either the number of stations is large or server utilization is high. In this case, the *mean-variance* approximation appears to work well in a large number of numerical tests.

The overall organization of the remainder of this chapter is as follows. Section 2.1 is the introduction, where we also talk about the related literature.

Section 2.2 defines the model and our notation. This is followed in section 2.3 by the expressions for the waiting time distribution and the mean waiting time at each station. The analysis is developed mainly in section 2.4, which is divided into three parts. The first part deals with an exact analysis of the two station system. The second presents a DFT based numerical procedure for finding the complete probability distribution of joint queue lengths at polling instants and the third part contains an approximate method for finding mean station waiting times. Section 2.5 contains a summary of the experimental setup and of numerical test results. A variety of different polling models which can be treated using the methods presented in this article are discussed in section 2.6.

2.1 Introduction

In polling models, the server behavior can be specified in terms of possible actions following instances when it either "polls" (checks queue status) or "completes" a tour of service at a station. Such protocols simplify management and offer decentralized control. A polling instant occurs immediately after the server moves to a station. The server requires a setup before it can commence service at the newly arrived station.

There are two possible courses of action at each station completion instant: either the server moves to the next station, irrespective of system state, or it idles until the system occupies some desired state. Similarly, two types of server actions may be specified at each polling instant. These determine whether or not the server sets up for service, and how many customers it serves before registering a station completion instant. Until recently, most of the existing literature assumes that the server never

idles and that it sets up irrespective of whether there are any waiting customers at the polled queue or not, since these assumptions make the model more tractable. Gupta and Srinivasan (1996) describe such models as having a *continuously roving server* with *state-independent* setups. Commonly employed regimes which govern the number of customers served at each server visit include exhaustive, gated and timer limited.

The focus of attention of this chapter is the class of models which can be described as having a *patient* server and *state-dependent* setups. In our models, the server remains idle at the station at which it registers a station completion instant if the entire system happens to be empty at that instant. It is restarted next by a new customer arrival at *any* station in the system. Similarly, setups are incurred only when the polled station queue is not empty. Otherwise, the polled station is skipped and the server moves on to the next station. We present detailed analysis assuming exhaustive service strategy at each station. However, our analysis extends in a straightforward fashion to include gated regime as well.

Polling models with a patient server and state dependent setups are particularly suitable for representing systems for which avoiding unnecessary switching and setups is desirable (Gupta and Srinivasan 1996). For example, they mimic the operation of many computer communication networks (Bradlow and Byrd, 1987, and Ferguson, 1986) and production systems (Gupta and Srinivasan, 1996, and Srinivasan and Gupta, 1996). Furthermore, models which have either a continuously roving server or state independent setups or both can be studied as special instances of the patient server, state dependent setups model (see section 2.3). Therefore, it is a fundamentally important class of models for carrying out a detailed analysis and

performance evaluation. Unfortunately, the analysis of such models is not as well developed as it ought to be. The only papers that consider state dependent setups appear to be the ones by Ferguson (1986), Bradlow and Byrd (1987), Gupta and Srinivasan (1996) and Olsen (1996).

With the exception of the exact analysis for the two station model in Gupta and Srinivasan (1996), other analyses presented in all of the studies mentioned above are approximate analyses. As such, the problem is widely regarded as a difficult problem to analyze. Ferguson (1986) first observes the difficulty in 'solving' the functional equations which relate intervisit times at a station to the server sojourn time at that station. He develops some bounds and approximations for the mean waiting times. Gupta and Srinivasan (1996) present a counter example in which all of Ferguson's upper bounds for the mean waiting times of a two station system are lower than their corresponding exact values. Gupta and Srinivasan (1996) also develop an approximation procedure which takes the average of two separate ways of approximating the system behavior. The average value of the station mean waiting times obtained from the two approximations have been found to be typically within 5% of the simulated values in a large number of test cases. This makes their approximation the most accurate among those reported in the literature.

Bradlow and Byrd (1987) set out to analyze a patient server model with zero switchover times and state dependent setups. However, in their approximation scheme they assume that the probability of having the system empty is zero. This assumption effectively reduces their model to the continuously roving server model studied by Ferguson, and by Gupta and Srinivasan. Olsen (1996) proposes a distribution approximation for waiting times, concentrating on the simplicity of the approximation

rather than on its accuracy.

In this chapter, first we show that the two station system with a patient server and state dependent setups can be analyzed using the computationally efficient *descendant sets* technique (see, for example, Konheim *et al.*, 1994, for details of the descendant sets method). The same two stations model, but in which setup times are called changeover times, has been analyzed earlier by Eisenberg 1971). Next, we develop a numerical procedure, utilizing Discrete Fourier Transforms, for finding the joint queue length probabilities at polling instants for systems with arbitrary numbers of stations. These probabilities immediately yield the desired performance measures. The procedure assumes that the size of the buffer at each station is finite. Although this makes the technique mathematically approximate, numerical tests show it to be highly accurate when the limiting buffer sizes are chosen carefully. From a practical standpoint, finite buffers are more realistic.

The DFT-based technique requires considerable computer resources. It appears to be practical only when the number of stations is small and overall server utilization is relatively low (≤ 0.5). Therefore, we also present two approximations. One of the two is an adaptation of an approximation scheme by Gupta and Srinivasan (1996), to handle systems with high server utilization and/or a large number of stations. The second approximation is a modification of the first one. It seems to work more efficiently for all system parameters with an average error less than 3.5%. Thus, the problem can be analyzed completely and with reasonable accuracy with the help of the two complementary approaches we present.

2.2 Preliminaries and Notation

Let stations $Q_1, Q_2, \ldots, Q_N$ be indexed in the same order as they are visited and let the customers served at Q_i be known as type-i customers. Once a tour of service begins, customers at each station are served exhaustively and in the First-Come-First-Served (FCFS) fashion. Let λ_i denote the arrival rate of the type-i customers and let the random variables, T_i, S_i and B_i represent the setup time, the service time and the busy period at Q_i, $i = 1, \ldots, N$, respectively. Whereas arrivals follow independent Poisson patterns, random variables T_i, and S_i have arbitrary distributions.

The overall arrival rate is denoted by λ, i.e., $\lambda = \sum_{i=1}^{N} \lambda_i$. The long run proportion of type-i arrivals to the system, p_i, equals λ_i/λ; utilization (traffic intensity) at station i, ρ_i, equals $\lambda_i E[S_i]$; and $\rho = \sum_{i=1}^{N} \rho_i$ denotes the overall utilization of the server. Note that $\rho < 1$ is the necessary and sufficient condition for the stability of the system (see e.g., Altman *et al.* 1992) and this is assumed to be the case. Our aim is to present numerical methods to calculate the steady state probabilities.

The Cumulative Distribution Function (CDF) of a random variable A is denoted by A(t) and its Laplace-Stieltjes Transform (LST), defined as $E[e^{-sA}]$, by $A^*(s)$. If the random variable is discrete, then its Probability Generating Function (PGF), $E[z^A]$, is denoted by $A(z)$. The k^{th} moment of a random variable A is indicated by $a^{(k)}$, where $a^{(1)} \triangleq E[A] = \bar{a}$. We use the convention that any empty sum equals zero and that any empty product equals 1.

Four time epochs, at which Markov chains are imbedded, are of special significance (see Eisenberg 1972 and 1995). These are: polling instants, station-

beginning instants, station-completion instants and switch points. As explained below, every switch point has a corresponding station completion instant and similarly each polling instant can be associated with a station beginning instant, but the opposite relationships do not always hold. The polling cycle for station-i, C_i, is defined as the time interval between two consecutive polling instants of Q_i.

At each station-i polling instant the server checks Q_i and if it finds any type-i customers waiting, then a setup is performed. Otherwise, i.e., in case Q_i is empty, the server simply skips that station and immediately switches to the next station. Although the server does not serve any type-i customers during that visit, we mark both a station i beginning and a station i completion instant. At each station-i completion instant, the server looks at the whole system and continues to rove only if there is at least one station, other than i, which is not empty. However, if it observes an empty system, it then idles at station i and waits for an arrival to occur. A station-i completion epoch is registered, but no switch point is marked, since the server does not move out of Q_i.

The server is activated next by the first new arrival to the system. We define the length of time which starts by this arrival, and ends when the server stops again, as a *super cycle* (see Cooper and Murray, 1969). If this arrival happens to be at Q_i, the station where the server resides, then another station-i beginning instant is registered and service resumes (no setup required). Otherwise, a type-i switch point is registered and the server leaves Q_i immediately, moving in the cyclic sequence and arriving instantaneously at the desired station. Thus, during any cycle, there is exactly one polling instant and one switch point for each station, but there may be more than one (but equal number of) station-i beginning and completion instants,

for $i = 1, \ldots, N$.

Let ${}^{i}(n_1, \ldots, n_N)$ denote the state of the system at an observation epoch which is either a polling instant, or a station completion epoch at station-i. The term n_j represents the number of customers waiting at queue j. Next, let $f^{i}(\bar{n})$ and $g^{i}(\bar{n})$, where $\bar{n} = [n_1, \ldots, n_N]$, denote the state probability at a polling instant and at a station completion epoch of Q_i, respectively. These probabilities are calculated using the long run proportion of the number of times the system state is ${}^{i}(n_1, \ldots, n_N)$ at such observation epochs to the total number of station completion events encountered. The PGFs of these probability distributions are defined as, $f_i(\bar{z})$, and $g_i(\bar{z})$. We use the upper case letters to denote the conditional probability and PGF of system state. Put differently, $F^{i}(\bar{n})$ $(G^{i}(\bar{n}))$ is the joint probability that queue lengths are $(n_1, \ldots, n_N)$ given that a station-i has been polled (completed) and $F_i(\bar{z})$ $(G_i(\bar{z}))$ is the corresponding PGF. It follows that:

$$F_i(\bar{z}) = f_i(\bar{z})/f_i(\bar{1}), \qquad i = 1, \ldots, N, \tag{2.1}$$

where $\bar{1}$ is a $1 \times N$ vector of 1's.

2.3 Waiting Times

The customer waiting time at Q_i, denoted by W_i, can be obtained using intuitive arguments if we imagine that at each station i completion epoch the server goes on *vacation* which ends at the next station-i station-beginning instant. Let $K_i(z)$, be the PGF of the number of type-i customers present at Q_i at the end of such a vacation. Then, Fuhrmann and Cooper's "Stochastic Decomposition (SD) Theorem" (see Fuhrmann and Cooper, 1985, for details) applies and the LST of the waiting

time of a type-i customer is the product of the following two terms:

$$W_i^*(\lambda_i - \lambda_i z) = \frac{1 - K_i(z)}{K_i'(1)} \left(\frac{(1-\rho_i)}{S_i^*(\lambda_i - \lambda_i z) - z} \right). \tag{2.2}$$

In our model, a station beginning epoch occurs either after a polling instant is registered at station-i and setup is completed; or after a server idle period at station-i which is followed by a new type-i arrival. If $K_i(z) = k_i(z)/k_i(1)$, then the previous statement permits us to write:

$$k_i(z) = [f_i(1,\ldots,1,z,1,\ldots,1) - f_i(\bar{1}_i)]T_i^*(\lambda_i - \lambda_i z) + f_i(\bar{1}_i) + g_i(\bar{0})p_i z, \tag{2.3}$$

where $g_i(\bar{0})$ is the *empty system* probability for $i = 1,\ldots,N$, and $\bar{1}_i = [1,\ldots,1,0,1,\ldots,1]$.

Theorem 2.1 *The LST of the waiting time distribution of type-i customers in a polling model with a patient server and state dependent setups is*

$$W_i^*(\lambda_i - \lambda_i z) = \frac{1 - F_i(1,..,z,..,1)T_i^*(\lambda_i - \lambda_i z) - (1 - T_i^*(\lambda_i - \lambda_i z))f_i(0) + \vartheta_i p_i(1-z)}{F_i'(\bar{1}) + (1 - f_i(0))\lambda_i \bar{t}_i + \vartheta_i p_i} \cdot \frac{(1-\rho_i)}{S_i^*(\lambda_i - \lambda_i z) - z}, \tag{2.4}$$

and the mean waiting time is

$$E[W_i] = \frac{f_i^{(2)} + 2\lambda_i \bar{t}_i f_i^{(1)} + (1 - f_i(0))\lambda_i^2 t_i^{(2)}}{2\lambda_i (f_i^{(1)} + (1 - f_i(0))\lambda_i \bar{t}_i + \vartheta_i p_i)} + \frac{\lambda_i E[S_i^2]}{2(1-\rho_i)}, \tag{2.5}$$

where $f_i(0) \triangleq F_i(1,\ldots,1,0,1,\ldots,1)$ is the empty station probability, $\vartheta_i \triangleq g_i(\bar{0})/f_i(\bar{1})$, $f_i^{(1)} \triangleq \partial F_i(1,\ldots,z,\ldots,1)/\partial z \,|_{z=1}$, and $f_i^{(2)} \triangleq \partial^2 F_i(1,\ldots,z,\ldots,1)/\partial z^2 \,|_{z=1}$.

Proof: Setting $z = 1$ in equation (2.3) we obtain

$$k_i(\bar{1}) = (f_i(\bar{1}) - f_i(\bar{1}_i))T_i^*(0) + f_i(\bar{1}_i) + g_i(\bar{0})p_i, \tag{2.6}$$

$$= f_i(\bar{1}) + g_i(\bar{0})p_i. \tag{2.7}$$

Using equations (2.3) and (2.7), we get

$$K_i(z) = \frac{(F_i(1,\ldots,z,\ldots,1) - f_i(0))T_i^*(\lambda_i - \lambda_i z) + f_i(0) + \vartheta_i p_i z}{1 + \vartheta_i p_i}, \quad (2.8)$$

and by differentiating it with respect to z:

$$K_i'(1) = \frac{(f_i^{(1)} + (1 - f_i(0))\lambda_i \bar{t}_i + \vartheta_i p_i}{1 + \vartheta_i p_i}. \quad (2.9)$$

Substituting equations (2.8) and (2.9) into relation (2.2) leads to

$$W_i^*(\lambda_i - \lambda_i z) = \frac{(1 + \vartheta_i p_i) - (F_i(1,..,z,..,1) - f_i(0))T_i^*(\lambda_i - \lambda_i z) + f_i(0) + \vartheta_i p_i z}{f_i^{(1)} + (1 - f_i(0))\lambda_i \bar{t}_i + \vartheta_i p_i} \cdot \frac{(1 - \rho_i)}{S_i^*(\lambda_i - \lambda_i z) - z}. \quad (2.10)$$

Rearranging terms in the numerator of the above relationship results in (2.4). Then, taking the derivative of equation (2.10) with respect to z, and setting $z = 1$ and dividing by λ_i we obtain the mean waiting time of type-i customers shown in equation (2.5). #

Next we observe that several of the previously studied polling models can be looked upon as special cases of our model and hence their customer waiting time distributions can be obtained from Theorem 2.1.

Observation 2.1 A patient server polling model with state independent setups is obtained by setting the empty station probabilities, $f_i(0)$s, to zero in equations (2.4) and (2.5).

By setting $f_i(0) = 0$ in equations (2.4) and (2.5), we get

$$W_i^*(\lambda_i - \lambda_i z) = \frac{1 - F_i(1,..,z,..,1)T_i^*(\lambda_i - \lambda_i z) + \vartheta_i p_i(1 - z)}{F_i'(\bar{1}) + \lambda_i \bar{t}_i + \vartheta_i p_i}$$

$$\cdot \frac{(1-\rho_i)}{S_i^*(\lambda_i - \lambda_i z) - z}, \quad \text{and} \tag{2.11}$$

$$E[W_i] = \frac{f_i^{(2)} + 2\lambda_i \bar{t}_i f_i^{(1)} + \lambda_i^2 t_i^{(2)}}{2\lambda_i (f_i^{(1)} + \lambda_i \bar{t}_i + \vartheta_i p_i)} + \frac{\lambda_i E[S_i^2]}{2(1-\rho_i)}. \tag{2.12}$$

Note that the observation epochs in Srinivasan and Gupta, 1996, correspond to the station beginning instants in our analysis. Therefore, using equation (2.3) we obtain

$$W_i^*(\lambda_i - \lambda_i z) = \frac{k_i(\bar{1}) - k_i(1, \ldots, z, \ldots, 1)}{k_i'(\bar{1})} \left(\frac{(1-\rho_i)}{S_i^*(\lambda_i - \lambda_i z) - z} \right). \tag{2.13}$$

Above relationship is equivalent to equation (21) of Srinivasan and Gupta (1996) after appropriate notational changes, and by differentiating it with respect to z and simplifying the resultant expression result in equation (30) of the same paper.

Observation 2.2 The zero switchover time model, first studied by Cooper and Murray (1969) is obtained when we set $T_i = 0$, $i = 1, \ldots, N$. This means we replace $T_i^*(\cdot)$ by 1 in equation (2.5), to obtain an equivalent of equation (33) in Cooper and Murray (1969).

Observation 2.3 Continuously roving server polling models are also a special case of our model. By setting $\vartheta_i = 0$, in (2.4) and (2.5) we obtain the continuously roving server system with state dependent setups (see equations (13) and (16) of Gupta and Srinivasan, 1996). Furthermore, setting $f_i(0) = 0$, $i = 1, \ldots, N$, reduces our model to the basic polling model having a continuously roving server and state independent setups (see Takagi 1994 for a variety of methods dealing with this model).

From theorem 1, it is clear that the moments of station waiting time depend on $F_i(1, \ldots, 1, z, 1, \ldots, 1)$ through the latter's moments. Hence, we devote the

next section to finding moments of queue lengths at polling instants. To make the exposition simple, and without any loss of generality, we restrict our attention to station 1 only. Similar results for other stations can be obtained simply by changing the station indices.

2.4 Queue Lengths at Polling Instants

Consider an arbitrary polling instant at Q_1 which we shall call the *reference point*. The main idea of the descendant sets method is to express the queue length at Q_1 at the reference point as the sum of "contributions" from all previous customers until the reference point is reached. To facilitate in this process, we index cycles backward in time and develop a recursive method for finding the descendants of each customer until the reference point is reached.

Recall that a polling cycle is the length of time that elapses between any two consecutive polling instants at a station. Cycles are indexed such that the cycle which finishes just before the reference point is $c = 0$, the cycle prior to it is $c = 1$ and so on .... Thus, the reference point is the start of a cycle which is indexed $c = -1$. Let $C_{i,c}$ denote a type-i customer who gets service during cycle c. The descendant set of $C_{i,c}$ consists of itself, all customers who arrive during its service time (children), children of those customers (grand-children), their grand-children, and so on, until the reference point. Thus the descendant set of a customer is a proper subset of its parent's descendant set, if it has one. Since we can count the size of the descendant set of each customer, we only need to count the number of customers from $-\infty$ to the reference point which do not have a parent. Such customers are those that either

start a super cycle or arrive during setup periods. They have been called *original customers* by Konheim *et al.* 1994.

Using these ideas, the PGF of the size of the descendant set of $C_{i,c}$, which we denote by $L_{i,c}(z)$, can be calculated recursively by the following expression:

$$L_{i,c}(z) \triangleq B_i^* \left(\sum_{j=i+1}^{N} [\lambda_j - \lambda_j L_{j,c}(z)] + \sum_{j=1}^{i-1} [\lambda_j - \lambda_j L_{j,c-1}(z)] \right). \tag{2.14}$$

Boundary values are set as follows: $L_{1,-1}(z) = z$ and $L_{i,-1}(z) = 1$ so long as $i > 1$. Similarly, $\mathcal{T}_{i,c}(z)$ is the PGF of the contributions of all original customers who arrive during a setup which occurs c cycles prior to the reference point. This can be calculated as:

$$\mathcal{T}_{i,c}(z) \triangleq T_i^* \left(\sum_{j=i}^{N} [\lambda_j - \lambda_j L_{j,c}(z)] + \sum_{j=1}^{i-1} [\lambda_j - \lambda_j L_{j,c-1}(z)] \right). \tag{2.15}$$

Finally, the station 1 queue length at the reference point can be written in terms of the sum of contributions of all customers without a parent. The resultant relationship is given below:

$$F_1(z,1,\ldots,1) = \prod_{c=0}^{\infty} \prod_{i=1}^{N} \mathcal{T}_{i,c}(z) + \sum_{c=0}^{\infty} \sum_{i=1}^{N} \mathcal{F}_{i,c}^0(z)(1-\mathcal{T}_{i,c}(z))\sigma_{i,c}(z) - \sum_{c=0}^{\infty} \sum_{i=1}^{N} \vartheta_i J_{i,c}(z)\sigma_{i,c}(z), \tag{2.16}$$

where for $i = 1,\ldots,N$,

$$J_{i,c}(z) = \sum_{j=i}^{N} p_j - p_j L_{j,c}(z) + \sum_{j=1}^{i-1} p_j - p_j L_{j,c-1}(z), \quad \text{and} \tag{2.17}$$

$$\sigma_{i,c}(z) = \prod_{j=i+1}^{N} \mathcal{T}_{j,c}(z) \prod_{p=0}^{c-1} \prod_{k=1}^{N} \mathcal{T}_{k,p}(z). \tag{2.18}$$

$\mathcal{F}_{i,c}^0(z)$ is the PGF of the sum of contributions from all customers present in the system at a polling instant of Q_i, $i = 1,\ldots,N$, when Q_i is empty. An empty queue

does not "contribute" anything to station 1 queue at the reference point. Thus

$$\mathcal{F}^0_{i,c}(z) \triangleq F_i(L_{1,c-1}(z), \ldots, L_{i-1,c-1}(z), 1, L_{i+1,c}(z), \ldots, L_{N,c}(z)). \tag{2.19}$$

Let $\mathcal{D}$ denote $f_i(\bar{1})$, the probability of polling station-i. Note that this probability is independent of the station index since each station is polled exactly once in each cycle. Each station-$(i+1)$ polling instant is preceded by a station-i completion epoch, except when the latter happens to be an empty system epoch to which the first arrival is of type-i. Therefore,

$$\mathcal{D} = g_i(\bar{1}) - p_i g_i(\bar{0}), \tag{2.20}$$

and if we sum this relation for all i, we obtain $N\mathcal{D} = 1 - \sum_{i=1}^{N} p_i g_i(\bar{0})$. The empty system probabilities, $g_i(\bar{0})$s, can be found by solving a system of N linear equations as shown in Srinivasan and Gupta. These equations have the following form (for details see Srinivasan and Gupta, 1996, section 3.3)

$$\sum_{j=1}^{N} g_j(\bar{0}) \left(\sum_{c=0}^{\infty} J_{j,c}(\bar{0}) \sigma_{j,c}(\bar{0}) \right) = f_i(\bar{1}) \prod_{c=0}^{\infty} \prod_{j=1}^{N} \mathcal{T}_{j,c}(\bar{0}), \tag{2.21}$$

for $i = 1, \ldots, N$.

Observation 2.4 By setting $\vartheta_i = 0$ in equation (2.16) we obtain the PGF of the queue length at the Q_1 polling instants for the continuously roving server model with state dependent setups (see equation (23) of Gupta and Srinivasan 1996). Similarly, after eliminating expressions containing the $\mathcal{F}^0_i(z)$ terms from equation (2.16), the resulting PGF corresponds to a patient server model with state independent setups (see equation (12) of Srinivasan and Gupta 1996).

2.4.1 Two Station Models

When $N = 2$, notice that $\mathcal{F}_i^0(z) = 0$, $i = 1, 2$, since these correspond to the polling instants of Q_i when the system is empty, and that never happens. Upon eliminating these terms the resulting mathematical model reduces to a patient server model with state independent setups. Such a model is analyzed in detail by Srinivasan and Gupta (1996). Using their technique one can calculate the empty system probabilities, $g_i(\bar{0})$, $i = 1, 2$, by solving a set of linear equations like equation (2.21) above.

However, for $N > 2$ the $\mathcal{F}_i^0(z)$ terms remain unknown. Notice that these terms represent PGFs and not simple probabilities, which makes the problem particularly unwieldy. In the following two subsections we present two different ways which can be used to overcome this difficulty.

2.4.2 The Method of Discrete Fourier Transforms

In this subsection we use DFT to calculate the entire distribution of joint queue lengths at all polling instants. Our technique is fashioned after a numerical approximation used by Leung (1991) to analyze polling models with a continuously roving server, state independent setups, and probabilistic service strategy. Under the probabilistic service regime, of which K-limited service is a special case, the server processes at most j customers at its visit to station-i with probability a_i^j. Different service strategies can be obtained by specifying the probability distribution a_i^j for each i. Our approach is a numerical approximation since it requires the maximum station queue lengths to be finite. However, this limitation does not seem to have a significant effect on the accuracy of our calculations, so long as the limiting queue lengths are chosen carefully. Note that while computing queue length moments for

even the most simple polling models, the accuracy of calculations is limited either by the machine accuracy or by a user-specified tolerance level or both. Therefore, when limiting queue sizes are chosen carefully, our method yields nearly exact values of the mean station waiting times. The maximum buffer size at Q_i, which we denote by $\mathcal{L}_i$, $i = 1, \ldots, N$, is chosen such that $P(\text{number at station } Q_i \geq \mathcal{L}_i)$ is a small quantity (e.g. $\leq 10^{-8}$).

Another important aspect of the algorithm is that the maximum number of service completions at each visit to Q_i is assumed to be known in advance. Even for the exhaustive service discipline, we know that in steady state there exists a finite $\mathcal{K}_i$ such that

$$P(\text{Number of customers served at each visit to } Q_i \leq \mathcal{K}_i) = 1.$$

We defer the discussion of how to set $\mathcal{K}_i$ to section 2.5 and for the time being simply assume that this number is known for each station.

Before describing the algorithm, let us derive the functional relationship between the PGF of queue lengths at polling instants. First, we define $v_{i,j}(\bar{z})$ as the PGF of the queue lengths immediately after the j^{th} service completion at Q_i, for $j = 0, \ldots, \mathcal{K}_i$. Note that the end of the 0^{th} service actually corresponds to the station beginning epoch at Q_i. Suppose Q_i becomes empty after $\mathcal{Y}$ service completions, $\mathcal{Y} \leq \mathcal{K}_i$. We still perform the $\mathcal{K}_i - \mathcal{Y}$ fictitious service completions, but the system state does not change as a result of these service completions.

Theorem 2.2 *The relationship, $f_{i+1}(\bar{z}) = \mathcal{M}_i\{f_i(\bar{z})\}$, between the PGFs of the queue length distributions at consecutive polling instants of Q_i and Q_{i+1} is obtained from*

the following equations:

$$v_{i,0}(\bar{z}) = [f_i(\bar{z}) - f_i(\bar{z}_i)]T_i^* \left(\sum_{j=1}^{N} \lambda_j - \lambda_j z_j\right) + f_i(\bar{z}_i) + g_i(\bar{0})p_i z_i, \quad (2.22)$$

$$v_{i,k+1}(\bar{z}) = v_{i,k}(\bar{z}_i) + [v_{i,k}(\bar{z}) - v_{i,k}(\bar{z}_i)] \frac{S_i^* \left(\sum_{j=1}^{N} \lambda_j - \lambda_j z_j\right)}{z_i}, \; k = 0, .., \mathcal{K}_i, \quad (2.23)$$

$$g_i(\bar{z}_i) = v_{i,\mathcal{K}_i}(\bar{z}_i), \quad (2.24)$$

$$f_{i+1}(\bar{z}) = [g_i(\bar{z}_i) - g_i(\bar{0})] + g_i(\bar{0}) \sum_{\substack{j=1 \\ j \neq i}}^{N} p_j z_j, \quad (2.25)$$

where $\bar{z}_i = [z_1, \ldots, z_{i-1}, 0, z_{i+1}, \ldots, z_N]$.

Proof: The following arguments lead to relationships (2.22) – (2.25):

- The number of customers present in the system at a station-i beginning epoch are either those that were already in the system at the polling instant of Q_i or those that arrive during the setup period which follows. Furthermore, if the system becomes empty at a station-i completion epoch, then with probability p_i another station-i beginning with only one customer in Q_i can be observed. Putting these together results in equation (2.22).

- There are exactly $\mathcal{K}_i$ services at each server visit to Q_i and the system state changes only when Q_i is not empty. Equation (2.23) accounts for the change in system state during these service completions.

- By definition, the $\mathcal{K}_i^{\text{th}}$ service completion exhausts Q_i, yielding (2.24).

- The polling instant of Q_{i+1} matches with the station-i completion epoch, unless the server finds the system empty. In that case, the polling instant coincides

with the arrival epoch of the type-j customer, where $j \neq i$. The resulting relationship is equation (2.25).

Hence it is proved. #

Let Ω_i be the functional relationship between the PGF of the queue lengths at two consecutive polling instants of Q_i, defined as

$$\Omega_i[f_i(\bar{z})] = \mathcal{M}_{i-1}\{\mathcal{M}_{i-2}\{\cdots\mathcal{M}_{i+1}\{\mathcal{M}_i\{f_i(\bar{z})\}\}\cdots\}\}. \tag{2.26}$$

Then, starting with an arbitrary initial value for $f_i(\bar{z})$ and repeatedly applying $\Omega_i[\cdot]$ infinitely many times, yields the steady state $f_i(\bar{z})$. A detailed account of the convergence of such procedures as the number of times Ω_i is applied approaches infinity can be found in Cooper and Murray, 1969. Eisenberg (1972) also reported that the rate of convergence is fast for reasonable ρ values.

Even though the nested mappings described in relationship (2.26) ought to, in principle, yield the steady state PGF of queue lengths at polling instants, the steps described in equations (2.22)-(2.25) require knowledge of PGFs $f_i(\bar{z}_i)$, which cannot be derived from other known relationships. Therefore we use the DFT of the state probabilities for which the analogous unknown DFTs *can* be obtained. Details of the method are described next.

For a finite discrete random variable, one can view the DFT of this random variable as a finite sampling of its PGF on the unit circle and obtain it simply by replacing z with $e^{-j\omega}$ for $-\pi \leq \omega \leq \pi$ and $j = \sqrt{-1}$. Then for an N dimensional PGF, such as $f_i(\bar{z})$, after defining the maximum (possible) queue length as $\mathcal{L}_i$, $i = 1, \ldots, N$,

its DFT equivalent is (see e.g., Poularikas and Seely, 1991, section 10.7)

$$\hat{f}_i(\bar{k}) \triangleq \sum_{n_1=0}^{\mathcal{L}_1-1} \cdots \sum_{n_N=0}^{\mathcal{L}_N-1} f^i(n_1,\ldots,n_N) w_1^{k_1 n_1} \cdots w_N^{k_N n_N}, \tag{2.27}$$

where $w_i = e^{-2\pi j/\mathcal{L}_i}$ and $\bar{k} \in \mathcal{S} = \{(k_1,\ldots,k_N) : k_i = 0,\ldots,\mathcal{L}_i-1, \text{ for each } i = 1,\ldots,N\}$.

Similarly we define the DFTs of $g^i(\bar{n})$ and $v^{i,j}(\bar{n})$ for $j = 0,\ldots,\mathcal{K}_i$ and $i = 1,\ldots,N$, and denote them as $\hat{g}_i(\bar{k})$ and $\hat{v}_{i,j}(\bar{k})$, $\bar{k} \in \mathcal{S}$ respectively. Note that just like the PGF $g_i(\bar{z}_i)$, the DFT $\hat{g}_i(\bar{k})$ also does not depend on the i^{th} parameter, k_i, since $g^i(\bar{n}) = 0$, unless $n_i = 0$. Furthermore, the DFTs of arrivals during the service time and the setup time are denoted by $\hat{S}_i^*(\bar{k})$ and $\hat{T}_i^*(\bar{k})$ for $\bar{k} \in \mathcal{S}$ and $i = 1,\ldots,N$, respectively and given as follows:

$$\hat{S}_i^*(\bar{k}) = S_i^*\left(\sum_{j=1}^{N} \lambda_j - \lambda_j w_j^{k_j}\right), \tag{2.28}$$

$$\hat{T}_i^*(\bar{k}) = T_i^*\left(\sum_{j=1}^{N} \lambda_j - \lambda_j w_j^{k_j}\right). \tag{2.29}$$

Let $\hat{f}_i^0(\bar{k}_i)$ denote the DFTs which corresponds to the PGFs, $f_i(\bar{z}_i)$, $i = 1,\ldots,N$. Then,

$$\hat{f}_i^0(\bar{k}_i) \triangleq \sum_{n_1=0}^{\mathcal{L}_1-1} \cdots \sum_{n_{i-1}=0}^{\mathcal{L}_{i-1}-1} \sum_{n_{i+1}=0}^{\mathcal{L}_{i+1}-1} \cdots \sum_{n_N=0}^{\mathcal{L}_N-1} f^i(\bar{n}_i) w_1^{k_1 n_1} \cdots w_{i-1}^{k_{i-1} n_{i-1}} w_{i+1}^{k_{i+1} n_{i+1}} \cdots w_N^{k_N n_N}. \tag{2.30}$$

Notice that, $\hat{f}_i^0(\cdot)$ has $N-1$ independent variables. To indicate this, we define the domain set as $\mathcal{S}^i = \{(k_1,\ldots,\mathcal{L}_i,\ldots,k_N) : k_j = 0,\ldots,\mathcal{L}_j-1, \ \ j = 1,\ldots,N \text{ and } j \neq i\}$. Therefore, $\mathcal{S}^i$ is the projection of $\mathcal{S}$ into I^{N-1}, where I denotes the set of natural numbers. Finally, by using the *initial value property* of DFT (Poularikas and Seely,

1991) we obtain

$$\hat{f}_i^0(\bar{k}_i) = \frac{1}{\mathcal{L}_i} \sum_{k_i=0}^{\mathcal{L}_i-1} \hat{f}_i(\bar{k}). \tag{2.31}$$

Similarly, the DFT of queue lengths at the j^{th} service completion epoch when Q_i is empty, $\hat{v}_{i,j}^0(\bar{k}_i)$ can be calculated by taking the average of $\hat{v}_{i,j}^0(\bar{k})$ with respect to the variable k_i for each $j = 1, \ldots, \mathcal{K}_i$. Next the empty system probabilities, $g_i(\bar{0})$, $i = 1, \ldots, N$, can be evaluated by taking the average of $\hat{g}_i(\bar{k})$ with respect to all N variables, i.e.,

$$\hat{g}_i(0) = \frac{1}{\mathcal{L}_1 \cdots \mathcal{L}_N} \sum_{k_1=0}^{\mathcal{L}_1-1} \cdots \sum_{k_N=0}^{\mathcal{L}_N-1} \hat{g}_i(\bar{k}). \tag{2.32}$$

We are now ready to state the following key Theorem, concerning the PGF style mappings for the DFTs.

Theorem 2.3 *The DFT of the steady state probabilities of queue lengths at polling instants, $\hat{f}_i(\bar{k})$, satisfies the mapping, Ω_i, defined for PGFs in equation 2.26 with the PGFs replaced by the appropriate DFTs, i.e.,*

$$\hat{f}_i(\bar{k}) = \Omega_i(\hat{f}_i(\bar{k})), \qquad \forall\, \bar{k} \in \mathcal{S}. \tag{2.33}$$

Proof: Recall that, we get the PGF of the steady state probabilities of queue lengths at polling instants by applying the mapping, $\Omega_i(\cdot)$, infinite times (in a nested manner) to the PGF of an arbitrary queue length distribution. Then the resultant PGF satisfies the relationship (2.26). Furthermore, for a random distribution with discrete and finite state space its DFT is a discrete sampling of its PGF that represents it uniquely (see section 10.7 of Poularikas and Seely,1991).

Although our model assumes infinite buffer sizes, for all practical purposes we can assume that the set, $\mathcal{S}$, represents the system state space, since the probability of observing a system state, $\bar{n} \notin \mathcal{S}$, is very small (i.e., less than ϵ). Thus, there exists a unique DFT for each queue length distribution at polling instants, $f^i(\bar{n})$, $i = 1, \ldots, N$, and these DFT's correspond to the discrete sampling functions of the PGF's of the queue length distributions, i.e., $f_i(\bar{z})$, $i = 1, \ldots, N$. Therefore, unique discrete sampling of the PGF of the steady state distribution of queue lengths at type-i polling instants, i.e., $\hat{f}_i(\bar{k})$, would satisfy relationship (2.26), as well. Hence it is proved. #

The following iterative algorithm is based on Theorem 2.3, and assumes that the critical values, $\mathcal{L}_i$ and $\mathcal{K}_i$, $i = 1, \ldots, N$, are known.

Algorithm:

0. Calculate $\hat{S}_i^*(\bar{k})$ and $\hat{T}_i^*(\bar{k})$ $\forall$ $\bar{k} \in \mathcal{S}$, $i = 1, \ldots, N$. Set the initial state (arbitrarily) to idle system with server resting at station 1, i.e., $\hat{g}_1(\bar{k}) = 1$, $\hat{g}_j(\bar{k}) = 0$, $j > 1$, and $f_1(\bar{k}) = 0$, $\forall\, \bar{k} \in \mathcal{S}$.

1. For $i = 1$ to N do: $\hat{f}_{i+1}(\bar{k}) = \mathcal{M}_i(\hat{f}_i(\bar{k}))$ $\forall\, \bar{k} \in \mathcal{S}$.

2. If $\hat{f}_1(\bar{k})$ converges $\forall\, \bar{k} \in \mathcal{S}$, then continue. Else go to step (1).

3. For $i = 1$ to N do:

   (i) Invert $\hat{f}_i(\bar{k})$ to get the joint probabilities of queue lengths at polling instants, $f^i(\bar{n})$, $i = 1, \ldots, N$ and $\bar{n} \in \mathcal{S}$.

   (ii) Calculate the marginal probabilities of queue length at polling instants, $f_i(n_i)$, $i = 1, \ldots, N$, which also includes the empty station probability,

$f_i(0)$. Then, calculate the factorial moments of the queue length at polling instant of Q_i.

(iii) Calculate the empty system probability, $g_i(\bar{0})$ using equation (2.32). If $(i < N)$, then generate $\hat{f}_{i+1}(\bar{k})$ utilizing the mapping $\mathcal{M}_i$.

4. Calculate the expected customer waiting times by substituting the values from step (3) in equation (2.5).

The algorithm can be adapted to solve several variations of this base model. For example, continuous roving, non-zero switchover times and gated service are readily handled. We discuss these extensions in section 2.6.

2.4.3 The Approximate Methods

We develop two approximate algorithms. One is a modification of approximation scheme (B) of Gupta and Srinivasan (1996) which they developed for the continuously roving server model. We call this one a *mean approximation*, since it approximates (at the steady state) only the mean of the setup times of the actual model. The other one also emerges from a similar idea, but it considers the higher moments of the setup times, as well. Thus, we call this approximation a *mean-variance approximation*. Before starting to discuss these approximations, we want to define a new variable: setup time per cycle. This variable has a different distribution than the (original) system parameter, setup time. Setup time per cycle is zero with some probability (the probability of no-setup per cycle) and equals the setup time with one minus the "no-setup" probability.

Mean Approximation

This approximation assumes state independence of setups, but scales the original setup time, T_i, to $(1 - f_i(0))T_i$. Notice that $(1 - f_i(0))E[T_i]$ is nothing but the long run average amount of setup time that the server spends at Q_i per cycle. However the probabilities, $f_i(0)$, $i = 1, \ldots, N$, are unknown. In order to estimate these probabilities, as well as the empty system probabilities, we use the following iterative procedure.

Suppose that the current iteration index is k, $k \geq 1$, and that $f_i^{(k-1)}(0)$, $i = 1, \ldots, N$, are known. Then the scaled values of the setup times during the k^{th} iteration are

$$T_i^{(k)} = (1 - f_i^{(k-1)}(0))T_i, \qquad i = 1, \ldots, N, \tag{2.34}$$

and the PGF of the sum of contributions of all original customers who arrive during a type-i setup time which is c cycles prior to the reference point at iteration k, $\mathcal{T}_{i,c}^{(k)}(z)$, is obtained from equation (2.15).

Once the $T_{i,c}^{(k)}$ terms have been calculated, we solve a patient server model with state independent setups (see Srinivasan and Gupta, 1996, for details). The empty station polling probabilities, $f_i^{(k)}(0)$, can be calculated by setting $z = 0$ in the $f_i(1, \ldots, z, 1, \ldots, 1)$ expression, i.e.,

$$f_i^{(k)}(0) = \prod_{c=0}^{\infty} \prod_{j=1}^{N} \mathcal{T}_{j,c}^{(k)}(0) - \sum_{c=0}^{\infty} \sum_{j=1}^{N} \vartheta_j^{(k)} J_{j,c}(0) \sigma_{j,c}^{(k)}(0), \tag{2.35}$$

where $\sigma_{j,c}^{(k)}(0)$ is calculated using LST of the scaled setup distribution at iteration k.

We start the iterative procedure with $f_i^{(0)}(0) = 0$, $i = 1, \ldots, N$, and stop when all $f_i^{(k)}(0)$ values, $i = 1, \ldots, N$, converge within a specified tolerance. The

convergence occurred in 20 iterations or less in all our experiments during the course of extensive numerical testing. Finally, the expected waiting times are calculated by substituting $g_i^{(\cdot)}(\bar{0})$, $f_i^{(\cdot)}(0)$, and the moments of $\mathcal{F}_i^{(\cdot)}(z)$, i.e., $f_i^{(1)}$ and $f_i^{(2)}$, obtained from the final iteration, into equation (2.5).

Mean-Variance Approximation

Notice that scaling the setup time distribution in the mean (M) approximation not only reduces the mean setup time per cycle, but also decreases the variance of the setup time per cycle. From equation (2.34) we get $\mathrm{VAR}(T_i^k) = (1 - f_i^{(k-1)}(0))^2\mathrm{VAR}(T_i)$. However, in the actual model while the mean setup time per cycle is less than the original mean of the setup time, the variance of setup time per cycle is greater than the variance of the (state independent) setup time. For example, even for the constant setup times, the state-dependent setup policy introduces variance to the setup time per cycle. We know that in steady state the server sets up at station i with probability $(1 - f_i(0))$ and skips that station with probability $f_i(0)$. Using this observation, we develop the mean-variance (M-V) approximation method to calculate the mean waiting times. The M-V algorithm also assumes state-independent setup times, but instead of scaling the setup times, it calculates the setup time per cycle distribution using the current iteration "setup" and "not setup" probabilities, i.e., $1 - f_i^{(k-1)}(0)$ and $f_i^{(k-1)}(0)$. That is, for $i = 1, \ldots, N$, and $c \geq 0$ we calculate

$$\mathcal{T}_{i,c}^{(k)}(z) = f_i^{(k-1)}(0) + (1 - f_i^{(k-1)}(0))T_i^*\Big(\sum_{j=i}^{N}[\lambda_j - \lambda_j L_{j,c}(z)] + \sum_{j=0}^{i-1}[\lambda_j - \lambda_j L_{j,c-1}(z)]\Big), \tag{2.36}$$

and use them in equation (2.35) to update the empty station polling probabilities. The algorithm starts with $f_i^{(0)}(0) = 0$, $i = 1, \ldots, N$, and terminates when the convergence

on $f_i^{(k)}(0)$ is achieved at some $k > 0$, for all $i = 1, \ldots, N$.

2.5 Numerical Tests

Numerical experiments are performed to gain a better understanding of the DFT based approach as well as the iterative approximation scheme of section 2.4.3. Since testing involved a large number of data sets, we start this section by describing the test data. Later, we report the results of the analysis in the form of three tables. The accuracy of the approximation is tested by comparing its results with those obtained from a computer simulation of the corresponding system. We also report on the limitations of each method.

The test data varies in terms of number of stations, total traffic intensity, service and setup time distributions, degree of symmetry in the system and relative magnitude of the setup times with respect to the service times. We consider 5 and 10 station problems for testing the approximation. However, for the DFT based method, N is chosen to be either 2 or 5. For all problem sets the arrival rate of each customer type is kept constant at 1.0 and the load is varied by changing the mean service time at stations. We test the problems for low (L), medium (M) and high (H) total work load, which are 0.1, 0.5 and 0.9 respectively. Either constant or exponential setup and service time distributions are used.

Our data labeling scheme consists of a seven character alphanumeric code. The first two digits of the code denote the number of stations, N. The third digit is used for the total work load of the system (L, M or H). The fourth digit indicates whether or not the system consists of symmetric stations. A system is called

symmetric, denoted by S, if all the stations are identical. In contrast, an asymmetric system, denoted by A, has $\rho_1 = 0.9\rho$ and $\rho_j = 0.1\rho/(N-1)$, for $j = 2,\ldots,N$. The relative magnitude of the mean setup time to the mean service time is indicated by the fifth digit: the fifth digit values of 0, 1 and 2 correspond to the problem instants in which the mean setup time is one tenth of, equal to, and ten times the mean service time, respectively. The last two digits are reserved for service time and setup time distributions; 0 (constant) or 1 (exponential). This way we obtain twenty four different problem sets for each problem size, i.e., for each (N, ρ) couple.

In order to evaluate the performance of the approximation and of the DFT based approach, we have also simulated the system operation using an event-driven simulation program written in Ansi C. The simulation program is run on a 486 PC, with 60 *MHz* clock frequency and it takes anywhere from 0.25 hours to 4 hours depending on the size of the data set, N, and the total traffic intensity, ρ. Computer codes of the approximation and the numerical method are also written in Ansi C. For the matrix and DFT inversions required in our methods, we used standard subroutines from the book *"Numerical Recipes in C"* (Press *et al.* 1992).

2.5.1 Test Results For The DFT Based Algorithm

The DFT based method yields near exact values of system performance measures, if $\mathcal{L}_i$ and $\mathcal{K}_i$, $i = 1,\ldots,N$, are chosen carefully. Of these, the choice of suitable $\mathcal{L}_i$ values is extremely important. If chosen $\mathcal{L}_i$ values are too small, then large errors can result from having an excessively truncated state space. If chosen $\mathcal{L}_i$ values are too large, extensive computer resources are needed to perform computations.

We use the following procedure for obtaining initial estimates of $\mathcal{L}_i$'s (Leung,

1991). First, the mean customer waiting times are estimated from the approximation of section 2.4.3. From this, the expected station queue lengths are found using Little's Law. We equate $E[\mathcal{L}_i]$, $i = 1, \ldots, N$, to these expected queue lengths. Next, the server utilization, $\tilde{\rho}_i$, for an equivalent M/M/1 queue which has the same expected queue length is calculated, i.e., $\tilde{\rho}_i$ is chosen such that $E[\mathcal{L}_i] = \tilde{\rho}_i/(1-\tilde{\rho}_i)$. Finally, we set the $\mathcal{L}_i$ value such that probability of observing a queue length greater than or equal to $\mathcal{L}_i$ in the M/M/1 model is less than ϵ (which is set to 10^{-8}). Put differently, the chosen $\mathcal{L}_i$ satisfies $\tilde{\rho}_i^{\mathcal{L}_i} < \epsilon$.

The goodness of our initial estimates for $\mathcal{L}_i$s can be ascertained by checking the marginal queue length probabilities obtained from the DFT based algorithm. We expect these probabilities to reduce to ϵ or less as the queue size approaches $\mathcal{L}_i - 1$. If this does not happen, the algorithm is rerun, after increasing $\mathcal{L}_i$ values. In our experiments, we doubled the initial $\mathcal{L}_i$ estimate for such cases.

In the implementation of the DFT based method, instead of performing exactly $\mathcal{K}_i$ services at each server visit to Q_i, we let the server continue serving type-i customers until Q_i is empty. In other words, we repeatedly evaluate equation (2.23) until $\hat{v}_{i,j}(\bar{k})$ converges for all $\bar{k} \in \mathcal{S}$.

The CPU time and computer memory requirements necessary to implement the algorithm grow exponentially in N. These requirements also grow in ρ since we have to increase $\mathcal{L}_i$s accordingly. Both the time complexity and the space requirement of the algorithm are exponential in the number of stations, N. Therefore, even for $N = 6$ the algorithm becomes inefficient. Table 2.1 presents the results for $N = 2$ and $N = 5$. The mean waiting times calculated by the exact method (for $N = 2$) and the simulation (for $N = 5$) are also included in the table in parentheses. The

Table 2.1: A sample of mean waiting times obtained from the DFT based algorithm. Values shown in parentheses are exact results for $N = 2$ and the simulation results for $N = 5$. Experiments are performed on a SUN SPARC 2 workstation with 80 *MHz* clock speed.

Code	$(L_1, \cdots, L_N)$	$(E[W_1], E[\widehat{W_j}])$	CPU time (min.)	Memory (Kbytes)
05LS011	(8,8,8,8,8)	0.0040,0.0040 (0.0040,0.0040)	17	7788
05LA101	(16,8,8,8,8)	0.0235,0.0265 (0.0238,0.0252)	50	14400
05MS011	(8,8,8,8,8)	0.0333,0.0333 (0.0332,0.0332)	24	7788
02MS011	(32,32)	0.2762,0.2762 (0.2760,0.2760)	1	450
02MA201	(64,64)	4.0694,6.6296 (4.0690,6.6328)	11	1800

same data code is used with a minor change; for data sets with $N = 5$, "M" as the third character of the code represents $\rho = 0.3$, instead of $\rho = 0.5$. We report expected waiting times as $(E[W_1], E[\widehat{W_j}])$, where $E[\widehat{W_j}]$ represents the average of the mean waiting times of type-j customers for $j = 2, \ldots, N$.

As seen in table 2.1, when $N = 2$ choosing $\mathcal{L}_i$ values as high as 64 does not create excessive requirements for either CPU time or memory. However, this changes when $N = 5$. Large CPU times are necessary whenever $\mathcal{L}_i > 8$. Because of the memory limitations of PC's, the DFT based algorithm was run on a SUN SPARC 2 workstation with a clock speed of 80 *MHz*.

As a final remark about the DFT based algorithm we would like to point out that for the purpose of carrying out numerical tests, the computer implementation of the algorithm was not optimized and no special data structure was used. The

algorithm is also suitable for parallel computing. All these measures and the availability of greater computing power can substantially improve the performance of the DFT based algorithm, making it practical even for systems with many stations and large ρ.

2.5.2 Test Results For Approximate Methods

The approximate method is tested over 144 different data sets. Since there is no *work conservation* relationships available for state dependent setup models, simulation is used as a benchmark for comparison. We let these simulations go on until the 95% C.I. on the mean waiting times is in the $\pm 0.5\%$ neighborhood of its point estimate. In contrast, both Ferguson (1986) and Bradlow and Byrd (1987) use time limited and/or event limited stopping criteria. Our computational experience has shown that such criteria do not necessarily yield the steady state performance measures. A major difference between our experiments and those reported in previous studies (Bradlow and Bryd, 1987, Ferguson, 1986) is that the latter consider only "heavily loaded" systems and do not report empty station and/or empty system probabilities. In contrast, we examine systems with a range of values of overall server utilization and report empty station and empty system probabilities. We believe that in addition to having a direct impact on the accuracy of the estimate of the mean waiting time, these probabilities are important performance measures. For example, if the empty system probabilities are very small, then we know that the system behaves like a continuously roving server model (see Observation 2.3 in section 2.3) and can be approximated as such.

In tables 2.2 and 2.3 we report the mean, the standard deviation and the

worst case values of the absolute errors for the mean approximation and the mean-variance approximation, respectively, for the expected waiting time at station 1. The same three statistics are also reported for the average of the mean waiting times at all other stations, which we denote by $\widehat{E[W_j]}$, in both tables. These descriptive statistics are calculated after first representing each absolute error as a percentage of the corresponding value obtained from simulation. For the empty station and empty system probabilities, similar statistics are calculated (for both methods) over their absolute errors, since these values, rather than percent errors, better reflect their effect on the accuracy of the mean waiting time estimates. The terms $\widehat{f_j(\vec{0})}$ and $\widehat{g_j(\vec{0})}$ denote respectively the average of the empty station and empty system probabilities when the server is at Q_j, $j = 2, \ldots, N$. Recall that even for the "asymmetric" problem sets, stations $Q_2 \cdots Q_N$ are identical and therefore little information is lost by such averaging.

Mean Approximation: The absolute percent error in $E[W_1]$ over all data sets has a mean of 5.80%, and a standard deviation of 7.28%. In the worst case, the absolute percent error can be as high as 30.77%. However, such instances are rare and happen when low overall server utilization is combined with very large setup times, i.e., when the mean setup time is ten times as large as the mean service time at the corresponding station. Note that such instances have not been studied before. For example, Bradlow and Byrd (1987) consider cases in which the setup times are at most of the same magnitude as the service times. Still, they report 10% to 15% error in their estimates of the mean waiting times for low load ($\rho = 0.3$) systems. If such extreme cases are excluded from our data set and we then consider only low load systems, i.e., when $\rho = 0.1$, the absolute percent error in $E[W_1]$ has a mean of 1.16% and a standard

deviation of 1.70% (for $E[\widehat{W_j}]$ these statistics are 1.44% and 1.49% respectively). If examples having proportionally very large setup times are eliminated from all (N, ρ) combinations, then the overall average of the absolute percent error in mean waiting times lies well within 4%.

Mean-Variance Approximation: The absolute percent error in $E[W_1]$ over all data sets (with $N = 5$) has a mean of 2.68%, and a standard deviation of 2.64%. In the worst case, the absolute percent error can be as high as 10.23%. The performance of the approximation for $E[\widehat{W_j}]$ is slightly worse; the mean and standard deviation of the set of absolute percent errors are 3.15% and 3.16%, respectively. In the worst case, the absolute percent error for $E[\widehat{W_j}]$ can be as high as 17.42%.

Tables 2.2 and 2.3 also show that both iterative approximations are very effective in estimating the empty station and the empty system probabilities. In most cases these probabilities can be estimated to lie within two digit accuracy. The absolute error in $f_i(0)$ has a mean of 0.012 and a standard deviation of 0.019 and the absolute error in $g_i(\bar{0})$ has a mean of 0.024 and a standard deviation of 0.018. This observation explains why the difference between two approximations is more for low traffic systems. In both methods the basic approximation idea is the same: iteratively forcing the moments of the setup time per cycle in the approximate model to converge to that of the actual system. Accurate estimates of probabilities (i.e., $f_i(0)$ and $g_i(\bar{0})$) indicate that the approximate model acts very close to the actual system. However, since the mean approximation poorly estimates the second moment of the setup time per cycle variable, its error in calculation of the mean waiting times is higher. Notice that $f_i(0)$ is high for low traffic intensity data sets. Thus the mean approximation method's error in estimating the variance of the setup time per cycle is higher than

Table 2.2: The performance statistics of the mean approximation. Statistics are calculated on the absolute percent error for mean waiting times and on the absolute error for empty station and empty system probabilities.

N	ρ		$E[W_1]$	$\widetilde{E[W_j]}$	$f_1(0)$	$\widehat{f_j(0)}$	$g_1(\vec{0})$	$\widehat{g_j(\vec{0})}$
5	L	Mean	6.67	6.76	0.010	0.010	0.067	0.089
		Std.	8.27	8.18	0.009	0.010	0.026	0.028
		Worst	22.45	22.20	0.027	0.029	0.104	0.118
	M	Mean	3.72	3.90	0.006	0.008	0.017	0.033
		Std.	4.59	4.69	0.006	0.008	0.017	0.033
		Worst	15.14	15.40	0.017	0.025	0.051	0.096
	H	Mean	4.12	4.80	0.008	0.012	0.003	0.014
		Std.	2.80	3.77	0.009	0.017	0.003	0.022
		Worst	10.22	17.41	0.032	0.054	0.011	0.065
10	L	Mean	9.11	9.45	0.013	0.013	0.015	0.025
		Std.	11.74	11.63	0.014	0.015	0.012	0.012
		Worst	30.77	30.75	0.035	0.038	0.032	0.036
	M	Mean	6.17	6.81	0.007	0.011	0.005	0.010
		Std.	8.27	8.40	0.007	0.011	0.005	0.010
		Worst	26.31	26.42	0.022	0.031	0.014	0.029
	H	Mean	5.01	6.28	0.018	0.025	0.002	0.007
		Std.	3.84	3.74	0.032	0.046	0.002	0.010
		Worst	12.99	13.74	0.096	0.131	0.006	0.028
Overall		Mean	5.80	6.34	0.010	0.013	0.018	0.029
		Std.	7.28	7.34	0.016	0.022	0.014	0.021
		Worst	30.77	30.75	0.096	0.131	0.104	0.118

for high traffic data sets. Therefore, the mean-variance approximation performs much better for low traffic data sets (see Table 2.2 and 2.3 for ρ=L row).

Table 2.3: Performance statistics of the mean-variance approximation, for five station data sets. Statistics are calculated on the absolute percent error for mean waiting times and on the absolute error for empty station and empty system probabilities.

N	ρ		$E[W_1]$	$\widehat{E[W_j]}$	$f_1(\bar{0})$	$\widehat{f_j(\bar{0})}$	$g_1(\bar{0})$	$\widehat{g_j(\bar{0})}$
5	L	Avg.	1.95	2.01	0.010	0.011	0.070	0.093
		Std.	2.25	2.41	0.011	0.011	0.025	0.025
		Max.	6.76	7.34	0.031	0.033	0.104	0.118
	M	Avg.	2.12	2.49	0.008	0.010	0.015	0.038
		Std.	2.32	2.46	0.008	0.008	0.017	0.034
		Max.	7.23	7.64	0.030	0.027	0.051	0.096
	H	Avg.	3.96	4.94	0.007	0.012	0.003	0.013
		Std.	2.83	3.63	0.008	0.015	0.003	0.022
		Max.	10.23	17.42	0.030	0.050	0.011	0.065
Overall (5)		Mean	2.68	3.15	0.008	0.011	0.029	0.048
		Std.	2.64	3.16	0.012	0.012	0.042	0.043
		Worst	10.23	17.42	0.031	0.049	0.104	0.118

Table 2.4 lists the station mean waiting times obtained from the simulation and both approximation procedures for problem instants with $N = 5$. From the table, it is clear that the mean-variance approximation outperforms the mean approximation method. There are only a few instants where mean approximation performs slightly better. This might be due to the error in the simulation point estimate. Therefore, we strongly believe that mean-variance approximation is superior. Also, the error in estimating $E[W_1]$ and $\widehat{E[W_j]}$ is small for all problem instants and the approximation is not biased in the sense of estimating mean waiting times more accurately for some stations. Furthermore, figures 2.1 and 2.2 present the same output graphically for the

mean approximation. Note that, in those figures the simulation results are denoted by "*" and the approximation results are joined by the broken line.

The approximate algorithm utilizes a descendant-sets-based recursive algorithm with the result that it is very fast when compared to either the simulation or the DFT-based method. It takes less than 4 minutes to run 24 data sets on a 486 PC. The simulation of a single data set could take between 15 minutes to 4 hours on the same PC. However, if the problem of interest has low load and large setup times in relation to service times, we recommend that the DFT-based algorithm be used. Although this would require considerable computer resources, these would still be less than that required by the simulation. Furthermore, the DFT-based algorithm provides not only the mean waiting times, but also the higher moments of waiting times with negligible extra effort.

Table 2.4: A comparison of the mean waiting times obtained from the mean (M) and mean-variance (M-V) approximations with those obtained from the computer simulation.

Data Code	$E[W_1]$					$E[W_j]$				
	Sim	M	% Err	M-V	% Err	Sim	M	% Err	M-V	% Err
05LS000	0.003	0.003	-0.32	0.003	-0.16	0.003	0.003	-0.14	0.003	0.02
05LS001	0.003	0.003	-0.15	0.003	0.17	0.003	0.003	-0.45	0.003	-0.13
05LS010	0.004	0.004	-0.03	0.004	0.09	0.004	0.004	-0.24	0.004	-0.12
05LS011	0.004	0.004	-0.23	0.004	0.01	0.004	0.004	0.11	0.004	0.35
05LA000	0.006	0.006	-0.10	0.006	-0.03	0.007	0.007	-0.28	0.007	-0.21
05LA001	0.006	0.006	-0.10	0.006	0.05	0.007	0.007	-0.79	0.007	-0.65
05LA010	0.011	0.011	-0.34	0.011	-0.29	0.012	0.012	0.15	0.012	0.19
05LA011	0.011	0.011	-0.85	0.011	-0.77	0.012	0.012	-0.57	0.012	-0.49
05LS100	0.020	0.020	-0.97	0.020	1.39	0.020	0.020	-0.90	0.020	1.45
05LS101	0.021	0.020	-3.97	0.021	0.48	0.021	0.020	-3.85	0.021	0.60
05LS110	0.021	0.021	-0.79	0.021	1.43	0.021	0.021	-0.94	0.021	1.28
05LS111	0.022	0.021	-3.44	0.022	0.78	0.022	0.021	-3.82	0.022	0.38
05LA100	0.023	0.023	-0.82	0.023	1.20	0.024	0.024	-1.09	0.024	0.86
05LA101	0.024	0.023	-3.38	0.024	0.47	0.025	0.024	-3.49	0.025	0.21
05LA110	0.027	0.027	-0.71	0.027	0.97	0.029	0.029	-1.13	0.030	0.46
05LA111	0.028	0.027	-2.62	0.028	0.62	0.030	0.029	-3.14	0.030	-0.08
05LS200	0.330	0.286	-13.37	0.319	-3.39	0.330	0.286	-13.40	0.319	-3.43
05LS201	0.414	0.323	-22.05	0.387	-6.41	0.414	0.323	-22.15	0.387	-6.53
05LS210	0.331	0.287	-13.18	0.320	-3.24	0.331	0.287	-13.26	0.320	-3.34
05LS211	0.415	0.324	-21.96	0.388	-6.40	0.416	0.324	-22.19	0.388	-6.67
05LA200	0.313	0.272	-13.05	0.305	-2.65	0.340	0.295	-13.10	0.328	-3.38
05LA201	0.398	0.309	-22.25	0.373	-6.24	0.425	0.332	-21.72	0.397	-6.44
05LA210	0.318	0.276	-13.02	0.309	-2.81	0.347	0.301	-13.17	0.334	-3.66
05LA211	0.404	0.313	-22.45	0.376	-6.76	0.434	0.338	-22.20	0.402	-7.34
05MS000	0.069	0.068	-1.01	0.068	-0.72	0.069	0.068	-1.33	0.068	-1.03
05MS001	0.070	0.068	-2.42	0.068	-1.85	0.070	0.068	-2.19	0.068	-1.62
05MS010	0.119	0.119	-0.07	0.119	0.09	0.119	0.119	-0.66	0.119	-0.50
05MS011	0.121	0.119	-1.92	0.119	-1.61	0.121	0.119	-1.94	0.119	-1.63
05MA000	0.198	0.198	0.09	0.198	0.16	0.381	0.377	-1.18	0.377	-1.13
05MA001	0.199	0.198	-0.47	0.198	-0.33	0.378	0.377	-0.43	0.377	-0.32
05MA010	0.380	0.384	1.05	0.384	1.09	0.726	0.731	0.69	0.732	0.72
05MA011	0.390	0.384	-1.65	0.384	-1.58	0.746	0.732	-1.90	0.732	-1.85
05MS100	0.327	0.289	-11.58	0.304	-7.11	0.327	0.289	-11.72	0.304	-7.26
05MS101	0.360	0.306	-15.14	0.334	-7.23	0.362	0.306	-15.40	0.334	-7.52
05MS110	0.383	0.343	-10.55	0.357	-6.89	0.384	0.343	-10.64	0.357	-6.99
05MS111	0.415	0.359	-13.33	0.386	-6.79	0.418	0.359	-14.12	0.386	-7.64

Table 2.4: (continued).

Data Code	$E[W_1]$					$E[W_j]$				
	Sim	M	% Err	M-V	% Err	Sim	M	% Err	M-V	% Err
05MA100	0.351	0.332	-5.26	0.344	-2.01	0.669	0.625	-6.67	0.642	-4.14
05MA101	0.382	0.347	-9.07	0.369	-3.31	0.714	0.645	-9.75	0.677	-5.16
05MA110	0.535	0.513	-4.02	0.524	-1.91	1.022	0.979	-4.24	0.995	-2.61
05MA111	0.562	0.526	-6.33	0.548	-2.48	1.050	0.997	-5.08	1.029	-2.03
05MS200	4.543	4.548	0.10	4.535	-0.19	4.546	4.548	0.04	4.535	-0.26
05MS201	5.050	5.030	-0.40	5.035	-0.31	5.065	5.030	-0.69	5.035	-0.61
05MS210	4.620	4.597	-0.49	4.581	-0.84	4.603	4.597	-0.12	4.581	-0.47
05MS211	5.043	5.076	0.66	5.081	0.77	5.009	5.076	1.33	5.081	1.44
05MA200	2.927	2.933	0.21	2.917	-0.32	5.287	5.277	-0.19	5.231	-1.06
05MA201	3.440	3.384	-1.62	3.398	-1.21	5.900	5.881	-0.31	5.904	0.08
05MA210	3.140	3.117	-0.72	3.097	-1.36	5.687	5.629	-1.01	5.575	-1.97
05MA211	3.599	3.560	-1.08	3.578	-0.60	6.352	6.221	-2.06	6.248	-1.64
05HS000	1.099	1.045	-4.98	1.046	-4.86	1.103	1.045	-5.26	1.046	-5.14
05HS001	1.122	1.047	-6.67	1.049	-6.45	1.125	1.047	-6.93	1.049	-6.71
05HS010	2.012	1.858	-7.67	1.859	-7.62	2.017	1.858	-7.88	1.859	-7.83
05HS011	1.986	1.860	-6.36	1.861	-6.27	1.985	1.860	-6.30	1.861	-6.20
05HA000	2.280	2.326	2.03	2.327	2.05	12.216	12.698	3.94	12.700	3.96
05HA001	2.268	2.327	2.59	2.328	2.63	12.207	12.701	4.04	12.705	4.08
05HA010	4.174	4.601	10.22	4.601	10.23	21.406	25.133	17.41	25.135	17.42
05HA011	4.358	4.601	5.58	4.602	5.60	23.162	25.135	8.52	25.139	8.53
05HS100	4.705	4.453	-5.35	4.440	-5.63	4.720	4.453	-5.64	4.440	-5.92
05HS101	4.550	4.527	-0.50	4.530	-0.43	4.577	4.527	-1.09	4.530	-1.02
05HS110	5.150	5.192	0.82	5.173	0.45	5.158	5.192	0.67	5.173	0.30
05HS111	5.117	5.256	2.71	5.263	2.85	5.123	5.256	2.59	5.263	2.73
05HA100	3.046	3.000	-1.52	3.003	-1.41	15.360	16.054	1.22	16.073	1.34
05HA101	2.949	3.041	3.13	3.063	3.85	15.753	16.251	3.16	16.360	3.85
05HA110	4.998	5.238	4.80	5.247	4.98	27.055	28.281	4.53	28.328	4.70
05HA111	5.820	5.276	-9.35	5.304	-8.88	31.618	28.466	-9.97	28.603	-9.53
05HS200	37.037	37.710	1.82	37.710	1.82	37.064	37.710	1.74	37.710	1.74
05HS201	38.084	38.610	1.38	38.610	1.38	37.725	38.610	2.35	38.610	2.35
05HS210	37.295	38.520	3.28	38.520	3.28	37.271	38.520	3.35	38.520	3.35
05HS211	36.305	39.420	8.58	39.420	8.58	36.438	39.420	8.18	39.420	8.18
05HA200	11.037	10.824	-1.92	10.823	-1.94	57.841	56.423	-2.45	56.409	-2.47
05HA201	11.720	11.477	-2.07	11.478	-2.07	63.733	59.538	-6.58	59.539	-6.58
05HA210	12.510	13.099	4.71	12.660	1.20	68.920	68.859	-0.09	66.590	-3.38
05HA211	13.848	13.751	-0.70	13.752	-0.70	72.932	71.969	-1.32	71.970	-1.32

Figure 2.1: Mean Waiting Times at Station 1: simulation and approximation results.

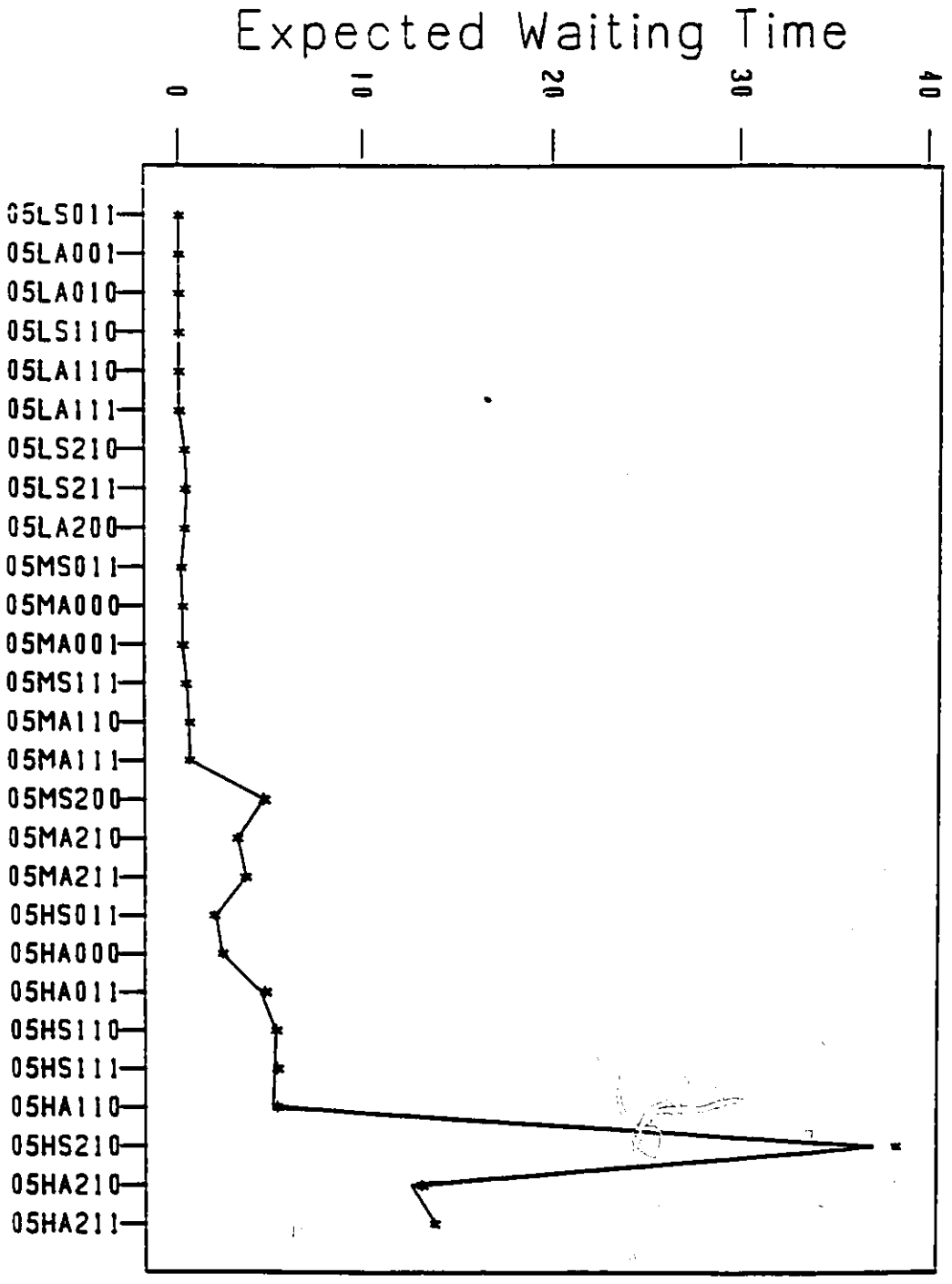
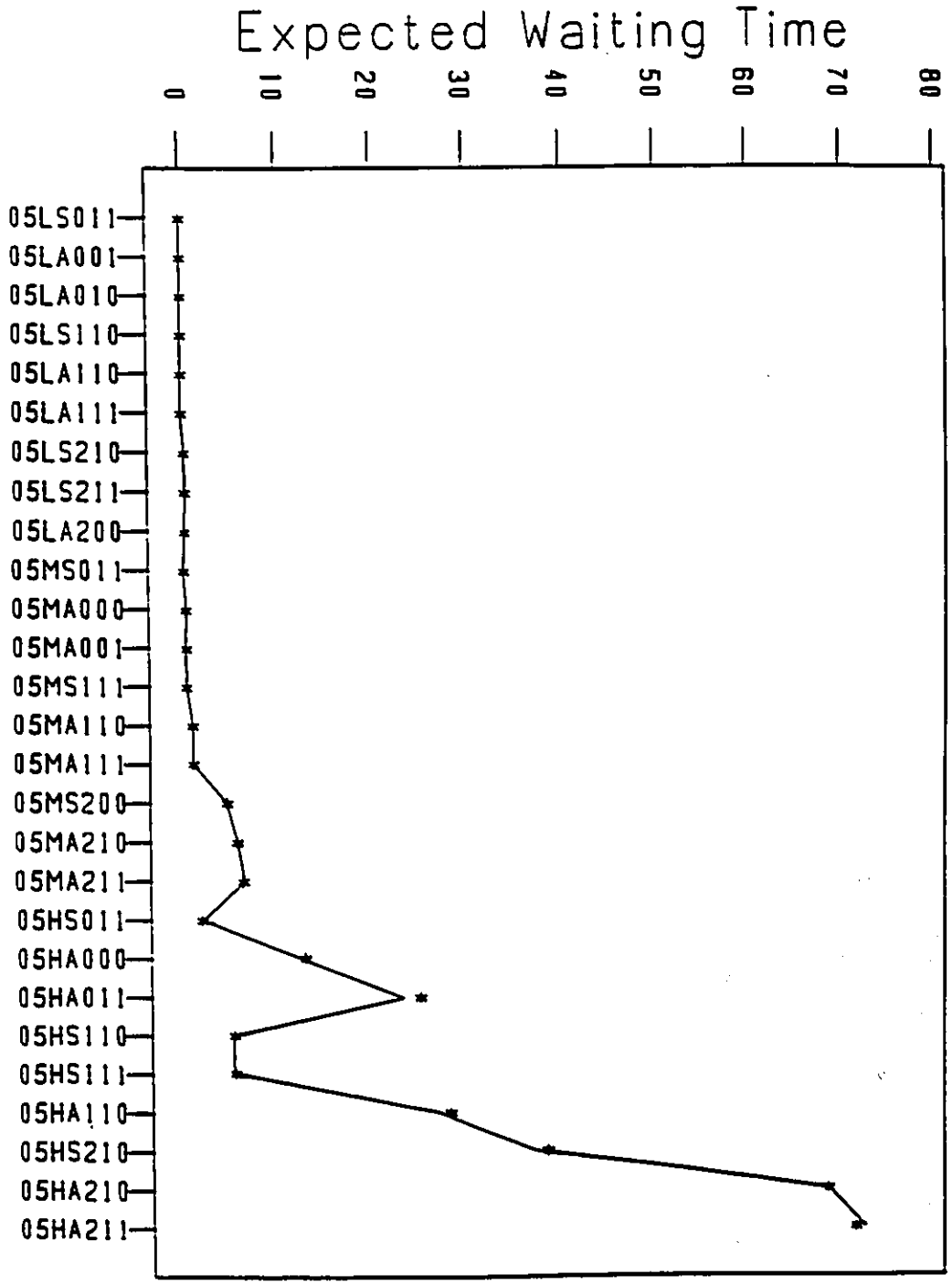


Figure 2.2: Mean Waiting Times at Stations 2, 3, 4, and 5: simulation and approximation results.



2.6 Extensions

Both the DFT based method and the iterative approximation can be used to analyze a variety of different models, involving nonzero switchover times, continuously roving server, state independent setups and for systems with some or all stations using the gated service regime. Before describing these extensions, we first introduce notation for nonzero switchover times.

Let R_i, $i = 1, \ldots, N$, denote the random switching time needed by the server to travel from station-i to station-$(i+1)$. The patient server polling system with both switchover and setup times has two types of station overheads: i) state dependent setup times and ii) state independent switchover times. Since the switchover times are state independent their presence in the system affects the customer waiting times in an additive fashion (see Srinivasan, Niu and Cooper, 1995). Therefore, the analysis of such system is similar to what we presented in section 2.3.

The DFT-based method can be modified to handle the continuously roving (CR) server polling system with state dependent setups. This is done by setting the empty system probabilities to zero in Theorem 2.2 and introducing nonzero switchover times. Next, we show the necessary changes needed to handle the gated service strategy. We assume that a gate closes behind all type-i customers present at station-i at its station beginning instant. The server processes only those customers that are in front of the gate before registering a station completion instant. Thus, the server vacation from station-i ends either by the end of a setup period or by a type-i arrival to the system when the server is idling at Q_i, and the relationship in equation (2.3) is still valid. The waiting time distribution for customers at station-i is found by applying

the Fuhrmann and Cooper's Stochastic decomposition theorem (see example 3 in Fuhrmann and Cooper, 1985).

$$W_i^*(\lambda_i - \lambda_i z) = \frac{K_i(S_i^*(\lambda_i - \lambda_i z)) - K_i(z)}{K_i'(1)} \left(\frac{1}{S_i^*(\lambda_i - \lambda_i z) - z}\right). \tag{2.37}$$

Upon taking the derivative of the LST above and then setting $s = 0$, i.e., $z = 1$, we obtain the mean waiting time of type-i customers as:

$$E[W_i] = \frac{(1+\rho_i)(f_i^{(2)} + 2\lambda_i \bar{t}_i f_i^{(1)} + (1 - f_i(0))\lambda_i^2 t_i^{(2)})}{2\lambda_i(f_i^{(1)} + (1 - f_i(0))\lambda_i \bar{t}_i + \vartheta_i p_i)}. \tag{2.38}$$

It once again depends on the first and second moments of queue lengths at polling instants. In order to find these moments we can use either one of the two methods described in section 2.4. However, during calculations we need to do the following:

i) PGF and corresponding DFT mapping relations have to be changed. For example, equations (2.23) and (2.24) are replaced with

$$g_i(\bar{z}) = \sum_{n_1=0}^{\infty} \cdots \sum_{n_i=0}^{\infty} \cdots \sum_{n_N=0}^{\infty} v^{i,0}(n_1, \ldots, n_i, \ldots, n_N) \left[S_i^*\left(\sum_{j=1}^{N} \lambda_j - \lambda_j z_j\right)\right]^{n_i} \prod_{\substack{j=1 \\ j \neq i}}^{N} z_j^{n_j}, \tag{2.39}$$

where $v^{i,0}(\bar{n}_i)$ gives the joint probabilities of queue lengths at station beginning instants, and can be calculated by inverting its DFT at each iteration. Note that for the gated service regime $g_i(\bar{z})$ is no longer independent of the i^{th} element, z_i.

ii) In the iterative approximation, the PGF of the contribution of a type-i customer

who is served at the c^{th} cycle is redefined as

$$L_{i,c}(z) = S_i^* \left(\sum_{j=i+1}^{N} [\lambda_j - \lambda_j L_{j,c}(z)] + \sum_{j=1}^{i} [\lambda_j - \lambda_j L_{j,c-1}(z)] \right). \quad (2.40)$$

For simplicity, the above expression has been shown without the iteration index, i.e., superscript k.

Chapter 3

A Polling System with Threshold Setup Control Policy

In this chapter we analyze polling systems with threshold setups, which is a generalization of the state-dependent setups model discussed in the previous chapter. In a threshold setups model each station has a minimum lot size, *threshold level* $h_i \geq 0$, and the server skips station i, if its queue length at the polling instant is less than h_i, $i = 1, \ldots, N$. Therefore, this model is not only a generalization of the state-dependent setups model, but also a generalization of the *N-policy* in M/G/1 queueing system with generalized vacations (see e.g., Takagi 1991).

This model is a logical extension of the state-dependent setups model, and has been considered in different fields, especially in *Stochastic Economic Lot Scheduling Problems* (S-ELSP). However, there are very few articles in the polling literature dealing with threshold setups models. Hofri and Ross (1987) studied the patient server polling model with threshold setups for two stations. They showed that an exhaustive service regime with individual thresholds for each station is the optimum strategy to follow in order to minimize the mean total number in the system, if service

time distributions are identical. They stated that the model becomes too difficult to analyze when there are more than two stations.

Polling models with threshold setups are suggested to analyze the S-ELSP (see e.g., Federgruen and Katalan, 1996, and Markowitz *et al.*, 1995). Federgruen and Katalan (1993) stated that the threshold setups polling model can be used to calculate the *economic lot sizes* of a multi-product manufacturing facility in a stochastic environment. They used an accurate queue length distribution approximation for general polling systems (see Federgruen and Katalan 1994) in their analysis of a class of cyclic production policies for the S-ELSP. Markowitz *et al.* (1995) acknowledged this observation and used the *heavy-traffic approximation* (Coffman, Puhalskii and Reiman, 1995) to analyze the threshold setups model. Their comparison of different production policies showed that the threshold setups control policy is not optimum for S-ELSP. We note that the threshold setups model defined by Markowitz *et al.* (1995) is different from our model in two ways: i) they used a dynamic server scheduling rule and ii) when the server polls a station which has less number of customers waiting than its threshold level, they let the server be idle, instead of skipping that station. In contrast, our model has a static cycling rule (i.e., the visiting sequence of stations and the threshold levels are fixed for all cycles) and the server never becomes idle. In the patient server variation of our model the server idling is allowed only when all stations have queue lengths less than their threshold level.

We assume exhaustive service discipline and modify the near-exact algorithm presented in the previous chapter to calculate the mean waiting times for a continuously roving server polling system with threshold setups and non-zero switchover times. For the case of gated service regime at some or all stations, the

changes to the algorithm are obvious. They are not presented here, since the gated or limited service regime with threshold setup server scheduling policy does not make much sense. However, we do state changes that would be necessary to apply our analysis to the patient server model with threshold setups. Note that our algorithm is not an optimization procedure; it is meant to be a tool for performance evaluation when model parameters (including the threshold levels) are given. But, it is possible to calculate the best combination of threshold levels to optimize the performance measure, by using our algorithm and totally enumerating all possible threshold values. More importantly, this analysis can be used to develop *fast* approximate/heuristic algorithms which could in turn be used to optimize system parameters. Then the "near-exact" algorithm can be used instead of simulation to test such methods.

This chapter is organized as follows. In section 3.1, the model is described and additional notation is introduced. Section 3.2 is used to derive expressions for the mean waiting times. We present the modified near-exact algorithm to calculate the queue length distribution at polling instants in section 3.3. Section 3.4 is reserved for discussion of computational issues and numerical examples. In the section 3.5, we mention the extension of our analysis to the patient server model. Notice that we do not provide an introduction section for this chapter. Since the model described in this chapter is a generalization of the state-dependent setups model, the related literature survey and the motivation of such models are already discussed in section 2.1.

3.1 Model Description

We consider a continuously roving server polling model with threshold setups, h_i, $i = 1, \ldots, N$, and non-zero switchover times. The service regime is exhaustive and customers are served in a FCFS manner. The observation epochs of the system are as defined in the previous chapter, with some minor modifications, since the server is continuously roving in this model. These changes are:

i) all switch points are also service completion epochs; we use only station completion epochs (from this couple) to describe the system,

ii) station beginning instants are only marked at the end of setups. Therefore, if we skip a station (due to the threshold level) we do not observe a station beginning instant for that station in that cycle. However, every station has exactly one polling instant in each cycle.

We utilize Markov chains imbedded at polling instants and station completion epochs (switch points) to analyze queue lengths.

We are now ready to describe the system using the above observation epochs. At a station i polling instant the server checks the status of its queue. If the number of customers in station i is at least the threshold level, h_i, then the server sets up and starts serving customers exhaustively. Otherwise, if there are not enough accumulated customers at station i the server immediately switches to the next station; a station i completion event is instantaneously marked.

In this chapter we use the same notation as in chapter 2, with some additions. The threshold levels, h_i, $i = 1, \ldots, N$, are already defined. Also, the system has non-

zero switchover times, which we denote by R_i, $i = 1, \ldots, N$. The random variable R_i is independent of other system parameters and has a general distribution. In addition to joint and conditional PGF's defined in chapter 2, we also define several *partial* PGF's. Let $f_{i,n_i}(\bar{z})$ denote the partial PGF of queue lengths given that there are n_i type-i customers at a polling instant, $n_i = 0, \ldots, h_i - 1$, and $i = 1, \ldots, N$. Then,

$$f_{i,n_i}(\bar{z}) = \sum_{n_1=0}^{\infty} \cdots \sum_{n_{i-1}=0}^{\infty} \sum_{n_{i+1}=0}^{\infty} \cdots \sum_{n_N=0}^{\infty} f^i(n_1, \ldots, n_i, \ldots, n_N) \prod_{j \neq i} z_j^{n_j}. \qquad (3.1)$$

Notice that, the above expression is independent of z_i, and therefore we implicitly set $z_i = 1$ for notational brevity. The probability of polling station i and finding n_i customers in its queue is $f_{i,n_i}(\bar{1})$, $n_i \geq 0$, $i = 1, \ldots, N$. Furthermore, let F_{i,h_i} represent the probability of not having a setup at a station-i polling instant. Put differently, it is the (total) probability of the queue length being less than h_i given a station i polling instant has occurred, i.e.,

$$F_{i,h_i} = \sum_{n_i=0}^{h_i-1} \frac{f_{i,n_i}(\bar{1})}{f_i(\bar{1})}, \qquad h_i \geq 0, \ \ i = 1, \ldots, N. \qquad (3.2)$$

Note that by definition an empty sum equals zero, and we have $F_{i,0} = 0$, whenever $h_i = 0$, i.e., for the state-independent setups model.

3.2 Mean Waiting Times

In this section, we focus on station 1 and its mean waiting time. The mean waiting time of other stations can be obtained from this result by rotating the station indices. We use the queue length distributions at station beginning instants to calculate the mean waiting times. Note that, for M/G/1 queues with (general) threshold setups,

the Fuhrmann-Cooper decomposition applies to the distribution of the number in the system, but not to the distribution of time in the system, unless $h_i = 1$, $i = 1, \ldots, N$ (see Exercise 12, part h, pp. 222-223 of Cooper 1990, and Fuhrmann and Cooper, 1985). However, as we show next, we can obtain expressions for the mean waiting times.

Let $\Pi_1(\bar{z})$ denote the PGF for the stationary distribution of queue lengths at a departure instant of a type-1 customer, and $K_1(z, 1, \ldots, 1)$ be the PGF of station 1 queue length at the end of a server vacation. Then using the Stochastic Decomposition Theorem (Fuhrmann and Cooper, 1985, Proposition 2) for $M/G/1$ queueing systems with generalized vacations, we obtain

$$\Pi_1(z, 1, \ldots, 1) = \frac{1 - K_1(z, 1, \ldots, 1)}{K_1'(\bar{1})} \times \frac{(1 - \rho_1) S_1^*(\lambda_1 - \lambda_1 z)}{S_1^*(\lambda_1 - \lambda_1 z) - z}. \tag{3.3}$$

In this model, when the server exhausts the queue at station 1, it starts a vacation from that station. The vacation ends at the end of the next setup at station 1. Since setups occur only when there are at least h_1 customers waiting in the queue at a station 1 polling instant, we can calculate $K_1(z, 1, \ldots, 1)$ as follows:

$$K_1(z) = \frac{\left[F_1(z, 1, \ldots, 1) - \sum_{n_1=0}^{h_1-1} F_{1,n_1}(\bar{1}) z_1^{n_1}\right] T_1^*(\lambda_1 - \lambda_1 z)}{1 - F_{1,h_1}}. \tag{3.4}$$

Substituting from above in equation (3.3) and simplifying we obtain

$$\begin{aligned} \Pi_1(z, 1, \ldots, 1) = & \frac{(1 - F_{1,h_1}) - \left[F_1(z, 1, \ldots, 1) - \sum_{n_1=0}^{h_1-1} F_{1,n_1}(\bar{1}) z_1^{n_1}\right] T_1^*(\lambda_1 - \lambda_1 z)}{F_1'(\bar{1}) - \sum_{n_1=0}^{h_1-1} n_1 F_{1,n_1}(\bar{1}) + (1 - F_{1,h_1}) \lambda_1 \bar{t}_1} \\ & \times \frac{(1 - \rho_1) S_1^*(\lambda_1 - \lambda_1 z)}{S_1^*(\lambda_1 - \lambda_1 z) - z}, \end{aligned} \tag{3.5}$$

Differentiating equation (3.5) with respect to z and then setting $z = 1$, we get the expected queue length of station 1 at a customer departure instant, which

is also the average queue length at station 1 at an arbitrary observation epoch (see e.g., Cooper 1990, pp. 186-188). Then, using Little's Law the mean waiting time of type-1 customers can be calculated from Theorem 3.1, which is presented below. Notice that, we assume *non-zero* switchover times in the theorem. Because it has not been shown yet that the mean waiting time decomposition similar to Cooper, Niu and Sirinivasan (1996) is valid for the state-dependent setup models. Thus, CR server models assumes switchover times to be non-zero, and the zero switchover times are considered in PS models which is discussed in section 3.5.

Theorem 3.1 *The mean waiting time of type-1 customers in a continuously roving server polling model with threshold setups and nonzero switchover times is*

$$E[W_1] = \frac{f_1^{(2)} + 2\lambda_1 \bar{t}_1 f_1^{(1)} + (1 - F_{1,h_1})\lambda_1^2 t_1^{(2)}}{2\lambda_1 (f_1^{(1)} + (1 - F_{1,h_1})\lambda_1 \bar{t}_1)} + \frac{\lambda_1 E[S_1^2]}{2(1-\rho_1)}, \tag{3.6}$$

where $f_1^{(1)} \triangleq \sum_{n_1=h_1}^{\infty} n_1 (f_{1,n_1}(\bar{1})/f_1(\bar{1}))$ *and* $f_1^{(2)} \triangleq \sum_{n_1=h_1}^{\infty} n_1(n_1-1)(f_{1,n_1}(\bar{1})/f_1(\bar{1}))$.

Proof: is given Appendix A. #

Notice that the definitions of $f_1^{(1)}$ and $f_1^{(2)}$ are different than what they were in chapter 2. In this chapter, we call them as *"partial" moments* of queue lengths at polling instants. It can be easily verified that for $h_i = 1, i = 1, \ldots, N$, they are equal to the factorial moments of queue lengths at polling instants, defined in the previous chapter.

Remark 3.1 Other continuously roving polling models with exhaustive service regime can be viewed as special cases of our model. By setting $h_i = 0, i = 1, \ldots, N$, the (empty) summation terms disappear in equation (3.6), and we obtain the mean waiting time expression for the (basic) polling model with non-zero switchover times

(see e.g., Eisenberg 1972). Similarly, by setting all the threshold levels to one, $h_i = 1$, $i = 1, \ldots, N$, we get $F_{i,1} = f_i(0)$, where $f_i(0)$ is defined as the empty station polling probability in chapter 2. Furthermore, $f_1^{(1)}$ and $f_1^{(2)}$ become equal to their counterparts in the state-dependent setup times model. Replacing $F_{1,1}$ by $f_1(0)$ in equation (3.6) leads to equivalent relationship in continuously roving server polling models with state-dependent setups (see Gupta and Srinivasan 1996).

Remark 3.2 Let the polling system consist of a single station. Then our analysis provides similar results to the M/G/1 queue with setup times operating under the N-policy (see e.g., Takagi, 1991, p. 135). Note that, if the polling system consists of a single station, there is no switching, i.e., $R = 0$. Furthermore, the queue length at polling instants is always equal to the threshold level, h. Thus, its PGF is $F(z) = z^h$, and the probability of not having a setup is $F_h = 0$ (the subscript for station index is omitted, since there is only one station in the system). Taking derivatives of $F(z)$ twice at $z = 1$ and substituting them and the no setup probability into equation (3.6), we get

$$E[W] = \frac{\Lambda E[S^2]}{2(1-\rho)} + \frac{h(h-1) + 2h\Lambda E[T] + \Lambda^2 E[T^2]}{2\Lambda(h + \Lambda E[T])}. \tag{3.7}$$

Accounting for differences in notation, equation (3.7) is the same as equation (2.36b) of Takagi (1991), which gives the mean waiting time in a $M/G/1$ queue operating under the N-policy.

From Theorem 3.1, it is clear that the mean station waiting time depends on $F_1(z, 1, \ldots, 1)$ through the latter's (partial) moments, i.e., $f_1^{(1)}$ and $f_1^{(2)}$. Hence, we devote the next section to calculating the distribution of queue lengths at polling instants.

3.3 Queue Lengths at Polling Instants

In this section we modify the DFT-based "near-exact" algorithm developed in the previous chapter to calculate the queue length distribution at station i polling instants, from which we can obtain $f_1^{(1)}$ and $f_1^{(2)}$ numerically. Calculating the mean of waiting times via Theorem 3.1 is then simplified. Before describing the algorithm, let us derive the functional relationship between the PGF of queue lengths at polling instants.

A polling instant always succeeds a switch point after a delay equal to the switchover time from that station. The customers in the system at a station i polling instant are those who are in the system at the switch point from station $i-1$ plus the arrivals during the switchover period, R_{i-1}. Since switch points and station completion epochs are identical, we can write the PGF of queue lengths at a station i polling instant as:

$$f_i(\bar{z}) = g_{i-1}(\bar{z}) R_{i-1}^* \left(\sum_{j=1}^{N} \lambda_j - \lambda_j z_j \right), \tag{3.8}$$

A station i completion epoch occurs either:

i) at the same instant at which the station i is polled (this happens if the server does not find enough customers at station i queue and skips that station); or

ii) after the setup and sojourn times have elapsed following a station i polling instant.

Therefore, the PGF of queue lengths at a station i completion epoch is

$$g_i(\bar{z}) = \left[f_i(z_1, \ldots, B_i^* \left(\sum_{j \neq i} \lambda_j - \lambda_j z_j \right), \ldots, z_N) - \sum_{n_i=0}^{h_i-1} f_{i,n_i}(\bar{z}) B_i^* \left(\sum_{j \neq i} \lambda_j - \lambda_j z_j \right)^{n_i} \right]$$

$$\times U_i^*\left(\sum_{j\neq i}\lambda_j-\lambda_j z_j\right)+\sum_{n_i=0}^{h_i-1} f_{i,n_i}(\bar{z})z_i^{n_i},\quad i=1,\ldots,N, \tag{3.9}$$

where $U_i^*(s)\triangleq T_i^*(s+\lambda_i-\lambda_i B_i^*(s))$ is the LST of the busy period generated by the setup time distribution.

Equations (3.8) and (3.9) define $\overline{\mathcal{M}}_{i-1}\{\cdot\}$, the functional relation between $f_i(\bar{z})$ and $f_{i+1}(\bar{z})$, i.e., $f_{i+1}(\bar{z})=\overline{\mathcal{M}}_i\{f_i(\bar{z})\}$.

Remark 3.3 The set of functional equations, $\overline{\mathcal{M}}_i\{\cdot\}$, are derived without $\mathcal{K}_i$, $i=1,\ldots,N$, the maximum number of service completions in station i per visit, which were used in the previous chapter. Leung (1991) introduced $\mathcal{K}_i$ values so that his probabilistic service regime could include the exhaustive service regime. However, these values are very hard to estimate, and they play a significant role in the implementation of the DFT-based algorithm for the E service regime models. For example, in chapter 2 during the implementation of the near-exact algorithm we do perform successive type-i services until the DFT of queue length distributions at a type-i customer departure instant converges, i.e., station i queue is exhausted. This is done instead of using a rough estimate of $\mathcal{K}_i$, $i=1,\ldots,N$. Taking this difficulty (of estimating $\mathcal{K}_i$ values) into account, Federgruen and Katalan (1993) stated that the DFT-based numeric algorithm was unable to handle polling models with exhaustive service discipline. However, we show that the DFT-based approach can be used for exhaustive service regime models. The details are presented below, after we introduce the DFT equivalents of the PGFs.

Recall that $\hat{f}_i(\tilde{k})$ denotes the DFT of queue lengths at a station i polling

instant, and is defined as

$$\hat{f}_i(\bar{k}) \triangleq \sum_{n_1=0}^{\mathcal{L}_1-1} \cdots \sum_{n_N=0}^{\mathcal{L}_N-1} f^i(\bar{n}) \prod_{j=1}^{N} w_j^{k_j n_j}, \qquad \bar{k} \in \mathcal{S}. \tag{3.10}$$

Similarly, the DFT analog of a partial PGF of queue lengths at a station i, $i = 1, \ldots, N$, polling instant can be calculated as follows:

$$\hat{f}_{i,n_i}(\bar{k}) \triangleq \sum_{n_1=0}^{\mathcal{L}_1-1} \cdots \sum_{n_{i-1}=0}^{\mathcal{L}_{i-1}-1} \sum_{n_{i+1}=0}^{\mathcal{L}_{i+1}-1} \cdots \sum_{n_N=0}^{\mathcal{L}_N-1} f^i(\bar{n}) \prod_{j \neq i} w_j^{k_j n_j}, \; n_i = 0, \ldots, \mathcal{L}_i - 1. \tag{3.11}$$

Notice that, $\hat{f}_{i,n_i}(\bar{k})$ has $N-1$ independent variables. To indicate this, we define the domain set as $\mathcal{S}^i = \{(k_1, \ldots, \mathcal{L}_i, \ldots, k_N) : k_j = 0, \ldots, \mathcal{L}_j - 1, \; j = 1, \ldots, N \text{ and } j \neq i\}$. Therefore, $\mathcal{S}^i$ is the projection of $\mathcal{S}$ into I^{N-1}, where I denotes the set of natural numbers. Finally, by taking the inverse (DFT) of $\hat{f}_i(\bar{k})$ with respect to i^{th} index (see e.g., Poularikas and Seely, 1991) we obtain

$$\hat{f}_{i,n_i}(\bar{k}) = \frac{1}{\mathcal{L}_i} \sum_{l_i=0}^{\mathcal{L}_i-1} \hat{f}_i(\bar{k}) w_i^{-n_i l_i}, \qquad n_i = 0, \ldots, \mathcal{L}_i - 1, \text{ and } i = 1, \ldots, N. \tag{3.12}$$

Recall that $\hat{S}_i^*(\bar{k})$ and $\hat{T}_i^*(\bar{k})$ for $\bar{k} \in \mathcal{S}$ and $i = 1, \ldots, N$, were defined as the DFTs of arrivals during the service time and the setup time in the previous chapter. Now we add the DFTs of arrivals during a switchover time, busy period and the busy period generated by a setup to this list as follows:

$$\hat{R}_i^*(\bar{k}) = R_i^*\left(\sum_{j=1}^{N} \lambda_j - \lambda_j w_j^{k_j}\right), \qquad \bar{k} \in \mathcal{S}, \tag{3.13}$$

$$\hat{B}_i^*(\bar{k}) = B_i^*\left(\sum_{\substack{j=1 \\ j \neq i}}^{N} \lambda_j - \lambda_j w_j^{k_j}\right), \qquad \bar{k} \in \mathcal{S}^i, \tag{3.14}$$

$$\hat{U}_i^*(\bar{k}) = T_i^*\left(\sum_{\substack{j=1 \\ j \neq i}}^{N} \lambda_j - \lambda_j w_j^{k_j} + \lambda_i - \lambda_i \hat{B}_i^*(\bar{k})\right), \quad \bar{k} \in \mathcal{S}^i. \tag{3.15}$$

Replacing $w_i^{k_i n_i}$ by $\hat{B}_i^*(\bar{k})$ in equation (3.10) we obtain the DFT equivalent of $f_i(z_1, \ldots, B_i^*(\cdot), \ldots, z_N)$, which gives the PGF of queue lengths at a station i completion instant in terms of the DFT of the queue lengths at a station i polling instant in the E service regime models (see e.g., Eisenberg 1972):

$$\hat{f}_i(k_1, \ldots, \hat{B}_i^*(\bar{k}), \ldots, k_N) = \sum_{n_1=0}^{\mathcal{L}_1-1} \cdots \sum_{n_N=0}^{\mathcal{L}_i-1} \cdots \sum_{n_N=0}^{\mathcal{L}_N-1} f^i(n_1, \ldots, n_i, \ldots, n_N) \prod_{j \neq i} w_j^{k_j n_j} \hat{B}_i^*(\bar{k})^{n_i}, \quad \bar{k} \in \mathcal{S}^i. \tag{3.16}$$

Equation (3.16) is the key relationship mentioned in Remark 3.3 to be used instead of $\mathcal{K}_i$ successive service completions (i.e., equation (2.23)) in E service regime models. Furthermore, using relations (3.12) and (3.14) in equation (3.16), we obtain

$$\hat{f}_i(k_1, \ldots, \hat{B}_i^*(\bar{k}), \ldots, k_N) = \frac{1}{\mathcal{L}_i} \sum_{n_i=0}^{\mathcal{L}_i-1} \sum_{l_i=0}^{\mathcal{L}_i-1} \hat{f}_i(\bar{k}) w_i^{-l_i n_i} \hat{B}_i^*(\bar{k})^{n_i}, \quad \bar{k} \in \mathcal{S}^i. \tag{3.17}$$

The DFTs of queue lengths at polling instants can be related using $\overline{\mathcal{M}}_i\{\cdot\}$ due to Theorem 2.2. We present the DFT equivalent of equations (3.8) and (3.9) below, in compact form. They are

$$\hat{g}_i(\bar{k}) = \left(\sum_{n_i=0}^{h_i-1} \hat{f}_{i,n_i}(\bar{k}) w_i^{k_i n_i} + \sum_{n_i=h_i}^{\mathcal{L}_i-1} \hat{f}_{i,n_i}(\bar{k}) \hat{B}_i^*(\bar{k})^{n_i} \hat{U}_i^*(\bar{k}) \right), \quad \text{and} \tag{3.18}$$

$$\hat{f}_{i+1}(\bar{k}) = \hat{g}_i(\bar{k}) \hat{R}_i^*(\bar{k}), \tag{3.19}$$

where $\hat{f}_{i,n_i}(\bar{k})$ is given in equation (3.12).

Using the nested transformations of $\overline{\mathcal{M}}_i$, $i = 1, \ldots, N$, we define the mapping $\overline{\Omega}_i(\cdots)$,

$$\overline{\Omega}_i(\cdots) = \mathcal{M}_{i-1}\{\mathcal{M}_{i-2}\{\cdots \mathcal{M}_{i+1}\{\mathcal{M}_i\{\cdot\}\}\cdots\}\}. \tag{3.20}$$

Then using Theorem 2.3 we obtain the DFT of the steady state distribution of queue lengths at polling instants, i.e., $\hat{f}_i(\bar{k})$ as follows:

$$\hat{f}_i(\bar{k}) = \overline{\Omega}_i(\hat{f}_i(\cdot)), \qquad \forall\, \bar{k} \in \mathcal{S}. \tag{3.21}$$

The following algorithm, which is a modified version of the one described in the previous chapter, gives us the distribution of queue lengths at polling instants.

Algorithm–New:

0. Calculate $\hat{B}_i^*(\bar{k})$, $\hat{R}_i^*(\bar{k})$ and $\hat{U}_i^*(\bar{k})$ $\forall\ \bar{k} \in \mathcal{S}$, $i = 1, \ldots, N$. Set the initial state (arbitrarily) to polling station 1 and finding the system empty, i.e., $\hat{f}_1(\bar{k}) = 1$, $\forall\ \bar{k} \in \mathcal{S}$.

1. For $i = 1$ to N do:

   i) calculate $\hat{f}_{i,n_i}(\bar{k})$ for $n_i = 0, \ldots, \mathcal{L}_i - 1$,

   ii) perform $\hat{f}_{i+1}(\bar{k}) = \overline{\mathcal{M}}_i(\hat{f}_i(\bar{k}))\ \ \forall\ \bar{k} \in \mathcal{S}$.

2. If $\hat{f}_1(\bar{k})$ converges $\forall\ \bar{k} \in \mathcal{S}$, then continue. Else go to step (1).

3. For $i = 1$ to N do:

   (i) Invert $\hat{f}_i(0, \ldots, k_i, \ldots, 0)$ to obtain the marginal probabilities of the queue length at station i polling instants, $f^i(n_i)$, $n_i = 0, \ldots, \mathcal{L}_i - 1$.

   (ii) Calculate the factorial (partial) moments of the queue length at station i polling instants.

   (iii) Calculate no setup probabilities, F_{i,h_i}, $i = 1, \ldots, N$. If $(i < N)$, then generate $\hat{f}_{i+1}(\bar{k})$ utilizing the mapping $\overline{\mathcal{M}}_i$.

4. Calculate the expected customer waiting times by substituting the values from step (3) into equation (3.6).

Remark 3.4 In our analysis, we consider the exhaustive service regime at all stations. But the algorithm can also handle a gated service regime. In section 2.6 the necessary changes for the stations with gated service regime was explained for the state-dependent setups model. These modifications work for the threshold setups model, as well.

3.4 Numerical Tests

In this section we present the results of some of the experiments and discuss the performance of the model. Before discussing these experiments, we want to discuss the computational challenges we faced in the implementation phase of the DFT-based algorithm.

3.4.1 Computational Issues

As mentioned in Remark 3.3, we do not need to estimate K_i, $i = 1, \ldots, N$, any more. This is a big improvement for the method. However, we still need to estimate the maximum queue sizes, $\mathcal{L}_i$, $i = 1, \ldots, N$, which are very important for the efficiency of the algorithm. Notice that the computational complexity of this DFT based algorithm is exponential in number of stations, and its space and time requirements are $o(\mathcal{L}^N)$ and $o(\mathcal{L}^{3N} N \log_\rho \epsilon)$ respectively, where $\mathcal{L} = max_{i=1}^{N}\{\mathcal{L}_i\}$ and ϵ is the error tolerance in recursions. Therefore, the algorithm is viable only for small number of stations, N.

By running the algorithm for the same data set for different $\mathcal{L}_i$ values we make the following observations in the queue length distributions (see Figure 3.1):

i) Increasing a single $\mathcal{L}_i$, while keeping others unchanged decreases the tail

probability of type-i queue. But it can increase the tail probability at type-j, $j \neq i$, queue. Note that the *tail probability* (here) refers to $P((\text{Queue length at station}) \; i = \mathcal{L}_i - 1), i = 1, \ldots, N$.

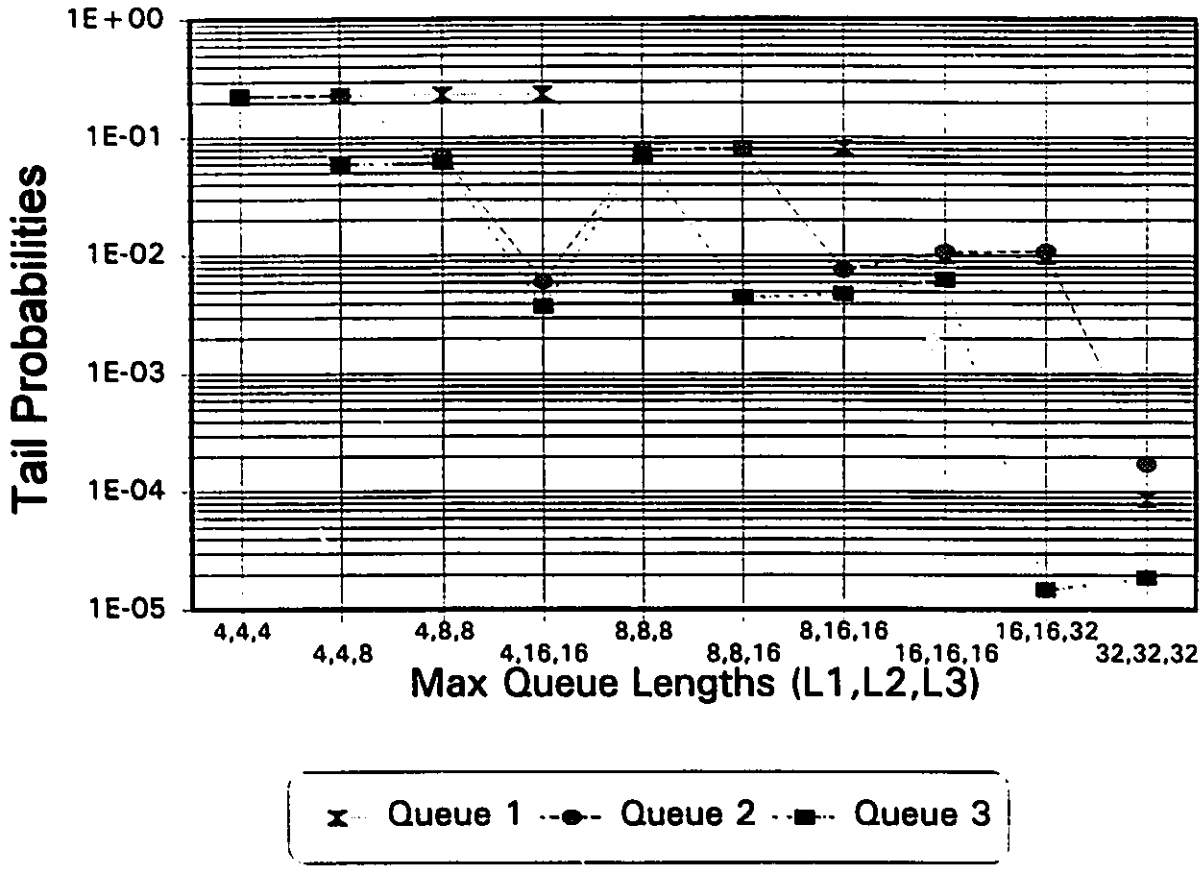
ii) If the tail probability for a station, say i, is less than ϵ (the accuracy level, which was set to 10^{-6} for the experiments), decreasing $\mathcal{L}_i$ until the tail probability becomes equal to ϵ does not increase the tail probabilities at other stations.

iii) Increasing the threshold levels, h_i, $i = 1, \ldots, N$, does not require higher $\mathcal{L}_i$ values to keep the same level of accuracy. This is important, since it implies that once the maximum queue lengths are chosen we can use them to analyze the model with any combination of threshold levels (of course, we assume that $\mathcal{L}_i$ is considerable larger than h_i, for all $i = 1, \ldots, N$).

Therefore, during the experimentation, we use a simple search mechanism to choose these values. We set $\mathcal{L}_i = 8$, $i = 1, \ldots, N$ at the beginning and run the algorithm for the continuously roving model, i.e., $h_i = 0$, $i = 1, \ldots, N$. We keep doubling the $\mathcal{L}_i$ values at each run until the tail probabilities becomes less than $\epsilon = 10^{-6}$. When that condition is satisfied for a queue, we stop increasing $\mathcal{L}_i$ and even decrease to a level such that the tail probability of any queue is just below ϵ. When all the $\mathcal{L}_i$ values are set this way, we start the experiment to find the optimum threshold level vector, $(h_1, \ldots, h_N)$.

Unfortunately, it is very hard to estimate the maximum queue lengths, $\mathcal{L}_i$, $i = 1, \ldots, N$, for the threshold setups model, and there are no known procedure for it so far. In other implementations of DFT based algorithms (see e.g., Leung and Eisenberg, 1990, and Leung, 1991) approximate methods to estimate $\mathcal{L}_i$

Figure 3.1: Comparison of the tail probabilities for a three-station system with parameters: $\lambda_i = 1$, $R_i = 0.1$ constant for all $i = 1, 2, 3$, T_i is exponential with means 1, 2 and 2.5, and B_i is exponential with means 0.2, 0.05 and 0.05, for $i = 1, 2, 3$, respectively. The threshold level vector, $(h_1, h_2, h_3) = (1, 1, 1)$.



values are presented, and also, Federgruen and Katalan (1993) provide an efficient approximation for the queue length distribution of a standard (continuously roving server) polling model. However, all these methods require an approximate solution of the moments of queue lengths or waiting times. Since no known approximation for the threshold setups model is available in literature (to the best of author's knowledge), it is not possible even test these approximation methods to calculate the $\mathcal{L}_i$ values for our model. Furthermore, approximate estimation of $\mathcal{L}_i$ can over estimate as well, and due to the computational complexity of the DFT based algorithm over estimating $\mathcal{L}_i$ values is more costly than under estimating them.

Our naive method of searching for $\mathcal{L}_i$ values works with smaller values than required. Therefore the extra time spend to find the best $\mathcal{L}_i$, $i = 1, \ldots, N$, values might be less costly than using higher values for $\mathcal{L}_i$. This can be seen easily from the computational complexity of the algorithm; at each increase of the $\mathcal{L}_i$ values the extra time spend in the previous run is in the order of $o((1/2)^{3N})$, i.e., if $N = 2$, it is 12.5%, and it can be much less for larger values of N, e.g., $N = 5$. In order to compare the cost of over estimation with the time lost in our method consider the following example: Assume it is a two-station model ($N = 2$), and the best maximum queue lengths are: $\mathcal{L}_1 = \mathcal{L}_2 = 64$. Using our method we solve the problem for $\mathcal{L}_i = 8, 16, 64$. Then, the extra time spend in the first and second runs is $(1/2)^8 + (1/4)^8 = 0.140625$, i.e., 14.06% of the original run ($\mathcal{L}_i = 64$, $i = 1, 2$ case). However, if we over estimate the maximum queue lengths by only two units, the extra time spend due to the over estimation of $\mathcal{L}_i$ values will be in the order of $66^{3N}/64^{3N} - 1$, which is approximately 20.28% of the time spend for the original problem, for $N = 2$. Also, for $N > 2$ the difference will be more drastic. Therefore, we can claim that our naive approach of

finding $\mathcal{L}_i$ values is a good one, because of the following reasons:

- there are no approximate methods to calculate the moments of neither the queue lengths nor the waiting times of threshold setups models;
- over estimating of $\mathcal{L}_i$ values might be too costly for the DFT-based algorithm.

The engine of the algorithm is the DFT routine. It is clear that an accurate DFT routine is a must for our method. We used two different routines in our experiments. One of them is a "Fast Fourier Transform" (FFT) routine from the book *Numerical Recipes in C* (Press *et al.* 1992), and the other one is a naive DFT routine generated by the author. On one hand, FFT routine is very efficient and accurate, but requires the $\mathcal{L}_i$ values to be powers of two. This requirement creates problems in terms of the computational efficiency. On the other hand, the DFT routine used requires $\mathcal{L}_i$ values to be even only. But its accuracy is low. Therefore, in order to use the algorithm efficiently, the DFT routine has to be improved.

3.4.2 Experimentation

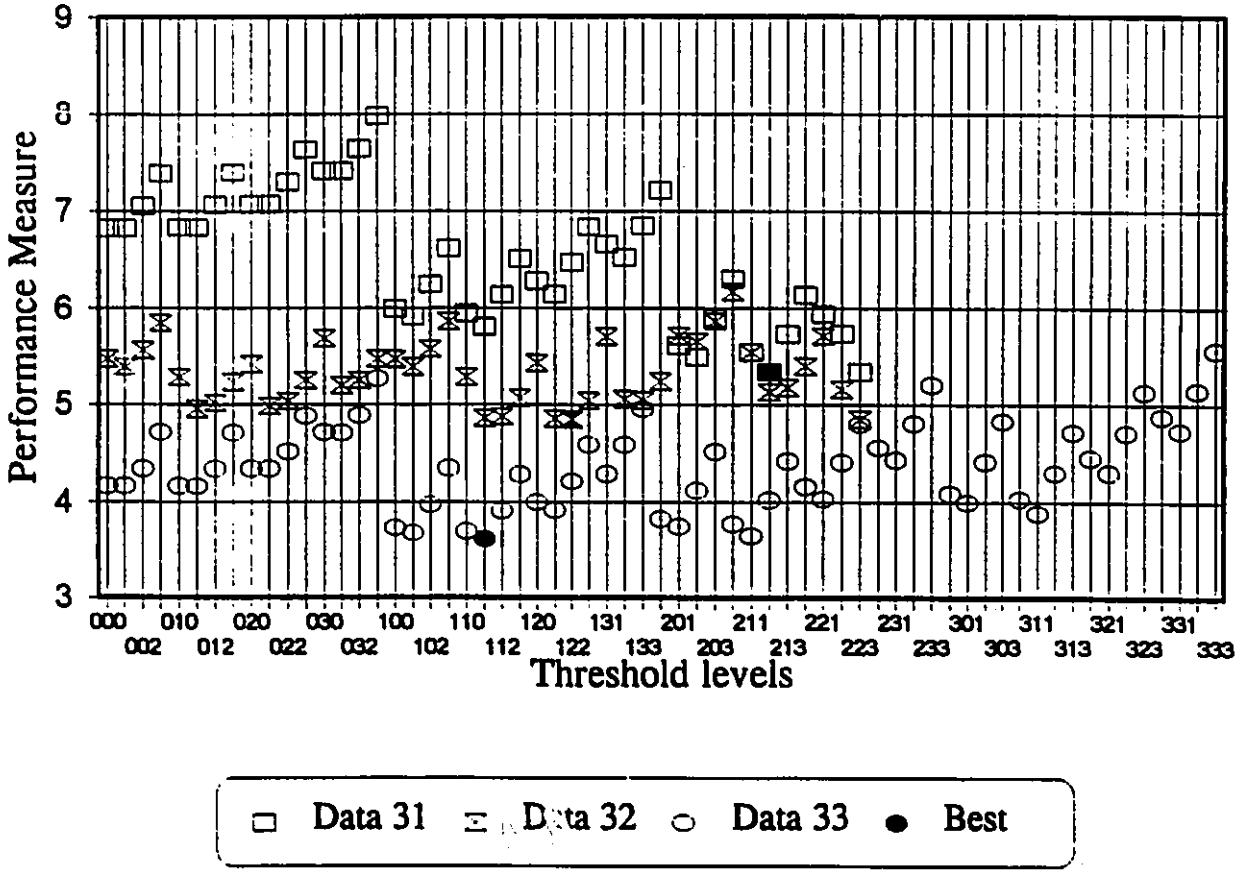
We perform experiments over a wide range of data sets to test our algorithm. We observe that choosing high arrival rates require larger $\mathcal{L}_i$ values. Also smaller setup time parameters cause the optimum threshold levels to be one or even zero, which is intuitive. As mentioned in the previous chapter, choosing $\mathcal{L}_i$, $i = 1, \ldots, N$ effectively is very important for the algorithm. Larger $\mathcal{L}_i$ values increase the precision in the calculation, but decrease the computational performance (both in terms of memory requirement and CPU time). However, we observe that the effective $\mathcal{L}_i$ values are very sensitive to the system parameters, e.g., if the arrival rates are relatively small

with respect to the mean service and setup, but not the switchover time, we can choose small $\mathcal{L}_i$ values, as well. This makes sense because for such a system the expected arrivals during a cycle is small. Therefore, we conclude that the near-exact algorithm is more suitable for systems with relatively smaller arrival rates, and use such numeric examples in our experiments.

We report results of three data sets, that require reasonable computer resources and have non-trivial optimum threshold level vector, $(h_1, \ldots, h_N)$. The three-station examples — data sets 31, 32 and 33 — have the following parameters in common: i) the arrival process is Poisson with $\lambda_1 = 0.1$ and $\lambda_2 = \lambda_3 = 0.05$; ii) service time distributions are exponential with rate 1; iii) switchover times are constant with mean 0.5. However, we vary the setup times. Data set 31 has exponential setup times with means $E[T_1] = 20$, and $E[T_j] = 1$, $j = 2, 3$. In data set 32 setup times are constant with the same mean values as in data set 31. Data set 33 also has exponential setup times too, but with means $E[T_1] = 1$, and $E[T_j] = 20$, $j = 2, 3$. The calculated mean unfinished work in system values for data sets 31, 32 and 33 with threshold levels $(h_1, h_2, h_3) \in \{(0,0,0), (0,0,1), \cdots, (3,3,3)\}$ are plotted in Figure 3.2.

Intuitively we expect a direct relationship between the optimum threshold level and the mean setup time of a station. Comparison of data sets 31 and 33 reveals more than that and shows that the best threshold level also depends on the work load of a station. Furthermore the distribution of setup times affects the optimum threshold levels (see the curve for data 32).

Figure 3.2: The performance of Data sets 31, 32 and 33 for different threshold level vectors, (h_1, h_2, h_3).



3.5 Extensions

In this section, we present the necessary changes to the functional (mapping) equation set, $\overline{\mathcal{M}}_i$, $i = 1, \ldots, N$, to accommodate the patient server models. As Hofri and Ross (1987) have stated, for such models the optimum stopping rule is not known. We assume an "N-level threshold" control policy and present the necessary changes in the algorithm to calculate the mean waiting times for that system. In a patient server polling system with N-level threshold control, the server keeps roving as long as there exists a station in the system whose queue has more customers than its threshold level, h_i, $i = 1, \ldots, N$. Therefore, the patient server polling model with state-dependent setup times (which is described in chapter 2 for a model with zero switchover times) is a special case of this model with all thresholds set to one, i.e., $h_i = 1$, $i = 1, \ldots, N$. When the server stops at a station, it stays dormant until the queue of any station accumulates enough customers to setup, and it immediately starts moving towards that station. Alternatively, an arrival may occur at the same station where the server is idling, in which case service at that station resumes.

Let $\mathcal{A}$ denote the set of all stopping states at a station completion epoch, i.e.,

$$\mathcal{A} = \{\bar{n} \in I_+^N : n_j < h_j,\ j = 1, \ldots, N\}, \tag{3.22}$$

where I_+ is the set of natural numbers, and $\mathcal{A}_i$ denotes the set of stopping states observed at a station i completion epoch, i.e., $\mathcal{A}_i = \{\bar{n} \in \mathcal{S} : n_i = 0\}$, $i = 1, \ldots, N$. Assume the server is idle at station i, $i = 1, \ldots, N$, and let U_i and U_i^o denote the states for the server to start-up by serving customers at station i, and by switching from station i, respectively. By the definition of the "N-level threshold" policy, while the server is idle at station i any arrival to station i re-starts the service at that

station: or arrivals to another station must reach to its threshold level, h_j, $j \neq i$, to re-activate the server. That is,

$$U_i = \{\bar{n} \in I_+^N : n_j < h_j,\ \ j = 1, \ldots, N,\ \ j \neq i,\ \text{and } n_i = 1\}, \qquad \text{and,} \tag{3.23}$$

$$U_i^o = \{\bar{n} \in I_+^N : \exists k \neq i,\ \text{s.t. } n_k = h_k,\ \ n_i = 0,\ \text{and } n_j < h_j\ \ j \neq k, i\}. \tag{3.24}$$

Let $p(i, \bar{m}, \bar{n})$ be the transition probability from the stopping state $\bar{m} \in \mathcal{A}_i$ to a start-up state $\bar{n} \in U_i \cup U_i^o$. Since all arrival processes are Poisson, we can write the transition probability as:

$$p(i, \bar{m}, \bar{n}) = \frac{\left(\sum_{j=1}^N n_j - m_j\right)!}{(n_1 - m_1)! \cdots (n_N - m_N)!} \prod_{j=1}^N p_j^{n_j - m_j}, \qquad \bar{m} \in \mathcal{A}_i,\ \ \bar{n} \in U_i \cup U_i^o. \tag{3.25}$$

Note that, the server departure epochs and switch points are no longer identical. Thus, let $l_i(\bar{z})$ denote the PGF of queue lengths at a switch point from station i, and the functional equation set for the patient server model, $\mathcal{P}_i\{\cdot\}$, is defined as follows:

$$f_{i+1}(\bar{z}) = l_i(\bar{z}) R_i^*\left(\sum_{j=1}^N \lambda_j - \lambda_j z_j\right), \tag{3.26}$$

$$l_i(\bar{z}) = g_i(\bar{z}) - \sum_{\bar{n} \in S_i} g^i(\bar{n}) J_i(\bar{z}) \prod_{j=1}^N z_j^{n_j} + \sum_{\bar{m} \in \mathcal{A}_i} \sum_{\bar{n} \in U_i^o} g^i(\bar{m}) p(i, \bar{m}, \bar{n}) \prod_{\substack{j=1 \\ j \neq i}}^N z_j^{n_j}, \tag{3.27}$$

$$\begin{aligned} g_i(\bar{z}) = {} & \left[f_i(z_1, \ldots, B_{i-1}^*\Big(\sum_{j \neq (i-1)} \lambda_j - \lambda_j z_j\Big), \ldots, z_N) - \sum_{n_i=0}^{h_i - 1} f_{i,n_i}(\bar{z}) \right. \\ & \left. \times B_i^*\Big(\sum_{j \neq (i-1)} \lambda_j - \lambda_j z_j\Big)^{n_i} \right] U_i^*\left(\sum_{j=1}^N \lambda_j - \lambda_j z_j\right) + \sum_{n_i=0}^{h_i - 1} f_{i,n_i}(\bar{z}) z_i^{n_i} \\ & + \sum_{\bar{m} \in \mathcal{A}_i} \sum_{\bar{n} \in U_i} g^i(\bar{m}) p(i, \bar{m}, \bar{n}) \prod_{\substack{j=1 \\ j \neq i}}^N z_j^{n_j} B_i^*\left(\sum_{j \neq i}^N \lambda_j - \lambda_j z_j\right). \end{aligned} \tag{3.28}$$

By first converting the functional equation sets, $\mathcal{P}_i\{\cdot\}$, $i = 1, \ldots, N$, to their DFT analogs and then using them in the infinite recursion (i.e., step 1 and step

3(iii)) of Algorithm-New, the queue length distribution at polling instants for the patient server model can be obtained. We also need to update the mean waiting time expression for the patient server model. Using similar arguments which lead to Theorem 3.1, we get the following expression for the mean waiting time at station 1:

$$E[W_1] = \frac{f_1^{(2)} + 2\lambda_1 \bar{l}_1 f_1^{(1)} + (1 - F_{1,h_1})\lambda_1^2 l_1^{(2)}}{2\lambda_1 (f_1^{(1)} + (1 - F_{1,h_1})\lambda_1 \bar{l}_1 + \sum_{m \in A_1} (g_1(\bar{m})/f_1(\bar{1}))p_1)} + \frac{\lambda_1 E[S_1^2]}{2(1-\rho_1)}. \quad (3.29)$$

Chapter 4

Threshold Start-Up Control Policy for Polling Systems

A threshold start-up policy is appealing for manufacturing (service) facilities that incur a cost for keeping the machine (server) on, as well as for each restart of the server from its dormant state. Analysis of single product (customer) systems operating under such a policy, also known as the N policy, has been available for some time. In this chapter we develop a mathematical analysis for multiproduct systems operating under a cyclic exhaustive or globally gated service regime and a threshold start-up rule. It pays particular attention to modeling switchover (setup) times. The analysis extends/unifies the existing literature on *polling models* by obtaining as special cases, the continuously roving server and patient server polling models and the standard $M/G/1$ queue with N-policy. We provide a computationally efficient algorithm for finding aggregate performance measures: the mean waiting time for each customer type and the mean unfinished work in system. We also show that the search for the optimal threshold level can be restricted to a finite set of possibilities.

The plan of this chapter is as follows. In section 4.1 we present a brief

literature review and the motivation of the problem. We discuss details of the model and notation in section 4.2. The analysis needed to find the distribution of queue lengths at polling instants is presented in section 4.3. Section 4.4 contains explicit expressions for the mean station waiting times and the pseudo-conservation law for both E and GG service regimes. Numerical experiments and insights are presented in section 4.5. This chapter concludes with a procedure to incorporate the G service discipline and a rule for determining the optimum order of visitation for the GG service discipline. Both these are presented in section 4.6 as possible extensions of our analysis. Also, the reader must note that in this chapter M is used to denote the number of customer classes and N is used to represent the threshold level.

4.1 Introduction

Under the threshold start-up control policy, once the server becomes dormant (that happens whenever the system is empty), it would restart only when the number of new arrivals to the system reaches a critical value. A threshold start-up control regime is relevant when there are costs, such as wages and energy costs, that are paid only when the server is available, but not when it is dormant. The threshold is simply 1, if there are no additional start-up costs. However, in general, there might also be fixed start-up costs, such as a power surge, requirement of additional personnel, or a one-time setup. In that case, it makes sense to choose the threshold carefully to realize an optimum balance between start-up and waiting costs. For single-customer-class systems with a removable server this threshold start-up policy is widely known as the N-policy and it has been shown to be optimal with certain cost assumptions

(Heyman and Sobel 1984, Vol. II, Section 7-2). Our aim is to calculate the mean waiting time of customers in multiproduct manufacturing systems with switchover (setup) times and a given start-up threshold of N.

In addition to start-up, shut-down, and cycling rules, a polling model must also specify how many jobs of any one type are processed after each new setup. In this chapter, we analyze the E and GG service policies in detail. The analysis of systems with a G service regime, or with some E and some G stations, is similar, and it is discussed only briefly. E and G service regimes combined with a threshold start-up server control can mimic the operation of many different production, telecommunication, and service systems, where the system control is decentralized to stations. On the other hand, the GG service discipline is a centralized control mechanism and might be suitable for modeling a travelling repair crew and automated guided vehicle (AGV) systems.

There is a large body of literature dealing with polling systems in which the server never stops, i.e., $N = 0$. Such models have lately been labeled continuously roving server models and they are reviewed extensively by Takagi (1986, 1990 and 1994). If the server stops whenever the system is empty and restarts as soon as a new customer arrives, i.e., if threshold $N = 1$, we obtain a special instance of our problem. For the E service regime, this model has been studied in two recent articles: Eisenberg (1995), and Srinivasan and Gupta (1996). This special case has been called the *stopping server* regime by Eisenberg (1995), and the *patient server* regime by Srinivasan and Gupta (1996). Similar special cases, i.e., $N = 0$ and $N = 1$, of the model with GG service discipline were studied by Boxma *et al.* (1993) and Borst (1995), respectively. Our work can be seen as a generalization of these recent studies

to any arbitrary threshold level. We provide a new unification of literature on polling models: by setting $N = 0$, our model and the entire analysis reduces to the standard continuously roving server model, and by setting it equal to 1, a similar match occurs with the patient server model. Similarly, upon setting the number of customer classes, $M = 1$, and accounting for some model differences, we obtain the mean waiting time under the well known N-policy for $M/G/1$ queues. Such transparent relationships build bridges, where none existed before, and provide new insights, similar in spirit to the recent work of Srinivasan, Niu and Cooper (1995).

The analysis contained in this chapter is based on the following plan. We first use the descendant sets method (Konheim *et al.*, 1994, and Srinivasan and Gupta, 1996), both for the E and the GG service regimes, to find the distribution of queue length at a selected station at its polling instant For the E service regime, we then apply Fuhrmann-Cooper decomposition (Fuhrmann and Cooper, 1985) to convert this queue length distribution into the queue length distribution at a service completion epoch, from which the mean station waiting times can be readily obtained. For the GG service regime, we use the polling instant queue length distribution to find the distribution of the length of a cycle, which is the time spent by the server to complete a tour of all stations. Next, we use mean value arguments, similar to Boxma *et al.* (1993) and Borst (1995), to obtain the mean station waiting time which depends on the first two moments of the cycle time. Finally, in both cases, we also find the weighted sum of mean waiting times (or the pseudo-conservation law).

For the E service discipline, we use the simplicity of a symmetric model to find a critical threshold level, $\bar{N}$, beyond which the system performance measure (pseudo-conservation law) never improves over what we can obtain by setting $N = 0$.

We can also show that this result holds even when instead of using the pseudo-conservation law, which incorporates a weight equal to station load for each station, we use arbitrary weights (holding costs). The system performance with $N = 0$ threshold is called the *base* performance level. Although the asymmetric case is more complex, and therefore an explicit expression for $\bar{N}$ difficult to find, we can both argue its existence and obtain an explicit upper bound. The GG service regime leads to simple enough expressions that we find an explicit expression for $\bar{N}$ even for asymmetric models. Thus, in each case, the optimum threshold can be obtained by searching in a finite interval, $[0, \bar{N}]$.

We include several examples to illustrate the effect of system parameters on $\bar{N}$, and the optimal threshold N^*. Since we use the same data sets, we are also able to compare the effectiveness of E and GG strategies, in a limited fashion. For the E service regime, N^* and the system performance measure are quite sensitive to the mean arrival rate and the variance of the switchover times. Performance under the GG service regime is relatively insensitive to changes in N. When the threshold is set at its optimum value in each case, the E policy always outperforms the GG policy in terms of the pseudo-conservation law; our overall system performance measure. However, as Figures 4.1 and 4.2 demonstrate, if we choose a high threshold level, performance under the E policy can deteriorate rapidly and this can easily make it much worse than the GG policy.

Our numerical analysis serendipitously revealed another interesting fact. Performance under the GG policy is very sensitive to the order in which queues are visited. In fact, the effect of order of visitation appears to be far greater than the effect of threshold N. This is in direct contrast with the E policy for which the

order of visitation has a very small impact on overall system performance. Seeing its importance for design of GG controlled systems, we have obtained a simple sequencing rule which minimizes the pseudo-conservation law for a given set of system parameters. The optimal sequence is independent of N.

4.2 Model Description and Notation

Note that, in this chapter we denote the total number of customer types by M, and N represents the threshold level. There are few more notational changes, which we are going to mention in this section. We denote the work load at a station by $\rho_i = \lambda_i E[B_i]$, the total system load by $\rho = \sum_{i=1}^{M} \rho_i$, the overall arrival rate by $\Lambda = \sum_{i=1}^{M} \lambda_i$, and the probability that an arbitrary arrival is type-i by $p_i = \lambda_i/\Lambda$. Similarly, the time to switch from station i to station $i+1$ is denoted by R_i, the total switchover time in a cycle by $R_T = \sum_{i=1}^{M} R_i$, service time of type-i customers by B_i, and the busy period generated by a type i service by Θ_i.

Some notational conventions used in this chapter (different than the previous chapters) are as follows. For a random variable A, we use $A(t)$, $\tilde{A}(s)$, $E[A]$ and $E[A^2]$ to denote the cumulative distribution function, the Laplace-Stieltjes transform (LST), the mean, and the second moment, respectively. If A is discrete then $A(z) \triangleq E[z^A]$ denotes its PGF. Single and double prime notation is used to denote, respectively, the first and second derivatives with respect to z. The notation $\bar{n}$ (or $\bar{z}$) represents a $1 \times M$ vector of n_i's (or z_i's). Parameter N is used in parenthesis to denote the dependence of a performance measure on the threshold level. However, this notation is suppressed wherever such dependence is obvious from the context.

Under service regime E, the server checks the status of queues at all stations whenever it finishes its work at any station (thus, ready to switch to the next station); this instant is called a *server-departure epoch*. If all queues are empty at such an epoch, the server becomes idle and remains at that station. It lies in this dormant mode until the system is populated by exactly N customers. Service always resumes from the station where the server had stopped. Therefore, the server switches at the end of an idle period only if none of the N arrivals occur to the station where it is waiting.

If the system is not empty at a server-departure epoch, the server immediately departs from that station, registering a *switch point*. After passage of an appropriate switchover time, the server arrives at the next station (ready to serve customers waiting at that station), and this instant is called a *polling instant*. Note that the restart of service at station i following an idle period at that station is not a polling instant. Furthermore, a *station beginning* instant is defined as the instant when the server is ready to serve customers at a queue. Thus, a station beginning instant is either a polling instant or an instant at which the server is reactivated following an idle period at that station. We use $f_i(n_1,\ldots,n_M)$ to denote the joint probability that the server polls station-i and finds n_j customers at station j, $j = 1,\ldots,M$. The corresponding probability generating function (PGF) is $f_i(z_1,\ldots,z_M)$. Similarly, $k_i(z_1,\ldots,z_M)$, $g_i(z_1,\ldots,z_M)$ and $h_i(z_1,\ldots,z_M)$ denote the PGF of the joint probability of queue lengths at a type-i station beginning epoch, server-departure instant and switch point, respectively. Furthermore, we use uppercase letters in PGF's to indicate that the event is conditioned on the station type, e.g., $F_i(\bar{z})$ is the PGF of queue lengths given that it is a station i polling instant,

and it is calculated as $F_i(\bar{z}) = f_i(\bar{z})/f_i(\bar{1})$, $i = 1, \ldots, M$.

The various concepts introduced in the previous two paragraphs apply also to the GG service model with some minor modifications, which we shall discuss next. Under the GG service regime, the server checks the status of queues only when it arrives at the *home base*, which we choose to be station 1, WLOG. Thus, at a station 1 polling (system polling) instant the server becomes dormant if it finds the whole system empty, and stays dormant until a total of N customers accumulate in the system. The N^{th} arrival re-activates the server and it starts cycling immediately. In the cycle that follows an idle period exactly N customers are served. But, as in the E service discipline, this re-start instant at station 1 is not marked as another system polling instant. If the system is not empty at the system polling instant, then the server processes only those customers that are already in the system at the polling instant. Let $F(z_1, \ldots, z_M)$ denote the PGF of queue lengths at a system (station 1) polling instant. Notice that unlike the E service discipline, we do not have a partial PGF, or $f(\cdot)$. Also, we do not need a subscript to denote the station index since the system is polled only at station 1.

For both E and GG disciplines, the server stops only when it finds the system empty at the appropriate observation epoch. However, depending on the threshold level, the number of *start-up states* can vary. Let $U(N)$ be the set of all states with exactly N customers in the system. Then,

$$U(N) = \left\{(n_1, \ldots, n_M) \in I_+^M : n_1 + \cdots + n_M = N\right\}. \tag{4.1}$$

We also define $U_i(N)$, a proper subset of $U(N)$, to be the set of states with n_i greater

than zero, i.e.,

$$U_i(N) = \{(n_1, \ldots, n_M) \in U(N) : n_i > 0\}, \qquad i = 1, \ldots, M. \tag{4.2}$$

Finally, $U_i^c(N) \triangleq U(N) \setminus U_i(N)$ represents the complement of $U_i(N)$.

4.3 Queue Length Distributions at Polling Instants

Let the *reference point* be an arbitrary polling instant of station 1, the time period between two successive polling instants of station 1 be a *cycle*, and Q_1 be the station 1 queue length at the reference point. Then, the "contribution" to Q_1 from a test customer C is a subset of Q_1 comprising of the *off-springs* of C. The complete set of off-springs of a customer consists of itself, any customers that arrive during its service time, and all their offsprings. Let $L_{i,c}$ denote the contribution of a type-i customer that is served c cycles prior to the reference point, and let $L_{i,c}(z)$ represent its PGF. Similarly, let $R_{i,c}(z)$ denote the PGF of the contribution of arrivals during a switchover period from station i to $i+1$, c cycles prior to the reference point. These PGF's depend on the service discipline, and procedures for calculating them are presented separately for each service regime.

4.3.1 Exhaustive Service Regime

We demonstrate our procedure, WLOG, for station 1. Similar results for other stations can be obtained simply by rotating the station index. The PGF's $L_{i,c}(z)$

and $R_{i,c}(z)$ can be calculated recursively as follows:

$$L_{i,c}(z) = \tilde{\Theta}_i \left(\sum_{j=i+1}^{M} [\lambda_j - \lambda_j L_{j,c}(z)] + \sum_{j=1}^{i-1} [\lambda_j - \lambda_j L_{j,c-1}(z)] \right), \quad i = 1, \ldots, M, \quad c \geq 0, \tag{4.3}$$

and

$$R_{i,c}(z) = \tilde{R}_i \left(\sum_{j=i+1}^{M} [\lambda_j - \lambda_j L_{j,c}(z)] + \sum_{j=1}^{i} [\lambda_j - \lambda_j L_{j,c-1}(z)] \right), \quad i = 1, \ldots, M, \quad c \geq 0. \tag{4.4}$$

The boundary conditions are as follows: $L_{1,-1}(z) = z$ and $L_{i,-1}(z) = 1$, for all $i > 1$. Let $p(\bar{n})$ represent the probability of observing state $\bar{n} \in U(N)$ at a start-up instant, and $u_{i,c}(\bar{n}, z)$ denote the PGF of the total contribution to Q_1 from this state when the server is at station i, c cycles prior to the reference point. Then we have

$$p(\bar{n}) = \frac{N!}{n_1! \cdots n_M!} \prod_{j=1}^{M} p_j^{n_j}, \qquad \text{and} \tag{4.5}$$

$$u_{i,c}(\bar{n}, z) = \prod_{j=i}^{M} L_{j,c}(z)^{n_j} \prod_{j=1}^{i-1} L_{j,c-1}(z)^{n_j}, \qquad i = 1, \ldots, M, \quad c \geq 0. \tag{4.6}$$

We are now ready to state the main result of this section as Theorem 4.1.

Theorem 4.1 *For an N-threshold start-up control polling system with E service discipline, the PGF of station 1 queue length at an arbitrary polling instant of that station is given by*

$$f_1(z, 1, \ldots, 1) = \Phi \left[\prod_{c=0}^{\infty} \prod_{i=1}^{M} R_{i,c}(z) - \sum_{c=0}^{\infty} \sum_{i=1}^{M} \vartheta_i J_{i,c}(z) E_{i,c}(z) \right], \tag{4.7}$$

where the following definitions have been used:

$$\Phi = f_i(\bar{1}), \tag{4.8}$$

$$\vartheta_i = g_i(\bar{0})/\Phi, \qquad i = 1, \ldots, M, \tag{4.9}$$

$$J_{i,c}(z) = 1 - \sum_{\bar{n} \in U(N)} p(\bar{n}) u_{i,c}(\bar{n}, z), \quad i = 1, \ldots, M, \ c \geq 0, \ \text{and} \tag{4.10}$$

$$E_{i,c}(z) = \prod_{j=i}^{M} R_{j,c}(z) \prod_{l=0}^{c-1} \prod_{k=1}^{M} R_{k,l}(z), \quad i = 1 \ldots, M, \ c \geq 0. \tag{4.11}$$

Proof is presented in Appendix B. #

Notice that by setting $N = 1$ in equation (4.10), we obtain the same definition for $J_{i,c}(z)$ in chapter 2 (see equation 2.17). The scaled empty system probabilities ϑ_i, $i = 1, \ldots, M$, and the constant Φ can be calculated by following a method of solving M linear equations in as many unknowns, shown in Srinivasan and Gupta (1996). These linear equations have the following form:

$$\sum_{i=1}^{M} a_i^{(j)} \vartheta_i = b^{(j)}, \quad j = 1, \ldots, M, \tag{4.12}$$

where

$$a_i^{(j)} = \sum_{c=0}^{\infty} J_{i,c}(\bar{0}) E_{i,c}(\bar{0}), \quad i = 1, \ldots, M, \quad \text{and} \tag{4.13}$$

$$b^{(j)} = \prod_{c=0}^{\infty} \prod_{i=1}^{M} R_{i,c}(\bar{0}). \tag{4.14}$$

The superscript j indicates that the reference point is a station j polling instant, and thus the starting point of the recursion for $L_{i,c}(\bar{0})$ terms is the index $(j-1)$ for any $c \geq 0$. $L_{i,c}(\bar{z})$ denotes the PGF of the joint contribution to *all* queues Q_k, $k = 1, \ldots, M$, by a type-i customer served c cycles prior to the reference point. Similarly $R_{i,c}(\bar{z})$, $i = 1, \ldots, M$, $c \geq 0$, is defined as the total joint contribution to all queues from a station i to $i+1$ switchover period, c cycles prior to the reference point. Then, setting $\bar{z} = \bar{0}$ and using the fact that the system is never empty at

a polling instant, i.e., $f_i(\bar{0}) = 0$, $i = 1, \ldots, M$, we obtain equation (4.12). After evaluating ϑ_i, we obtain Φ by summing up equation (4.8) for all i, simplifying its RHS from relationships (B.10) and (B.11) of Appendix B, and using the normalization $\sum_{i=1}^{M} g_i(\bar{1}) = 1$. This yields

$$\Phi = \frac{1}{M + \sum_{i=1}^{M} \vartheta_i (1 - (1 - p_i)^N)}. \tag{4.15}$$

Thus, we have evaluated the PGF of queue lengths at polling instants in systems with E service discipline.

4.3.2 Globally Gated Service Regime

It is clear from the description of the GG model in section 4.2 that arrivals during cycle $c \geq 0$ are served in cycle $c - 1$, unless they arrive during the idle period. Therefore, PGFs of contributions of arrivals during a service time and the sum of switchover periods per cycle can be written as follows:

$$L_{i,c}(z) = \tilde{B}_i \left(\sum_{j=1}^{M} [\lambda_j - \lambda_j L_{j,c-1}(z)] \right), \quad i = 1, \ldots, M, \quad c \geq 0, \tag{4.16}$$

and

$$R_{T,c}(z) = \tilde{R}_T \left(\sum_{j=1}^{M} [\lambda_j - \lambda_j L_{j,c-1}(z)] \right), \qquad c \geq 0. \tag{4.17}$$

Notice that in the GG model, we are concerned only with the PGF of the total contributions of arrivals during the sum of all switchover times. This happens because arrivals during switchover times incurred c cycles prior to the reference point are always served in the cycle indexed $c - 1$. The boundary conditions are the same as in the E service model, i.e., $L_{1,-1}(z) = z$ and $L_{j,-1} = 1$, for all $j > 1$.

By analogy to section 4.3.1, the PGF of contributions of all customers in the system at a system polling instant c cycles prior to the reference point is defined as

follows:

$$F_c(z) \triangleq F(L_{1,c}(z),\ldots,L_{M,c}(z)). \quad i=1,\ldots,M, \quad c \geq -1. \tag{4.18}$$

The total contribution of the system (to Q_1) at a polling instant c cycles prior to the reference point is equal to the contribution $c+1$ cycles ago, plus the contribution from all those customers that arrive during switchover periods of that cycle. If the system is empty at the previous polling instant and the server restarts with the system in state $\bar{n} \in U(N)$, then the PGF of the system's contribution to Q_1 is simply $u_{1,c}(\bar{n},z)$. Putting it all together, we get

$$F_c(z) = (F_{c+1}(z) - F(\bar{0}))R_{T,c+1}(z) + F(\bar{0}) \sum_{\bar{n}\in U(N)} p(\bar{n}) u_{1,c+1}(\bar{n},z) R_{T,c+1}(z), \qquad c \geq -1. \tag{4.19}$$

Setting $c=-1$ in equation (4.19) and then writing $F_{-1}(z)$ in terms of $F_0(z)$, $F_0(z)$ in terms of $F_1(z)$ and so on, we obtain an infinite recursion. Using arguments similar to the E service discipline models, we simplify this expression and obtain Theorem 4.2.

Theorem 4.2 *For an N-threshold start-up control polling system with GG service regime, the PGF of station 1 queue length at an arbitrary system polling instant is given as*

$$F(z,1,\ldots,1) = \left[\prod_{c=0}^{\infty} R_{T,c}(z) - F(\bar{0}) \sum_{c=0}^{\infty} J_{1,c}(z) \prod_{m=0}^{c} R_{T,m}(z)\right], \tag{4.20}$$

where $J_{1,c}(z)$ is defined as in equation (4.10), and $F(\bar{0})$ is the empty system probability at a polling instant.

The empty system probability can be calculated in the same manner as the E service model. However, here the emptiness of the system matters only when it

occurs at a system polling instant. We denote this probability by $F(\bar{0})$. It is obtained by first writing equation (4.20) for $F(\bar{z})$, and then setting $\bar{z} = \bar{0}$ to yield

$$F(\bar{0}) = \frac{\prod_{c=0}^{\infty} R_{T,c}(\bar{0})}{1 + \sum_{c=0}^{\infty} J_{1,c}(\bar{0}) \prod_{m=0}^{c} R_{T,m}(\bar{0})}. \tag{4.21}$$

4.4 Mean Waiting Times

We use queue length distributions at polling instants to calculate the mean waiting times. As stated in section 3.2 we can only obtain the mean waiting time for the general ($N > 1$) threshold start-up models using the Fuhrmann-Cooper decomposition. In the following sections we present expressions for mean waiting times and the pseudo-conservation law for both E and GG service models.

4.4.1 Exhaustive Service Regime

Recall that, $\Pi_1(z, 1, \ldots, 1)$ denote the PGF for the stationary distribution of station 1 queue length at an arbitrary instant (see Remark 2 in Fuhrmann and Cooper 1985). If we treat each server departure instant from station 1 as the start of a server vacation, the following station 1 beginning instant as the end of that vacation, and apply the Stochastic Decomposition Theorem (see Proposition 2 in Fuhrmann and Cooper 1985) for a $M/G/1$ queueing system with generalized vacations, we obtain

$$\Pi_1(z, 1, \ldots, 1) = \frac{k_1(\bar{1}) - k_1(z, 1, \ldots, 1)}{k_1'(\bar{1})} \times \frac{(1 - \rho_1)\tilde{B}_1(\lambda_1 - \lambda_1 z)}{\tilde{B}_1(\lambda_1 - \lambda_1 z) - z}. \tag{4.22}$$

Differentiating equation (4.22) with respect to z and then setting $z = 1$, we get the average queue length at station 1 at an arbitrary observation instant. Then, using Little's Law the mean waiting time of type-1 customers can be calculated as

follows:

$$E[W_1] = \frac{k_1''(\bar{1})}{2\lambda_1 k_1'(\bar{1})} + \frac{\lambda_1 E[B_1^2]}{2(1-\rho_1)}. \tag{4.23}$$

Recall that a station beginning instant is either a polling instant or an instant at which the server is reactivated following an idle period at the same station. Therefore, the PGF of the queue length at an arbitrary station 1 beginning instant is

$$k_1(z,1,\ldots,1) = f_1(z,1,\ldots,1) + g_1(\bar{0}) \sum_{\bar{n}\in U_1(N)} p(\bar{n}) z^{n_1}. \tag{4.24}$$

Substituting from Theorem 4.1, we can simplify $k_1(z,1,\ldots,1)$ as follows:

$$k_1(z,1,\ldots,1) = \Phi\left[\prod_{c=0}^{\infty}\prod_{i=1}^{M} R_{i,c}(z) - \sum_{c=0}^{\infty}\sum_{i=1}^{M} \vartheta_i J_{i,c}(z) E_{i,c}(z) \vartheta_1 \sum_{\bar{n}\in U_1(N)} p(\bar{n}) u_{1,-1}(\bar{n},z)\right], \tag{4.25}$$

where $u_{1,-1}(\bar{n},z) = z^{n_1}$ for $\bar{n} \in U(N)$.

Next, defining $\gamma_{i,c} = (\lambda_i/\lambda_1) L_{i,c}'(1)$, $i = 1,\ldots,M$, $c \geq -1$, then differentiating equation (4.25) with respect to z two times, and setting $z = 1$, we obtain

$$k_1'(\bar{1}) = \Phi\lambda_1(1-\rho_1)\left[\frac{\sum_{i=1}^{M} E[R_i] + (N\vartheta_i)/\Lambda}{1-\rho}\right], \quad \text{and} \tag{4.26}$$

$$k_1''(\bar{1}) = \Phi\lambda_1^2\left[\mathrm{VAR}(R_M) + \frac{N(N-1)\vartheta_1}{\Lambda^2} + \sum_{i=1}^{M}\left(\frac{\Gamma_i}{\rho_i^2}\right)\left(\mathrm{VAR} + \lambda_i E[B_i^2]E[C] + \frac{N(N-1)\vartheta_i}{\Lambda^2}\right) + \left(\sum_{i=1}^{M} E[R_i]\left(\frac{1-\rho_1}{1-\rho}\right)\right)^2 + 2\sum_{i=1}^{M}\frac{N\vartheta_i}{\Lambda}\sum_{c=0}^{\infty}\left(\frac{\gamma_{i,c}t_{i,c}}{\rho_i}\right)\right], \tag{4.27}$$

where $t_{i,c} = E_{i,c}'(1)/\lambda_1$, $i = 1,\ldots,M$ and $c \geq 0$. That is

$$t_{i,c} = \sum_{j=i+1}^{M} E[R_{j-1}]\frac{\gamma_{j,c}}{\rho_j} + \sum_{l=0}^{c-1}\sum_{k=1}^{M} E[R_{k-1}]\frac{\gamma_{k,l}}{\rho_k} + E[R_M], \tag{4.28}$$

and

$$\Gamma_i \triangleq \sum_{c=0}^{\infty} \gamma_{i,c}^2. \tag{4.29}$$

In equation (4.27), we have presented a much simplified form of the second factorial moment. Steps involved in the derivation of this expression are shown in Appendix C. Since the mean queue length at a station 1 beginning instant can also be written as $k_1'(\bar{1}) = \Phi\lambda_1(1-\rho_1)E[C]$, we have the following new formula for the average cycle length:

$$E[C] = \frac{\sum_{i=1}^{M} E[R_i] + (N\vartheta_i)/\Lambda}{1-\rho}. \tag{4.30}$$

Finally, substituting equations (4.26), (4.27) and (4.30) into relation (4.23), we obtain Theorem 4.3.

Theorem 4.3 *The mean waiting time of type-1 customers in an N-threshold start-up system with cyclic exhaustive service regime is*

$$\begin{aligned} E[W_1] = {} & \frac{1-\rho_1}{2E[C]}\left[\frac{E[R_T]}{1-\rho}\right]^2 + \frac{\lambda_1 E[B_1^2] + [VAR(R_M) + N(N-1)\vartheta_1/\Lambda^2]/E[C]}{2(1-\rho_1)} \\ & + \sum_{i=1}^{M}\left(\frac{\Gamma_i}{\rho_i^2}\right)\frac{\lambda_i E[B_i^2] + [VAR(R_{i-1}) + N(N-1)\vartheta_i/\Lambda^2]/E[C]}{2(1-\rho_1)} \\ & + \sum_{i=1}^{M}\frac{N\vartheta_i}{\Lambda E[C]}\sum_{c=0}^{\infty}\frac{\gamma_{i,c}t_{i,c}}{\rho_i(1-\rho_1)}. \end{aligned} \tag{4.31}$$

Another way of calculating $k_i''(\bar{1})$, $i = 1, \ldots, M$, is to differentiate functional relations for joint queue lengths (similar to equations (B.6)–(B.9)) twice and to then set $\bar{z} = \bar{1}$. This method requires the solution of M^3 equations in that many unknowns in order to get the M terms of interest (see e.g., Takagi 1986). But the coefficient

matrix of this equation set is sparse and symmetric, and upon carefully manipulating its rows, we are able to relate the weighted sum of the second derivatives of $k_i(\bar{z})$ to their first derivatives. The latter can be written explicitly in terms of the system parameters. By substituting this weighted sum of $k_i''(\bar{1})$'s into the formula for the pseudo conservation law for E service models, we obtain

$$\sum_{j=1}^{M}\rho_j E[W_j] = \frac{\rho}{2(1-\rho)}\sum_{i=1}^{M}(\lambda_i E[B_i^2]+\frac{N(N-1)\vartheta_i/\Lambda^2}{E[C]})+\frac{\rho E[R_T^2]}{2(1-\rho)E[C]}+ \frac{E[R_T]}{2(1-\rho)}\left(\rho^2-\sum_{i=1}^{M}\rho_i^2\right)+\sum_{i=1}^{M}\frac{N\rho_i Y_i}{1-\rho}, \quad (4.32)$$

where Y_i is defined as:

$$Y_i=\sum_{j=1}^{i-1}\frac{E[R_j]}{\Lambda E[C]}\left(\sum_{k=i+1}^{M}\vartheta_k+\sum_{k=1}^{j}\vartheta_k\right)+\sum_{j=i+1}^{M}\frac{E[R_j]}{\Lambda E[C]}\sum_{k=i+1}^{j}\vartheta_k, \quad i=1,\ldots,M. \quad (4.33)$$

Note that the above expression is explicit if the (scaled) empty system probabilities, $\vartheta_i, i=1,\ldots,M$ are calculated from (4.12). In equations (4.31) and (4.32) by putting $N=0$ we get, respectively, the mean waiting time and the pseudo conservation law for the system in which the server never stops (Boxma and Groenendijk, 1987). Similarly, when $N=1$ is substituted, we get the corresponding results for the patient server model (Srinivasan and Gupta, 1996). Note that, Srinivasan and Gupta (1996) define the switchover times differently: in their notation station $i \rightarrow i+1$ switchover time is denoted by R_{i+1}.

4.4.2 Globally Gated Service Regime

Since the server stops upon finding the system empty at a system polling instant, a cycle might contain an idle period. We denote this idle period by the random variable, I. Also, we define the portion of the cycle time in which the server is busy serving

customers or switching from one station to the next as the *busy segment* and denote it by the random variable S. Thus, $C = I + S$.

If the server becomes idle at a polling instant, it remains idle until N customers accumulate in the system. Therefore, I has a N-phase Erlang distribution with probability $F(\bar{0})$, and it is zero with probability $1 - F(\bar{0})$. The mean server idle period per cycle is

$$E[I] = \frac{NF(\bar{0})}{\Lambda}, \tag{4.34}$$

and higher moments can be calculated similarly. Using equation (4.34) and the fact that the expected number of arrivals in a cycle must be served during an average cycle length, we obtain

$$E[C] = \frac{E[R_T] + NF(\bar{0})/\Lambda}{1-\rho}. \tag{4.35}$$

We further classify customers with respect to the server status at their arrival instant. A customer who finds the server idle is *class-I* and a customer who finds the server busy (with service or switching) is a *class-S* customer. Then, the mean waiting time of a type-i customer can be calculated as

$$E[W_i] = E[W_i^I]P_I + E[W_i^S]P_S, \tag{4.36}$$

where P_I and P_S are the probabilities that a customer belongs to class-I and class-S, respectively ($P_I + P_S = 1$). Similarly, $E[W_i^I]$ and $E[W_i^S]$ denote the mean waiting time of a type-i customer which belongs to class-I and class-S, respectively.

A class-I customer finds the system empty upon its arrival. Recall that $F(\bar{0})$ is the probability of finding the system empty at a system polling instant. However, P_I is the probability of finding the system empty at an arbitrary point in time and is

equal to

$$P_I = \frac{NF(\bar{0})}{\Lambda E[C]}. \tag{4.37}$$

Now, we analyze the waiting time distribution of an arbitrary type-i customer with respect to its class. Consider a tagged type-i customer that also belongs to class-I, then its waiting time, W_i^I, is composed of the following periods:

i) the remainder of the idle period, which has the expected length $(N-1)/2\Lambda$,

ii) the delay due to the service and switchover times of station-j, $j = 1, \ldots, i-1$, which has the expected duration $\sum_{j=1}^{i-1}(E[R_j] + [(N-1)/\Lambda]\rho_j)$, and

iii) the delay due to the service times of type-i customers which arrive during the same idle period, but before the tagged customer, which has the expected duration $[(N-1)/2\Lambda]\rho_i$.

Therefore,

$$E[W_i^I] = \sum_{j=1}^{i-1} E[R_j] + \frac{N-1}{2\Lambda}\left(1 + 2\sum_{j=1}^{i-1}\rho_j + \rho_i\right). \tag{4.38}$$

We now explain how components of the right hand side of equation (4.38) are obtained.

Note that the delay in item (i) is not equal to the expected residual life of the idle period. There are two reasons for this: (1) during the first phase of the Erlang, the system is still empty and thus nobody is waiting in the first phase of the idle period, and (2) the customer that triggers the server, i.e., the N^{th} arrival during the idle period, does not observe this portion of the waiting time in station-i. Hence the average contribution of the idle period to mean waiting time of a customer is $(N/2\Lambda)((N-1)/N) = (N-1)/2\Lambda$.

The second term, i.e., the delay due to station-j, for $j < i$, consists of the switchover period from station j and the service times of the type-j customers that are present in the system (with the tagged customer) at the end of the idle period. Since the arrivals are Poisson, the batch size of the type-j customers at the end of the idle period is determined by a Binomial process with success rate p_j. This Binomial process has $(N-1)$ trials, because one of the N customers at the start-up instant is the tagged customer, which we know is type-i. Then the mean delay due to type-j, $j < i$, customers is $E[R_j] + ((N-1)\rho_j/\Lambda)$. However, we expect the tagged customer to arrive in a batch which is larger than an average batch (for details of such length bias in batch arrivals see Burke 1975). Therefore, the mean delay within the type-i batch is given as in (iii).

In case the tagged type-i customer belongs to class-S, the next cycle has no idle period and the waiting time, W_i^S, of such a customer is composed of the following:

i) the residual length of the busy segment in which it arrives, which has the expected length $E[S^2]/2E[S]$,

ii) the total service time of all type-j, $j < i$, arrivals during the same residual length of the busy segment, which has the expected duration $\rho_j E[S^2]/2E[S]$,

iii) the total service time of all type-j, $j \leq i$, (including type-i customers who arrive before the tagged customer in the same cycle) arrivals during the portion of the busy segment that has already elapsed, which has the expected duration $\rho_j E[S^2]/2E[S]$, and

iv) the sum of switchover times from station 1 to station i, which has the expected

duration $\sum_{j=1}^{i-1} E[R_j]$.

Therefore,

$$E[W_i^S] = \sum_{j=1}^{i-1} E[R_j] + \left(1 + 2\sum_{j=1}^{i-1}\rho_j + \rho_i\right)\frac{E[S^2]}{2E[S]}. \quad (4.39)$$

In continuously roving server ($N = 0$) case all customers belong to class-S, and hence the above relationship can also be obtained from equation (26) in Boxma *et al.* (1993). Substituting equations (4.37), (4.38) and (4.39) into (4.36), we get

$$E[W_i] = \sum_{j=1}^{i-1} E[R_j] + \left(1 + 2\sum_{j=1}^{i-1}\rho_j + \rho_i\right)\left(\frac{N(N-1)F(\bar{0})}{2\Lambda^2 E[C]} + \frac{E[S^2]}{2E[C]}\right). \quad (4.40)$$

After calculating the moments of the length of the busy segment, we obtain Theorem 4.4 below.

Theorem 4.4 *The mean waiting time of type-i customers in the globally gated polling system with N-threshold start-up policy is given by*

$$E[W_i] = \sum_{j=1}^{i-1} E[R_j] + \left(1 + 2\sum_{j=1}^{i-1}\rho_j + \rho_i\right)\left(\frac{\Lambda^2 E[R_T^2] + N(N-1)F(\bar{0})}{2\Lambda(1+\rho)(\Lambda E[R_T] + NF(\bar{0}))} + \frac{\sum_{j=1}^{M}\lambda_j E[B_j^2] + 2\rho E[R_T]}{2(1-\rho^2)}\right), \qquad i = 1, \ldots, M. \quad (4.41)$$

Proof: is given in Appendix D. #

In equation (4.41) by setting $N = 0$ and $N = 1$ we can obtain results of Boxma *et al.* (1993) for the continuously roving server model, and Borst (1995) for the dormant server model, respectively. Also, using Theorem 4.4 we can derive the pseudo-conservation law for the N-threshold polling systems with globally gated service regime as follows:

$$\sum_{i=1}^{M}\rho_i E[W_i] = \sum_{i=1}^{M}\rho_i\sum_{j=1}^{i-1} E[R_j] + \frac{\rho(\Lambda^2 E[R_T^2] + N(N-1)F(\bar{0}))}{2\Lambda(\Lambda E[R_T] + NF(\bar{0}))} + \frac{\rho\sum_{j=1}^{M}\lambda_j E[B_j^2] + 2\rho^2 E[R_T]}{2(1-\rho)}. \quad (4.42)$$

4.5 Numerical Results

In this section, we set out to calculate the threshold level that minimizes the mean unfinished work in system. Although it is difficult to obtain an explicit expression for the optimum threshold (since empty system probabilities cannot be calculated explicitly) we do manage to show that there exists a (finite) critical threshold level, $\bar{N}$, that bounds the optimum, N^*. Thus, it suffices to enumerate system performance in the interval $[0, \bar{N}]$ only, in order to find the optimum threshold level. Let $\overline{W(N)}$ denote the mean unfinished work in system, also called the pseudo-conservation law. We provide numerical examples to demonstrate the relationship between N and the performance measure, $\overline{W(N)}$. Notice that in this section, we explicitly show the argument N to emphasize dependence on the threshold level.

We construct 7 examples – three for symmetric, and four for asymmetric systems, and examine the optimum threshold level for E and GG service regimes. All examples have 5 stations. Data sets I, II and III correspond to symmetric systems. Data sets I and II have $\Lambda = 0.1$, and exponential service times with mean $E[B] = 0.4$. In data set I, the switchover time has an exponential distribution with $E[R] = 1$, and in set II, the switchover time is either 1, with probability 0.9, or 100, with probability 0.1. Data set III has the same switchover time and service time distribution as set II, but the total arrival rate $\Lambda = 1$.

All asymmetric systems have the following common parameters: $\rho = 0.04$, $\lambda_i = 0.02$, $i = 1, \ldots, 5$. The service time distributions are exponential with station 1 having 75% of the total work load, i.e., $E[B_1] = 1.5$ and $E[B_j] = 0.125$, $j > 1$. For data set IV, R_i is either 1, with probability 0.9, or 100, with probability 0.1, for all

i. For data set V, R_5 is either 20 or 95, with probabilities 0.8 and 0.2, and for data set VI, R_5 is either 64 or 130, with probabilities 0.75 and 0.25, respectively. The R_i, $i < 5$, are identical in data sets IV, V and VI. Notice that $E[R_5]$ equals 10.9, 35 and 80.5 for data IV, V and VI, respectively, and VAR(R_5) remains (almost) unchanged at 880. Data set VII is a copy of data set VI, with the only difference is that R_5 and R_1 have been switched. Thus, while station 1 is still the heavily loaded station, $E[R_1] = 80.5$, and $E[R_j] = 10.9$, $j = 2, 3, 4, 5$.

4.5.1 Exhaustive Service Regime

First, we consider a symmetric system and calculate the critical threshold value, $\tilde{N}$. Then, the critical threshold level in asymmetric systems is discussed. Finally, the results of numerical experiments are presented. Let $R_i = R$, $B_i = B$ and $\lambda_i = \lambda$, $i = 1, \ldots, M$, i.e., suppose we have a symmetric system. Then, $\lambda_i = \Lambda/M$, $\rho_i = \rho/M$ and $p_i = 1/M$. Furthermore, the empty system probabilities, ϑ_i, are the same for all stations, and we denote them by $\vartheta(N)$, $N \geq 0$. Symmetry allows us to greatly simplify the analysis presented in the previous sections. For example, the mean cycle length becomes

$$E[C(N)] = \frac{M(\Lambda E[R] + N\vartheta(N))}{\Lambda(1-\rho)}. \tag{4.43}$$

Since there is only one empty system probability to find, equations (4.12) reduce to a single equation, and the threshold dependent terms, $J_{i,c}(\bar{0})$, $i = 1, \ldots, M$ and $c \geq 0$, can be simplified further. The resulting equation is

$$\vartheta(N) = \frac{\prod_{i=1}^{M} \prod_{c=0}^{\infty} R_{i,c}(\bar{0})}{\sum_{i=1}^{M} \sum_{c=0}^{\infty} (1 - \bar{\eta}_{i,c}^{N}) E_{i,c}(\bar{0})}, \tag{4.44}$$

where $\bar{\eta}_{i,c}$ is the average of the $L_{j,c}(\bar{0})$ terms, that is,

$$\bar{\eta}_{i,c} = (1/M)\left(\sum_{j=i}^{M} L_{j,c}(\bar{0}) + \sum_{j=1}^{i-1} L_{j,c-1}(\bar{0})\right). \tag{4.45}$$

The mean waiting time can now be obtained explicitly as shown below (after simplifications):

$$\begin{aligned} E[W(N)] &= \frac{(N-1+(M-1)\Lambda E[R])N\vartheta(N)+\Lambda^2(E[R^2]+(M-1)E[R]^2)}{2\Lambda(N\vartheta(N)+\Lambda E[R])} \\ &\quad + \frac{\Lambda E[B^2]+\rho(M-1)E[R]}{2(1-\rho)}. \end{aligned} \tag{4.46}$$

Remark 4.1 By setting $M = 1$, we obtain the mean waiting time for the M/G/1 queueing model operating under the N-policy (Heyman and Sobel, 1982, Vol. I, pp. 444-447). Note that for a single station model the switchover time is zero, i.e., $E[R] = 0$, and the system is empty at all server departure epochs, i.e., $\vartheta(N) = 1$. Upon substituting these variables in equation (4.46), we get

$$E[W(N)] = \frac{(N-1)}{2\Lambda} + \frac{\Lambda E[B^2]}{2(1-\rho)}. \tag{4.47}$$

Accounting for differences in notation, equation (4.47) is the same as equation (11-117a) of Heyman and Sobel (1982, Vol. I, p. 445) which gives the mean waiting time in a M/G/1 queue operating under the N-policy. Similarly, upon setting $M = 1$ and $E[R_T] = 0$ in equation (4.41) of the GG service policy, we obtain equation (4.47). This makes sense since the GG service discipline behaves exactly like the E service discipline when $M = 1$; at every server departure instant the server immediately restarts working at the same queue, unless the system is empty.

Liu, Nain and Towsley (1992) have shown that for symmetric systems a patient server ($N = 1$) is superior to the continuously roving server protocol ($N = 0$)

for minimizing unfinished work in system. From the explicit representation of the expected work in system in equation (4.46), it is easy to confirm this result from our analysis as well. What is even more interesting that we can use the $N = 0$ model as a benchmark to find a range of values $1 \leq N \leq \bar{N}$ outside which $\overline{W(N)} > \overline{W(0)}$ (or equivalently $E[W(N)] > E[W(0)]$, because of symmetry).

Proposition 4.1 *For symmetric systems with an E service regime, there exists an* $\bar{N} \geq 1$, *such that* $\overline{W(N)} > \overline{W(0)}$ *for* $N > \bar{N}$, *where*

$$\bar{N} = 1 + \left\lfloor \frac{\Lambda E[R^2]}{E[R]} \right\rfloor. \tag{4.48}$$

Proof: Let $\Delta(N) \triangleq \overline{W(N)} - \overline{W(0)}$. Then using equation (4.46) we obtain,

$$\Delta(N) = \frac{[(N-1)E[R] - \Lambda E[R^2]]N\rho\vartheta(N)}{2\Lambda E[R](N\vartheta(N) + \Lambda E[R])}. \tag{4.49}$$

Since $\vartheta(N) > 0$ for all $N \geq 0$, $\Delta(N) > 0$ if and only if $N > 1 + \Lambda E[R^2]/E[R]$. Thus, setting $\bar{N}$ to be 1 plus the integer floor of $\Lambda E[R^2]/E[R]$ completes the proof. #

Proposition 1 is useful since in order to find the optimum threshold level, N^*, it is now sufficient to enumerate $\overline{W(N)}$ in the interval $[1, \bar{N}]$. Furthermore, we strongly believe that the optimal N is the smallest positive integer for which the mean unfinished work in system increases upon increasing the threshold by 1. There are two lines of reasoning behind this belief. First, in all numerical experiments, $\overline{W(N)}$ has turned out to be convex in the threshold level, N, whenever $0 \leq N \leq \bar{N}$. Secondly, we have been able to prove certain properties of underlying functions that suggest convexity (though we lack formal proof). For example, we have been able to show that $\vartheta(N)$ is convex and strictly decreasing in N, and that $N\vartheta(N)$ is strictly increasing in N. In order to establish convexity of $E[W(N)]$ (and thus of $\overline{W(N)}$, in

the symmetric case), we need to show that $N\vartheta(N)$ is concave in $[0, \bar{N}]$. Although our analysis suggests that $N\vartheta(N)$ should increase in N with a decreasing rate, a formal proof eludes us since that requires an explicit expression for $\vartheta(N)$; a quantity that we can only find numerically.

Like their symmetric counterparts, asymmetric systems also have a critical threshold level, $\bar{N}$, after which increasing the start-up threshold does not improve system performance. Notice that in this case, the optimum threshold can be 0 (see Srinivasan and Gupta 1996 for some examples). Unfortunately, it is difficult to find $\bar{N}$ in an explicit form similar to relationship (4.48). However, as the following proposition proves, there exists an upper bound for the critical threshold, $\bar{N}$.

Proposition 4.2 *For asymmetric systems with an E service regime, there exists an* $\bar{N} \geq 1$, *such that* $\overline{W(N)} > \overline{W(0)}$ *for* $N > \bar{N}$, *where*

$$\bar{N} \leq 1 + \left\lfloor \frac{\Lambda E[R_T^2]}{E[R_T]} \right\rfloor. \tag{4.50}$$

Proof: For the asymmetric model, using equations (4.32) and (4.33) the difference function $\Delta(N)$ can be written as:

$$\Delta(N) = \frac{[(N-1)E[R_T] - \Lambda E[R_T^2]]N\rho\sum_{i=1}^{M}\vartheta_i(N)}{2\Lambda E[R_T](N\sum_{i=1}^{M}\vartheta_i(N) + \Lambda E[R_T])} + \frac{N\sum_{i=1}^{M}\rho_i Y_i}{(1-\rho)}. \tag{4.51}$$

Since $Y_i > 0$ for all $i = 1, \ldots, M$,

$$\Delta(N) > \frac{[(N-1)E[R_T] - \Lambda E[R_T^2]]N\rho\sum_{i=1}^{M}\vartheta_i(N)}{2\Lambda E[R_T](N\sum_{i=1}^{M}\vartheta_i(N) + \Lambda E[R_T])}. \tag{4.52}$$

Thus, $N \geq 1 + \Lambda E[R_T^2]/E[R_T]$ is sufficient to ensure that $\Delta(N) > 0$ (since $\vartheta_i(N) > 0$). Hence proved. #

The optimum threshold levels, N^*, and the critical threshold value, $\bar{N}$, for each symmetric system data set are presented in the table below. Notice that the

Table 4.1: The critical and optimal threshold levels for symmetric systems with the E service regime.

Data-Set	$\bar{N}$	N^*	$E[W(0)]$	$E[W(N^*)]$	$\overline{W(0)}$	$\overline{W(N^*)}$
I	1	1	3.100	2.292	0.124	0.092
II	10	3	68.638	56.148	2.746	2.246
III	92	45	82.513	81.875	33.005	32.750

critical threshold level, $\bar{N}$, increases with increasing variance of switchover time, and with increasing total arrival rate (as seen in equation 4.48). However, at high arrival rates the mean waiting time appears to be robust with respect to the threshold level and small changes in N do not change $E[W(N)]$ (or $\overline{W(N)}$) significantly (see the third row of Table 4.1).

The performance of systems corresponding to data sets IV, V, VI and VII, as measured by the pseudo-conservation law is shown in Figure 4.1. We observe that, asymmetric systems have a finite critical threshold value, $\bar{N}$, and that $\bar{N}$ is influenced a great deal by the total arrival rate and the variance of switchover times. The critical threshold level, $\bar{N}$, and thus N^*, decreases when the mean switchover time to the heavily loaded station is large relative to the mean switchover times to the low traffic stations. We also observe that the right hand side of inequality (4.50) is not affected by this decrease in $\bar{N}$, implying that the upper bound is relatively more loose when both processing and switchover times are not balanced.

4.5.2 Globally Gated Service Regime

Since in the GG service systems there is only one unknown empty system probability, $F(\bar{0})$, the analysis of asymmetric systems is not any more difficult than symmetric ones. Furthermore, even in symmetric GG systems the mean waiting times differ from

Figure 4.1: The performance of asymmetric systems with the E service regime and N-threshold start-ups.

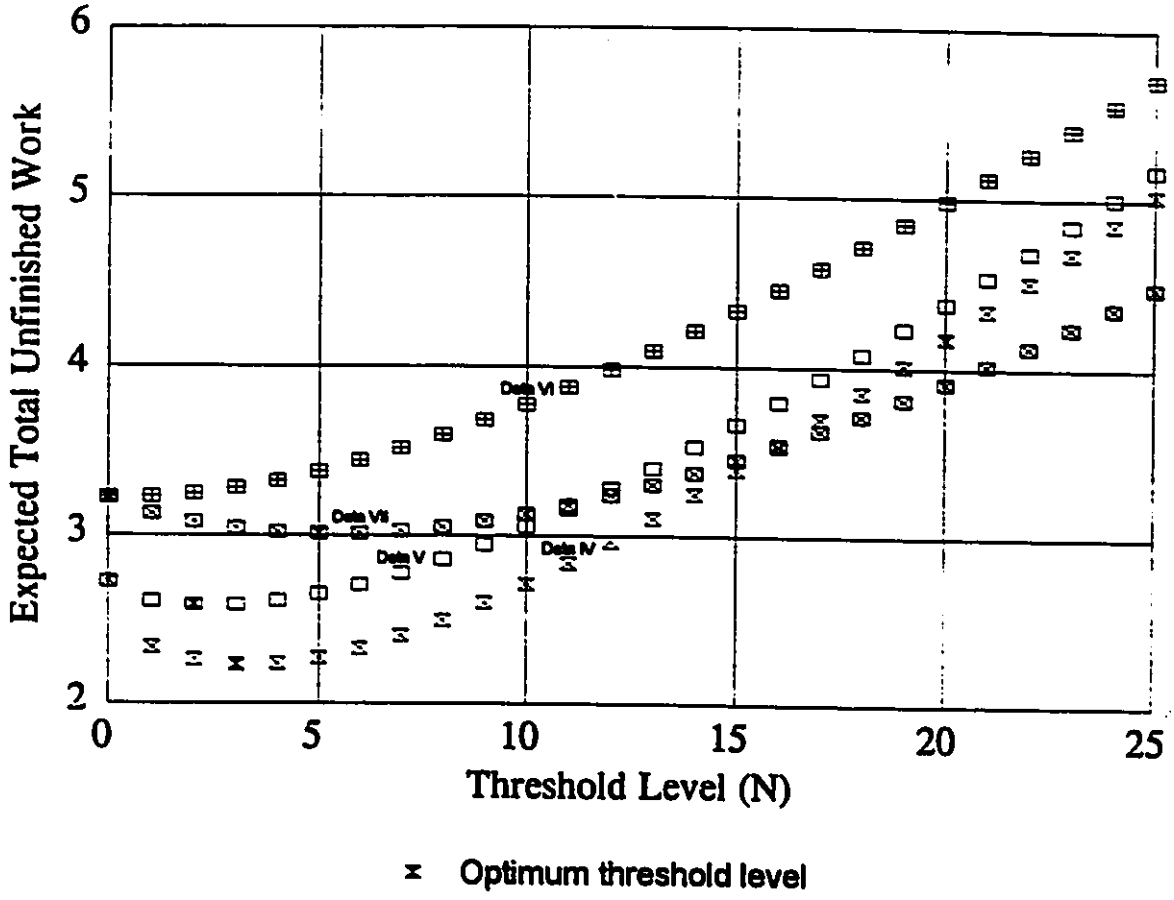


Table 4.2: The critical and optimal threshold levels for symmetric systems with the GG service regime.

Data-Set	$\bar{N}$	N^*	$\overline{W(0)}$	$\overline{W(N^*)}$
I	1	1	0.209	0.144
II	13	5	5.962	5.176
III	138	68	50.445	50.444

one station to the next. They depend significantly on the order of station visitation (sequence). In contrast, station mean waiting times in symmetric E service models are obviously not affected by the order of visitation, and furthermore, the impact of sequencing is small even in asymmetric models.

In a recent study, Borst (1995) showed that for a fixed sequence the dormant server ($N = 1$) model dominates the continuously roving model in the sense of having a smaller mean unfinished work in system. Now, we extend his results to find the optimum threshold value, N^*. Using the $N = 0$ model as a benchmark, Proposition 4.3 presents a range of threshold levels, $[1, \bar{N}]$, that contains the optimum threshold level. We omit the proof of this proposition, since it is similar to the exhaustive service case.

Proposition 4.3 *For GG service systems, there exists an* $\bar{N} \geq 1$, *such that* $\overline{W(N)} > \overline{W(0)}$ *for* $N > \bar{N}$, *where*

$$\bar{N} = 1 + \left\lfloor \frac{\Lambda E[R_T^2]}{E[R_T]} \right\rfloor. \tag{4.53}$$

For symmetric system examples (data sets I, II and III) the optimum threshold level, N^*, and the critical value, $\bar{N}$, are presented in Table 4.2. Performance

of asymmetric systems corresponding to data sets IV, V, VI and VII, is shown in Figure 4.2. Notice that both N^* and $\bar{N}$ values are almost the same for E and GG service regimes. However, when optimum threshold values are used, the mean unfinished work in system for the E service regime is considerably less than that of the GG service model. Comparison of figures 4.1 and 4.2 shows that although E service model performs better than GG service model when thresholds are chosen optimally, this advantage is quickly lost as N becomes large.

Even in asymmetric GG systems, the performance measure is not affected significantly by the threshold level. Furthermore, when switchover times are large, $\overline{W(N)}$ varies even less with N. When switchover times are interchanged (see data VI and data VII), the performance measure is affected significantly by this change even though the optimum threshold does not change much. This unusual effect led us to investigate the impact of the order of visiting stations. We generated all possible sequences for data VI and identified the best and worst sequences for the performance measure, $\overline{W(N)}$. We found that sequence can affect the performance measure significantly and that its effect is independent of N. Figure 4.3 shows the performance measure for the best and worst sequences for the GG service regime. In contrast, the maximum spread between the best and the worst sequence for the E service regime was at most 4.3% for $N \leq 25$. Therefore, the third curve in Figure 4.3, marked "exhaustive," represents $\overline{W(N)}$ that corresponds to the best sequence for each threshold level under the E policy. Note that in this case, the best sequence may be different for different values of N.

Figure 4.2: The performance of asymmetric systems with the GG service regime and N-threshold start-ups.

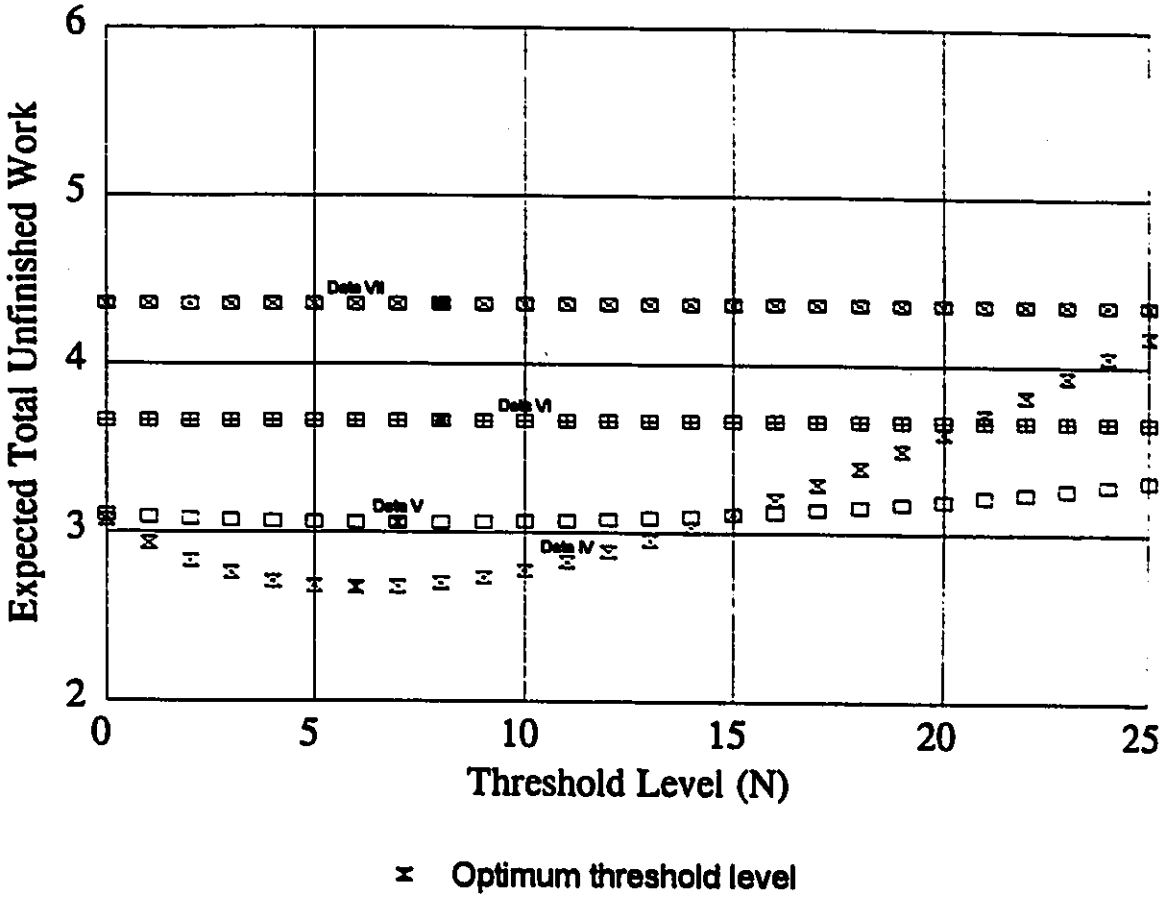
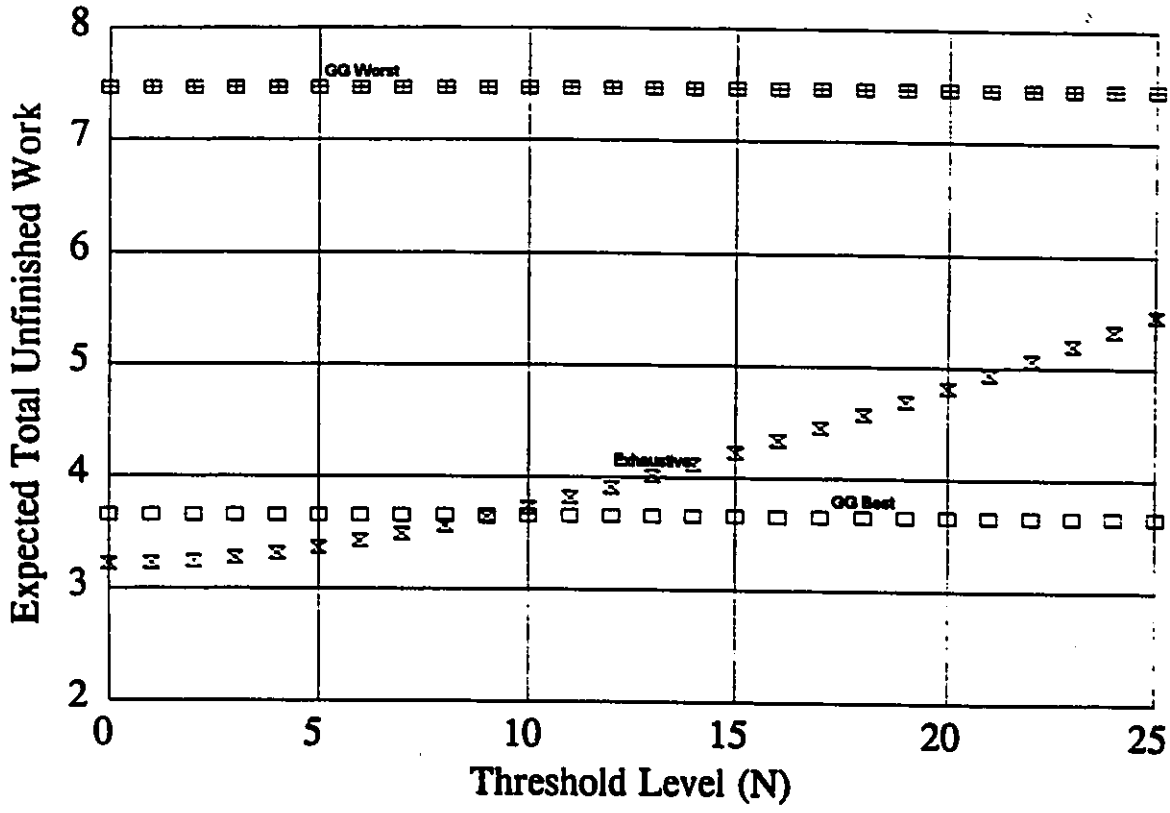


Figure 4.3: The effect of the sequence on the mean unfinished work in system for threshold start-up models.



4.6 Extensions

The DS method can be applied to models with other service disciplines, including the situation in which all stations do not have the same service policy (Konheim *et al.* 1994). In this section, we show what changes would be necessary to include the gated service strategy at select stations. The remaining stations are assumed to be managed by the E policy. According to the G policy, a gate closes behind all customers present at a station at its polling instant, and the server processes only those customers that are in front of the gate during its sojourn time at that station.

Using the "vacation" analogy, each station polling instant is a vacation ending epoch for the G service discipline as well. However, under the G regime the queue is not necessarily empty at the start of a vacation, which is a server departure instant. Therefore, by applying Fuhrmann and Cooper's Stochastic Decomposition Theorem (1985), the PGF of the queue length in station-i at a customer departure instant is obtained as follows:

$$\Pi_i(1,\ldots,z_i,\ldots,1) = \frac{[k_i(1,\ldots,\tilde{B}_i(\lambda_i-\lambda_i z_i),\ldots,1) - k_i(1,\ldots,z_i,\ldots,1)]\tilde{B}_i(\lambda_i-\lambda_i z_i)}{k_i'(\bar{1})[\tilde{B}_i(\lambda_i-\lambda_i z_i) - z_i]}. \tag{4.54}$$

As before, the mean waiting time of type-i customers is derived using the mean queue length of station i, which is obtained from equation (4.54). The result is as follows:

$$E[W_i] = \frac{(1+\rho_i)k_i''(\bar{1})}{2\lambda_i k_i'(\bar{1})}. \tag{4.55}$$

Once again, the main task is to calculate the first and second moments of queue lengths at polling instants. Since the observation epochs, i.e., polling instants, are defined precisely the same way as in the E policy, Theorem 4.1 is still valid and

equation (4.7) gives the PGF of station 1 queue length at its polling instant. However, the PGF of the contribution of a type-i customer who gets service c cycles prior to the reference point, $L_{i,c}(z)$, $c \geq 0$, is now defined as follows:

$$L_{i,c}(z) = \tilde{B}_i \left(\sum_{j=i+1}^{N} [\lambda_j - \lambda_j L_{j,c}(z)] + \sum_{j=1}^{i} [\lambda_j - \lambda_j L_{j,c-1}(z)] \right). \quad (4.56)$$

The moments of queue lengths at polling instants can now be derived using an approach similar to Appendix C.

In order to evaluate the performance of systems with mixed service strategies, i.e., some stations operating under the E and others under the G policy, we first create two groups of stations: $\mathcal{E}$ and $\mathcal{G}$, depending on whether the service regime is E or G, respectively. Next, to calculate the PGF of the contribution of a customer who belongs to set $\mathcal{E}$ or $\mathcal{G}$ either equation (4.3) or equation (4.56) is used. Finally the mean waiting time of type-i customers is obtained using either equation (4.23) or equation (4.55) depending on whether the customer belongs to set $\mathcal{E}$ or $\mathcal{G}$.

The mean unfinished work in system is also affected by the order in which the server visits stations. Our numerical studies show that the E service systems are much less sensitive to the sequence in which stations are arranged. However, the sequence has a large effect on the mean unfinished work in system for GG service models. Therefore, we next show how our analysis can be used to provide an optimal sequence.

We assume that there are no physical or user-defined constraints in regard to ordering of stations in the system and let Ω denote the set of all possible sequences. Then Ω is the set of all possible permutations of the $1 \times M$ vector representing station sequences, and $| \Omega | = M!$. For a sequence, $\omega \in \Omega$, the station index at location i,

for $i = 1, \ldots, M$, is given by $\omega(i) = j$, $j = 1, \ldots, M$. Theorem 4.5, below gives the ordering rule for generating the optimal sequence of stations.

Theorem 4.5 *In a GG service system with N-threshold start-up policy the best sequence, ω^*, that minimizes $\bar{W}(N)$, for $N \geq 0$, satisfies the following condition:*

$$\frac{E[R_{\omega^*(i)}]}{\rho_{\omega^*(i)}} \leq \frac{E[R_{\omega^*(i+1)}]}{\rho_{\omega^*(i+1)}} \qquad i = 1, \ldots, M-1. \tag{4.57}$$

Proof: Given that ω^* satisfies equation (4.57) we want to show that from any sequence $\omega \in \Omega$, and $\omega \neq \omega^*$ we can construct the sequence ω^* by a finite number of interchanges involving neighboring stations, such that at each interchange the objective function, i.e., $\overline{W(N)}$, improves. Since $\omega \neq \omega^*$, there exists at least one pair of stations that does not satisfy equation (4.57). Let $(k, k+1)$ be the first such pair in the sequence, for $k = 1, \ldots, M-1$, i.e.,

$$\frac{E[R_{\omega(k)}]}{\rho_{\omega(k)}} > \frac{E[R_{\omega(k+1)}]}{\rho_{\omega(k+1)}}. \tag{4.58}$$

By switching stations in k^{th} and $(k+1)^{\text{th}}$ place in the sequence, we obtain a new sequence, ω', such that $\omega'(i) = \omega(i)$, $i \neq k, k+1$, and $\omega'(k) = \omega(k+1)$ and $\omega'(k+1) = \omega(k)$. Let $\Delta = \overline{W(N,\omega)} - \overline{W(N,\omega')}$, where $\overline{W(N,\omega)}$ and $\overline{W(N,\omega')}$ denote the objective function of sequences ω and ω', respectively. From equation (4.42) we get

$$\Delta = \rho_{\omega(k+1)} E[R_{\omega(k)}] - \rho_{\omega(k)} E[R_{\omega(k+1)}]. \tag{4.59}$$

From equation (4.58), Δ is clearly positive. Thus, by switching stations in k^{th} and $(k+1)^{\text{th}}$ positions, we improve the objective function. Proceeding with such

interchanges, the number of stations which do not satisfy equation (4.57) decreases, and at each step the objective function improves. The sequence ω^* is reached after a finite number of interchanges. Note that when $E[R_i]/\rho_i = E[R_j]/\rho_j$, for some i and j, ω^* is not a unique sequence. Since the starting sequence, $\omega \in \Omega$, is arbitrary, we have now proved that ω^* is the optimum sequence (Smith 1956). #

Fortunately, the best sequence for the GG service regime does not depend on the threshold level, N. Thus, first finding the best sequence of stations and then searching for the optimum threshold level in the $[1, \bar{N}]$ range will give the minimum mean unfinished work in system.

Chapter 5

Conclusion

In this dissertation we have developed models to study a class of polling models with state-dependent server scheduling rules. Whereas nearly all previous studies assume that the server continuously sets up (switches) irrespective of system state, this behavior is not desirable when setup times are large. Making server movement state-dependent makes the mathematical analysis of our models much more challenging. The following state-dependent server scheduling rules have been considered in this thesis:

(1) non-empty queue setups (chapter 2),

(2) threshold controlled setups (chapter 3), and

(3) threshold controlled start-ups (chapter 4).

Polling systems governed by rules (1) and (2) have been analyzed in previous studies. However, prior to this dissertation, mathematically exact analysis could be performed only for two station systems (Eisenberg, 1971, and Gupta and Srinivasan, 1996). Previous researchers have also identified these problems as hard problems

(Ferguson, 1986, and Hofri and Ross, 1987). In chapters 2 and 3, we present efficient approximation methods for rule (1), and a near-exact algorithm that utilizes the DFT approach (Leung 1991) for rules (1) and (2). The DFT algorithm calculates queue length distributions from which mean waiting times can be readily obtained. Unfortunately, the algorithm is computationally demanding. However, we believe its computational performance can be greatly enhanced by paying careful attention to data structure and by using parallel computing techniques.

Rule (3) is a generalization of the patient server behavior (see Eisenberg, 1995, and Srinivasan and Gupta, 1996). We allow the start-up threshold, N, to be any arbitrary non-negative integer. Thus, by setting $N = 1$, we obtain the patient server model, and by setting it equal to 0, we obtain the continuously roving server model. Another feature of our analysis is that if we make the number of stations $M = 1$, it reduces to the standard $M/G/1$ queue with N-policy for controlling server start-ups. In this way, rule (3) makes it possible to achieve a new unification of literature on queueing models. In chapter 4 we present the analysis for both the E and GG service regimes. The GG service discipline is known to be fairer but less efficient of the two (in terms of minimizing the mean unfinished work in system). Interestingly, we observe that for high threshold levels the GG service regime is superior to E.

Rules (2) and (3) motivate the following question: "what are the optimum setup and start-up thresholds?" Both the near-exact algorithm and the modified DS approach (presented in chapter 4) are tools to compute performance measures when all system parameters (including the threshold levels) have been specified; they are not optimization tools. One possible solution is to use these algorithms to compute system performance for all possible threshold values and to then choose the optimal

values. This can be very costly, especially for rule (2), where there are as many thresholds as the number of stations and thus, the number of different permutations of threshold levels is exponential in the number of stations. For rule (3), we have been successful in limiting the search for an optimal start-up threshold to a finite set, but this set can be large. Therefore, there is a need to develop approximations/heuristics that are amenable to analytical optimization methods. In that case, our algorithms can be used as benchmark against which approximations can be tested. Numerical experiments reported in this thesis also help provide a better understanding of system parameters that affect its performance the most, and in this way, provide the basis for constructing approximations.

Appendix A

Proof of Theorem 3.1

Theorem 3.1

$$E[W_1] = \frac{f_1^{(2)} + 2\lambda_1 \bar{t}_1 f_1^{(1)} + (1 - F_{1,h_1})\lambda_1^2 t_1^{(2)}}{2\lambda_1 (f_1^{(1)} + (1 - F_{1,h_1})\lambda_1 \bar{t}_1)} + \frac{\lambda_1 E[S_1^2]}{2(1-\rho_1)}. \quad \text{(A.1)}$$

Proof: Taking the derivative of equation (3.5) with respect to z, we obtain

$$\begin{aligned}\frac{\partial}{\partial z}\Pi_1(z,1,\ldots,1) = {} & \frac{(1-\rho_1)\left[-(F_1'(z,1,\ldots,1) - \sum_{n_1=0}^{h_1-1} n_1 F_{1,n_1}(\bar{1}) z_1^{n_1-1}) T_1^*(\lambda_1 - \lambda_1 z)\right.}{(F_1'(\bar{1}) - \sum_{n_1=0}^{h_1-1} n_1 F_{1,n_1}(\bar{1}) + (1 - F_{1,h_1})\lambda_1 \bar{t}_1)(1 - z/S_1^*(\lambda_1 - \lambda_1 z))} \\ & + \frac{\left.\lambda_1 T_1'(\lambda_1 - \lambda_1 z)(F_1'(z,1,\ldots,1) - \sum_{n_1=0}^{h_1-1} F_{1,n_1}(\bar{1}) z_1^{n_1})\right]}{(F_1'(\bar{1}) - \sum_{n_1=0}^{h_1-1} n_1 F_{1,n_1}(\bar{1}) + (1 - F_{1,h_1})\lambda_1 \bar{t}_1)(1 - z/S_1^*(\lambda_1 - \lambda_1 z))} \\ & + \frac{(1 - F_{1,h_1}) - (F_1(z,1,\ldots,1) - \sum_{n_1=0}^{h_1-1} F_{1,n_1}(\bar{1}) z_1^{n_1}) T_1^*(\lambda_1 - \lambda_1 z)}{(F_1'(\bar{1}) - \sum_{n_1=0}^{h_1-1} n_1 F_{1,n_1}(\bar{1}) + (1 - F_{1,h_1})\lambda_1 \bar{t}_1)(1 - z/S_1^*(\lambda_1 - \lambda_1 z))^2} \\ & \frac{(1-\rho_1)\left(S_1^*(\lambda_1 - \lambda_1 z) - \lambda_1 z S_1'(\lambda_1 - \lambda_1 z)\right)}{S_1^*(\lambda_1 - \lambda_1 z)^2}. \end{aligned} \quad \text{(A.2)}$$

Setting $z = 1$ in equation (A.2) results in an indefinite expression of $0/0$, and therefore, we need to apply L'Hopitàl's rule. After applying L'Hopitàl's rule twice, we get a definite expression for $z = 1$, and simplifying it leads to

$$\frac{\partial}{\partial z}\Pi_1(z,1,\ldots,1) = \frac{f_1^{(2)} + 2\lambda_1 \bar{t}_1 f_1^{(1)} + (1 - F_{1,h_1})\lambda_1^2 t_1^{(2)}}{2(f_1^{(1)} + (1 - F_{1,h_1})\lambda_1 \bar{t}_1)} + \frac{\lambda_1^2 E[S_1^2]}{2(1-\rho_1)}, \quad \text{(A.3)}$$

where $f_1^{(1)}$ and $f_1^{(2)}$ are defined in Theorem 3.1.

Then using Little's Law the mean waiting time of type-1 customers is obtained by dividing equation (A.3) by the arrival rate of type-1 customers, λ_1. Hence it is proved. #

Appendix B

Proof of Theorem 4.1

Theorem 4.1

$$f_1(z,1,\ldots,1) \;=\; \Phi\left[\prod_{c=0}^{\infty}\prod_{i=1}^{M} R_{i,c}(z) - \sum_{c=0}^{\infty}\sum_{i=1}^{M} \vartheta_i J_{i,c}(z) E_{i,c}(z)\right]. \qquad \text{(B.1)}$$

Proof: The total contribution to Q_1 from all customers present in the system at station beginning instants, polling epochs, server-departure instants and switch points ($c \geq 0$ cycles prior to the reference point) denoted, respectively, as $k_{i,c}(z)$, $f_{i,c}(z)$, $g_{i,c}(z)$, and $h_{i,c}(z)$, $i = 1,\ldots,M$, are defined as follows:

$$k_{i,c}(z) \;\triangleq\; k_i(L_{1,c-1}(z),\ldots,L_{i-1,c-1}(z),L_{i,c}(z),L_{i+1,c}(z),\ldots,L_{M,c}(z)), \qquad \text{(B.2)}$$

$$f_{i,c}(z) \;\triangleq\; f_i(L_{1,c-1}(z),\ldots,L_{i-1,c-1}(z),L_{i,c}(z),L_{i+1,c}(z),\ldots,L_{M,c}(z)), \qquad \text{(B.3)}$$

$$g_{i,c}(z) \;\triangleq\; g_i(L_{1,c-1}(z),\ldots,L_{i-1,c-1}(z),0,L_{i+1,c}(z),\ldots,L_{M,c}(z)), \quad \text{and} \qquad \text{(B.4)}$$

$$h_{i,c}(z) \;\triangleq\; h_i(L_{1,c-1}(z),\ldots,L_{i-1,c-1}(z),0,L_{i+1,c}(z),\ldots,L_{M,c}(z)). \qquad \text{(B.5)}$$

The zero in the argument of the right hand side of equations (B.4) and (B.5) is a notation we have adopted to emphasize that the corresponding queue is empty. At a station i server-departure instant and switch point $c \geq 0$ cycles ago, there are no waiting customers at that station and therefore their contribution to station 1 queue at reference point is zero. Since by stationarity the probability of finding the system empty at a server-departure instant from station i does not depend on the cycle index, i.e., $g_{i,c}(\bar{0}) = g_i(\bar{0})$, $c \geq 0$, the relationship between $h_{i,c}(z)$ and $g_{i,c}(z)$ can be written as follows:

$$h_{i,c}(z) = g_{i,c}(z) - g_i(\bar{0}) + g_i(\bar{0}) \sum_{\bar{n}\in U_i^c(N)} p(\bar{n}) u_{i,c}(\bar{n},z), \;\; i = 1,\ldots,M, \;\; c \geq 0. \qquad \text{(B.6)}$$

Similarly, we can relate $k_{i,c}(z)$ to $f_{i,c}(z)$ and $g_i(\bar{0})$ as follows

$$k_{i,c}(z) = f_{i,c}(z) + g_i(\bar{0}) \sum_{\bar{n} \in U_i(N)} p(\bar{n}) u_{i,c}(\bar{n}, z), \; i = 1, \ldots, M, \; c \geq 0. \tag{B.7}$$

Furthermore, using the definition of polling instants, we obtain

$$f_{i,c}(z) = h_{i-1,c}(z) R_{i-1,c}(z), \qquad i = 2, \ldots, M, \; c \geq 0, \tag{B.8}$$

and

$$f_{1,c}(z) = h_{M,c+1}(z) R_{M,c+1}(z), \qquad c \geq -1. \tag{B.9}$$

By definition, when we set $z = 1$, we obtain $f_{i,c}(1) = f_i(\bar{1})$. Similarly, $g_{i,c}(1) = g_i(\bar{1})$, $k_{i,c}(1) = k_i(\bar{1})$ and $h_{i,c}(1) = h_i(\bar{1})$, $i = 1, \ldots, M$, independent of c. Since each station i beginning instant is followed by a server-departure epoch from that station, we have $k_i(\bar{1}) = g_i(\bar{1})$, $i = 1, \ldots, M$. Using the fact that for each polling instant of station i, there exists a switch point from that station and setting $z = 1$ in equation (B.8), we get

$$f_i(\bar{1}) = h_i(\bar{1}) = h_{i-1}(\bar{1}) = \Phi, \qquad i = 1, \ldots, M, \tag{B.10}$$

i.e., the probability of polling station i is constant for all $i = 1, \ldots, M$. Furthermore, setting $z = 1$ leads to $u_{i,c}(\bar{n}, 1) = 1$ in equations (B.6) and (B.7), and thus

$$h_i(\bar{1}) = g_i(\bar{1}) - g_i(\bar{0})(1 - (1 - p_i)^N), \quad i = 1, \ldots, M, \tag{B.11}$$

and

$$k_i(\bar{1}) = \Phi[1 + \vartheta_i(1 - (1 - p_i)^N)], \qquad i = 1, \ldots, M, \tag{B.12}$$

where ϑ_i, $i = 1, \ldots, M$, is the scaled probability of finding the system empty at a server-departure instant from station i, $i = 1, \ldots, M$ (see also Theorem 4.1).

Realizing that there is no net change in the contribution to Q_1 between a station beginning instance and a server-departure instance at a station, i.e., $k_{i,c}(z) = g_{i,c}(z)$, and utilizing previous relationships, we obtain the basic relationship between the PGF of the overall contribution to Q_1 at station i and station $i+1$ polling instants as

$$f_{i+1,c}(z) = f_{i,c}(z) R_{i,c}(z) - g_i(\bar{0}) J_{i,c}(z) R_{i,c}(z), \tag{B.13}$$

where $J_{i,c}(z)$ is defined in Theorem 4.1. Now, if we set $i = M$ and express $f_{1,c}(z)$ in terms of $f_{M,c+1}(z)$, and then, express $f_{M,c+1}(z)$ in terms of $f_{M-1,c+1}(z)$, and so on; after M such recursions we obtain

$$f_{1,c}(z) = f_{1,c+1}(z) \prod_{i=1}^{M} R_{i,c+1}(z) - \sum_{i=1}^{M} g_i(\bar{0}) J_{i,c+1}(z) \prod_{j=i}^{M} R_{j,c+1}(z). \tag{B.14}$$

If similar recursions are performed over the cycle index, c, starting from $c = -1$ and iterating $m+1$ times, we get

$$f_{1,-1}(z) = f_{1,m}(z) \prod_{c=0}^{m} \prod_{i=1}^{M} R_{i,c}(z) - \sum_{i=1}^{M} g_i(\bar{0}) \sum_{c=0}^{m} J_{i,c}(z) \prod_{j=i}^{M} R_{j,c}(z) \left[\prod_{l=0}^{c-1} \prod_{k=1}^{M} R_{k,l}(z) \right]. \tag{B.15}$$

Next, we let $m \to \infty$ and use the fact that the contribution of a customer who is served c cycles ago reduces to 0 as $c \to \infty$. Thus $f_{1,\infty}(z) = f_1(\bar{1})$, and since $f_{1,-1}(z) = f_1(z, 1, \ldots, 1)$, we obtain Theorem 4.1. #

Appendix C

Calculating Moments of $k_1(z, 1, \ldots, 1)$

Taking the first and second derivative of $k_{1,-1}(z)$ with respect to z, and then setting $z = 1$, we obtain

$$k_1'(\bar{1}) = \Phi\left[\sum_{c=0}^{\infty}\sum_{i=1}^{M} R_{i,c}'(1) - \sum_{c=0}^{\infty}\sum_{i=1}^{M}\vartheta_i J_{i,c}'(1) + \vartheta_1 \sum_{\bar{n}\in U_1(N)} p(\bar{n}) u_{1,-1}'(\bar{n}, 1)\right], \quad \text{(C.1)}$$

$$k_1''(\bar{1}) = \Phi\left[\sum_{c=0}^{\infty}\sum_{i=1}^{M}(R_{i,c}''(1) - R_{i,c}'(1)^2) + \left(\sum_{c=0}^{\infty}\sum_{i=1}^{M} R_{i,c}'(1)\right)^2 - \sum_{c=0}^{\infty}\sum_{i=1}^{M}\vartheta_i[J_{i,c}''(1) + 2J_{i,c}'(1)E_{i,c}'(1)] + \vartheta_1 \sum_{\bar{n}\in U_1(N)} p(\bar{n}) u_{1,-1}''(\bar{n}, 1)\right] \quad \text{(C.2)}$$

Since $f_1(z, 1, \ldots, 1)$ for the N-threshold start-up model is similar to the corresponding PGF in a patient server model, many of the steps involved in finding their factorial moments are common. The main differences come from the $J_{i,c}(z)$, $i = 1, \ldots, M$, $c \geq 0$ terms which now involve N. Therefore, we begin first by stating some results that are derived in Srinivasan and Gupta (1996) and also apply to our model. Later, we shall calculate the derivatives of $J_{i,c}(z)$, i.e., the terms $J_{i,c}'(1)$ and $J_{i,c}''(1)$.

From the definition of $\gamma_{i,c}$ and equation (4.3) we obtain

$$\gamma_{i,c} = \rho_i\left(\sum_{j=i}^{M}\gamma_{j,c} + \sum_{j=1}^{i-1}\gamma_{j,c-1}\right), \qquad i = 1, \ldots, M, \ c \geq 0. \quad \text{(C.3)}$$

Summing these $\gamma_{i,c}$, terms up, we get

$$\sum_{c=-1}^{\infty}\sum_{i=1}^{M}\gamma_{i,c} = \frac{1-\rho_1}{1-\rho}, \quad \text{and} \quad \text{(C.4)}$$

$$\sum_{c=0}^{\infty}\sum_{i=1}^{M}\gamma_{i,c}^{(2)} = \frac{\lambda_1}{1-\rho_1}\sum_{i=1}^{M}\left(\frac{\Gamma_i}{\rho_i^2}\right)\lambda_i E[B_i^2], \tag{C.5}$$

where Γ_i, $i=1,\ldots,M$, are as defined in equation (4.29), and $\gamma_{i,c}=(\lambda_i/\lambda_1)L''_{i,c}(1)$, $i=1,\ldots,M$, $c\geq -1$. Next, using definition (C.3) we obtain

$$\sum_{j=i+1}^{M}\gamma_{j,c}+\sum_{j=1}^{i}\gamma_{j,c-1} = \frac{\gamma_{i+1,c}}{\rho_{i+1}}, \qquad i=1,\ldots,M,\ c\geq 0. \tag{C.6}$$

Differentiating $J_{i,c}(z)$ from equation (4.10), and simplifying yields

$$J'_{i,c}(1)=-\sum_{j=i}^{M}L'_{j,c}(1)\sum_{n\in U(N)}p(\bar{n})n_j-\sum_{j=1}^{i-1}L'_{j,c-1}(1)\sum_{n\in U(N)}p(\bar{n})n_j. \tag{C.7}$$

Similarly,

$$\begin{aligned}
J''_{i,c}(1) = & -\sum_{j=i}^{M}(L''_{j,c}(1)-(L'_{j,c}(1))^2)\sum_{n\in U(N)}p(\bar{n})n_j-\sum_{j=i}^{M}(L'_{j,c}(1))^2\sum_{n\in U(N)}p(\bar{n})n_j^2 \\
& -\sum_{j=1}^{i-1}(L''_{j,c-1}(1)-(L'_{j,c-1}(1))^2)\sum_{n\in U(N)}p(\bar{n})n_j-\sum_{j=1}^{i-1}(L'_{j,c-1}(1))^2\sum_{n\in U(N)}p(\bar{n})n_j^2 \\
& -2\left(\sum_{j=i}^{M}\sum_{k=i}^{j-1}L'_{j,c}(1)L'_{k,c}(1)\sum_{n\in U(N)}p(\bar{n})n_jn_k+\sum_{j=i}^{M}\sum_{k=1}^{i-1}L'_{j,c}(1)L'_{k,c-1}(1)\right. \\
& \left.\cdot\sum_{n\in U(N)}p(\bar{n})n_jn_k+\sum_{j=1}^{i-1}\sum_{k=1}^{j-1}L'_{j,c-1}(1)L'_{k,c-1}(1)\sum_{n\in U(N)}p(\bar{n})n_jn_k\right).
\end{aligned} \tag{C.8}$$

In order to further simplify these equations, we first define the following shorthand for quantities appearing on their right hand side:

$$E[X_j] \triangleq \sum_{n\in U(N)}p(\bar{n})n_j, \tag{C.9}$$

$$E[X_j^2] \triangleq \sum_{n\in U(N)}p(\bar{n})n_j^2, \quad \text{and} \tag{C.10}$$

$$E[X_j,X_k] \triangleq \sum_{n\in U(N)}p(\bar{n})n_jn_k, \qquad j\neq k. \tag{C.11}$$

Notice that the random variable X_j also equals the total number of type-j outcomes in N successive trials of a multinomial distribution with p_j, $j=1,\ldots,M$, success rate for a type-j outcome. Using this equivalence, the above moments can be evaluated

as: $E[X_j] = Np_j$, $E[X_j^2] = N(N-1)p_j^2 + Np_j$ and $E[X_j, X_k] = N(N-1)p_jp_k$, for $j \neq k$ and $j,k = 1,\ldots,M$. We omit these details in the interest of brevity.

Next, after substituting from equations (C.9), (C.10) and (C.11) into equations (C.7) and (C.8), the following expressions are obtained:

$$J'_{i,c}(1) = -\sum_{j=i}^{M} L'_{j,c}(1)E[X_j] - \sum_{j=1}^{i-1} L'_{j,c-1}(1)E[X_j], \quad \text{and} \tag{C.12}$$

$$\begin{aligned} J''_{i,c}(1) = & -\sum_{j=i}^{M}(L''_{j,c}(1) - (L'_{j,c}(1))^2)E[X_j] - \sum_{j=i}^{M}(L'_{j,c}(1))^2E[X_j^2] \\ & -\sum_{j=1}^{i-1}(L''_{j,c-1}(1) - (L'_{j,c-1}(1))^2)E[X_j] - \sum_{j=1}^{i-1}(L'_{j,c-1}(1))^2E[X_j^2] \\ & -2\left(\sum_{j=i}^{M}\sum_{k=i}^{j-1} L'_{j,c}(1)L'_{k,c}(1)E[X_j,X_k] + \sum_{j=i}^{M}\sum_{k=1}^{i-1} L'_{j,c}(1)L'_{k,c-1}(1)E[X_j,X_k]\right. \\ & \left. +\sum_{j=1}^{i-1}\sum_{k=1}^{j-1} L'_{j,c-1}(1)L'_{k,c-1}(1)E[X_j,X_k]\right). \end{aligned} \tag{C.13}$$

Making all the relevant substitutions from above into equation (C.1), we obtain equation (4.26). Similar, substitutions into equation (C.2) give

$$\begin{aligned} k''_1(\bar{1}) = & \Phi\lambda_1^2\left[\sum_{i=1}^{M}\left(\frac{\Gamma_i}{\rho_i^2}\right)\left(\lambda_i E[B_i^2]E[C] + +\mathrm{VAR}(R_{i-1}) + \frac{N(N-1)\vartheta_i}{\Lambda^2}\right) + \mathrm{VAR}(R_M)\right. \\ & \left. +\frac{N(N-1)\vartheta_1}{\Lambda^2} + \left(\frac{(1-\rho_1)E[R_T]}{1-\rho}\right)^2 + 2\sum_{i=1}^{M}\frac{N\vartheta_i}{\Lambda}\sum_{c=0}^{\infty}\left(\frac{\gamma_{i,c}t_{i,c}}{\rho_i}\right)\right]. \end{aligned} \tag{C.14}$$

Appendix D

Proof of Theorem 4.4

Theorem 4.4

$$E[W_i] = \sum_{j=1}^{i-1} E[R_j] + \left(1 + 2\sum_{j=1}^{i-1}\rho_j + \rho_i\right)\left(\frac{\Lambda^2 E[R_T^2] + N(N-1)F(\bar{0})}{2\Lambda(1+\rho)(\Lambda E[R_T] + NF(\bar{0}))} + \frac{\sum_{j=1}^{N}\lambda_j E[B_j^2] + 2\rho E[R_T]}{2(1-\rho^2)}\right), \qquad i = 1, \ldots, M. \tag{D.1}$$

Proof: In order to obtain equation (4.41), we need the first and second moments of the busy segment of the cycle. The distribution of the busy segment can be calculated using the PGF of joint queue lengths at system polling instants. Note that, the queue length of any station at a system polling instant is equal to the number of arrivals to that station during the previous busy segment, S. Therefore,

$$F(1, \ldots, 1, z_i, 1, \ldots, 1) = \tilde{S}(\lambda_i - \lambda_i z_i), \qquad i = 1, \ldots, M. \tag{D.2}$$

First, the moments of station 1 queue length can be obtained by differentiating equation (4.20) with respect to z. After setting $z = 1$ in these derivatives we get

$$F'(\bar{1}) = \sum_{c=0}^{\infty} R'_{T,c}(1) - F(\bar{0})\sum_{c=0}^{\infty} J'_{1,c}(1), \tag{D.3}$$

$$F''(\bar{1}) = \sum_{c=0}^{\infty} R''_{T,c}(1) - R'_{T,c}(1)^2 + \left(\sum_{c=0}^{\infty} R'_{T,c}(1)\right)^2 - F(\bar{0})\left(\sum_{c=0}^{\infty} J''_{1,c}(1) + 2J'_{1,c}(1)\sum_{m=0}^{c} R'_{T,m}(1)\right). \tag{D.4}$$

Using similar arguments to those used in Appendix A, we simplify equations (D.3) and (D.4) as

$$F'(\bar{1}) = \frac{\lambda_1 \left(E[R_T] + \rho(NF(\bar{0})/\Lambda)\right)}{1-\rho}, \tag{D.5}$$

$$F''(\bar{1}) = \lambda_1^2 \left(\frac{E[R_T^2] + \rho^2 N(N-1)F(\bar{0})/\Lambda^2}{1-\rho^2} + \left(\frac{E[R_T] + (NF(\bar{0})/\Lambda)}{1-\rho} \right) \times \left[\frac{\sum_{j=1}^{M} \lambda_j E[B_j^2] + 2\rho E[R_T]}{1-\rho^2} \right] \right). \tag{D.6}$$

Then, the mean and second moment of the busy segment are obtained using equations (D.2), (D.5) and (D.6) as follows:

$$E[S] = \frac{E[R_T] + \rho N F(\bar{0})/\Lambda}{1-\rho}, \tag{D.7}$$

$$E[S^2] = \frac{E[R_T^2] + \rho^2 N(N-1)F(\bar{0})/\Lambda^2 + E[C]\left(\sum_{j=1}^{M} \lambda_j E[B_j^2] + 2\rho E[R_T]\right)}{1-\rho^2}. \tag{D.8}$$

Substituting equation (D.8) into (4.40) completes the proof. #

Remark D.1 If we modify equation (D.2) for the joint queue length distribution of all stations at a polling instant and then, set $\bar{z} = \bar{0}$ we obtain $F(\bar{0}) = \tilde{S}(\Lambda)$, which is used by Borst (1995) to denote the empty system probability.

Bibliography

[1] Altman, E., P. Konstantopoulos and Z. Liu, "Stability, Monotonicity and Invariant Quantities in General Polling Systems," *Queueing Systems*, **11** (1992), 35–57.

[2] Borst, S. C., "A Globally Gated Polling System With A Dormant Server," *Probability in the Engineering and Information Sciences*, **9** (1995), 239–254.

[3] Boxma, O. J., "Efficient Visit Orders for Polling Systems," *Performance Evaluation*, **18** (1993), 103–123.

[4] Boxma O. J., "Analysis and Optimization of Polling Systems," *Queueing, Performance and Control in ATM (ITC-13)*, J.W. Cohen and C.D. Pack (Editors) Elsevier Science Publishers B.V. (North-Holland), 1991.

[5] Boxma, O. J., W. P. Groenendijk and J. A. Weststrate, "A Pseudoconservation Law for Service Systems with a Polling Table," *IEEE Transactions on Communications*, **38** (1990), 1865–1870.

[6] Boxma, O. J. and W. P. Groenendijk, "Pseudo-Conservation Laws In Cyclic-Service Systems," *J. Appl. Prob.*, **24** (1987), 949–964.

[7] Boxma, O. J., H. Levy and U. Yechiali, "A Globally Gated Polling System with Server Interruptions, and Applications to the Repairman Problem," *Probability in the Engineering and Information Sciences*, **7** (1993), 187–208.

[8] Bozer, Y. and M. M. Srinivasan, "Tandem configurations for Automated Guided Vehicle Systems and the Analysis of Single Vehicle Loops," *IIE Transactions*, **23** (1991), 72–82.

[9] Bradlow, H. S. and H. F. Byrd, "Mean Waiting Time Evaluation of Packet Switches for Centrally Controlled PBX's," *Performance Evaluation*, **7** (1987), 309–327.

[10] Browne, S. and U. Yechiali, "Dynamic Priority Rules for Cyclic-Type Queues," *Advances in Applied Probability*, **21** (1989), 432–450.

[11] Burke P. J., "Delays in Single-Server Queues with Batch Input," *Operations Research*, **23**, (1975), 830–833.

[12] Coffman, E. G., Jr., A. A. Puhalskii and M. I. Reiman, "Polling Systems with Zero Switchover Times: A Heavy-Traffic Averaging Principle," *Annals of Applied Probability* **5** (1995), 681–719.

[13] Cooper, R. B., *Introduction to Queueing Theory*, Third Edition, CEEPress Books, 1990.

[14] Cooper, R. B., "Queues Served in Cyclic Order: Waiting Times," *The Bell System Technical Journal*, **48** (1970), 399–413.

[15] Cooper, R. B., S. C. Niu, and M. M. Srinivasan, "A Decomposition Theorem for Polling Models: The Switchover Times are Effectively Additive," *Operations Research*, **44** (1996) .

[16] Cooper, R. B. and G. Murray, "Queues Served in Cyclic Order," *The Bell System Technical Journal*, **19** (1969), 675–689.

[17] Duenyas, I. and M. P. Van Oyen, (1993), "Stochastic Scheduling of Parallel Queues with Set-up Costs," *Queueing Systems*, **19** (1995), 421–444.

[18] Eisenberg M. (1995), "Polling Systems with a Stopping Server," *Queueing Systems*, **18** (1969), 389–431.

[19] Eisenberg M., "Queues with Periodic Service and Changeover Time," *Operations Research*, **20** (1972), 440–451.

[20] Eisenberg M., "Two Queues with Changeover Times," *Operations Research*, **19** (1971), 386–401.

[21] Eisenberg M., "Multi-Queues with Changeover Times," *Technical Report No. 35*, Operations Research Center, M.I.T. 1968.

[22] Everitt D., "Note on the Pseudoconversation Laws for Cyclic Service Systems with Limited Service Disciplines," *IEEE Transactions on Communications*, **37** (1989), 781–783.

[23] Everitt D., "Simple Approximations For Token Rings," *IEEE Transactions on Communications*, **34** (1986), 719–721.

[24] Federgruen, A. and Z. Katalan, "Approximating Queue Size and Waiting Time Distribution in General Polling Systems," *Queueing Systems*, **18** (1994), 353–386.

[25] Federgruen, A. and Z. Katalan, "The Stochastic Economic Lot Scheduling Problem: Cyclical Base-Stock Policies with Idle Times," *Management Science*, **42** (1996), 783–796.

[26] Ferguson, M. J., "Mean Waiting Time for Token Ring with Station Dependent Overheads," *Local Area and Multiple Access Networks*, R.L. Pickholtz (Editor), Computer Science Press, (1986), 43–67.

[27] Ferguson, M. J. and Y. J. Aminetzah, "Exact Results for Nonsymmetric Token Ring Systems," *IEEE Transactions on Communications*, **33** (1985), 223–231.

[28] Fuhrmann S. W., "A Decomposition Result for a Class of Polling Models," *Queueing Systems*, **11** (1992), 109–120.

[29] Fuhrmann, S. W. and R. B. Cooper, "Stochastic Decompositions in the M/G/1 Queue with Generalized Vacations," *Operations Research*, **33** (1985), 1117–1129.

[30] Heyman, D. P. and M. J. Sobel, *Stochastic Models in Operations Research, Volume II: Stochastic Optimization*, McGraw Hill Book Company, New York, 1984.

[31] Heyman, D. P. and M. J. Sobel, *Stochastic Models in Operations Research, Volume I: Stochastic Processes and Operating Characteristics*, McGraw Hill Book Company, New York, 1982.

[32] Günalay, Y. and D. Gupta, " Comparison of Exact and Approximate Algorithms to Compute the Moments of Waiting Times of a Polling System," presented at the *ORSA/TIMS Joint National Meeting* at Phoenix, U.S.A, November 2nd, 1993.

[33] Gupta, D. and M. M. Srinivasan, "Polling Systems with State Dependent Setup Times," *Queueing Systems* to appear (1996).

[34] Hofri, M. and K. W. Ross, "On the Optimal Control of Two Queues with Server Setup Times and Its Analysis," *SIAM Journal on Computing*, **16** (1987), 399–420.

[35] Hofri M., "Two Queues and One Server with Threshold Switching," *Teletraffic Analysis and Computer Performance Evaluation*, O.J. Boxma, J.W. Cohen, H.C.

Tijms (Editors) Elsevier Science Publishers B.V. (North-Holland) (1986), 409–423.

[36] Kleinrock L., *Queueing Systems, Volume I: Theory*, John Wiley & Sons, New York, 1975.

[37] Kleinrock L., *Queueing Systems, Volume II: Applications*, John Wiley & Sons, New York, 1976.

[38] Konheim, A. G., H. Levy and M. M. Srinivasan, "Descendant Set: An Efficient Approach For The Analysis of Polling Systems," *IEEE Transactions on Communications*, **42** (1994), 1245–1253.

[39] Leung K. K., "Cyclic-Service Systems with Probabilistically-Limited Service," *IEEE Journal on Selected Areas in Communications*, **9** (1991), 185–193.

[40] Leung K. K. and M. Eisenberg, "A Single Server Queue with Vacations and Gated Time-Limited Service," *IEEE Transactions on Comm.*, **38** (1990), 1454–1462.

[41] Levy, H., Sidi, M. and Boxma, O., J., "Dominance Relations in Polling Systems," *Queueing Systems*, **6**, (1990), 155–172.

[42] Liu, Z., P. Nain and D. Towsley, "On Optimal Polling Policies," *Queueing Systems*, **11** (1992), 59–83.

[43] Markowitz, D. M., M. I. Reiman and L. M. Wein, "The Stochastic Economic Lot Scheduling Problem: Heavy Traffic Analysis of Dynamic Cyclic Policies," Private Communication from Prof. David M. Markowitz: Operations Research Center, M.I.T., 1995.

[44] Olsen T. L., "Approximations for the Waiting Time Distribution in Polling Models With and Without State-Dependent Setups," Private Communication from Dr. Tava Lenon-Olsen: Dept. of Industrial Engineering, University of Michigan, Ann Arbor, 1996.

[45] Poularikas, A. D. and S. Seely, *Signals and Systems*, PWS-KENT Publishing Com., Boston, 1991.

[46] Press, W. H., S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, *Numerical Recipies in C*, Cambridge University Press, 1992.

[47] Sarkar, D. and W. I. Zangwill, "Variance Effects in Cyclic Production Systems," *Management Science*, **37** (1991), 444–453.

[48] Sarkar, D. and W. I. Zangwill, "Expected Waiting Time for Nonsymmmetric Cyclic Queueing Systems - Exact Results and Applications," *Management Science*, **35** (1989), 1463–1474.

[49] Smith, W. E., "Various Optimizers For Single-Stage Production," *Naval Research Logistics, Quarterly*, **3**, (1956), 59–66.

[50] Srinivasan, M. M. and D. Gupta, "When Should a Roving Server be Patient?," *Management Science*, **42** (1996), 437–451.

[51] Srinivasan, M. M., H. Levy and A. G. Konheim, "The Individual Station Technique for the Analysis of Polling Systems," *Naval Research Logistic*, **43** (1996) 79–101.

[52] Srinivasan, M. M., S. C. Niu and R. B. Cooper, "Relating Polling Models with Zero and Nonzero Switchover Times," *Queueing Systems*, **19** (1995), 149–168.

[53] Sykes J. S., "Simplified Analysis of an Alternating-Priority Queueing Model with Setup Times", *Operations Research*, **18**, No. 6 (1970), 1182–1192.

[54] Takács L., "Two Queues Attended by a Single Server," *Operations Research*, **16**, No. 3 (1968), 639–650.

[55] Takács L., *Introduction to the Theory of Queues*, Oxford, 1962.

[56] Takagi H., *Queueing Analysis of Polling Models: Progress in 1990–93*, Institute of Socio-Economic Planning, University of Tsukuba, Japan, 1994.

[57] Takagi H., *Queueing Analysis, Volume 1: Vacation and Priority Systems, Part 1*, Elsevier Science Publishers B.V. (North Holland), Amsterdam, 1991.

[58] Takagi H., "Queuing Analysis of Polling Models: An Update," *Stochastic Analysis of Computer and Communication Systems*, H. Takagi (Editor), Elsevier Science Publishers B.V. (North-Holland), (1990), 267–318.

[59] Takagi H., "Queueing Analysis of Polling Systems with Zero Switchover Times," Private Communication from Dr. Hidea Takagi, IBM, Tokyo, *TRL Research Report* #87 – 0034, (1987).

[60] Takagi H., *Analysis of Polling Systems*, MIT Press, Cambridge, M.A., 1986.

[61] Watson K. S., "Performance Evaluation of Cyclic Service Strategies - A Survey," *Performance '84*, E. Gelenbe (Editor), Elsevier Science Publishers B.V. (North-Holland), (1984), 521–533.

[62] Wolff R. W., "Poisson Arrivals See Time Averages," *Operations Research*, **30** (1982), 223–231.